Effect of Potential Model Pruning on Official-Sized Board in Monte-Carlo *GO*

Makoto Oshima-So[1†],
m.sou@okiu.ac.jp
Department of Industry and Information Science, Okinawa International University, Okinawa, 901-2701, Japan

Summary
Monte-Carlo *GO* is a computer *GO* program that is sufficiently competent without using knowledge expressions of *IGO*. Although it is computationally intensive, the computational complexity can be reduced by properly pruning the *IGO* game tree. Here, I achieve 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 using the potential model categorizes legal moves into effective and ineffective moves in accordance with the potential threshold. Here, certain pruning strategies based on potentials and potential gradients are experimentally evaluated. For different-sized boards, including an official-sized board, the effects of pruning strategies are evaluated in terms of their robustness. I successfully demonstrate pruning using a potential model to reduce the computational complexity of *GO* as well as the robustness of this effect across different-sized boards.
Key words:
Reducing Computational Complexity, Knowledge Expression, Heuristic, Potential, Geometric Information Systems, Potential Gradient, Official Size Board

1. Introduction

In this study, I reduce computational complexity by pruning an *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*, is 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. Because Monte-Carlo *GO* shows no deviation in the sequence of moves for *IGO*, the effects of the heuristics generated by a potential model are demonstrated correctly.

This study builds upon existing research on potential model pruning in Monte-Carlo *GO* [2, 3]. In the previous experiment [3], the effects of nine potential models on Monte-Carlo *GO* were demonstrated on 9 × 9 and 13 × 13 boards. In this experiment, the nine potential models are demonstrated on 9 × 9, 13 × 13, and 19 × 19 (i.e., official size) boards.

Manuscript received June 5, 2021
Manuscript revised June 20, 2021
https://doi.org/10.22937/IJCSNS.2021.21.6.9

2. Proposed Method

The method proposed herein consists of Monte-Carlo *GO* and a potential model.

2.1 Monte-Carlo *GO*

Monte-Carlo *GO* evaluates legal moves in each phase to select the next move via a simulation based on a Monte-Carlo search process comprising 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.

$$\overline{X_i} = X_i / s_i \tag{1}$$

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.

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, like in previous studies [4, 5].

2.2.1 Definition of Potential

The potential is defined in Eqs. 2–4 and Table 1. Fig. 1 shows a calculation example. The sign of Eq. 3 was switched depending on the setting of the proposed method. The potential distribution on the *GO* board was calculated by these equations. If necessary, the potential gradient was subsequently calculated according to the gradient method by using geographical information systems [6] with the potential distribution. Fig. 2 shows a schematic diagram of this process.

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

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

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

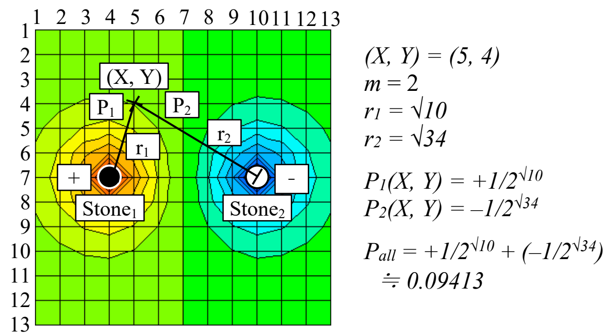


Fig. 1 Example of potential calculation.

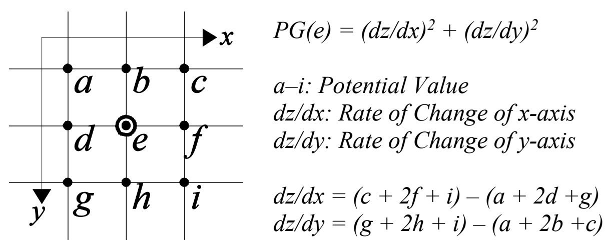


Fig. 2 Schematic of potential gradient.

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
$Pall(X, Y)$	:	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:

(i) Calculate the potential distribution resulting from the arrangement of stones on the board.
(ii) Rank legal moves by each magnitude of potential (or potential gradient).
(iii) Categorize ranked legal moves into effective and ineffective moves according to the thresholds for the ranking. (Each PF has a unique threshold level.)
(iv) 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 was reduced; i.e., PFs reduced the range of search spaces on the board.

Potential Filter Configurations:
Table 2 lists the configurations of the five filters (the Random Filter and PFs 1–4), listing the ranking, attenuation rate of the potential *m*, polar characteristics 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 according to each threshold condition for the ranking. PFs 1–4 are the same as PFs 1–4 in the previous experiment [3].

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 according to each threshold condition for the ranking. In PFs 5–9, *m* critically involved their filtering functions. According to the magnitude of *m*, potential gradient values of intersections surrounding each stone increased. In contrast, the lower *m* became, the higher the potential gradients of intersections between black 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 and white stone, is ranked 77th in order of magnitude of potential gradient. With PF 6, the intersection marked x ranked 55th; with PF 7, 33rd; with PF 8, 11th; and with PF 9, 5th.

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

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 81 when the board size was 9 × 9 (=81), from 2 to 169 when the size was 13 × 13 (=169), or from 2 to 361 when the size was 19 × 19 (=361).

During 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 at which winning percentages exceeded the average winning percentage between two normal Monte-Carlo *GO* programs (57% with a board size of 9 × 9, 51% with 13 × 13, or 50% with 19 × 19).

The performance of the Monte-Carlo search was higher when the game tree was small; it deteriorated as the game tree became larger. Thus, pruning was effective in the opening game. However, pruning gradually became ineffective thereafter as legal moves on the *GO* board decreased.

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 57% with a 9 × 9 board, 51% with 13 × 13, or 50% with 19 × 19 (the winning percentage of the initiative move exceeded 50% because this move was advantageous). Therefore, 57%, 51%, or 50% was considered the average level of normal competence.

4. Results and Observation

Figs. 3 and 4 show the winning percentages of Monte-Carlo *GO* with PFs along the left-hand axis (upper graphs, 9 × 9 board size; middle graphs, 13 × 13 board size; lower graphs, 19 × 19 board size). The level of competence varied with the filter and switching point. The normal winning percentage of 57%, 51%, or 50% and the calculated results of the Random Filter were important for comparing and evaluating the effects and tendencies of the PFs. Figs. 3 and 4 show the number of total Play Out times required for one game for three 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

The Random Filter pruned 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 the areas between black and white stones were closed. The lower the value of m, the stronger the bias and effect of pruning.

The characteristics of each PF were unique. However, they were all capable of properly pruning ineffective moves that the Monte-Carlo search could not when the winning percentage exceeded the average (57%, 51%, or 50%). 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 the upper, middle, and lower graphs in Figs. 3 and 4, if the x-axes were scaled to the same width, each winning percentage curve of PFs 1–9 in these graphs was similar and had 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 indicated that the PF effects were robust to the size of the *GO* board; they depended only on the ratio of the number of legal moves to that of all intersections on the *GO* board.

5. Summary

Here, I reduced computational complexity of Monte-Carlo *GO* by pruning the *IGO* game tree using a potential model based on the knowledge expression of *IGO*. In my experiments, the effects of nine types of pruning strategies (PFs) were evaluated on different-sized boards, including an official-sized board. Each PF had a specific effect on *IGO*, which was maintained on the 9 × 9, 13 × 13, and 19 × 19 boards.

I successfully demonstrated pruning by using the potential model to reduce the computational complexity of *GO*, as well as the robustness of the PF effects to the size of the *GO* board. These results indicated the possibility that knowledge expressions of stones as potentials contribute to the enhancement of existing computer *GO* and the understanding of the essence of *IGO*. However, my experiments were limited as the Play Out number was set to 100. For future research, I intend to expand the proposed strategy to address more complex games with larger Play Out numbers.

References

[1] Brugmann, B.: *Monte Carlo Go*. Technical Report, Physics Department Syracuse University (1993)

[2] Oshima, M., Yamada, K., Endo, S.: *Probability of Potential Model Pruning in Monte-Carlo Go*. Procedia Computer Science, vol. 6, pp. 237–242 (2011)

[3] Oshima, M., Yamada, K., Endo, S.: *Effect of Potential Model Pruning on Different-Sized Boards in Monte-Carlo GO*. IJCSNS, vol. 12(11), pp. 17–22 (2012)

[4] Zobrist, A.L.: A model of visual organization for the game of GO. In: Proc. of the AFIPS Spring Joint Computer Conference, vol. 34, pp. 103–112 (1969)

[5] Nakamura, K., Kitoma, S.: *Analyzing Go Board Patterns Based on Numerical Features*. IPSJ Journal, vol. 43(10), pp. 3021–3029 (2002)

[6] Burrough, P.A., McDonnell, R.A.: *Principles of Geographical Information Systems*. p. 190 Oxford University Press (1998)

Table 2. Types of potential filters (Random Filter and PFs 1–4)

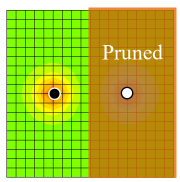
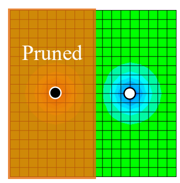
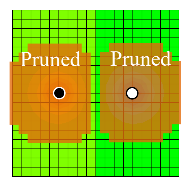
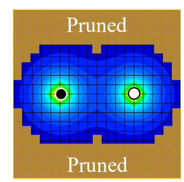
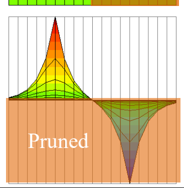
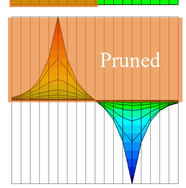
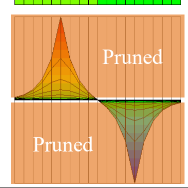
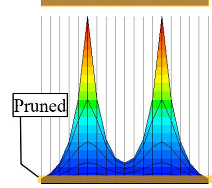
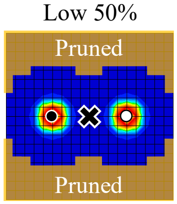
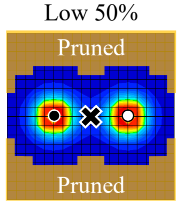
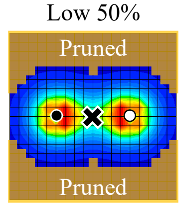
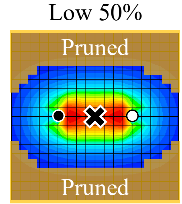
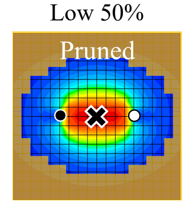
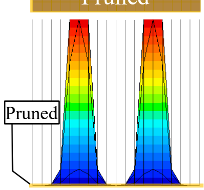
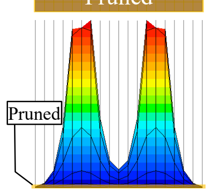
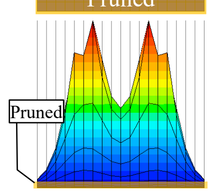
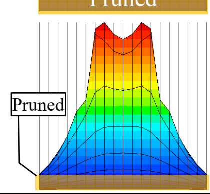
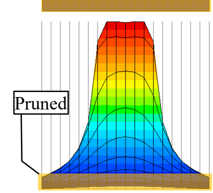
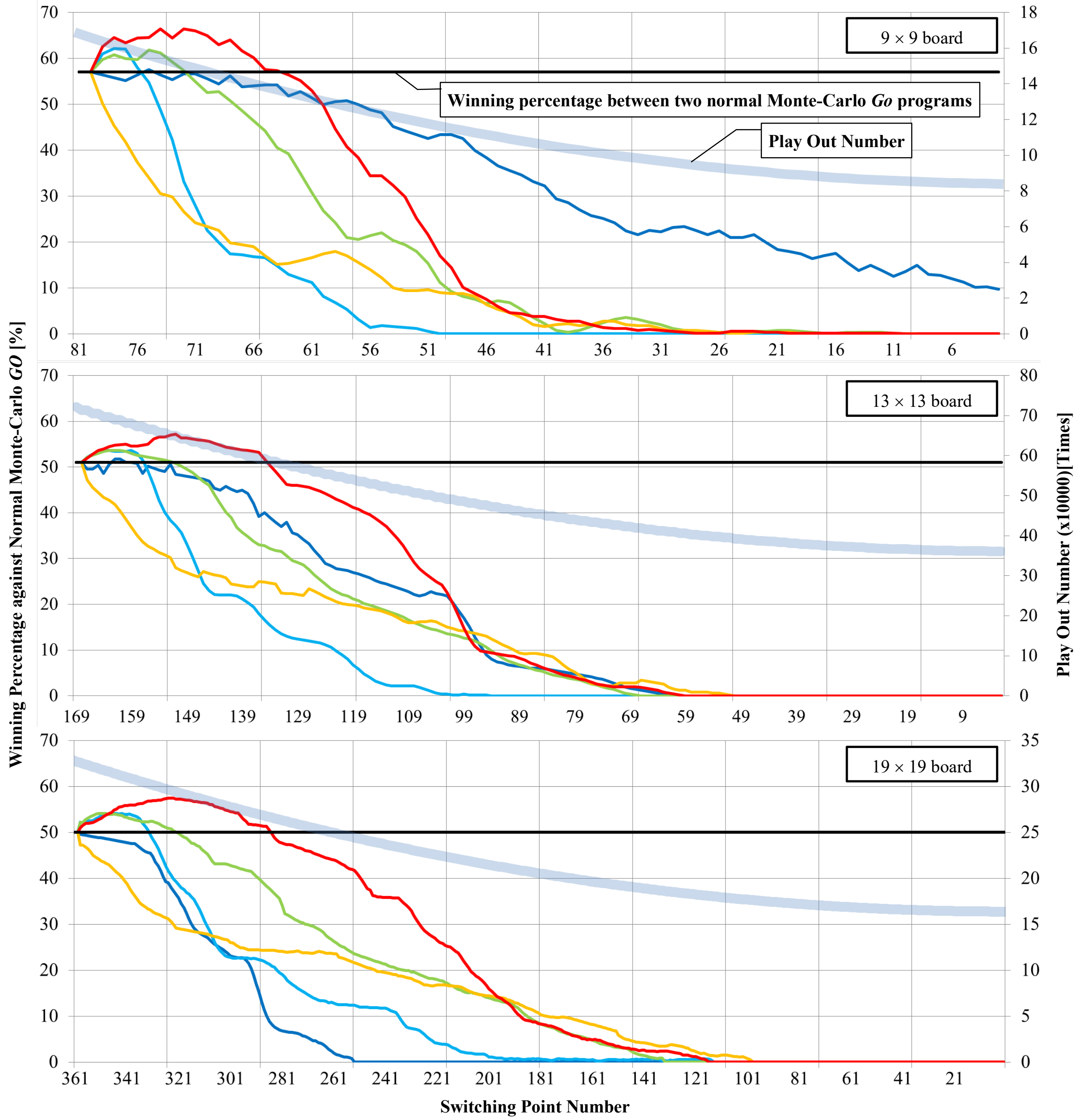
Method	Random Filter	PF 1	PF 2	PF 3	PF 4
Ranking	-	Potential	Potential	Potential	Potential
m	-	2	2	2	2
Black/White	-	+/–	+/–	+/–	+/+
Filtering	Random	Low 50%	Top 50%	Above 25% and below 75%	Low 50%
Overhead	-				
Landscape	-				

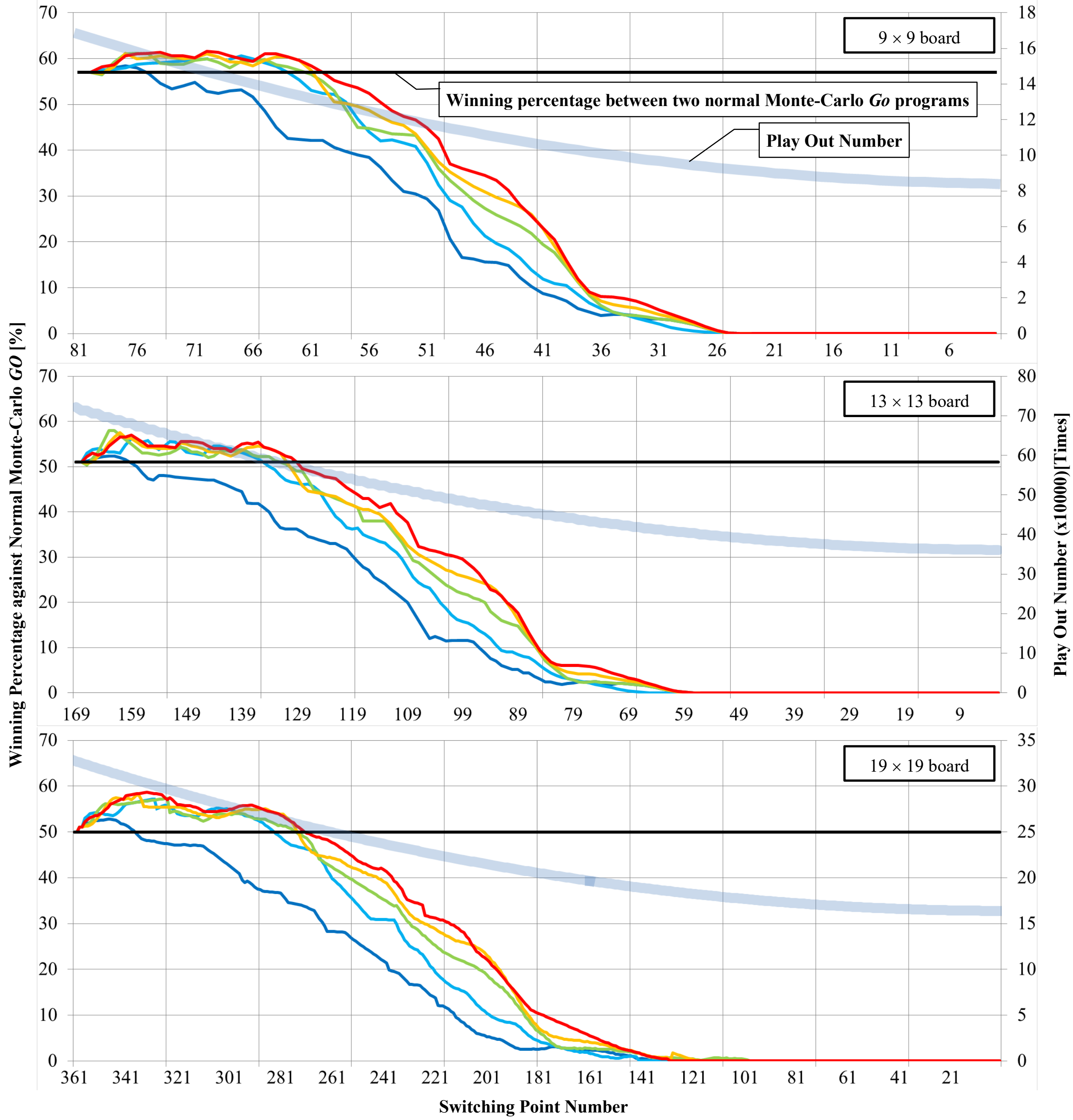
Table 3. Types of potential filters (Random Filter and PFs 5–9)

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					



Method		9 × 9 Board Size			13 × 13 Board Size			19 × 19 Board Size		
		Border	Play Out Number	Reduction Rate	Border	Play Out Number	Reduction Rate	Border	Play Out Number	Reduction Rate
Random		-	168,000	0.0	-	722,400	0.00	-	3,276,000	00.0
PF 1		77	160,000	4.7	157	673,500	06.8	333	3,032,400	07.4
PF 2		73	152,400	9.2	153	658,000	08.9	325	2,967,300	09.4
PF 3		-	168,000	0.0	-	722,400	00.0	-	3,276,000	00.0
PF 4		65	138,400	17.4	135	593,200	17.9	285	2,662,300	18.7

Fig. 3. Winning percentages of Monte-Carlo *GO* with potential filters 1–4



Method	9 × 9 Board Size			13 × 13 Board Size			19 × 19 Board Size		
	Border	Play Out Number	Reduction Rate	Border	Play Out Number	Reduction Rate	Border	Play Out Number	Reduction Rate
PF 5	77	160,100	04.7	159	681,400	05.7	343	3,117,600	04.8
PF 6	65	138,800	17.4	135	593,200	18.0	283	3,648,100	19.2
PF 7	63	135,600	19.3	131	579,900	19.7	277	2,606,100	20.4
PF 8	61	132,500	21.1	131	579,900	19.7	275	2,592,300	20.9
PF 9	61	132,500	21.1	129	573,400	20.6	273	2,578,600	21.3

Fig. 4. Winning percentages of Monte-Carlo *GO* with potential filters 5–9

Makoto Oshima-So received his B.E., M.E. and Dr. Eng. degrees from the University of the Ryukyus in 2003, 2005 and 2013, respectively. He has lectured at Okinawa International University since 2014. His research interests include artificial intelligence (AI), game programing and complex systems engineering. He is a member of JSAI and IPSJ.