Title	Group Performance on Computers Games in Chess and Checkers
Author(s)	SIRIVICHAYAKUL, THANATCHAI
Citation	
Issue Date	2019-06
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/16048
Rights	
Description	Supervisor: 飯田 弘之, 先端科学技術研究科, 修士(情報科学)

Master's Thesis

Group Performance on Computer Games in Chess and Checkers

1710097 Sirivichayakul Thanatchai

Supervisor	Hiroyuki Iida
Main Examiner	Hiroyuki Iida
Examiners	Ikeda Kokolo
	Nguyen Minh Le
	Shirai Kiyoaki

Graduate School of Advanced Science and Technology
Japan Advanced Institute of Science and Technology
(Information Science)

May 2019

Abstract

To study on the group benefit and performance, this research started with sub-research question *"What is the effect of group size when playing the game of Chess?"*. The following research objective is formulated, investigation on Majority Voting is conducted in order to find the essential information. After that, next purpose is to accurate the experiment result, locate support evidence from other perspective and further investigate on stability issue. The principal aim is to find the relationship between group benefit and performance in Majority Voting, and factors which can further improve efficiency and stability. The suggested hypothesis is that 'as stability increases the performance increases'. First, experiments were performed on groups of Chess programs to test the effect of group size on performance. Observation is mainly on homogeneous groups (copies of the same Chess program), as opposed to heterogeneous groups (different Chess programs). Groups were made up of Stockfish. Simple Majority Voting was used to mechanically combine the individual Chess program's decisions into a group decision. Games of Chess were played between groups of increasing size, and individual Stockfish was used as an opponent. Results show that winning rate increases as group size increases. However, second question is that *"Dose the effect of increasing in performance also apply to game other than Chess?"*. This question is to confirm that performance in artificial intelligence other than chess can be improved by this method because just investigating on Chess, the result might be biased and insufficient. This time, the experiment subject will change to checkers due to concern of appropriateness in complexity of game level and for diversity in experiment result. The principal aim is to find and verify the relationship between performance and stability performed by Majority Voting, and factors which can further improve performance. To further investigate the possible potential, Checkers is an interesting subject. Chess is much above in complexity compare to Checkers, the experiment might results in variety due to situation. The experiment subject will move to Checkers for diversity in experiment result for further conclusion. For further investigate in Majority Voting, experiments on groups of Checkers programs, playing by majority voting, were performed to investigate performance and stability. Homogeneous groups, copies of the same program, was also used to perform these experiments instead of heterogeneous group that was more complicated by factors of different programs. Experiments were performed based on a search-depth of 5, 10 and 12 using the Samuel checkers program. Games of Checkers were played between groups of size ranging from 1 up to 10 for each side. Experimental results of majority voting in Checkers suggest that group performance increases as a kind of logarithm function as the group size gradually increases for stronger player, and the performance slowly decreases in the case of a weaker player. In addition, stability seems to increase as the group size increases. The result can be assumed that for higher *n* number of group size, the smaller in difference of the average interval of winning ratio between each node, which means the decreasing of fluctuations and an eventual stabilization of the average value. Lastly, the next question is *"Can the performance enhanced methods work with each other?"*. Another effective method that was universally used nowaday is Machine learning, for the past

decade Machine Learning has become more influential in information technology and artificial intelligence. Experiments on groups of Checkers programs, playing by majority voting of reinforcement learning AIs, were performed to investigate performance and other relationship. Homogeneous group, copies of the same program, was still used to perform these experiments. Experiments were performed and observed on variation of number of 500, 1,000 and 1,500 train games using reinforcement learning Checkers program. Games of Checkers were played between groups of size ranging from 3 up to 13 reinforcement learning Checkers program and a standalone traditional alpha-beta pruning Checkers program as a base performance comparison. The main purpose of this experiment is to verify that difference enhanced methods can work together and further improve each other. Result of experiment still suggest that group performance increases in boundary of logarithm function as the group size increases for stronger player, and the performance slowly decreases in the case of a weaker player. Furthermore, we can presume that by increasing in train games and members in group makes the program more stable and also increasing in performance.. However, the results of this study also indicate that as group size become larger, at one point the performance will stop increasing noticeably and sometime the performance swing between increasing and decreasing in kind of see-saw effect but overall, the performance satisfactorily increase.

Contents

List of Figures

List of Tables

Chapter 1
Introduction

1.1 Background

Work on group superiority was carried out in 1898 by Triplett [1] who found that a group of individuals, each completing the same task, would start to compete. The additional effort by individuals resulted in the group outperforming each individual working alone. In this early work the group members were acting separately, but group superiority also exists in groups of interacting agents.

Various aspects of group performance have been studied since. Hackman and Morris [2] proposed an interaction process where *"at one extreme, for example, group members may work together so badly that members do not share with one another uniquely held information. On the other hand, group members may operate in great harmony, with the comments of one member prompting quick and sometimes innovative responses in another."* In another study, Woolley et al [3] found that intelligence, averaged from tests on different problems, exists for groups, and *"is correlated with the average social sensitivity of group members, the equality in distribution of conversational turn taking, and the proportion of females in the group."* Also, Webber [4] concluded that younger groups were more able to improve on individual performance, and Thomas and Fink [5] concluded *"group size is an important variable which should be taken into account in any theory of group behavior"*.

Machine group performance was investigated in 2009 by Hanawa and Ito [6]. They used mini-Shogi as a test problem, and formed a group of machines independently searching for moves to play. The group decision was a combination of the independent decisions, and was formed by taking the majority choice. They concluded *"This experiment hints at the possibility that consultation by majority is more effective than consultation by human players"*.

Group intelligence has been applied to games playing. Obata et al. [8] combined the decisions of three Shogi programs using the majority algorithm, and showed that the group played a stronger game of Shogi than any of the three members as individuals. In 2000, Kasparov [9] played a game of Chess against a team whose moves were combined by plurality vote. Kasparov won the game, however, he stated that he read the team's discussion forum during the game. Marcolino et al. [10] performed a study on Go, using the same procedure as Simple Majority Voting (except that they named it Plurality Voting) and concluded that *"it is possible for a team of weak but diverse agents to perform better than a uniform team made of copies of the strongest agent"*.

Many experiments and studies of Majority Voting indicate that, under certain conditions, the group Majority Voting performs better than the standalone components, group performance fluctuate on various aspect such as information sharing between group members, personality of members and also group size. Group benefit, in the case of game research, under certain conditions also signifies that direction, however there are still some ambiguities and areas for improvement. Further studies and experiments are needed to understand those conditions, and may contribute to enhance the performance of Majority Voting.

The term "homogeneous" here refers to the use of copies of the same Chess or Checkers game playing program, as opposed to a 'heterogeneous group' which would be composed of different Chess or Checkers programs.

1.2 Objective

The first objective of this research is to answer the sub-research question *"What is the effect of group size when playing the game of Chess?"*. The following research objective is formulated, investigation on Majority Voting is conducted in order to find the essential information. After that, next purpose is to accurate the experiment result, locate support evidence from other perspective and further investigate on stability issue. The principal aim is to find the relationship between group benefit and performance in Majority Voting, and factors which can further improve efficiency and stability. The suggested hypothesis is that 'as stability increases the performance increases'.

However, second question is that *"Dose the effect of increasing in performance also apply to game other than chess?"*. This question is to confirm that performance in artificial intelligence other than chess can be improved by this method. This time, the experiment subject will change to checkers due to concern of appropriateness in complexity of game level and for diversity in experiment result. The principal aim is to find and verify the relationship between performance and stability performed by Majority Voting, and factors which can further improve performance.

Next question is *"Can the performance enhanced methods work with each other?"*. Another effective method that was universally used nowaday is Machine learning, for the past decade Machine Learning has become more influential in information technology and artificial intelligence. The experiment was designed using reinforcement learning in checkers program and also apply majority voting in the calculating process to perform experiment against an individual traditional alpha-beta pruning Checkers program. The main purpose of this experiment is to verify that difference enhanced methods can work together and further improve each other.

Chapter 2
Homogeneous Group Performance In Chess

We performed experiments on groups of Chess programs to test the effect of group size on performance. We studied homogeneous groups (copies of the same Chess program), as opposed to heterogeneous groups (different Chess programs). Groups were made up of Stockfish. Simple Majority Voting was used to mechanically combine the individual Chess program's decisions into a group decision. Games of Chess were played between groups of increasing size, and individual Stockfish was used as an opponent. Results show that winning rate increases with group size.

2.1 Introduction

Work on group superiority was carried out in 1898 by Triplett [1] who found that a group of individuals, each completing the same task, would start to compete. The additional effort by individuals resulted in the group outperforming each individual working alone. In this early work the group members were acting separately, but group superiority also exists in groups of interacting agents.

Various aspects of group performance have been studied since. Hackman and Morris [2] proposed an interaction process where *"at one extreme, for example, group members may work together so badly that members do not share with one another uniquely held information. On the other hand, group members may operate in great harmony, with the comments of one member prompting quick and sometimes innovative responses in another."* In another study, Woolley et al [3] found that intelligence, averaged from tests on different problems, exists for groups, and *"is correlated with the average social sensitivity of group members, the equality in distribution of conversational turn taking, and the proportion of females in the group."* Also, Webber [4] concluded that younger groups were more able to improve on individual performance, and Thomas and Fink [5] concluded *"group size is an important variable which should be taken into account in any theory of group behavior"*.

Machine group performance was investigated in 2009 by Hanawa and Ito [6]. They used mini-Shogi as a test problem, and formed a group of machines independently searching for moves to play. The group decision was a combination of the independent decisions, and was formed by taking the majority choice. They concluded *"This experiment hints at the possibility that consultation by majority is more effective than consultation by human players"*.

Group intelligence has been applied to games playing. Obata et al. [8] combined the decisions of three Shogi programs using the majority algorithm, and showed that the group played a stronger game of Shogi than any of the three members as individuals. In 2000, Kasparov [9] played a game of Chess against a team whose moves were combined by plurality vote. Kasparov won the game, however, he stated that he read the team's discussion forum during the game. Marcolino et al. [10] performed a study on Go, using the same procedure as Simple Majority Voting (except that they named it Plurality Voting) and concluded that *"it is possible for a team of weak but diverse agents to perform better than a uniform team made of copies of the strongest agent"*.

In this work presented here, using the terminology 'homogeneous group' define a group of copies of the same Chess program, as opposed to a 'heterogeneous group' which would be composed of different Chess programs. Group performance has been considered by Sato et al. [7] whose model better suits a homogeneous group. This work presents an investigation of machine groups and aim to answer the sub-research question *"What is the effect of group size when playing the game of Chess?"*.

2.2 Method

We performed experiments on homogeneous groups of Chess programs to test the effect of group size on performance. S_n represents a group made up of *n* Stockfish† Chess programs. Every Stockfish program searched with a depth limit of 11.

1. Simple Majority Voting

The final group decision was computed using simple majority voting as described by Obata et al. [8]. The procedure is

1. compute n candidate moves by searching with each of the n programs in the group.
2. sum the total count of each candidate move.
3. if there is a majority candidate, select it as the group move.
4. if there is not a majority candidate, use the leader's (strongest member) proposed move to break the tie.

The experiment performed by using a group made up of *n* number of Stockfish programs of equal strength, therefore, the leader of this type of group was defined as the first member added to the group.

2. Winning Rate

1,000 games of Chess were played between S_n and Stockfish where $3 \leq n \leq 12$ the winning_rate was calculated using the following formula

$$winning\_rate = (win + 0.5 * draws)/1000$$

3. Denial Percentage

During the same experiments, statistics were gathered about the situations where the leader program had its proposed candidate move denied by the group majority. See Table 2.1.

Situation	Leader's candidate move accepted/denied
All programs propose same move	accepted
All programs propose different moves	accepted
Leader is included in majority	accepted
Leader is outvoted by majority	denied

Table 2.1. Enumeration of voting situations.

We calculated denial_percentage, where

$$denial\_percentage = denials/(group\_moves - opening\_book\_moves)$$

Here denials is the total number of denial situations in all games played by that group (see Table 2.1), *group_moves* is the total number of moves played by the group, and *opening_book_moves* is the total number of moves played from the opening book for that group.

2.3 Result

Fig. 2.1 shows the winning rate of groups of different size. As can be seen in the figure, there is a peak winning rate at s_6 and a second peak at s_{12}. There is also a valley in between s_6 and s_{12}. Also, the figure shows that winning rate increases with group size.

Fig. 2.2 shows the denial percentage of teams of increasing size. This figure shows that a team with an odd number of members appears to have a higher denial percentage than a team with an even number of members. It also shows that the denial percentage increases as the team size increases. Peak percentage in this graph is s_9, however we do not have data for s_{13} and greater.

There is a correlation between denial percentage and winning rate, in that both denial percentage and winning rate increase as the group size increases. However, the peak winning rates at s_6 and s_{12} cannot be correlated with denial percentage.

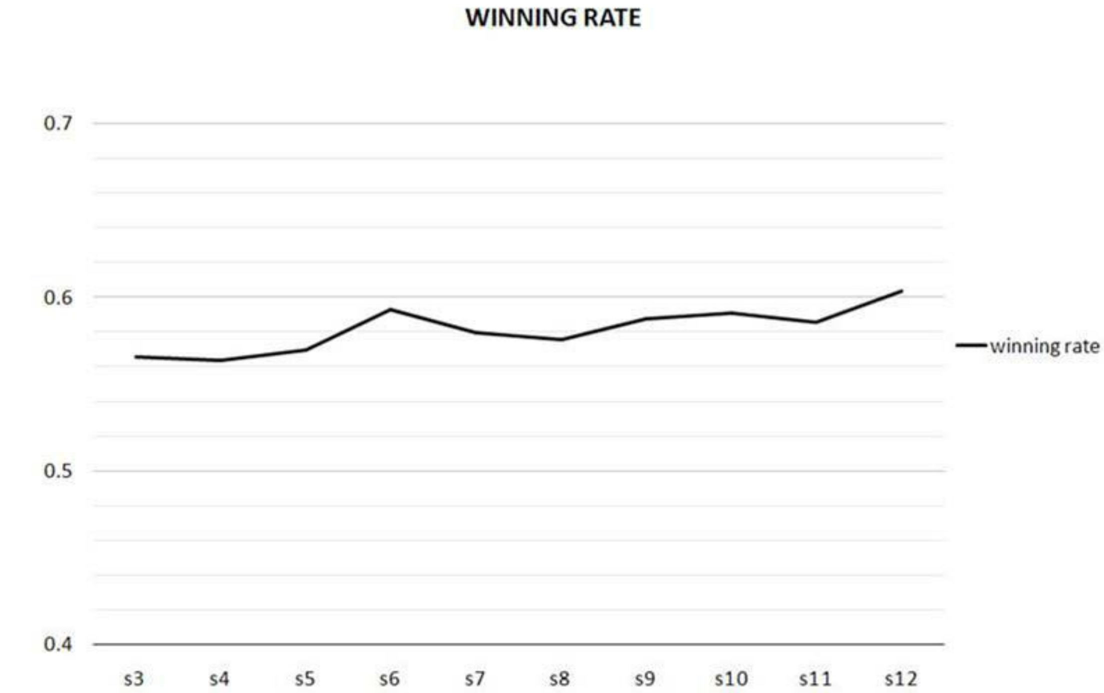

Fig. 2.1 – Winning rate of group

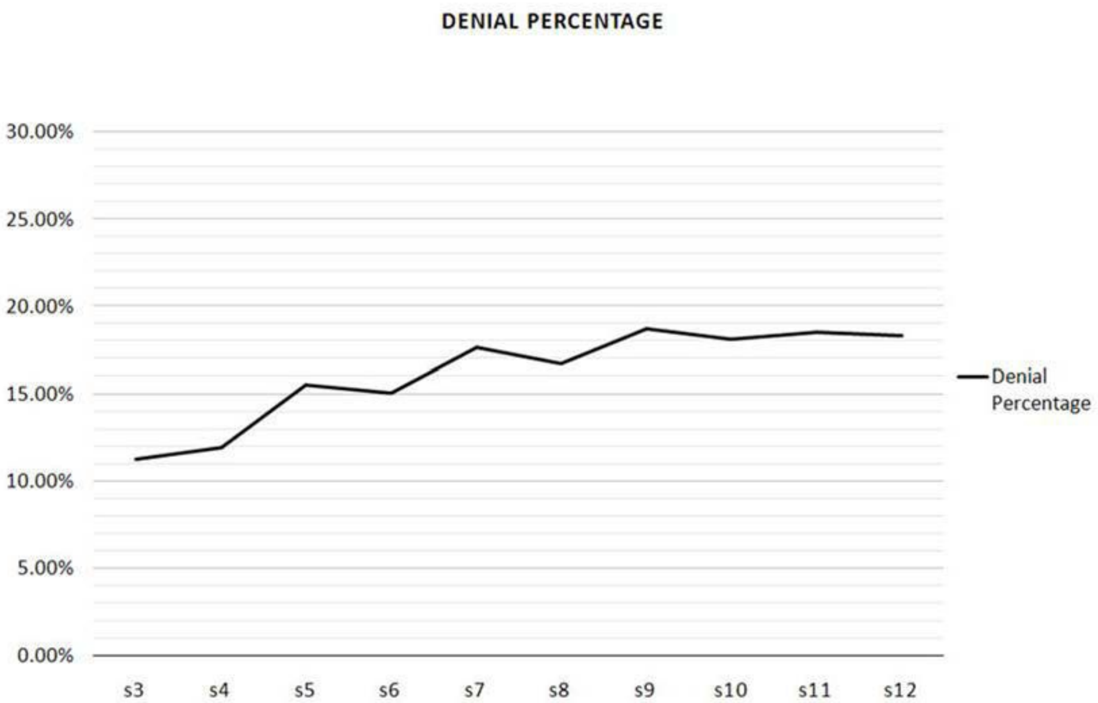

Fig. 2.2 – Denial percentage of group

2.4 Conclusion

In answer to the sub-research question detailed in the introduction, *"What is the affect of group size when playing the game of Chess?"* our results suggest that the winning rate increases as the group size increases.

However, just investigating on chess, the result might be biased and insufficient. To further investigate the possible potential, Checkers is an interesting subject. Chess is much above in complexity compare to Checkers, the experiment might results in variety due to situation. The experiment subject will move to Checkers for diversity in experiment result for further conclusion.

Chapter 3
Homogeneous Group Performance And Stability In Checkers

Experiments on groups of Checkers programs, playing by majority voting, were performed to investigate performance and stability. Homogeneous groups, copies of the same program, was used to perform these experiments instead of heterogeneous group that was more complicated by factors of different programs. Experiments were performed based on a search-depth of 5, 10 and 12 using the Samuel checkers program. Games of checkers were played between groups of size ranging from 1 up to 10 for each side.

3.1 Introduction

Studies on group benefit and performance have been conducted since 1898 [1]. In that study it was stated that collaboration of individuals, each completing the same task, resulted in the group outperforming each individual working alone. Many works indicated that group performance fluctuate on various aspect such as information sharing between group members [2], personality of members [4] and also group size [5]. Group benefit, in the case of game research, under certain conditions also signifies that direction, however there are still some ambiguities and areas for improvement.

Many experiments and studies of Majority Voting indicate that, under certain conditions, the group Majority Voting performs better than the standalone components. Further studies and experiments are needed to understand those conditions, and may contribute to enhance the performance of Majority Voting.

In previous topic, experiments mainly focus around Chess which is more complicated and harder to verify. Henceforth, the experiment subject will change to Checkers due to concern of appropriateness in complexity of game level and for diversity in experiment result. The purpose of this experiment is to investigate Majority Voting in Checkers. The principal aim is to find the relationship between performance and stability performed by Majority Voting, and factors which can further improve performance.

3.2 Method

1. Experiment design

Experiments on groups of Checkers programs were performed to test the effect of group size on performance and stability, with the 'simple majority voting' rule applied. Experiments were performed using a search-depth of 5, 10 and 12, with the Samuel checkers program[1]. The advantage of a player depends on search depth, as shown in [11] which shows the winning-rate of players in each depth and color. W_n represents a group on white side made up of *n* group size and R_m represents a group on red side made up of *m* group size.

First, the experiment was performed with the fixed search depth 10 because of suitability in strength and resource limit. Later, in consideration for relationship finding, experiments on search depth 5 and 12 were performed to compare and verify observations from the prior experiment. While focusing on selected search depth, experiments were performed with various changes in value of other variables.

2. Variables

- Group size: the number of player agents in a given side.
- Search Depth: Player's skill level.
- Player color: the side the player takes in the game; red and white. The player with the red pieces moves first.

3. Evaluation

Observations were made from both players' perspective (white player and red player). A series of experiments were situationally prepared and the data was collected for statistical analysis. To evaluate the group performance, several statistics such as winning rate and average branching factor are compared and computed.

The procedure of simple majority voting rule is as follows.

1. compute n candidate moves by searching with each of the n programs in the group.
2. sum the total count of each candidate move.
3. if there is a majority candidate, select it as the group move.
4. If there is not a majority candidate, use the strongest member's proposed move to break the tie. In the case of a homogeneous group, the move of the first candidate is selected.

1,000 games were played between two checkers programs to find W_n and R_m (where $1 \leq n, m \leq 8$). The winning rate was calculated using the following formula.

$$winningrate = (win + 0.5 * draws)/1000$$

Average interval was used to smooth out short-term fluctuations in a series of data obtained from the experiments in order to more clearly recognize and analyze longer-term trends, cycles and stability. The average interval, or moving average, is sometimes referred to as a rolling average or a running average. A moving average is a series of numbers, each of which represents the average of an interval of specified number of previous periods. The 2nd interval was used in this experiment.

The 2nd-degree average interval can be calculated using the following formula

$$avgInterval_{n+1} = (Winningrate_n + Winningrate_{n+1})/2$$

3.3 Result

1. Experiment on Search Depth 10

Fig. 3.1 shows the winning ratio of W_n against R_m, with the overall average at around 0.6. From the viewpoint of performance, the winning ratio steadily swings around the average value and does not clearly show any specific sign of raising or lowering in value. However, from the viewpoint of stability we can see that with *n* getting close to 1, the difference between *n-1*, *n* and *n+1* becomes higher compared to when *n* gets close to 8.

Fig. 3.2 shows the winning ratio of W_n against R_m, with the overall average at around 0.77. After adding the factor of draw into the winning ratio, the results are similar to the results in Fig. 3.1, the winning ratio swings steadily around average and with increasing *n* the value gets closer to the average value.

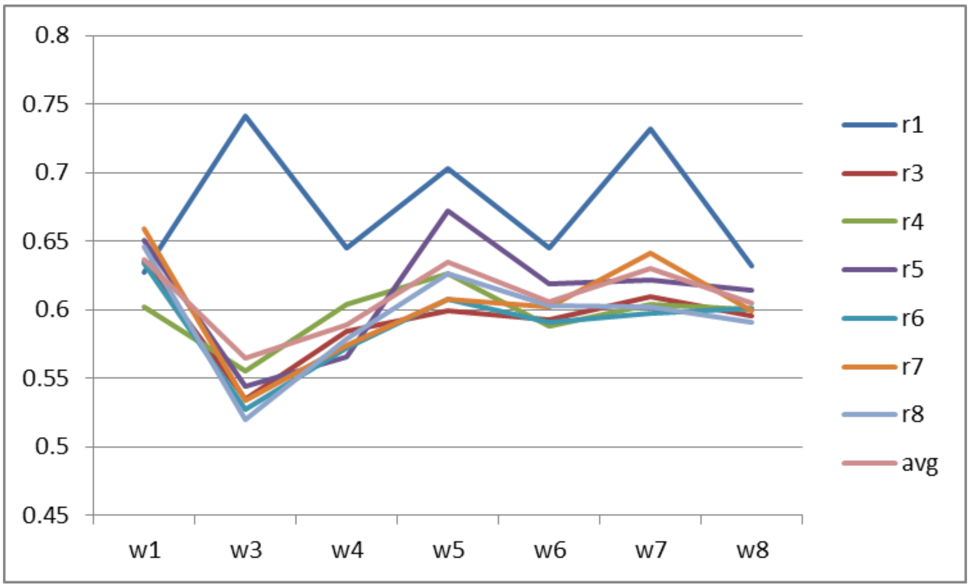

Figure 3.1 – Winning ratio of groups at search depth 10 (draw is not considered)

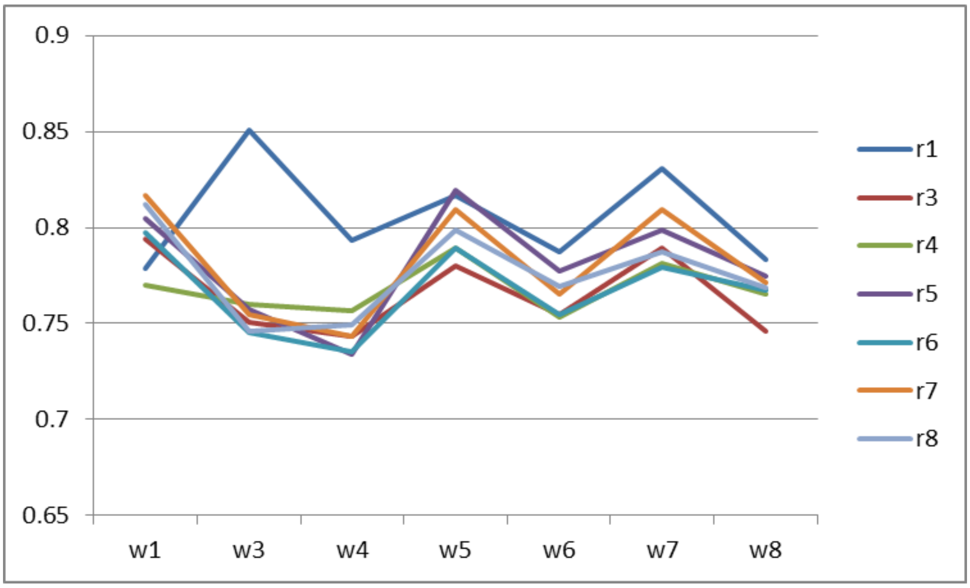

Figure 3.2 – Winning ratio of groups at search depth 10 (draw is considered)

Fig. 3.3 shows the average winning ratio of W_n against R_m, with the overall average at around 0.78. This figure shows an average value of Fig. 3.2 for accuracy to analyse the overall results, and shows that the winning ratio swings steadily around the overall average. Additionally, with increasing the group size, the fluctuations decrease and the interval range between nodes becomes closer, which implies that the average value gradually stabilizes. The average interval also shows that there is a tendency for the winning ratio to increase as the group size increases.

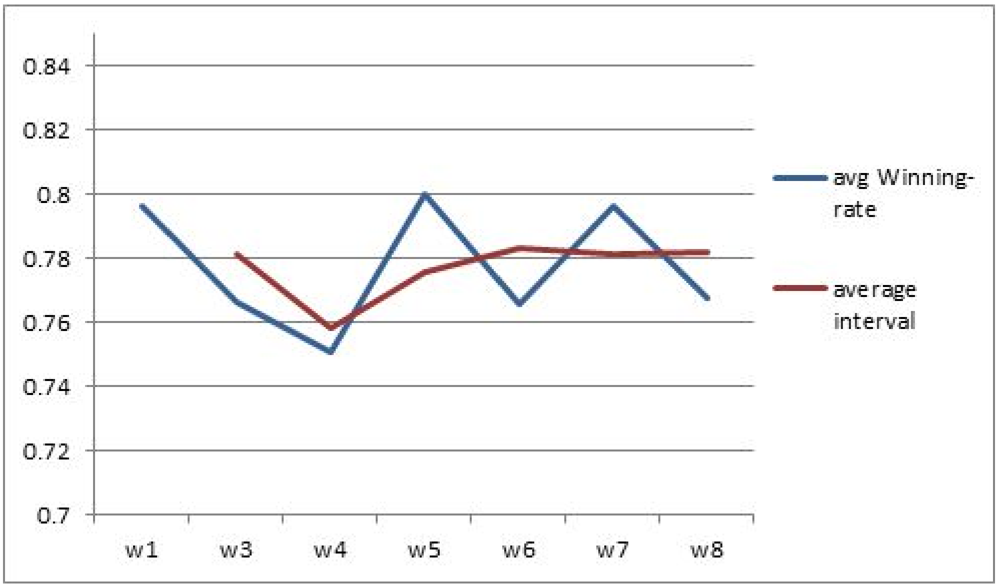

Figure 3.3 – Winning ratio of groups (W_n over R_m) and its interval at search depth 10

Fig. 3.4 shows the average winning ratio of groups R_m against W_n. The graph shows that the winning ratio swings steadily around the overall average value. The interval range between nodes becomes closer as the group size increases, whereas the trend shows that the performance has a tendency to gradually decrease, which is similar to the results in Fig. 3.3.

Figure 3.4 – Winning ratio of groups (R_m over W_n) and its interval at search depth 10

The results obtained from the experiments using a search depth of 10 can be summarised into 2 conjectures.

- Conjecture 1: The winning ratio becomes more stable as the group size increases since fluctuations in the results decrease and the average value gradually stabilizes [12].
- Conjecture 2: For stronger players, the performance increases as the group size increases, whereas the performance decreases as the group size increases for weaker players.

2. Experiment on Search Depth 5

In order to confirm these hypotheses, experiments were performed with depth 5 and 12, with the group size starting from 1 up to 12. Fig. 3.5 and 3.6 show the result of experiments at search depth 5. In this level red is the stronger player in accordance with the results in [11], but the analysis also resembles the experiment at search depth 10. Fig. 3.5 shows the average winning ratio of W_n against R_m and the average interval, which decreases in performance as the group size increases. On the other hand, Fig. 3.6 shows the average winning ratio of R_m against W_n and the average interval, which increases in performance as the group size increases. Both figures show similar results, in fluctuations of the graph and the stability of the average value.

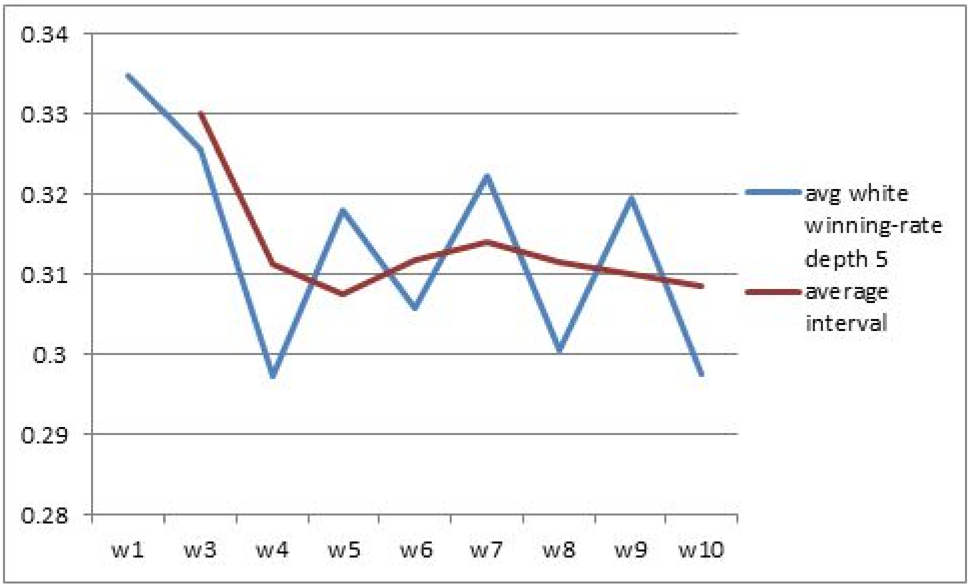

Figure 3.5 – Winning ratio of groups (W_n over R_m) and its interval at search depth 5

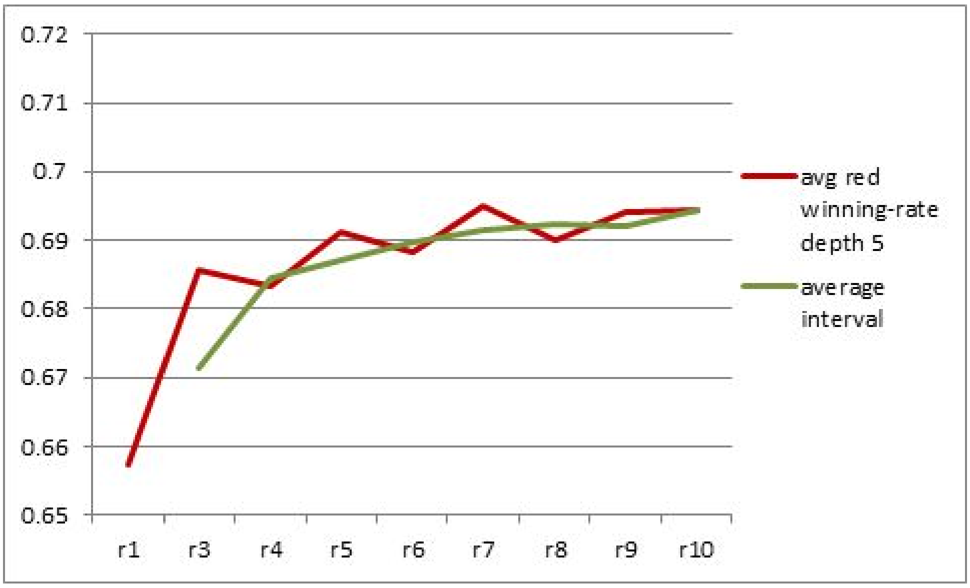

Figure 3.6 – Winning ratio of groups (R_m over W_n) and its interval at search depth 5

3. Experiment on Search Depth 12

Lastly, Fig. 3.7 shows both average winning ratio of W_n against R_m, the average winning ratio of R_m against W_n and the average interval at search depth 12. The results show the exceptional performance by non-majority voting player (standalone program) which goes contrary to the presumption. However, ignoring the result from the non-majority voting player, the overall results performed by the majority voting group present a similar outcome to previous experiments. Group performance steadily improves as the group size increases for the stronger white, declining in group performance as the group size increases for the weaker red, and furthermore the average value stabilizes by decreasing in fluctuations.

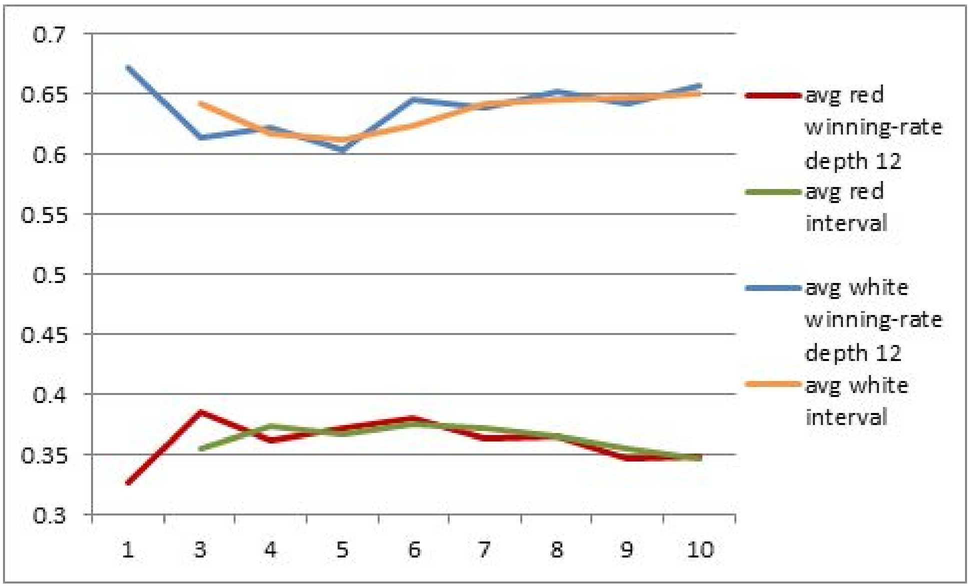

Figure 3.7 – Winning ratio of groups (W_n over R_m),

(R_m over W_n) and its interval at search depth 12

4.Probability Analyses

The probability of the team picking moves which are a majority vote, *PrM,* is given by

$$PrM=\frac{M}{b^n}$$

where M is the number of all possible majority vote combinations and can be easily counted combinatorially. For a team size $1 \leq n \leq 8$, and using Checkers with an average branching factor b of 3, then b^n is the total number of all move combinations, including majorities and ties, by the team.

The probability of the team picking moves are a tie, PrT, is easily given by

$$PrT = 1 - PrM$$

n	*M*	b^n	$PrM = M/b^n$	$PrT = 1\text{-}PrM$
1	3	3	1	0
2	3	9	0.333	0.666
3	21	27	0.777	0.222
4	63	81	0.777	0.222
5	153	243	0.630	0.370
6	579	729	0.794	0.206
7	1767	2187	0.808	0.192
8	4671	6561	0.712	0.288

Table 3.1 – Calculation of majority and tie probabilities

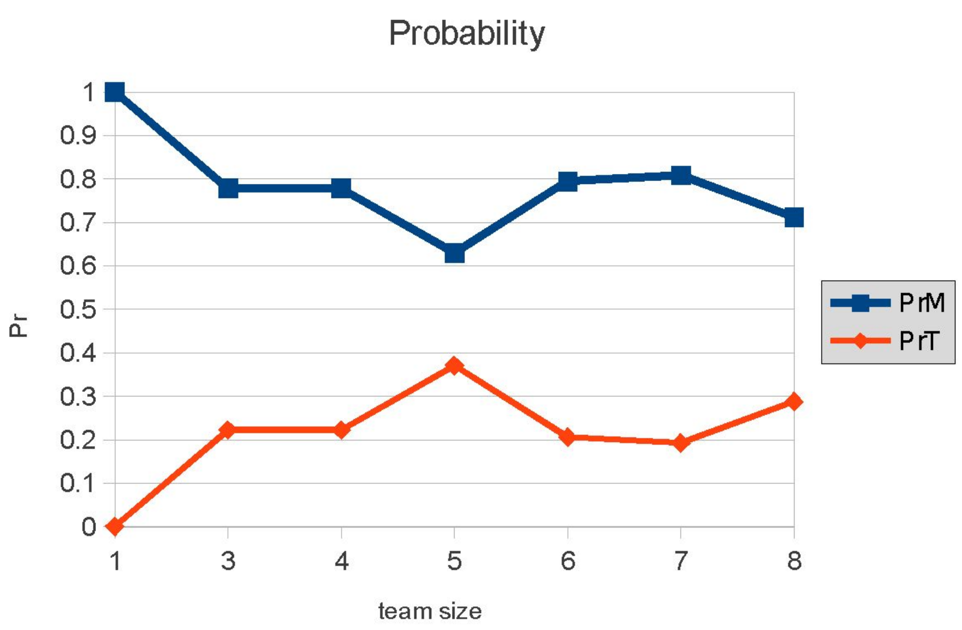

Figure 3.8 – Probability of majority *PrM* and tie *PrT*, for team size *n*.

Comparing *PrM* from Fig. 3.8 and the winning ratio from Fig. 3.2 shows some correlation, although team size 5 is inconsistent. In fact, for *PrT* then team size 5 is consistent.

The probability of the team picking moves which mean the leader's vote is played by the team, *PrL*, is given by

$$PrL=\frac{L}{{}_{b}n}$$

where *L* is the number of all possible situation combinations where the leader's move is picked and can be easily counted combinatorially by the following logic.

pickLeaderMove **IFF** isNotMajority **OR** (isMajority **AND** isLeaderInMajority)

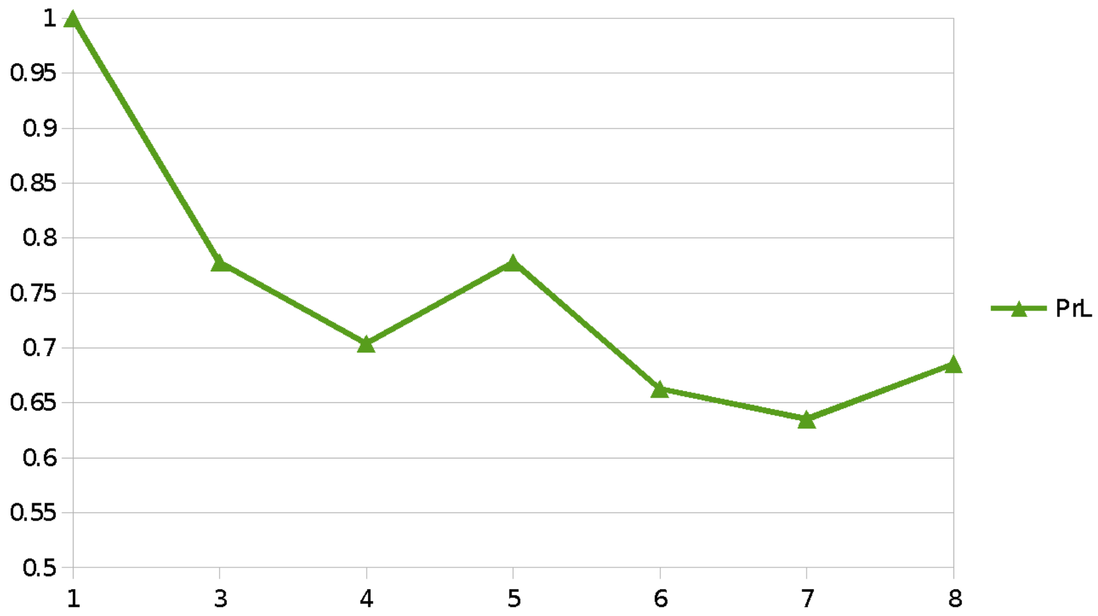

Figure 3.9 – Probability of leader's move *PrL*, for team size *n*.

Comparing *PrL* from Fig. 3.9 and the winning ratio from Fig. 3.2 we see some correlation, although team size 7 is inconsistent.

3.4 Conclusion

In summary, experimental results of majority voting in Checkers, suggest that group performance increases as a kind of logarithm function as the group size gradually increases for stronger player, and the performance slowly decreases in the case of a weaker player.

In addition, stability seems to increase as the group size increases. The result can be assumed that for higher *n,* the difference of the average interval of winning ratio between each node should be closer, which means the decreasing of fluctuations and an eventual stabilization of the average value.

We have also presented a probability analysis as a theory to explaining some of our present findings.

The results of this study can be concluded that with larger group size of a certain level players, majority voting would improve the group performance. Furthermore, that majority voting makes the homogeneous group more stable. It indicates a strong link between stability and quality in group performance.

Chapter 4
Investigate of Homogeneous Group Performance via Machine Learning

4.1 Method

Experiments on groups of Checkers programs, playing by majority voting of reinforcement learning AIs, were performed to investigate performance and other relationship. Homogeneous groups, copies of the same program, was used to perform these experiments instead of heterogeneous group that was more complicated by factors of different programs. Experiments were performed and observed on variation of number of 500, 1,000 and 1,500 train games using Reinforcement Learning Checkers program. Games of Checkers were played between groups of size ranging from 3 up to 13 reinforcement learning Checkers program and an individual traditional alpha-beta pruning Checkers program for base performance comparison.

1. Experiment design

Experiments on groups of reinforcement learning AI Checkers programs based on ideas of Q-learning were performed to test the effect of group size on performance and stability, with the 'simple majority voting' rule applied. Experiments were performed in variant setting in attribute of reinforcement learning AI Checkers against a traditional alpha-beta pruning AI with specified depth. Objective of this experiment was focused in majority voting, individual performance was omitted. L_n represents winning-rate of group made up of *n* group size in learning phase and T_n represents winning-rate of group made up of *n* group size in test phase.

The experiment was performed with the number of train games in one round of 100, 200 and 300 for comparison in performance and learning rate. In one set of experiment, number of training rounds was fixed at 5 rounds which means 500 training games, 1000 training games and 1500 training games respectively per experiment. After learning phase, total of 100 test games were played in one set of experiment. Other than number of training games and member in each group, each experiments have been performed in unbiased environment.

2. Variables

- Group size: the number of player agents in Reinforcement Learning AI Player.
- Numbers of Train games : games in learning phase of Reinforcement Learning AI Player in one round.
- Training Round : number of rounds performed in one experiment.

- Search Depth: Alpha-Beta AI Player's skill level was fixed in this experiment.
- Player color: the side the player takes in the game; red and white. The player with the red pieces moves first. Each player participates in same amount of games in each color.

3. Evaluation

Observations were made from both players' perspective (white player and red player). A series of experiments were situationally prepared and the data was collected for statistical analysis. To evaluate the group performance, several statistics such as winning rate and average branching factor are compared and computed.

The procedure of simple majority voting rule is as follows.

1. compute n candidate moves by searching with each of the n programs in the group.
2. sum the total count of each candidate move.
3. if there is a majority candidate, select it as the group move.
4. If there is not a majority candidate, use the strongest member's proposed move to break the tie. In the case of a homogeneous group, the move of the first candidate is selected.

5 rounds of training phase and test phase were played between group of Reinforcement Learning AI player and a traditional alpha-beta pruning AI with specified depth to find L_n and T_n (where $3 \leq n \leq 13$). The winning rate was calculated using the following formula.

$$winningrate = (win + 0.5 * draws)/total\ number\ of\ games$$

4.2 Result

Fig. 4.1 shows L_n, the winning rate of of group of Reinforcement Learning AI in learning phase. This figure shows that winning-rate decreases as the group size increases from 3 to 11 but suddenly increase when group size reach 13. Generally, the result is in harmony with the presumption from chapter 4, 'the performance slowly decreases in the case of a weaker player' since in learning phase, reinforcement player is significantly weak.

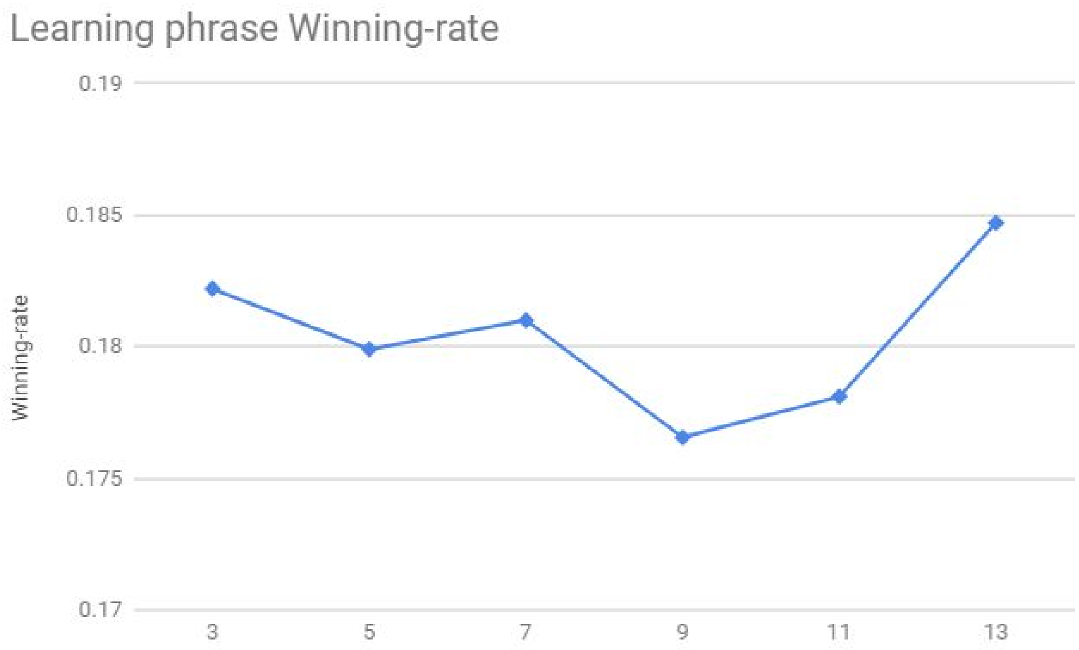

Figure 4.1 – winning-rate of group of Reinforcement Learning AI in learning phase

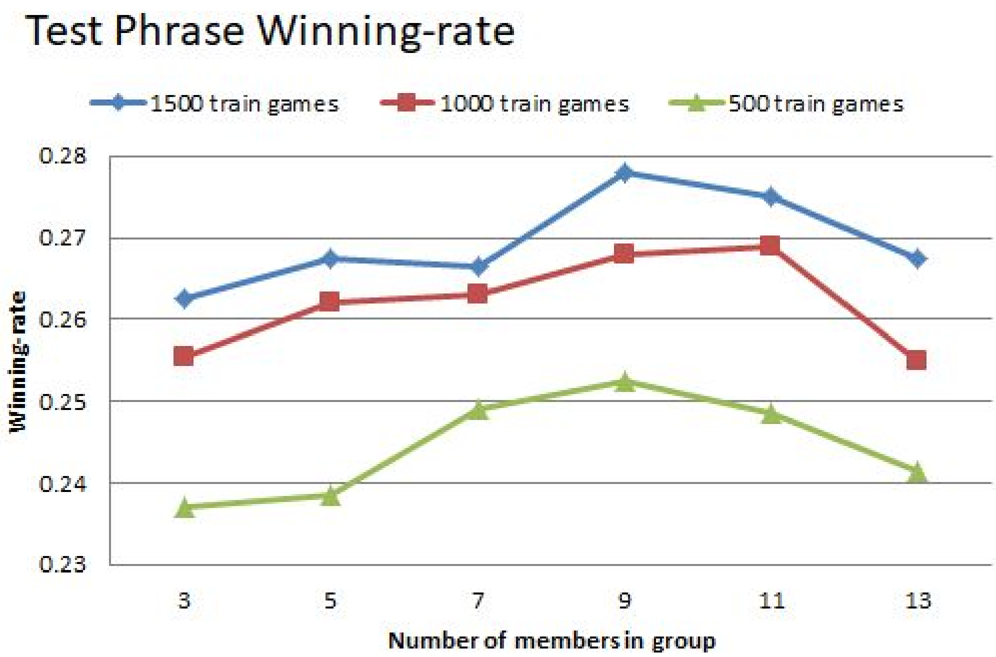

Figure 4.2 – winning-rate of group of Reinforcement Learning AI in test phase

Fig. 4.2 shows 3 sets of T_n, the winning rate of of group of Reinforcement Learning AI in test phase, in different environment of number of games to learn 500, 1000 and 1500. This figure shows that both dataset winning-rate increase as the group size increases from 3 to 11 but suddenly decrease when group size reach 13. Set of 500 train games has average of winning rate at 0.244, set of 1000 train games has average of winning rate at 0.262 and set of 1500 train games has average of winning rate at 0.269. Serie of 500 train games is peak in performance at 9 group members, Serie of 1000 train games is peak in performance at 11 group members and Serie of 1500 train games is peak in performance at 9 group members.

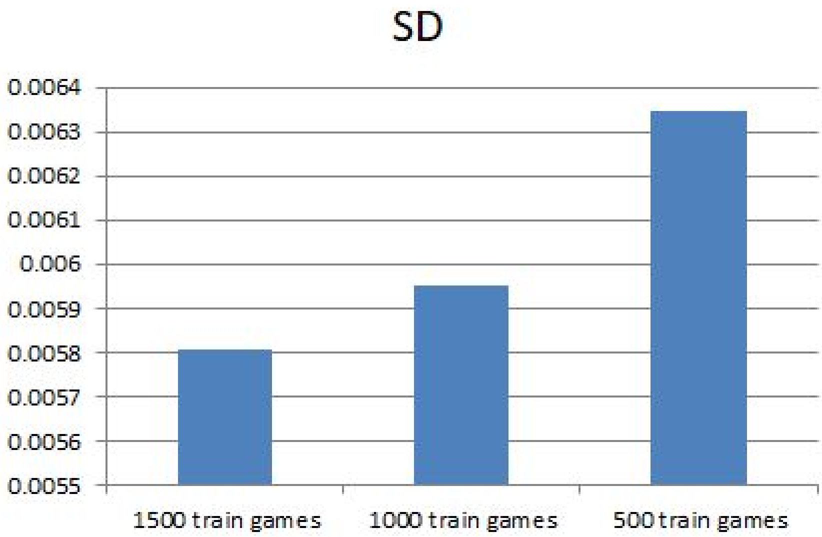

Figure 4.3 – comparison of SD of 500, 1000 and 1500 train games series

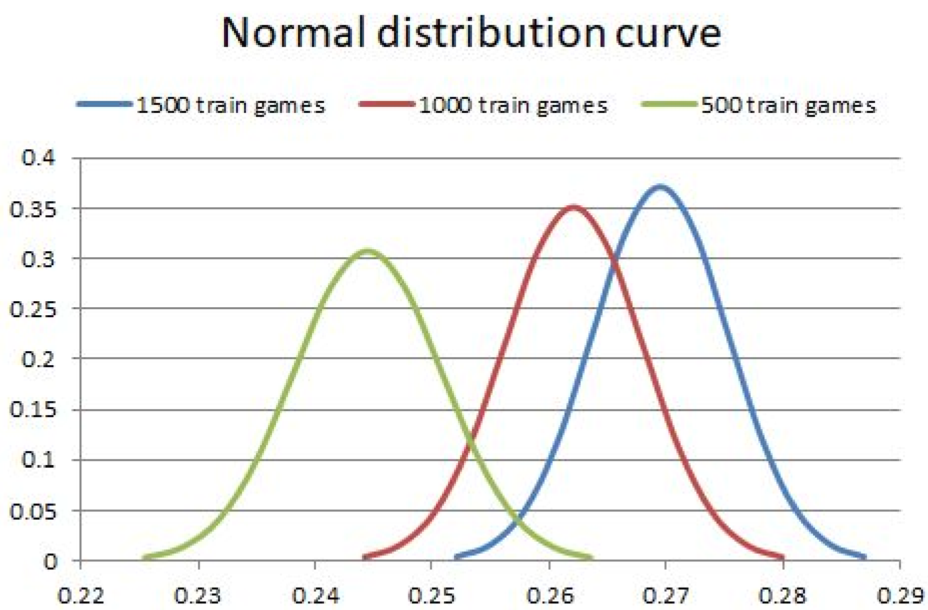

Figure 4.4 – Normal distribution curve of 500, 1000 and 1500 train games series

Data-set from winning-rate in Fig. 4.2 can be further analyse by calculating SD. Fig. 4.3 shows comparison of SD from each group of Reinforcement Learning AIs at number of train games 500, 1000 and 1500. The graph shows that SD is highest in serie of 500 train games and is lowest in serie of 1500 train game. From calculated SD, Normal distribution curve of each series can be drawn in Fig. 4.4, which shows that series of 1500 train games has the highest winning-rate and also lowest SD which means the lowest fluctuate in winning-rate distribution, on the other side series of 500 train games has the lowest winning-rate and also highest SD which means the highest fluctuate in winning-rate distribution. From this analysis, we can imply that increasing in train games and members in group makes the program more stable and also increasing in performance.

4.3 Conclusion

Experimental results suggest that group performance increases in boundary of logarithm function as the group size increases for stronger player, and the performance slowly decreases in the case of a weaker player. Furthermore, we can presume that by increasing in train games and members in group makes the program more stable and also increasing in performance.

However, the results of this study also indicate that as group size become larger, at one point the performance will stop increasing noticeably and sometime the performance swing between increasing and decreasing in kind of see-saw effect but overall, the performance satisfactorily increase. In this experiment, the most suitable number of group size for performance and stability is around 7 to 11, more than this will result in lower performance and inefficient in resource utilization.

Chapter 5
Conclusion

Majority voting can be used to increase performance and stability of any individual program as the group performance and stability increases in some kind of logarithm function as the group size increases to some point for stronger player, however the performance slowly decreases in the case of a weaker player. It can be concluded that with suitable group size of a certain level players, majority voting would improve the group performance, despite the suitable number of group size different in each program. Furthermore, that majority voting makes the homogeneous group more stable. It indicates a strong link between stability and quality in group performance.

Enhanced method for artificial intelligence such as majority voting and reinforcement learning can further enhance each others.

In aspect of computational resource, combination of group benefit and reinforcement learning can be sometime outperform other method such as search deeper since increasing group size only increase resource consumption in linear rate but search deeper increase resource consumption in exponential rate, in other word efficiency can be depended upon appropriate organization.

Difference between heterogeneous and homogeneous group is that in heterogeneous case compatibility between group member is importance but hard to adjust, if there is too much distance in level of each member or incompatible composition, group benefit will guarantee to be negative. In reverse, homogeneous group has no problem in compatibility and can expect reasonably group benefit. In summary, homogeneous group is easier to use and can anticipate some advantage, but despite harder to use and complicated configuration, heterogeneous group can expect more performance. This also means homogeneous group is more stable than heterogeneous group.

Bibliography

[1] Triplett, N., The dynamogenic factors in pacemaking and competition, American Journal of Psychology 9, pp. 507–533 (1898)
[2] Hackman J.R., and Morris C., Group Tasks, Group Interaction Process, and Group Performance Effectiveness: A Review and Proposed Integration, Advances in Experimental Social Psychology, vol.8, pp. 45–98 (1975)
[3] Woolley A., Chabris C., Pentland A., Hasmi N., Malone T., Evidence for a Collective Intelligence Factor in the Performance of Human Groups, Science 2010;330(6004): 686-688.
[4] Webber, R.A., The Relation of Group Performance to the Age of Members in Homogeneous Groups, The Academy of Management Journal, vol.17, no.3, pp.570-574 (1974)
[5] Thomas, E.J., Fink, C.F., Effects of group size, Psychological Bulletin, vol.60, no.4, pp. 371-384 (1963)
[6] Hanawa, M., and Ito, T., The optimal consultation system in a thought algorithm, The 3rd Symposium of Entertainment and Cognitive Science, pp. 72–75 (2009)
[7] Sato, Y., Cincotti, A., and Iida, H., 2012. An analysis of voting algorithm in games, Computer Games Workshop at European Conference on Artificial Intelligence (ECAI), pp. 102–113 (2012)
[8] Obata, T., Sugiyama, T., Hoki, K., and Ito, T., Consultation algorithm for computer Shogi: Move decisions by majority, The 7th International Conference on Computers and Games (CG2010), pp. 156–165 (2011)
[9] Kasparov, Kimovich G., Kasparov Against the World: The Story Of the Greatest Online Challenge, KasparovChess Online, (2000)
[10] Marcolino, L.S., Jiang, A.X., and Tambe, M., Diversity beats strength? - Towards forming a powerful team, In 15th International Workshop on Coordination, Organisations, Institutions and Norms (2013)
[11] Carvalho, D.S., Nguyen, M.L., Iida, H. “An analysis of majority voting in homogeneous groups for checkers: Understanding group performance through unbalance”, in Advances in Computer Games (M.H. Winands, H.J. van den Herik, and W.A. Kosters, eds.), (Cham), pp. 213–223, Springer International Publishing (2017)
[12] Gorban, I.I. The Statistical Stability Phenomenon, Springer International Publishing AG (2017)