

The Effect of Percentage of Attack/Defense Value on Chess Computer Performance

Kevin Huang

Percentage of Attack/Defense Style on Chess Computer Performance

The style at which a chess computer plays affects the move it plays and its relative performance. Kevin Huang studied the performance of different computer settings with different style proportions and their relative strength. The computer (100% Attack, 50% Attack/50% Defense, 100% Defense) along with the control, software default played individual matches against one another through a series of 15 two-game matches. Each variable played against the software default and other variables. No linear regression was found between percent of attack in computer's style and its performance. The data refutes the hypothesis because 100% Attack did not perform the best. It was clear that 50% Attack/50% Defense performed 30% relatively better than 100% Attack and 100% Defense. 100% Attack was slightly better than 100% Defense, however the data is insignificant to prove the definite relationship. Real-life application of this experiment is in the field of computing. Computers can be improved in the chess community with better-characterized style. Also this impacts on methods computers use to process (chess used as a good example to compare different computers) and their performance. Future research could be to extrapolate the experiment and relate style (in this case, how a computer decides to make move in chess) to search processes in real-life computers, making processing more efficient when the brute-force method may not be logical. This will allow less processing for better results.

Review of Literature

Chess is a board game in which players take turns to move their pieces. The objective of the game is to checkmate the opponent's king, by attacking the king with another piece (other than your king) and forcing the king to have no legal move to escape, block, or take the piece that is attacking it (ICC Rules). Therefore, the king is the most valuable piece as it determines the result of the game. In order to checkmate the enemy king, one must have some type of advantage (an imbalance) to form a strong attack to win. An imbalance is a significant difference in a position between white and black. (Silman Imbalances). Without the concept of imbalances, the game would be dead equal and results in a draw. Chess theorists/players have attempted to find the outcome of the game during perfect play. Perfect play in game theory is when both sides play the best possible move each turn leading to the ultimate result. Wilhelm Steinitz (1836-1900, World Chess Champion 1886-1894) introduced a new idea of *equilibrium* (Seirawan Winning Chess Openings). The starting positions of the pieces mirror each other, so they are in balance. White disturbs the equilibrium by moving first and gains the advantage because he leads in development of pieces (one kind of imbalance). Black responds in a way to restore the equilibrium, which states the concept of constant shifting in equilibrium. White breaks the equilibrium to his favor and black restores it. If white plays perfectly, then black should be playing to catch up until the forces of the armies are traded and the game is drawn. Theoretically, the result would always be a draw and only when one side makes a mistake (unable to restore the equilibrium) then a game can be won (Shabazz, Daaim). However, humans and even computers are unable to achieve perfect play for the whole game because chess isn't a solved game, yet. There isn't a best

move for each side that reaches the same result every time. For example, tic-tac-toe is solved game. Perfect play from both sides will always results in a draw. Game trees are used to display possible moves and computers are used to search all possibilities and generate a result. Chess has far more possibilities than tic-tac-toe (a deeper and more complex game tree). Claude Shannon, an American mathematician and engineer (1916-2001) estimated that there are about 10^{120} different possible games in chess and 10^{50} legal positions. Because perfect play in chess can't be achieved yet, it is true experimentally that white has a slight advantage (Chessgames Statistics). Breaking the equilibrium and creating an imbalance, it is important to note that attacking or having the attack is stronger than defense (Silman Imbalances). Attacking gives different kinds of advantages that contribute to winning a game, when defense is to try to prevent or reduce the attack from the opposing side and restore equilibrium (Pogonina Initiative). Defense is commonly used against stronger players to gain a draw because weaker players make more mistakes and they are less familiar about conducting an attack. In some sense defense supports attack, but attack is the dominant key in chess before perfect play is achieved.

Computer chess has affected chess in many ways, both positively and negatively. It has allowed humans to learn more about chess because of the strong calculation and search depth of the computer but also killing creativity, aiming for perfect play. Computers are very strong tacticians being able to search many moves in a short amount of time and have access to databases giving it all the opening repertoires (no human can match), but very weak in strategizing and pattern recognition. In the Deep Blue (chess supercomputer created by IBM) vs. Kasparov (World Chess Champion 1985-2000)

match, though Deep Blue was able to calculate 200 million positions per second it lost to Kasparov (Deep Blue's Victory). The important idea was that Kasparov's intuition and strategy largely compensated for his weaker calculation. Computers may play defensively to wear out the human player then attack as a strategical approach. Computers are very stable and aren't affected by the environment at all. In the first match of Deep Blue vs. Kasparov, the computer played what was a huge risk for human players yet it won which proves computers aren't psychologically affected either. Such stable conditions make computers easy to test with. Computers "think": If I do a and he does b , etc....then we reach position x . But if I do c and he does d , etc....then we reach position y . After judging the best end results (on a far more complex scale than the given example), the computer makes the move. This commonly used method is known as the brute-force method. Deep Blue used the brute-force method: searching all the possible moves of a certain number of moves to pick out the best move based on the programming code which tells it the desired result. (Deep Blue's Victory) The deeper and faster the computers search, the stronger the computer is. However because of the number of possible moves in chess, it would take an infinite amount of time to search all the moves not including that most of them are bad moves or experimentally impossible positions. The codes which computers follow are also very important. Different codes give computers different styles of play, which is why, creators of chess computers: Fritz, Rybka, Shredder, etc. don't just aim to improve the speed of the computers (Two Kinds of Smarts). Take a position with there are tow choices: take a pawn, which damages the position (known as a gambit), or decline the offer. This happens many times in the opening in which computers follow human games. The choice of openings is usually to choose the best ones, but there has to

be some style to chess computers in order for them to pick (ICS). In each program, there are different given values for each aspect of the game (e.g. king safety, material, development, etc.) and each computer make any move following a basic set of instructions corresponding to the different values. Take two different strong chess computers to strong players such as Kasparov and he'll be able to tell the difference and even name the computers.

Computer performance is a combination of speed and style; therefore it has a strong interaction with attack/defense value, one kind of style. This style is inputted in the code the computers process to tell them how to make moves, etc. The style value tells the computer how to evaluate positions based on the programming code and contributes to computer strength. The style can't be described with one word, but it is the unique process the computer goes through to choose the best move. The style sets computers apart from each other and based upon computer championships, it is clear that some styles are weaker than others. Of course, no style remains the best because of continuous technological progress. But one can draw general conclusions to characteristics of a promising style and an ineffective one. By using a stable factor, computers, to play out the different styles, there will be an accurate picture of performance (Two Kinds of Smarts).

In 1996-1997, Deep Blue played Garry Kasparov in a series of two matches. The first version played Kasparov in 1996 and lost the six game match (4-2). Deep Blue was then upgraded, doubling its processing power and its chess knowledge was tuned by Grandmaster Joel Benjamin. "Deeper Blue" won the rematch (3.5-2.5). It was the first time, a reigning world-champion lost to a computer in a match at standard time controls.

This data can be compared because this project was focusing on another factor of chess computer performance, speed. Through the increase of processing power when using the brute-force method, Deep Blue was able to increase its strength or performance in chess (measurable by the Elo rating system) as well as more knowledge on opening book moves.

A research was done by Mark Glickman to come up with a new method of measuring chess ratings to improve and contribute to some original methods (Glicko System). Based upon the equations to find chess ratings by previous systems (USCF, FIDE, etc.), Glickman used his theory upon chess performance to come up with Glicko system. The Glicko system did not equalize the amount of rating gained and lost unlike the Elo system. The Glicko system valued ratings deviation, RD, to judge the amount of rating gained and lost. The more games and frequency, one plays chess, the smaller the RD is. These findings of this research show that many games need to be played for one's RD to be smaller and more accurate. When testing to see whether an attacking style or defensive style is stronger, many games should be played to get an accurate result.

For this experiment to become valid the computer that is used must be the same and the amount of value on attack/defense must remain equal. Because perfect play in chess has not been achieved yet and having the attack gives more winning chances, it is thought that the style value affects the strength of computers as well as search depth (Two Kinds of Smarts). By conducting this experiment, this will help humans improve their play based upon their decision-making process of choosing the best move they think. This will also help humans make better and stronger chess-playing computers by balancing style value and search depth. The purpose of this experiment is to find out what

style of play is better and if having the attack is ultimately better than defense. It is hypothesized that if the computer plays a attacking style (aggressive, tactical, then the computer will be stronger (perform better and have a plus score against) than a computer that is playing a defensive style because by having the attack, one gets advantages which will increase one's winning chances despite the fact that black can always restore the equilibrium because perfect play has not been achieved yet.

Experimental Design

The Effect of Attack/Defense Value on Chess Computer Performance

Hypothesis: If the computer plays an attacking style, then the computer will be stronger than the same computer that is playing a defensive style because gaining the attack or initiative will increase its winning chances despite the fact that black can always restore the equilibrium because perfect play has not been achieved yet.

Control: Style Value of computer is 50% attack, 50% defense

Constants: chess computer used, the amount of value on attack/defense, values for other aspects of playing, number of trials for each independent variable, etc.

Independent Variable: Attack/Defense Proportion

Dependent Variable: Chess Computer Performance

Materials

- 1 computer
- 1 chess computer program - Chessmaster 3000
- Data collection tools (varies)

Procedures

1. Using the computer, open Chessmaster 3000 file.
2. Click on "Play". Scroll down to "Player Styles" and click "Create..."
3. Name the player "100% Attack".
4. Go under the "Style" section and move slider to 100% attacker (The percentage of the style measures how much of attack or defense the computer will play; a proportion between attack and defense). If not already, slide "material vs. position" to 50%, "vision" to 100%, "book depth" to 35(max), turn on deep thinking. Leave material points untouched and "Treat Draw As..." select "Draw". Save the player.
5. Repeat step 2. Name the player "100% Defense". Repeat step 4 except move the slider to 100% Defender.
6. Repeat step 2. Name the player "50% Attack/Defense". Repeat step 4 except move the slider to the middle (50% Attacker and Defender).
7. Click on "Play". Scroll down to "Tournament" and click "Create..."
8. Under the "Players" box scroll until "100% Attack" is seen and click on it. Add 100% Attack to Participants.
9. Under the "Players" box scroll until "Chessmaster" is seen and click on it. Add "Chessmaster" to Participants. Click "OK"
10. Next, enter "30" for the number of rounds for the tournament. Press play. This will set 100% Attack to play Chessmaster 30 times.

11. Record percentage scores for 100% Attack (e.g. 50% +15 =0 -15) and the game results (number of wins, draws, losses).
12. Repeat steps 7-10 twice only replacing 100% Attack with 100% Defense the first time, then 50% Attack/Defense the second time. Compare results between 100% Attack and 50% Attack/Defense, 100% Defense and 50% Attack/Defense, and 100% Attack and 100% Defense. This will give their relative placement.
13. Create another tournament using step 9 this time between 100% Attack and 100% Defense. Set number of rounds to 30. Record the percentage and results.
14. Repeat step 13, only replacing 100% Attack with 50% Attack/Defense and play another tournament with 30 games. Then repeat step 13 again only replacing 100% Defense with 50% Attack/Defense and play another tournament with 30 games.
15. Analyze and compare the results of all trials.

Data Analysis/Results

Table 1, Table 2, Graph 1, and Graph 2 all show the results of experiment conducted to measure chess computer performance in attack/defense style value among three different computer “players.” This was measured by playing the computer setting with the software default. In addition to the performance, data was collected for the style percentages in each computer setting. Table 1 shows the results of 15 trials (two-game matches) for each computer setting when playing against other settings and default.

Table 1: The Effect of Attack/Defense Value on Chess Computer Performance

(Game points in match)

Match	Chessmaster :Attack*	Chessmaster :Middle*	Chessmaster :Defense*	Defense :Attack*	Middle :Attack*	Defense :Middle*
1	1.5:0.5	1:1	2:0	0.5:1.5	1.5:0.5	1.5:0.5
2	0.5:1.5	1:1	2:0	1.5:0.5	2:0	1:1
3	2:0	1:1	1:1	1:1	2:0	0:2
4	0.5:1.5	1:1	0.5:1.5	1.5:0.5	1.5:0.5	0:2
5	1.5:0.5	1:1	1:1	1.5:0.5	1.5:0.5	1:1
6	1.5:0.5	1:1	1.5:0.5	0:2	0.5:1.5	1.5:0.5
7	1:1	2:0	2:0	1:1	1.5:0.5	0:2
8	1.5:0.5	0.5:1.5	1.5:0.5	0:2	0.5:1.5	1:1
9	0.5:1.5	1.5:0.5	1:1	1:1	1.5:0.5	1:1
10	0.5:1.5	0:2	0:2	1:1	1.5:0.5	0.5:1.5
11	1.5:0.5	1.5:0.5	1:1	1.5:0.5	1:1	1:1
12	1:1	1.5:0.5	1:1	0:2	1:1	0.5:1.5
13	1:1	0.5:1.5	1:1	2:0	2:0	1.5:0.5
14	2:0	1:1	1:1	1:1	2:0	1:1
15	1:1	0.5:1.5	1.5:0.5	1:1	0:2	1:1

*The unit is game points (match score).

The best value was calculated for each computer setting by taking the average of the 15 trials. The averages are shown below in Table 2.

Table 2: The Effect of Attack/Defense Value on Average Chess Computer

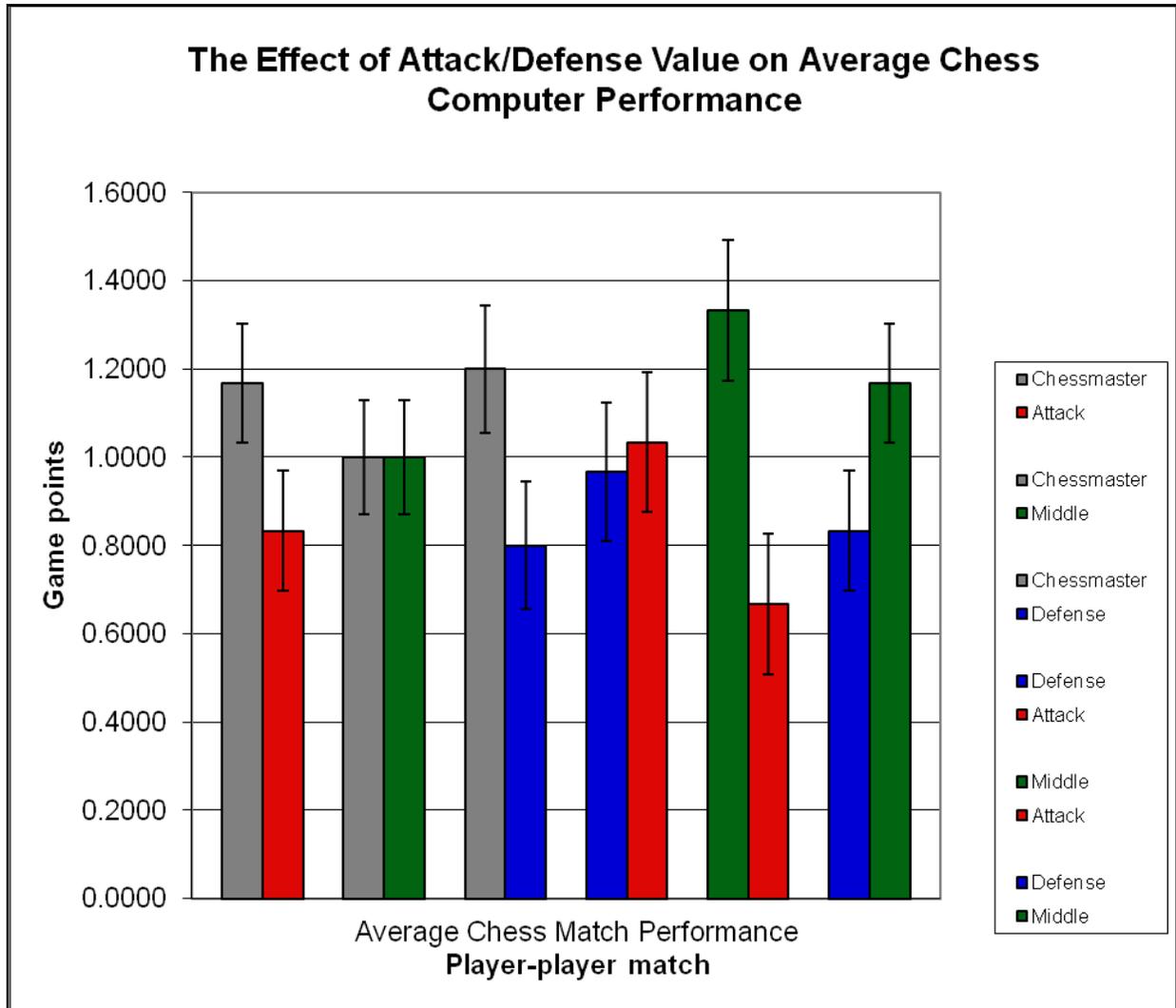
Performance (Game points in match)

Matches	Average Match Score
Chessmaster vs Attack	1.1667: 0.8333
Chessmaster vs Middle	1.0000: 1.0000
Chessmaster vs Defense	1.2000: 0.8000
Defense vs Attack	0.9667: 1.0333
Middle vs Attack	1.3333: 0.6667
Defense vs Middle	0.8333: 1.1667

On average, (out of two-game matches) it was clear that Middle was better than Attack (1.3333-0.6667) and Chessmaster was better than Defense (1.2000-0.80000). Chessmaster on average was fairly better than Attack (1.1667-0.8333) and Middle was fairly better than Attack (1.1667-0.8333). Attack and Defense were very close (1.0333-0.9667) along with Chessmaster-Middle (1.0000-1.0000) finishing the same. Graph 1 shows the averages for each computer setting.

Graph 1: *The Effect of Attack/Defense Value on Average Chess Computer*

Performance (Game points in match)



Variations among the trials created some uncertainty, as observed in the error bars above. Table 3 shows the standard error of the mean (SEM) by game points. The SEM was used to find the range of the accuracy the data (closeness to population mean) rather than variance from the sample mean (standard deviation). It will approximate the accuracy of the experiment instead of the precision.

Table 3: *The Effect of Attack/Defense Value on Average Chess Computer Performance (Game points in match)*

Average and Standard Error of the Mean

(Measured by game points in match)

Matches	Average Match Score	Standard Error of the Mean (SEM)
Chessmaster: Attack	1.1667: 0.8333	0.1351
Chessmaster: Middle	1.0000: 1.0000	0.1291
Chessmaster: Defense	1.2000: 0.8000	0.1447
Defense: Attack	0.9667: 1.0333	0.1579
Middle: Attack	1.3333: 0.6667	0.1594
Defense: Middle	0.8333: 1.1667	0.1594

When compared using error bars to include error (standard error of the mean), it is clear that Middle is better than Attack with an average match score between 1.1739 and 1.4924 to between 0.5073 and 0.8261. Chessmaster is better than Defense (1.0553-1.3447) to (0.6553-0.9447). Chessmaster is fairly better than Attack with an average match score between (1.5319) and (1.8021) to between (0.6982) and (0.9684). Middle is also better than Defense proportionally compared to Chessmaster against Attack. The value ranges following overlap because they had closer means. Attack against Defense (0.8754-1.1912) to (0.8088-1.1246) isn't significant because they finished nearly equal. Finally, Chessmaster against Middle was an equal fight between (0.8709-1.1291).

Table 4: *The Effect of Attack/Defense Value on Average Chess Computer*

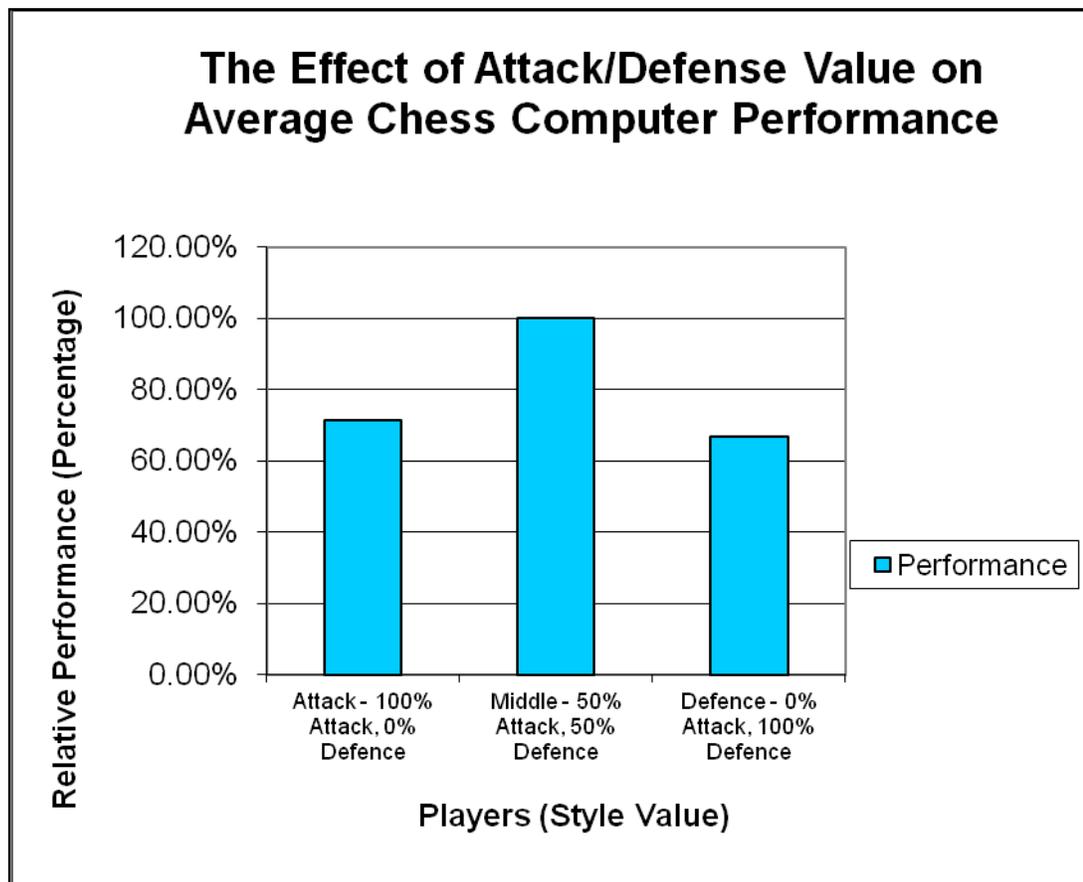
Performance (Game points in match) and Attack/Defense Style (Percentage) for Each

Computer

Matches	Average Match Score	Attack/Defense Style (Percentage)
Chessmaster: Attack	1.1667: 0.8333	Default : 100% Attack, 0% Defense
Chessmaster: Middle	1.0000: 1.0000	Default : 50% Attack, 50% Defense
Chessmaster: Defense	1.2000: 0.8000	Default : 0% Attack, 100% Defense
Defense: Attack	0.9667: 1.0333	0% Attack, 100% Defense : 100% Attack, 0% Defense
Middle: Attack	1.3333: 0.6667	50% Attack, 50% Defense : 100% Attack, 0% Defense
Defense: Middle	0.8333: 1.1667	0% Attack, 100% Defense : 50% Attack, 50% Defense

Graph 2: *The Effect of Attack/Defense Value on Average Chess Computer*

Performance (Game points in match)



During this experiment, no linear regression was found between the attack/defense style and chess computer performance. It was a combination of attack and defense which decided the outcome of the match.

The hypothesis was if the computer plays an attacking style, then the computer will be stronger than the same computer that is playing a defensive style because gaining the attack or initiative will increase its winning chances despite the fact that black can always restore the equilibrium, because perfect play has not been achieved yet.

According to the data, the hypothesis was mainly refuted. The relative performance of Middle to 100% Attack and 100% Defense was about 30% better. Both 100% Attack and 100% Defense lost to Chessmaster (default) while Middle did not. Although attack did slightly better than Defense, the data is insignificant and insufficient to prove that Attack is better.

Based upon the data collected in this experiment, one can conclude that Attack may be slightly better than Defense though the data shows it wasn't significant. It was statistically significant that Middle was the best compared to Attack and Defense. The data shows the value ranges of Middle were clearly higher than Attack and Defense during the matches (Middle vs. Attack and Middle vs. Defense). Also according to graph 2, Middle performed approximately 30% relatively better than Attack and Defense.

Conclusion

A study measuring the performance by chess computers was conducted to examine the relationship between attack/defense style percentage and performance in three different settings of a chess computer. The settings were 100% Attack (Attack), 50% Attack/50% Defense (Middle), and 100% Defense (Defense). It is hypothesized that if the computer plays an attacking style (aggressive, tactical), then the computer will be stronger than a computer that is playing a defensive style because by having the attack, one gets the advantage which will increase one's winning chances despite the fact that black can always restore the equilibrium because perfect play has not been achieved yet. These three settings and a software default (control) played each other in separate 15 two-game matches. It was found that Middle performed best of the three, approximately 30% relatively better than Attack and Defense. The standard error of the mean was used to decide the range of values because it shows the precision of the data to its true theoretical value. The data shows a significant amount of better performance for Middle. Middle is better than Attack with range of (1.11739-1.4924) to (0.5073-0.8261), an average match score out of two games (win=1, draw=0.5, lose=0 system). Chessmaster (software default or control) is better than Defense, (1.0553-1.3447) to (0.6553-0.9447). Chessmaster is fairly better than Attack with an average match score of (1.5319-1.8021) to (0.6982-0.9684). Middle is better than Defense with the same average. Attack against Defense (0.8754-1.1912) to (0.8088-1.1246) is not significant because they finished nearly equal. Finally, the data shows Middle and Chessmaster were equal (0.08709-1.1291) to (0.08709-1.1291). Attack performed slightly better than defense, but the data is insignificant to prove the correlation exists. The data refutes the hypothesis that the computer, which plays an attacking style, will perform the best. Ideally, each computer would have an average performance range that was statistically significant, but the researcher feels that the data refutes the relationship between attacking style value and performance mostly.

In 1996-1997, Deep Blue a chess computer developed by IBM was used to play the then-world chess champion, Garry Kasparov. The first version played Kasparov in 1996 and lost the six game match (4-2). Deep Blue was then upgraded, doubling its processing power and its chess knowledge was fine-tuned by Grandmaster Joel Benjamin. "Deeper Blue" won the rematch (3.5-2.5). It was the first time, a reigning world-champion lost to a computer in a match at standard time controls. This data can be compared because this project was focusing on another factor of chess computer performance, speed. Through the increase of processing power when using the brute-force method, Deep Blue was able to increase its strength or performance in chess (measurable by the Elo rating system) as well as more knowledge on opening book moves. A primary reason why Middle performed better can be explained by the strategy/principles of chess as well as use of computers. Because computers are stable to test with, their style percentage determined how moves were played throughout the entire game. Attack was aggressive, so it may have lacked in defense allowing counterattacking chances by the opponent. This may be why Attack could not beat Middle in a match. Defense was more passive, so it would be more difficult for it to exploit the weaknesses of the opponent. Chess is balanced; attack and defense are both important in playing the game. Purely aggressive or passive play isn't enough and Middle balances these two

factors to play stronger than Attack and Defense. (Pogonina Initiative) According to current chess statistics, attack is stronger than defense (white's statistical advantage isn't theoretically significant). (Chessgames Statistics) But because computers are accurate they play closer to the level of equilibrium, attack isn't stronger than defense. One can conclude, that by using both attack and defense in chess, the best performance can be derived (Seirawan Winning Chess Openings).

It is possible that some sources of error exist, as measurement of chess performance was incredibly difficult. Chess performance is measured by results of games, and the skill at which a player plays is not definite (varies). The computer during some games may play slightly better than other games. Ten trials (two-game matches) is a small sample size, so if possible it would be ideal to test more times (100-1000). There are many methods in which this experiment could be improved. The use of stronger computers could make play more accurate so style can be more analyzed further in detail. Also the stability of the chess computer could be improved so the strength of the computer has a smaller range. Another way, the results could be analyzed via real-life chess rating systems such as the Elo rating system and Glicko rating system to give numerical values to the performance of computers for better understanding. Further study of the experiment could be to relate the style (how the computer makes a move in chess) to search processes in real-life computers, making their processing more efficient when the brute-force method may not be logical.

Bibliography

- Chessgames.com. "Statistics Page." *Chessgames.com*. Chessgames Services LLC, 2011. Web. 16 Jan. 2012. <<http://www.chessgames.com/>>.
- Glickman, Mark. "The Glicko system." *Glicko System*. Mark Glickman, n.d. Web. 16 Jan. 2012. <<http://www.glicko.net/>>.
- ICC. "Rules." *ICC*. Internet Chess Club, n.d. Web. 16 Jan. 2012. <<http://www.chessclub.com/resources/rules/>>.
- International Chess School. "Making decisions in chess." *ICS*. ChessMasterSchool.com, n.d. Web. 16 Jan. 2012. <http://www.chessmasterschool.com/making_decisions.pdf>.
- Krauthammer, Charles. "Be Afraid. The Meaning of Deep Blue's Victory." *Weekly Standard*. 26 May 1997: 19-23. *SIRS Issues Researcher*. Web. 16 Jan. 2012. <<http://sks.sirs.com/bin/article-display?id=SMD1570H-0-9662&artno=0000018931&type=ART&shfilter=U&key=chess&title=Be%20Afraid.%20The%20Meaning%20of%20Deep%20Blue%27s%20Victory&res=Y&ren=Y&gov=Y&lnk=Y&ic=Y>>.
- Lyman, Shelby. "Joel Benjamin: The Champion Behind Ibm's Deep Blue." *Chicago Tribune*. Chicago Tribune, 12 Oct. 1997. Web. 3 Dec. 2012. <http://articles.chicagotribune.com/1997-10-12/news/9710120477_1_deep-blue-chess-champion-grandmaster>.
- Pogonina, Natalia. Online posting. *Initiative*. Chess.com, 20 Dec. 2010. Web. 16 Jan. 2012. <<http://www.chess.com/>>.
- Seirawan, Yasser. "Classical King Pawn Openings." *Winning Chess Openings*. Illus. Horatio Monteverde. Ed. First Rank Publishing. Vol. 5. London: Everyman Chess, 2003. 33-34. Print. Winning Chess.
- Shabazz, Daaim. "The Socio-Politics of the First Move in Chess." *The Chess Drum's 65th Square*. N.p., 24 Jan. 2008. Web. 16 Jan. 2012. <http://www.thechessdrum.net/thSquare/_janfeb08.html>.
- Silman, Jeremy. "Imbalances." *How to Reassess Your Chess*. Illus. Wade Lageose. 4th ed. Los Angeles: Siles Press, 2010. 3-13. Print.
- "Two Kinds of Smarts." *Why Files (National Science Foundation)* 27 May 1997: n. pag. *SIRS Discoverer*. Web. 16 Jan. 2012. <<http://discoverer.prod.sirs.com///?urn=urn%3Asirs%3AUS%3BARTICLE%3BART%3B0000031195>>.

Acknowledgements

I would like to thank my parents for their support and help during the course of this research project especially with processing the data.

I would also like to thank my science teachers for guiding me to reach the requirements for the research project step-by-step.