Japan Advanced Institute of Science and Technology

Various Approaches to Improving Entertainment Impact in Games

by

Chetprayoon Panumate

A thesis submitted in partial fulfillment for the degree of Master of Information Science

Written under the direction of
Professor Hiroyuki Iida
School of Information Science

September 2016

Abstract

Human stress in one of the most important topics in psychology and game can be one of the most efficient tools to relieve stress. The more quality of the game has, the more efficient that game can relieve stress. That is why we should concern the enjoyment and the entertainment in games. Therefore, in this thesis, we present three ways to improve the entertainment impact in games.

The first way is to use game refinement theory, a unique theory that has been used as a reliable tool for measuring the attractiveness and sophistication of the games considered. This theory is originated from the concept of outcome uncertainty and was invented by Iida *et al.*. A game refinement measure is derived from a game information progress model and has been applied in various games. In this thesis, we apply game refinement theory to Pokemon, sports, RoShamBo and Snake game. We show how game refinement theory can be an essential tool to find comfortable settings of the games and improve the entertainment impact in games.

The second way is to develop an emotional AI. We believe that AI can improve the entertainment of the games significantly. This is because player usually spends a lot of time interacting with AIs in video games. So, adding emotional component in AI, which will makes the AI's behaviour more interesting and realistic, will explore the new experience for player. Therefore, we present a generic model for emotional AI in real-Time multiplayer fighting games. Furthermore, we implement emotional AI to our simulated game and perform evaluation experiment in order to verify the efficient of the proposed model.

The third way is to propose a new model for quantifying enjoyment between match in the same game. We believe that if we can figure out a reasonable model to quantify the enjoyment between matches in the same games, it means that we can truly touch the enjoyment which is so abstract. So, we can improve the entertainment impact of the game respectively. Therefore, we present the basis idea of this model and propose many further works in order to improve the proposed model.

We believe that these three approaches presented can effectively improve the entertainment impact in games. For each chapter, results obtained are discussed and concluding remarks are given. Further works should be investigated in many points as we have proposed in order to tangibly touch the entertainment.

Acknowledgements

First of all, I would like to express my sincere gratitude to Professor Hiroyuki Iida for being my great advisor and supporting me many things during my life as a master's student at Research Center for Entertainment Science, Japan Advanced Institute of Science and Technology. From the first day I enrolled JAIST until today, he taught me many priceless lessons, not only how to do an excellent research or how to write a good academic paper, but also how to live in our society and how to spend your life meaningfully. I have learned so many things from him and I sincerely appreciated his favor from the bottom of my heart.

Secondly, I would like to thank Associate Professor Ikeda Kokolo and Associate Professor Hasegawa Shinobu for being my committee. They gave me a lot of suggestions. Also, another teacher and teaching assistant who taught me in the lecture classes, I would like to say thank you for teaching me many things.

Next, during my master's student life, I had a chance to be an internship student in SQUARE ENIX CO., LTD. around 4 months. Therefore, I would like to thank SQUARE ENIX CO., LTD. for giving me a precious chance to prove myself as an internship student. Also, I would like to thank my supervisor, Youichiro Miyake, my Thai seniors and many friends in the company. Without them, I cannot pass this great challenge. The result from my internship was written in Chapter 6. Also, during my internship, I had a chance to stay with Japanese host family. Therefore, I would like to thank Nakamura family for giving me a comfortable accommodation and many kind supports. They taught me many things such as Japanese culture and Japanese language. It made me understand Japan more clearly. Also, I thank Mrs. Asakura for helping me contact to Nakamura family.

Furthermore, I would like to thank Professor Ogata Kazuhiro for helping me find a job in a great country likes Japan. Without his helps, I could not pass the interviews and got the job offer from the company. Beside these contributions, I would like to thank my seniors, my friends and my juniors, both from Thailand and foreigner. They fulfilled my master's student life and made me enjoy spending time in Japan. Moreover, I would like to thank Mr. Chanyachatchawan Sapa for the cover page. Additionally, I thank JAIST office staffs for helping me many things. I appreciated their favor very much.

Finally, I must express my very profound gratitude to my parents and my family for helping me to challenge one of the biggest goal in my life, studying masters degree course abroad. This accomplishment would not have been possible without them. Thank you.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis presents various approaches to improve the entertainment impact in games. In this chapter, we first explain the background of this study. Then, we set our problem statement clearly. Finally, the structure of this thesis is explained.

1.1 Background

Nowadays, there are many pressures and stresses that come with being a human in our society. It comes through our daily life such as studying, working, commuting to work, taking care the relations between someone and so on. For everyday, bad feelings occur many times and we have to control it in order to keep our daily life does on well. That is why human stress is one of the most important topics in psychology. To surpass those bad feelings, taking a rest is the most important thing. There are many ways to rest such as shopping, watching movie, listen to music, etc., and this depends on person. Among them, playing game is one of the most outstanding way which can release our stress while enjoying the game.

Game is all around us. It can be seen ubiquitously in our daily life. Moreover, the developing of technology creates the larger space for game such as in video game consoles, computer or in mobile phones. From this point, we will see that game has conquered to those electronic devices and it is very easy to spend the time to play game.

However, does this situation means that we can relax our stress easily? The answer is 'no'. Of course, the easier we can play the game means the more chance that we can relax. However, if the game is not efficient, it can not help us to relax our stress effectively. To be a good tool for helping human relaxes his stress, the game must be efficient and deep enough to control the feeling of players while they are playing.

There are many works which focus on the effect of the game to player [65] [48] [17]. Among them, [109] confirms that game can relieve stress significantly. Playing video games can relieve your stress, reduce your depression, and make you feel better. Player will forget everything in the real world, go in to the world on game and enjoy with them happily. However, as we have said, it depends on the quality of the game. Therefore, the word 'entertainment', 'enjoyment' or 'excitement' are concerned seriously.

When you think about playing game, you may think about many teen boys sitting in front of a big-screen TV like zombies, incessantly jamming buttons on their controllers, while aggressively playing violent video games. However, many studies [107] show that now the game players are not only teen boys, but also teen girls and adults. It means that, nowadays, game is widely played. Therefore, if we can improve the enjoyment of the games, it will affect not only to some groups of teen boys, but it will affect to a lot of people effectively.

Many studies related to games are investigated such as developing the strongest AI under some limited constraints. Previously, the research in games field usually focuses on the game's AI. This is because it is so amazing that human can play interactively with AI. Moreover, AI with a reasonable behaviour can be applied to another domain such as in economics [74]. Also, it can be applied as a decision making tool to another domain such as marketing [37] and political science [134]. We will see that focusing in this way can make the great contributions. The stronger AI can be, the more technology can develop.

However, doing research in that way cannot answer one of the most important purposes of game, relaxation. Therefore, this work focuses on the different point from those works, the entertainment impact in games. We believe that improving the entertainment impact in games can be one of the essential ways to help human releases their stress comfortably. Therefore, we propose our thesis, various approaches to improve the entertainment impact in games.

1.2 Problem Statement

As we have state clearly, the purpose of this thesis is that to improve the entertainment impact in games. Therefore, the problem statement will be as follow.

Problem Statement How to improve the entertainment impact in games?

In this thesis, we focus on many points in game in order to fully understand about the game. For each chapter, we set the main research questions and try to answer by using some methodology which will be proposed. Many components from many domains of games are analyzed reasonably in order to reach our final goal, to improve the entertainment impact in games

1.3 Structure of the Thesis

There are 7 chapters in this thesis. We present three ways to improve the entertainment impact in games. Chapter 1 gives the background and the problem statement of this study.

In Chapter 2, we present the first way to improve entertainment impact in games which is our purpose. Game refinement is employed as an essential tool for this study. Therefore, this chapter provide the background knowledge of game refinement theory and its recent works. Also, the related works are provided.

In Chapter 3, we present the first example of using game refinement theory. We apply game refinement theory and show how game refinement theory can improve entertainment impact in Pokemon game. In this chapter, we consider three sub-domains in Pokemon consist of Pokemon battle, catching Pokemon and gameplay of Pokemon.

In Chapter 4, we present the second example of using game refinement theory. We apply game refinement theory to three different sports. The first sports is baseball which can be considered as time limit sports. For the second sports, we choose boxing as a representative of fighting sports. Finally, the third sports is tennis which is score limit sports. We show how game refinement theory can improve entertainment impact in sports.

In Chapter 5, we present the third example of using game refinement theory. We apply game refinement theory to one of the most well-known gesture game, RoShamBo, and one of the most classic video game, Snake game. We found that game refinement theory can be used for finding comfortable settings in games. For example, what is the number of appropriate players in RoShamBo?

In Chapter 6, we change the aspect of improving entertainment impact in games, from rule designer as game refinement theory in previous sections did to AI designer. We found that improving the quality of AI can be one significant way to explore new player's experience. Therefore, this chapter presents a generic model for emotional AI in the domain of real-time multiplayer fighting games. We implement the model to our simulated game and the evaluation experiments are performed by human.

In Chapter 7, we present the new approach which was inspired from game refinement theory. We change the aspect from quantifying the enjoyment between games to quantifying the enjoyment between matches in the same game. We propose our new model and many experiments are performed. We believe that understanding the enjoyment of each match can lead us to improve the entertainment impact in games which is our purpose.

Finally, Chapter 8 gives the conclusion of this thesis. According to the outcomes from previous chapters, we found that the three approaches proposed work well as it is explained in each chapter and the results confirm.

Chapter 2

Game Refinement Theory

We present our first attempt to improve the entertainment impact in games. We use game refinement theory [128], a unique theory that has been used as a reliable tool for measuring the attractiveness and sophistication of the games considered. This theory was invented by Iida *et al.* and was proved by many publications [59] [150] [149].

This chapter provides the fundamental idea of game refinement theory and its previous studies. For our challenge, the applications of this theory to many domains of games will be explained in Chapter 3, Chapter 4 and Chapter 5.

2.1 Introduction

Many efforts have been devoted to the discovery of theoretical aspects of increasing attractiveness of games and its sophistication. An early work has been done by Iida *et al.* [59] to set the foundation for the direction. In the work, game refinement theory was proposed based on the concept of game outcome uncertainty. Game refinement theory is another game theory focusing on the attractiveness and the sophistication of games. A game progress model is analyzed by using mathematical model. Then, it was applied to various domains such as board game, video game and sports [150].

Game theory [39] originated with the idea of the existence of mixed-strategy equilibrium in two-person zero sum games. It has been widely applied as a powerful tool in many fields such as economics, political science and computer science. Classical game theory concerns the winning strategy from the player's point of view, whereas game refinement theory concerns the entertaining impact from the game designer's point of view.

In this section, we introduce a basic idea of game refinement theory and its measure called game refinement measure. Then, the previous works which apply game refinement theory to various domains are described.

2.2 Basic Idea of Game Refinement Theory

A general model of game refinement theory was proposed based on the concept of game information progress [128]. It bridges a gap between sports games and board games. We first describe a general model of game progress in order to derive a game refinement measure. Then, we apply this idea to various games while identifying reasonable game progress models of given games, and compare them using game refinement measures.

The game progress is twofold. One is game speed or scoring rate, while another one is game information progress with focus on the game outcome. Game information progress presents the degree of certainty of a game's result in time or in steps. Having full information of the game progress, i.e. after its conclusion, game progress $x(t)$ will be given as a linear function of time t with $0 \leq t \leq t_k$ and $0 \leq x(t) \leq x(t_k)$, as shown in Equation (2.1).

$$x(t) = \frac{x(t_k)}{t_k}\, t \tag{2.1}$$

However, the game information progress given by Equation (2.1) is unknown during the in-game period. The presence of uncertainty during the game, often until the final moments of a game, reasonably renders game progress as exponential. Hence, a realistic model of game information progress is given by Equation (2.2).

$$x(t) = x(t_k)(\frac{t}{t_k})^n \tag{2.2}$$

Here n stands for a constant parameter which is given based on the perspective of an observer of the game considered. Only a very boring game would progress in a linear function however, and most of course do not. Therefore, it is reasonable to assume a parameter n, based on the perception of game progress prior to completion. If the information of the game is completely known (i.e., after the end of the game) and the value of n is 1, the game progress curve appears as a straight line. In most games, especially in competitive ones, much of the information is incomplete, the value of n cannot be assumed, and therefore game progress is a steep curve until its completion, along with $x(t_k)$, t_k, $x(t)$ and t, just prior to game's end.

FIGURE 2.1: Decision tree of a two-person game

Then acceleration of game information progress is obtained by deriving Equation (2.2) twice. Solving it at $t = t_k$, we have Equation (2.3).

$$x''(t_k) = \frac{x(t_k)}{(t_k)^n}(t_k)^{n-2}\ n(n-1) = \frac{x(t_k)}{(t_k)^2}\ n(n-1) \tag{2.3}$$

It is assumed in the current model that game information progress in any type of game is encoded and transported in our brains. We do not yet know about the physics of information in the brain, but it is likely that the acceleration of information progress is subject to the forces and laws of physics. Too little game information acceleration may be easy for human observers and players to compute, and becomes boring. In contrast, too much game information acceleration surpasses the entertaining range and will be frustration, and at some points beyond that could become overwhelming and incomprehensible. Therefore, we expect that the larger the value $\frac{x(t_k)}{(t_k)^2}$ is, the more the game becomes exciting, due in part to the uncertainty of game outcome. Thus, we use its root square, $\frac{\sqrt{x(t_k)}}{t_k}$, as a game refinement measure for the game under consideration. We call it R value for short shown in Equation (2.4).

$$R = \frac{\sqrt{x(t_k)}}{t_k} \tag{2.4}$$

2.3 Game Progress Model

It is an important and challenging task to figure out a reasonable model of game progress and game information progress in a target game, from which a measure of game refinement can be derived. Below we give a short sketch of various game progress models, which have been investigated in some domains such as boardgames, time limit sports, score limit sports and fighting sports.

2.3.1 Board games

We consider the gap between board games and sports games by deriving a formula to calculate the game information progress of board games. Let B and D be an average branching factor (number of possible options) and average game length (depth of whole game tree), respectively. One round in board games can be illustrated as decision tree shown in Figure (2.1).

At each depth of the game tree, one will choose a move to progress. The distance d, which has been shown in Figure (2.1), can be found by using simple Pythaoras theorem, thus resulting in $d = \sqrt{\Delta l^2 + 1}$.

Assuming that the approximate value of horizontal difference between nodes is $\Delta l = \frac{B}{2}$. Then we can make a substitution and get $d = \sqrt{(\frac{B}{2})^2 + 1}$. The game progress for one game is the total level of game tree times d. For the meantime, we do not consider Δt^2 because the value $(\Delta t^2 = 1)$ is assumed to be much smaller compared to B. The game length will be normalized by the average game length D. Then the game progress $x(t)$ is given by $x(t) = \frac{t}{D} d = \frac{t}{D}\sqrt{(\frac{B}{2})^2} = \frac{Bt}{2D}$

Therefore, in general, R value for board game is shown in Equation (2.5).

$$R = \frac{\sqrt{B}}{D} \tag{2.5}$$

Iida *et al.* [59] calculate the game refinement values for various board games such as chess, Go and Mah Jong [58]. We show, in Table 2.1, the results.

TABLE 2.1: Measures of game refinement for major board games

Game	B	D	R
Western chess	35	80	0.074
Chinese chess	38	95	0.065
Japanese chess	80	115	0.078
Go	250	208	0.076
Mah Jong	10.36	49.36	0.078

2.3.2 Time limit sports

In the previous works, in time limit sport domains such as football and basketball, the game refinement measure R was calculated by $R = \frac{\sqrt{G}}{T}$ where G and T stand for the average number of goals and the average number of attacks or shot attempts respectively. Also, the values G and T correspond to $x(t_k)$ and t_k in the previous discussion. Therefore, we can find R value shown in Equation (2.6). Moreover, we show the results from [128] in Table 2.2.

$$GR = \frac{\sqrt{G}}{T} \tag{2.6}$$

Table 2.2: Measures of game refinement for basketball and football

Sports	G	T	R
Basketball	36.38	82.01	0.073
Football	2.64	22	0.073

2.3.3 Score limit sports

We consider another type of sports such as volleyball, badminton and table tennis. In these sports, there are no time limit but the game is regulated by a score limit. For this type of sport, we construct a game progress model with focus on W and T which stand for the average winner's scores and the average total scores of entire game respectively. So, the game refinement measure R was calculated as shown in Equation (2.7). The values W and T correspond to $x(t_k)$ and t_k in the previous discussion. We show the results from our previous studies in Table 2.3.

$$GR = \frac{\sqrt{W}}{T} \tag{2.7}$$

Table 2.3: Measures of game refinement for volleyball, badminton and table tennis

Sports	Version	W	T	R
Volleyball	Side-out system(15pts)	15	52.52	0.074
	Rally point system(30pts)	30	53	0.103
	Rally point system(25pts)	25	44	0.114
Badminton	Old scoring system	30.07	45.15	0.121
	New scoring system	46.34	79.34	0.086
Table tennis	Pre-2000	57.87	101.53	0.075
	Post-2000	54.86	96.47	0.077

2.4 Related Works

Vecer *et al.* [139] introduce a concept of measuring the excitement in sports games. They relate the excitement to the variability of the win expectancy. The larger is this variability, the higher is the excitement. The win expectancy varies more if there is a number of swings during the game, as opposed to a one-sided game. Win expectancy also changes more the closer to the end of the game a decisive event happens, or a more unexpected is the upset of a favorite team. They illustrated this concept at soccer games for which the theoretical win expectancy can be computed from a Poisson model of scoring.

This approach is promising to measure the excitement with focus on an individual game under consideration. Indeed it was employed in video processing domain [72]. However, our purpose is different from this direction. Our study does not focus on an individual game, but on large number of games to assess the sophistication of the game under consideration.

Rottenberg and Simon[111] note that the nature of sports is such that competitors must be of approximately equal ability if any are to be financially successful. In recent years, sports commentators and fans, Major League Baseball itself, and even some economists have expressed growing concern about the widening disparities among team expenditures and the growing concentrations of postseason contenders and championships.

Sanderson *et al.* [117] compare different concepts of competitive balance, review the theoretical and empirical scholarship on competitive balance and the relationship between payrolls and performance, describe the natural forces and institutional rules and regulations that contribute to observed distributions of playing performances, and evaluate the likely impact of several popular proposals payroll and salary caps, luxury taxes, and increased revenue sharing. on competitive balance. They made frequent comparisons to other sports leagues, including collegiate athletics and individual sports.

Newton *et al.* [94] develop a stochastic Markov chain model to obtain the probability density function for a player to win a match in tennis. By analyzing many data, they show that a player's probability of winning a point on serve and while receiving serve varies from match to match and can be modeled as Gaussian distributed random variables. The model is composed of four essential parameters, sample means associated with each player's probability of winning a point on serve and while receiving serve and its standard deviations. Next, the experiments are performed by running full tournament simulations of the four Grand Slam events. Also, they describe how to use the results as the basis for ranking methods with natural probabilistic interpretations.

Govan *et al.* [45] present a flexible, easily coded, fast, iterative approach to generate team ratings. This approach's name is Offense-Defense Model (ODM). O'Malley *et al.* [98] derive an expression for the probability of winning a game in a tennis match under the assumption that the outcome of each point is identically and independently distributed. In contrast to our approach, they try to construct a reasonable model to estimate the strength of the player or their team while our model try to quantify the entertainment of the game.

2.5 Conclusion

As many previous studies confirm, it is obvious that game refinement theory can effectively be applied in many domains of game such as classical board game, score limit sports and time limit sports. It can be used as a helpful tool to measure an attractiveness of a game and it also enables game designers to make a target game more sophisticated. This section provide some fundamental details of game refinement theory and its recent works. As a tentative conclusion, we observed that suitable game refinement value is around $0.07 - 0.08$, with many previous studies confirmed.

The next chapter will demonstrate how to apply game refinement theory to many domains of game beside the previous works. We will show how game refinement theory can be used as an essential tool to improve the entertainment impact in games which is one of the ways in our research.

Chapter 3

Comfortable Setting Identification in Pokemon Game

This chapter is an updated and abridged version of work previously published in

(1) Chetprayoon Panumate, Shuo Xiong and Hiroyuki Iida. An Approach to Quantifying Pokemon Entertainment Impact with Focus on Battle. In Applied Computing and Information Technology/2nd International Conference on Computational Science and Intelligence (ACIT-CSI), 2015 3rd International Conference on, pages 60-66. IEEE, 2015.

(2) Chetprayoon Panumate, Shuo Xiong, Hiroyuki Iida and Toshiaki Kondo. Evolutionary Changes of Pokemon Game: A Case Study With Focus on Catching Pokemon. In Entertainment Computing-ICEC 2015, pages 182-194. Springer, 2015.

(3) Ho Xuan-Vinh, Chetprayoon Panumate, Hiroyuki Iida and Jean-Christophe Terillion. Deciphering the Entertaining Impact of Pokemon. International Journal of Engineering Research and Management (IJERM), 2(11):83-90, 2015.

(4) Chetprayoon Panumate and Hiroyuki Iida. Finding Comfortable Settings of Video Games: Using Game Refinement Measure in Pokemon Battle AI. IPSJ SIG-GI Technical Report, vol. 2016-GI-36, no. 6, pages 1-7.

(5) Chetprayoon Panumate and Hiroyuki Iida. Developing Pokemon AI for Finding Comfortable Settings. Proceedings of 2016 Summer Conference, Digital Games Research Association JAPAN, August 2016, pages 168-171.

This chapter explores an innovative way to find a comfortable setting of video games. It is our first example of using game refinement theory to improve the entertainment impact of game. Pokemon, which is a turn-based role-playing game and one of the most popular video games, is chosen as a benchmark in this study. Pokemon has various entertaining aspects to be considered, of which we focus on three important aspects: Pokemon battle, catching Pokemon and gameplay in Pokemon.

3.1 Introduction

Every game would have its own suitable setting to maximize the entertaining impact, which makes the game different from others. However, it is a big challenge to identify such a suitable setting. For example, the rules of chess have been elaborated with many changes in its evolutionary history [23] [92]. We know through our experience in developing game systems that a setting is decided by some hidden mechanisms to make the game survive for many decades or to be more successful in the market.

Nowadays, there are many video games. Some are so popular but others are not so. There are various reasons behind this fact, e.g., its story, graphics etc. In a practical sense, one of the most important points is the setting of some variables in the game. Its setting may often considerably affect the entertaining aspect which leads the game to be exciting or boring. That is the reason why we believe that the setting of some variables in a game has its own reasons behind this, and therefore finding a comfortable setting can be significantly important to maximize the entertainment in playing the game under consideration. Thus, this study demonstrates an innovative way to find a comfortable setting of video games.

In this study, we use the game of Pokemon (see Figure 3.1) as a benchmark, which is one of the most popular video games [53]. While many efforts have been devoted to the study of Pokemon with focus on different aspects such as education [75], media science [96], social science [138] and computer science [20], the present study focuses on another important aspect of Pokemon: engagement or entertainment. We start with a foundational question: "Why has Pokemon been popular so long time?"

Actually, there are many factors to be considered in Pokemon because Pokemon world is very wide and it has many things that player can do and enjoy the game. Therefore,

FIGURE 3.1: A screenshot of Pokemon Emerald

we separate Pokemon into three main parts: (1) Pokemon battle, (2) catching Pokemon and (3) gameplay of Pokemon. Then, we focus on each part thoroughly by analyzing the related factors. Moreover, in each part there are still many factors to be considered. Some of them have been added, deleted or changed in order to improve the quality of Pokemon. For example, in Pokemon battle part, some abilities, some moves, the number of types and many things have been changed. We believe that to maximize the entertainment of the game, game designers have to find a comfortable setting for each factor. That is why some factors have been changed consecutively.

By applying game refinement theory to Pokemon with an appropriate game progress model, we can go deeply while focusing on two main research goals:

1. To find a reason why Pokemon has been popular so long time

2. To find comfortable settings of Pokemon

To tackle these challenges, we focus on three main parts: Pokemon battle, catching Pokemon and gameplay of Pokemon. For each part, we figure out a reasonable game progress model to derive an appropriate measure of game refinement. We then collect data using various methods. For example, some data are collected by computer simulation and AIs while some data are collected from reliable resource on Internet. Thus, the results obtained are analyzed in order to accomplish our main research goals.

The structure of this chapter is as follows. We first briefly introduce the game of Pokemon and its history in Section 3.2. By applying game refinement theory, the analysis of Pokemon battle is presented in Section 3.3. The analysis of catching Pokemon is described in Section 3.4. The analysis of gameplay of Pokemon is shown in Section 3.5. Finally, the results obtained are discussed and concluding remarks are given in Section 3.6.

3.2 The Pokemon Game

In this section, we first provide a brief history of the game of Pokemon. Then we present its overview.

3.2.1 History of Pokemon

Looking back to the history of video games, there are many video games which have been invented. In 1972, the first commercial video game console that could be played at home is Odyssey, which was released by Magnavox and designed by Ralph Baer. Since that time, video games truly became popular, with many game consoles and many game software developed. In order to see the influence of Pokemon, we cannot compare it with classical board games such as chess which has survived and evolved for more than 1500 years [92]. However, in the video game world, a game franchise which has prospered for some 20 years and continued being played globally must be not trivial. This game is a series developed by Game Freak and Creatures Inc. [1] [12]. First released in 1996 in Japan for the Game Boy, the main series of role-playing video games (RPG) has continued on each generation of handheld Nintendo consoles, including Game Boy Color, Game Boy Advance, Nintendo DS and Nintendo 3DS [7].

Pokemon is the most successful computer game ever made, the top globally selling trading-card game of all time and one of the most successful children's television programs ever broadcast [133]. While the main series consists of role-playing games, spinoffs encompass other genres such as action role-playing, puzzle and digital pet games. Up to now, the game has 721 Pokemons and six generations. Moreover, the seventh generation named Pokemon Sun and Pokemon Moon are announced to be released in 2016. We show, in Table 3.1, a historical overview of Pokemon. Based on this success, many different series of manga and anime follow. Even in the academic field, many papers have been written about Pokemon in different aspects. All of them try to analyze and explain why this franchise is growing so fast and makes everyone know the cute yellow mouse Pikachu [133].

3.2.2 Overview of Pokemon

All creatures living in the game world except humans are Pokemons. They can be captured by a Pokeball. When a Pokemon fights and defeats another Pokemon, the winner Pokemon is able to level up so that her stats increase and become stronger. One

[1] All products, company names, brand names, trademarks, and sprites are properties of their respective owners. Sprites are used here under Fair Use for the educational purpose.

Table 3.1: A historical overview of Pokemon

<table>
<tr><th>Generation</th><th>Year</th><th>Version</th></tr>
<tr><td rowspan="3">1st</td><td>1996</td><td>Pokemon Red & Pokemon Green</td></tr>
<tr><td>1997</td><td>Pokemon Blue</td></tr>
<tr><td>1998</td><td>Pokemon Yellow</td></tr>
<tr><td rowspan="2">2nd</td><td>1999</td><td>Pokemon Gold & Pokemon Silver</td></tr>
<tr><td>2000</td><td>Pokemon Crystal</td></tr>
<tr><td rowspan="3">3rd</td><td>2002</td><td>Pokemon Ruby & Pokemon Sapphire</td></tr>
<tr><td rowspan="2">2004</td><td>Pokemon Fire Red & Pokemon Leaf Green</td></tr>
<tr><td>Pokemon Emerald</td></tr>
<tr><td rowspan="3">4th</td><td>2006</td><td>Pokemon Diamond & Pokemon Pearl</td></tr>
<tr><td>2008</td><td>Pokemon Platinum</td></tr>
<tr><td>2009</td><td>Pokemon Heart Gold & Pokemon Soul Silver</td></tr>
<tr><td rowspan="2">5th</td><td>2010</td><td>Pokemon Black & Pokemon White</td></tr>
<tr><td>2012</td><td>Pokemon Black2 & Pokemon White2</td></tr>
<tr><td rowspan="2">6th</td><td>2013</td><td>Pokemon X & Pokemon Y</td></tr>
<tr><td>2014</td><td>Pokemon Omega Ruby & Pokemon Alpha Sapphire</td></tr>
<tr><td>7th</td><td>2016</td><td>Pokemon Sun & Pokemon Moon</td></tr>
</table>

of the most interesting Pokemon's characteristics is that she can evolve, which makes her stronger, learn new skills, and sometimes change to a new form. There are many different kinds of evolution. For example, a player has to raise his Pokemon's happiness, trade his Pokemon with another trainer or give a special item to his Pokemon.

Basically, the goal of Pokemon is to win the badges of gyms and become the champion of the league [76]. For this purpose, a player has to win every battle in the game. In each version, there is a common format. A player begins as a boy or girl who wants to become a Pokemon master. The player will accept the request by the professor to fulfill the Pokedex that is a portable device which provides information regarding the diversified species of Pokemon. The ultimate goal of Pokemon is to catch every Pokemon and make the Pokedex completed. To reach the goal, a player starts by choosing one of three starters to begin the journey. In the adventure, he cannot finish the game without challenging other trainers in a battle. Gym Leaders, Elite Fours and the Rival are the strongest trainers among them. Beside catching Pokemon as many as players can, a player needs to build his own team so that it becomes strong enough to win every single battle which the players are engaged in. If the player wins, the player will get not only the experience points but also the money to buy items to support his team.

3.3 Analysis of Pokemon Battle

Pokemon battle is the core part of Pokemon. To analyze Pokemon, we need to focus on this part firstly. Actually, there are many related factors which have been changed successively. However, a few factors are still remained after the first episode of Pokemon was released in 1996. Among them, the number of Pokemon ($n = 6$) that one trainer can carry has never been changed.

This section, therefore, first introduces an overview of Pokemon battle. For the assessment of settings of the game, we need to collect data for computing game refinement measure. For this purpose, the following two approaches are employed in this study.

1. Pokemon AI approach
2. Human data approach

The first approach is to collect data by using our simulated AI which is implemented on our Pokemon battle simulator. We perform many experiments in order to obtain a reliable R value. Moreover, in this approach, we can adjust n, the number of Pokemon that one trainer can carry, which is one of the important settings of Pokemon battle. In contrast, the second approach collects human data from reliable source. Finally, these analyzing can answer our main research purposes.

3.3.1 Overview of Pokemon battle

The goal of Pokemon [76] is to win the badges of gyms and become the champion of the league. For this purpose, one has to win every battle in the game. In this study, we focus on each battle. The goal of Pokemon battle is to fight until the opponent has no Pokemon remained. HP (hit points or health points) [90] is an attribute assigned to each entity in the game that indicates its state in combat. One Pokemon will be fainted if its HP reaches zero.

Each player will be called as a trainer because the duty of the player is to train his or her Pokemon to be powerful enough to clear the game. Each trainer can bring up six Pokemons and each Pokemon can remember four moves. Each Pokemon has six kinds of stats which consist of HP, attack, defense, special attack, special defense and speed. There are eighteen types of Pokemons and one Pokemon can be only one or two types such as Pikachu which is an electric type or Golem which is both rock type and ground type. The eighteen types consist of bug, dark, dragon, electric, fairy, fight, fire, flying, ghost, grass, ground, ice, normal, poison, psychic, rock, steel and water. The effectiveness of each move depends on the type of move and type of Pokemons that received the move. We show, in Table 3.2, some examples.

Normally, a trainer can choose what he/she wants to do among four choices: fight, switching Pokemon, using Items and run, as shown in Figure 3.2. However, in the

TABLE 3.2: Examples of types and its effectiveness

	Defending Type					
	Type	Electric	Fire	Grass	Ground	Water
Attacking Type	Electric	×0.5		×0.5	×0	×2
	Fire		×0.5	×2	×0.5	×0.5
	Grass		×0.5	×0.5	×2	×2
	Ground	×2	×2	×0.5		
	Water		×2	×0.5	×2	×0.5

FIGURE 3.2: A screenshot of Pokemon battle from Pokemon Emerald

Pokemon battle rules, a trainer cannot choose run or using item, so the trainer can choose only fight or changing Pokemon. There are many kinds of moves. It varies from offensive attacks like tackle to defensive moves like protect. Some attack moves are physical attacks which will use one's attack stats and the opponent's defense stats to be calculated the results such as Slam while some attack moves are special attack which will use one's special attack stats and the opponent's special defense stats to be calculated the results such as Psybeam. Some moves can change one's stats and/or the opponent's stats such as Tail Whip and Growl. Some moves inflict stats effects to the target like paralyzed and poison such as Thunder Wave and Toxic.

The order of turn to play is decided by the speed of the two Pokemons which are currently fighting. The faster one simply goes first. Switching Pokemon always can go before attacks. Moreover, each Pokemon has a special ability which will effect in Pokemon battle such as Sniper ability that let the critical hit more powerful. Each Pokemon can hold one item, the item makes holder more forceful such as Leftovers which make holder regains its HP every turn. The trainer who has any remained Pokemon while the opponent has no Pokemon is winner.

3.3.2 Collecting data

One possible way to collect many data of Pokemon battle which have been performed by human players is given by online Pokemon battle simulator [103]. It will be used in human data approach. However, in Pokemon AI approach, we need to change the number of Pokemons that one trainer can carry in order to find a comfortable setting. Therefore, we simulate the Pokemon battle game by simplifying many factors in the game.

3.3.2.1 Formulas used in Pokemon battle simulator

First, we introduce some equations from [15] and [121] which are essential to create a simulated Pokemon battle game. When we generate a Pokemon, its stat needs to be generated. Since one Pokemon is composed of six stats, its HP stat can be defined by Equation (3.1).

$$HP = \left\lfloor \frac{(2 \times Base + IV + \lfloor \frac{EV}{4} \rfloor) \times Lv}{100} \right\rfloor + Lv + 10 \tag{3.1}$$

Where IV stands for an individual value which is randomly generated by the game at the time when one meets that Pokemon first. There are six IV because each Pokemon has six battle stats. $Base$ means the initial value of that stat. There are six $Base$ because each Pokemon has six battle stats. To calculate HP, IV of HP and $Base$ of HP are used. For stats of Pokemon considered, it depends on the kind of ones Pokemon. Some Pokemons have outstanding $Base$ of HP while other Pokemons have poor $Base$ of HP. Rationally, Pokemon with poor $Base$ of HP should have another great initial stat. However, legendary Pokemon may have an excellent value in every initial stat. EV stands for a special value which Pokemon will receive after the battle. It depends on the kind of Pokemon which has been defeated. Some Pokemons give EV of HP while other Pokemons give EV of other stats. Lv denotes a level of Pokemon considered. It simply starts from 1 to 100. The more Pokemons level increases, the stronger Pokemon becomes.

Equation (3.2) shows how to calculate OS, which means other stats. Simply, other stats are calculated by using IV, EV and $Base$ of the stat considered.

$$OS = \left\lfloor \left(\left\lfloor \frac{(2 \times Base + IV + \lfloor \frac{EV}{4} \rfloor) \times Lv}{100} \right\rfloor + 5 \right) \times Nat \right\rfloor \tag{3.2}$$

Where Nat is the nature of the Pokemon considered. It is the mechanic that influences how a Pokemon's stats grow. There are 25 natures which are composed of hardy, lonely, brave, adamant, naughty, bold, docile, relaxed, impish, lax, timid, hasty, serious, jolly, naive, modest, mild, quiet, bashful, rash, calm, gentle, sassy, careful and quirky. Each nature has its own property. For example, adamant nature will increase the attack stat while reducing the special attack stat.

Equation (3.3) shows how to calculate damage.

$$Damage = (\frac{2 \times Lv + 10}{250} \times \frac{Atk}{Def} \times Base + 2) \times Mod \tag{3.3}$$

Where Lv is the level of the attacking Pokemon. Atk and Def are the working attack and defense stats of the attacking and defending Pokemon, respectively. If the attack is special, the special attack and special defense stats are used instead. $Base$ is the base power of the attack. Mod can be defined by Equation (3.4).

$$Mod = STAB \times Type \times Critical \times Other \times Random \tag{3.4}$$

Where $STAB$ is the same-type attack bonus. It is equal to 1.5 if the attack is the same type as the user, and 1 otherwise. $Type$ is the type effectiveness. This can be either 0, 0.25, 0.5, 1, 2, or 4 that is determined depending on the type of attack and the type of the defending Pokemon. $Critical$ is 1.5 if this is a critical attack, and 1 otherwise. $Other$ stands for the things like held items, abilities, field advantages, and whether the battle is a double battle or triple battle or not. $Random$ is a random number from 0.85 to 1.00.

However, some complex factors such as holding items component, abilities component and some complex moves are not considered. Finally, we simulate the Pokemon battle game. We implement 20 of the most used Pokemons from the total number of 718 Pokemons [145]. We also implement its most used moves according to reliable site [130].

3.3.2.2 Pokemon AI approach - developing Pokemon battle AIs

After finishing the simulated game, we have to implement Pokemon battle AI in order to perform experiments. We propose four kinds of AI which can be described below.

1. Random AI will choose at random every possible choice including the move and the change of Pokemons. This algorithm is simply described in Algorithm 1.

2. Attack AI will choose a move that makes the highest damage from the currently used Pokemon. It will not change the currently used Pokemon. This algorithm is described in Algorithm 2.

Algorithm 1 Random AI

```
procedure RandomAIDecision
    n1 ← number of Moves
    n2 ← number of remainning Pokemons
    r ← random from 0 to (n1 + n2 − 1)
    if r < n1 then
        return useMove(r)
    else
        return changePokemonTo(r − n1 + 1)
    end if
end procedure
```

Algorithm 2 Attack AI

```
procedure AttackAIDecision
    n ← list of Moves
    dmgTemp1 ← calDmg(n0, opponent)
    moveIndex ← 0
    for each ni in n do
        dmgTemp2 ← calDmg(ni)
        if dmgTemp2 > dmgTemp1 then
            dmgTemp1 ← dmgTemp2
            moveIndex ← i
        end if
    end for
    return useMove(moveIndex)
end procedure
```

3. Smart-Attack AI checks at first whether or not the currently used Pokemon has a move that wins the opponent's type. If yes, it selects the best move that makes the highest damage among the moves that currently used Pokemon has. If not, it checks whether or not it has other Pokemons that have a move that wins the opponent's type. If yes, it changes the currently used Pokemon with other Pokemon that has a move that makes the highest damage to the opponent. If not, it has to select the move that makes the highest damage from the currently used Pokemon inevitably. This algorithm is described in Algorithm 3.

4. Smart-Defense AI checks at first whether or not the currently used Pokemon's type wins the opponent's type. If yes, it selects the best move that makes the highest damage from the currently used Pokemon same as Smart-Attack AI. If not, it checks whether or not it has other Pokemons whose type wins the opponent's type. If yes, it changes the currently used Pokemon with the Pokemon whose type wins the opponent's type and has a move that makes the highest damage to the opponent. If not, it has to select the move that makes the highest damage from the currently used Pokemon inevitably. This algorithm is described in Algorithm 4.

Algorithm 3 Smart-Attack AI

```
procedure SmartAttackAIDecision
    n ← list of Pokemons
    opponent ← opponent's current Pokemon
    moveIndex ← findMaxDmgMove()
    if hasMoveWinType(n_0, opponent) then
        return useMove(moveIndex)
    else
        changePokemon ← false
        dmgTemp ← 0
        pokemonIndex ← 1
        for each n_i in n do
            if hasMoveWinType(n_i, opponent) then
                changePokemon ← true
                dmgTemp2 ← findMaxDmg()
                if dmgTemp2 > dmgTemp then
                    dmgTemp ← dmgTemp2
                    pokemonIndex ← i
                end if
            end if
        end for
        if changePokemon then
            return changePokemon(pokemonIndex)
        else
            return useMove(moveIndex)
        end if
    end if
end procedure
```

Smart AI is designed with the purpose to create a simple human-like Pokemon battle AI. We create two types of Smart AIs: Smart-Attack AI and Smart-Defense AI. The significant difference between Smart-Attack AI and Smart-Defense AI is that Smart-Attack AI checks type from its moves while Smart-Defense AI checks type from Pokemon. So, Smart-Attack AI would make higher damage because the damage is based on the move. However, Smart-Defense AI is safer because if your Pokemon's type wins opponent's Pokemon's type, it is hard that your opponent can make high damage to you. This significant difference leads Smart-Attack AI to complete the game faster than Smart-Defense AI.

3.3.2.3 Human data approach - using Pokemon Showdown

Another way to collect data of Pokemon battle is to collect from human players. Pokemon Showdown [2] [103] is an online Pokemon battle simulator. It allows us to play

[2] All products, company names, brand names, trademarks, and sprites are properties of their respective owners. Sprites are used here under Fair Use for the educational purpose.

Algorithm 4 Smart-Defense AI

```
procedure SmartDefenseAIDecision
    n ← list of Pokemons
    opponent ← opponent's current Pokemon
    moveIndex ← findMaxDmgMove()
    if winType(n_0, opponent) then
        return useMove(moveIndex)
    else
        changePokemon ← false
        dmgTemp ← 0
        pokemonIndex ← 1
        for each n_i in n do
            if winType(n_i, opponent) then
                changePokemon ← true
                dmgTemp2 ← findMaxDmg()
                if dmgTemp2 > dmgTemp then
                    dmgTemp ← dmgTemp2
                    pokemonIndex ← i
                end if
            end if
        end for
        if changePokemon then
            return changePokemon(pokemonIndex)
        else
            return useMove(moveIndex)
        end if
    end if
end procedure
```

Pokemon battles online freely, while playing with randomly generated teams, or building one's own as shown in Figure 3.3. We can collect data from this source.

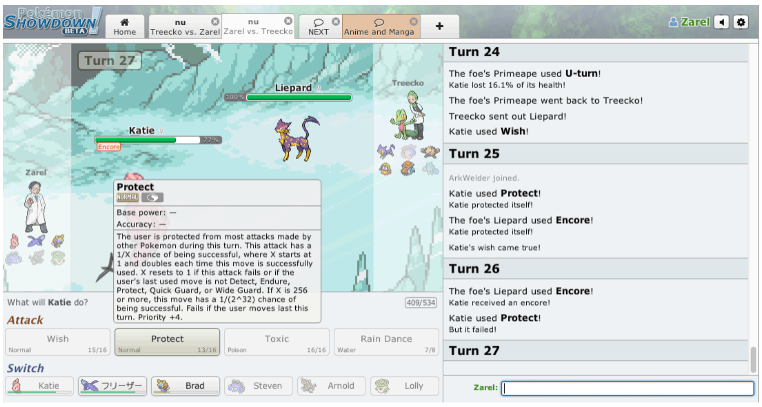

Figure 3.3: A screenshot of Pokemon Showdown

3.3.3 Board game approach

In Chapter 2, the basic idea of game refinement theory was presented. In order to apply game refinement theory to Pokemon battle, we use the board game approach.

Pokemon battle is a turn-based game where each player has to choose what to do at his/her turn, same as board games. So we can suppose Pokemon battle as a board game [99]. However, there are many important differences between Pokemon battle and board games, as described below.

One-to-one fighting

Pokemon battle is one-to-one fighting. This means that each player can control only one Pokemon to fight with the opponent's Pokemon. If one wants to use another Pokemon, he/she has to change his/her current used Pokemon. [3] Chess, a typical board game, is a kind of war with a lot of pieces in the battlefield. One can choose which piece that he/she wants to control in his/her turn. Thus, chess is run by many pieces against many pieces of the opponent.

Position

Chess has a position on a board at each turn. This is a very important factor for board games. That is why we call 'board' game. But in Pokemon battle, there is no such position. A game is normally run by one Pokemon per player, i.e., one's Pokemon and opponent's Pokemon. They play what they want to do, whereas no positions are considered.

Turn's sequence and turn's meaning

Turn in chess means the chance that one can do what he/she wants. Turn is run by switching system. $A's\,turn \rightarrow B's\,turn \rightarrow A's\,turn \rightarrow B's\,turn \rightarrow ...$, simple like this. In Pokemon battle, one turn means that both players have to choose what they want to do. Then, the speed (one of Pokemon's stats) of current used Pokemon is the factor which is decided on who will first start this turn. Pokemon with a faster speed can do first in that turn, then Pokemon with a lower speed will do afterward. To summarize the meaning of turn, one turn in chess is for only one side player's action (one ply), whereas one turn in Pokemon battle means simultaneous actions of both players. As for

[3] There are two-to-two or three-to-three Pokemon fighting mode but not so popular.

the turn's sequence, it is a simple order in chess but it is ordered in Pokemon battle by speed of current used Pokemon.

Hence, game-refinement theory can be applied by using the same idea as board games. Normally, we find R value by using the average possible options (say B) and game length (say D), as shown in Equation. (2.5). For possible options B, in Pokemon battle, they are very limited because one Pokemon can remember only four moves and player can change his/her current used Pokemon with other Pokemons that player possesses. In case of using some items such as medicine and potion, it is illegal in Pokemon contests, so we do not consider that case. For example, basically, one can carry at most six Pokemons, so he/she can choose five possible options when he/she wants to change the current used Pokemon. Therefore, we may use possible options equal to 9 which comes from four moves and five remained Pokemon. However, if the time passes, the number of Pokemons that player can change will be reduced because their Pokemons are fainted due to the battle. Also, the possible options will be reduced. Therefore, we can find the average number of possible options by the summation of possible options at each turn divided by the number of turns.

For game length D, the meaning of one turn in Pokemon battle is both players' simultaneous actions. Hence, there are two actions in one turn. So, we have to multiply 2 to this value. Finally, with this simple method, we can completely find R value as shown in Equation (3.5).

$$R = \frac{\sqrt{B}}{D} = \frac{\sqrt{Average\_number\_of\_options}}{Average\_number\_of\_turns \times 2} \tag{3.5}$$

However, for human data, due to its limitation, we cannot collect the $Average\_number\_of\_options$ thoroughly. Therefore, we estimate it by using $B = 9$ constantly. So, it will be shown in Equation (3.6).

$$R = \frac{\sqrt{B}}{D} = \frac{\sqrt{9}}{Average\_number\_of\_turns \times 2} \tag{3.6}$$

3.3.4 Application to Pokemon battle

We show the application of our two approaches: Pokemon AI approach and Human data approach respectively.

3.3.4.1 Pokemon AI approach

Experiments are performed by adjusting the number of Pokemons that one trainer can carry, denoted by n. We use AI in our simulated Pokemon battle game in four algorithms, described in Section 3.3.2.2. For each n in each type of AI, we perform one million games. Then, we collect B and D. We then calculate game refinement values by using Equation (3.5) and the results are shown in Table 3.3.

Table 3.3: Measures of game refinement for Pokemon battle

n	Random AI	Attack AI	S-Atk AI	S-Dfs AI
1	0.222	0.296	0.296	0.296
2	0.124	0.173	0.169	0.160
3	0.083	0.127	0.119	0.107
4	0.060	0.103	0.093	0.079
5	0.046	0.088	0.076	0.063
6	0.037	0.077	0.065	0.052
7	0.031	0.069	0.057	0.044
8	0.026	0.063	0.051	0.039
9	0.022	0.059	0.046	0.035
10	0.020	0.055	0.043	0.032

3.3.4.2 Human data approach

From the samples of 250 games, we obtained $Average\_number\_of\_turns = 25.796$ with a standard deviation $SD\_number\_of\_turns = 11.53$, the results are given in Equation (3.7).

$$R = \frac{\sqrt{9}}{25.796 \times 2} = 0.058 \tag{3.7}$$

Interestingly, if we apply the board game approach to Smart-Defense AI by setting $n = 6$ and using this approach, setting $B = 9$ constantly. We obtain $R = 0.058$ which is the same as the human data. It can confirm that our simple Smart-Defense AI can be a rough representative of Pokemon human player.

3.3.5 Discussion on Pokemon battle

The results obtained from the previous section will be discussed in this section. We will discuss in two aspects based on our research questions.

3.3.5.1 Analyzing Pokemon battle

In the previous studies, it is found that sophisticated games would have R value between $0.07-0.08$. We see that the results from Table 3.3 when we set $n = 6$ and Equation (3.7) show slightly lower values for Pokemon battle. It may imply that Pokemon is one of great games, with a twenty years history and many best sale done before [53], and Pokemon is suitable for children. Because in general, the appropriate game refinement value calculated for sophisticated games is around 0.07-0.08 and if it is more than this, it will be too much excited which is suitable for especial viewer such as Boxing which is extremely exciting sport. For Pokemon battle, the R value is lower than 0.07 and we conclude that this is suitable for children which is same as the target of Pokemon game.

However, in Pokemon battle, we explore that game refinement value $R = 0.061$ and $R = 0.058$ can be improved. For example, branching factors $B = 9$ is too narrow. If we improve this point, game refinement value will increase.

- Increasing the number of Pokemons: Since the first episode of Pokemon, Pokemon Red and Pokemon Blue in 1996, one trainer (trainer means the Pokemon's owner) can carry only six Pokemons. Of course, they can catch many Pokemons, but they cannot carry more than six Pokemons. In one battle, trainers have to choose what Pokemon they want to use in this battle. And normally, one player will maintain only six Pokemons, because they can use only six in battle. So, there is no reason to maintain more than six Pokemons. By increasing the number of Pokemons that one trainer can carry, trainer will have more number of Pokemons in battle. This means that trainers will have more element, more technics and more excited to play.

- Increasing the number of moves: One Pokemon always can remember only four moves, since the first episode, this is the essential limitations of Pokemon. If we increase the number of moves that one Pokemon can remember, maybe from four to five or more, the game will have more possible options to choose at one's turn, so the battle will be more complex and more excited.

- Controlling HP and another status: Due to $R = \frac{\sqrt{B}}{D}$, we can also increase R value (or decrease, for some purpose) by changing some parameters related to D. For example, if we increase HP, that means the game is longer and the value of D is more. Then, R value is lower. If we increase attack status or special attack status that means each Pokemon may be fainted easier and the game is shorter. Then, D is lower and R value is higher. We can apply the same idea to another status such as defense or special defense, both increase and decrease, it depends on what R value we want.

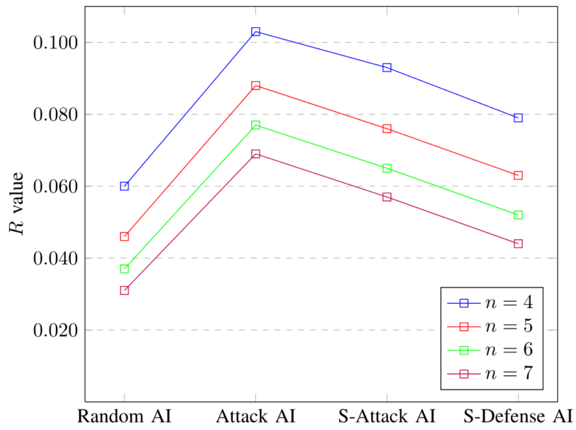

FIGURE 3.4: R values and performance level for various number of Pokemons (n)

3.3.5.2 Finding a comfortable setting

We will find the number of Pokemons that one trainer can carry n by analyzing the data by Pokemon AI approach, as shown in Table 3.3.

For Random AI, due to its random algorithm, there are many times that it selects changing Pokemon or selects some bad moves which are not effective. These decisions make the game end slower than expected. Actually, real human player would select a reasonable choice which makes the game end faster than Random AI. In contrast, Attack AI would select the best choice which maximizes the damage without changing Pokemon and it leads the game to end very fast. Practically, real human players should not be able to end the game as fast as Attack AI can. Also, human players should not be able to end the game as slow as Random AI does. Therefore, we can simply conclude that these two kinds of AI can show the possible range of R value based on player's skill. Random AI and Attack AI will be a lower and upper bound, respectively.

To go deeply, we simulate two types of Smart AI: Smart-Attack AI and Smart-Defense AI. It is expected that the two AIs show human-like performance. We can roughly classify Pokemon's player in many levels such as novice, beginner, normal and expert. Novice players would play like Random AI due to the very poor knowledge, so they do not know how to choose a good move. Beginner players would play like Attack AI because they try to end the game as fast as possible by choosing the maximum damage move which is not hard to calculate. Normal player would play like Smart AI by using simple decision making as we describe. However, higher level players likes expert players would know that Pokemon battle is more complex than other levels think. So, these level players would choose many complex moves which make the game somehow long but not so long as Random AI does. We show, in Figure 3.4, the relation between R-value and performance level for various number of Pokemons that a trainer can carry.

According to our previous studies, it is found that many sophisticated games have R value between $0.07 - 0.08$ which may be suitable for normal viewers. It suggests that for the Pokemon battle $n = 5$ may be the best setting. On the other hands, it is observed in [99] that the main target of Pokemon game is children. So its R value should be slightly lower than the sophisticated zone in order to have the appropriate level of excitement. Hence it is reasonably assumed that the appropriate R value for Pokemon battle is around $0.06 - 0.07$. Under the assumption, $n = 6$ is the best setting.

Moreover, $n = 5$ and $n = 7$ are also reasonable settings for some specific levels of Pokemon player. For example, if we focus on beginner's level, $n = 7$ may be nice. If we focus on expert's level, $n = 5$ may be fine. However, when focusing on children's performance, we see that $n = 6$ has the best R value because the range is very comprehensive. It can hold an acceptable value for novice, beginner, normal and expert. This is the answer to the research question in this study: Why the number of Pokemons that one trainer can carry is six? Because it is an appropriate number which is optimal for every level of players. This is the reason why the number six has never been changed after the first episode of Pokemon was released in 1996.

Our approach to find comfortable settings of Pokemon can be summarized as follows. We first simulate a Pokemon battle game by simplifying many factors. The simulated game should be as simple as possible, but it should be sufficiently complex in order to create a realistic simulated game. Then, we try to implement some realistic AIs for gathering data. However, it is hard to make a perfect human-like AI. Therefore, our solution is that we create some AIs which can be a lower bound and the upper bound of the human performance. Then, we try to create a human-like AI as realistic as possible. We can estimate the real human's performance based on his level by considering these AIs. Finally, a reasonable game refinement model is applied and we perform our experiment by the numerous games.

3.4 Analysis of Catching Pokemon

Catching Pokemon [100] is an important part that makes Pokemon very attractive and widely popular. It makes Pokemon unique from other games because player can catch Pokemon and build his/her own team. We raised the fundamental research question: "why has catching Pokemon mechanism been changed so many times in its history?" It means that game designers of Pokemon try to find a comfortable setting of catching Pokemon mechanism by changing its equations many times. Therefore, in this part, we quantify the attractiveness of catching Pokemon in every episodes based on the game refinement measure. We figure out a reasonable model of game information progress to

FIGURE 3.5: A screenshot of catching Pokemon

derive a game refinement measure for catching Pokemon, while we consider the prize cost to catch Pokemons. We first explain the basic idea of catching Pokemon such as some essential formulas. Next, we propose our new model called game refinement model with consideration on prize cost. By applying this model, analyzed results obtained from various episodes are discussed.

3.4.1 Overview of catching Pokemon

At the very initial stage to start a game, a Pokemon is given as a starter Pokemon for the coming adventure. A player may be able to catch other Pokemons by his/her effort, except some Pokemons given automatically due to the story of the game. Importantly, the final goal of Pokemon is to catch every Pokemon and make one's Pokedex, a portable device which provides information regarding the diversified species of Pokemon, completed. Moreover, Table 3.1 shows that the number of Pokemons incessantly increases. We therefore understand that catching Pokemon is one of the most important parts in Pokemon.

Catching Pokemon, as shown in Figure 3.5, can be simply described that a player has to reduce the current HP of a target Pokemon as much as possible. When HP of a player character reaches zero, the player may lose a life or their character might become incapacitated or die. So, to catch Pokemon, it is reasonable that the more Pokemon's HP is reduced, the weaker the target Pokemon is. Then, the player has to throw a ball, working as a catching device, to the Pokemon. Importantly, if the Pokemon is fainted, the catching attempt is unquestionably failed. Additionally, using high quality ball or giving bad stats to target Pokemon makes it easier to be caught.

3.4.2 Formulas used in catching Pokemon

As summarized in Table 3.1, Pokemon has six generations and the seventh generation is released in 2016. Each generation has own different catch rate mechanism except the third generation and fourth generation, where these two generations follow the same mechanism. Below we show the detail for each catching mechanism [32].

$$P_1 = \frac{S}{B} + \frac{\min(C+1, B-S)}{B} \times \frac{\min(255, F)+1}{256} \tag{3.8}$$

Where, $F = \left\lfloor \frac{\lfloor \frac{M \cdot 255}{G} \rfloor}{\max(1, \lfloor \frac{H}{4} \rfloor)} \right\rfloor$

$$P_2 = \frac{\max(\frac{(3M-2H) \cdot C}{3M}, 1) + S + 1}{256} \tag{3.9}$$

$$P_{34} = \frac{\frac{(3M-2H) \cdot CB}{3M} \cdot S}{255} \tag{3.10}$$

$$P_5 = \frac{\frac{(3M-2H) \cdot GCB}{3M} \cdot S \cdot \frac{E}{100}}{255} \tag{3.11}$$

$$P_6 = \frac{\frac{(3M-2H) \cdot GCB}{3M} \cdot S \cdot O}{255} \tag{3.12}$$

P_i stands for probability of catching Pokemon at ith generation. S is a variable for additional status. Normally, it is easiest to catch a Pokemon when its status is either asleep or frozen. The difficulty increases if the status is poisoned, burned, or paralyzed. The status none is the hardest case because it means that Pokemon is now very strong and ready to break any balls. C is a capture rate. Every Pokemon has its own capture rate status between 3 and 255. The value 3 means that it is very hard to catch that Pokemon, it is for super rare or legendary Pokemon. The value 255 means that it is very easy to catch, it is for common Pokemon which can be found regularly. B is a variable for ball used. There are many kinds of ball in this game. Some balls have special property which fit for some Pokemons whereas it also does not fit for other Pokemons. In this experiment, we focus on three kinds of common balls: Poke Ball, Great Ball and Ultra Ball, which can be bought in mart.

G (for P_1) represents a variable for Great Ball modifier. Due to some bugs in the first generation, this variable makes Great Ball has a higher average catch rate than Ultra ball even though Ultra Ball is more expensive. G (for P_5 and P_6) is a variable for grass modifier. It depends on the place where the player meets Pokemon. For example, if the action catching Pokemon takes place in thick grass, it is harder than normal grass. E denotes Entralink power. During normal gameplay this value is not effective. However, by playing Entralink missions with their friends over local wireless, they can receive capturing power from another player, which enables them to increase the chance of catch rate. O stands for O-Power bonus. This value replaces the Entralink modifier of the fifth-generation games to factor in Entralink powers sixth generation analogue, O-Powers.

M means maximum HP. It can be exactly calculated by Equation (3.13) for first and second generation, and by Equation (3.14) for third generation onward. H stands for current HP. It is reasonable that the more Pokemon's HP is reduced, the easier target Pokemon is caught. Importantly, if one makes the target Pokemon fainted, the catching attempt is unquestionably failed.

$$HP_{12} = \frac{(IV + BaseHP + \frac{EV}{8} + 50) \times Level}{50} + 10 \tag{3.13}$$

$$HP_{3456} = \frac{(IV + 2 \cdot BaseHP + \frac{EV}{4} + 100) \times Level}{100} + 10 \tag{3.14}$$

IV is an individual value which is randomly generated by the game at the time when one meets that Pokemon first. There are six IV due to each Pokemon has six battle status. To calculate HP, IV of HP is used. $BaseHP$ means initial HP status of Pokemon considered. It depends on what kind of one's Pokemon is. Some Pokemons have outstanding BaseHP while other Pokemons have poor $BaseHP$. Rationally, Pokemon with poor $BaseHP$ should have another great initial status. However, legendary Pokemon may have excellent value in every initial status. EV stands for special value which Pokemon will receive when finishing a battle. It depends on what kind of Pokemon has been defeated. Some Pokemons give EV of HP while other Pokemons give EV of another status. $Level$ denotes a level of Pokemon considered. It simply starts from 1 to 100. The more Pokemon's level increase, the stronger Pokemon becomes.

Pokemon Catch Rate Calculator [32] is an application that enables us to calculate Pokemon catch rate in many situations and every generations. We use this tool for calculating the chance of catching Pokemon. In the next section, we show our new model, game refinement model with consideration on prize cost, which is applied to catching Pokemon.

3.4.3 Game refinement model with consideration on prizing cost

In order to figure out a suitable game progress model for catching Pokemon, we need to apply the old approaches described in Chapter 2. While an early work [21] focuses on the playing cost, here we focus on the prizing cost. We propose V as a value of each prize captured and P as a probability of successful capturing the prize. Then, the game information $x(t_k)$ can be described as the average of P and V, as given in Equation (3.15).

$$x(t_k) = \frac{1}{n} \sum_{0<i<n} P_i V_i \tag{3.15}$$

Next, we apply our $x(t_k)$ in game refinement measure R in Equation. (2.4). For this case, we have to calculate by percentage as shown in Equation (3.16).

$$R = \frac{1}{10} \sqrt{\frac{1}{n} \sum_{0<i<n} P_i V_i} \tag{3.16}$$

In order to apply this model to catching Pokemon, V, a value of each prize captured, can be calculated by the degree of rareness of target Pokemon. Normally, each Pokemon has its own capture rate which shows how hard to capture the target Pokemon, which means like a rareness of Pokemon. Therefore, we propose an equation for calculating the rareness of each Pokemon, V, as shown in Equation (3.17)

$$V = \frac{Max - Cap + Min}{Max - Avg} \tag{3.17}$$

Here Max means the maximum of capture rate and Min means the minimum of capture rate. Likewise, Avg means the average of capture rate and Cap means that target Pokemon's capture rate. Generally, the minimum of capture rate in Pokemon game is 3, which means that it is very hard to catch. The maximum of capture rate in Pokemon game is 255 which means that it is very easy to catch this Pokemon. For the average of capture rate, it can be directly calculated which equals 100.25. Therefore, Equation (3.17) can be reduced in Equation (3.18).

$$V = \frac{256 - Cap + 3}{256 - Avg} \tag{3.18}$$

For probability of successful prize capturing, P, it can be calculated by Pokemon Catch Rate Calculator [32]. To use this calculator, we reasonably assume some parameters as explained below.

- Used Pokemon: We use 110 Pokemon samples which come from the first episode of Pokemon because we cannot use Pokemon from new episode to the first episode Pokemon Catch Rate Calculator. Moreover, the first episode of Pokemon contains 151 Pokemons but some Pokemon can be found from evolution. So we considerably cut some Pokemons which cannot be found as natural one.
- Current HP: We use 50 percent of full HP. It is the middle from 0 to 100.

- Pokemon level: It should be the average level of natural Pokemon in every episode which is approximately calculated by the lowest level and the maximum level of natural Pokemon in every episode. Finally, we already calculated average Pokemon level equals 25.69, approximately 26.

- Ball: There are many kinds of ball in Pokemon game. In this experiment, we focus on three main balls: Poke Ball, Great Ball and Ultra Ball.

- Status: Generally, if a Pokemon is asleep or frozen, it will be easiest to catch. If it is poisoned, burned or paralyzed, it is easier to catch but harder than asleep or frozen status. The hardest status for catching is none status. Hence, we use paralyzed status as an average status.

3.4.4 Application to catching Pokemon

Under the assumptions above, we obtain the average of P for each generation and ball used as shown in Table 3.4. Finally, we apply our new game refinement model with

Table 3.4: Average catch rate

Generation	$R_{PokeBall}$	$R_{GreatBall}$	$R_{UltraBall}$
1st	39.14	63.98	52.55
2nd	33.36	42.60	47.86
3rd & 4th	53.33	67.12	74.85
5th	57.03	69.54	76.50
6th	57.07	69.51	76.50

consideration on prize cost to Catching Pokemon. According to Equation (3.16), the results are shown in Table 3.5.

Table 3.5: Measures of game refinement for catching Pokemon with three main balls

Generation	$R_{PokeBall}$	$R_{GreatBall}$	$R_{UltraBall}$
1st	0.047	0.062	0.059
2nd	0.042	0.050	0.056
3rd & 4th	0.054	0.064	0.071
5th	0.058	0.067	0.072
6th	0.058	0.067	0.072

3.4.5 Discussion on catching Pokemon

For our new game refinement model with consideration on prize cost as shown in Equation (3.16), we can consider it as a new approach to quantify attractiveness of games.

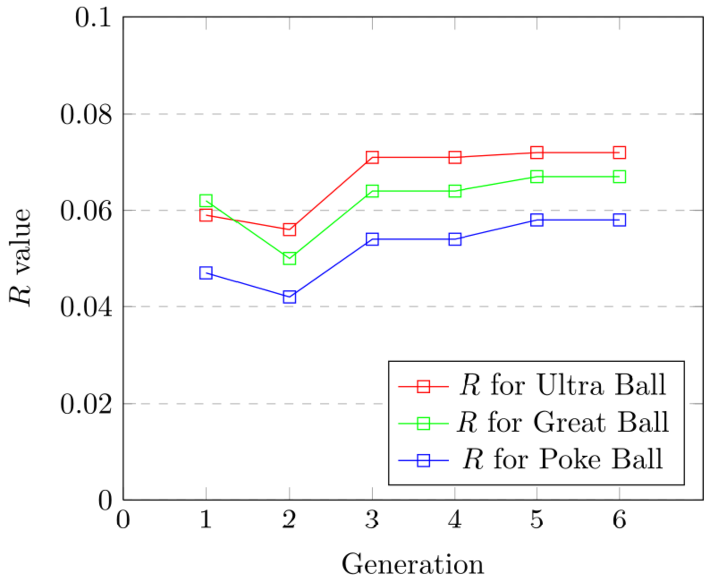

FIGURE 3.6: Changes of game refinement values with three main balls compared

With this model, it enables us to explore new domains of game which cannot be investigated by the previous models. The core of this model consists of two parameters, P that stands for the probability of successful capturing and V that stands for a value of each prize captured. For P, if we know the equation it can be calculated with the answer as the probability. If not, it can be calculated by simulating and collecting the data. For V, we have to carefully consider on what the related parameters of the prize's value are. Then, we establish the reasonable equation in order to measure the value of each prize.

Moreover, it has to be calculated in percentage system. We can apply this model to another game by creating reasonable equation. For the application to catching Pokemon, we collected data of catching Pokemon from Pokemon Catch Rate Calculator [32]. Then, we applied game refinement theory in the manner prescribed in Section 3.4.3. The results in Table 3.5 are compared in Figure 3.6.

As for Poke Ball, we can see that in first generation the R value is quite low. Furthermore, in second generation, the R value is lower than first generation. Nevertheless, R value is extremely increased in third generation and continuously go this way in fifth and sixth generation. Finally, it reaches 0.058 which is the maximum of R value using Poke Ball in sixth generation.

As for Great Ball, we can see that in first generation the R value is not too low. However, in second generation, the R value is awfully decreased from 0.062 in first generation to 0.50. In third generation, R comes back to 0.064 which is slightly higher than first generation. Then, it continuously increases and finally reach 0.067 in sixth generation.

As for Ultra Ball, we can see that in first generation R value is rather low. It then decreased from 0.059 in first generation to 0.056 in second generation. However, in third generation, R significantly increased to 0.071 which is more than another R value from another ball in another generation mentioned before. Then, it continuously increased to 0.072 in sixth generation which falls between the appropriate ranges of game refinement value: $0.07 - 0.08$.

Next, we consider each generation applied. According to Figure 3.6, in first generation, due to some errors, although Ultra ball is the best ball in these three kinds of balls but it has lower average catch rate and R value than Great ball. That means in first generation some mechanism need to be fixed. In second generation, we can see that R value for Ultra Ball is larger than Great Ball. That means the problem in previous generation was fixed. However, in second generation, every ball has trivially lower R value than other generations. That means the second generation's mechanism which is fixed from first generation does not work well and it should be fixed again. Suddenly, in third generation, R value is remarkably developed and every ball in this generation has quite good R value. Moreover, third generation catching mechanism is used in fourth generation which confirm the efficiency of this mechanism. For fifth and sixth generation, even though it uses the different catching mechanism but the calculated R value is closely same. The calculated R value of these mechanisms is very fine.

Moreover, we can consider the equation, both catching Pokemon equation and HP equation, directly. We see that the catching Pokemon mechanism which has been changed so many times try to add new parameters. For example, in fifth generation, Equation (3.11), proposing E which is Entralink power. This enables players to receive a capture power from another player. This variable is changed to O which is an O-Power bonus in sixth generation, Equation (3.12). We will see that both E and O increase a chance of capturing which may increase R value.

To compare HP equation, we consider the modification from HP_{12} to HP_{3456} by reducing Equation (3.14) to Equation (3.19).

$$HP_{3456} = \frac{(\frac{IV}{2} + BaseHP + \frac{EV}{8} + 50) \times \frac{Level}{2}}{50} + 10 \tag{3.19}$$

By comparing Equation (3.13) with Equation (3.19), we will see that $BaseHP$ and EV has the same coefficient while IV and $Level$ is changed. With this modification each Pokemon has lower HP value which trivially makes catching Pokemon easier.

Nevertheless, to get the more exact R value, we should apply the idea of focusing on playing cost from early work [21]. In catching Pokemon, we can apply this idea by considering the cost of ball. We know that Poke Ball's cost is 200, Great Ball cost is 600 and Ultra Ball cost is 1200. These costs have not been changed since the first generation of Pokemon until now. However, we cannot directly use the pure playing cost value because it will lead the R value to be unreliable. We need to propose a reasonable function such as prize cost factor which was shown in Equation (3.16). To propose a reasonable function of playing cost, we need more data from other balls. Therefore, this work ignores this point and further works should carefully consider this issue.

Moreover, the experiments conducted in this study have many assumptions mentioned before. Further investigation may be made from difference aspects. Below we show a few examples.

- Used Pokemon: Normally, each episode will have its local Pokemon, a Pokemon that can be found in nature in that episode. We may consider each episode by those local Pokemons. So, each episode will use different Pokemon.
- Current HP: Current HP may be given randomly.
- Pokemon level: By following the used Pokemon assumption, we will consider each episode so that we can use each episode's average level of natural Pokemon as a Pokemon level in the experiment. So, each episode will use different Pokemon level.
- Ball: We may focus on another ball.
- Status: We may assign a status for each Pokemon at random.

In conclusion, We can see that in first generation the result shows slightly lower values but it gradually increases in third generation and also in fifth and sixth generations. Finally, for Ultra Ball in third to sixth generation, the calculated R value of catching Pokemon falls between $0.07 - 0.08$ which has been supposed to be a reliable game refinement value for many games that have undergone sophistication. The slightly lower value for the first and second generations implies that at that time the balance of catching mechanism, the relation between rareness of Pokemon and chance of catching was not optimized yet. Therefore, the catching mechanism was developed successively and it eventually reaches the appropriate range of R value. However, the experiments in this study was performed with a simple model under many assumptions. Further works may try to focus on playing cost or use new assumptions.

3.5 Analysis of Gameplay in Pokemon

Besides Pokemon battle and catching Pokemon, this section considers about the effect of the setting of gameplay in Pokemon [151]. There are many settings of gameplay in Pokemon such as the kind and level of wild Pokemon and also the kind and level of Pokemon used by trainer. These settings can lead the game to be hard or easy. The appropriate level of difficulty would be the best setting of gameplay in Pokemon which game designer should expect. That is why is has been changed in every episode of Pokemon.

Therefore, in this part, we present an approach to find comfortable settings of gameplay in Pokemon. We wonder that what the trend of the changing of setting of gameplay in Pokemon is. So, we consider a suitable model of the game information progress to derive the game refinement value of Pokemon. Then we apply our new game refinement model, an experience based-model, to three version of Pokemon games consist of Pokemon Red, Pokemon Fire Red and Pokemon Soul Silver. If the game refinement values are in the window, we can carefully deduce that this game is entertaining enough to attract many players and continue being played all over the world for twenty more years.

We first explain the basic idea of gameplay Pokemon. Next, we present our new game refinement model called experience based-model. By applying this model, analyzed results obtained from various episodes are discussed.

3.5.1 Overview of gameplay in Pokemon

According to Table 3.1, Pokemon now has six generations and the seventh generation will be released in this 2016. This part focuses on three versions from different generations consists of Pokemon Red from the first generation, Pokemon Fire Red from the third generation and Pokemon Soul Silver from the fourth generation. The term 'generation' used here refers to the platform version in which the game is built. In this study, we consider the entire game with hundreds of battles. This kind of battle occurs in every place in game between human and computer AI as shown in Figure 3.7. Even if player is a newcomer, the chance to win against his opponent is still high. So in our experiments, we do not concern ourselves with the affect of battle type, under the assumption that one knows the basic battle type of computer AI.

Basically, each Pokemon has its unique characteristic. In combination with its level, it can change the outcome of the match massively. So if we meet a strong Pokemon too early, our team will dominate the opponent's team, which makes it not fun at all. With the same idea, if the setting of Pokemon in the opponent's team are too high, player will

FIGURE 3.7: A screenshot of gameplay Pokemon

lose the game many times. Both of these situations lead to be bored and frustration in the player's mind, and then players will not want to play anymore. With these concerns, we collect data and propose our models.

3.5.2 Formulas used in gameplay in Pokemon

The growth rate is a term used to determine how many experience points it takes to level up one Pokemon. A Pokemon that has a 'Fast' growth rate will need less experience points to get to level 100 than one that has a 'Medium' growth rate. According to reliable website [15], Pokemons have six different growth rate functions, with n standing for the target level, and its maximum value is 100. The details can be described in Eqs. (3.20) to (3.25).

$$\text{Erratic}EXP = \begin{cases} \frac{n^3(100-n)}{50} & \text{n} \leq 50 \\ \frac{n^3(150-n)}{100} & 50 \leq \text{n} \leq 68 \\ \frac{n^3(\frac{1911-10n}{3})}{100} & 68 \leq \text{n} \leq 98 \\ \frac{n^3(160-n)}{100} & 50 \leq \text{n} \leq 68 \end{cases} \tag{3.20}$$

$$\text{Fast}EXP = \frac{4n^3}{5} \tag{3.21}$$

$$\text{Medium Fast}EXP = n^3 \tag{3.22}$$

$$\text{Medium Slow}EXP = \frac{6}{5}n^3 - 15n^2 + 100n - 140 \tag{3.23}$$

$$\text{Slow}EXP = \frac{5n^3}{4} \tag{3.24}$$

$$\text{Fluctuating}EXP = \begin{cases} n^3(\frac{(\frac{n+1}{3})+24}{50}) & \text{n} \leq 15 \\ n^3(\frac{n+14}{50}) & 15 \leq \text{n} \leq 36 \\ n^3(\frac{(\frac{n}{2})+32}{50}) & 36 \leq \text{n} \leq 100 \end{cases} \tag{3.25}$$

With these equations, with the experience points player's Pokemon has, player can know which level it is. [15] also let us know the number of Pokemons in each groups which is shown in Table 3.6. So, we can calculate the experience points that a Pokemon needs to level up for each growth rate group as shown in Table 3.7. From then, in order to keep generalization, we calculate the average experience shown in Table 3.8.

TABLE 3.6: Growth rate group

Growth Rate	Number of Pokemons
Erratic	22
Fast	53
Medium Fast	296
Medium Slow	189
Slow	147
Fluctuating	14

The equation to calculate the experience points is given in [121] that player will get when defeating another trainer, which is shown in Equation (3.26).

$$EXP = \frac{a \times t \times b \times e \times L \times p \times f \times v}{7 \times s} \tag{3.26}$$

In this experiment, player only concerns himself with a battle with trainers in the game, not a battle in the wild, and players are not allowed to hold or use support items. Therefore, we can simplify Equation (3.26) to Equation (3.27) as shown below.

$$EXP = \frac{1.5 \times b \times L}{7 \times s} \tag{3.27}$$

Where:

- b: base experience of the Pokemon that was defeated [15].
- L: the level of the Pokemon that was defeated.
- s: the number of Pokemons that participated in the battle that were not defeated.

TABLE 3.7: Sample of experience in each level of each Growth rate group

Lv	Experience					
	Erratic	Fast	MedFast	MedSlow	Slow	Fluc
1	0	0	0	0	0	0
5	237	100	125	135	156	65
10	1800	800	1000	560	1250	540
15	5737	2700	3375	2035	4218	1957
20	12800	6400	8000	5460	10000	5440
25	23437	12500	15625	11735	19531	12187
30	37800	21600	27000	21760	33750	23760
35	55737	34300	42875	36435	53593	42017
40	76800	51200	64000	56660	80000	66560
45	100237	72900	91125	83335	113906	98415
50	125000	100000	125000	117360	156250	142500
55	158056	133100	166375	159635	207968	196322
60	194400	172800	216000	211060	270000	267840
65	233431	219700	274625	272535	343281	351520
70	276458	274400	343000	344960	428750	459620
75	326531	337500	421875	429235	527343	582187
80	378880	409600	512000	526260	640000	737280
85	433572	491300	614125	636935	767656	908905
90	491346	583200	729000	762160	911250	1122660
95	548720	685900	857375	902835	1071718	1354652
100	600000	800000	1000000	1059860	1250000	1640000

3.5.3 Experience-based model

In this measurement, we concern about the balance of setting in the gameplay of Pokemon. If opponent is strong, player must be strong at least as strong as his opponent is so player can have a fair chance to defeat him. To apply this model, we ignore the battle strategy and many minor details in the battle such as Pokemon's type, Pokemon's move and support item.

The proposed idea is the extension of score-limited model. According to previous studies in score-limited sports, $x(t_k)$ and t_k is the average winner's score and the average total score in one game respectively. We improve the score-limited model from previous studies by changing the aspect and adjusting some factors described below.

Equation (3.27) shows that experience points are based on the level of Pokemon and their base experience, together with the fact that a strong Pokemon has a high base experience, therefore the experience points that player gets when player defeats opponent can reasonably reflects how strong his opponent is.

Moreover, to assess the setting of gameplay in Pokemon, we need to consider every battle in game. Therefore, we cannot consider every detail in each battle likes assessment

Table 3.8: Average experience in each level

Level	Average Experience
1	0
5	132
10	958
15	3224
20	7750
25	15315
30	26711
35	42730
40	63736
45	90282
50	123518
55	164696
60	214816
65	273361
70	343098
75	423382
80	516936
85	621611
90	742302
95	874954
100	1024976

of Pokemon battle does but we need to consider the whole game of Pokemon as a one game and consider the battles as a progress of the main game. Next, one thing that you receive as a reward for each battle is the experience points. Therefore, from the basis of score-limited model, every experience point that you receive from the opponents in each battle can be considered as a score in the game progress model of score limited game. So, for $x(t_k)$, winner's score should be represented by the opponent's experience. Respectively, the total of score should be represented by using the summation of experience of opponent and our experience.

$$R = \frac{\sqrt{EXP\ of\ opponent\ team}}{EXP\ of\ opponent\ team\ +\ EXP\ of\ player\ team} \tag{3.28}$$

According to Equation (3.28), the experience that player gets lies in a wide range so we have to scale this value to percentage. This means that the summation of player's experience points and his opponent's experience points is equal to 100%. It can be described by Equation (3.29) and Equation (3.30).

$$R = \frac{\sqrt{EXP\ of\ opponent\ team\ (\%)}}{EXP\ of\ opponent\ team\ (\%)\ +\ EXP\ of\ player\ team\ (\%)} \tag{3.29}$$

$$R = \frac{\sqrt{EXP\ of\ opponent\ team\ (\%)}}{100\%} \tag{3.30}$$

3.5.4 Application to gameplay in Pokemon

Player can obtain data about the Pokemon team and the experience points of Gym Leaders, Elite Fours and the rival via walkthrough [125]. Unfortunately, the experience player acquire in the game [40] is a little bit different from the equation, so we check in both cases. In order to calculate the experience points player's opponent will obtain if he defeats player, we use the following assumptions.

- At the beginning, player has one Pokemon with level five in his team. After defeating the first Gym Leader, player will have one more Pokemon in his team. After defeating the second Gym Leader, player will have another one etc. However, the process will end when player defeats the sixth Gym Leader because player can only carry up to six Pokemons at the same time.
- All player's Pokemons will have his base experience equals to 136. It was calculated based on the average value from the base experience table in bulbapedia.
- The growth rate of all Pokemons is the same, which is the average of the six formulas shown in Table 3.8.
- The level of the new Pokemon in player's team is the average level of the wild Pokemon in that catch zone [125].
- The experience player acquires when player challenges with normal trainer will be divided equally between all of the Pokemons in his team.
- From the experience points Pokemon has, we can find the level of that Pokemon by the average Growth rate.

Table 3.9 shows an example of player's team's information with the first four Gym Leaders by using the assumption mentioned above. In each Gym, there are 2 rows: the value player gets in Real Playing and in Equation (3.27). So, we can calculate the refinement value by using Equation (3.30). Then, Table 3.10 shows the result obtained by choosing Pokemon Fire Red version as an example.

In Table 3.10, *ExpOp* indicates the experience points player obtains when the opponent is defeated, *ExpPly* indicates the experience points the opponent obtains when the player is defeated, *%Op* is the percentage of *ExpOp* and *%Ply* is the percentage of *ExpPly* in the total experience of both teams.

TABLE 3.9: Example of player team's information with the first 4 Gym Leaders

	Number of Pokemons	Level of each Pokemons					
		1st	2nd	3rd	4th	5th	6th
Gym1	1	12	0	0	0	0	0
	1	12	0	0	0	0	0
Gym2	2	22	21	0	0	0	0
	2	22	21	0	0	0	0
Gym3	3	27	26	20	0	0	0
	3	27	22	20	0	0	0
Gym4	4	31	31	27	23	0	0
	4	31	31	27	23	0	0

Finally, we apply our experience-based model to two versions remained, Pokemon Red and Pokemon Soul Silver. The results are shown in Table 3.11. $Rreal$ indicates the average R value which was calculated with the data obtained from the real game [40] [125], while $Rformula$ indicating the average of R value which is calculated by using the equation mentioned in Section 3.5.2.

3.5.5 Discussion on gameplay in Pokemon

We conduct the experiment with data from reliable websites. Then, we apply game refinement theory, an experience-based model, to Pokemon Red, Pokemon Fire Red, and Pokemon Soul Silver. The results of the experiment are shown in Table 3.11. However, despite the fact that the average refinement values lie in the window, from Table 3.10, the R value of individual battles with Gym Leaders is lower. The reason is that in the real game, a battle is not balanced, especially in the number of Pokemons. In the first Gym, player have only one Pokemon but player has to fight with two higher-level Pokemons (level 12 versus level 12 and level 14). At the end of the game, player's team is full with six Pokemons, but his opponent only has five, which gives player a lot of advantage.

We have two ways to explain why the result fits in the window. In the battle with the first Gym Leader and the Rival Champion, the R value is too high which makes the average R value fit in the window even though the values of other battles are below 0.07. So its true value should be below the window but greater than 0.06. The other explanation can be found in the game of chess. In chess, although it is an interesting game, its R value is not high because it usually ends with a draw result. One possible solution is to use a round match tournament, so the player who wins more matches will be the winner of the whole game. We can explain it here in the same way. Each battle is usually not too exciting, because in most of the games, it is easy to predict who is

Table 3.10: Refinement value when using the experience points measurement obtained in real playing and with the model of the score-limited equation

		ExpOp	ExpPly	%Op	%Ply	R
Gym1	RealPlay	544	349	60.92	39.08	0.0781
	Formula	385	349	52.45	47.55	0.0724
Gym2	RealPlay	1339	1253	51.66	48.34	0.0719
	Formula	1081	1253	46.32	53.68	0.0681
Gym3	RealPlay	1405	2125	39.8	60.2	0.0631
	Formula	1802	2125	45.89	54.11	0.0677
Gym4	RealPlay	3182	3262	47.18	52.82	0.0687
	Formula	3131	3262	48.98	51.02	0.0700
Gym5	RealPlay	4710	4487	51.21	48.79	0.0716
	Formula	4124	4487	47.89	52.11	0.0692
Gym6	RealPlay	5093	6117	45.43	54.57	0.0674
	Formula	5738	6117	48.04	51.96	0.0693
Gym7	RealPlay	5994	6992	46.16	53.84	0.0679
	Formula	4860	6904	41.31	58.69	0.0643
Gym8	RealPlay	7830	7603	50.74	49.26	0.0712
	Formula	6934	7458	48.18	51.82	0.0694
Elite1	RealPlay	10122	8100	55.55	44.45	0.0745
	Formula	9777	7923	55.24	44.76	0.0743
Elite2	RealPlay	7912	8566	48.02	51.98	0.0693
	Formula	8066	8333	49.19	50.81	0.0701
Elite3	RealPlay	9731	8973	52.03	47.97	0.0721
	Formula	10686	8711	55.1	44.9	0.0742
Elite4	RealPlay	11209	9323	54.59	45.41	0.0739
	Formula	11378	9060	55.67	44.33	0.0746
Champ	RealPlay	15318	9644	61.81	38.19	0.0786
	Formula	15572	9293	62.63	37.37	0.0791

Table 3.11: Comparison of game refinement values

Version	$Rreal$	$Rformula$
Pokemon Red	0.0720	0.0714
Pokemon Fire Red	0.0714	0.0710
Pokemon Soul Silver	0.0691	0.0698

the winner. But in a whole series of battles, there will be a higher chance that a player loses to another trainer, which makes the refinement value fit in the window.

In a real game, the player not only fights trainers, but also with wild Pokemon on grass to obtain experience points. This leads to the deduction that the Pokemon team of players will be stronger, thus the R value in the experience-based model will be decreased. This also means that this game will be easier than the experiment of the model, so it will be suitable for children to enjoy. Player does not have to spend too much time to struggle with a tough Gym leader, instead, player can enjoy the story line and doing the side

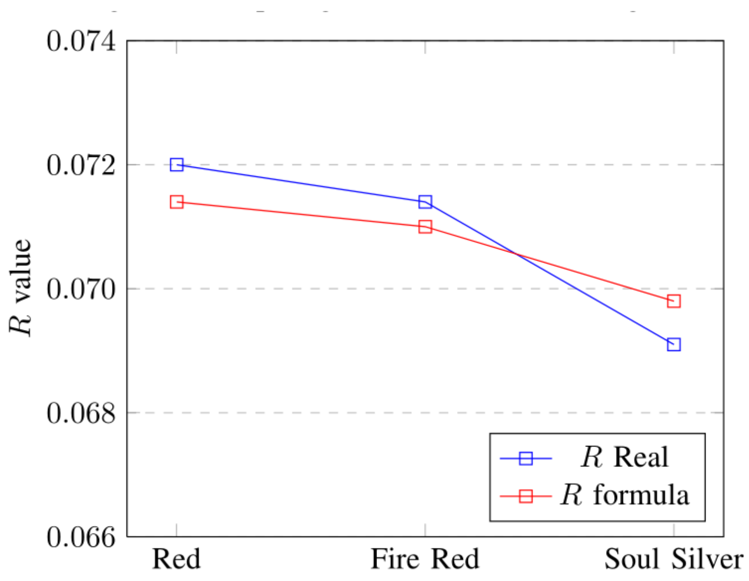

FIGURE 3.8: Changes of game refinement values with three versions of Pokemon

quests too. Another factor is that we make an assumption that the base experience of all Pokemons is 136, which is usually higher than in the real play.

It is too early to say that normal battles are not interesting because the individual game's values lie under the window. However, if we want to raise the R value in order to obtain a comfortable setting, there are several ways.

- Increase the number and the level of Pokemons in the Gym Leader team.
- Decrease the number of battles with trainers so player's Pokemon will have a lower level.
- Decrease the level of the wild Pokemon and the wild Pokemon should have a low base experience.
- The Pokemon that most people love to choose in their team should have a 'Slow' growth rate or a slower one.
- The Pokemon team of Players will not be filled soon in the first six Gyms. A strong Pokemon should only be found in the later stage.

For the trend of changing of setting of gameplay, we start by comparing the results in Table 3.11 as shown in Figure 3.8.

According to Figure 3.8, we will see that the R value of the Pokemon which is applied by using the experience-based model is $0.069 - 0.072$, while previous studies confirm that sophisticated games will have an R value between $0.07 - 0.08$. So, we can say that Pokemon has an R value at the edge of the appropriate range.

Importantly, we can see in Figure 3.8 that the trend of the R value is decreasing. This may lead to a Pokemon's excitement is being lower from an old version to a new version. The slightly low R value indicates that it is suitable for children who are the main target of the Pokemon game.

Pokemon Red in general is the same setting as Pokemon Fire Red. There are just some minor differences such as in the summation of experience points player obtains if player defeat all trainers in the game: 608,098 Exp to 672,388 Exp of Fire Red. The main reason is that the level of trainers in the Red version is lower than in the Fire Red version which affects the R value and makes it higher. This is suitable with our suggestion above.

For Pokemon Soul Silver, the story takes place in two regions. In the first one, the setting of the Gym leader's team is quite low, while in the second one, it is really high with the R value of the last battle reaching 0.0819. However, when we calculate the average, it does not fit in the window. This can be explained as a trade-off when player has a longer story line with two regions compared to one region such as Pokemon Red or Pokemon Fire Red.

Therefore, for the question, what is the trend of the changing of setting of gameplay in Pokemon, we can answer that R value is being decreased and there are many reasons explained above.

These are our efforts to see the evolution in the gameplay and the changing in the R value. In future work, we will conduct a survey with Pokemon players to see how they rank each version in a scale up to 10 with the best rank being 10. After that, we can look for a satisfactory explanation for the connection between the feeling and the gameplay in the human brain. Next, we can consider the R value so we can see the appropriate range of the R value and how the Pokemon game has evolved since the first version. Moreover, the later version comes with a new kind of competition such as beauty contest so we can also apply game refinement theory to this part of the game.

3.6 Conclusion

This chapter presented an assessment of Pokemon in three important components of Pokemon consist of Pokemon battle, catching Pokemon and gameplay in Pokemon. We aim to find a reason why Pokemon has been popular so long time and to find comfortable settings of Pokemon. To tackle these challenges, game refinement measure was used as an essential tool for the assessment.

For Pokemon battle, by collecting data from online Pokemon battle simulator and using our Pokemon battle AI, we apply board game approach and weexplore that Pokemon battle has slightly lower game refinement value than expected. We conclude that this slightly low R means that it is suitable for children with not too much excitement. With this assumption, we found that the setting ($n = 6$) is most comfortable. Moreover, there are other reasonable settings for some specific levels. For example, $n = 5$ or $n = 7$ may be a comfortable setting for experts or beginners, respectively.

For catching Pokemon, we introduce a value of prize captured, V, because each prize has its own unique value. For the Pokemon case, we can calculate the value of captured prize by considering the rareness. Our result confirms that the mechanism for catching Pokemons has been changed in a proper way by editing some details in the catch mechanism which directly increases R value since the first generation to sixth generation. With the changing to an appropriate R value, we conclude that this is the reason why the mechanism for catching Pokemon has been changed so many times. Game designer tries to improve the catching mechanism by finding and editng the setting of some factors in catching mechanism. Moreover, we can predict that in the next generation, the catching mechanism will have R value in an appropriate range.

For gameplay in Pokemon, the application of our new game refinement model, an experience-based model, to three version of Pokemon games consist of Pokemon Red, Pokemon Fire Red and Pokemon Soul Silver is presented. According to the results, we determined that Pokemon has a refinement value at the edge of the window as we expected. The trend of the R value of Pokemon is being decreased. This is because the target players of Pokemon are children. Moreover we suggest several ideas to increase the R value for the battle in the game.

Nevertheless, the fun of Pokemon is not derived only from these component of Pokemon. Because Pokemon world has many component which is not investigated yet. The fun of Pokemon depends on many factors in RPG mode such as game story, collecting Pokemons, doing a quest and so on. For example, customizing one's own Pokemon is one of important entertaining factors. If we improve the customization of each Pokemon such as skill or moves, it may be more excited. Moreover, the more customizing each Pokemon can be realized, the more the difference between those Pokemon will appear. Because now the main difference between Pokemons (excepts its appearance and little factors) is only its status and its move. Therefore, for current 721 Pokemons, it is too narrow to differentiate each Pokemon. If we have more detail such as skill tree or more kind of status it will be more attractiveness.

In conclusion, it is obvious that game refinement theory can effectively be used in many domains of game such as classical board game, video game and sport game, of course in Pokemon, by establishing a reasonable game information progress model. It can be used as a helpful tool to measure the attractiveness of a game and it also enables game designers to make a target game more sophisticated.

It is understood that the work presented here is a simple model with no complicated factors and more studies are required. In Pokemon battle AI, this work has some limitations such as the number of Pokemons implemented, Pokemon battle's AI and some Pokemon battle mechanics which is hard to implement are ignored. Further works may improve these points. For catching Pokemon, further works may include to collect data in other types of ball or in other games which may have a catching component. Also, we can aim to investigated another video game by using these models as the basis.

Chapter 4

Analyzing Sports using Game Refinement Measure

This chapter is an updated and abridged version of work previously published in

(6) Chetprayoon Panumate, Hiroyuki Iida and Abu-Bakar Nordin. Measuring the Excitement in Boxing Game with Game Refinement Theory. International Conference on Hospitality, Leisure, Sports, and Tourism (HLST 2016).

(7) Anh-Tuan Dinh, Chetprayoon Panumate and Hiroyuki Iida. Attractiveness of Tennis Over Conservative Rule Systems. International Conference on Information and Social Science (ISS 2016).

(8) Yuranan Kitrungrotsakul, Chetprayoon Panumate, Hiroyuki Iida and Kiyofumi Tanaka. Measuring Sophistication of Sports Games: The First Result from Baseball. IPSJ SIG-GI Technical Report, vol. 2016-GI-36, no. 4, pages 1-7.

(9) Chetprayoon Panumate, Ryo Takahashi and Hiroyuki Iida. Analyzing Sports using Game Refinement Measure. ODISHA JOURNAL OF SOCIAL SCIENCE. submitted.

This chapter presents an application of game refinement theory to sports. We first raise many interesting research questions. Then, we show how to answer those questions by applying game refinement theory to many domains of sports composed of score limit sport, time limit sports and fighting sport.

4.1 Introduction

Sports are artificial tests established by the rules that not only prescribe the use of less efficient means to achieve the goal stipulated but also require the implementation of physical skills to do so [64]. It is generally considered as an activity which uses many combinations of player's physical skills such as strength, vitality, agility, dexterity and so on. To have these great skills, training is the most important thing that a player has to do. The more you train the better you get is a very common and essential quote that everyone knows. Usually the contest is performed between two sides, both single player and team sports [81], and the results are winning, losing or draw. In order to be a winner, a player has to devote everything that he has under the considered sports' rule. Using intelligence to improve the benefit of our team and defeat the opponent's team is common thing. Also, luck can be considered as one significant thing that affect the game's result. By these factors mentioned before, we can consider sports as a game [115] [119].

Sports are one kind of games which can be seen every day ubiquitously. It can be played by children, teenagers, adults and elders. There are many kinds of sports. For example, badminton, ballooning, boxing, dancing, fencing, fishing, golf, kendo, kickboxing, kung fu, paragliding, snowboarding, surfing, tennis, volleyball, wakeboarding, etc. Also, there are many classification of sports. For example, we can consider badminton, tennis and volleyball as a ball-over-net game and boxing, fencing, kendo, kickboxing and kung fu as a fighting sports. We see that there are many types and kind of sports. This is the reason why sports are played universally and can be seen ubiquitously. It is like a basic activity of human. However, among the sports, some of them are very popular, widely played and the competition's winning prize is very high such as soccer, whereas others are not so. The most significant thing that makes the difference between those sports is the rules.

Many modern sports games have a long history. It is evolved over decades or centuries of playing. Changing the rules of the game in order to improve the quality in various aspects is a common practice [142]. Some of sports try to improve the excitement of the game by adjusting some factors. Some other sports try to balance between the luck and skill used in the game by adjusting some factors. Even though in the same kind of sports, there are also many variants. For example, the difference between the competitions makes the rules of the game slightly different. The difference of external factors such as competition's place also make the game different. Moreover, comparison of changing of the rules for each year is an interesting thing which should be investigated. These little difference can lead the game progress and its result totally different. By changing the rules, it can make the game so popular or unpopular. As we have said that there are many types of sports, it can be roughly approximate that there are 8,000 indigenous sports and sporting games [77]. Unfortunately, there are just only less than 100 sports that is well known. The reason behind this fact is the setting of the rules of the considered sport. That is why sports have been developed by changing the rules.

Beside this, the studies of strategic decision making in the framework of game theory with a focus on mathematical models of conflict and cooperation between intelligent rational decision-makers or game players have been conducted for a many years [39]. The origin of game theory is the idea regarding the existence of mixed-strategy equilibrium in two-person zero-sum games. The idea has been widely utilized in many fields such as economics, political science and psychology. In contrast, game refinement theory is another game theory focusing on attractiveness and sophistication of games based on uncertainty of game outcome [128]. It has been applied in various domains such as boardgames, video games and sports games. Interestingly, it is observed that many sophisticated boardgames and popular sports games have a similar game refinement measure, which locates somewhere between 0.07 and 0.08. The previous works support that the game refinement is directly related to entertainment or engagement for spectators and players [58].

As we introduced that sports can be considered as a game, therefore, game refinement theory can be applied to sports. Hence, in this research, we demonstrate the application of game refinement theory to sports. We explain how to figure out a reasonable model of game progress in the target sports. Moreover, we show the procedure for how game refinement theory can be a helpful tool to answer research questions in each assessment. For each assessment, the meaningful research questions are raised and answered by applying game refinement theory. As we explained that there are many type of sports, we try to choose the type of sports differently in order to do a comprehensive research. We, therefore, select baseball as a representative of time limit sports, boxing as a representative of fighting sports, and tennis as a representative of score limit sports. Then, for each sports, we collect data from the reliable sources and apply game refinement theory with appropriate model. Finally, the results obtained are analyzed.

The structure of this paper is as follows. We first briefly introduce the basic idea of game refinement theory with its background and recent development in the domain of sports games in Section 2. Secondly, Section 4.2 shows our analysis of baseball. Also, Section 4.3 shows our analysis of boxing and Section 4.4 shows our analysis of tennis. Finally, the results obtained are discussed and concluding remarks are given in Section 4.5.

4.2 Analysis of Baseball

Baseball is investigated in many points such as multimedia [113], biomechanics [101] and pharmacy [124]. In contrast with those papers, this section aims to quantify entertainment impact of baseball. As we have explained that sports can be considered as a game, respectively, baseball also can be considered as a game. Thus, game refinement measure can be used as a tool to analyze the entertainment of baseball. We here raise main research questions below.

1. What is the trends of baseball?
2. What is the difference between leagues?
3. What is the difference between regions?
4. What is the difference between baseball and softball?
5. What is the future of baseball?

In this section, we first briefly introduce some details of baseball. Then, we explain how to apply game refinement theory to baseball. Finally, the results obtained are discussed.

4.2.1 Overview of Baseball

Baseball is a competitive sports between two teams. Each game consists of nine innings that each team plays as an offense and defense side, respectively. In order to earn scores, the players on the offense team must run through four bases around the field, while the players on the defense team try to defend the base. Baseball has been quite popular especially in Japan. However, nowadays, it seems that baseball is less attractive than it was. The reasons behind its decreased popularity are doubtful.

After being admitted to the Olympics as a medal sport beginning with the 1992 Games, baseball was dropped from the 2012 Summer Olympic Games at the 2005 International Olympic Committee (IOC) meeting. It remained part of the 2008 Games. The elimination of baseball, along with softball, from the 2012 Olympic program enabled the IOC to consider adding two different sports, but none received the votes required for inclusion [2] [85].

In this study, the data from leagues in each year from 2000 to 2015 were collected from reliable source [108]. These data include the leagues from several countries such as USA, Korea and Japan. Moreover, the classification of leagues such as Major, Triple-A, Double-A are gathered.

4.2.2 Game progress model of Baseball

Baseball does not have a time limitation like soccer or basketball, nor does have a score limitation like volleyball. Baseball consists of 9 innings per one game. The players from 2 teams play as an offensive and defensive side in each inning. In addition, the winner team is decided by the score at the end of last inning. Thus, the number of innings is a limitation for each player to earn the score. Therefore, the baseball can be considered as a time limit sport even if it does not have a real time limitation in game.

In sports games such as soccer and basketball, the scoring rate is calculated by two factors: (1) goal, i.e., total score and (2) time or steps to achieve the goal. For example, in basketball the total score is given by the average number of successful shoots, whereas the steps to achieve the goal is estimated by the average number of shoots attempted [129]. Then the game speed of basketball is given by

$$\frac{average\_number\_of\_successful\_shoots}{average\_number\_of\_shoots}$$

By considering as a time limit sports domain, the game refinement model for time limit domain can be applied to the baseball. However, each team has only one chance, which is playing as the offensive side, in each inning to gain the score. The others sport in time limit domain can gain the score for the entire game. This is a major difference between baseball and other time limit sports. In order to get the score in baseball, the batter, who swings a bat, must hit the ball, which pitcher throws, and reaches the four bases. Thus, the number of hits in each game is considered as a game progress. On the other hand, the average score in each game is directly proportional to the number of hits. Then the game speed of baseball is given by

$$\frac{average\_number\_of\_scores}{average\_number\_of\_hits}$$

Let S and H be the average number of scores and hits in each game, respectively. The game progress model of baseball is given by Equation (4.1).

$$x(t) = \frac{S}{H}t \tag{4.1}$$

Following the same procedure to obtain Equation (2.4), the game refinement measure of baseball is derived from Equation (4.2).

$$R = \frac{\sqrt{S}}{H} \tag{4.2}$$

4.2.3 Results and discussion

Game refinement measure is employed for the game informatical analysis of baseball. All data in this study were collected from the baseball leagues that conducted from 2000 to 2015 [108]. We focus on three categories: years, leagues and regions.

4.2.3.1 Trends for the years 2000 - 2015

The baseball leagues are analyzed to obtain the average R values for the years 2000-2015. The results are summarised in Table 4.1, which shows that baseball's R values for the period are located at somewhere between 0.063 and 0.066. The results are depicted in

Table 4.1: Baseball: Scores, Hits and R values, 2000 - 2015

Years	Number of games	Scores	Hits	R
2000	35374	176960	1190171	0.066
2001	34920	164468	1178324	0.064
2002	35240	164313	1187817	0.064
2003	35524	164126	1190869	0.064
2004	34612	169478	1168346	0.065
2005	36539	181316	1243020	0.065
2006	39896	185088	1337652	0.064
2007	42132	203348	1419498	0.065
2008	42342	205503	1430130	0.065
2009	41176	195358	1382997	0.064
2010	42120	200161	1416249	0.064
2011	41339	197598	1390301	0.065
2012	41574	193862	1397980	0.064
2013	41309	187857	1383344	0.063
2014	41128	190937	1385524	0.063
2015	41501	185823	1394460	0.063

Figure 4.1. Figure 4.1 shows that the highest value comes from the year 2000 while the lowest one comes form the last 3 years: 2013-2015. To be more precise, R values reduce consecutively during the years 2000 - 2015.

For the comparison, we show, in Figure 4.2, the trend of soccer's R values for the years 2003 - 2013 [129]. For example, R value in 2003 is 0.073, which is in the sophisticated zone. However, it has deceased until 2006, which takes the lowest value 0.065. Moreover, we show, in Figure 4.3, the trend of basketball's R values for the years 2003-2013 [129]. R values are above the sophisticated zone, however, it has increased until 2007, which is the highest value (around 0.087), but later decreased to be in the sophisticated zone.

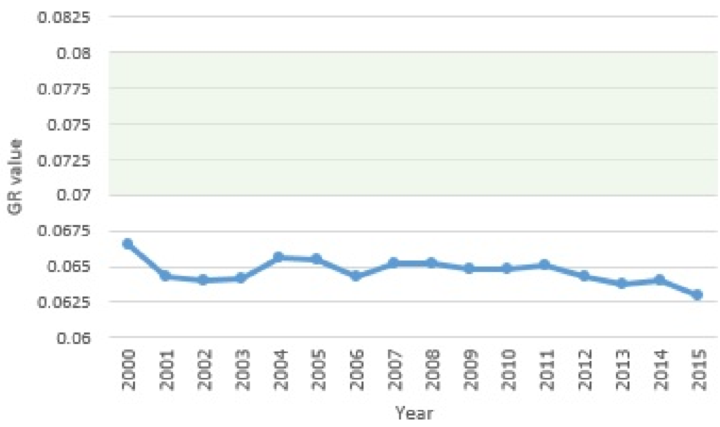

Figure 4.1: Baseball's R values for the years 2000-2015

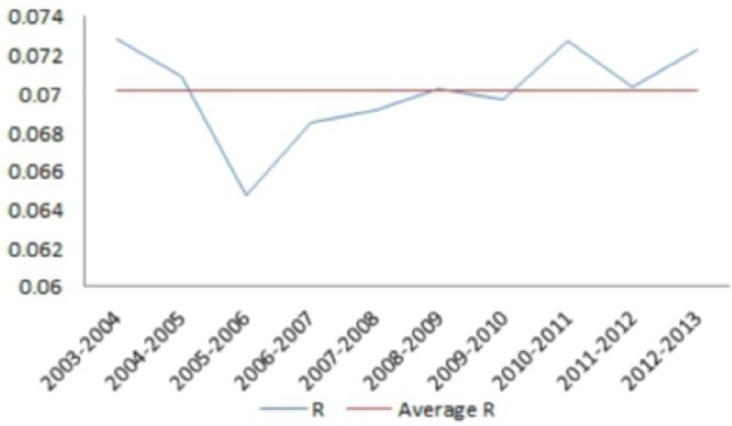

Figure 4.2: Soccer's R values for the years 2003-2013

Both popular sports: soccer and basketball, are lower and higher than the sophisticated zone at some point, respectively. Baseball's R value dropped in 2006, and during 2007 - 2011 R values are stable at 0.065. Then, it has decreased consecutively until 2015. As R values of soccer and basketball have been changed due to the minor rule changes, there is a tendency that R value moves to the sophisticated zone, while it increased in soccer and it decreased in basketball, respectively. However, baseball's R value did not move to the sophisticated zone yet, instead R value has decreased moderately.

4.2.3.2 Leagues - performance quality

Major League Baseball (MLB) is a professional baseball organization that is the oldest of the four major professional sports leagues in the United States and Canada. A total of 30 teams now play in the American League (AL) and National League (NL), with 15 teams in each league.

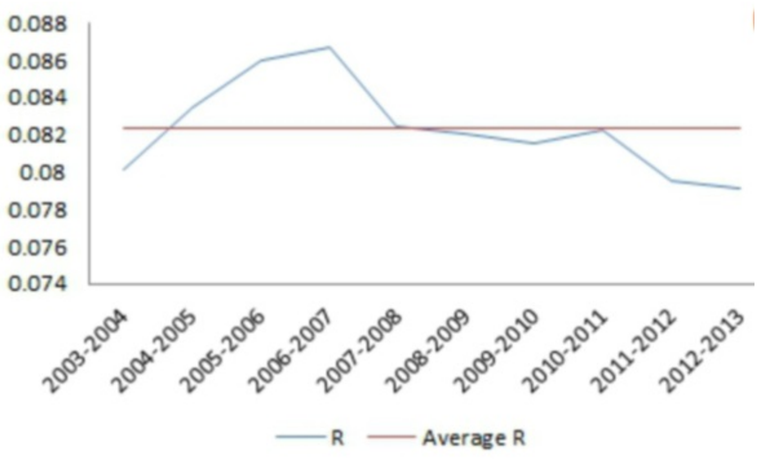

FIGURE 4.3: Basketball's R values for the years 2003-2013

Minor League Baseball is a hierarchy of professional baseball leagues in USA that compete at levels below MLB and provide opportunities for player development and a way to prepare for the major leagues. All of the minor leagues are operated as independent businesses. Most are members of the umbrella organization known as Minor League Baseball (MiLB), which operates under the Commissioner of Baseball within the scope of organized baseball. Several leagues, known as independent baseball leagues, do not have any official links to Major League Baseball.

The current minor league classification system divides leagues into one of five classes, those being Triple-A (AAA), Double-A (AA), Class A (Single-A or A), Class A Short Season, and Rookie. Furthermore, Class A is further subdivided into Class A and Class A-Advanced (often called Low-A and High-A, respectively), and Rookie is further subdivided into Rookie Advanced, Complex-based Rookie and international summer baseball. Under the rules governing the affiliated minor leagues (specifically Major League Baseball Rule 51), Class A Short Season is a separate classification from the other leagues bearing the "Class A" name, despite the similarity in name.

In this study, six baseball leagues including Major, TripleA, DoubleA, AdvanceA, A and Rookie, are analyzed. The results are summarized in Table 4.2.

The results are depicted in Figure 4.4.

Baseball's R value in each league is somewhere between 0.062 and 0.067. It seems that R value is inverse proportional to the playing performance quality.

The game refinement values of all leagues are lower than the sophisticated zone. Major league has the lowest value, whereas Rookie league has the highest value. It is likely that the main difference between Major league and Rookie league is the performance gap between batters and pitchers. In Major league, the performance of batters is lower

TABLE 4.2: Baseball: Scores, Hits and R values at 6 leagues

Leagues	Number of games	Scores	Hits	R
Major	77740	354978	2660699	0.062
AAA	96540	473002	3270860	0.065
AA	67210	301019	2240816	0.063
Adv A	66615	310083	2234234	0.064
A	66120	299532	2204732	0.064
Rookie	42816	215175	1433607	0.067

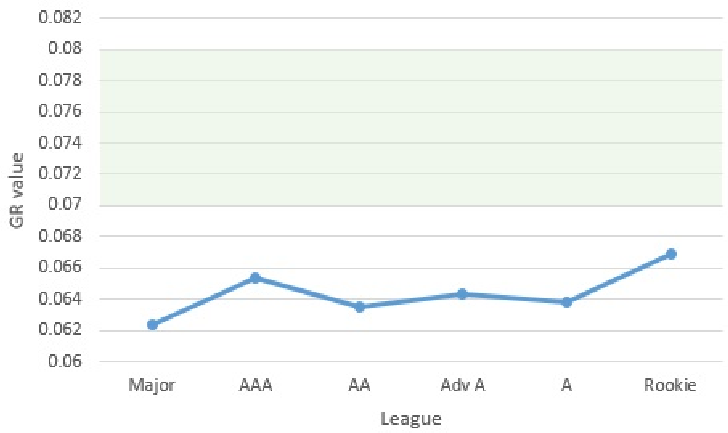

FIGURE 4.4: Baseball's R values at 6 leagues

than pitchers, whereas the performance gap in Rookie between batters and pitchers is not so significance. Then, the successful percentage of hits in Major league is lower than Rookie league. Thus, the average score in Rookie league is higher than Major league. Therefore, Rookie's R value is closer to the sophisticated zone than Major league.

4.2.3.3 Regions - locality and diversity

The baseball leagues from 6 countries: United States, Australia, Holland, Italian, Japan and Korean, are analyzed. The results are summarized in Table 4.3. The R values vary between 0.060 and 0.067. Holland takes the highest R value, whereas Japan takes the lowest one. The results are depicted in Figure 4.5.

The game refinement measure of Japan leagues is significantly different from others. Japan baseball leagues has a different rule with others. Japan leagues use a smaller baseball, strike zone, and playing field. The Japanese baseball is wound more tightly and is harder than others. These difference directly affect the score that can be earned in each game. Thus, the game refinement of Japan is totally different from other countries.

TABLE 4.3: Baseball: Scores, Hits and R values in 6 countries

Countries	Number of games	Scores	Hits	R
US	550317	2619772	18549940	0.065
Australia	1612	7810	53976	0.066
Holland	2266	10813	74035	0.067
Italy	2252	10675	75054	0.065
Japan	27212	112293	915472	0.060
Korea	17287	82856	587741	0.064

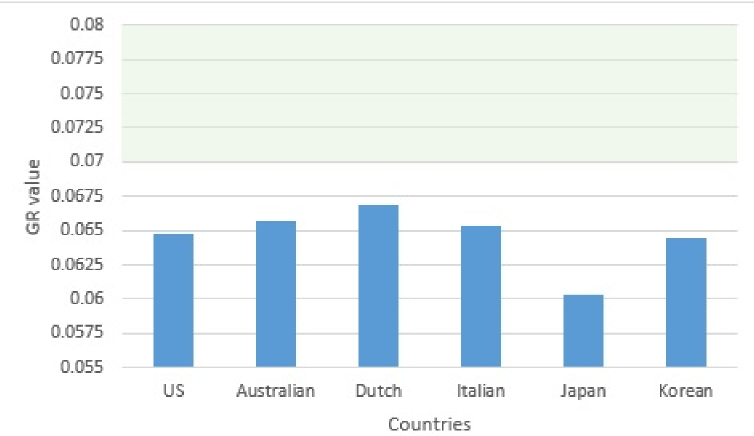

FIGURE 4.5: Baseball's R values in 6 countries

4.2.3.4 Baseball and Softball compared

Softball is a variant of baseball played with a larger ball on a smaller field. It was invented in 1887 in Chicago as an indoor game. It was at various times called indoor baseball, mush ball, playground, softbound ball, kitten ball, and, because it was also played by women, ladies' baseball. The name softball was given to the game in 1926 [3]. Women's softball made its first Olympic appearance in 1996 and made its final Olympic appearance in the 2008 games [42]. Softball and baseball have both failed attempts to be reinstated to the Olympics for 2012 and 2016. In 2012 the heads of the International Softball and Baseball Federations announced that they were uniting to increase their chances in playing in the 2020 games.

Softball is analyzed using the data from 2010 to 2015. The results are summarized in Table 4.4. An important difference between softball and baseball when applying the game refinement theory is the number of innings. Baseball consists of 9 innings a game but softball consists of 7 innings. Softball's R values fall between 0.074 and 0.080, that is the sophisticated zone. The results are depicted in Figure 4.6.

TABLE 4.4: Softball: Scores, Hits and R values for the years 2010 - 2015

Years	Number of games	Scores	Hits	R
2010	474	2106	12536	0.080
2011	454	1761	11809	0.076
2012	482	1966	12738	0.076
2013	398	1498	10471	0.074
2014	378	1438	9856	0.075
2015	398	1796	10616	0.080

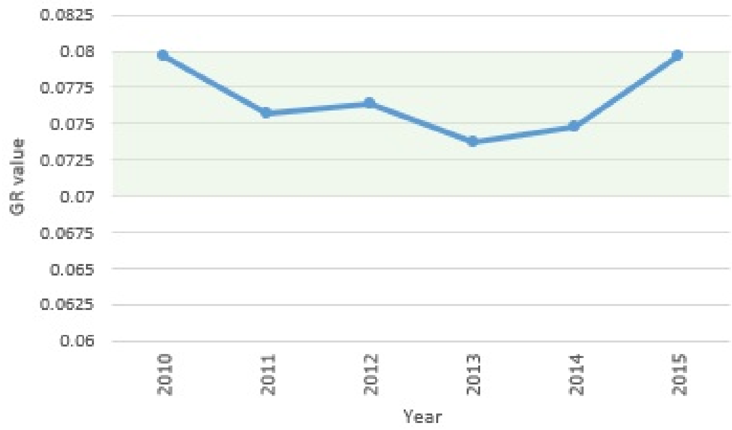

FIGURE 4.6: Softball's R values for the years 2010 - 2015

4.2.3.5 Baseball in the future - possible enhancements

The results from each year are slightly lower than the sophisticated zone. According to these measurements, if the rules of baseball are not changed in the future, the difference between its game refinement measure and the sophisticated zone will increase. This forecasting is shown in Figure 4.1. In order to increase the game refinement value of baseball, the scores and home runs in each game must be increasing while the attempt does not increase as much as scores and home runs. According to the above result, changing some rules such as decreasing the number of innings, decreasing the field size or increasing the strike zone should be preferred to increasing the game refinement measure of baseball.

4.2.4 Summary

Baseball has been one of the most popular sports, especially in Japan, with a long history and widely played in many countries. However, it does not have any of changed rules that impact the sports. Thus, it is likely that the popularity of baseball has been decreasing over time. Baseball's entertainment was evaluated by the game refinement measure, which concerns about attractiveness and sophistication. The model of time limit sports was applied to baseball, which considers the innings as the time. The baseball leagues both major and minor from many countries were chosen to apply the game refinement theory.

The range of game refinement value of baseball is around 0.065, which is lower than the sophisticated zone. Moreover, it trends to decreasing in the future. Thus, baseball will not attract some viewers. The value of game refinement is decreasing because the rules does not change for a while. In contrast, the rules of other popular sports have been changed, which lead their game refinement value to the sophisticated zone. In term of countries, baseball was the most popular sports in Japan. However, the game refinement value shows that Japan has the lowest value among other countries compared. The reason is that the rules of Japan leagues that difference, which trend to get score harder than others. In term of leagues, Rookie league has the highest game refinement value. In contrast, Major league has the lowest one. Thus, the average scores in each game of Major league is lower than Rookies league. The reason is that the performance gap between batters and pitchers. Lastly, the softball's game refinement value falls in the sophisticated zone due to the effort for changing its rules, which are a huge impact to the games, promote the scoring than baseball.

4.3 Analysis of Boxing

While many efforts have been devoted to the studies of Boxing with focus on many points [73] [91] [143]. This study's main aim is to quantify entertainment impact of boxing. We set our main research questions shown below.

1. What is the current trend of boxing's rule changing?
2. What is the difference between amateur boxing and professional boxing?

For these purposes, we propose a new game refinement model using score ratio system called winning percentage method. Then, we choose Olympic boxing [97] as a representative of amateur boxing and World Boxing Association (WBA) [144] as a representative of professional boxing. We believe that knock out case is one of the most important reasons which makes boxing outstanding.

In this section, we first provides some details of Olympic boxing and World Boxing Association (WBA). Next, we describe how game refinement theory is fine tuned and apply it to boxing. Finally, the results obtained are discussed.

4.3.1 Overview of Boxing

In this study, we collect the data from Olympic Boxing in 10 continued programs: Light flyweight, Flyweight, Bantamweight, Lightweight, Light welterweight, Welterweight, Middleweight, Light heavyweight, Heavyweight and Super heavyweight, as shown in Table 4.5.

TABLE 4.5: Olympic boxing men's weight classed

Programs	Year		
	1992 – 2000	2004 – 2008	2012
Light flyweight	−48		46 – 49
Flyweight	48 – 51		49 – 52
Bantamweight	51 – 54		52 – 56
Lightweight	57 – 60		56 – 60
Light welterweight	60 – 63.5	60 – 64	
Welterweight	63.5 – 67	64 – 69	
Middleweight	71 – 75	69 – 75	
Light heavyweight	75 – 81		
Heavyweight	81 – 91		
Super heavyweight	91+		

For WBA boxing, we collect the data from WBA boxing in 17 programs: Mini flyweight, Light flyweight, Flyweight, Super flyweight, Bantamweight, Super bantamweight, Featherweight, Super featherweight, Lightweight, Super lightweight, Welterweight, Super welterweight, Middleweight, Super middleweight, Light Heavyweight, Cruiserweight and Heavyweight, as shown in Table 4.6.

In contrast with amateur boxing [5], professional bouts are typically much longer and can last up to twelve rounds, though less significant fights can be as short as four rounds. Protective headgear is not permitted and boxers are generally allowed to take substantial punishment before a fight is halted. Professional boxing has enjoyed a much higher profile than amateur boxing throughout the 20th century and beyond.

Next, we propose a new model of game refinement theory based on score ratio of the winning side over the losing one which can be called winning percentage method. Finally, by applying our new model of game refinement theory, we can find the game refinement measure of boxing.

4.3.2 Game progress model of Boxing

In this section, we first present our approach for fighting sports called winning percentage method. Then, the analysis of Olympic boxing and WBA boxing are presented.

Table 4.6: World Boxing Association men's weight classed

Programs	Limits in pounds	Limits in kilos
Mini flyweight	105	47.63
Light flyweight	108	48.99
Flyweight	112	50.80
Super flyweight	115	52.16
Bantamweight	118	53.52
Super bantamweight	122	55.34
Featherweight	126	57.15
Super featherweight	130	58.97
Lightweight	135	61.23
Super lightweight	140	63.50
Welterweight	147	66.68
Super welterweight	154	69.85
Middleweight	160	72.57
Super middleweight	168	76.20
Light Heavyweight	175	79.38
Cruiserweight	200	90.89
Heavyweight	200 Plus	90.89 Plus

4.3.2.1 Winning percentage method

We principally consider boxing as a time limit game because one of each boxing contest can be ended by four main exit points shown in Figure (4.7).

1. Timeout: If the time runs out, the game will end.
2. KO (Knock Out): This case is like a sudden death case. If one of the boxers is knocked out, the other boxer will automatically win.
3. Stopping the contest by some reasons: For example, RSC, it means referee stopped contest because an important reason such as RSC-I, referee stopped contest because of an injury of the opponent, RSC-OS, referee stopped contest because the opponent is outscored, RTD, a fighter refuses to continue or a fighter's corner refuses to allow the fighter to continue and so on. These case in professional boxing will be called TKO, technical knockout. TKO is declared when the referee, official ring physician, the fighter, or the fighter's corner man decide that a fighter cannot safely continue the match.
4. WO (won by walkover case): WO is an award of victory to a contestant because there are no other contestants, or because the other contestants have been disqualified or have forfeited.

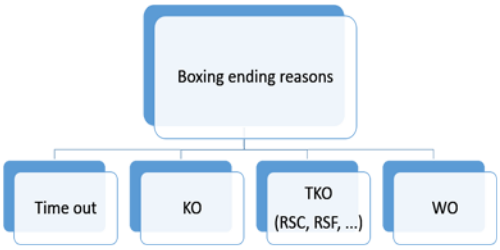

FIGURE 4.7: Four exit points in a Boxing match

However, the time limit in boxing is divided into rounds and each round consists of many points. So, we can consider boxing as a round-match scoring sports. We may obtain the data of the average winner's score and the average total scores, as previously discussed in the domain of scoring sports. However, boxing is a different case where the score for each boxer in each match may be varied, due to the difference of boxers' skill and referees' discretion. To tackle this problem, we use the winning percentage as shown in Equation (4.3).

$$P_w = \frac{WS}{WS + LS} \times 100 \tag{4.3}$$

Where P_w stands for winning percentage, WS winner's score and LS loser's score. Then, we can find G as the percentage of average P_w. It is shown in Equation (4.4).

$$G = \frac{1}{n} \sum_{0<i<n} \frac{WS_i}{WS_i + LS_i} \times 100 \tag{4.4}$$

Since T always equals 100, then we obtain R value as shown in Equation (4.5).

$$R = \frac{1}{10} \sqrt{\frac{1}{n} \sum_{0<i<n} \frac{WS_i}{WS_i + LS_i}} \tag{4.5}$$

Moreover, boxing match has a few exit points as mentioned in Figure 4.7.

For a knock out case, the match will instantly end and the winner is a boxer who make a knock out. We will also consider the knock out case into our game refinement approach by giving this match's winning score equals to the maximum of winner's score which ever be and we will give the loser's score equals 0 because the knock out is a super exciting case that viewer really want to see. By this approach we can calculate $Winning \quad percentage = 100$ in knock out case. This is special for boxing which is different from other sports. In other sports,such as soccer, we ordinarily wait the result of contest by scoring system [49]. But in boxing, we absolutely expect the KO and it is very exciting.

Likewise, for TKO and another stopping decision by a referee, we will calculate with the same idea like KO case. Because TKO makes a game instantly stop and viewer also expect it to happen. So, we let $Winning\ \ percentage = 100$ in technical knock out case.

Finally, for WO case that make the boxer win without contest. It is obvious that the match is boring and viewers do not expect a game to be like that. So, for this case, we will consider this in our game refinement approach by giving this match's $Winning\ \ percentage = 0$.

4.3.2.2 Olympic Boxing analysis

The game refinement measure (R) of Olympic boxing is fine tuned by applying winning percentage method. For example, the statistical results of Light welterweight class in 1996 are shown in Table 4.7.

The $winning\ \ percentage$ is computed by $\frac{WS_i}{WS_i+LS_i} \times 100$. The accumulated values of $winning\ percentage$ are summed and averaged as prescribed by the equation above.

We apply this approach for Olympic boxing's data in ten weight classes. Each class runs from 1992 to 2014. The values of R are shown in Table 4.8.

Table 4.9 shows R for 3 different group of years. The range of weight in each class differs and can be grouped into 3 distinct years

4.3.2.3 WBA Boxing analysis

In WBA boxing, since winning is the accumulated scores from three judges, the following equation is used to calculate the winning percentage ;

$$P_w = \frac{1}{n} \sum_{0<i<n} \frac{WS_i}{WS_i + LS_i} \tag{4.6}$$

Where P_w stands for winning percentage, WS is the winner's score, LS holds the loser's score and $n = 3$ refers to the number of judges involved. Then, G as the percentage of average P_w as in Equation (4.4) and we can calculate R by using Equation (4.5). The data for WBA Boxing in November - December 2010 are shown in Table 4.10.

The results of game refinement measure are shown in Table 4.11.

TABLE 4.7: Statistical results of light welterweight class

Year	Winner's score	Loser's score	R
1996	RSC		1.00
	21	5	0.81
	16	12	0.57
	21	6	0.78
	16	3	0.84
	25	3	0.89
	26	12	0.68
	11	3	0.79
	18	2	0.90
	17	3	0.85
	32	7	0.82
	25	9	0.73
	RSC		1.00
	19	2	0.90
	11	11	0.50
	KO		1.00
	23	1	0.96
	11	9	0.55
	11	3	0.79
	16	7	0.70
	19	8	0.70
	16	6	0.73
	14	6	0.70
	17	6	0.74
	15	7	0.68
	13	8	0.62
	16	15	0.52
	19	8	0.70
	23	6	0.79
	20	6	0.77
	20	13	0.61

In Olympic boxing, the data start from March 2008 to December 2014. In each year, the number of matches is unequal and the total number of matches is 301. The result shows in Table 4.12.

4.3.3 Results and discussion

The results for both, Olympic boxing and WBA boxing, are produced through the data collected from 1992, 1996, 2000, 2004, 2008 and 2012 matches in ten different weight classes with a total of 1555 matches; and WBA boxing in 2008, 2009, 2010, 2011, 2012, 2013, 2014 in seventeen different weight classes,and a total 301 matches. The game refinement theory was applied in the manner prescribed in Section 2. The results of

Table 4.8: Measures of game refinement for Olympic boxing

Weight class	R
Light flyweight	0.084
Flyweight	0.083
Bantamweight	0.085
Lightweight	0.084
Light welterweight	0.085
Welterweight	0.083
Middleweight	0.084
Light heavyweight	0.085
Heavyweight	0.085
Super heavyweight	0.086
Average	0.084

Table 4.9: Measures of game refinement for Olympic boxing

Year	R
1992 – 2000	0.087
2004 – 2008	0.083
2012	0.078
Average	0.084

the experiment are compared in Table 4.8 for each weight class and Table 4.9 for each group of year(s) in Olympic boxing and in Table 4.12 and Table 4.12 for WBA boxing respectively.

4.3.3.1 Winning percentage method

The nature of boxing game makes it a challenging task to analyse and quantify the excitement and entertainment levelIt is common knowledge that boxing is exciting and entertaining but to show it scientifically is a different story. Since boxing game is a time limit game but can terminate at anytime, the uncertainty is very high and that seems to capture viewer's excitement. The data collected has been shown to reflect this uncertainty and revealed aggressive standard deviation figures. The use of winning percentage to feed into thegame refinement theory has resolved the high variability issues that might intimidate the results. The scores are shown in Table 4.13.

In Olympic boxing, scores are received , among others, through the accuracy and input of the punches. This is highly subjective as the assessment of a punch differs from one judge to another. For example, some punches that make three judges click make our mind feels excited while some punches that make only two judges click will also make us feel excited; but maybe lower than the previous punch. To solve this problem, we

Table 4.10: Statistical results of WBA boxing in Nov-Dec 2010

Month	Weight Class	WS	LS	R
Nov	Mini flyweight	115	114	0.50
		116	112	
		114	115	
	Lightweight	TKO		1.00
		TKO		
		TKO		
	Super middleweight	118	110	0.52
		120	108	
		118	110	
	Super flyweight	TKO		1.00
		TKO		
		TKO		
Dec	Featherweight	119	109	0.52
		119	109	
		120	108	
	Super lightweight	114	111	0.51
		114	111	
		113	112	
	Super flyweight	117	111	0.51
		117	111	
		116	112	
	Bantamweight	115	111	0.52
		116	109	
		117	109	
	Flyweight	116	112	0.49
		110	118	
		115	113	

consider the weight of each punch by capturing the click of the judges. Through this method, we can classify each punch into 6 level of clicks (from 0 click to 5 clicks).

4.3.3.2 The game refinement values for Boxing

According to Table 4.8, we can see that the calculated game refinement measure of amateur boxing, Olympic boxing, falls between $0.083 - 0.086$ and an average of all weight class is 0.084. Likewise, according to Table 4.11, the game refinement measure of professional boxing, WBA boxing, falls between $0.080 - 0.093$ and an average of all weight class is 0.085. According to the previous studies, $0.07 - 0.08$ is an appropriate value which has been shown to be a reliable game refinement measure of many games that have undergone sophistication. The results show that both amateur boxing and professional boxing have a high game refinement measure. This is because boxing is an

TABLE 4.11: Measures of game refinement(R) for WBA and Olympic boxing

Class	*WBA*	*Olympic*
Mini flyweight	0.089	0.000
Light flyweight	0.086	0.084
Flyweight	0.082	0.083
Super flyweight	0.082	0.000
Bantamweight	0.080	0.085
Super bantamweight	0.083	0.000
Featherweight	0.084	0.000
Super featherweight	0.089	0.000
Lightweight	0.086	0.084
Super lightweight	0.081	0.000
Welterweight	0.091	0.083
Super welterweight	0.084	0.000
Middleweight	0.087	0.084
Super middleweight	0.081	0.000
Light Heavyweight	0.080	0.085
Cruiserweight	0.093	0.000
Heavyweight	0.088	0.085
Average	0.085	0.084

TABLE 4.12: Measures of game refinement(R) for WBA boxing

Year	R
2008	0.087
2009	0.084
2010	0.087
2011	0.087
2012	0.083
2013	0.083
2014	0.086
Average	0.085

excessively exciting sport, with a KO and a TKO case that make viewers excited. KO and TKO are significant exit points that allow boxer to instantly win or lose the match

To compare between boxing and other sports done in previous research such as football or volleyball; boxing has higher refinement values.A higher game refinement value means the game is exciting and highly sought after.

4.3.3.3 Amateur Boxing and professional Boxing

We can see that game refinement value for amateur boxing is 0.084 while for professional boxing is 0.085. There are slightly different between amateur boxing and professional boxing [122]. The main difference is the rule. For professional boxing, boxer has to fight

Table 4.13: Some variant examples of Olympic boxing

Weight class	Year	Winner's score	Loser's score
Light flyweight	2004	49	35
Flyweight	1992	14	0
Bantamweight	1992	3	0
Bantamweight	1992	4	0
Bantamweight	2008	1	1
Lightweight	1996	3	3
Lightweight	2004	40	26
Light welterweight	1992	5	0
Light welterweight	2000	42	41
Light welterweight	2004	42	40
Light welterweight	2004	48	31
Welterweight	2004	54	27
Welterweight	1992	6	0
Welterweight	1996	4	4
Middleweight	1992	4	0
Middleweight	2004	41	36
Light heavyweight	2008	5	5
Heavyweight	1992	13	0
Super heavyweight	1992	21	1
Super heavyweight	2008	15	0

twelve rounds, it is remarkably much longer than amateur boxing which has only three rounds. It gives the boxer more chance to blow a KO or TKO because the game time is much longer. Moreover, in professional boxing, protective headgears are not permitted. This also increases the percentage of KO and TKO because headgear is an important equipment to protect the head which is an important part to blow a deadly KO. The statistical results of number of KO and TKO and the percentage was calculated and show in Table 4.14.

Table 4.14: KO and TKO compared between Olympic and WBA

	Olympic	WBA
Number of Match	1555	301
KO	11 (0.71%)	44 (14.62%)
TKO	209 (13.44%)	95 (31.56%)
KO + TKO	220 (14.15%)	139 (46.18%)

Outstandingly, for KO and TKO cases, a sudden death makes a game suddenly ends and the result depends on who makes the knock out. Even though your score is much lower than opponent's score, if you can make a knock out, you will unquestionably win. At the same time, although your score is much higher than your opponent's , if you are knocked out, you will lose the game. The sudden death component makes the game much more exciting because it is an uncertainty of game outcome, you will not know

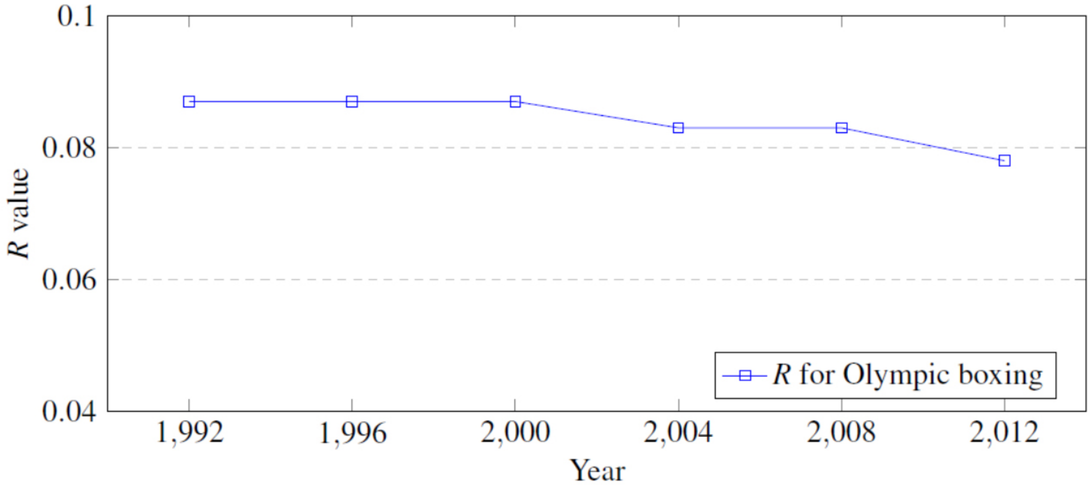

FIGURE 4.8: Comparing of game refinement measure of Olympic boxing

who the winner is until the game ended. KO and TKO case is a sudden death case same as killing king in chess, destroying a final ancient in DotA or getting royal flush in poker card game.

However, there is a difference between boxing and those games mentioned above. For chess and DotA, the killing king and destroying a final ancient are the only condition to win the game, you cannot win by any other way and the attractiveness of the game depends on the method to complete this condition. But in boxing, it provides sometimes such as twelve rounds in professional boxing for letting a boxer to deliver a knock out. However, if the time runs out, the game will end and the winner will be judged by simple scoring system. For poker, the game will end when the limited card is fully used. You will win if you have higher rank card than opponent but if you get royal flush which is the highest rank, it will surely confirm that you will win the match.

We will see that there are some games which have a sudden death case and this factor makes the game very exciting. For extremely exciting game likes boxing, viewer will certainly expect that KO or TKO will happen sometime in the match, the game's result can still be unexpected and it decisively makes the outcome of the game uncertain.

In contrast between amateur boxing and professional boxing, according to Table 4.14, Olympic boxing has little KO and TKO cases, it is very rare to occur, while WBA has a lot of KO and TKO cases. Hence, for research question, what is the difference between amateur boxing and professional boxing, we can shortly conclude that amateur boxing is a professional boxing without sudden death case.

However, for WBA boxing, game refinement measure is still around 0.085 shown in Figure (4.9). It is noticeably separated between each type of boxing. Therefore, for

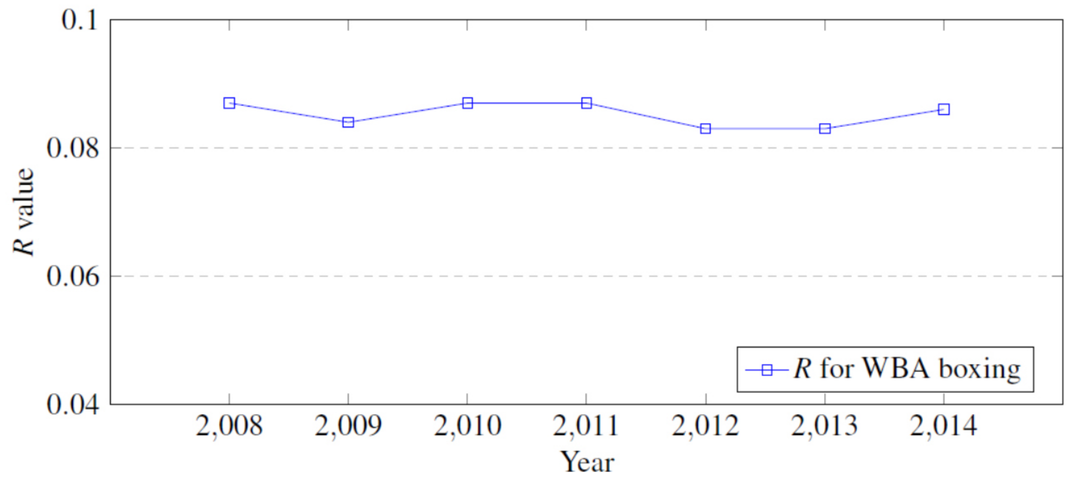

FIGURE 4.9: Comparing of game refinement measure of WBA boxing

research question mentioned before, what is the current trend of boxing's rule changing, we can conclude that amateur boxing, Olympic boxing, will be in suitable zone for all viewer same as another sport while professional boxing, WBA boxing, is still being in exciting zone with a sudden death component for especial viewer.

4.3.4 Summary

Boxing is one of the most exciting and sophisticated sports with a very long history and widely played in many countries. This analysis is an attempt to quantify entertainment impact of boxing. We believe that knock out case is one of the most important reasons which make boxing so exciting. We figured out a game progress model of boxing using the winning percentage method. We have chosen Olympic boxing as a representative of amateur boxing and WBA boxing as a representative of professional boxing because it is official and admitted, then we applied our model. We see that the average of game refinement measure of Olympic boxing and WBA boxing, 0.084 and 0.085, are much high. So, boxing may be suitable for some especial viewers. As we have believed, the essential reason is because KO and TKO case which is a sudden death case that makes the game instantly ends. Nevertheless, it is a very rare case for amateur boxing likes Olympic boxing but it is a normal case for professional boxing likes WBA because the difference of the rules.

For research question, what is the difference between amateur boxing and professional boxing, we can shortly conclude that amateur boxing is a professional boxing without sudden death case. Further, for research question, what is the current trend of boxing's rule changing, we can answer that amateur boxing will be in suitable zone for all viewers while professional boxing is still being in exciting zone for especial viewer.

It is understood that the work presented here is simple yet undeniably important for designers and programmers to make the digital games competitively exciting. Further works may include various data such as from women boxing. Other boxing competitions beside Olympic boxing and WBA boxing such as World Boxing Council (WBC) or International Boxing Organization (IBO) boxing matches. Moreover, we can apply this model in another fighting sport. There are many fighting sports such as Taekwondo, Judo, Karate, Muay Thai, Sumo, Jujutsu and Kick boxing. The fighting sport that use weapon such as fencing can also be included. The core of martial arts and the detailed rules are different but the main rules remain the same; to fight and win the battle. Respectively, we can also use winning percentage method with a handling of sudden death case as illustrated in this study.

4.4 Analysis of Tennis

Tennis is one of the most attractive sports which has been popular in many countries for many centuries. This section describes an application of game refinement theory to tennis, especially in the evaluation of scoring system adapted in Grand Slam tournaments. The Grand Slam tournaments, also called Majors, are the four most important annual tennis events. The Grand Slam consists of the Australian Open, the French Open, the Wimbledon, and the U.S. Open. They have a long history of fame and well-protected rule systems.

However, there exist inconsistencies in the scoring system inside each tournament and between the tournaments. For example, best-of-five rule is applied to men's matches while women's matches use best-of-three rule. In addition, Grand Slam tournaments are consistent in using best-of-five for men while other tournaments use best-of-three for both men and women.

Another example is that nonetheless, tie break rule is universally used in the tournaments, only U.S. Open Tournament accepts tie break in the final set (fifth set of male tournament and third set of female tournament). The differences raise the awareness of researchers about whether the old, conservative rules can keep Majors tournaments interesting and pleasant. By applying game refinement measure from tennis world class tournaments, the results are hoped to give an acceptable answer to these questions.

While many efforts have been devoted to the studies of tennis with a focus on many points [25] [104] [127]. In this section, differences in scoring systems of Grand Slam tournaments are considered. The contribution of the differences on the attractiveness of the tennis tournaments was evaluated with game refinement measurement. Based on the results of the analysis, many controversial questions can be answered completely. By applying game refinement theory to tennis with an appropriate game progress model, we can go deeply while focusing on three main research questions:

1. Is best-of-three scoring system better than best-of-five scoring system?
2. Is final set tiebreak system better than no final set tiebreak system?
3. What is the difference between court surfaces?
4. What is the trends of tennis?

In this section, we first briefly describe some important things about tennis such as its rule and how to collect data. Then, we apply game refinement theory to tennis. Finally, the results obtained are discussed.

4.4.1 Overview of Tennis

Tennis [146] is a racket sport that can be played individually against a single opponent (singles) or between two teams of two players each (doubles). Each player uses a tennis racket that is strung with cord to strike a hollow rubber ball covered with felt over or around a net and into the opponent's court. The object of the game is to play the ball in such a way that the opponent is not able to play a valid return. The player who is unable to return the ball will not gain a point, while the opposite player will.

Tennis is played by millions of recreational players and is also a popular worldwide spectator sport. The four Grand Slam tournaments (also referred to as the "Majors") are especially popular: the Australian Open played on hard courts, the French Open played on red clay courts, Wimbledon played on grass courts, and the US Open played also on hard courts.

The modern game of tennis dates back to 19th century England when lawn tennis was played. Surprisingly, most of the rules have not changed since the lawn tennis days other than a few modifications. The rules of tennis have changed little since the 1890s. Two exceptions are that from 1908 to 1961 the server had to keep one foot on the ground at all times, and the adoption of the tiebreak in the 1970s. A recent addition to professional tennis has been the adoption of electronic review technology coupled with a point challenge system, which allows a player to contest the line call of a point.

According to the 2016 International Tennis Federation Handbook [61], tennis is played by two players in a single match or between two groups of two players in a doubles match. The court is rectangular, 78 feet long by 27 feet wide for singles matches and the doubles court is 78 feet long by 36 feet wide. There is a net suspended by a metal cable or cord dividing the court across the middle attached to the net posts at a height of 3.5 feet. There are also specified regulation on both the ball used in competition as well as acceptable racket specifications a player use.

From [26], standard scoring of a tennis game proceeds as follows: (0) also "love";(15) First Point; (30) Second Point; (40) Third Point (with a tie at 40 being known as a "deuce"; and Fourth or Game Point, except in the case of a deuce, in which case this point becomes known as Advantage. It is concluded shown in Table 4.15.

Table 4.15: How to call score in tennis

Point	Call
No point	"Love"
First point	15
Second point	30
Third point	40
Fourth point	"Game"

This ensures that games must be won by a margin of at least two points (and in the case of a deuce, by two consecutive points). There is some ambivalence, however, since the rules of tennis are open to several alternative scoring methods. Under the No-ad or tiebreak scoring alternative to the advantage rule, a game proceeds to a decisive game point directly from a deuce.

The points of one player in a match are calculated by summing all his points in each game in the match. The following example shown in Table 4.16 explains how to estimate the points of Kei Nishikori in the second game and his points in the first tiebreak game in semi-final of the U.S. Open 2014.

Table 4.16: An example of normal game and tiebreak game in semi-final of the the U.S. Open 2014

<table>
<tr><th>Player1</th><th>Player2</th><th>Type of game</th><th>Result</th></tr>
<tr><td rowspan="2">Kei Nishikori</td><td rowspan="2">Novak Djokovic</td><td>Normal game</td><td>15-0, 15-15, 30-15, 30-30,40-30,
Deuce, A-40, Deuce, A-40, Game</td></tr>
<tr><td>Tiebreak game</td><td>1-0, 2-0, 3-0, 4-0, 4-1,
4-2, 5-2, 5-3, 5-4, 6-4, Game</td></tr>
</table>

In the normal game, we recall that 15 means 1 point, 30 means 2 points, and 40 means 3 points. When the score is tied at 40-40, it is deuce. To the first deuce of the normal game, each player had 4 points. Then Nishikori won the next point or the advantage point. The score was A-40. His score was 5. He needed another point to win the game but he failed. The score was deuce again and Djokovic won one break point. So each

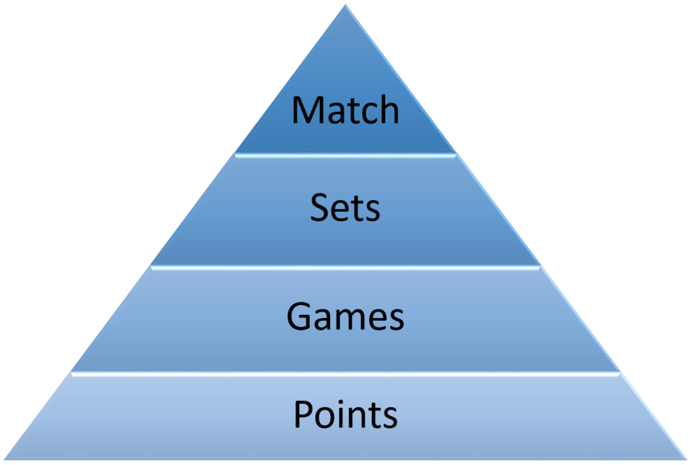

Figure 4.10: Hierachy of Tennis scoring system

player had 5 points. After that, Nishikori won the next advantage point and won another point to end the game. So Nishikori gained 7 points and Djokovic had 5 points in the game. In the tiebreak game, the first player who scores 7 points out of 12 points win the game. It is trivial to see that Nishikori won 7 points and Djokovic had 4 points. With the advantage rule, total points of a player each game are not limited to 4 points.

Similarly, a set of at least six games must be won by a margin of two games under Advantage Set rules, with no limitation to the number of games required. Alternatively in a tiebreak set, winners are determined within a maximum of 13 games. Matches too can be comprised of best-of-three or best-of-five (providing a two-game lead), or any of several alternative Short Sets (first to win four, provided a two-game lead) and tiebreak match systems.

To ensure a clear understanding of point-game-set-match relationship, Figure (4.10) represents the hierarchy of the tennis scoring system.

4.4.2 Game progress model of Tennis

Let us consider some of the major differences between the top level tennis events by applying a score litmit approach of the model.

The Wimbledon, U.S. Open, French Open, and Australian Open are the best known annual tennis events. These tournaments employ best-of-five set matches for men singles and doubles, and best-of-three set matches for women and mixed tournaments. All four tournaments now employ the tiebreak rules, except in the fifth (final) set, which only the U.S. Open allows to be decided in a tiebreaker.

Masters 1000 includes: IndianWells, Miami, Monte-Carlo, Madrid, Rome, Canada, Cincinnati, Shanghai, and Paris Masters, all of which are played in the best-of-three, tiebreak format. For the purpose of comparison to the other formats we only need consider only the data of Masters 1000 Men.

The intended research requires complete point score totals for the matches, which are not published on ITF. While IBM SlamTracker tracks point totals per player, their dataset is not available for independent research. Observations are taken from data of recent years four major world tournaments the Majors, and Masters 1000 tournaments as they were found on a popular online source [132].

The data consists of total points for the winning side, and total match points for every Majors (also Grand Slam) tournament match from 2004 to 2014, 5080 matches; and Masters 1000 Men's tournaments from 2006 to 2014, 4536 matches.

In total, we collect 9616 matches shown in Table 4.17.

TABLE 4.17: Total number of matches in data analysis

Tournament	Matches/Tour	Tours/year	Matches/year	Interval	Total
Grand Slams	127	4	508	10 years	5080
Masters	63	9	567	8 years	4536

By applying game refinement theory with score limit approach, we obtain many interesting data shown in Table 4.18.

TABLE 4.18: Measures of game refinement for tennis

Association	Rule	Tournament	Surface	Avg.	Avg	R
ATP (Men's)	5 sets tiebreak	U.S. Open	Hard	105.4	193.3	0.053
	No T/B final	Australian	Hard	119.8	218.9	0.05
	No T/B final	French Open	Clay	119.5	217.9	0.05
	No T/B final	Wimbeledon	Grass	122.8	226.6	0.049
WTA (Women's)	3 sets tiebreak	U.S. Open	Hard	77.9	138	0.064
	No T/B final	Australian	Hard	77.53	139.4	0.063
	No T/B final	French Open	Clay	77.4	139.4	0.063
	No T/B final	Wimbeledon	Grass	77.96	140.8	0.062
Masters1000 (Men's)	3 sets tiebreak	(all)	(all)	82.37	141.8	0.064

4.4.3 Results and discussion

In this section, we show the results from applying game refinement theory to tennis. Also, the results obtained are discussed.

4.4.3.1 Best-of-three and best-of-five scoring system

Best-of-five match only occurs in Major tennis events, especially for men. The system ensures that one player can outperform his opponent. It is hard to understand why the best-of-five system is only use in men's tournaments instead of women. This leads to famous tennis players including Andy Murray to suggest women to play the same length of match as the men if they are getting paid equally. And female players such as Martina Navratilova thinks that men should play best-of-three instead of best-of-five matches. Game refinement measure is hoped to show which of the two systems is better.

According to Table 4.18, best-of-five matches seem to appear only in the domain of men's Major tournaments. Most competitions are decided in best-of-three, including mixed doubles, women, and all Masters 1000 events. Of the many proximal or distal reasons which could be imagined for having a best-of-five for men's competition in the Majors, perhaps tradition or the test of endurance might receive the most mention.

The topic seems to be under somewhat vigorous discussion in the tennis community [87]. Many good arguments exist on both sides of course, so the discussion here will be confined to the mathematical realm.

However, it is interesting that two typical arguments, one of each for and one against the 5-set match, are of a mathematical nature. One states that the best-of-five is more interesting than a best-of-three contest, because best-of-three format (say, Player A vs. Player B) can produce only four distinct three-set outcomes: ABA, ABB, BAA and BAB. A five-set match, on the other hand, has 12 possible set outcomes, creating a host of ways momentum can twist and turn over the course of several hours.

Opposite this, it is opined that best-of-five-set matches have little effect on outcome in men's tennis [87], with Miller citing that only about 2.5 to 3.5 percent of game outcomes change between the third and fifth sets.

From a game refinement standpoint, it can be seen that the R values of best-of-three competitions more closely conform to the R values found in other games under the model than do best-of-five. This is clearly the greatest difference in the measures of game refinement among the tournaments.

It can also be noticed that, although it is true that there is a gender difference to consider between best-of-three and best-of-five contests in the Majors, the R values for men's best-of-three contests in the Masters 1000 are quite nearly the same as those for women's in the Majors. For the eight years of observation from Masters 1000 matches, we found a R values of 0.064, while those for the three-set Majors ranged from 0.062 to

0.064. For this reason we believe that the difference in R values between best-of-three and best-of-five-set competition is not attributable to gender.

TABLE 4.19: Comparison between best-of-three and best-of-five scoring system

Tournament	R value	
	Best-of-three	Best-of-five
U.S. Open	0.064	0.053
Australian Open	0.063	0.05
French Open	0.063	0.05
Wimbledon	0.062	0.049

Table 4.19 shows that R value of best-of-three variant is greater than best-of-five version. The best-of-three scoring system should be used to keep the major tournaments interesting and pleasant.

4.4.3.2 The necessity of tiebreak in final set

The aim of a tiebreak game is to settle down a set when the score is tied at six games all. Without tiebreak game, it can take a very long time to finish a set. The longest tennis match was recorded in the Wimbledon where the final set is an advantage. The match between John Isner and Nicolas Mahut lasts about 11 hours. An advantage final set requires consistence and endurance from players. But the players have to face the risks of injuries which can ruin their remaining matches in a tournament or even their own career. At the moment, only the U.S. Open has tiebreak final set. The main concern of the section is whether it is necessary to have a tiebreak final set.

Table 4.18 shows game refinement values of No tiebreak and tiebreak final set versions of tennis scoring system. With tiebreak, the U.S. Open has the smallest average total points per match (193.3 for men and 138 for women). The R value of the U.S. Open (0.053 for men and 0.064 for women) is bigger than those of the other 3 Grand Slam tournaments, which indicates that using tiebreak final set makes the game more exciting than no tiebreak final set. The difference is small with women tournaments. The reason is that women tournament use best-of-three match instead of best-of-five match in men tournament. How game sophistication varies with best-of-three and best-of-five systems is the next concern in the work.

How well the male tournaments in the Majors can become if they change their scoring system? Fortunately, the changes were applied to other tournaments: the Masters 1000. Masters 1000 are the secondly famous annual tennis tournaments. It's mandatory for famous players to join in the tournaments. 1000 stands for the number of points a champion can gain for his ranking. Unlike Grand Slam, Masters 1000 tournaments have a new scoring system with best-of-three and tiebreak for all sets. Table 4.20 shows the average R values of Majors and Masters 1000 for male tournaments.

Table 4.20: Comparison between tiebreak and no tiebreak system

Variants	Tournament	R value
No tiebreak final set and best-of-five match	Australian Open French Open Wimbledon	0.05
Tiebreak final set and best-of-five match	U.S. Open	0.053
Tiebreak final set and best-of-three match	Masters 1000	0.064

According to Table 4.20, game refinement value increases when tiebreak final set and best-of-three system is used.

4.4.3.3 Court surfaces

Tennis is played on different court surfaces. There are four main types of tennis court surfaces: carpet, clay, grass and hard surface. Each type has a variety of characteristics which affect speed and style of play. The four Majors tournaments are played on various surfaces and there used to be player's domination based on a surface type. For example, Pete Sampras dominated Wimbledon's grass courts or Raphael Nadal is "the king of clay". What surface brings up the most excitement for tennis game? In this section, game refinement theory is used to examine how different court surfaces affect tennis game sophistication.

First of all, we explain the characteristic of court surfaces and player's sufficient style of play:

- Clay court is crushed shale, stone or brick, and deep red. The French Open is the only Grand Slam tournament played on clay, and the greatest clay tournament in the world. Clay is the slowest surface, with high-bouncing balls; the balls tend to bounce instead of skidding. As a result, the clay court takes away some advantage of big servers. It favors baseline players, and not the huge servers with the serve-and-volley style of playing.

- The grass court is the fastest surface and is used at the Wimbledon. Up until 1970s, grass was used at three of the four Majors tournaments except French Open. Grass is the surface of serve-and-volley style game. It is an unpredictable surface. The bounce of the ball depends on how recently the grass has been mowed, the health of the grass and how much it's previously been played on.

- Hard courts are the most common at tennis sports centers for both indoor and outdoor courts. Both the Australian Open and U.S. Open use types of hard court surfaces; the Australian Open uses a synthetic surface called Plexicushion and the U.S. Open uses a product called Deco Turf. A hard court is usually made of asphalt or concrete that has a layer of padding, which is the covered with paint that has sand mixed in. The more sand is added to the paint, the slower the surface becomes. Hard courts are considered the middle ground between clay and grass; it's a fast surface, but the flat, uniform surface is more predictable without the surprises of a grass or clay court. It's a good surface for a broader range of player styles.

- "Carpet" in tennis means any removable court covering. Indoor arenas store rolls of rubber-backed court surfacing and install it temporarily for tennis events, however they are not in use any more for professional events. A short piled form of artificial turf infilled with sand is used for some outdoor courts, particularly in Asia. Carpet is generally a fast surface, faster than hardcourt, with low bounce [61].

Beginning the 1990s, tennis fans began to complain that the faster surfaces afforded a more boring tennis-viewing experience. As a result, many tournaments started to slow down their surface in the past decade. Wimbledon organizers changed the composition of their grass, the Australian Open and U.S. Open added more sand to their courts to slow down balls.

The R value in Table 4.18 among the group of 3 tournaments: the Australian Open, the French Open, and the Wimbledon which have the same scoring system but different court surface are analyzed. The Wimbledon has the fastest court surface, so it has the smallest R value (0.049 for men and 0.062 for women). R values of the tournaments with hard and clay surfaces are almost similar. There are some small differences between the three R values, which indicates that the organizers seem to be successful in slowing down their surfaces and the 3 kinds of court surfaces do not have much different influences on tennis game excitement in the Majors tournaments, nowadays.

Court surfaces affect game speed, which has a measurable effect on refinement values for tennis. Under the model it would seem that speeding up, and reducing point totals would result in higher R values closer to a 0.07 to 0.08 range. However, faster, lower bouncing balls also result in fewer and shorter rallies which tends to have an opposing effect on enjoyability.

Somewhere is a balance to be struck, and the ITF [61] has been slowing court surfaces down in recent years. There is a wide variety of court surfaces available, the specifications of which directly affect the particulars of bounce, skid, traction, cushion, reliability, absorbency, light refraction and maintenance. Grass courts are the most traditional, and the fastest of tennis courts used in the ITF. Changes to the underlying soil composition, grass species, length and moisture content provide some controllability to these factors.

Wimbledon is the only major played on grass. Clay has a reputation as the slowest surface, with the greatest bounce. The French Open is played on clay. The surfaces known as hard courts are layers of synthetic materials over a uniform underlayment of concrete or asphalt. The upper layers are a textured, pigmented, sanded, resin-bound acrylic coating designed to give a controlled playing experience somewhere between that of grass and clay.

Knowingly or unknowingly, the governing boards of many sports have also been actively involved in controlling the breadth and length of their respective games. Prior to 1974 the U.S. Open and French Open were played on grass courts, which have been known for their fast, sometimes slippery surfaces, and given to a particular type of game play. The last remaining Major to continue the tradition of grass courts, the Wimbledon has acted to harden the underlayment and slow the upper surface.

The differences in court types today are not as great as they have been in the past, and the gap is narrowing. Because of these adjustments, there is very little remaining difference in R values directly attributable to the differences between court surfaces in the Major tournaments today.

4.4.3.4 Comparing by years

It has long been understood that games must continue to develop and evolve in order to maintain fairness, safety, or enjoyability for players and spectators. According to what we know from game refinement theory, in those games which are the surviving variants of long processes of sophistication, game information tends to arrive in approximately 0.07 to 0.08 R value for many games and game types.

Game Refinement theorizes that 0.07 to 0.08 R value is optimally accessible both for play and observation, although we can only speculate why this may be. Below this value, as in the case of best-of-five tennis matches, players or spectators might experience a lack of excitement awaiting the arrival of new game information. Also, above this value is not good which previous studies confirm.

The Figure (4.11) shows how game refinement values increased each time changes in the history of men's tennis tournaments.

Before 1974, the U.S. Open, the French Open and the Wimbledon were played on grass courts, which is proved as the most inefficient court surface in the Section 4.4.3.3. From

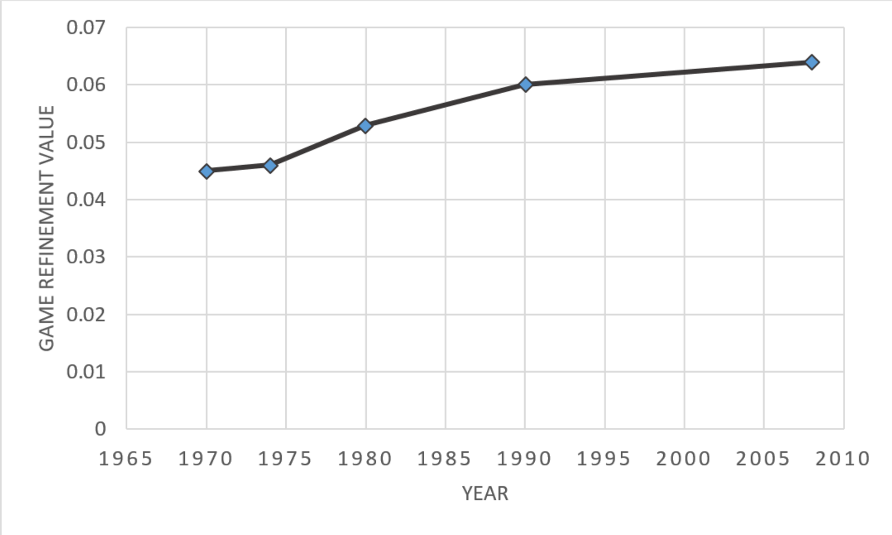

FIGURE 4.11: Evolution of Tennis game refinement value in male tournaments

1974, only the Wimbledon used court surface and its organizer started to slow down the surface. In 1980, tiebreak was universally applied in the Grand Slam. The tiebreak helped speed up the game and kept it exciting. However, the conservative organizers of the Australian Open, the French Open, and the Wimbledon decided that tiebreak can be used for all sets except final sets. From 1990, the Masters 1000 was introduced, tiebreak for all sets and best-of-three match was used for men's tournaments except the final match.

In the past 2 decades, there were no considerable changes in tennis rules instead of some small modifications. In 2008, the Masters 1000's scoring system became more advance than those of Grand Slam when all matches in the Masters 1000 are best-of-three with tiebreak. The changes in rule made the game sophistication comes closer and closer to the balanced window, 0.07-0.08. Although the value has not been inside the desired window, but the increment in game refinement values indicates the effeteness of tiebreak and best-of-three in scoring system. It may be the right time for Grand Slam's conservative organizers to change and keep the tournaments exciting.

Perhaps more significantly, in terms of game refinement theory, from 1980 tiebreak sets were introduced to the Majors except in the final set, which the U.S. Open already allows to be settled by tiebreak. Tennis is a traditional and conservatively controlled sport, but applying the score limit approach we see these increments gently moving tennis closer toward a 0.07 R value.

4.4.4 Summary

Tennis is one of the most attractive and sophisticated sports with a long history and widely played in many countries. We raised three main research questions: "Is best-of-three scoring system better than best-of-five scoring system?", "Is final set tiebreak system better than no final set tiebreak system?" and "What is the difference between court surfaces?" For this purpose, game refinement measure was employed and round-match approach was used to figure out a game progress model of tennis. We made observations from recent years, four major world tournaments or Majors, and Masters 1000 tournaments data obtained from a reliable online source [132].

We observed the differences within the scoring systems inside and between Grand Slam tournaments. There are numerous debates about keeping or changing the differences but there is no convincing evidence to ensure that the refinement can help to make the tournaments exciting. Should we use best-of-three or best-of-five system? Should we have a tiebreak final set or not? After comparing the value of game refinement measurement of the variants, our experiments has shown that best-of-three system is much better than best-of-five and tiebreak is necessary in final set. Also, for the difference between court surfaces, we found that the difference is not as great as it has been in the past since there are many adjustments.

4.5 Conclusion

This work is an attempt to present an application of game refinement theory to answer the various research questions in many kinds of sports. We select baseball as a representative of time limit sports, boxing as a representative of fighting sports and tennis as a representative of score limit sports. Then, for each sports, we explain how to figure out a reasonable game progress model and apply it to the target sports.

For baseball, the model of time limit sports from game refinement theory was applied to baseball, which considers the innings as the time. The baseball leagues both major and minor from many countries were chosen. The range of game refinement value of baseball is around 0.065, which is lower than the sophisticated zone. Moreover, it trends to decreasing in the future. Thus, baseball will not attract some viewers. The value of game refinement is decreasing because the rules does not change for a while. In contrast, the rules of other popular sports have been changed, which lead their game refinement value to the sophisticated zone. In term of countries, baseball was the most popular sports in Japan. However, the game refinement value shows that Japan has the lowest value among other countries compared. The reason is that the rules of Japan leagues that difference, which trend to get score harder than others. In term of leagues, Rookie league has the highest game refinement value. In contrast, Major league has the lowest one. Thus, the average scores in each game of Major league is lower than Rookies league. The reason is that the performance gap between batters and pitchers. Lastly, the softball's game refinement value falls in the sophisticated zone due to the effort for changing its rules, which are a huge impact to the games, promote the scoring than baseball.

For boxing, a new approach for fighting game called winning percentage method is applied. We can see that the average of game refinement measure of Olympic boxing and WBA boxing, 0.084 and 0.085, are much high. So, boxing may suitable for some especial viewers. As we have believed, the essential reason is because KO and TKO case which is a sudden death case that makes the game instantly ends. Nevertheless, it is a very rare case for amateur boxing likes Olympic boxing but it is a normal case for professional boxing likes WBA because the difference of rule. For research question, what is the difference between amateur boxing and professional boxing, we can shortly conclude that amateur boxing is a professional boxing without sudden death case. Further, for research question, what is the current trend of boxing's rule changing, we can answer that amateur boxing will be in suitable zone for all viewer while professional boxing is still being in exciting zone for especial viewer.

For tennis, a use of score limit approach of game refinement theory applying to tennis was described. We are interested in the differences within the scoring systems inside and between Grand Slam tournaments. There are numerous debates about keeping or changing the differences but there is no convincing evidence to ensure the refinement can help to make the tournaments exciting. Should we use best-of-three or best-of-five system? Should we have a tiebreak final set or not? After comparing the value of game refinement measurement of the variants, our experiments show that best-of-three system is much better than best-of-five and tiebreak is necessary in final set. Also, for the difference between court surfaces, we found that the difference is not as great as it has been in the past since there are many adjustments. For the trend of tennis, it will go to the sophisticated zone, $0.07 - 0.08$.

In conclusion, it is practically proved that game refinement theory can effectively be used in many many domains of games such as classical board games, video games and sports, including baseball, boxing and tennis, by establishing a reasonable game information progress model. It can be used as a helpful tool to measure an attractiveness and sophistication of a game considered and it also enables game designers to make a target game more sophisticated. As confirmed by our previous studies, we observed that suitable game refinement measure is around $0.07 - 0.08$.

It is understood that the work presented here is simple yet undeniably important for designers and programmers to make the digital games competitively exciting. Further works may include various data. For example, in boxing, we may use data from women boxing, other boxing competitions beside Olympic boxing and WBA boxing such as World Boxing Council (WBC) or International Boxing Organization (IBO) boxing matches. Also, in tennis and baseball, we can gather data from another league and another year. Respectively, we can apply game refinement theory to another sport. For example, we can apply winning percentage method from boxing to another fighting sport. There are many fighting sports such as Taekwondo, Judo, Karate, Muay Thai, Sumo, Jujutsu and Kick boxing. The fighting sports that use weapon such as fencing can also be included. The core of martial arts and the detailed rules are different but the main rules remain the same; to fight and win the battle. By this reason, we can also use winning percentage method with a handling of sudden death case as illustrated in this study. Furthermore, we can use these exist models as a basis in order to create a new model which is suitable for the target game considered.

Chapter 5

Finding Reasonable Settings in Games

This chapter is an updated and abridged version of work previously published in

(10) Anunpattana Punyawee, Chetprayoon Panumate and Hiroyuki Iida. Finding Comfortable Settings of Snake Game using Game Refinement Measurement. 8th International Conference on Computer Science and its Applications (CSA-16). in press.

(11) Chetprayoon Panumate, Jean-Christophe Terillion and Hiroyuki Iida. A Game Informatical Analysis of RoShamBo. 2016 Conference on Technologies and Applications of Artificial Intelligence (TAAI 2016). submitted.

This chapter presents the last example of using game refinement theory to improve the entertainment impact of game. RoShamBo, a well-known gesture game, and Snake game, an well-known arcade maze video game, are chosen as a testbed. We present an application of game refinement theory to RoShamBo and Snake game. Also, it can be use for finding comfortable settings and answering some research questions. We show how game refinement theory can be used as a reliable tool to improve the entertainment impact of game.

5.1 Introduction

As we have introduced in Chapter 3 that setting is one of the most important factors in games. It can lead the game to be so exciting or so boring. This is the reason why we can improve the entertainment impact of the game by finding reasonable settings in games. Therefore, this chapter present an approach to find reasonable settings in games.

RoShamBo [135] is a two-player game where both players pick a move that will defeat their opponent. The moves are rock, paper and scissors. Each player chooses a move in secret. After a predetermined amount of time, player will reveal their chosen move simultaneously. The winner is determined by the one with the symbol that is stronger than the other. The standard rule is that rock wins paper, paper wins scissors and scissors win rock.

With this simple rule, RoShamBo is commonly used for simply making a decision (similar to a heads/tails coin toss) or to peacefully settle a dispute [95]. However, children are not the only ones to play it. Adults have also been known to use it when they cannot agree or would rather leave a decision to chance [38].

Beside this, Snake game is an arcade maze video game played by controlling snake. Originally, a snake can be controlled in four directions consist of up, down, left, right. The goal of this game is to survive and collect the scores as many as possible. The player loses when the snake runs into the screen border, a trail, or another obstacle. The snake will be longer when player collect more score. With these simple rules, Snake game is so popular and is played for many decades.

In this chapter, we aims to find an reasonable setting of RoShamBo and Snake game. Also, we raise some research questions and answer it by applying game refinement theory. We show how to figure out the reasonable game progress model and perform the experiment. Section 5.2 shows the game informatical analysis of RoShamBo. Section 5.3 shows the game informatical analysis of Snake game. Then, the concluding remarks are given in Section 5.4.

5.2 Game Informatical Analysis of RoShamBo

This chapter present an approach to find reasonable settings in RoShamBo. The structure of this chapter is as follows. First of all, we briefly introduces the basic idea of RoShamBo in Section 5.2.1. Next, in Section 5.2.2, we show how to apply game refinement theory to RoShamBo. Then, Section 5.2.3 and Section 5.2.4 explain how to apply game refinement theory to RoShamBo's variants and another RoShamBo's based game. Finally, the obtained results are analyzed in Section 5.2.5.

5.2.1 Overview of RoShamBo

The standard rule is depicted in Figure 5.1 where arrow means win. If the result is a draw, then another round may be played.

Below we quote from [95] the definitions of rock, paper and scissors.

- Rock: A clenched fist represents the rock. Rock defeats the scissors by blunting it but it is susceptible to paper which wraps around it.
- Paper: The paper symbol is conveyed by a flat, outstretched palm with the fingers held against each other. Paper defeats rock by wrapping around it but is beaten by the scissors that cut through it.
- Scissors: The scissors are represented similar to the rock except that the index and middle fingers which are fully extended towards the opponent. Scissors can defeat paper by slicing through it but are defeated by the rock, which dulls their blade.

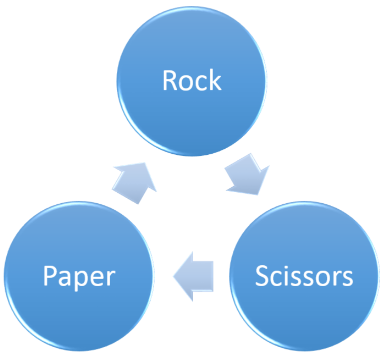

FIGURE 5.1: The rule of standard RoShamBo

RoShamBo is also known by different names such as Rock Paper Scissors, Bato Bato Pik, Jan Ken Po and Quartz Parchment Shears [136]. Whatever its name, it is widely played in many countries with a long history and no one knows exactly who invented this game. In this study, we use the name RoShamBo.

While many efforts have been devoted to the study of RoShamBo with a focus on different points such as algorithm analysis [51], imitation of opponents gestures [27] and smart robot [105]. This chapter raises new research questions as follow.

With the classical question "why RoShamBo has been popular so long time", we can go deeply while focusing on three main research questions: (1) Why three possible move style (rock, paper and scissors) has been used, (2) What are the appropriate game-refinement values for RoShamBo, and (3) How is it possible to improve RoShamBo. To tackle these challenges, game refinement theory [59] [58] [128] is used as an important tool for game informatical analysis. We consider mathematical equations and reasonable game information progress model to derive a game refinement measure for RoShamBo. Then many RoShamBo variants are considered and evaluated by adjusting many factors.

5.2.2 Game refinement measure for RoShamBo

A family of RoShamBo game can be denoted by $RSB(n, b, s, r)$.

- n is the number of players.
- b is the number of possible options.
- s is the number of winning regulations.
- r is the number of rounds required to be a winner.

In this section, with the appropriate approach, a game informatical analysis of $RSB(n, b, s, r)$ is performed by using computer players and simulating a million RoShamBo games. In order to apply the boardgame approach to RoShamBo, according to Equation (2.5), we have to find the average branching factor (say b) and average game length (say D).

In classical RoShamBo, we can denote by using $RSB(2, 3, 1, 1)$. For average branching factor b, it is limited because each time player has to choose one possible choice among rock, scissors and paper. It can be easily counted as $b = 3$ constantly.

For average game length d, it can be calculated as shown below.

Table 5.1: Possible results for each round in classical RoShamBo

Player	Opponent	Result
Rock	Rock	Draw
	Paper	Lose
	Scissors	Win
Paper	Rock	Win
	Paper	Draw
	Scissors	Lose
Scissors	Rock	Lose
	Paper	Win
	Scissors	Draw

According to Table 5.1, for each round, the chance that game will be continued because it is draw is 3 from 9 possible results. Also, the chance that game will be ended is 6 from 9 possible results which can be calculated by chance of win plus chance of lose. We can reduce this to 2/3 and 1/3 shown in Equation (5.1).

$$
\begin{aligned}
GameEnd &= \frac{1}{3} \\
GameContinue &= \frac{2}{3}
\end{aligned} \tag{5.1}
$$

Then, we can calculate the chance that game will be ended in which round by using $GameContinue$ and $GameEnd$

$$
\begin{aligned}
End_1 &= GameEnd \\
End_2 &= (GameContinue)GameEnd \\
End_3 &= (GameContinue)^2 GameEnd \\
End_4 &= (GameContinue)^3 GameEnd \\
End_5 &= (GameContinue)^4 GameEnd \\
&\dots \\
End_i &= (GameContinue)^{(n-1)} GameEnd
\end{aligned} \tag{5.2}
$$

End_1 means the chance that game will be ended in the first round. So, End_i means the chance that game will be ended in the round number i.

Next, we can find average game length (D) shown in Equation (5.3).

$$
D = \sum_{i=1}^{\infty} i(GameContinue)^{(i-1)} GameEnd \tag{5.3}
$$

For RoShamBo, we substitute $GameContinue$ and $GameEnd$ from Equation (5.1). It will be in Equation (5.4)

$$
D = \sum_{i=1}^{\infty} i(\frac{1}{3})^{(i-1)}(\frac{2}{3}) \approx 1.5 \tag{5.4}
$$

Finally, we can calculate R from Equation (2.5) by using $b = 3$ and $d = 1.5$ which is explained above. Equation (5.5) shows that R value of classical RoShamBo is 1.155.

$$
GR = \frac{\sqrt{b}}{d} = \frac{\sqrt{3}}{1.5} = 1.155 \tag{5.5}
$$

5.2.3 RoShamBo variants

From Equation (5.5), we finish our first attempt to apply game refinement theory to classical RoShamBo. However, in order to improve RoShamBo, we consider the related factors in RoShamBo.

5.2.3.1 Number of players

From Section 5.2.2, we consider only two players RoShamBo. However, in fact, we can play RoShamBo with three players or more. Hence, we propose n which stands for the number of players.

For classical RoShamBo with $n = 2$, we have two players consist of p_1 and p_2 and the winner will be judged based on the rule shown in Figure (5.1). However, what happens when we increase number of players, for example from $n = 2$ to $n = 3$.

TABLE 5.2: Possible results for each round in classical RoShamBo

p_1	p_2	p_3	Result
Rock	Rock	Rock	Draw
		Paper	n=2
		Scissors	End
	Paper	Rock	End
		Paper	n=2
		Scissors	Draw
	Scissors	Rock	P=2
		Paper	Draw
		Scissors	End
Paper	Rock	Rock	End
		Paper	n=2
		Scissors	Draw
	Paper	Rock	P=2
		Paper	Draw
		Scissors	End
	Scissors	Rock	Draw
		Paper	End
		Scissors	n=2
Scissors	Rock	Rock	P=2
		Paper	Draw
		Scissors	End
	Paper	Rock	Draw
		Paper	End
		Scissors	n=2
	Scissors	Rock	End
		Paper	n=2
		Scissors	Draw

According to Table 5.2, we will see that when we use $n = 3$, it is more complex than $n = 2$ because we have a case that the game is not ended and it is also not draw which can be called reducing player such as from $n = 3$ to $n = 2$. It means there is a player who lose in this round, however, there still have remained players who do not lose in this round. Therefore, we have to play a new round by cutting the loser in this round. Then, we play again until we have only one remained player who can be called as a winner.

To explain the rule more clearly, for each round, we have to check that the choice b_i chosen by player p_i can kill another choice chosen by another player or not. Then, we check every choice's effect. Finally, we can find a survivor choice which is not defeated by another choice. For draw result, it will be occurred in only two case shown below.

1. There is no loser in this round. For example, if every player selects the same choice so it will be draw in this round.

2. There is no survivor in this round. For example, if every choice can kill each other completely, it will be draw in this round.

If it is not draw, we cut these losers then we continue to find the winner by playing new round.

Next, imagine that we increase n, for example, $n = 4$, $n = 5$ or $n = 10$. That means there are many cases and many rounds that we have to play in order to find the true winner. To solve this complicated problem, we create a simulated adjustable RoShamBo program. Then, we can find the average game length (d) by using this program. We assume that the technique that players in this simulated game use is random technique.

TABLE 5.3: Measures of game refinement for RoShamBo (adjusting n)

n	d	R
2	1.5	1.155
3	2.25	0.770
4	3.28	0.528
5	4.84	0.358
6	7.33	0.236
7	11.55	0.150
8	18.85	0.092
9	31.65	0.055
10	55.25	0.031
11	97.58	0.018
12	176.22	0.010
13	322.82	0.005
14	592.33	0.003
15	1099.57	0.002

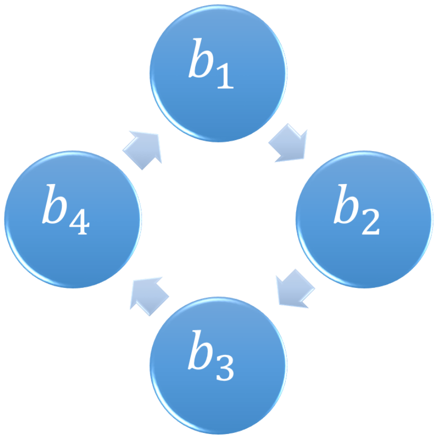

FIGURE 5.2: Rule of RoShamBo when $b = 4$

Table 5.3 shows the results by using simulated RoShamBo program, $RSB(n, 3, 1, 1)$. For each n, we run 1,000,000 games and we compute the average game length (d). Then, we can find R value.

5.2.3.2 Number of possible options

According to Table 5.1 and Figure (5.1), we can say that it is for classical RoShamBo which its branching factor equals 3 ($b = 3$). Suppose that player has four choices ($b = 4$) consist of b_1, b_2, b_3 and b_4, the rule will be following Figure (5.2). We can conclude the possible results shown in Table 5.4

TABLE 5.4: Possible results for each round when $b = 4$

Player	Opponent	Result
b_1	b_1	Draw
	b_2	Win
	b_3	Draw
	b_4	Lose
b_2	b_1	Lose
	b_2	Draw
	b_3	Win
	b_4	Draw
b_3	b_1	Draw
	b_2	Lose
	b_3	Draw
	b_4	Lose
b_4	b_1	Win
	b_2	Draw
	b_3	Lose
	b_4	Draw

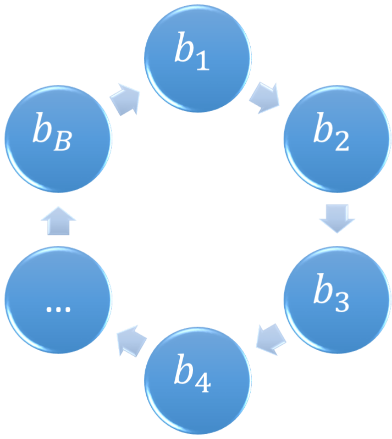

FIGURE 5.3: General rule with b branching factors

By this idea, we can conclude that the chance that game will be continued because it is draw is 8 from 16 possible results. Also, the chance that game will be ended is 8 from 16 possible results which can be calculated by chance of win plus chance of lose. We can reduce this to $GameContinue = 2/4$ and $GameEnd = 2/4$ respectively.

Next, Figure (5.3) shows the general rule when we have b branching factors.

With the same idea, if we use $b = 5$, the chance that game will be continued will be 15 from 25 possible results and the chance that game will be ended is 10 from 25 possible results. We can reduce this to $GameContinue = 3/5$ and $GameEnd = 2/5$.

To summarize, $GameContinue$ and $GameEnd$ from Equation (5.1) will be a general case written in Equation (5.6)

$$\begin{aligned} GameEnd &= \frac{b-2}{b} \\ GameContinue &= \frac{2}{b} \end{aligned} \tag{5.6}$$

In order to find average game length (d), we substitute Equation (5.6) to Equation (5.3). It will be an Equation (5.7)

$$d = \sum_{i=1}^{\infty} i(\frac{b-2}{b})^{(i-1)}(\frac{2}{b}) \tag{5.7}$$

Finally, we can calculate R value in many points of view by adjusting b, $RSB(2, b, 1, 1)$, shown in Table 5.5.

TABLE 5.5: Measures of game refinement for RoShamBo (adjusting b)

b	d	R
3	1.5	1.155
4	2	1
5	2.5	0.894
6	3	0.817
7	3.5	0.756
8	4	0.707
9	4.5	0.667
10	5	0.633
11	5.5	0.603
12	6	0.577
13	6.5	0.555
14	7	0.535
15	7.5	0.516
16	8	0.5
17	8.5	0.485
18	9	0.471
19	9.5	0.459
20	10	0.447

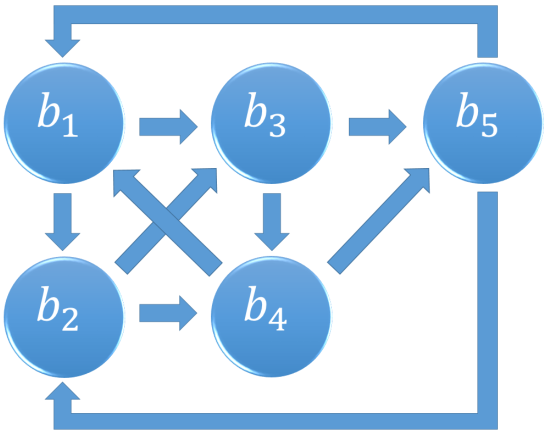

FIGURE 5.4: Rule when $s = 2$ and $b = 5$

5.2.3.3 Number of strategies

From Figure (5.1), the classical rule of RoShamBo is that it is a one way cycle. Also, its general form is shown in Figure (5.3). By considering this point, we can improve it by adding more arrow in Figure (5.3) which means we increase number of ways to win. We will call it as s, number of winning regulations, which stands for the number of arrows go out from one choice. For example, if $s = 2$, one choice chosen can win other two choices and also one choice can be defeated by two choices. Its rule is shown in Figure (5.4) and the conclusion of possible results of each round is shown in Table 5.6.

TABLE 5.6: Possible results for each round when $s = 2$ and $b = 5$

Player	Opponent	Result
b_1	b_1	Draw
	b_2	Win
	b_3	Win
	b_4	Lose
	b_5	Lose
b_2	b_1	Lose
	b_2	Draw
	b_3	Win
	b_4	Win
	b_5	Lose
b_3	b_1	Lose
	b_2	Lose
	b_3	Draw
	b_4	Win
	b_5	Win
b_4	b_1	Win
	b_2	Lose
	b_3	Lose
	b_4	Draw
	b_5	Win
b_5	b_1	Win
	b_2	Win
	b_3	Lose
	b_4	Lose
	b_5	Draw

We will see that we cannot create RoShamBo game with $s = 2$ when $b < 5$ because the branching factor is not enough to create a balance cycle. For $s = 2$, we need 2 branching factors which are defeated by choice b_i, and also we need 2 branching factors which are able to defeat choice b_i. Finally, we has to count choice b_i itself. Therefore, in case that $s = 2$, we need at least 5 branching factors (2 for losing, 2 for winning and 1 for itself). A generic relation of s and b is shown in Equation (5.8).

$$b >= 2s + 1 \tag{5.8}$$

When we add s, *GameContinue* and *GameEnd* from Equation (5.6) will be changed because it is a general form when $s = 1$. So, it general form with s is shown in Equation (5.9)

$$\begin{aligned} GameEnd &= \frac{b-2s}{b} \\ GameContinue &= \frac{2s}{b} \end{aligned} \tag{5.9}$$

To find average game length (d), we substitute Equation (5.9) to Equation (5.3). It will be an Equation (5.10)

$$d = \sum_{i=1}^{\infty} i(\frac{b-2s}{b})^{(i-1)}(\frac{2s}{b}) \tag{5.10}$$

To clarify this idea, we run many experiments by setting $n = 2$, adjusting s and we set $b = 2s + 1$, $RSB(2, 2s + 1, s, 1)$. It is shown in Table 5.7.

TABLE 5.7: Measures of game refinement for RoShamBo (adjusting s)

s	b	d	R
1	3	1.5	1.155
2	5	1.25	1.789
3	7	1.1667	2.268
4	9	1.125	2.667
5	11	1.1	3.015
6	13	1.0833	3.328
7	15	1.0714	3.615
8	17	1.0625	3.881
9	19	1.0556	4.130
10	21	1.05	4.364
11	23	1.0455	4.587
12	25	1.0417	4.8
13	27	1.0385	5.004
14	29	1.0357	5.200
15	31	1.0333	5.388
16	33	1.0312	5.571
17	35	1.0294	5.747
18	37	1.0278	5.918
19	39	1.0263	6.085
20	41	1.025	6.247

5.2.3.4 Number of rounds

However, one round of RoShamBo is quite short. Actually, sometimes, we play RoShamBo more than one round so as to find a winner precisely such as in RoShamBo AI programming contest [13]. The game can be extended to a series of matches, where the winner is whoever wins a majority of matches (for example, best 2 out of 3) [43].

Hence, from Equation (2.5), we should add r which stands for the number of rounds required to be a winner. For example, if player needs 5 rounds for winning, that means player has to play at least 5 rounds but not more than 10 rounds.

By adding r, it directly affects to d. We try many r by using $RSB(2, 3, 1, r)$ shown in Table 5.8.

TABLE 5.8: Measures of game refinement for RoShamBo (adjusting r)

r	d	R
1	1.50	1.155
2	3.75	0.462
3	6.18	0.280
4	8.72	0.199
5	11.30	0.153
6	13.93	0.124
7	16.60	0.104
8	19.28	0.090
9	21.99	0.079
10	24.71	0.070
11	27.45	0.063
12	30.18	0.057
13	32.94	0.053
14	35.71	0.049
15	38.46	0.045

5.2.4 Analysis of RoShamBo-based games

There are many game which is directly related to RoShamBo. It uses classical RoShamBo as a basis idea of game play. Then, the detail of rule is adjusted based on each culture. In this section, we focus on GuRiKo and King PaoYingShub which are the most popular application of classical RoShamBo from Japan and Thailand respectively.

5.2.4.1 GuRiKo

GuRiKo is one of the applications of RoShamBo which is widely played in Japan. It is directly related to RoShamBo because its rule is almost similar to classical RoShamBo. The purpose of this game is to go to the top of the stair. Player starts from the bottom of the stair then try to go to the top of the stair. In order to move up each step of stair, player has to win another player by playing RoShamBo. Therefore, this game can be concluded that player has to go to the top of the stair by winning RoShamBo. Moreover, the difference of winning method is related to the steps that you can go. It is concluded shown in Table 5.9.

Table 5.9: GuRiKo's Rule

Win by using	*Steps recieved*
Rock	3
Paper	6
Scissors	6

For this investigation, we set the research question that what is the appropriate number of stair steps? To solve this problem, we apply game refinement theory to GuRiKo by using our model mentioned above shown in Table 5.10.

Table 5.10: Measures of game refinement for GuRiKo

n	*Steps*	d	R
2	10	5.03	0.344
2	20	9.75	0.178
2	30	14.38	0.120
2	40	20.81	0.083
2	43	22.43	0.077
2	48	24.05	0.072
2	50	25.7	0.067
3	10	8.87	0.195
3	20	18.55	0.093
3	22	21.54	0.080
3	27	24.73	0.070
3	30	28.16	0.062

5.2.4.2 King PaoYingShub

With the same idea as GuRiKo, King PaoYingShub from Thailand sets the objective of the game as to go to the top of stair by using RoShamBo similarly. However, for King PaoYingShub, whichever choice you choose, it gives you only one step that you can go up. But, if you lose, you have to go down one step. Moreover, in King PaoYingShub, player starts from the middle of the stair so that player can go up and go down. Moreover, if player is in the bottom of the stair player cannot go down more.

To answer the same research question as GuRiKo, what is the appropriate number of stair steps?, we apply game refinement theory to King PaoYingShub by using our model mentioned above. The results are shown in Table 5.11.

5.2.4.3 Other variants

Players have developed numerous cultural and personal variations on the game, from simply playing the same game with different objects, to expanding into more weapons

TABLE 5.11: Measures of game refinement for King PaoYingShub

n	*Steps*	d	R
2	3	5.97	0.290
2	4	6.00	0.289
2	5	13.46	0.129
2	6	13.51	0.128
2	7	23.89	0.073
2	8	24.05	0.072
2	9	37.42	0.046
2	10	37.61	0.046
3	3	18.03	0.096
3	4	31.49	0.055
3	5	101.87	0.017

and rules [147]. For exmaple, in Korea, a two-player upgraded version exists by the name muk-jji-ppa [52]. In muk-jji-ppa, each person starts with a regular rock-paper-scissors game. The current winner has to say their next hand and change their hand to the corresponding one. This is usually done by shouting. In the same moment, the loser shows their selected hand. The point is to get your opponent to make the same hand as you in order to be a winner. However, if your hands are different, the new winner starts the process again.

In Malaysia, scissors are replaced by bird, represented with the finger tips of five fingers brought together to form a beak. The open palm represents water. Bird beats water (by drinking it); stone beats bird (by hitting it); and stone loses to water (because it sinks in it).

Also, there are many RoShamBo-based games with some additional weapons such as pierre-papier-ciseaux-puits which can be called as stone-paper-scissors-well from French. Among them, one of the most popular RoShamBo-based games is rock-paper-scissors-spock-lizard which has five weapons. It was invented by Sam Kass and Karen Bryla [1]. Spock is signified with the Star Trek Vulcan salute, while lizard is shown by forming the hand into a sock-puppet-like mouth. Spock smashes scissors and vaporizes rock; he is poisoned by lizard and disproven by paper. Lizard poisons Spock and eats paper; it is crushed by rock and decapitated by scissors. It can be written as $RSB(n, 5, 2, r)$.

5.2.5 Discussion

We start this section by concluding our research questions shown below.

1. Why human being selected RoShamBo?

2. What are the appropriate values for classical RoShamBo and its applications?

   2.1 What is the appropriate number of players for classical RoShamBo?

   2.2 What is the appropriate number of rounds for classical RoShamBo?

   2.3 What is the appropriate number of stair steps for GuRiKo?

   2.4 What is the appropriate number of stair steps for King PaoYingShub?

3. How to improve RoShamBo?

To answer our research questions, we propose a simulated RoShamBo program and many equations for analyzing RoShamBo, also, many ways to improve RoShamBo by adjusting some factors related. Then, we applied game refinement theory in the manner prescribed in Section 5.2. The results are compared in Table 5.3, Table 5.5, Table 5.7,Table 5.8, Table 5.10 and Table 5.11.

5.2.5.1 Classical RoShamBo

In the previous studies, it is found that sophisticated games would have R value between $0.07 - 0.08$. We see that the result of classical RoShamBo in Equation (5.5, 1.155, is considerably different from previous studies. This can be justified that the purpose of RoShamBo game is to make a simple decision [95]. So, it is entirely different from another game in previous studies whose purpose is enjoyment [29] [141].

By definition, a game presents an entertaining challenge to the player or players, a challenge which the player or players can understand and may be able to succeed at using their wits, dexterity, luck, or some combination thereof [112]. We will see that luck is one of the most important factors in game design. In order to make a fair decision, game designer has to increase luck factor so that every player has a fair chance to win this game. Also, RoShamBo is an absolutely fair game by using luck. For another game, it may mainly use other factors such as skill or strategy because the purpose is different.

Moreover, one factor which is important for making a decision in real situation is time. Generally, we do not spend long time for an unimportant decision. If the decision is important, we may use another method which is more reasonable. But for an unimportant decision, we can use a simple decision making game likes RoShamBo. Usually, it take just only 5-10 seconds to complete this game so it is suitable for a real situation.

To limit the time of playing, it means we have to limit d as low as we can. Obviously, from Equation (2.5), if d is low, R value will be high. But if we try to increase d, the time spending for playing is surely increased because the chance of draw is increased and we have to play it again so as to find a winner. Then, that long time spending is not suitable for a human's decision making in real situation.

To summary, we can conclude that the reason why R value of RoShamBo is quite high compared to another sophisticated game is the purpose of game. RoShamBo's purpose is to make a quick and simple decision while another game's purpose is an entertainment or so on.

To answer the first research question, why human being selected RoShamBo, we can shortly say that because its purpose. Human always has to make a simple decision with a limited time and RoShamBo is suitable for this situation.

We also can use another game such as an extension of RoShamBo whose R value is near classical RoShamBo. It can work as a simple decision making tool as well as RoShamBo works. However, the problem is the rule is quite difficult to understand. To be selected by human being, the game's rule need to be as uncomplicated as possible. It should be easy to understand, remember and teach so that the game can be inherited by human for many generations. That is why RoShamBo has been used as a simple decision making tool for a long time.

5.2.5.2 Extensions of RoShamBo

We can simply say that RoShamBo game has four attributes consist of n, b, s and r which can be denoted by $RSB(n, b, s, r)$. For classical RoShamBo, it will be $RSB(2, 3, 1, 1)$. Next, we try to change many factors in order to improve RoShamBo.

1. Increasing number of players (p) If we increase p, the game length (d) is increased. This is because the chance of draw is increased and we have to play many rounds in order to find the true winner. Therefore, R is decreased.

2. Increasing branching factors (b) If we increase b without increasing s, that means we increase the chance of draw which makes the game longer. So, it means we increase d and decrease R.

3. Increasing number of strategies (s) If we increase s, it automatically means we increase b. However, in total, it decreased d so R is increased.

4. Increasing number of rounds (r) If we increase r, it means we increase d so R is decreased.

In order to make a sophisticated ordinary game whose purpose is an entertainment, we have to adjust the rule. Many previous studies confirm that suitable game refinement

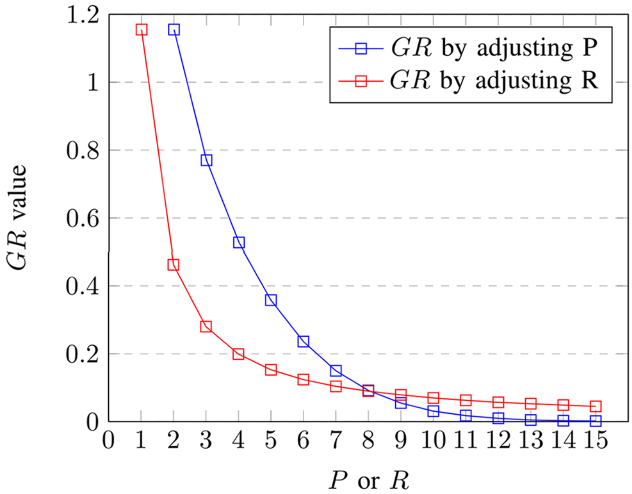

FIGURE 5.5: R value by adjusting p or r

value is around $0.07 - 0.08$. So, we can improve R value of RoShamBo by adjusting its factor mentioned above.

Therefore, to answer the research question number 2.1, what is the appropriate number of players for classical RoShamBo? According to Table 5.3, we will see that if we set other attributes as classical RoShamBo, $b = 3$, $s = 1$ and $r = 1$, $n = 8$ and $n = 9$ will be an appropriate number of players which makes R value near the sophisticated zone.

Also, for research question number 2.2, what is the appropriate number of rounds for classical RoShamBo? According to Table 5.8, we will see that if we set other attributes as classical RoShamBo, $n = 2$, $b = 3$ and $s = 1$, $r = 9$ and $r = 10$ will be an appropriate number of rounds which makes R value in the sophisticated zone.

By comparing n and r, we show our results in Figure (5.5).

Moreover, by combining these ideas, we set $b = 3$ and $s = 1$ which are normal setting of classical RoShamBo then we adjust n and r. The results are shown in Table 5.12

TABLE 5.12: Measures of game refinement for RoShamBo (adjusting p and r)

n	r	d	R
3	3	11.37	0.152
3	4	16.52	0.105
3	5	21.91	0.079
4	3	19.33	0.090
4	4	28.69	0.060
4	5	38.64	0.045
5	3	32.06	0.054
5	4	48.32	0.036
5	5	65.88	0.026

According to Table 5.12, we explore that if we use $n = 3$ and $r = 5$, R value will be in sophisticated zone.

Next, to improve this game, we can adjust every attribute. By implementing this idea, we show some examples of an extension of RoShamBo whose R value is in sophisticated zone in Table 5.13

Table 5.13: Some examples of extensions of RoShamBo

n	b	s	r	R
2	3	1	10	0.077
2	4	1	9	0.074
2	5	1	8	0.074
2	5	2	15	0.079
2	7	3	20	0.075
2	9	4	25	0.071
3	3	1	5	0.079
4	5	2	5	0.071

We will see that, in order to have R value in the sophisticated zone, the game needs time to play. This idea is same as many games applied before such as academic board game, sports and video game. Every game needs long time to play. Also, by improving this point, this new RoShamBo needs time to play. Furthermore, its purpose is changed, from making a decision to entertainment.

To summarize, we propose many factors which directly related to RoShamBo's fun consist of n, b, s and r. All of this can answer the third research question, how to improve RoShamBo.

Nevertheless, the fun of a game likes RoShamBo is not derived only from battle component. We can also add some intangible components which make player feels enjoyable such as improving graphic or adding game story. All of this answers the second research question, how to improve RoShamBo.

5.2.5.3 RoShamBo-based games

To clarify this idea, we apply our model to applications of RoShamBo such as GuRiKo and King PaoYingShub. These two games are quite similar so we use the same question in research question number 2.3 and 2.4, what is the appropriate number of stair steps?

Due to its implementation, we can consider the number of steps as r. Therefore, these applications of RoShamBo can be written as $RSB(n, 3, 1, r)$. Because, $b = 3$ and $s = 1$ are set constantly and we try to find an appropriate r for each n.

For GuRiKo, according to Table 5.10, we can conclude that, for $n = 2$ GuRiKo, the appropriate number of steps is 43-48, and for $n = 3$ GuRiKo, the appropriate number of steps is 22-27 whose R values are in sophisticated zone, $0.07 - 0.08$. These can answer the research question number 2.3.

Also, to answer the research question number 2.4, we use the results from Table 5.11. We can conclude that, for $n = 2$ King PaoYingShub, the appropriate number of steps is 7-8, and for $n = 3$ King PaoYingShub, the appropriate steps should be around 3-4.

We also can apply this idea to GuRiKo and King PaoYingShub whose n more than 3 in order to find an appropriate number of steps of stair for that n chosen. Also, we can find another game which is directly related to RoShamBo and apply our game refinement model so that we can find some appropriate values.

5.3 Game Informatical Analysis of Snake Game

This chapter present an approach to find reasonable settings in Snake game. The structure of this chapter is as follows. We firstly introduces the foundation idea of Snake game in Section 5.3.1. Next, in Section 5.3.2, we show how to apply game refinement theory to RoShamBo. Then, Section 5.3.3 analyzes Snake game by focusing on each factor.

5.3.1 Overview of Snake game

Snake game [22] is a type of arcade maze games originally developing from "Blockade" which has been developed by Gremlin Industries and published by Sega in October 1976 [9]. There are many clone games based on Blockade game inspiration, e.g., Bigfoot Bonkers, Surround, Dominos, etc [44] [28]. Snake games are considered to be a skillful game, players try to achieve maximum score as high as possible. We show, in Figure 5.6, a screenshot of Snake game on Nokia 3310.

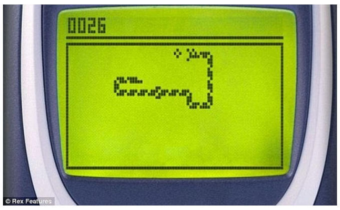

FIGURE 5.6: A screenshot of Snake game on Nokia 3310

Nokia is well-known for putting Snake game in their phones [131] [50]. Towards the end of the year 2000, Nokia released one of the most successful phones, Nokia 3310, whose Snake game is so popular. Snake games are still included in some new phones from Nokia and available for all platforms. The history of Snake games on Nokia mobile phones [148] is shown in Table 5.14. We can control the snake in Snake game by using

Table 5.14: A historical overview of Snake games on Nokia mobile phones

Year	Version
1997	Snake
2000	Snake II
2002	Snake EX
2003	Snake EX2
2005	Snakes
2005	Snake III
2006	Snake Xenzia
2008	Snakes Subsonic

the four direction buttons relative to the direction it is heading in. The snake increases its speed as it gets longer by eating fruits. The goal of this game is to collect fruits as many as possible. A player loses when the snake crashes the wall or crashes itself [33].

However, some variants have the regulation that the snake's speed will go up when eating fruits and the snake can pass through the wall, then appear on the opposite side. Actually, there are many factors affected to Snake game. It can be summarized as shown below.

Notation 1. A family of snake games can be denoted by **SNG(s, a, m, c, l)** with the initial physical speed of snake in frame per second **s**, acceleration of snake when eating each fruit in frame per second **a**, size of the map **m** specifying in terms of total area $(pixel \times pixel)$, wall condition **c** specifying trigger between 0 and 1 which means no wall and have a wall respectively, and initial length of snake in pixel **l**.

While there are many variants of Snake game, some of them are so popular and some of them are not. The main reason behind this fact is the setting of some variables in the game. Its setting may often considerably affect the entertaining aspect which leads the game to be exciting or boring. That is the reason why we assume that each setting of video game has its own reasons behind this, and therefore finding a comfortable setting can be significantly important to maximize the entertainment in playing the game considered. Thus, this paper presents an approach to find comfortable settings of Snake game. To tackle this challenge, game refinement theory [128] is used as an important tool for the assessment. We consider mathematical equations and a reasonable game information progress model to derive a game refinement measure for Snake game and its variants. Then $SNG(s, a, m, c, l)$ are considered and evaluated by adjusting each factor.

5.3.2 Game refinement measure for Snake game

In soccer and basketball [129], a game progress model is constructed with a focus on the number of goals G and the number of attacks or shot attempts T. We then obtain $R = \frac{\sqrt{G}}{T}$. With the same idea, in Snake game, we construct a functional game progress model with a focus on the average total scores per game (say G) and average number of moves counted from the number of clicking button (say T) as shown in Equation (5.11).

$$R = \frac{\sqrt{G}}{T} = \frac{\sqrt{(average\ total\ scores\ per\ game)}}{average\ total\ number\ of\ moves\ per\ game} \tag{5.11}$$

We first develop the AI for snake game in order to perform many experiments. The basic objective of ordinary players is to eat fruits with low move and avoid hitting itself. However, most human players would have their own mistakes, so AI should be randomized in the position of snake head for miss turning of snake to the fruits. Due to the limitation that AI cannot make a decision spontaneously like a human, we try to implement AI which can be assumed as a human player as much as possible, which is described in Algorithm 5.

Algorithm 5 Snake AI

```
procedure DirectionDecision
    i ← random from 0 to 2
    fruitPos ← random fruit position
    snakeX and snakeY ← position of snake's head
    headIndex ← i
    snakeX and snakeY ← headIndex
    if snakeY > fruitPos then
        direction = up
    else if snakeY < fruitPos then
        direction = down
    else if snakeY = fruitPos then
        if snakeX > fruitPos then
            direction = left
        else if snakeX < fruitPos then
            direction = right
    end if
    return direction
end procedure
```

Then, Table 5.15 shows the comparison of results using AI and human. For each setting, we play 250 games and the version used for this test is the replica of the game as it currently exists on Nokia. The value of G and T which are collected from AI might be slightly different from human data because human players can develop their skills while playing. However, the R value is not significantly different, so we can use AI instead of human. Moreover, we will see that, with the wall condition, game will be more difficult because it is easier to die, so it makes R value higher.

TABLE 5.15: An example of game refinement measures for *SNG(15, 1, 240, c, 3)*

	c	G	T	R
Human	0	16.5	72.6	0.056
	1	6.1	31	0.080
AI	0	4.44	34.98	0.060
	1	2.18	17.03	0.087

5.3.3 Snake Game Variants

We demonstrate an analysis of Snake game variants $SNG(s, a, m, c, l)$ for finding comfortable settings. For this purpose we develop a computer player (AI) of playing Snake game.

5.3.3.1 Analysis of $SNG(s, 1, 240, 1, 3)$: Speed

Table 5.16 shows the measures of game refinement for the analysis of snake's speed with $5 \leq s \leq 30$. If the speed is high, the controlling is too difficult which will increase the number of moves for surviving. Increasing s means increasing T because player will make many mistakes while moving with the fast speed, so the number of moves, T, is increased and it leads R value decreased respectively.

TABLE 5.16: Measures of game refinement for $SNG(s, 1, 240, 1, 3)$

s	G	T	R
5	5.51	12.8	0.183
10	4.28	11.8	0.175
15	2.18	17.03	0.087
20	2.54	19.46	0.082
25	2.34	20.07	0.076
30	2.06	20.24	0.071

TABLE 5.17: Measures of game refinement for $SNG(15, a, 240, 1, 3)$

a	G	T	R
0.2	2.26	16.85	0.089
0.5	2.31	18.62	0.082
1	2.18	17.03	0.087
2	1.96	16.024	0.087
3	2.43	18.88	0.083
4	2.32	21.92	0.079

5.3.3.2 Analysis of $SNG(15, a, 240, 1, 3)$: Acceleration

Table 5.17 shows the measures of game refinement for snake's acceleration, a. In detail, the information about acceleration cannot clearly observe while playing because players should focus on their own turn in Snake game. In game progress and case of higher speed, players will die quickly if they reach at some points. In sense of entertaining, acceleration is quite less impact to this game progress model. When game is progressing for many games, the average number of moves is the same on each value of acceleration.

5.3.3.3 Analysis of $SNG(s, 1, m, 1, 3)$: Speed and Map Size

Table 5.18 shows the measures of game refinement for snake's speed in different map size. We focus on the values of s and m with $15 \le s \le 30$ and $m = (160, 240, 336, 448, 960)$ by using $pixel \times pixel$. The size of map will control the game length because if the map is larger, it will increase the number of survival ways for players. By adjusting these factors, we can identify the comfortable settings of Snake game.

TABLE 5.18: Measures of game refinement for $SNG(s, 1, m, 1, 3)$

s	m	G	T	R
15	160	1.80	12.8	0.112
20	160	1.96	13.7	0.102
25	160	1.91	14.23	0.097
30	160	1.96	14.50	0.0.96
15	240	2.18	17.03	0.087
15	336	2.50	20.33	0.079
20	336	2.30	19.92	0.076
25	336	2.40	20.90	0.074
30	336	2.52	21.67	0.073
15	448	2.44	22.40	0.070
20	448	2.62	22.43	0.072
15	960	3.12	27.80	0.063

TABLE 5.19: Measures of game refinement for $SNG(s, 1, 240, 1, l)$

s	l	G	T	R
5	3	5.51	12.80	0.183
15	3	2.18	17.03	0.087
30	3	2.06	20.24	0.071
5	6	4.79	11.09	0.197
15	6	1.25	10.18	0.110
30	6	1.16	10.23	0.105
5	9	4.55	10.48	0.200
15	9	0.92	17.40	0.130
30	9	0.82	8.90	0.102

5.3.3.4 Analysis of $SNG(s, 1, 240, 1, l)$: Speed and Length of Snake

Table 5.19 shows the measures of game refinement with a focus on the initial snake's length and speed. Each variant of Snake game has different length of snake and speed. In practical, snake length is difficult to see clearly in player's view because the screen does not show obviously pixel. However, a big gap in different length of snake will affect to the difficulty obviously. By adjusting these factors, we can find the snake game with the comfortable settings, $R = 0.07 - 0.08$, by setting $SNG(30, 1, 240, 1, 3)$.

5.4 Conclusion

In conclusion, this paper presents an approach to find the reasonable settings of RoShambo and Snake game. Game informatical analysis using game refinement measure was carried out with a focus on the game of RoShamBo and Snake. For this purpose a computer program has been developed for the simulated play of various matches of RoShamBo games and Snake.

In RoShamBo, from the results, we conjecture that $RSB(n, 3, 1, 1)$ is best to play with $n = 8$ or 9, whereas $RSB(2, 3, 1, r)$ is best to play with $r = 9$ or 10. Our analysis of RoShamBo variants enables to pick up several plausible settings of its regulations under the assumption that its R values are within sophisticated zone of game refinement. Moreover, it is confirmed that popular RoShamBo-based games such as GuRiKo from Japan and King PaoYingShub from Thailand have been played with reasonable settings to optimize the game refinement values. Further investigation may be made in the direction to apply the game informatical analysis to many RoShamBo-based games from other countries.

In Snake, we analyzed what factors had more effect to the entertainment of the game. We explored that the physical speed of snake, map size and wall condition can be seen impact apparently in player's view. Under the assumption that the sophisticated zone of R value is around $0.07 - 0.08$ which many previous studies confirm, we found the comfortable settings of Snake game for $SNG(s, a, m, c, l)$. First, s is around 25-30 for $SNG(s, 1, 240, 1, 3)$. Secondly, $SNG(15, 1, 240, c, 3)$ can play either $c = 0$ and $c = 1$, condition of the wall indicates the difficulty of the game. In the same manner with $SNG(15, 1, m, 1, 3)$, it is best to play with the appropriate map size as same as a mobile phone. For other analysis, we explored that acceleration a and length of snake l are less impact to game entertainment, this is because these two variants are subsections of speed. Further works can improve in many points such as improving the quality of AI, adding more factors and investigating in another domain.

Chapter 6

Emotional AI in Real-Time Multiplayer Fighting Games

This chapter is an updated and abridged version of work previously published in

(12) Chetprayoon Panumate, Youichiro Miyake and Hiroyuki Iida. A Generic Model for Emotional AI in Real-Time Multiplayer Fighting Games. The Games and Learning Alliance conference (GALA 2016). submitted.

This chapter presents another approach to improve the entertainment impact of game. We change the aspect from rule designer aspect to AI designer aspect. We cannot refuse that player usually spend a lot of time interact with many AIs in video games. That is why we can improve the entertainment impact of game from this side.

Therefore, this study explores a generic model for emotional AI in the domain of real-time multiplayer fighting games. It proposes a notion of emotional component with a focus on two aspects: handling the emotion of an agent and using the emotion for decision making. The emotional component contains three main factors: personality, memory and mood. All three factors are interrelated and may affect the decision making. Moreover, some enhancements are incorporated to make the proposed emotional AI more realistic.

6.1 Introduction

Calculators are much faster than humans. Machines are more efficient and accurate than humans in many domains. Computer games are stronger than humans, for example in chess [93]. However, there are still many remaining challenges which prevent humans and AI from melding together, such as creativity and human-likeness. One of which, the emotion of humans [47] [82] is a big issue.

The game industry is concerned with player experience [102], including sadness, anger, fear and enjoyment. In video games, there are many factors which may affect player experience, such as the story, animation and rules of the game considered. On the other hand, a game player usually spends a lot of time while interacting with characters which are automatically controlled by AI [89]. This implies that AI can be one of the most important factors which directly affects player experience.

While building a strong game, AI has been the main-stream in the domain of board games such as computer chess [16]. In the game industry, the main focus has been upon AI which makes users' experience more enjoyable. This research topic is called *entertainment AI* [88]. It is expected that an advanced game AI technique would enable us to achieve a better player experience. AI containing an emotional component would advance towards the next step while creating and exploring a new player experience. Thus, this study aims to investigate the emotional AI.

Many efforts have been devoted to the study of emotional AI with a focus on different aspects [35] [14] [46]. In this study, we propose a generic model of emotional AI in the domain of real-time multiplayer fighting games. Then, we verify the efficiency of the proposed emotional AI model through an actual assessment with human subjects. Importantly, the efficiency in this context does not mean the strength of an emotional AI, but it means to know if one can sense exactly the emotions of emotional AI. This is our main research question in this study.

The structure of this chapter is as follows. We first introduce some background ideas in Section 6.2. Then, Section 6.3 describes the details of the emotional AI such as the agent structure and the emotional AI model. Next, we implement an emotional AI on our simulated game and the experimental test performed with human subjects is presented in Section 6.4. Finally, the results obtained are discussed in Section 6.5, and concluding remarks are given in Section 6.6.

6.2 Emotional AI

In order to construct a generic model of emotional AI, we have to look at its foundation. Therefore, this section briefly introduces some background ideas.

6.2.1 Definition of emotional AI

Emotions are facts not only in humans but also animals. The emotional control is widespread in nature and seems to serve several crucial roles in animals and humans alike. It is an essential part of our life because it influences how we think, adapt, learn, behave and communicate with others [83].

We start with a fundamental question: what is emotional AI? There are many attempts to define emotions in various aspects and fields such as in biology, philosophy and psychology [110] [62]. Among them, a comprehensive definition of emotions is given in [66] which we summarize below.

> Emotion is a complex set of interactions among subjective and objective factors, mediated by neural hormonal systems, which can
>
> 1. give rise to affective experiences such as feelings of arousal, pleasure/displeasure;
> 2. generate cognitive processes such as emotionally relevant perceptual effects, appraisals, labeling processes;
> 3. activate widespread physiological adjustments to the arousing conditions;
> 4. lead to behavior that is often, but not always, expressive, goal-directed, and adaptive.

Five issues of emotional intelligence are discussed in [116], including knowing one's emotions, managing emotions, motivating oneself, recognizing emotions in others and handling relationships.

While the previous studies give us many definitions of emotions in various aspects as described above, in this study we propose our definition of emotions by simplifying those definitions.

Definition 1. **Emotion** is a reason why an agent takes a certain action that may not be necessarily taken.

For example, when an agent is angry, some changes would happen in the moving speed, face, voice and speaking style while using impolite words or expressing negative emotions through certain actions. We then see that the reasons behind these actions are emotions. So, we can simply describe emotional AI as an AI displaying emotions.

Definition 2. **Emotional AI** is an AI displaying emotions.

6.2.2 Emotional AI modeling

Principally, conventional AI will optimize the answer under some limited constraints in order to complete the goal [118]. However, it is likely that emotions do not always make the answer optimized. So, can we conclude that emotions bring about irrational actions? The answer should be no. Our proposition is that emotions bring about rational actions based on an emotional component [84] [79].

In [34] some neurological evidence supports the fact that emotions play an important and active role in the human decision-making process. Also, it is assumed that an emotional component is one subjective component in our brain which directly manages the emotions [84]. Then, our emotions affect our actions respectively. So, this research focuses on the emotional component. We consider in two points: (1) how an emotional component generates emotions and (2) how emotions affect actions. Therefore, the problem of emotion can be separated in two aspects: emotion generation and emotion effect [54].

1. Emotion generation is a method to construct a model of emotions inside an agent such as one's personality or current feeling.
2. Emotion effect is a method which focuses on how emotions generated affect the output of an agent such as facial expressions, verbal expressions and behavioral expressions.

In order to accomplish our goal, we need to evaluate both emotion generation and emotion effect. However, it is impossible to do so directly because both emotion generation and emotion effect occur inside AI. So, in our experiment a tester or subject will see actions, the output of AI, which would be affected by emotions. It is assumed that the emotions are derived from the emotional component. Thus, we can check whether testers can recognize any emotion expressed by emotional AI correctly or not while looking back from AI's action to AI's emotion [6]. The details will be explained in Section 6.4.

As the simplest emotional AI, we use only personality as the whole emotional component [70]. We first generate the personality of an agent. Then, this personality will control the agent's actions completely. For example, a greedy agent would prioritize money first, a timid agent would go far from a dangerous place and try to avoid any battles.

However, human actions are more complex. Sometimes, with the same person and same conditions, the reaction may be different. This is because humans have a mood which may change in time [4]. For example, one's mood today may definitely be different from that of yesterday's.

Additionally, various roles of emotions for AI such as action selection, goal management and memory control are discussed in [83]. For example, you remember that you hate a person because the person did something bad to you, so you are biased when you have to do something related to that person. Also, those memories can be deleted depending on time.

With these reasons, we cannot use only personality as the whole emotional component. Therefore, in this study it is assumed that an emotional component consists of the following three factors: Personality, Memory and Mood.

Nevertheless, emotions are very in range and it is hard to construct an emotional AI model which can be applied to every domain of games. So, we focus on a real-time fighting game in this study. We do not focus on more complex emotions such as shyness, love, flirt or jealousy. Because those emotions may not be related to fighting games or it is hard to relate them to the game.

Then, in this domain we try to construct a model as general as possible. This means that the model can be applied to every game in this domain by adjusting some factors. Moreover, the model should be simple and easy to implement. So some rare cases or other cases where it is hard to understand or implement are not considered.

6.3 A Generic Model for Emotional AI

In this section we explain a generic model for emotional AI precisely. We present each factor in the emotional component which consists of personality, memory and mood. We then consider some enhancements which make the emotional AI more realistic such as physical state, human-likeness and intention. These concepts also directly affect the emotional component.

6.3.1 Emotional component

In order to implement emotions in AI, we have to add an emotional component [84]. We propose an emotional AI agent architecture based on the concept described in [114]. In this model the inside of the agent is connected to the outside environment via a sensor, decision making and actuators by combining personality, memory and mood.

We start with the simplest agent structure as shown in Figure 6.1. The real world is the environment in games such as the item's position, another player's position and another player's status. Sensors will obtain information from the input which comes from the

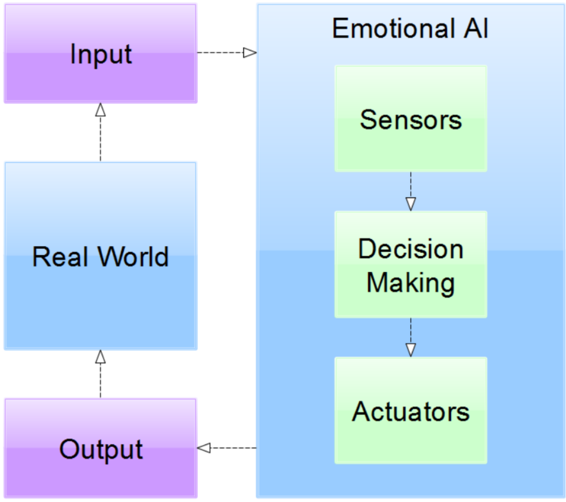

FIGURE 6.1: The simplest structure of Emotional AI

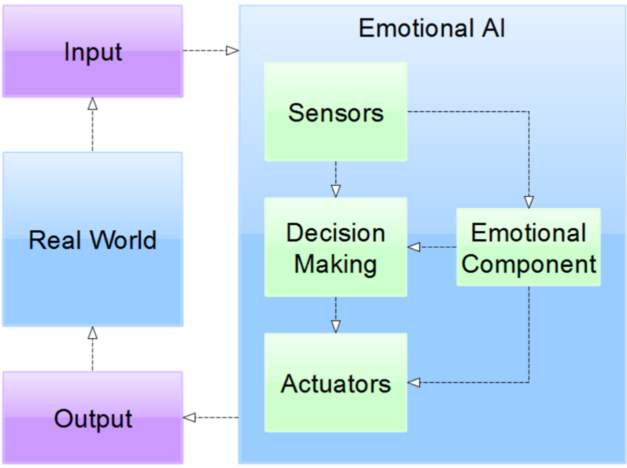

FIGURE 6.2: Emotional component added

real world and send it to the inside of the agent. Decision making will receive input from the sensors and make a decision based on their algorithm. Then, the output is sent to actuators. Actuators will make an action as an output of the agent. Then the output will affect the real world. It will be a cycle as the one shown and information will flow through this cycle.

Next, we add an emotional component in the agent architecture as shown in Figure 6.2. We see that sensors receive input from the real world. The input will be sent to the decision making and emotional components. Then, the emotional component will generate its current emotion, and this emotion will affect decision making. Moreover, this emotion also affects actuators. For example, an emotion affects decision making such as choosing what to do next. But, for actuators the emotion affects actuators such as walking path, movement speed and shooting speed.

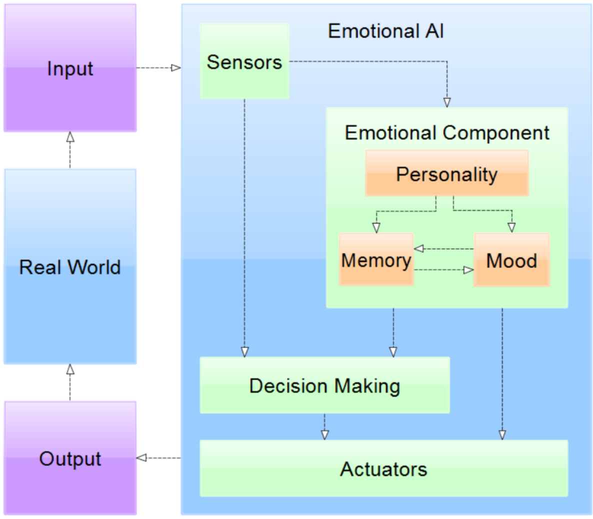

Figure 6.3: A structure of Emotional AI

By combining these meaningful factors: personality, memory and mood, we can construct an emotional AI agent structure as shown in Figure 6.3. We see that personality affects mood and memory. Also, memory and mood affect each other.

6.3.1.1 Personality

Personality is represented by the initialized values that make an agent unique. For example, we may use real human's classification system like the sixteen personality types [19], the OCEAN model [31] [86] or the PEN model [36] for complex games. Personality should be carefully classified depending on the game that is considered. For example, in simple games such as Pokemon, using limited personality may be enough.

Moreover, in the human's real life, personality may be changing. For example, one's personality in childhood should be different from one's personality as an adult. However, the change of a human's personality will take a long time. In this study we consider a simple model, so we recognize personality as an initialized constant.

We consider in this study the following five axes of personality.

1. Calm and Hasty
2. Mild and Cruel
3. Timid and Brave
4. Neat and Naughty
5. Inattentive and Dedicated

It is likely that these five axes of personality are sufficient in order to create a simple emotional AI. However, we may add a new axis of personality if necessary in a target game. For example, we may add Docile and Stubborn axis in a team game which needs teamwork. Also, we may cut some unnecessary axes of personality in the case where the game considered does not need the axis of personality. It can be adjusted by the AI designer depending on the game. We should make the system as simple as we can but it should be sufficiently complex in order to create a realistic emotional AI.

Consider the above five axes of personality. Each agent will have these values in a scale from zero to ten. Zero indicates a personality on the left hand side, and ten indicates a personality on the right hand side. For example, agent A has a Calm-Hasty value of 3, Mild-Cruel value of 8, Timid-Brave value of 7, Neat-Naughty value of 1 and Inattentive-Dedicated value of 5. This means that agent A is calm, cruel, brave and very neat and in the middle for inattentive and dedicated.

However, there are some inconsistent combinations. An example is shown below.

- A Calm agent can be neat
- A Calm agent cannot be naughty
- A Hasty agent cannot be neat
- A Hasty agent can be naughty

To solve this problem, we add a dependency between the axes of personality. We limit the possible range of each personality axis based on other axes as described in Table 6.1.

The inconsistent combination mentioned above is symmetric. However, there are some inconsistent combinations which are non-symmetric.

- A Calm agent can be timid
- A Calm agent can be brave
- A Hasty agent cannot be timid
- A Hasty agent can be brave

Also, we can limit the possible range of each personality axis based on other axes as shown in Table 6.2.

In this study, we can limit the range of each rule. For example, in the example shown above, we use the range which equals four (0-3) for every rule. Moreover, we can add

TABLE 6.1: An example of possible range of symmetric inconsistent combination

Calm-Hasty	Neat-Naughty
0	$0-3$
1	$0-4$
2	$0-5$
3	$0-6$
4	$1-7$
5	$2-8$
6	$3-9$
7	$4-10$
8	$5-10$
9	$6-10$
10	$7-10$
$0-3$	0
$0-4$	1
$0-5$	2
$0-6$	3
$1-7$	4
$2-8$	5
$3-9$	6
$4-10$	7
$5-10$	8
$6-10$	9
$7-10$	10

more rules appropriately depending on the game that is considered and emotional AI design. An example of inconsistent combination rules is shown in Table 6.3.

By adding the dependency to these rules with a range of four, we obtain 49,254 consistent combinations (calculated by the program) from 161,051 combinations (calculated by five axes of personality and each axis consists of 11 possible values, from 0 to 11). We think that this is suitable for our simple simulated game.

Then, personality will affect an agent in many ways because every variable of the agent will be represented as a function of personality; $f(CalmHasty, MildCruel, TimidBrave, NeatNaughty, InattentiveDedicated)$. Also, personality directly affects the decision making.

6.3.1.2 Memory

Clocksin [24] explored the issues in memory and affect in connection with possible architectures for artificial cognition. Memory [11] [10] is a part for memo events in games. Memory consists of two types. One type can be simply described as a relationship between an agent and other players. This type of memory remembers what happened

Table 6.2: An example of possible range of non-symmetric inconsistent combinations

Calm-Hasty	Timid-Brave
0	0 – 10
1	0 – 10
2	0 – 10
3	0 – 10
4	1 – 10
5	2 – 10
6	3 – 10
7	4 – 10
8	5 – 10
9	6 – 10
10	7 – 10
0 – 3	0
0 – 4	1
0 – 5	2
0 – 6	3
0 – 7	4
0 – 8	5
0 – 9	6
0 – 10	7
0 – 10	8
0 – 10	9
0 – 10	10

Table 6.3: Example of inconsistent combination rules

Calm	can	Mild		Calm	can	Cruel
Hasty	cannot	Mild		Hasty	can	Cruel
Calm	can	Timid		Calm	can	Brave
Hasty	cannot	Timid		Hasty	can	Brave
Calm	can	Neat		Calm	cannot	Naughty
Hasty	cannot	Neat		Hasty	can	Naughty
Mild	can	Neat		Mild	cannot	Naughty
Cruel	can	Neat		Cruel	can	Naughty
Timid	can	Neat		Timid	cannot	Naughty
Brave	can	Neat		Brave	can	Naughty
Neat	can	Inattentive		Neat	can	Dedicated
Naughty	can	Inattentive		Naughty	cannot	Dedicated

between players. Another type is the memory relating to an agent itself. This second type of memory involves remembering an event which is not related to the other players.

For example, there are four players, say Players A, B, C and D. It is assumed that Player A has four memories which consist of the memory from A to B, memory from A to C, memory from A to D and the memory from A to itself. So, for n players in a given game, each player will have n memories: $n-1$ memories for the relationship between an agent and other players and 1 memory for itself.

To implement this, memory is a list of events. Each event has the hate value (+ or -). For example, if agent A kills agent B, agent A will have a 'getKill' event (hate value = - which means that agent A does not hate agent B) in its memory for agent B. Agent B will have event 'isKilled' event (hate values = + which means that agent B hates agent A) in his memory for agent A. However, the event will be deleted as time passes. This is the same as a human's memory, when time passes, we forget some events.

However, memory may keep good events (hate value = -). For example, you feel good about someone when the person helps you. We try to implement this on our test game. Suppose that agent B tries to kill agent A. However, agent C kills agent B, so agent A survives. In this case, agent A will feel good about agent C. We can apply this idea to the memory model in another game which is more complex.

Moreover, for some cruel agents, when one receives the same event from the same dealer, it will recognize an old memory. It means that it can recall the memory. For example, agent A kills agent B. Agent B remembers that agent A killed it. Simply, B hates A. As the time passes, the hate value from B to A will decrease. But this event will not be deleted. It will be remembered (with hate =0). When agent B is killed by agent A again, he will remember that A has killed it before. So this is the second time it kills agent B. Agent B will hate agent A more than before. This idea might be same as in a human. Occasionally, we can recall an old memory when we meet something similar to our memory.

Memory is affected by personality. For example, the hasty agent and dedicated agent have a high hate value while it is hard for cruel agents to forget a hate event. Memory affects the decision making, i.e., whether to decide if the agent kills that player.

6.3.1.3 Mood

Mood is the current feeling of an agent. In contrast with personality, as time passes, mood is changing [140]. It is observed in [106] saying that "Human emotion includes basic emotion and multiple emotions. Basic emotion is a basic element in human emotion while multiple emotions are complicated and changeable". Also, current mood can be represented as a linear combination of basic moods, as shown in Equation (6.1).

$$Mo_i(t) = \sum_{j=1}^{l} k_j e_j(t) \tag{6.1}$$

Where

- $Mo_i(t)$ is the current mood of $agent_i$ at time t
- l is the number of basic moods
- e_j is the basic mood
- k_j is the affective coefficient of e_j

To simplify this, in this study we focus on four basic moods:

1. Joy
2. Anger
3. Boredom
4. Fear

The mood range is given as $[0 - 100]$. This is because a negative mood value does not have meaning such as what the meaning of anger $= -50$ is. However, mood will affect an agent when its value is larger than a threshold. We think that this idea is the same as for a real human. Sometimes, we can keep our feeling to ourselves so it does not have an effect on our action. For example, for a given threshold $= 20$, mood $[0, 20)$ does not affect the agent, but mood $[20, 100]$ may affect it. In this way, a mood threshold can be adjusted in an "Ad Hoc" way depending on the game considered and the AI design.

Mood is affected by personality. For example, a hasty agent easily becomes angry, it will be hard for a brave agent to feel fear and an inattentive agent easily becomes bored. Mood affects memory. For example, you feel bad and you experience a bad event, even though that event is just a little event, because you are feeling bad, it has more effect. Also, memory affects mood because mood can be quantified by events kept in memory.

Mood affects the output of an agent such as decision making. For example, agents which feel fear would try to run away. Angry agents would move fast and try to kill other players. Bored agents would move slowly and not care about the game. However, it depends on personality.

By combining these meaningful factors, we can construct an emotional AI model as a mathematical equation as shown in Equation (6.2).

$$O_i(t) = f(RW(t),\ P_i,\ Me_i(t),\ Mo_i(t)) \tag{6.2}$$

Where

- $O_i(t)$ is the output of $agent_i$ at time t
- f is the emotional AI decision making function
- $RW(t)$ is the real world situation at time t
- P_i is the personality of $agent_i$
- $Me_i(t)$ is the memory of $agent_i$ at time t
- $Mo_i(t)$ is the mood of $agent_i$ at time t

6.3.2 Realistic enhancement

Although we discussed some aspects of an emotional component, there are many remaining issues which are essential. In order to make agents more realistic, we should consider these issues to implement on emotional AI system. These issues are affected by personality, memory and mood, and also they affect memory, mood and the output directly.

6.3.2.1 Physical state

The physical state is the current state of an agent. For example, the remaining HP of the agent is the physical state in a fighting game. It affects the agent's mood. If the agent is nearly dead, the agent may feel fear and try to run away. However, it depends on the agent's personality. A brave agent would not fear. The physical state directly affects the agent's mood and decision making.

There are many physical states such as the remaining time in time-limited games or remaining scores in score-limited games, the current ranking of an agent in score games and so on. We can apply this idea to our emotional AI system.

6.3.2.2 Human-likeness

Some minor details should be added for an agent to be more human-like. There are many aspects in the topic of human-likeness [78] and a tangible example is the marginal utility issue. The marginal utility comes from economics. The definition of the marginal utility [71] is that the same item may have a different utility for the same person at a different time. For example, the marginal utility of income declines as income increases. In another example, imagine that you eat a hot dog. The first hot dog will give you a lot of satisfaction. The second one will not give you as much satisfaction as the first one. Also, the third one will give less satisfaction.

To implement the marginal utility concept in game AI, we can apply this idea to the resource component which usually occurs in every game such as money and gold in Multiplayer On-line Battle Arena (MOBA), the bullet remaining in shooting games and petrol remaining in racing games.

Besides the marginal utility, there are many minor details which determine where a human usually does in each game. We can recognize it and implement it in our agent to make it more human-like.

6.3.2.3 Intention

Some actions occur because we intend to do while some actions occur without our intention. These can be described below.

1. Intentional Action

   A human has intentions. For example, we want to eat food so we may go to the supermarket and buy some fast foods. When we completely finish eating them, we feel good as expected. Also, an agent has intention. Simply, as shown in Figure 6.4, if it can accomplish its intention successfully, it feels good. If not, it feels bad. Importantly, the hurt resulting from failure depends on its attempt.

2. Unintentional Action

   Occasionally some events occur without intention. For example, we can consider human's unconscious automatic reactions, or some mistakes such as agent A unintentionally killing agent B. If agent A is mild, its mood would dramatically decrease because it unintentionally killed agent B. If agent A is not mild, it does not care. We can handle unintentional action by recognizing the game's possible unintentional action and apply it to AI.

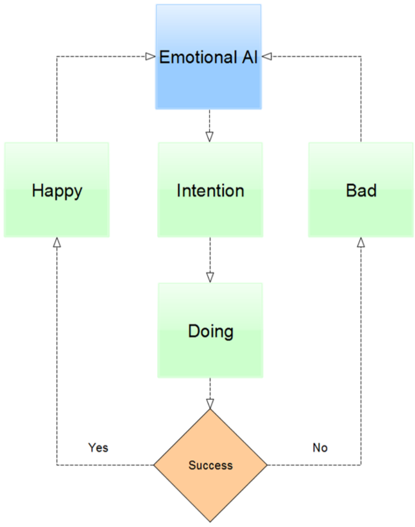

Figure 6.4: Diagram of intention in the Emotional AI framework

6.4 Evaluation Experiment

In this section, we first design the test which answers the research question. Next, we introduce our simulated game and present its implementation of emotional AI. Finally, the evaluation experiment is performed and the results are shown.

Our generic model of emotional AI is just a concept idea. To evaluate its effectiveness, we have to implement it as a concrete system which is measurable. Therefore, we create a suitable game and implement our model as a player in the game. Then, we can see how our emotional AI works by playing with it or watching its action. However, the topic of emotional AI is very subjective and abstract, so it is quite hard to verify the efficiency of the model that we proposed by using computer or mathematical analysis. Therefore, in order to test whether humans can understand our emotional AI correctly or not, then, the evaluation will be performed with human subjects [6].

As mentioned in Section 6.3, an emotional component in our emotional AI model consists of three main factors: personality, memory and mood. However, as we have explained, personality is the major factor which directly affects the other factors. Personality limits some actions, i.e., a timid agent would not disturb another player, and moreover, personality is the only factor which does not changed. Therefore, the evaluation test will be conducted while focusing on personality. Each tester has to play with emotional AI, and then answer what is the personality of AI. Furthermore, in order to verify some assumptions, we conduct this experiment with many rounds by changing some factors which we will explain later.

6.4.1 Test game

For the assessment of our proposed emotional AI model in this study, we create a multiplayer real-time shooting game as a benchmark. In this game, one can move on four directions by using an arrow keyboard and shoot the bullet by using the space bar button on the keyboard. The goal of the game is to get as many scores as possible. The player who achieve the best score will unquestionably be a winner. To get scores, a player has to destroy given point units. This means that the player does not need to kill other players, but the player has to focus on the point unit and try to collect scores as hard as he can. However, we see that even though the player does not need to kill other players, the player still may do so.

That is our main idea of this test game, which we have proposed in Section 6.2.1: even though one does not need to carry out such action as killing, one might do so. That is why we call it 'emotion'.

Below, we describe the rules of our test game.

- This game is a single player mode game. There is no team in this game.
- The goal is to get as high as possible scores in a limited time.
- In order to get scores, a player has to destroy given point units.
- A point unit has seven degrees of hit points (HP). One bullet will reduce one HP.
- A Player who gets a last shot, a shot that reduces a point unit's HP from one to zero, of point unit will receive a score.
- The game has the bullet system, which means that the number of bullets is limited. A player can reload a bullet by getting a bullet item.
- A bullet has five slots range.
- The game has initially some basic obstacles (walls) that the bullet and player cannot pass.
- A player can kill other players by shooting.
- A player has no HP. If a player touches the bullet, the player will die. When a player dies, the player has to wait five seconds before being reborn.
- An agent can display simple emoticons.
- An agent's mood is shown on the mood bar.

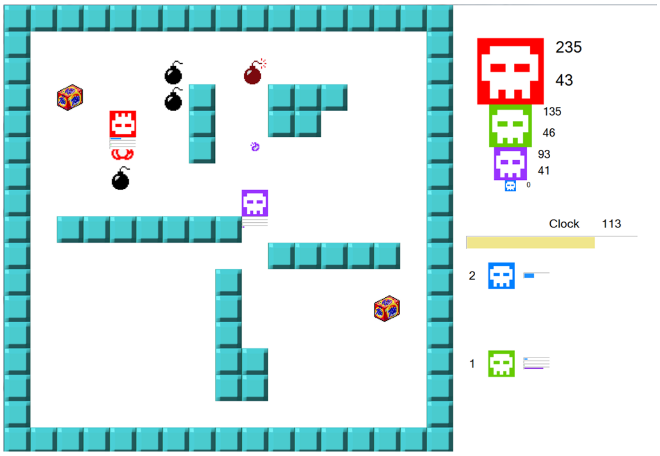

FIGURE 6.5: A screenshot of our test game

- A remaining bullet is shown on the bullet bar.
- The remaining time is shown on the clock bar.
- The current score is shown on the score bar.
- The respawn position is on the top left.

Also, we show a screenshot of the test game in Figure 6.5.

6.4.2 Implementation of emotional AI

By implementing our proposed emotional AI in the simulated game, there are many minor issues that are added, and some examples are described below.

- Keeping the bullet item: The marginal utility topic can be applied in keeping the bullet item which is related to the remaining bullet. If an agent has only a few bullets, the bullet items utility is quite high. If the agent has many bullets, the bullet items utility is quite low. Then, the utility that is received will affect the agent's mood.
- Check rank event: Sometimes, an agent can check its rank by looking at the leadership board at the top right of the screen. If the agent gets a good rank, it simply feels good, its joy would increase and its anger would decrease. If the agent gets a bad rank, it would simply feel bad, its joy would decrease and its anger would increase. In addition, the agent checks its rank every times it dies. This may be the same as a human player, when our character dies we have a short free time to check some details such as the score.

- Remaining time: An agent can check the time remaining on the clock bar. So, if the remaining time is short, the agent may walk fast and mood has a high effect because it is in a hurry. This means that the remaining time affects the agent. This is similar to a human's feeling, when the game is near the end, the human feels in a hurry. However, the detail depends on its personality. An inattentive agent and a calm agent would not care about this point.

- Shooting point unit's speed: When shooting a point unit, if the point unit is nearly dead, the shooting is slow in order to save bullet. However, the detail depends on personality. A hasty agent would not care about this.

- Shooting range: A brave agent would shoot at a short range, while a timid agent would shoot at a long range.

- Failures hurt depending on the attempt: If an agent spends many bullets shooting to this point unit but it was stolen, its mood would considerably decrease. If the agent walks to its target from a position far away but it was stolen, its mood would also considerably decrease. In these examples, we can use the number of bullets used and the number of steps walked as the attempts and we can calculate how much failures hurt, which affects mood and memory based on attempts and personality. For example, a mild agent would not be concerned about this point while a cruel agent would be concerned about it. In addition, a naughty agent would want to steal the last shot of point unit so the fewer bullets it used, the more happiness it experiences. The idea of intention can be applied to another game. Its result directly affects mood and memory

6.4.3 Experimental design

From the simulated game described above, we implement our emotional AI on this game and perform an evaluation experiment. Fourteen players participated as subjects in this experiment.

For each round, a player has to play with the emotional AI in this game. Next, the player is requested to answer the following question: What is the type of personality of each agent? The type of personality in this context means that a player has to provide an answer for all of the five the axes of personality. Then, the results will be calculated using the absolute error which is the absolute difference between the player's answer and the correct one. An example of how to calculate the error is shown in Table 6.4.

As explained in Section 6.4.1, an agent can use a simple emoticon and the agent's mood is shown on the mood bar. We call these functions as 'show emotion' function. However,

TABLE 6.4: Example of how to calculate errors

	Calm-Hasty	Mild-Cruel	Timid-Brave	Neat-Naughty	Inattentive-Dedicated
Agent A	3	8	7	1	5
Answer	1	6	3	0	5
Error	2	2	4	1	0
Average Error	1.8				

we wonder whether the shown emotion function helps the player to easily understand the agent's emotion.

Therefore, we perform this experiment both with turned on and turned off 'show emotion' function. Turning on the show emotion function means that a player can see the agent's mood bar and the agent can use an emoticon. Turning off the show emotion function means that a player cannot see the agent's mood bar and the agent cannot use an emoticon.

In addition, we wonder whether the number of agents playing with a tester affects the results. So, we use a different number of agents in different rounds. Furthermore, we wonder while playing with agents, it is hard to recognize the agent whether or not. So, we have a replay function which means in some rounds the player does not need to play but the player has to watch the replay and provide an answer about the agent's personality.

The detail of each round is described here. During the first round, a tester will play with one agent by turning off the show emotion function. During the second round, the situation is same as in first round, but the show emotion function is turned on. During the third round, the tester will play with three agents by turning off the show emotion function. During the fourth round, the tester does not play but he/she has to watch the third round's replay. The situation during the fifth and sixth rounds is also the same as in the third and fourth rounds respectively, but emotion function is turned on. We think that if there is one agent, it is easy to recognize its personality, so we set the time in the first and second rounds to 75 seconds. For the third round to sixth rounds, since there are three agents, we set the time to 150 seconds. The test is described in Table 6.5.

In this study, fourteen subjects participated in the test experiment and played six rounds.

6.4.4 Results

The results of the experimental test mentioned above are shown in Table 6.6 and Table 6.7.

Table 6.5: Detail of each round

Round	Human	Agent	Show Emotion	Replay	Time
1	1	1	Off	-	75
2	1	1	On	-	75
3	1	3	Off	-	150
4				Watch	
5	1	3	On	-	150
6				Watch	

Table 6.6: Test results for each error

Error	Frequency	Percent
0	140	14.29%
1	204	20.82%
2	199	20.31%
3	140	14.29%
4	74	7.55%
5	89	9.08%
6	40	4.08%
7	45	4.59%
8	33	3.37%
9	11	1.12%
10	5	0.51%

Table 6.7: Test results for each tester

Based Error	3.63
Best Error	2.21
Average Error	2.79
Worst Error	3.64

In Table 6.6, the total sum of frequencies is 980, which is calculated from 14 testers, 14 AIs (rounds 3-6 have 3 AIs per round) and 5 axes of personality. We see that usually the error is around 0-3 (69.71%).

In Table 6.7, the based error is calculated by fully random error, an error which is the average of all possibility error, which is equal to 3.63. The best error is 2.21, which means that it is better than a random 39.10%. The average error is 2.79, which shows that it is better than a random 23.36%. The worst error is 3.64, which means that it is worse than a random 0.18%.

Note that in rounds 1-2 we have one agent in each round and then in rounds 3-6 we have three agents in each round. So, the results for rounds 3-6 are the average error of three agents.

From Table 6.8, we see that, as we have wondered, turning on the show emotion function is better than when turning it off as we can see for rounds 1,2, rounds 3,5 and rounds

Table 6.8: Average error of each round

Round	Average Error
1	3.29
2	2.74
3	2.86
4	2.79
5	2.70
6	2.64
Average	2.79

Table 6.9: Average error for each personality

	Calm-Hasty	Mild-Cruel	Timid-Brave	Neat-Naughty	Inattentive-Dedicated
Round1	2.14	2.86	4.14	3.71	3.57
Round2	2.07	2.29	4.29	2.29	2.79
Round3	2.67	1.81	3.86	2.83	3.14
Round4	2.74	1.67	3.79	2.81	2.95
Round5	2.74	2.60	2.76	2.33	3.10
Round6	2.95	2.21	2.17	2.36	3.50
Average	2.68	2.14	3.30	2.64	3.17

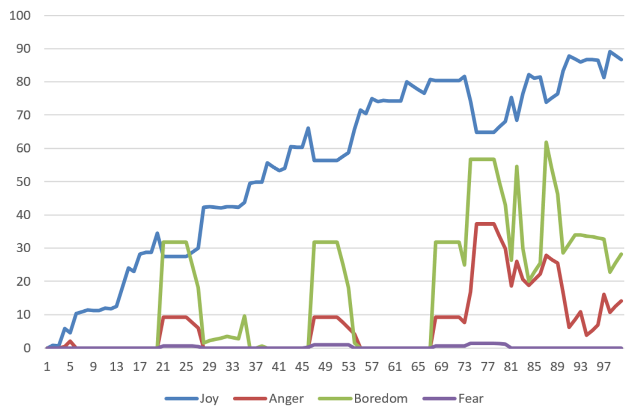

Figure 6.6: An example of the change of emotional AI's mood

4,6. Moreover, watching a replay is better than playing as we can see for rounds 3,4 and rounds 5,6. However, a high number of agents does not make the error worse, as we can see for rounds 1,3 and rounds 2,5. This is because we increase the time in rounds 3-6, which have three agents, from 75 seconds to 150 seconds. Moreover, the increase of the number of agents leads agents to fight against and interact with each other so agents will show emotions in this way.

According to Table 6.9, Mild-Cruel is the easiest to detect personality. Also, Neat-Naughty and Calm-Hasty are easy to detect. However, Inattentive-Dedicated and Timid-Brave are quite hard to detect.

Figure 6.6 shows a real-time mood change graph which was constructed from one of our experiments. This agent has a Calm-Hasty value of 6, Mild-Cruel value of 7, Timid-Brave value of 8, Neat-Naughty value of 9 and an Inattentive-Dedicated value of 1. We will see that, when joy is increased because of some positive events such as agent can collect scores, anger and boredom decrease simultaneously. Also, when joy decrease because of some negative events such as when the agent is killed, anger and boredom increase simultaneously. However, it is easier for boredom to increase because this agent is very inattentive. Moreover, it is very hard for fear to increase because this agent is very brave.

6.5 Discussion

The results presented in Section 6.4.4 are discussed in this section.

6.5.1 Test game

We see that the test game is too simple to show emotion because the output channels of this game are quite limited. The decision-making choice in this game can be simply concluded in three main choices which consist of keeping the bullet item, destroying point units and killing other players. Even though our model is complex enough to express human player's emotions realistically, there is no channel to express emotion in this simple game, so AI cannot express emotion at the output with the same level of complexity as it has inside the system. We can solve this problem by creating a new game which is more suitable and implement our model in that game.

Furthermore, sometimes, we cannot separate the cases between shooting by emotion and shooting by strategy. For example, to win the game, a player needs to collect as many score as possible, so sometimes players have to kill other players to steal the others' scores even though the player does not hate them. So, it is hard to explain where this shot comes from.

6.5.2 Emotional AI model

Basically, pure AI optimizes the answer as much as it can so there will be only one answer or a little set of possible choices. In contrast, adding emotion increases the number of possible choices. We believe that this point can improve the game's attractiveness because it creates various player experiences so a player can feel and enjoy a game.

FIGURE 6.7: Example of an agent's mood

However, in this model, there is no relationship between moods. For example, occasionally, an agent can feel fully bored and fearful such as the green agent in Figure 6.7. Even though a real human can experience many moods on one time, for instance, the purple agent feels joy and fear and a little anger, which means that he feels challenged and excited, but, to reach a maximum mood, there should be only one. To solve this problem, the mood variable should include a dependency between different moods. When one mood changes, it should affect the other moods.

Apart from this, the scalability of the memory component is one of this model's problems. According to Section 6.3.1.2, we have to use n memories for one agent in a n players game. So, in the case where we have numerous agents, we have to use a lot of memories. We can solve this problem by using the group relationship idea. For example, an agent can remember some events as a group so it can save memory.

In addition, some axes of personality are quite similar. For example, Calm-Hasty and Timid-Brave, Calm-Hasty and Neat-Naughty, Mild-Cruel and Neat-Naughty come from tester's comments. Actually, some minor details are different but it needs times to see and it is quite hard to separate. However, a real human is also complex.

Anyway, considering the inattentive agent, sometimes we cannot see other personalities in it because it is inattentive so it does not do anything. So, for the tester, it is hard to recognize its personality.

6.5.3 Assessment

From the tester's point-of-view, it is quite difficult to give a score when there is no reference. This is because testers have their own feelings. However, we actually want to provide the reference of volume of each axis of personality. But we should not give the reference of the type of action because the tester should understand it by himself/herself. For example, we will not tell the tester that a hasty agent will be easy to anger, and an angry agent will move fast and try to kill other players. This is a very simple action and we expect that the tester can guess and understand it by himself/herself.

By the way, there should be an agent without emotion for comparison with an agent with emotion. In this test, there is no normal agent, an agent without emotion, so it is hard to understand the emotion of an emotional agent when we do not have an agent without emotion for comparison. Nevertheless, the test conducted here tests only the personality factor. Actually, the emotional component consists of three main factors and many minor details. So, it would be better if we can test every factor in our model. However, it is hard to design a test which can test every factor.

Additionally, the test time is quite short. In fact, to understand one human being, we need a long time to learn and understand his/her personality. Also, even though our emotional AI is not complex, it needs more time in order to understand its personality.

Moreover, in this test, randomly generating an agent for each round is somehow not balanced for the tester. Sometimes, the combination of personalities of agents is easy to understand while sometimes that combination of personality of agent is hard to understand. However, if we fix the correct answer of the test constantly, that means we will test just only the combinations of personalities that we fix. So, randomly generating an agent and constantly generating an agent have their own weaknesses and strengths.

6.6 Conclusion

As we have explained, AI is one of the most important factors which directly affect the player experience. The evolution of AI technology means the evolution of making a new player experience. In this study, we presented a generic model for emotional AI in real-time multiplayer fighting games. We believe that the emotional component can be a significant way to improve the quality of AI for exploring a new player experience. We introduced many details about emotional AI and finally proposed an emotional component which consists of personality, memory and mood. Also, we proposed some enhancements which make the proposed emotional AI more realistic. Then, we created a simple game for implementing this AI. Next, the assessment test was conducted and the results obtained were discussed.

We proposed our definition of emotional AI and its constraints in Section 6.2. Then, we concluded our emotional AI model as shown in Equation (6.2) and its agent structure as shown in Figure (6.3). We conclude that our emotional AI implemented in a simple shooting game works well, as the results show that the average error is 2.79, which means that it is better than random 23.36%, and more details were presented in Section 6.4.4. These results answer the research question whether a real human player can understand the emotion of emotional AI correctly.

While there are many emotional AI models which focus on different points-of-views as we have mentioned before, this work is an extension of those works. We consider many important parts and try to simplify them as much as possible. Finally, we proposed our new emotional AI model. The contribution of this work is that our emotional AI can be a simple generic emotional AI model in the domain of real-time multiplayer fighting games. It can be used as a basis model to create a new emotional AI model in future works such as in another game, in another domain and so on. Also, we confirm that it can be applied in another game in this domain by adjusting some factors.

In conclusion, it is understood that the work presented here is a simple model, and more studies are required. Further works may include various approaches such as implementing emotional AI in a new simulated game. In addition, we can develop a new emotional AI model in another domain of games such as the turn-based game, the real-time strategy (RTS) or role-playing game (RPG). Furthermore, we can consider a new domain of emotional AI such as teammate emotional AI because this emotional AI model is the opponent emotional AI model.

Chapter 7

Quantifying Enjoyment and Game Progress Patterns

This chapter is an updated and abridged version of work previously published in

(13) Chetprayoon Panumate and Hiroyuki Iida. Quantifying Enjoyment of Individual Match in Games. 2016 The Annual Conference on Engineering and Applied Science (ACEAT 2016). in press.

This chapter present the last approach to improve the entertainment impact of games. Previously, game refinement theory [128] measures enjoyment of many games in various domains of game [150]. It finally can propose the appropriate range of game refinement value which sophisticated game should have. While game refinement theory is focusing on how to measures attractiveness between games, this research focuses on how to measure the attractiveness between matches in the same game.

If we can figure out a reasonable model to compare the enjoyment between matches, it means that we can truly understand about the enjoyment in game, at least more than before. So, we can improve the entertainment impact of the game which is our main thesis's topic by using this idea as a basis.

7.1 Introduction

Enjoyment is perhaps the most important issue in successful game design [63]. Nowadays, there are so many video games in game industry and it is implemented in many platforms such as mobile, personal computer, game console and so on. Some of them are so popular but others are not so. There are many reasons behind this fact such as its story, animation and graphics, setting of the gameplay and so on [67] [123].

Beside video games, there are other games which can be considered as a game such as sports and board game. With the same consideration, some sports and some board games are so popular while others are not so. Even in the same game, some matches are exciting but others are not so. For example, why is this football match so exciting and why do many people want to watch this match? If we clearly understand this point, it means that we can truly understand the enjoyment in games [68], at least more than before. So, we can improve the entertainment impact of the game by using this idea as a basis.

Therefore, the aim of this research is to establish a generic model for quantifying enjoyment of individual match in a game. To tackle this challenge, we propose a reasonable model by focusing on uncertainty of the game outcome. We believe that if the outcome of the game is unknown, it means that the game is exciting and able to keep attractive [69] [120].

Additionally, this model considers from viewer's point of view not from player's point of view. This is because a player basically prefers his team to be a winner. As a player, victory is the most important thing. Therefore, considering from the player's point of view has bias. In contrast, considering from the viewer should have lower bias. We cannot avoid the truth that human always has bias but choosing to consider from the viewer's point of view can reduce bias in this issue.

Firstly, we focus on the concept of outcome uncertainty. Principally, uncertainty of game outcome [80] was investigated by information dynamic principal [55] and was applied in many games. Generally, if the outcome of game is uncertainty, that means the game is exciting and it still be attractive. With the same idea, if the outcome of game is known, it will be boring. Therefore, we can measure the enjoyment of each match by comparing the advantage of each team in every time. That means we have to construct a graph of advantage and we can quantify the enjoyment by comparing the graph of advantage.

Definition 3. **Graph of advantage** is a graph that show the advantage between teams in time t.

The point is 'how to measure the advantage of each team?'. Basically, It is obvious that we can easily look from score such as in popular sport likes football, basketball and volleyball. The more score team gets simply means the more advantage team has. However, for some academic games like chess and go, it is too complicated to evaluate the advantage of each team by looking from current score. Moreover, for some digital match games which have many complex factors like MOBA (Multiplayer online battle arena), RTS (Real-time strategy) or some complex FTS (First person shooter) game, we surely cannot. Therefore, the one point we should concern is the question: how to construct the graph of advantage.

Next, suppose that we can create a reasonable graph of advantage, the second question is that how to compare these graph. Mostly, if a graph of this match is complicated as mentioned above, it means that the outcome of this match is uncertainty and this match should be exciting. However, there are many kinds of complicated graph which lead the game so exciting. Therefore, this research propose a reasonable model to quantify the enjoyment value from the graph of advantage.

However, it doesn't exactly mean that this match is truly fun. For example, if a match is approximate, team A and team B are both good, but there is no score in the end of game or the game's progress is very slow such as a football match which is no attempting to shoot and the result is 0-0. It absolutely means it is not fun as well, may be very boring. Therefore, besides outcome uncertainty component, the speed of game's progress directly affects to fun. If it has a lot of exciting shots in a game such as a shot of attempting to shoot in football or a shot of attempting to kill opponent, it means that match is attractive. So the second component is exciting shot component.

The third component is special technique component [60]. If players always use the same tactic, it may be boring. So, this component will focus on the technique which is used in that match considered. The final component is individual fondness component. Sometimes, some viewers like some teams or some players without reason. This is because sometime viewer have individual fondness and it also affects to the enjoyment of each match.

Therefore, our model is composed of the following components.

1. Uncertainty of outcome
2. Exciting shot
3. Special technique
4. Individual fondness

By combining these meaningful components, we can establish a general model for quantifying enjoyment between matches in the same game. Then, Defend of the ancient 2 (DotA 2), the new episode of the most well-known MOBA games, DotA [30], is used as a testbed of this model. DotA 2 is very complex so it is hard to propose advantage graph, detect exciting shots and define special techniques used. So, if our model can be applied to a complex game like DotA 2, it surely means that it can be applied to a simple game by reasonably simplifying some factors. Moreover, DotA 2 is very popular so we have enough information for analyzing some important knowledge used in equations. Therefore, we choose DotA 2 as a testbed of this research.

Additionally, table tennis is chosen as a testbed for the simple game. This is because we can obviously see the graph of advantage of table tennis. Therefore, in this research, we implement our models in two games consist of DotA 2 and table tennis.

The structure of this chapter is as follows. We first focus on the uncertainty outcome component which is our main target issue in Section 7.2. Next, in Section 7.3, we verify our model's efficient by performing an experiments with real data. Finally, the results obtained are discussed in Section 7.4, and concluding remarks are given in Section 7.5.

7.2 Uncertainty of outcome

[57] explained that games consist of the three elemental game progress patterns shown below.

1. Balanced game: Both of the teams have no goal through the game.
2. Seesaw game: One team leads, then the other team leads, and this may be repeatedly alternate.
3. One-sided game: The current goal sum of one team(winner) is always greater than that of the other team(loser), so that the goal difference between the two teams is always positive.

From the uncertainty of outcome's point of view, we expect that the seesaw game is the most exciting game because the outcome of the game is uncertain. For balanced game, the game does not have any progress so the game is boring. For one-sided game, as it is explained that there is separated between winner and loser obviously so the outcome of the game is not uncertain. Therefore, it is obvious that seesaw game is the most exciting game.

As it is explained in the previous section that our model has four components, in this study, we explain only the first component which is our main focus. Therefore, in this section, we explain two steps to obtain the uncertainty outcome value from one match of the game shown below.

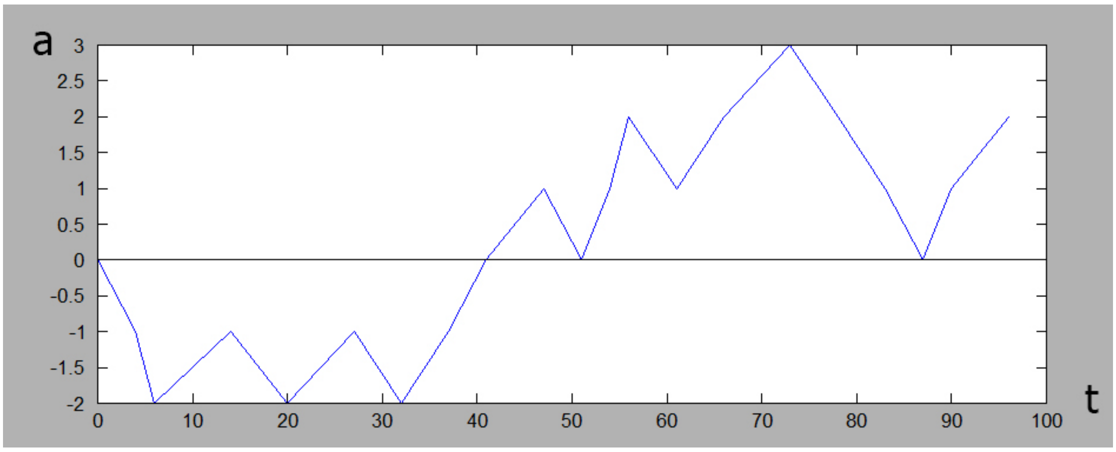

FIGURE 7.1: An example of graph of advantage in Tennis from Table 7.1

1. How to construct a graph of advantage
2. How to quantify the enjoyment value by using the graph of advantage

7.2.1 How to construct a graph of advantage

[56] proposed safety lead curve, the curve that once information of the game outcome goes above this curve, the advantageous team will win the game with 100% certainty. This idea is similar to graph of an advantage idea.

From the definition of graph of advantage, we can construct this graph by using the game progress viewer receive. As it was explained that this model focus on the enjoyment from viewer's aspect. Therefore, the game progress in this context means the information that the viewer can receive. To implement the graph of advantage, we use y-axis as an advantage and x-axis as time.

Actually, to construct the graph of advantage, we have many ways depends on the target game considered. In this section, we show two examples from Table tennis and DotA 2. Table tennis is a representative of a simple game which we can directly use score. Respectively, DotA 2 is a representative of a complex game that we cannot use score directly.

7.2.1.1 Table tennis

Table 7.1 shows the raw score from one match of Table tennis. Let t stands for the time in second. Δt will be the difference of the time that one player can make a score.

From this Table 7.1, we can construct a graph of advantage as shown in Figure (7.1).

Table 7.1: An example of score in table tennis

t (s)	Δt (s)	Player1's score	Player2's score
0	0	0	0
4	4	0	1
6	2	0	2
14	8	1	2
20	6	1	3
27	7	2	3
32	5	2	4
37	5	3	4
41	4	4	4
47	6	5	4
51	4	5	5
54	3	6	5
56	2	7	5
61	5	7	6
66	5	8	6
73	7	9	6
78	5	9	7
83	5	9	8
87	4	9	9
90	3	10	9
96	6	11	9

7.2.1.2 DotA 2

DotA 2 [137] is a multiplayer online battle arena (MOBA) video game developed and published by Valve Corporation. DotA 2 is played as a match game. The match is between two teams and each team consists of five players. When the game starts, each player can choose one from 111 playable characters, called "heroes". During a match, a player collects gold, items, and experience points for making their hero strong. Each hero has unique skills and abilities which make player can create his own playing style.

Due to DotA 2's complexity, we cannot directly use score as table tennis or some simple sports can. Actually, DotA 2 does not have score because the purpose of this game is to defend our base and attack opponent's base. The base can be called 'ancient'. That is why the name of this game is DotA, defense of the ancients. However, we cannot use the HP of the ancient as a score because usually when players have a chance to attack ancient, they can destroy it in only one minute or less than.

The proposed idea in this study is very simple. Normally, in DotA 2 tournament, we use these two things to estimate that which team has an advantage.

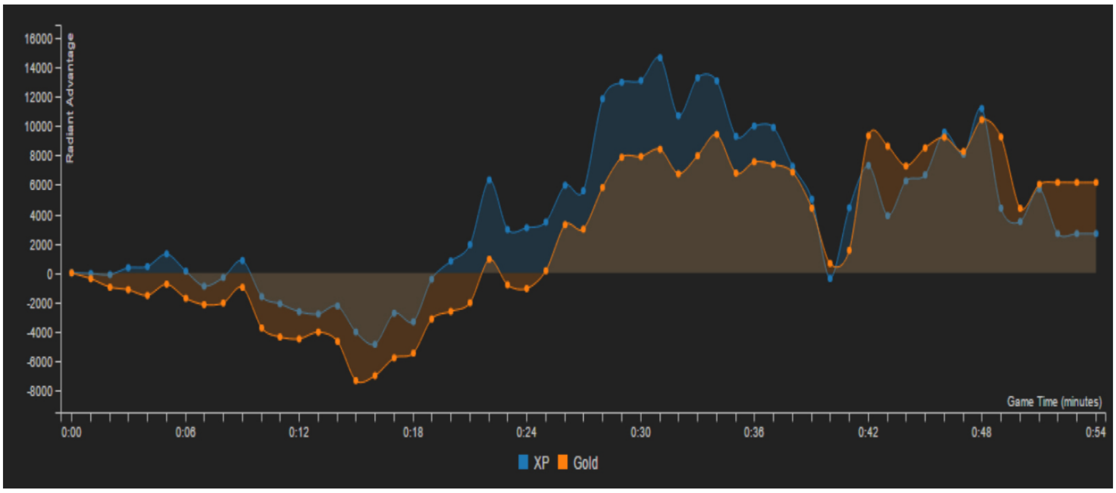

FIGURE 7.2: An example of graph of advantage in DotA 2

1. Gold: the summation of gold in each team. If the player has a lot of gold, he can purchase some expensive items that make his hero strong.
2. Experience: the summation of experience points in each team. Experience points represent the level of hero. If hero has high level, they have strong skills and its status is better.

So, in this study, we use these two factors as a representative of advantage. Figure (7.2) shows an example of the graph of advantage by using gold and experience. Note that XP stands for experience points. This graph was created from [152].

7.2.2 How to quantify the uncertainty of outcome value by using the graph of advantage

From the previous section, suppose that we can create a reasonable graph of advantage, this section explains how to reasonably figure out the uncertainty of outcome value, can be called OC, from the graph of advantage.

According to Figure 7.3, it shows the three possible zones in the graph of advantage. The main idea of this model is that the more seesaw the game is, the more exciting the game is. We can quantify the uncertain of the game from the change of game progress. The game progress is this section is denoted by the information that the viewer can receive. It was explained in the previous section about the graph of advantage.

So, the uncertainty of outcome value, OC, should be simply delivered from the summation of $differnce\_of\_degree$ as shown in Equation (7.1).

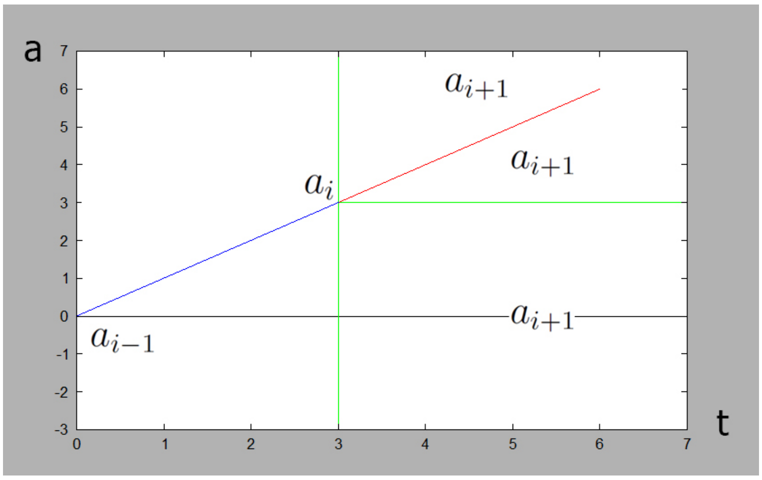

FIGURE 7.3: Three possible zones in the graph of advantage

$$OC = \sum^{point\ i} (differnce\_of\_degree) \tag{7.1}$$

However, the long game does not mean the exciting game because some long games are one-sided game as it was explained. Therefore, OC from Equation (7.1) should be divided by $number\_of\_point$ as shown in Equation (7.2).

$$OC = \frac{\sum^{point\ i} (differnce\_of\_degree)}{number\_of\_point} \tag{7.2}$$

Next, we found that the $differnce\_of\_degree$ has many kinds. For example, the $differnce\_of\_degree$ that make the game be a one-sided game and the $differnce\_of\_degree$ that make the game be a seesaw game respectively. In the first case, the $differnce\_of\_degree$ should be minus because it makes the game to be boring. For second case, the $differnce\_of\_degree$ should be plus because it makes the game to be exciting.

Moreover, using the degree directly may lead the OC value to be strange. Therefore, we purpose to add the constant for multiplying to the $differnce\_of\_degree$ as shown in Equation (7.3).

$$OC = \frac{\sum^{point\ i} (zone\_constant \times differnce\_of\_degree)}{number\_of\_point} \tag{7.3}$$

Actually, the $differnce\_of\_degree$ can be represented by the Equation (7.4) and $zone\_constant$ can can be represented by the Equation (7.5) respectively.

$$difference\_of\_degree = \left|\arctan(\frac{a_i - a_{i-1}}{dt_i}) - \arctan(\frac{a_{i+1} - a_i}{dt_{i+1}})\right| \quad (7.4)$$

$$zone\_constant = \lambda_j \quad (7.5)$$

Where

- OC is the outcome uncertainty value. It shows the enjoyment of this match by using outcome uncertainty component.
- i represent the considered point at time t in the graph.
- n is the number of point i.
- a_i the advantage at the considered point i at time t.
- $m_{(i-1,i)}$ is the slope delivered from a_{i-1} to a_i.
- j represent the name of zone.
- λ_j is the constant for zone j.

Nevertheless, to use the model, we can calculate only the point a_i which is in the middle of the graph. In any graph of advantage, we have n points. So, by cutting the points in the edge of the graph, we can use only $n-2$ points. Next, by substituting Equation (7.4) and Equation (7.5), we can obtain Equation (7.6).

$$OC = \frac{1}{n-2}\sum_{i=2}^{i=n-1} \lambda_j \left|\arctan(\frac{a_i - a_{i-1}}{dt_i}) - \arctan(\frac{a_{i+1} - a_i}{dt_{i+1}})\right| \quad (7.6)$$

To summarize, Equation (7.6) can be written in Equation (7.7)

$$OC = \frac{1}{n-2}\sum_{i=2}^{i=n-1} \lambda_j \left|\Delta\theta_{(l_{(i-1,i)},l_{(i,i+1)})}\right| \quad (7.7)$$

For the constant λ_j, we first separate the type of j in many cases depends on many factors such as a_{i-1}, a_i and a_{i+1}. From the proposed model, our assumption is that if the game is balanced, which means the advantage is around zero, it means that the outcome of the game is uncertain. So, for each time that the game's trend is changed, it is exciting and the more it changed the more exciting is. In the same idea, if the game's trend is the same or the advantageous team still be the advantageous team and trend to has more advantageous, these case makes game unexciting and the outcome is not uncertain.

From the above assumption, let us consider for each case of j, Suppose that blue line is the expected line drawn by using the data from a_{i-1} to a_i, horizontal green line is the line showing the value of a_i by projecting parallel to x-axis and the baseline means the $y = 0$ line which means the advantage equals zero.

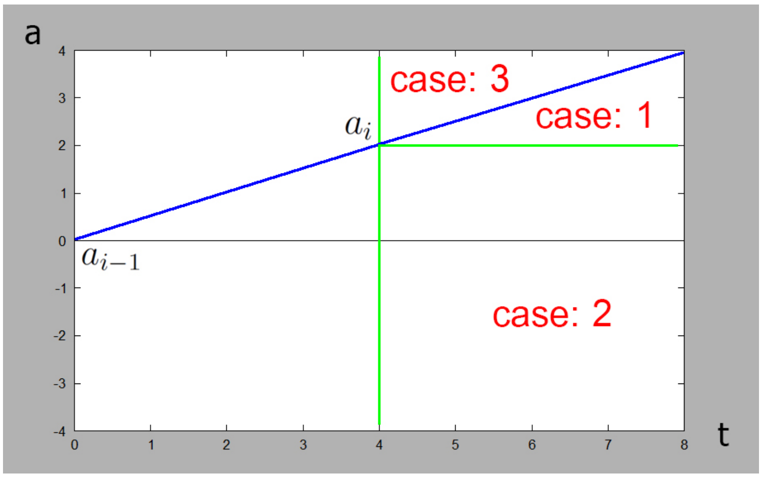

FIGURE 7.4: Many cases of graph of advantage: $Case1$ - $Case3$

From Figure 7.4, the situation is that upper team is advantageous and the previous trend from a_{i-1} to a_i show that the upper team is going to have more advantageous. We can separate this situation in three cases depends on where a_{i+1} will be.

$Case1$: a_{i+1} is below the blue line but not less than a_i. It means that this game's trend tries to change slightly. For this case, the game is exciting because the trend is changed but not much. We can write this case as a mathematical notation shown in Equation (7.8).

$$Case1 \Leftrightarrow (a_i > 0) \wedge (m_{(i-1,i)} \geq 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)}) \wedge (m_{(i,i+1)} \geq 0) \tag{7.8}$$

$Case2$: a_{i+1} is below the blue line and less than a_i. It means that this game's trend is changed largely and the advantage tries to reach zero, or it has already reached zero. For this case, the game is very exciting and the outcome of the game is very uncertain. We can write this case as a mathematical notation shown in Equation (7.9).

$$Case2 \Leftrightarrow (a_i > 0) \wedge (m_{(i-1,i)} \geq 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)}) \wedge (m_{(i,i+1)} < 0) \tag{7.9}$$

$Case3$: a_{i+1} is above the blue line. It means that this game's trend is changed far away from the baseline. For this case, the game is going to be a one-sided game and the game is surely not exciting. We can write this case as a mathematical notation shown in Equation (7.10).

$$Case3 \Leftrightarrow (a_i > 0) \wedge (m_{(i-1,i)} \geq 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)}) \tag{7.10}$$

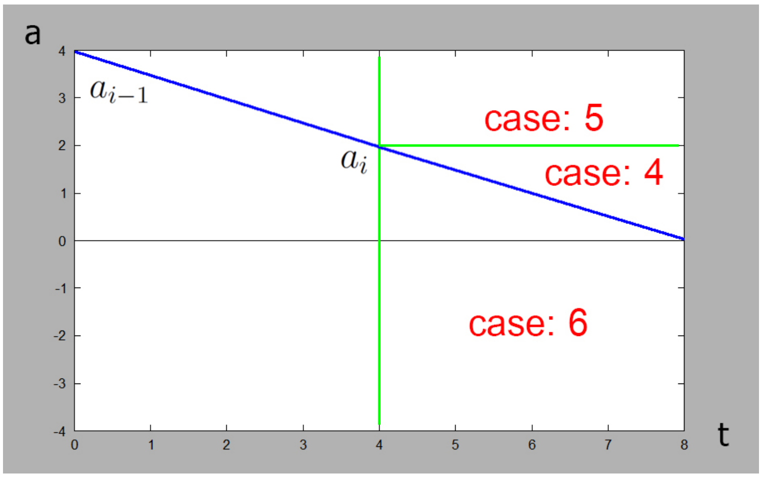

FIGURE 7.5: Many cases of graph of advantage: $Case4$ - $Case6$

From Figure 7.5, the situation is that upper team is advantageous and the previous trend from a_{i-1} to a_i show that the upper team is going to have less advantageous. We can separate this situation in three cases depends on where a_{i+1} will be.

$Case4$: a_{i+1} is above the blue line but not more than a_i. It means that this game's trend tries to change slightly. For this case, the game is exciting because the trend is changed but not much because it changes to the wrong way. We can write this case as a mathematical notation shown in Equation (7.11).

$$Case4 \Leftrightarrow (a_i > 0) \wedge (m_{(i-1,i)} < 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)}) \wedge (m_{(i,i+1)} \leq 0) \qquad (7.11)$$

$Case5$: a_{i+1} is above the blue line and more than a_i. It means that this game's trend is changed from going to the baseline to going out from the baseline. For this case, the game is changed in to the wrong way. We can write this case as a mathematical notation shown in Equation (7.10).

$$Case5 \Leftrightarrow (a_i > 0) \wedge (m_{(i-1,i)} < 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)}) \wedge (m_{(i,i+1)} > 0) \qquad (7.12)$$

$Case6$: a_{i+1} is below the blue line. It means that this game's trend tries to change largely and the advantage tries to reach zero, or it has already reached zero. For this case, the game is exciting and the outcome of the game is uncertain. We can write this case as a mathematical notation shown in Equation (7.13).

$$Case6 \Leftrightarrow (a_i > 0) \wedge (m_{(i-1,i)} < 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)}) \qquad (7.13)$$

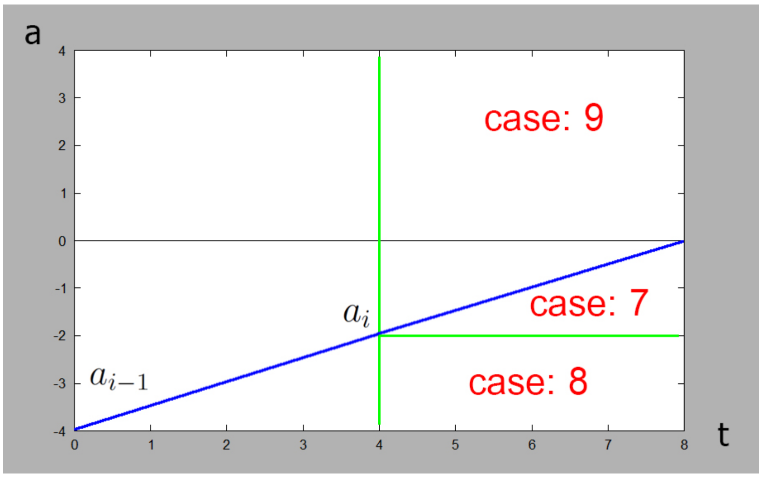

FIGURE 7.6: Many cases of graph of advantage: $Case7$ - $Case9$

From Figure 7.6, the situation is that lower team is advantageous and the previous trend from a_{i-1} to a_i show that the lower team is going to have less advantageous. We can separate this situation in three cases depends on where a_{i+1} will be.

$Case7$: a_{i+1} is below the blue line but not less than a_i. It means that this game's trend tries to change slightly. For this case, the game is exciting because the trend is changed but not much because it changes to the wrong way. We can write this case as a mathematical notation shown in Equation (7.14).

$$Case7 \Leftrightarrow (a_i < 0) \wedge (m_{(i-1,i)} > 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)}) \wedge (m_{(i,i+1)} \geq 0) \qquad (7.14)$$

$Case8$: a_{i+1} is below the blue line and less than a_i. It means that this game's trend is changed from going to the baseline to going out from the baseline. For this case, the game is changed in to the wrong way. We can write this case as a mathematical notation shown in Equation (7.15).

$$Case8 \Leftrightarrow (a_i < 0) \wedge (m_{(i-1,i)} > 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)}) \wedge (m_{(i,i+1)} < 0) \qquad (7.15)$$

$Case9$: a_{i+1} is above the blue line. It means that this game's trend tries to change largely and the advantage tries to reach zero, or it has already reached zero. For this case, the game is exciting and the outcome of the game is uncertain. We can write this case as a mathematical notation shown in Equation (7.16).

$$Case9 \Leftrightarrow (a_i < 0) \wedge (m_{(i-1,i)} > 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)}) \qquad (7.16)$$

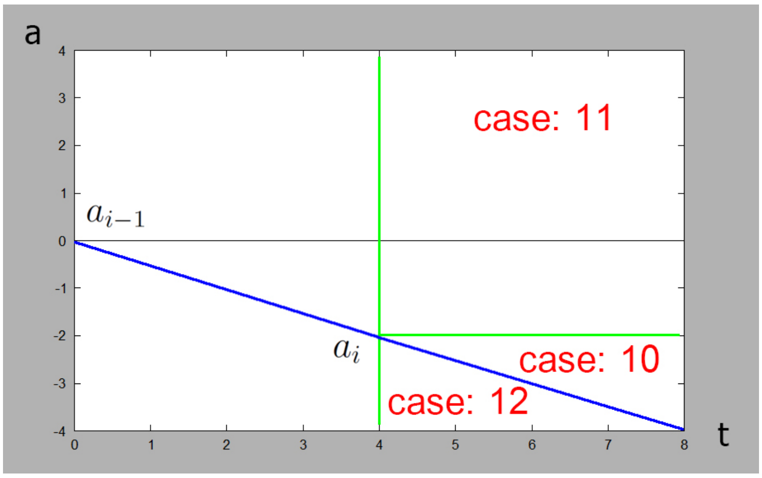

FIGURE 7.7: Many cases of graph of advantage: $Case10$ - $Case12$

From Figure 7.7, the situation is that lower team is advantageous and the previous trend from a_{i-1} to a_i show that the lower team is going to have more advantageous. We can separate this situation in three cases depends on where a_{i+1} will be.

$Case10$: a_{i+1} is above the blue line but not more than a_i. It means that this game's trend tries to change slightly. For this case, the game is exciting because the trend is changed but not much. We can write this case as a mathematical notation shown in Equation (7.17).

$$Case10 \Leftrightarrow (a_i < 0) \wedge (m_{(i-1,i)} \leq 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)}) \wedge (m_{(i,i+1)} \leq 0) \tag{7.17}$$

$Case11$: a_{i+1} is above the blue line and more than a_i. It means that this game's trend is changed largely and the advantage tries to reach zero, or it has already reached zero. For this case, the game is very exciting and the outcome of the game is very uncertain. We can write this case as a mathematical notation shown in Equation (7.18).

$$Case11 \Leftrightarrow (a_i < 0) \wedge (m_{(i-1,i)} \leq 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)}) \wedge (m_{(i,i+1)} > 0) \tag{7.18}$$

$Case12$: a_{i+1} is below the blue line. It means that this game's trend is changed far away from the baseline. For this case, the game is going to be a one-sided game and the game is surely not exciting. We can write this case as a mathematical notation shown in Equation (7.19).

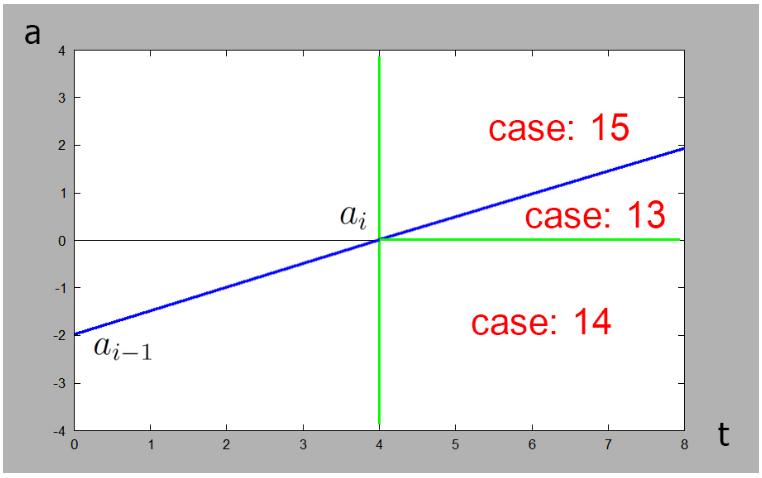

FIGURE 7.8: Many cases of graph of advantage: $Case13$ - $Case15$

$$Case12 \Leftrightarrow (a_i < 0) \wedge (m_{(i-1,i)} \leq 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)}) \tag{7.19}$$

From Figure 7.8, the situation is that the previous trend from a_{i-1} to a_i show that the lower team uses to have advantage but now the advantage is equal to zero. We can separate this situation in three cases depends on where a_{i+1} will be.

$Case13$: a_{i+1} is above the baseline but below the blue line. The game's trend still keep the balance. We can write this case as a mathematical notation shown in Equation (7.20).

$$Case13 \Leftrightarrow (a_i = 0) \wedge (m_{(i-1,i)} > 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)}) \wedge (m_{(i,i+1)} \geq 0) \tag{7.20}$$

$Case14$: a_{i+1} is below the baseline. It means that the game's trend is changed from the previous trend that a_{i-1} to a_i has done. The advantageous team is not changed, the lower team still be an advantageous team. We can write this case as a mathematical notation shown in Equation (7.21).

$$Case14 \Leftrightarrow (a_i = 0) \wedge (m_{(i-1,i)} > 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)}) \wedge (m_{(i,i+1)} < 0) \tag{7.21}$$

$Case15$: a_{i+1} is above the blue line. It means that the game's trend is changed in the same way as previous trend from a_{i-1} to a_i has done. The advantageous team is changed from the lower team to upper team. We can write this case as a mathematical notation shown in Equation (7.22).

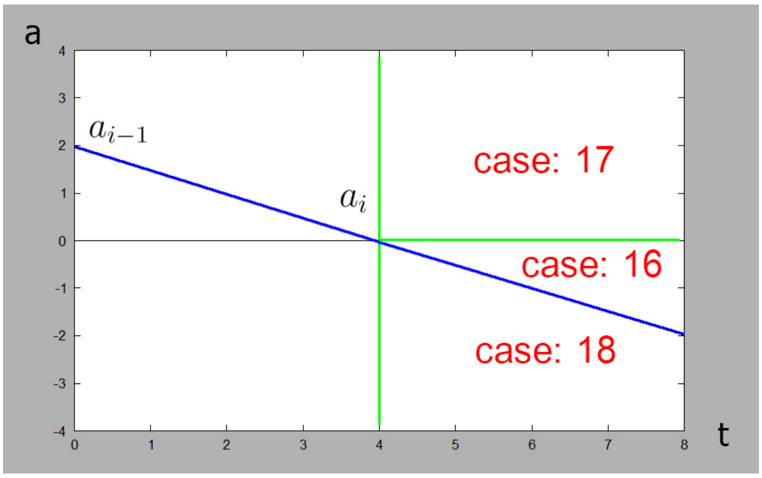

FIGURE 7.9: Many cases of graph of advantage: $Case16$ - $Case18$

$$Case15 \Leftrightarrow (a_i = 0) \wedge (m_{(i-1,i)} > 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)}) \tag{7.22}$$

From Figure 7.9, the situation is that the previous trend from a_{i-1} to a_i show that the upper team uses to have advantage but now the advantage is equal to zero. We can separate this situation in three cases depends on where a_{i+1} will be.

$Case16$: a_{i+1} is below the baseline but above the blue line. The game's trend still keep the balance. We can write this case as a mathematical notation shown in Equation (7.23).

$$Case16 \Leftrightarrow (a_i = 0) \wedge (m_{(i-1,i)} < 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)}) \wedge (m_{(i,i+1)} \leq 0) \tag{7.23}$$

$Case17$: a_{i+1} is above the baseline. It means that the game's trend is changed from the previous trend that a_{i-1} to a_i has done. The advantageous team is not changed, the upper team still be an advantageous team. We can write this case as a mathematical notation shown in Equation (7.24).

$$Case17 \Leftrightarrow (a_i = 0) \wedge (m_{(i-1,i)} < 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)}) \wedge (m_{(i,i+1)} > 0) \tag{7.24}$$

$Case18$: a_{i+1} is below the blue line. It means that the game's trend is changed in the same way as previous trend from a_{i-1} to a_i has done. The advantageous team is changed from the upper team to lower team. We can write this case as a mathematical notation shown in Equation (7.25).

$$Case18 \Leftrightarrow (a_i = 0) \wedge (m_{(i-1,i)} < 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)}) \tag{7.25}$$

However, there still be some exceptional cases shown in Equation (7.26).

$$Case19 \Leftrightarrow (a_i = 0) \wedge (m_{(i-1,i)} = 0) \tag{7.26}$$

To analyze the above 19 cases, we will see that some cases are similar. The difference in those similar cases are just only the advantageous team. For example, we will see that Figure 7.4 and Figure 7.7 is the same situation. Also, Figure 7.5 is the same as Figure 7.6 and Figure 7.8 is the same as Figure 7.9.

Therefore, we can combine some cases using the same j. By combining every case from Equation (7.8) to Equation (7.26), we can construct the full equation of j as shown in Equation (7.27)

$$
\begin{aligned}
&j = 1 \Leftrightarrow (a_i > 0) \wedge (m_{(i-1,i)} \geq 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)}) \wedge (m_{(i,i+1)} \geq 0)\\
&j = 2 \Leftrightarrow (a_i > 0) \wedge (m_{(i-1,i)} \geq 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)}) \wedge (m_{(i,i+1)} < 0)\\
&j = 3 \Leftrightarrow (a_i > 0) \wedge (m_{(i-1,i)} \geq 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)})\\
&j = 4 \Leftrightarrow (a_i > 0) \wedge (m_{(i-1,i)} < 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)}) \wedge (m_{(i,i+1)} \leq 0)\\
&j = 6 \Leftrightarrow (a_i > 0) \wedge (m_{(i-1,i)} < 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)}) \wedge (m_{(i,i+1)} > 0)\\
&j = 5 \Leftrightarrow (a_i > 0) \wedge (m_{(i-1,i)} < 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)})\\
&j = 4 \Leftrightarrow (a_i < 0) \wedge (m_{(i-1,i)} > 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)}) \wedge (m_{(i,i+1)} \geq 0)\\
&j = 6 \Leftrightarrow (a_i < 0) \wedge (m_{(i-1,i)} > 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)}) \wedge (m_{(i,i+1)} < 0)\\
&j = 5 \Leftrightarrow (a_i < 0) \wedge (m_{(i-1,i)} > 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)})\\
&j = 1 \Leftrightarrow (a_i < 0) \wedge (m_{(i-1,i)} \leq 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)}) \wedge (m_{(i,i+1)} \leq 0)\\
&j = 2 \Leftrightarrow (a_i < 0) \wedge (m_{(i-1,i)} \leq 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)}) \wedge (m_{(i,i+1)} > 0)\\
&j = 3 \Leftrightarrow (a_i < 0) \wedge (m_{(i-1,i)} \leq 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)})\\
&j = 7 \Leftrightarrow (a_i = 0) \wedge (m_{(i-1,i)} > 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)}) \wedge (m_{(i,i+1)} \geq 0)\\
&j = 8 \Leftrightarrow (a_i = 0) \wedge (m_{(i-1,i)} > 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)}) \wedge (m_{(i,i+1)} < 0)\\
&j = 9 \Leftrightarrow (a_i = 0) \wedge (m_{(i-1,i)} > 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)})\\
&j = 7 \Leftrightarrow (a_i = 0) \wedge (m_{(i-1,i)} < 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)}) \wedge (m_{(i,i+1)} \leq 0)\\
&j = 8 \Leftrightarrow (a_i = 0) \wedge (m_{(i-1,i)} < 0) \wedge (m_{(i,i+1)} > m_{(i-1,i)}) \wedge (m_{(i,i+1)} > 0)\\
&j = 9 \Leftrightarrow (a_i = 0) \wedge (m_{(i-1,i)} < 0) \wedge (m_{(i,i+1)} < m_{(i-1,i)})\\
&j = 10 \Leftrightarrow (a_i = 0) \wedge (m_{(i-1,i)} = 0)\\
&j = 11 \Leftrightarrow otherwise
\end{aligned}
\tag{7.27}
$$

7.3 Experiments

We have done many experiment by collecting the data from reliable sources and implementing our model. In this section, we show one example from table tennis and one example from DotA 2. We use the setting of λ_j as shown in Equation (7.28).

$$\lambda = [+1, +2, -1, +1, +2, -1, +1, +1, +1, +1, 0] \tag{7.28}$$

In table tennis, we would like to compare two matches, simply called *match A* and *match B*. The graphs of advantage are shown in Figure (7.10) and Figure (7.11) respectively.

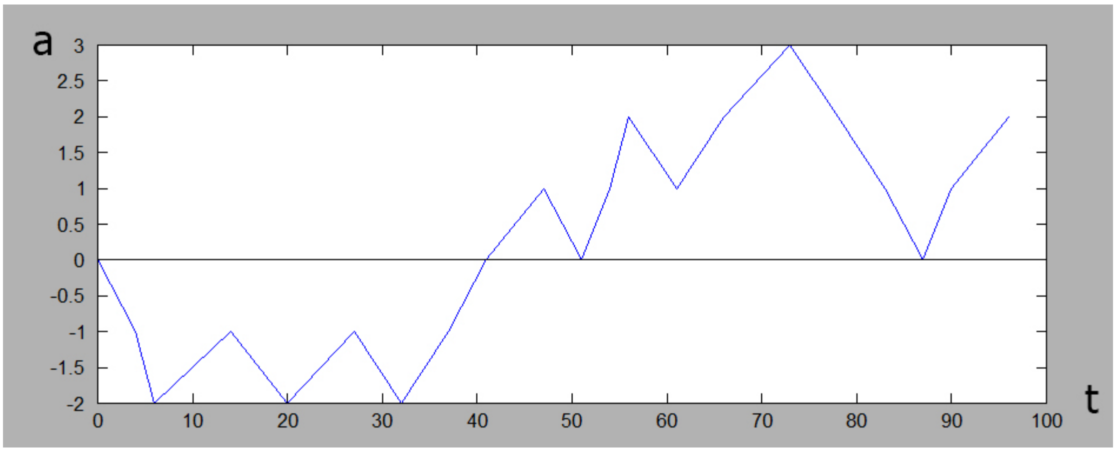

FIGURE 7.10: Match A's graph of advantage

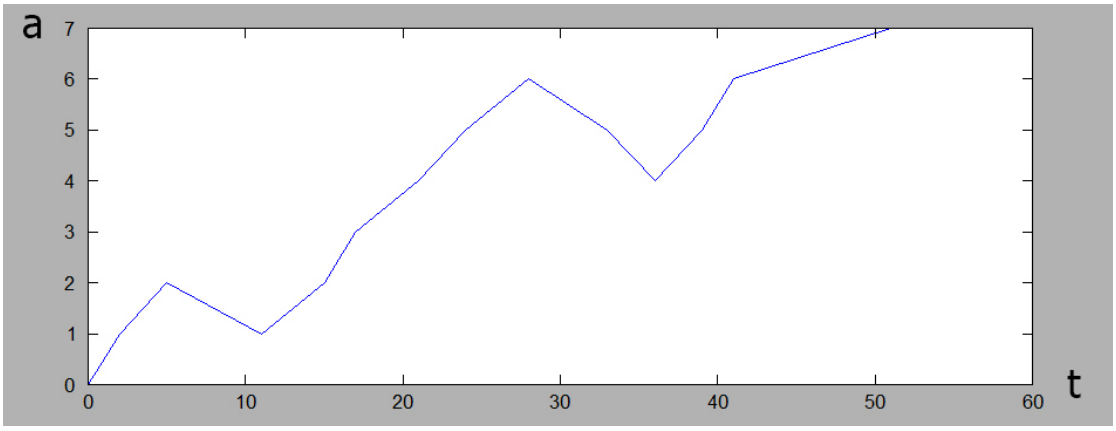

FIGURE 7.11: Match B's graph of advantage

We will see that *match A* is a seesaw game from the Table 7.1. However, for *match B*, it is a one-sided game. Therefore, we expect that *match A* should have *OC* higher than *match B*.

Next, for DotA 2, *match C* shown in Figure (7.12) and *match D* shown in Figure (7.13) will be compared. As we have said that, for DotA 2, we propose to use gold and experience as an advantage. So, we perform experiments both using gold and experience separately.

We will see that *match C* is a seesaw game and *match D* is a one-sided game. Therefore, we expect that *match C* should have *OC* higher than *match D*.

By using our proposed model, the results are shown in Table 7.2.

7.4 Discussion

In this section, we will analyze the results obtained and explain about the further work.

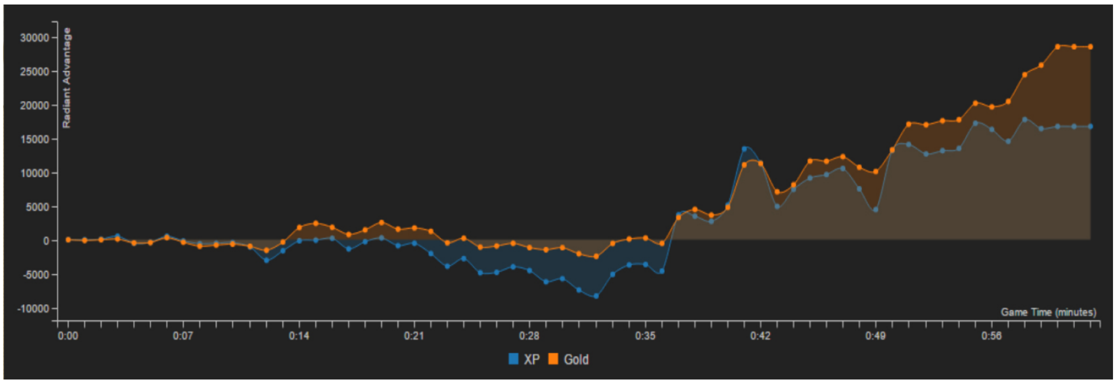

FIGURE 7.12: An example of graph of advantage from Table 7.1

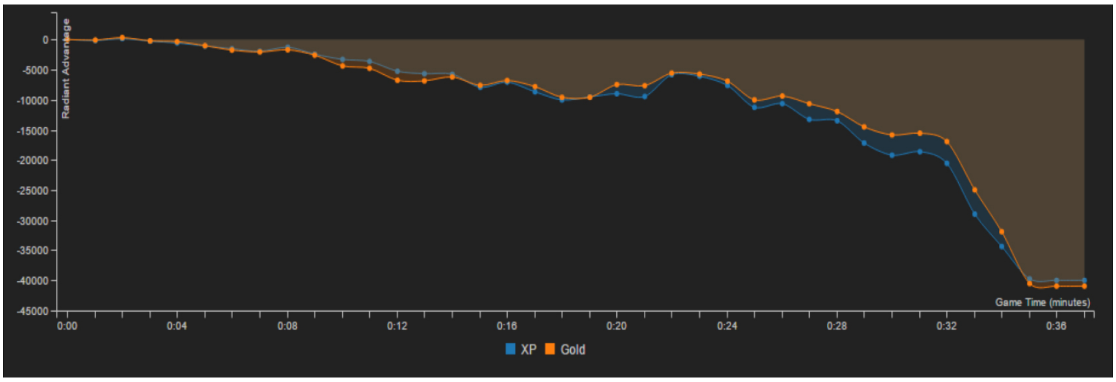

FIGURE 7.13: An example of graph of advantage from Table 7.1

TABLE 7.2: Comparison of OC for each match

match	OC value
match A	0.296
match B	0.118
match C (using XP)	1.480
match D (using XP)	0.841
match C (using gold)	1.565
match D (using gold)	0.895

7.4.1 Results analysis

The setting of λ_j shown in Equation (7.28) is proposed by performing many experiments and try to find a good setting. According to the result from Table 7.2, we will see that, as we have expected, *match A*'s OC is higher than *match B*'s OC and *match C*'s OC is higher than *match D*'s OC. The results confirm that our model works well.

For DotA 2, both using gold and using experience give us the good results. However, the best function should be the combining between important factors. It will be explained in next section.

By the way, as it was said that this research focuses on the enjoyment between matches in the same game, so the higher value of *match D*'s *OC* than *match A*'s *OC* does not mean *match D* is exciting than *match A*. This is because the type of the game is difference. We can simply conclude just only that *match A* is more exciting than *match B* and *match C* is more exciting than *match D* from the uncertainty outcome's point of view.

It is obvious that our proposed model can absolutely separate between seesaw game and one-sided game. However, human also can separate clearly between seesaw game and one-sided game. To go deeply, we should investigate in that issue. In the domain that human and human cannot answer the same answer.

7.4.2 Further work

For graph of advantage, We will see that, for table tennis, using direct score is reasonable enough to create a good graph of advantage. For DotA 2, this study estimate the advatage by using gold and experience as it were implemented. However, it is unclear that what the true generic model of the graph of advantage in a complex game likes DotA 2 is. Therefore, further work should investigate this point by combining many factors in DotA 2. Beside gold and experience that were considered in this research, the factors that should be included in the model are describing below.

- Number of kills and deaths: Kills and deaths are the progress of the game. If we can kill the opponent, we get money and our hero will be stronger. Usually, this factors can be consider as a score of DotA 2. However, there are some rare cases that even though the the team that has lower number of kills can be a winner. So, the number of kills and deaths is not the exact score but it can estimate the advantage of the game.
- Number of remaining buildings: This is because the goal of DotA 2 is to defend our base and attack opponent's base. To attack the base, we need to attack many buildings before attack the base. So, the number of remaining buildings can be considered as one of the progress of the game. Moreover, this factor can be more precise by changing from directly counting the number of remaining buildings to the summation of remaining buildings' HP. This is because the building has HP and sometimes it will be the case that building's hp is decreased but not be destroyed.

If we can reasonably combine these factors, we can propose a generic model to construct the graph of advantage in MOBA games. It should be applied on another MOBA games such as League of Legends [41] and Heroes of Newarth [126] for verifying the efficient of the proposed model.

Furthermore, another domain should be investigated. For example, in Pokemon, the advantage of the game can be estimated by using the remaining HP. Also, board game such as chess whose goal is to destroy opponent's king, is one of the game which cannot use score directly. However, in board games, there are many estimating function proposed in academic field. So, we can use those proposed function to create a graph of advantage.

Additionally, to do the research in the field of game's enjoyment or feeling of player, one thing which is the most important is that we have to directly collect data from human as it was done in Chapter 6. This is because we cannot exactly quantify the enjoyment value by using pure mathematics. Enjoyment is the feeling and only human has feeling. Even though human is very various, the variety of human is the challenge of this research.

That is why the actual experiment with human is needed. For example, in this study, we should evaluate our proposed model by performing the experiment with human. Then, we can use some techniques such as machine learning [8] and supervised learning [18] to find the better setting of λ_j. This will make our model more apparent and empirical.

As we have explain that our full model has four components composed of uncertainty of outcome, exciting shot, special technique and individual fondness. In this study, we focus only uncertainty of outcome component. So, further work should investigate these points.

7.5 Conclusion

This work is an one of attempts to improve the entertainment impact of games by proposing a new model called a generic model for quantifying enjoyment between matches in the same game. We believe that if we can reasonably decide which match is more exciting from the viewer's point of view, that means we can understand more clearly about the word 'enjoyment'. If we can do like that, we will know how to improve the entertainment impact of games which is our purpose of this thesis. That is why this work try to present a generic model for quantifying enjoyment between matches in the same game.

Our proposed model has four components composed of uncertainty of outcome, exciting shot, special technique and individual fondness. In this study, we focus on uncertainty of outcome component. We propose the graph of advantage and how to find the outcome uncertainty value by using this graph. We choose table tennis as a representative of simple game and choose DotA 2 as a representative of complex game. Finally, the experiments are performed and the results show that our model works well.

Chapter 8

Conclusion

This thesis presents various approaches to improve the entertainment impact in games. That is why the problem statement of this thesis is 'how to improve the entertainment impact in games?'. We present three ways from different aspect to archive our goal composed of using game refinement theory, developing emotional AI and proposing a new model for quantifying enjoyment between match in the same game.

For the first approach, using game refinement theory, we first give the fundamental idea of game refinement theory and its previous studies in Chapter 2. Next, the application of this method is presented in the next chapters. Chapter 3 applies game refinement theory to Pokemon. Chapter 4 applies game refinement theory to sports. Chapter 5 applies game refinement theory to RoShamBo and Snake game. From these results, we see that game refinement theory can be one of the significant ways to improve the entertainment impact in games. It can be used for finding comfortable settings and answering some interesting research questions.

For the second approach, developing an emotional AI, Chapter 6 presents a generic model for emotional AI in the domain of real-time multiplayer fighting games. We believe that adding emotional component in AI can make the AI's behaviour more interesting and realistic which makes the game more exciting. The emotional component contains three main factors: personality, memory and mood. Additionally, Moreover, some enhancements are introduced to make the proposed emotional AI more realistic. To assess the proposed idea, we implement the model to our simulated game and fourteen subjects participate in the evaluation experiments. The results show that human can understand the emotion of emotional AI correctly.

For the third approach, proposing a new model for quantifying enjoyment between match in the same game, it is explained in Chapter 7. We believe that if we can figure out a reasonable model to compare the enjoyment between matches, it means that we can truly understand about the enjoyment in game. So, we can improve the entertainment impact of the game reasonably. Our proposed model consists of four components: uncertainty of outcome, exciting shot, special technique and individual fondness. In this study, we focus on uncertainty of outcome component. The experiments are performed by applying the proposed model to table tennis and DotA 2. The results show that our model works well.

As a tentative conclusion, we found that these three approaches can be used for improving the entertainment impact in games. It can answer the problem statement clearly. However, the the work presented here is a basic model and more studies are required. Further works should investigate in many points as it was explained for each part. We hopefully expect that this thesis can be one of the basis to improve the entertainment impact in games and makes human can understand the emotional words like 'enjoyment', 'entertainment', and 'excitement' more clearly.

Bibliography

[1] (11 June 2005). London: The times online. [Retrieved 2009-06-09].

[2] (2008). BBC News. Fewer Sports for London Olympics. July 8, 2005.

[3] (2008). "softball." The Columbia Encyclopedia. New York: Columbia University Press, 2008.

[4] A. Egges, S. K. and Magnenat-Thalmann, N. (2003). A model for personality and emotion simulation. *Knowledge-Based Intelligent Information and Engineering Systems*, pages 453–461.

[5] AIBA (2016). Aiba official site. http://en.wikipedia.org/wiki/Boxing/.

[6] Allen, J. G. and Hamsher, J. H. (1974). The development and validation of a test of emotional styles. *Journal of Consulting and Clinical Psychology*, 42(5):663.

[7] Aloupis, G., Demaine, E. D., Guo, A., and Viglietta, G. (2015). Classic nintendo games are (computationally) hard. *Theoretical Computer Science*, 586:135–160.

[8] Andrieu, C., De Freitas, N., Doucet, A., and Jordan, M. I. (2003). An introduction to mcmc for machine learning. *Machine learning*, 50(1-2):5–43.

[9] arcade history (2016). An arcade game history site. http://www.arcade-history.com/.

[10] Atkinson, R. C. and Shiffrin, R. M. (1968). *Human Memory: A Proposed System and Its Control Processes*, volume 2. The Psychology of Learning and Motivation.

[11] Baddeley, A. D. (1997). *Human Memory: Theory and Practice*. Psychology Press Ltd, revised edition.

[12] Bainbridge, J. (2014). it is a pokémon world: The pokémon franchise and the environment. *International Journal of Cultural Studies*, 17(4):399–414.

[13] Billings, D. (2000). The first international roshambo programming competition. *International Computer Games Association (ICGA) Journal*, 23(1):42–50.

[14] Bo Li, Duoyong Sun, S. G. and Lin, Z. (2014). Agent-based simulation of group emotions evolution and strategy intervention in extreme events. *Discrete Dynamics in Nature and Society*. Article ID 464190.

[15] Bulbapedia (2016). An encyclopedia about pokemon. http://bulbapedia.bulbagarden.net/.

[16] Campbell, M., Hoane, A. J., and Hsu, F.-h. (2002). Deep blue. *Artificial intelligence*, 134(1):57–83.

[17] Carnagey, N. L., Anderson, C. A., and Bushman, B. J. (2007). The effect of video game violence on physiological desensitization to real-life violence. *Journal of Experimental Social Psychology*, 43(3):489–496.

[18] Caruana, R. and Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning*, pages 161–168. ACM.

[19] Cattell, H. E. and Mead, A. D. (2008). The sixteen personality factor questionnaire (16pf). *The SAGE handbook of personality theory and assessment*, 2:135–178.

[20] Chen, S. D. (2014). A crude analysis of twitch plays pokemon. *arXiv preprint arXiv:1408.4925*.

[21] Chiewvanichakorn, R., Nossal, N., and Iida, H. (2015). Game refinement model with consideration on playing cost: A case study using crane games. In *Knowledge and Smart Technology (KST), 2015 7th International Conference on*, pages 87–92. IEEE.

[22] Chittaro, L. and Sioni, R. (2012). Turning the classic snake mobile game into a location–based exergame that encourages walking. In *International Conference on Persuasive Technology*, pages 43–54. Springer.

[23] Cincotti, A., Iida, H., and Yoshimura, J. (2007). Refinement and complexity in the evolution of chess. In *the 10th International Conference on Computer Science and Informatics*, pages 650–654. Springer.

[24] Clocksin, W. (2004). *Memory and emotion in the cognitive architecture, in: D. Davis (Ed.)*. Visions of Mind, IDEA Group Publishing.

[25] Conaire, C. Ó., Kelly, P., Connaghan, D., and O'Connor, N. E. (2009). Tennissense: A platform for extracting semantic information from multi-camera tennis data. In *2009 16th International Conference on Digital Signal Processing*, pages 1–6. IEEE.

[26] Connaghan, D., Kelly, P., and O'Connor, N. E. (2011). Game, shot and match: Event-based indexing of tennis. In *Content-Based Multimedia Indexing (CBMI), 2011 9th International Workshop on*, pages 97–102. IEEE.

[27] Cook, R., Bird, G., Lünser, G., Huck, S., and Heyes, C. (2011). Automatic imitation in a strategic context: players of rock–paper–scissors imitate opponents' gestures. *Proceedings of the Royal Society of London B: Biological Sciences*.

[28] DeMaria, R. and Wilson, J. (2004). *High Score!: The Illustrated History of Electronic Games*. Computer Games Series. McGraw-Hill/Osborne.

[29] Deterding, S., Dixon, D., Khaled, R., and Nacke, L. (2011). From game design elements to gamefulness: defining gamification. In *Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments*, pages 9–15. ACM.

[30] Diah, N. M., Sutiono, A. P., Zuo, L., Nossal, N., Iida, H., Azan, N., and Zin, M. (2015). Quantifying engagement of video games: Pac-man and dota (defense of the ancients). In *17th International Conference on Mathematical and Computational Methods in Science and Engineering (MACMESE15), Kuala Lumpur*.

[31] Digman, J. M. (1990). Personality structure : Emergence of the five factor model. *Annual Review of Psychology*, 41:417–440.

[32] Dragonflycave (2016). A pokemon catch rate calculator. http://www.dragonflycave.com.

[33] Ehlis, T. (2000). Application of genetic programming to the "snake game". *Gamedev. Net*, 175.

[34] El-Nasr, M. S., Yen, J., and Ioerger, T. R. (2000). Flamefuzzy logic adaptive model of emotions. *Autonomous Agents and Multi-agent systems*, 3(3):219–257.

[35] Elisabeth Andre, Martin Klesen, P. G. S. A. and Rist, T. (1999). Integrating models of personality and emotions into lifelike characters. In *Proceedings of the workshop on Affect in Interactions - Towards a new Generation of Interfaces in conjunction with the 3rd i3 Annual Conference*, pages 136–149, Siena, Italy. A. Paiva and C. Martinho.

[36] Eysenck, H. (1990). *Biological dimensions of personality.* Pervin, L.A., ed.: Handbook of personality: Theory and research. New York: Guilford.

[37] Ferrell, O. C. and Gresham, L. G. (1985). A contingency framework for understanding ethical decision making in marketing. *The Journal of Marketing*, pages 87–96.

[38] Fisher, L. (2008). *Rock, paper, scissors: game theory in everyday life*. Basic Books.

[39] Fudenberg, D. and Tirole, J. (1991). *Game Theory*. MIT Press.

[40] Gamefaqs (2016). Guides and walkthroughs of many video games. http://www.gamefaqs.com/.

[41] Games, R. (2016). an official league of legends website. http://na.leagueoflegends.com/.

[42] Gettings, J. and from infoplease, M. Z. (2012). Olympic preview: Softball. http://www.infoplease.com/spot/summer-olympics-softball.html.

[43] Gingold, Y. I. (2006). From rock, paper, scissors to street fighter ii: Proof by construction. In *Proceedings of the 2006 ACM SIGGRAPH symposium on Videogames*, pages 155–158. ACM.

[44] Goggin, G. (2011). *Global Mobile Media*. Media and communication studies. Routledge.

[45] Govan, A. Y., Langville, A. N., Meyer, C. D., et al. (2009). Offense-defense approach to ranking team sports. *Journal of Quantitative Analysis in Sports*, 5(1):1151.

[46] Gratch, J. and Marsella, S. (2004). Evaluating a general model of emotional appraisal and coping. *AAAI Symposium on Architectures for modeling emotion: cross-disciplinary foundations*.

[47] Gross, James J.; Thompson, R. A. (2007). Emotion regulation: Conceptual foundations. *Handbook of emotion regulation*, pages 3–24.

[48] Gupta, R. and Derevensky, J. L. (1996). The relationship between gambling and video-game playing behavior in children and adolescents. *Journal of gambling studies*, 12(4):375–394.

[49] Hahn, A., Helmer, R., Kelly, T., Partridge, K., Krajewski, A., Blanchonette, I., Barker, J., Bruch, H., Brydon, M., Hooke, N., et al. (2010). Development of an automated scoring system for amateur boxing. *Procedia Engineering*, 2(2):3095–3101.

[50] Häikiö, M. et al. (2002). Nokia-the inside story.

[51] Halpern, J. Y. and Pass, R. (2011). Algorithmic rationality: Adding cost of computation to game theory. *ACM SIGecom Exchanges*, 10(2):9–15.

[52] Hancinema (2016). Hancinema pages. http://www.hancinema.net/play-muk-zzi-ppa-theupgraded-rock-paper-scissors-30907.html.

[53] Horton, J. (2012). got my shoes, got my pokémon: Everyday geographies of childrens popular culture. *Geoforum*, 43(1):4–13.

[54] Hudlicka, E. (2008). What are we modeling when we model emotion? In *AAAI spring symposium: emotion, personality, and social behavior*, pages 52–59.

[55] Iida, H. and Nakagawa, T. (2011). Game information dynamics. In *International Conference on Entertainment Computing*, pages 403–406. Springer.

[56] Iida, H., Nakagawa, T., Sone, S., Muangkasem, A., and Ishitobi, T. (2011a). Safety lead curve and entertainment in games.

[57] Iida, H., Nakagawa, T., Spoerer, K., and Sone, S. (2011b). Three elemental game progress patterns. In *International Conference on Intelligent Science and Intelligent Data Engineering*, pages 571–581. Springer.

[58] Iida, H., Takahara, K., Nagashima, J., Kajihara, Y., and Hashimoto, T. (2004). An application of game-refinement theory to mah jong. In *Entertainment Computing–ICEC 2004*, pages 333–338. Springer.

[59] Iida, H., Takeshita, N., and Yoshimura, J. (2003). A metric for entertainment of boardgames: its implication for evolution of chess variants. In *Entertainment Computing*, pages 65–72. Springer.

[60] Iqbal, M. A. B. M. (2008). *A discrete computational aesthetics model for a zero-sum perfect information game*. PhD thesis, UNIVERSITY OF MALAYA KUALA LUMPUR.

[61] ITF (2016). International tennis federation handbook. http://www.itftennis.com/officiating/rulebooks/.

[62] James, W. (1884). What is an emotion? *Mind*, 9:188–205.

[63] Jegers, K. (2007). Pervasive game flow: understanding player enjoyment in pervasive gaming. *Computers in Entertainment (CIE)*, 5(1):9.

[64] Jennings, L. (2014). *She's a Knockout!: A History of Women in Fighting Sports*. Rowman & Littlefield Publishers.

[65] Kim, E. J., Namkoong, K., Ku, T., and Kim, S. J. (2008). The relationship between online game addiction and aggression, self-control and narcissistic personality traits. *European psychiatry*, 23(3):212–218.

[66] Kleinginna Jr, P. R. and Kleinginna, A. M. (1981). A categorized list of emotion definitions, with suggestions for a consensual definition. *Motivation and emotion*, 5(4):345–379.

[67] Klimmt, C., Hartmann, T., and Frey, A. (2007). Effectance and control as determinants of video game enjoyment. *Cyberpsychology & behavior*, 10(6):845–848.

[68] Klimmt, C., Rizzo, A., Vorderer, P., Koch, J., and Fischer, T. (2009). Experimental evidence for suspense as determinant of video game enjoyment. *CyberPsychology & Behavior*, 12(1):29–31.

[69] Knowles, G., Sherony, K., and Haupert, M. (1992). The demand for major league baseball: A test of the uncertainty of outcome hypothesis. *The American Economist*, 36(2):72–80.

[70] Kshirsagar, S. and Magnenat-Thalmann, N. (2002). A multilayer personality model. *SMARTGRAPH '02 Proceedings of the 2nd international symposium on Smart graphics*, pages 107–115.

[71] Layard, R., Mayraz, G., and Nickell, S. (2008). The marginal utility of income. *Journal of Public Economics*, 92(8):1846–1857.

[72] Lee, G.-G., Kim, H.-k., and Kim, W.-Y. (2009a). Highlight generation for basketball video using probabilistic excitement. In *2009 IEEE International Conference on Multimedia and Expo*, pages 318–321. IEEE.

[73] Lee, H., Choi, H. J., Park, J. H., Lee, J. H., and Jung, S. (2009b). Center of gravity based control of a humanoid balancing robot for boxing games: Balbot v. In *ICCAS-SICE, 2009*, pages 124–128. IEEE.

[74] Lin, W.-M., Cheng, F.-S., and Tsay, M.-T. (2001). Nonconvex economic dispatch by integrated artificial intelligence. *IEEE Transactions on Power Systems*, 16(2):307–311.

[75] Lin, Y.-H. (2007a). Integrating scenarios of video games into classroom instruction. In *Information Technologies and Applications in Education, 2007. ISITAE'07. First IEEE International Symposium on*, pages 593–596. IEEE.

[76] Lin, Y.-H. (2007b). Pokemon: game play as multi-subject learning experience. In *Digital Game and Intelligent Toy Enhanced Learning, 2007. DIGITEL'07. The First IEEE International Workshop on*, pages 182–184. IEEE.

[77] Liponski, W. (2003). *World Sports Encyclopedia*. MBI.

[78] MacDorman, K. F. (2006). Subjective ratings of robot video clips for human likeness, familiarity, and eeriness: An exploration of the uncanny valley. In *ICCS/CogSci-2006 long symposium: Toward social mechanisms of android science*, pages 26–29.

[79] Magalie Ochs, Karl Devooght, D. S. and Pelachaud, C. (2006). A computational model of capability-based emotion elicitation for rational agent. In *Proceedings of the 1st workshop on Emotion and Computing - Current Research and Future Impact*, Bremen, Germany.

[80] Májek, P. and Iida, H. (2004). Uncertainty of game outcome. In *3rd International Conference on Global Research and Education in Intelligent Systems*, pages 171–180.

[81] Mandelbaum, M. (2004). *The Meaning of Sports: Why Americans Watch Baseball, Football, and Basketball, and what They See when They Do.* Public Affairs.

[82] Maria, K. A. and Zitar, R. A. (2007). Emotional agents: A modeling and an application. *Information and Software Technology*, 49(7):695–716.

[83] Mariaa, K. A. and Zitarb, R. A. (2007). Emotional agents: A modeling and an application. *Information and Software Technology*, 49:695–716.

[84] Martinez-Miranda, J. and Aldea, A. (2005). Emotions in human and artificial intelligence. *Computers in Human Behavior*, 21:323–341.

[85] McCauley, J. (2008). "MLB Wants Baseball Back in Olympics". Washington Times. Associated Press.

[86] McCrae, R. R. and John, O. P. (1992). An introduction to the five factor model and its applications. special issue: The fivefactor model: Issues and applications. *Journal of Personality*, 60:175–215.

[87] MILLER, S. (2014). Two extra sets, little added drama; best-of-five-set matches have little effect on outcome in men's tennis. *New York Times.*

[88] Miyake, Y. (2008). How to use ai technologies to develop digital games. *The Journal of Japanese Society for Artificial Intelligence*, pages 44–51.

[89] Miyake, Y. (2013). The fundamental theory of digital game ai. *Journal of Virtual Reality Society of Japan*, 18:28–33.

[90] Moore, M. (2016). *Basics of Game Design.* CRC Press.

[91] Mozgovoy, M. and Umarov, I. (2010). Building a believable and effective agent for a 3d boxing simulation game. In *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on*, volume 3, pages 14–18. IEEE.

[92] Murray, H. J. R. (1913). *A History of Chess.* Oxford University Press.

[93] Newborn, M. (1996). *Kasparov versus Deep Blue: Computer Chess Comes of Age.* Springer.

[94] Newton, P. K., Aslam, K., et al. (2009). Monte carlo tennis: a stochastic markov chain model. *Journal of Quantitative Analysis in Sports*, 5(3):1–42.

[95] Nolan, K. (2002). Bionic roshambo hand gesture recognition as an arcade gaming interface. Master's thesis, Department of Computer and Information Systems, College of Informatics and Electronics, University of Limerick.

[96] Ogletree, S. M., Martinez, C. N., Turner, T. R., and Mason, B. (2004). Pokémon: exploring the role of gender. *Sex roles*, 50(11-12):851–859.

[97] Olympic (2016). Olympic boxing official site. http://www.olympic.org/boxing/.

[98] O'Malley, A. J. et al. (2008). Probability formulas and statistical analysis in tennis. *Journal of Quantitative Analysis in Sports*, 4(2):15.

[99] Panumate, C., Xiong, S., and Iida, H. (2015a). An approach to quantifying pokemon's entertainment impact with focus on battle. In *Applied Computing and Information Technology/2nd International Conference on Computational Science and Intelligence (ACIT-CSI), 2015 3rd International Conference on*, pages 60–66. IEEE.

[100] Panumate, C., Xiong, S., Iida, H., and Kondo, T. (2015b). Evolutionary changes of pokemon game: A case study with focus on catching pokemon. In *Entertainment Computing-ICEC 2015*, pages 182–194. Springer.

[101] Pappas, A. M., Zawacki, R. M., and Sullivan, T. J. (1985). Biomechanics of baseball pitching a preliminary report. *The American journal of sports medicine*, 13(4):216–222.

[102] Pedersen, C., Togelius, J., and Yannakakis, G. N. (2009). Modeling player experience in super mario bros. In *2009 IEEE Symposium on Computational Intelligence and Games*, pages 132–139. IEEE.

[103] Pokemonshowdown (2016). An online pokemon battle simulator. http://pokemonshowdown.com.

[104] Polk, T., Yang, J., Hu, Y., and Zhao, Y. (2014). Tennivis: Visualization for tennis match analysis. *IEEE transactions on visualization and computer graphics*, 20(12):2339–2348.

[105] Pozzato, G., Michieletto, S., and Menegatti, E. (2013). Towards smart robots: Rock-paper-scissors gaming versus human players. In *PAI@ AI* IA*, pages 89–95. Citeseer.

[106] Qing-mei, M. and Ai-lian, C. (2011). Artificial emotional interaction model based on efsm. *Electronic and Mechanical Engineering and Information Technology (EMEIT)*, 4.

[107] Ray, S. G. (2006). Gender inclusive game design: Expanding the market (advances in computer graphics and game development series).

[108] reference, S. (2016). Baseball reference. http://www.baseball-reference.com.

[109] Reinecke, L. (2009). Games and recovery: The use of video and computer games to recuperate from stress and strain. *Journal of Media Psychology*, 21(3):126–142.

[110] Reisenzein, R. (2007). What is a definition of emotion? and are emotions mental-behavioral processes? *Social Science Information*, 7:26–29.

[111] Rottenberg, S. (1956). The baseball players' labor market. *The Journal of Political Economy*, pages 242–258.

[112] Rouse III, R. (2010). *Game design: Theory and practice*. Jones & Bartlett Learning.

[113] Rui, Y., Gupta, A., and Acero, A. (2000). Automatically extracting highlights for tv baseball programs. In *Proceedings of the eighth ACM international conference on Multimedia*, pages 105–115. ACM.

[114] Russell, S. and Norvig, P. (2009). *Artificial Intelligence: A Modern Approach*. Prentice Hall, third edition.

[115] Salen, K. and Zimmerman, E. (2004). *Rules of play: Game design fundamentals*. MIT press.

[116] Salovey, P. and Mayer, J. D. (1990). Emotional intelligence. *Imagination, cognition and personality*, 9(3):185–211.

[117] Sanderson, A. R. and Siegfried, J. J. (2003). Thinking about competitive balance. *Journal of Sports Economics*, 4(4):255–279.

[118] Schank, R. C. (1987). What is ai, anyway? *AI magazine @AAAI*, 8:59–65.

[119] Schell, J. (2014). *The Art of Game Design: A book of lenses*. CRC Press.

[120] Schreyer, D. and Torgler, B. (2016). On the role of race outcome uncertainty in the tv demand for formula 1 grands prix. *Journal of Sports Economics*, page 1527002515626223.

[121] Serebii (2016). A pokemon fan site. http://www.serebii.net/.

[122] Silva, J. J. R., Del Vecchio, F. B., Picanço, L. M., Takito, M. Y., and Franchini, E. (2011). Time-motion analysis in muay-thai and kick-boxing amateur matches.

[123] Skalski, P., Tamborini, R., Shelton, A., Buncher, M., and Lindmark, P. (2010). Mapping the road to fun: Natural video game controllers, presence, and game enjoyment. *New Media & Society*, page 1461444810370949.

[124] Staudohar, P. D. (2005). Performance-enhancing drugs in baseball. *Labor Law Journal*, 56(2):139.

[125] Strategywiki (2016). A wiki for all your video game strategy guide. http://strategywiki.org/wiki/.

[126] Studios, F. (2016). an official heroes of newerth website. http://www.heroesofnewerth.com/.

[127] Sudhir, G., Lee, J. C.-M., and Jain, A. K. (1998). Automatic classification of tennis video for high-level content-based retrieval. In *Content-Based Access of Image and Video Database, 1998. Proceedings., 1998 IEEE International Workshop on*, pages 81–90. IEEE.

[128] Sutiono, A. P., Purwarianti, A., and Iida, H. (2014). A mathematical model of game refinement. In *Intelligent Technologies for Interactive Entertainment*, pages 148–151. Springer.

[129] Sutiono, A. P., Ramadan, R., Jarukasetporn, P., Takeuchi, J., Purwarianti, A., and Iida, H. (2015). A mathematical model of game refinement and its applications to sports games.

[130] Sweepercalc (2016). Pokemon usage stats. http://sweepercalc.com/stats/.

[131] Team, M. D. (2016). The evolution of snake. http://blogs.windows.com/devices/2010/11/25/the-evolution-of-snake/.

[132] Tennisbetsite (2016). Tennis results site. http://www.tennisbetsite.com/tournaments.html.

[133] Tobin, J. (2004). *Pikachu's Global Adventure: The Rise and Fall of Pokémon*. Duke University Press.

[134] Tsebelis, G. (1995). Decision making in political systems: Veto players in presidentialism, parliamentarism, multicameralism and multipartyism. *British journal of political science*, 25(03):289–325.

[135] Valdez, S. E., Barayuga, V. J. D., and Fernandez, P. L. (2014). The effectiveness of using a historical sequence-based predictor algorithm in the first international roshambo tournament.

[136] Valdez, S. E., Siddayao, G. P., and Fernandez, P. L. (2015). The effectiveness of using a modified" beat frequent pick" algorithm in the first international roshambo tournament. *International Journal of Information and Education Technology*, 5(10):740.

[137] Valve (2016). an official dota 2 website. http://blog.dota2.com/.

[138] Vasquez, V. (2003). What pokémon can teach us about learning and literacy. *Language Arts*, 81(2):118–125.

[139] Vecer, J., Ichiba, T., Laudanovic, M., et al. (2007). On probabilistic excitement of sports games. *Journal of Quantitative Analysis in Sports*, 3(3):1–23.

[140] Velasquez, J. D. (1997). Modeling emotions and other motivations in synthetic agents. *AAAI/IAAI*, pages 10–15.

[141] Von Ahn, L. and Dabbish, L. (2008). Designing games with a purpose. *Communications of the ACM*, 51(8):58–67.

[142] Vossen, D. P. et al. (2004). The nature and classification of games. *AVANTE-ONTARIO-*, 10(1):53–68.

[143] Wattanamongkhol, N., Kumhom, P., and Chamnongthai, K. (2005). A method of glove tracking for amateur boxing refereeing. In *IEEE International Symposium on Communications and Information Technology, 2005. ISCIT 2005.*, volume 1, pages 6–9. IEEE.

[144] WBA (2016). Wba official site. http://www.wbanews.com/.

[145] Wig, W. (2016). *The Complete Pokemon Pokedex List (English Version)*. Gamas Publishing.

[146] Wikipedia (2016a). Tennis pages on wikipedia. http://www.tennisbetsite.com/tournaments.html.

[147] Wikipedia (2016b). Wikipedia rock-paper-scissors pages. https://en.wikipedia.org/wiki/Rock-paper-scissors.

[148] Wikipedia (2016c). Wikipedia snake game page. https://en.wikipedia.org/wiki/Snake_(video_game).

[149] Xiong, S. and Iida, H. (2014). Attractiveness of real time strategy games. In *Systems and Informatics (ICSAI), 2014 2nd International Conference on*, pages 271–276. IEEE.

[150] Xiong, S., Zuo, L., Chiewvanichakorn, R., and Iida, H. (2014). Quantifying engagement of various games. In *The 19th Game Programming Workshop 2014*. Information Processing Society of Japan.

[151] Xuan-Vinh, H., Panumate, C., Iida, H., and Jean-Christophe, T. (2015). Deciphering the entertaining impact of pokemon. *International Journal of Engineering Research and Management (IJERM)*, 2(11):83–90.

[152] YASP (2016). an open source dota 2 statistics. https://yasp.co/.

Curriculum Vitae

Chetprayoon Panumate was born on Wednesday 12th of February 1992 in Bangkok, Thailand. He is interested in game since he was young. After graduating high school from Bangkok Christian College, he enrolled in a department of computer engineering, faculty of engineering, Chulalongkorn University with a dream that he can be a game programmer or game designer in someday. During his bachelor life, he did many activities in university such as many kinds of volunteer camps, debate club and academic club. He was a president of academic affairs, engineering student council, faculty of engineering, Chulalongkorn University in 2013. Beside these activities, he is also interested in academic contests activities so he participated some game programming contests and won some prizes. After graduating, he decided to grab a great chance from Professor Hiroyuki Iida who is a specialist in the field of game research. He received the scholarship and enjoyed his life in Japan. In the first year, he was supported from his advisor to be a four-month internship student in SQUARE ENIX CO., LTD. and also had a chance to stay with Japanese host family. Finally, he got a job offer and he will continue his life in Japan as an AI related engineer.

Publications

Chetprayoon Panumate, Shuo Xiong and Hiroyuki Iida. An Approach to Quantifying Pokemon Entertainment Impact with Focus on Battle. In Applied Computing and Information Technology/2nd International Conference on Computational Science and Intelligence (ACIT-CSI), 2015 3rd International Conference on, pages 60-66. IEEE, 2015.

Chetprayoon Panumate, Shuo Xiong, Hiroyuki Iida and Toshiaki Kondo. Evolutionary Changes of Pokemon Game: A Case Study With Focus on Catching Pokemon. In Entertainment Computing-ICEC 2015, pages 182-194. Springer, 2015.

Ho Xuan-Vinh, Chetprayoon Panumate, Hiroyuki Iida and Jean-Christophe Terillion. Deciphering the Entertaining Impact of Pokemon. International Journal of Engineering Research and Management (IJERM), 2(11):83-90, 2015.

Chetprayoon Panumate, Hiroyuki Iida and Abu-Bakar Nordin. Measuring the Excitement in Boxing Game with Game Refinement Theory. International Conference on Hospitality, Leisure, Sports, and Tourism (HLST 2016).

Anh-Tuan Dinh, Chetprayoon Panumate and Hiroyuki Iida. Attractiveness of Tennis Over Conservative Rule Systems. International Conference on Information and Social Science (ISS 2016).

Yuranan Kitrungrotsakul, Chetprayoon Panumate, Hiroyuki Iida and Kiyofumi Tanaka. Measuring Sophistication of Sports Games: The First Result from Baseball. IPSJ SIG-GI Technical Report, vol. 2016-GI-36, no. 4, pages 1-7.

Chetprayoon Panumate and Hiroyuki Iida. Finding Comfortable Settings of Video Games: Using Game Refinement Measure in Pokemon Battle AI. IPSJ SIG-GI Technical Report, vol. 2016-GI-36, no. 6, pages 1-7.

Chetprayoon Panumate and Hiroyuki Iida. Developing Pokemon AI for Finding Comfortable Settings. Proceedings of 2016 Summer Conference, Digital Games Research Association JAPAN, August 2016, pages 168-171.

Anunpattana Punyawee, Chetprayoon Panumate and Hiroyuki Iida. Finding Comfortable Settings of Snake Game using Game Refinement Measurement. 8th International Conference on Computer Science and its Applications (CSA-16). in press.

Chetprayoon Panumate and Hiroyuki Iida. Quantifying Enjoyment of Individual Match in Games. 2016 The Annual Conference on Engineering and Applied Science (ACEAT 2016). in press.

Chetprayoon Panumate, Youichiro Miyake and Hiroyuki Iida. A Generic Model for Emotional AI in Real-Time Multiplayer Fighting Games. The Games and Learning Alliance conference (GALA 2016). submitted.

Chetprayoon Panumate, Jean-Christophe Terillion and Hiroyuki Iida. A Game Informatical Analysis of RoShamBo. 2016 Conference on Technologies and Applications of Artificial Intelligence (TAAI 2016). submitted.

Chetprayoon Panumate, Ryo Takahashi and Hiroyuki Iida. Analyzing Sports using Game Refinement Measure. ODISHA JOURNAL OF SOCIAL SCIENCE. submitted.