THE DEVELOPMENT OF A FRAMEWORK FOR WEAPON BALANCING IN MULTIPLAYER FIRST-PERSON SHOOTER GAMES

by

Carly Fox

A Thesis

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Master of Science

Department of Computer Graphics Technology

West Lafayette, Indiana

May 2020

THE PURDUE UNIVERSITY GRADUATE SCHOOL

STATEMENT OF COMMITTEE APPROVAL

Dr. David Whittinghill, Chair

Department of Computer Graphics Technology

Dr. Tim McGraw

Department of Computer Graphics Technology

Dr. Christos Mousas

Department of Computer Graphics Technology

Approved by:

Dr. Nicoletta Adamo

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

GLOSSARY

Deathmatch – a game mode in first-person shooter video games in which players duel each other in an attempt to be first to reach a specified number of kills

Game Balancing – "the modification of parameters of the constitutive and operational rules of a game . . . in order to achieve optimal configurations in terms of a set of goals" (Volz, Rudolph, & Naujoks, 2016, p. 1)

Parameter – "a variable in a game that has been deemed valuable for changing and balancing" (Morosan & Poli, 2018, p. 265)

Parameter Tuning – "making low-level changes to game mechanic settings such as character movement parameters, power-up item effects, or control sensitivity" (Zook, Fruchter, & Riedl, 2014, p. 1)

Weapon Loadout – weapons held by the player

ABSTRACT

Achieving a state of balance is essential when developing a video game (Schell, 2019, "Game Mechanics Must Be in Balance," para. 1). Despite this, game balancing is frequently overlooked in game development curricula (Schreiber, 2016, 00:30). This research describes the development and pilot study of a framework that junior game designers can utilize to gain valuable skills in the area of game balance. The framework produced by this research provides users with the ability to tune weapon parameters and see the effects these changes have on a first-person shooter deathmatch game in real time. Participants in the study utilized the framework to achieve three pacing and balance goals. Data regarding the weapon parameters selected by participants and information about the participants' usage of the framework is described in detail. This study serves as the groundwork for future research focused on finding a method for teaching junior game designers about game balance.

CHAPTER 1. INTRODUCTION

This research aims to develop a method for junior game designers to learn about game balance. This chapter serves as an introduction to the goals and significance of this research in addition to introducing its scope, assumptions, limitations, and delimitations.

1.1 Research Question

To what extent does this study's game balancing framework allow users to control the elements of game pace and symmetry to meet design scenario goals while maximizing creativity?

1.2 Statement of Problem

Game balancing is a complex, highly multivariant process that cannot be easily simplified to a single mathematical model. It requires a great deal of experience on the part of a game designer to know how changing parameters will impact the flow of a game. Currently, the primary method of gaining this experience is by working on projects in industry. Because of this, only experienced game designers are able work in this space.

1.3 Significance of Problem

Although the majority of games require some amount of balancing, there is a lack of formal instruction on the topic of game balance, even at universities that offer other game design coursework (Schreiber, 2016, 00:30). While introductory tools, such as Alice,[1] Kodu Game Lab,[2] and Scratch,[3] exist for other areas of game development, there is not currently a framework that allows junior game designers to learn about game balance. Tools such as those discussed by Vilaseca and Guardascione (2019), Lynn (2013), and Medler (2013) provide industry professionals with data visualizations. These tools provide graphs and heat maps based on data collected from either human or simulated playtesting. However, they do not provide a sandbox

[1] https://www.alice.org/
[2] https://www.kodugamelab.com/
[3] https://scratch.mit.edu/

environment in which game designers new to the industry can see the results of changing parameters in real time. Ensuring a game reaches a balanced state improves the player's overall enjoyment of the game (Schell, 2019, "Game Mechanics Must Be in Balance," para. 1). In recent years, many companies have adopted the games as a service model in which games receive regular updates with new content, often including balance changes (Mereu, Hudson, Nix, & Richards, 2013, p. 5). In order to avoid angering players with these balance changes, it is advantageous to test them *in silico* rather than *in situ*. Since balancing a game involves tuning a number of parameters that have an effect on the feel of the game, it may also help achieve desirable outcomes such as ensuring the game has the desired pacing, increasing the player's engagement and presence, and helping the player to reach a flow state.

1.4 Statement of Purpose

This research seeks to create a simulation-based, faciliatory framework that game designers can use to sharpen their insights for balancing multiplayer first-person shooters.

1.5 Scope

Junior designers can utilize the framework developed through this research to learn how changing parameters influences weapon balance in first-person shooter games.

1.6 Hypothesis

This study's game balancing framework will allow users to control the elements of game pace and symmetry to meet design scenario goals while maximizing creativity.

1.7 Assumptions

The assumptions of this research are:

1. The AI used in the simulations are sufficient to provide results that are similar to the results of human playtesting.
2. There are no systematic differences based on gender, race, or age.

1.8 Limitations

The limitations of this research are:

1. The insights provided by the framework are restricted to first-person shooter “deathmatch” games.
2. Preprogrammed bot behavior was utilized for the framework. Rewriting the AI’s behavior is beyond the scope of the project.
3. The AIs in the simulation cannot switch weapons on their own. This allows the user to always have full control over which weapons are used in combat.
4. Participants were recruited from game development courses at Purdue University. Although they all have some experience with game development, not all participants have a primary focus in game design.

1.9 Delimitations

The delimitations of this research are:

1. This research only targets the quantitative aspect of weapon balance and does not consider aspects that can only be identified through human testing such as weapon difficulty or fun.
2. The framework provides data to designers but does not provide a solution. The designer must analyze the data and reach their own conclusions.
3. Although the framework could be used to achieve various goals, this study focused exclusively on pacing and balance.
4. Although this approach could be applied to other areas of game balance such as maps, the framework developed during this research will only apply to weapons.

CHAPTER 2. REVIEW OF LITERATURE

This chapter introduces some general game design goals before delving into the concept of game balance and discussing its importance. After examining the general concept of game balance, the specifics of weapon balance are explored. This chapter also describes different methods used to balance games, including both traditional methods and more recent automated methods.

2.1 Game Design Goals

2.1.1 Pace

The desired pacing of a game depends upon that game's design goals. While the designers of a first-person shooter game will likely seek to create a fast-paced environment, the designers of a life simulation game, such as those in the *Animal Crossing* series, will aim for a much slower pace. Keeping in mind the desired pace for a game can help designers make design decisions. Davies (2009) examines the concept of game pacing as it applies to levels in-depth, but many of these same concepts can also be applied when designing weapons. He began by defining four elements that determine the pace of a game. The first of these is known as the movement impetus, which is what drives the player to move forward through the level. The choices made by a designer can either speed up or slow down the player's progress (p. 1). In a multiplayer first-person shooter game, the player's primary drive is dictated by the goal of the game mode they are currently playing, whether that goal is getting kills, defending an objective, or escorting a payload. While a player may act more passively when playing defensive game mode, a game mode focused on getting kills would typically result in a more aggressive playstyle and faster-paced gameplay. The second element defined by Davies (2009) is the sense of danger felt by the player. Typically, players who feel a greater sense of threat will move at a faster pace through a level, while those who feel there is a low level of threat will take a more leisurely approach (p. 1). In the case of multiplayer first-person shooter games, the amount of damage dealt by a weapon could impact the sense of threat felt by the player. If weapons deal a large amount of damage and can quickly kill the player, they will identify that weapon as a high-level threat and react accordingly. The next element defined by Davies (2009) is a sense of tension, which is caused by danger being perceived by the player. This element depends on the player identifying a threat but not knowing exactly when or where it will appear (p. 2). Although this is most applicable to atmospheric games, it still has some applications in multiplayer first-person shooter games. The knowledge that a sniper or flanker may appear from an unexpected location can introduce a sense of tension to a game. The final element defined by Davies (2009) is known as tempo. This describes the "level of intensity of action – how much concentration is required by the player to achieve their goal" (p. 2). While some games seek to achieve fast tempos, others aim to achieve a slower tempo. Tuning game mechanics such as player movement can allow the designer to achieve the desired tempo (p. 2). This can be expanded to weapon parameters such as fire rate or reload time that help to dictate the tempo of combat.

2.1.2 Engagement

Fullerton (2014) examined the concept of player engagement and concluded that a variety of elements can be responsible for creating an engaging experience for the player, and that different players are attracted to different game elements ("Engaging the Player," paras. 1-2). The first of these elements is challenge. Striking a balance between ensuring a player is consistently challenged as they progress through a game and preventing the game from becoming too difficult allows players to remain engaged throughout the experience ("Challenge," para. 1). The next game element that contributes to engagement is play. Games provide players with systems in which they can play. Different games feature different types of play ranging from structured competitive environments to story-driven fantasy environments ("Play," paras. 1-2). Different types of play appeal to different types of players, so identifying a target audience and shaping the type of play accordingly is essential. The third element is the game's premise. A game's premise gives context to everything that happens within the game and helps the player become emotionally invested ("Premise," para. 2). The next game element is character, which serves as the player's connection to the game world. The player interfaces with a game by controlling their character. In story-driven games, this typically inserts them into the story, but in all games, it gives them a vessel that fits within the structure of the game world ("Character," para. 1). The final element is story, which engages the player emotionally with the world (“Story,” para. 1). While not every game includes all of these elements, the inclusion of at least some of these ensures that the player remains engaged with the game as they play.

2.1.3 Presence

Presence is defined by Lombard and Ditton (2006) as “the perceptual illusion of nonmediation” (“Presence Explicated,” para. 1). In order to reach a state of presence, the consumer of a medium must be so immersed in that medium that they no longer perceive the media they are consuming as a medium, instead perceiving it as the real world. This however does not mean that the user cannot identify it as a medium when asked (“Presence Explicated,” paras. 1, 4). Gackenbach and Rosie (2011) examined elements of video games that are utilized to increase the sense of presence felt by the player. While graphics are one of these, there are many design aspects of a game that are just as important. Ensuring a game is designed in such a way that it remains interesting, challenging, fun, and engaging for the player can lead them to experience a greater sense of presence (p. 99).

2.1.4 Flow State

According to Csikszentmihalyi (1990), flow state occurs as a result of optimal experiences, which are experiences that result from the achievement of goals that a person has the skills to accomplish. These goals should be neither too easy or too difficult for the person, ideally their current skill level will match the goal exactly (p. 1). Optimal experiences produce feelings of pleasure and enjoyment. Csikszentmihalyi (1990) broke down this enjoyment into the following seven components:

1. Tasks with a reasonable chance of completion
2. Clear goals
3. Immediate feedback
4. Deep but effortless involvement that removes from awareness the frustrations and worries of everyday life.
5. Sense of control over our actions
6. No concern for the self

7. Alteration of the concept of time, hours can pass in minutes and minutes can look like hours. (pp. 2-3)

Ensuring the presence of these components in a game will help the player have an enjoyable experience, and therefore achieve a flow state. Jennett et al. (2008) explored the concept of flow as it relates to games. While they determined that immersion is a requirement of achieving a flow state while playing a game, a sense of immersion does not on its own mean that the player is in a flow state (pp. 642-643).

Sweetser and Wyeth (2005) devised a model known as GameFlow. This model is used specifically for measuring enjoyment in video games and is based upon the concept of flow. It relies on the evaluation of eight elements: concentration, challenge, player skills, control, clear goals, feedback, immersion, and social interaction. Concentration determines whether the player is presented with an appropriate number of stimuli. These stimuli need to be able to capture the player's attention without needlessly distracting them from other, potentially more important, in-game tasks. Challenge assesses whether the game's level of difficulty is appropriate for the player's skill level and whether it scales as the player's skill increases. The player skills element measures how easy and interesting the game is to learn. It also evaluates whether the game has functionality built in to assist struggling players. Control evaluates the level of control the player feels they have over the game, including components such as their character, the interface, and the in-game world. The clear goals element measures how clearly the player's goals are presented to them. Feedback measures the timeliness and frequency of the game's feedback system. Immersion determines how involved and invested in the game the player feels. Finally, social interaction measures the level of player interaction supported by the game, both in-game and externally (pp. 19-21). Keeping these elements in mind allows designers to shape game experiences in a way that allows players to achieve a flow state.

2.2 Game Balance Overview

Balance is an essential element of a well-made game (Schell, 2019, "Game Mechanics Must Be in Balance," para. 1). Unfortunately, it is a state that is difficult to achieve, in part because there is not always a single correct answer. Burgun (2011) discussed the concept of balance in depth. Although some games do lend themselves to a straightforward, mathematical method of balancing, for the majority, this is not an option. Compounding the issue is the fact that there is no exact method of measuring how balanced a game is. Balance is a very fluid concept that exists on a spectrum ranging from imbalanced to perfectly balanced, and there are no exact boundaries on the spectrum that determine where imbalance ends and balance begins. In addition, the concept of balance is subjective, and while designers may agree upon the state of balance of a game element that lies on either extreme of the balance spectrum, disagreements arise when the element lies somewhere in the middle (p. 1).

There are three types of imbalance as described by Elias, Garfield, and Gutschera (2012). The least severe of these occurs when a gameplay element is too weak. This type of imbalance only affects the weak game element, likely resulting in it not being used by players. Although having any type of imbalance is not ideal, having a weak gameplay element is the least problematic since it only impacts a single aspect of the game. The second type of imbalance occurs when a gameplay element is the strongest choice within its category. An example of this would be a certain ranged weapon that is more effective than other ranged weapons. While this type of imbalance will reduce the number of choices a player has within a single category, the player still has the option to select from different categories and the options they contain. The most severe type of imbalance occurs when a gameplay element is stronger than the majority of the choices offered to the player. This significantly reduces the number of gameplay elements that can be utilized by a player (pp. 132-133). Although difficult, ensuring that a game reaches and maintains a state of balance prevents a variety of unwanted consequences.

Morosan and Poli (2017) discussed the repercussions of an imbalanced game. When playing an imbalanced game, skilled players will typically find and utilize a single, optimal strategy to win. In this way, players create their own state of balance, different from the one intended by the developers. The imbalanced game effectively becomes balanced in this scenario since players that utilize the same strategy have an equal chance of winning. However, this often leaves entire game elements unused because they do not have a place in the optimal strategy. This indicates to the designer that a problem in game balance exists (pp. 377-378). Imbalance is most readily apparent in competitive games, where it also causes the biggest problems. In these games, it is essential that the skill of the player, rather than luck, is the primary decider of the game's outcome. These games need to be balanced in such a way that when two equally skilled players compete with one another, they each have an equal chance of winning (Volz, Rudolph, & Naujoks, 2016, p. 1). The rising popularity of esports causes this to be more important than ever due to the real-world stakes that are involved. In addition, the outcome of the game must feel fair not only to the players, but also to the audience. When a single strategy becomes dominant, professional players must adhere to that strategy or risk a high chance of a loss, which is frustrating to both players and audiences alike who grow bored of utilizing and watching the same strategies.

2.3 Weapon Balance

In any game centered around combat, weapons are the player's primary means of progressing through the game and interacting with the virtual world. Because of this, modifying weapon parameters allows designers to control player behavior. Drachen, Canossa, and Sørensen (2013) discussed how weapon range was used to control player behavior in the third-person shooter *Kane & Lynch: Dog Days.* Short-range weapons force the player to move forward through a level towards enemies, while long-range weapons encourage a less active playstyle. Because of this, short-range weapons tend to create more intense combat (pp. 293-294). By giving the player different weapons at different points within a game or level, the designer can shape the player's behavior to give them the intended experience.

Griesemer (2010) discussed the process of designing and balancing weapons in games. This process begins with a paper design that summarizes the weapon and describes its role in the game including components such as range, damage, and limitations (00:12:54). When deciding a weapon's role, the designer must strike a balance between simple and complex. If a weapon's role is too simple, the player will become bored, but if it is too complex, the player may become overwhelmed (00:16:51). When developing weapons, it is important to avoid strict dominance (00:20:10). If one weapon becomes more powerful than others or a certain weapon always dominates the others, the game becomes predictable. Burgun (2011) describes a method used by Robin Walker, the lead designer of *Team Fortress 2*, to prevent this issue. Classes in *Team Fortress 2* were designed to be defined by their weaknesses rather than their strengths. Ensuring that each class, and therefore their weapons, had a weakness means that there is always counterplay. This helps to prevent a single strategy from becoming dominant (Burgun, 2011, p. 3). Fullerton (2014) described using this same technique for balancing ("Dominant Objects," para. 1). Skilled players are able to understand the weaknesses of a weapon and learn how to play around them while making use of the weapon's strengths. Despite this, there are still cases in which certain classes or weapons become dominant in modern games.

Griesemer (2010) discussed an example of this occurring with the sniper rifle during the development of *Halo 3.* Through playtesting, it was discovered that all of the game's weapons felt weak in comparison to the sniper rifle. Optimizers, or playtesters who gravitate towards optimal weapons and strategies within a game, were exclusively using the sniper rifle. This led playtesters who chose to use other weapons to feel frustrated with the game. Although the role of a sniper rifle is to deal high damage for accurate shots at long range, players were finding success with the weapon even in close quarters combat where it should have been weakest. Because the sniper rifle was the strongest weapon in all situations, there were no options for counterplay. When deciding how to tune the weapon, the designers were careful not to reduce its effectiveness within its role or add additional weaknesses. Instead they chose to limit its strength by changing the fire rate from one shot every 0.5 seconds to one shot every 0.7 seconds (00:55:00). A decrease in fire rate allowed the weapon to keep its intended strengths while introducing more options for counterplay and increasing the consequences of missing a shot.

2.4 Methods of Balancing Games

2.4.1 Traditional Methods

Balancing is traditionally an iterative process that uses a combination of designer intuition, human playtesting, and mathematical modeling. Although mathematical modeling offers a method of achieving balance that provides a single, correct answer, in many situations it is not a viable option. Burgun (2011) discussed a case in which mathematical modeling would be ineffective in balancing game mechanics. In this example, a racing game was being developed in which each character has a special ability. If one character's special ability is an attack and another's is a speed boost, there is no mathematical way to compare the two abilities. All the designer can do is make their best guess for the parameters, then test and iterate on the design (p. 1). In a situation like this, playtesting must be utilized to evaluate game balance. Unfortunately, playtesting is costly and having multiple playtesters can introduce bias into the results (Volz, Rudolph, & Naujoks, 2016, pp. 1, 6). A playtester's evaluation of balance can differ significantly due to factors such as prior gaming experience, skill level, and personal preferences. When balancing multiplayer games, playtesters playing against others with different skill levels may be unable to accurately evaluate the game's balance due to the skill discrepancy.

When balancing a game, there are several elements the designer must consider. Jaffe, et al. (2012) described these in the form of seven questions to ask when determining a game's state of balance. The first, "How important is playing unpredictably?" seeks to identify how essential randomized strategies are to achieving success (p. 27). The second question, "To what extent must players react to the current state of the game?" determines whether players are locked into an action for long periods of time or if they have the freedom to adapt and react to their opponent (p. 27). The third question, "How powerful is a given action or combination of actions?" determines whether a single strategy, weapon, or playstyle is dominant (p. 27). The next question, "How much long-term strategy is necessary?" identifies how many steps ahead the player needs to be thinking and whether a long-term strategy is more effective than taking a game one turn at a time without considering what may happen in the future (p. 27). The next question, "Is the outcome known long before the game's end?" (p. 27) evaluates whether there is a point in the game at which the outcome becomes obvious, and whether the game is nearing its end at that point. Once the outcome becomes obvious, players begin to lose interest in the game and feel as though continuing is a waste of time (p. 27). The next question to consider is "What is the effect of avoiding certain end states?" (p. 27). This question is used to determine the viability of different win conditions (p. 27). The final question, "Are the starting conditions of the game fair?" asks whether one of the players begins the game with an advantage (p. 28). These questions can be utilized by both the designers and playtesters as the game's elements are tuned to achieve a state of balance.

The concept of mathematical modeling as applied to game balance is touched on, but not fully explored, in prominent game design literature. Schell (2019) discussed assigning numerical values to parameters in a hypothetical aerial dogfighting game to form a basic mathematical model for balancing the different airplanes in the game ("Biplane Battle"). Similarly, Fullerton (2014) emphasized the importance of spreadsheets for tracking and modeling game balance, but did not explore the concept in depth ("Spreadsheets"). Vilaseca and Guardascione (2019) examined the concept further through a discussion of the combat, progression, and chest systems in the cancelled mobile role-playing game *Hero.* The game followed a gear-based progression system in which equipping gear improves the player's stats, and as the player's level increases, the quality of the gear they can equip also increases (para. 6). The developers determined the impact these systems had on each other and the player through mathematical modeling, prototyping tools, and simulations (para. 12). The combat and progression systems work together to ensure that combat is balanced at all stages of the game. If the two systems do not scale accordingly, combat can be balanced at some stages, but imbalanced at others. In order to ensure the systems scaled accordingly, mathematical models were used. At the most basic level, a function similar to the following would be used:

$$N = \frac{H_E}{A_H}$$

Where N is the number of hits required to defeat an enemy, H_E is the enemy's health, and A_H is the amount of damage done by the player's attacks. When a progression system is introduced, this function becomes slightly more complicated:

$$N(x, y) = \frac{H_E(y)}{A_H(x)}$$

Where H_E is now a function of the enemy's level and A_H is a function of the player's level. This equation can be further complicated by additional elements commonly found in game combat systems such as critical hits, healing, and damage resistance ("Scaling of Combat: a formal approach," paras. 1-5, 10). Jaffe, et al. (2012) stated that while mathematical modeling can be used as an alternative to human testing, human playtesters are still required for evaluating some aspects of the game, such as those related to human psychology, the look and feel of the game, and the difficulty of performing certain actions (p. 30). While the need for human playtesting can be reduced through the introduction of more efficient methods for balancing, it will never be completely eliminated.

Vilaseca and Guardascione (2019) used the final mathematical model for *Hero* to develop a Python application that produced a visualization of the number of hits it takes a player to defeat or be defeated by an enemy. This visualization allowed the developers to see how changes in stats would affect combat at different points along the progression system ("Scaling of Combat: a formal approach," para. 11). Medler (2013) discussed a similar tool known as Data Cracker, which was developed to collect data about player behavior in *Dead Space 2* (p. 419). Different team members had different goals while using the tool, ranging from balancing gameplay mechanics to locating bugs (p. 426). The use of this tool is an example of a process known as data-driven design, a method in which data collected after each iteration of a product's design impacts its next iteration (Seufert, "Data-driven design," para. 2). The visualizations of data provided by Data Cracker helped developers identify and address undesirable patterns in player behavior (Medler, 2013, p. 419).

Lynn (2013) explored data visualization techniques used to assist game designers. The first of these techniques were heat maps, which show the distribution of data at different locations on a map. Level designers at Volition used these when working on *Red Faction: Armageddon* to determine areas on the map where players frequently died, ran out of ammo, and activated an in-game GPS. Based on this data, level designers could identify and address problem areas (pp. 501-504). Less complex visualizations were used for weapon data. Graphs were created to determine which weapons were most frequently used by players, as well as how many and what type of enemies they killed with their chosen weapon. Based on this data, designers were able to tune weapons so that none became the dominant choice (p. 505).

2.4.2 Automated Methods

More recently, there has been research into automated methods of game balancing and balance testing. Morosan and Poli (2017) utilized a genetic algorithm to make balance changes to two games: *Ms. PacMan* and *StarCraft*. The results of the study found that this method was effective in identifying potential balance solutions. The list of potential solutions produced by the algorithm is passed onto a designer, who is able to test the potential solutions, determine which most closely aligns with their game's goals, and make whatever changes are needed. Although the balancing process will most likely never be fully automated, this partial automation frees designers to work on deeper design problems instead of parameter tuning (p. 390).

Another example of automated game balancing is demonstrated in a study by Zook, Fruchter, and Riedl (2014). This study utilized active learning to tune spaceship control parameters (drag and thrust) within an arcade style shoot-'em-up game. After each wave of enemies, the player was asked to compare the most recent set of controls to the one they used in the previous wave. Based on their response, the controls were adjusted accordingly for the next wave. This active learning approach was compared to a random sampling approach. The results of the study indicated that the active learning method reduced the number of playtests required to determine optimal parameter settings. Utilizing active learning to tune parameters reduces the amount of playtesting required for a game (pp. 3-7). While this method did reduce the amount of playtesting required to balance the game, it still required human playtesters to evaluate each control iteration the algorithm developed.

Automated balance testing was explored by Karavolos, Liapis, and Yannakakis (2017), who conducted a study using machine learning to determine the state of balance of a first-person shooter. Although the study primarily focused on the balance of maps, weapon balance was also evaluated. The study consisted of teams of three AIs who participated in a team deathmatch contest. The first team to reach 20 kills was declared the winner of the game. For this study, a match was considered balanced if it either ended in a draw or with a marginal difference in kills between the two teams (p. 2). The results of the study showed that convolutional neural networks can make connections between level architecture and weapon parameters, while artificial neural networks do not perform much better than human perception. The model used in the study was able to select optimal weapons based on the design of a level. In addition, it could be used to balance existing weapons and create new weapons by changing the parameters of the weapons provided to it (p. 9).

The concept of automated weapon balancing was further explored by Gravina and Loiacono (2015). This study identified ten parameters used in weapon design. These parameters included rate of fire, spread, shot cost, life span, speed, damage, damage radius, gravity, explosive, and ammo (p. 3). By tuning these parameters, a variety of different weapons can be created. In this study, a genetic algorithm was used to tune parameters to create new weapons and balance existing ones in *Unreal Tournament III*. These weapons were first tested by bots, but after an initial evaluation, they were also playtested by human players to determine whether they were fun to use (pp. 3, 6). The results of the study found that the genetic algorithm was effective in creating weapons, however there were some issues where human players struggled to use weapons the bots found effective. In particular, there was a projectile weapon with a high gravity parameter with which players found it difficult to land shots. This issue was not apparent during bot testing due to the bots' ability to easily calculate their shot's trajectory (p. 7). This is an indication of the importance of human playtesting. While automated testing can be fairly accurate in determining the balance of game elements, there are certain things that bots cannot test for, such as the difficulty of a game mechanic. During testing, the bots focused on four main metrics. The first was balance, which was measured as the entropy of the kills distribution. If two weapons are balanced, each bot should have an equal chance of winning a duel regardless of the weapon they use. The second metric was effectiveness, which determined whether a weapon was capable of effectively killing an opponent. The third metric, safety, considered the amount of self-damage caused by a weapon and determines how likely it was to cause the user's death. The final metric considered the gameplay goals and examined statistics such as average hit distance, average hit time, and longest killstreak in order to determine whether they were in line with what the designer envisioned for the game (p. 3).

Although there are many examples of automated balance testing in academic research, the results of this research are rarely applicable to industry due to a lack of adaptability of the systems created by researchers and the complexity of these systems (Morosan & Poli, 2018, p. 263). Morosan and Poli (2018) sought to change this and worked with MindArk Sweden to apply an automated balancing system to their game *ComPet*. *ComPet* is a turn-based pet battling game similar to the *Pokémon* series (p. 264). An automated testing system was implemented where an AI would run through a gauntlet fighting a series of beasts. As the AI completed runs through the gauntlet, an evolutionary algorithm made changes to parameters that had been selected by a designer. Metrics for each run were reported to the designers so they could determine which set of parameters came closest to meeting their design goals. Designers found the data and suggestions provided by this system to be beneficial (pp. 264-269). While this implementation provided designers with tables of exact numerical data, the researchers suggested that for future implementations, a broader method of reporting data may be beneficial. Rather than reporting exact numbers, they suggested making more general comments such as indicating whether a change to a parameter was a slight or significant increase or decrease (p. 269). The method of reporting data in this study did lack a visual representation of data and chose to instead to report exact numerical values. The more general method of data reporting suggested by the researchers, though easier to understand, still lacks a visual component. Although there has been a variety of research into automating game balance, little of this has been applied to industry, so game designers must still rely heavily on their instincts. Unfortunately, these instincts can only be learned through experience, making it difficult for junior designers to approach game balance issues.

2.5 Conclusion

A framework intended to help game designers learn about game balance should reflect the iterative process of balancing games. Although other methods of game balancing such as mathematical modeling and automated systems are used, tuning parameters and observing the effect the changes have on the overall game is still a primary method that game designers use for balancing. There is rarely a single correct answer for balancing a game, so it is difficult for junior designers to learn about game balance through any means other than hands-on experience. Unfortunately, game balance is much more difficult to isolate than other aspects of game development such as art or programming, which makes it difficult to gain experience with balance unless a full game is made.

CHAPTER 3. METHODOLOGY

This chapter describes the methodology for this research. The features of the weapon balancing framework and the experimental design are described.

3.1 Research Type and Design

This research was a mixed methods study that examined the development and testing of a weapon balancing framework designed to teach junior game designers about balancing weapons in first-person shooter video games.

3.2 Instrumentation

The weapon balancing framework was developed for this study using Unreal Engine 4. The framework simulates how tuning weapon parameters affects gameplay flow and survivability in multiplayer first-person shooter deathmatch games. The user is presented with an interface that allows them to modify ten weapon parameters in real time. As two AIs duel each other, weapon performance metrics and game statistics are reported on screen.

3.2.1 Weapon Parameters

It was initially thought that three parameters: damage, clip size, and reload time, would be sufficient to define weapons, but after further consideration and research, it was found that these three parameters fell short in differentiating weapons from one another. In particular, many weapons are defined by their different rates of fire and bullet spread. A slightly modified version of the ten weapon parameters defined by Gravina and Loiacono (2015) were utilized in this study. Their original names and definitions were as follows:

1. Rate of Fire – Amount of ammo shot per second
2. Spread – Random variance in a bullet’s trajectory; causes bullets to spread out over large distances
3. Shot Cost – Amount of ammo required for a single shot
4. Life Span – Amount of time a bullet exists within the game

5. Speed – The bullet’s travel speed
6. Damage – Amount of damage dealt by each bullet
7. Damage Radius – Radius of the bullet’s hitbox
8. Gravity – Amount of gravity force applied to bullets
9. Explosive –Radius of an explosion caused by the bullet
10. Ammo – Amount of a weapon’s ammo that can be carried by the player (p. 3)

Two minor changes were made to these parameters for this study. First, damage radius was renamed bullet radius in order to make its meaning clearer to participants. Second, the definition of ammo was changed to indicate the number of bullets in a single magazine, which is the number of bullets that can be shot before reloading. This change was made due to the restrictions of the AIs used in the study. They did not have the functionality necessary to find and pick up ammo in the world, causing a stalemate to occur when they ran out of ammo.

3.2.2 Weapon Performance Metrics

As the two AIs duel each other, weapon performance metrics are reported on screen for each weapon. These include damage dealt per second (DPS) and time to kill (TTK). Of these, DPS is the most straightforward weapon performance metric. It can be used to quickly see a weapon’s damage output over time. For the purposes of this framework, TTK was defined as the time from spawn until death. TTK has a significant impact on the pacing of a game. Like most other aspects of game balancing, TTK does not have an ideal value and depends heavily on the game’s design goals.

3.2.3 Game Statistics

Several game statistics are displayed on screen as the simulation is run. These include the game’s current score, the number of kills per minute (KPM), the distance between the two combatants, and an indicator showing which of the two weapons currently has the advantage. These statistics have strong relationships with one another as well as with the weapon performance metrics.

3.2.4 Implementation Details

The weapon balancing framework extends upon the FPS Multiplayer Template,[4] an asset offered on the Unreal Engine Marketplace that contains basic first-person shooter deathmatch gameplay functionality and AIs. Building upon an existing template allowed the majority of development time to be focused on the weapon balancing framework itself, rather than basic gameplay functionality.

Although many modern first-person shooters offer unique and interesting weapon types, this research limits the weapons available to more traditional types that are common across the majority of first-person shooter games. The four weapon options offered by the framework are: pistols, shotguns, sniper rifles, and rocket launchers. The intention was to allow junior designers to gain experience balancing more common weapon types and to remove unnecessary complexity. As another method of reducing complexity, the user of the framework can assign weapons to the AIs, but the AIs cannot switch weapons on their own. This allows the user to have more control over the simulation and allows them to decide upon and analyze a desired weapon matchup without the added variable of weapon switching.

3.2.5 User Interface

The framework presents the user with an interface that offers a set of options on each side of the screen with the center of the screen displaying the combat simulation and game statistics. The interfaces on either side of the screen mirror each other, presenting the user with options and data for each of the AIs' weapons. Each interface offers a dropdown menu that allows the user to select the weapon used by the AI. Below the weapon selection dropdown is a series of sliders that can be used to tune the ten weapon parameters. There is an option to link or unlink each parameter with the opposing AI's matching parameter in order to keep their values the same. This gives the user the option to easily keep certain parameters the same across the two AIs' weapons. The randomize button randomizes all ten of the weapon's parameter values. There is also a button that allows the user to save the parameters as a preset so they can later be loaded and used again. At the bottom of the interface, weapon performance metrics (DPS and TTK) are reported and updated in real time as the simulation runs.

[4] https://www.unrealengine.com/marketplace/en-US/slug/fps-multiplayer-template

At the top of the screen between the two weapon parameter menus, the game statistics are reported in real time. These include KPM, score, distance between combatants, and an indicator showing the stronger weapon. Hovering over the name of any of the parameters, weapon performance metrics, or game statistics brings up a tooltip that provides a short explanation of the item. At the bottom of the screen, there is the option to play or pause the simulation or to run the simulation in real time, five times, ten times or twenty times speed. There is also a button to reset the statistics recorded by the framework. This allows the user to make large changes to weapons and instantly see the results of these changes without the previous settings influencing the averages. Finally, the spectate button allows the user to switch between spectating the red and blue AIs. The framework is shown in Figure 1.

Figure 1. Weapon Balancing Framework

3.3 Data Collection

To test the effectiveness of the framework, thirty participants were recruited to utilize the framework to achieve three design scenario goals related to pacing and balance. After a training module and three design goals were completed, participants were given a short post-study questionnaire.

3.3.1 Population

The target population for this study was junior game designers interested in learning about weapon balance.

3.3.2 Sample

Thirty participants were recruited from two different game development courses at Purdue University: CGT 255: Game Development II: Design and Psychology and CGT 390: Game Scripting and Coding.

3.3.3 Modified User Interface

Some modifications to the game balancing framework were made for testing purposes. The weapon selection dropdown menus were disabled so that all participants would utilize the same weapon matchup. As mentioned by Drachen, Canossa, and Sørensen (2013), the range of weapons can impact the intensity of combat, with short-range weapons potentially producing more intense combat than long-range weapons. This more intense experience is described as being "closer to the frontline" (pp. 293-294). However, due to the behavior of the AIs used in this simulation, the distance between them was not an accurate measure of intensity. Because of this, distance between combatants was removed from the interface and replaced with the user's target KPM value as well as the remaining time for achieving the current design goal. A continue button was added that sends the selected weapon parameters, weapon performance statistics, game statistics, time taken, number of clicks taken, and KPM over time to a CSV document for later analysis. Finally, a popup window was added that explains the current design goal. This modified user interface is shown in Figure 2.

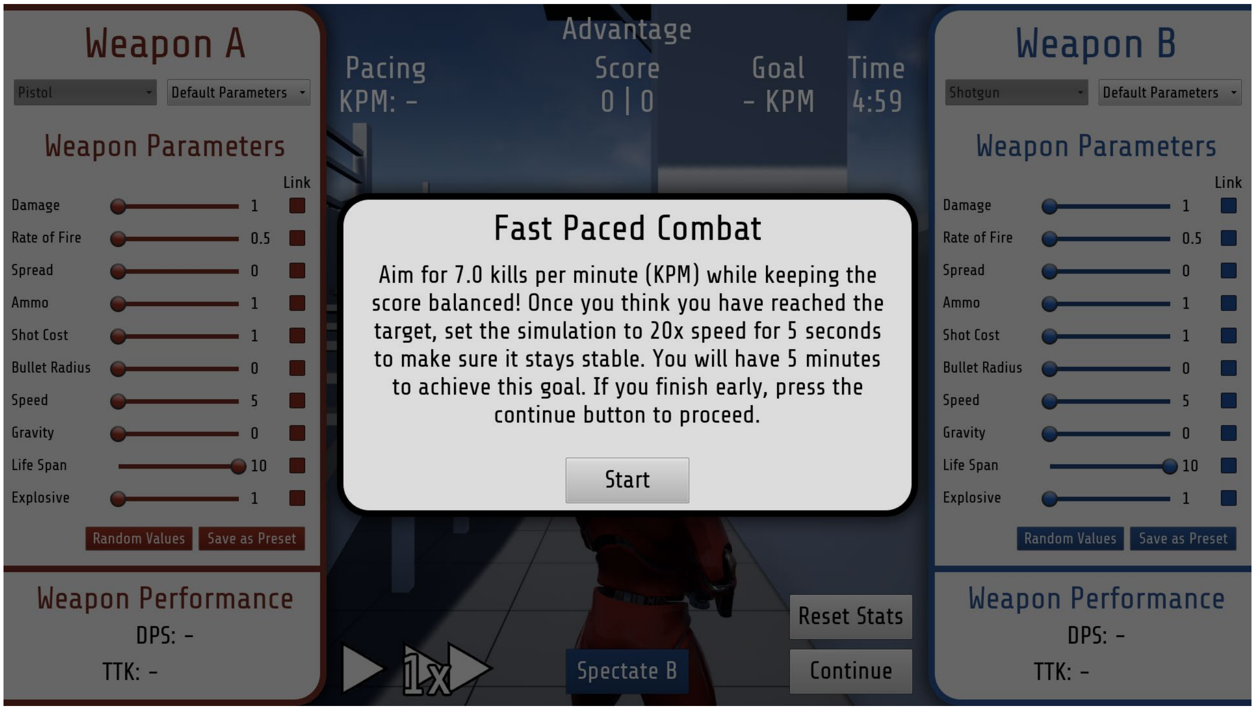


Figure 2. Modified Weapon Balancing Framework

3.3.4 Design Goals

Each participant was given the same three design goals that focused on pacing and balance. According to Burgun (2015) asymmetry occurs in games when two players begin a game with different abilities (para. 2). In games where players can select abilities, characters, or weapons, players typically assume that each of these choices has an equal chance of success, making each choice balanced despite their differences. In multiplayer first-person shooters, balanced asymmetry occurs when players using different weapons have an equal chance of winning a duel. For the testing of the weapon balancing framework, one of the AIs was assigned a pistol, while the other was assigned a shotgun. Participants were asked to tune parameters until each AI had an equal chance of winning the duel while also achieving a target KPM. To determine whether the AIs had an equal chance of winning a duel, participants were instructed to observe how close the game's score was. Participants were asked to achieve fast-, medium-, and slow-paced combat by reaching the goals of 7 KPM, 6 KPM, and 5 KPM respectively, while also keeping the scores balanced to achieve balanced asymmetry.

3.3.5 Testing Procedure

Participants who chose to participate in this study were first asked to read and sign a consent form describing the process. This form can be viewed in full in Appendix A. They then viewed a one-minute instructional video that explained how to use the framework and gave a brief description of their goals.[5] After viewing the video, they began using the tool. Their usage of the tool was divided into four stages, each with a five-minute time limit. At the start of each stage, each of the parameter values were reset to a default value. For all parameters except life span, this was the minimum value for the parameter. Life span was set to the maximum value because its lowest value caused bullets to be destroyed before they hit their target at long ranges. If the participant completed any of the stages early, they could use the continue button to move onto the next stage. The first stage was meant for training. Participants had the opportunity to use this time to experiment with the tool and learn about the parameters as well as gain familiarity with the interface. At the end of the training stage, the participant received their first design goal: fast-paced combat. Once they reached this goal, they received their next goal, medium-paced combat, and then their final goal, slow-paced combat. For each goal, the player was asked to achieve that goal's target KPM while keeping the score balanced. Once they reach the goal, they were asked to run the simulation at twenty times speed for five seconds to ensure the KPM and score remained stable over time. If they finished early, they could press the continue button to proceed, otherwise their information would be submitted automatically after their time ran out. Once they completed all three design goals, they were asked to fill out a short, qualitative questionnaire that asked three questions. These questions were as follows:

1. Which aspects of the framework worked well?
2. Which aspects of the framework worked poorly?
3. Do you have any suggestions for improving the framework?

The post-study questionnaire can be viewed in Appendix B.

[5] https://www.youtube.com/watch?v=L80nNzzZcqA

3.4 Conclusion

This chapter provided a description of the weapon balancing framework developed for this study in addition to the testing procedure utilized. Weapon parameters, weapon performance metrics, and game statistics that were used and collected for the study were described in depth.

CHAPTER 4. RESULTS

This chapter provides analysis and visualizations of the quantitative and qualitative data collected during the study.

4.1 KPM and Balance Goals

During data analysis, each participant's data was broken down into three separate datasets, one for each goal. Each of these datasets was assigned to one of five groups. The four primary groups were as follows: Met Goal and Balanced, Met Goal and Imbalanced, Did Not Meet Goal and Balanced, and Did Not Meet Goal and Imbalanced. Datasets that were assigned to one of the two Met Goal groups had a submitted KPM value within 0.5 of that goal's target KPM value. Datasets that were assigned to one of the two Balanced groups had a balance percentage of 15 percent or less. The formula used to determine the balance percentage was as follows:

$$\%\,Balance = \frac{|Red\ Score - Blue\ Score|}{Red\ Score + Blue\ Score}$$

Datasets submitted with zero clicks, zero KPM, or red or blue scores of zero were considered invalid and assigned to the fifth group. If the participant submitted data with zero clicks, it indicated that the participant skipped that stage of the study without interacting with the parameter sliders. Participants who notified the researcher of this during the study were given the opportunity to restart the stage they skipped, however, in most cases, the researcher was not notified until the participant had completed the study. Data with zero KPM or a score of zero indicates either that the stage was skipped or that the participant pressed the reset stats button immediately before submitting their parameters. Datasets in this group were not considered during analysis.

For each of the three goals, a histogram was constructed to show the distribution of submitted KPM values. In addition, for each goal a table presents a summary for each of the four categories of data.

4.1.1 Fast-Paced Combat

Fast-paced combat was the first goal participants were asked to achieve. It asked participants to achieve 7 KPM while keeping the score balanced. For analysis, any participant with a KPM value between 6.5 and 7.5 was considered to have met the goal. Four participants submitted data with zero clicks, zero KPM, or a score of zero. Their data was not included in analysis. A histogram of the submitted KPM values for all 30 participants is shown in Figure 3.

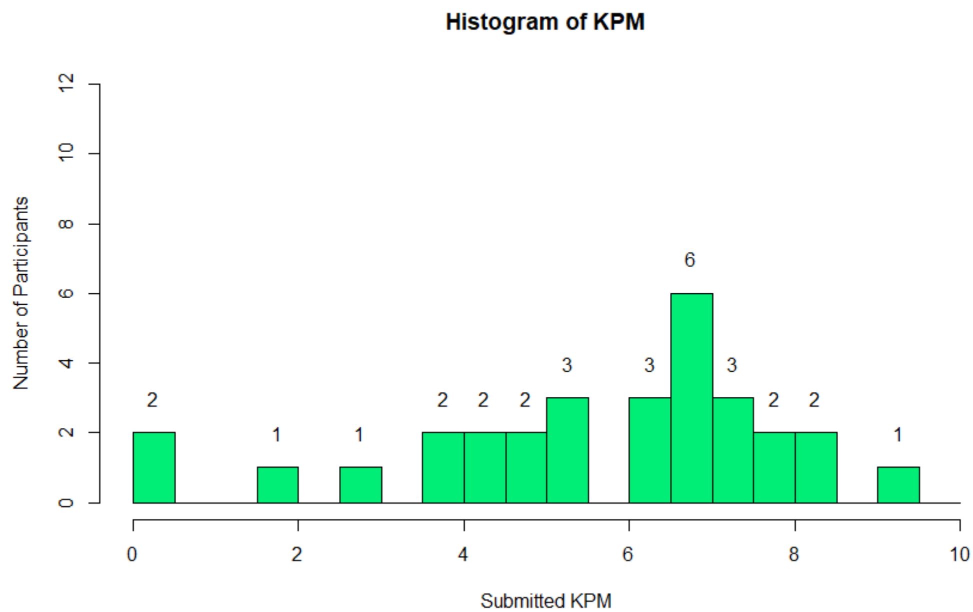


Figure 3. Histogram of Submitted KPM Values for Fast-Paced Combat

Table 1 presents a summary of data for each of the four categories of participant data for the fast-paced combat goal. The number of participants column indicates the number of participants whose data was assigned to that category. The average time column indicates the average amount of time in seconds that participants took to submit their parameters. There was a time limit of five minutes or 300 seconds, after which their parameters would be submitted automatically. The average clicks column indicates the average number of times that participants clicked on parameter sliders to change parameter values before submitting. Each time a participant clicked on a slider, the current KPM of the simulation was also recorded. These KPM values were used to determine whether the participant reached the KPM goal at any point during the stage. The number of participants who reached the KPM goal at some point during the stage are indicated in the met goal at least once column.

Table 1. Summary of Fast-Paced Combat Data by Category

Category	Number of Participants	Average Time (sec)	Average Clicks	Met Goal at Least Once
Met Goal & Balanced	3	286	28.7	3
Met Goal & Imbalanced	6	251	48.2	6
Did Not Meet Goal & Balanced	7	300	27.7	3
Did Not Meet Goal & Imbalanced	10	291	51.8	5
Total	26	283	41.8	17

4.1.2 Medium-Paced Combat

Medium-paced combat was the second goal participants were asked to meet. It asked participants to achieve 6 KPM while keeping the scores balanced. For analysis, any participant with a KPM value between 5.5 and 6.5 was considered to have met the goal. One participant submitted data with zero KPM. Their data was not included in analysis. A histogram of the submitted KPM values for all 30 participants is shown in Figure 4.

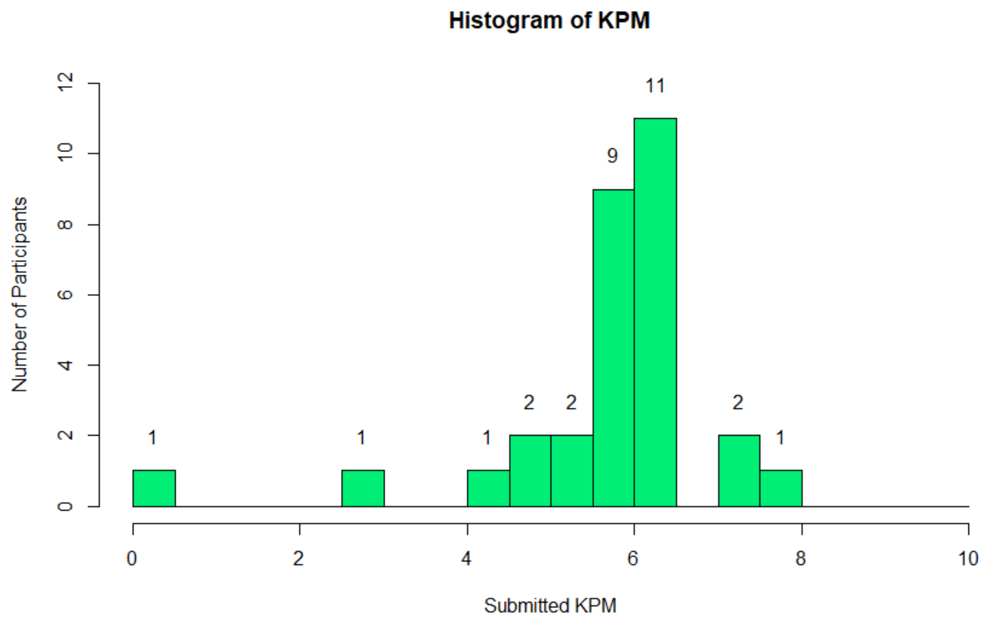


Figure 4. Histogram of Submitted KPM Values for Medium-Paced Combat

Table 2 presents a summary of data for each of the four categories of participant data for the medium-paced combat goal.

Table 2. Summary of Medium-Paced Combat Data by Category

Category	Number of Participants	Average Time (sec)	Average Clicks	Met Goal at Least Once
Met Goal & Balanced	12	210	23.7	12
Met Goal & Imbalanced	8	231	33.9	8
Did Not Meet Goal & Balanced	6	286	35.7	4
Did Not Meet Goal & Imbalanced	3	258	51.3	1
Total	29	236	31.8	25

4.1.3 Slow-Paced Combat

Slow-paced combat was the third and final goal participants were asked to meet. It asked participants to achieve 5 KPM while keeping the scores balanced. For analysis, any participant with a KPM value between 4.5 and 5.5 was considered to have met the goal. Two participants submitted data with zero KPM. Their data was not included in analysis. A histogram of the submitted KPM values for all 30 participants is shown in Figure 5.

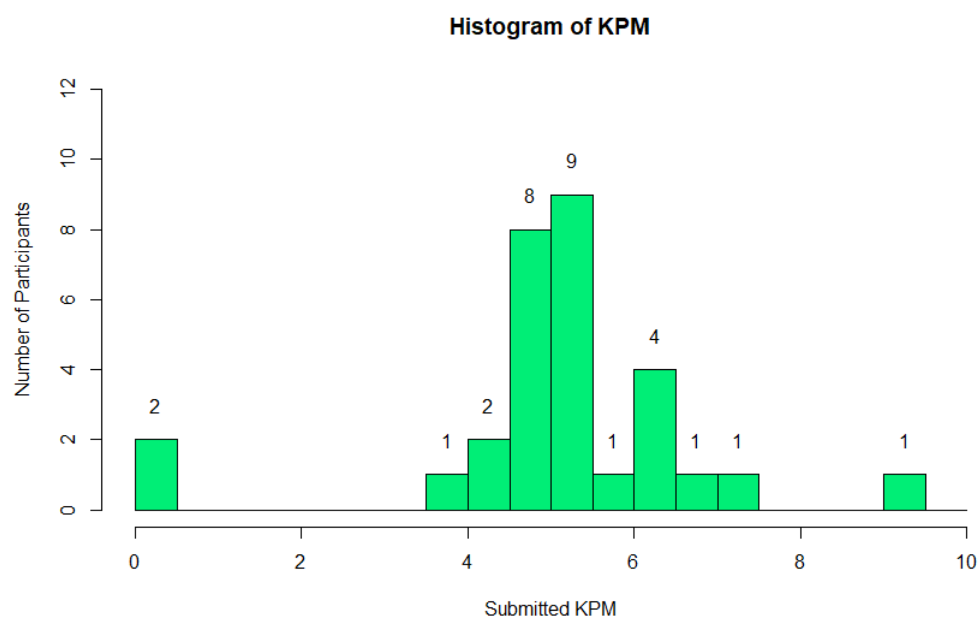


Figure 5. Histogram of Submitted KPM Values for Slow-Paced Combat

Table 3 presents a summary of data for each of the four categories of participant data for the slow-paced combat goal.

Table 3. Summary of Slow-Paced Combat Data by Category

Category	Number of Participants	Average Time (sec)	Average Clicks	Met Goal at Least Once
Met Goal & Balanced	10	206	24.1	10
Met Goal & Imbalanced	7	144	20.1	7
Did Not Meet Goal & Balanced	6	272	31.8	6
Did Not Meet Goal & Imbalanced	5	300	49.4	5
Total	28	222	29.3	28

4.1.4 Analysis

When examining the number of participants assigned to each category, it is important to note that a large amount of information was presented to participants through both the training video and the interface of the weapon balancing framework. While the KPM goal was indicated on screen for the entire duration of each stage, the reminder to keep the game balanced was only shown in the popup window at the start of each stage. This could have caused some participants to forget about the balance goal and focus exclusively on reaching the KPM goal, which could have led some participants to submit parameters that exclusively met the KPM goal.

As shown by the histograms and tables for each goal, the number of participants that met the medium- and slow-paced combat goals was a significant increase over the number of participants that met the fast-paced combat goal. Since the fast-paced combat goal was the first goal assigned to all participants, it is possible that this indicates that participants became more comfortable using the framework after completing their first goal or that they gained a better understanding of how the parameters impacted the simulation. In addition, the majority of participants reached the fast-, medium-, and slow-paced combat goals at least once during the duration of their respective stages. Since a comparatively smaller portion of participants submitted parameters that met the goal while also maintaining a balanced state, this could suggest that many participants first tried to meet the KPM goal and then attempted to balance the weapons, causing their KPM value to deviate. It could also indicate that these participants struggled to understand how to achieve a state of balance.

For each of the pacing goals, the average number of clicks for participants in the Met Goal and Balanced category was one of the lowest. For fast-paced combat, participants in the Met Goal and Balanced and Did Not Meet Goal and Balanced categories had a similar number of average clicks, while the number of clicks for the other two categories were significantly higher. For medium-paced combat, participants in the Met Goal and Balanced category had the lowest number of clicks on average, while all three other categories were significantly higher. For slow-paced combat, participants in the Met Goal and Balanced and Met Goal and Imbalanced categories had the lowest number of average clicks while the other two categories were significantly higher. In addition, participants in the Did Not Meet Goal and Imbalanced category had the highest number of average clicks across all three stages. This may indicate that users who met both the KPM and balance goals had a better understanding of what changes needed to be made than those who met only one or neither of these goals. As the participants progressed through the stages, the overall average number of clicks decreased, with the first stage having the highest average and the final stage having the lowest average. This, along with the increased number of participants achieving the second and third goals, may suggest that participants were able to gain a better understanding of the framework by using it for longer periods of time.

4.2 Creativity

When considering game design and balance, it is important to note that there is rarely a single correct answer to a problem. While it is possible to balance a first-person shooter deathmatch game by giving both players a single, identical weapon, this does not make for interesting or engaging gameplay. The intention of assigning a pistol to one of the AIs and a shotgun to the other was to encourage creativity in the parameter values selected by participants. For the purposes of this study, creativity is defined as a difference in parameter values between the two weapons.

4.2.1 Individual Parameter Values

Although the weapon balancing tool presents users with ten parameters, the explosive parameter did not have an effect on the weapons used in this simulation because it only applies to the rocket launcher. This was noted in a tooltip describing the parameter that could be viewed by hovering over its name. Although some participants did edit this parameter, its value was disregarded during analysis. The ranges for each of the nine parameters analyzed in this section are presented in Table 4.

Table 4. Parameter Value Ranges

Parameter	Range
Damage	1 – 100
Rate of Fire	0.5 – 100
Spread	0 – 100
Ammo	1 – 300
Shot Cost	1 – 15
Bullet Radius	0 – 50
Speed	5 – 100
Gravity	0 – 250
Life Span	0.1 – 10

In order to determine the amount of creativity displayed by participants during the study when tuning individual parameters to achieve design goals, the parameter values that participants submitted for each weapon were analyzed. For the purposes of analysis, parameters that are considered creative have at least a 15 percent difference between the medians or interquartile ranges for the two weapons. A boxplot visualizing a creative parameter can be viewed in Figure 6, while a boxplot visualizing a parameter that is not considered creative can be viewed in Figure 7. Boxplots of all nine parameters for each of the three KPM goals can be viewed in Appendix C.

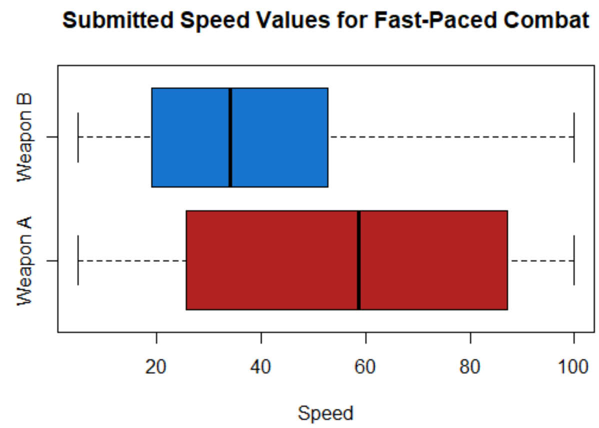


Figure 6. Boxplot for Speed, a Creative Parameter

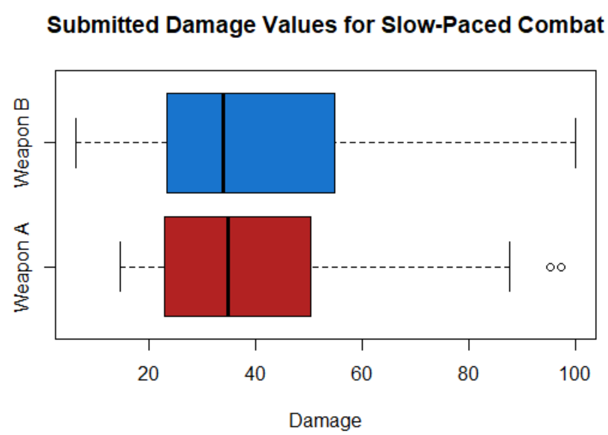


Figure 7. Boxplot for Damage, Not a Creative Parameter

It is important to note that all valid datapoints were considered when determining the creativity of parameters, not just those that met the balance or design goals. Table 5 categorizes each of the parameters as creative or not creative for each design goal.

Table 5. Parameters Categorized by Creativity

Fast		Medium		Slow	
Creative	Not Creative	Creative	Not Creative	Creative	Not Creative
Rate of Fire	Damage	Rate of Fire	Damage	Rate of Fire	Damage
Spread	Gravity	Spread	Ammo	Ammo	Spread
Ammo			Shot Cost	Speed	Shot Cost
Shot Cost			Bullet Radius		Bullet Radius
Bullet Radius			Speed		Gravity
Speed			Gravity		Life Span
Life Span			Life Span		

Rate of fire was the only parameter that remained creative across all three design goals, while damage and gravity remained not creative across all three goals. It is interesting to note that a much larger number of parameters were considered creative for the fast-paced combat goal. By viewing the parameter boxplots in Appendix C, it can be observed that a large number of participants left some of the not creative parameters at or near their default values during the medium- and slow-paced combat goals. This may suggest that participants identified which parameters had the greatest impact on KPM during the fast-paced combat stage, and then primarily edited those parameters in later stages. A decrease in parameter variability as participants progress through the three stages of design goals can be observed in the three radial plots in Figure 8. Parameters that are not creative could be viewed as most essential for achieving the KPM and balance goals since their values are similar across the two weapons, while creative parameters are less essential and allow for a lot more variance. Unfortunately, due to the relatively small number of participants and many participants leaving nonessential parameters at their default values, nothing conclusive can be drawn from this table or plots.

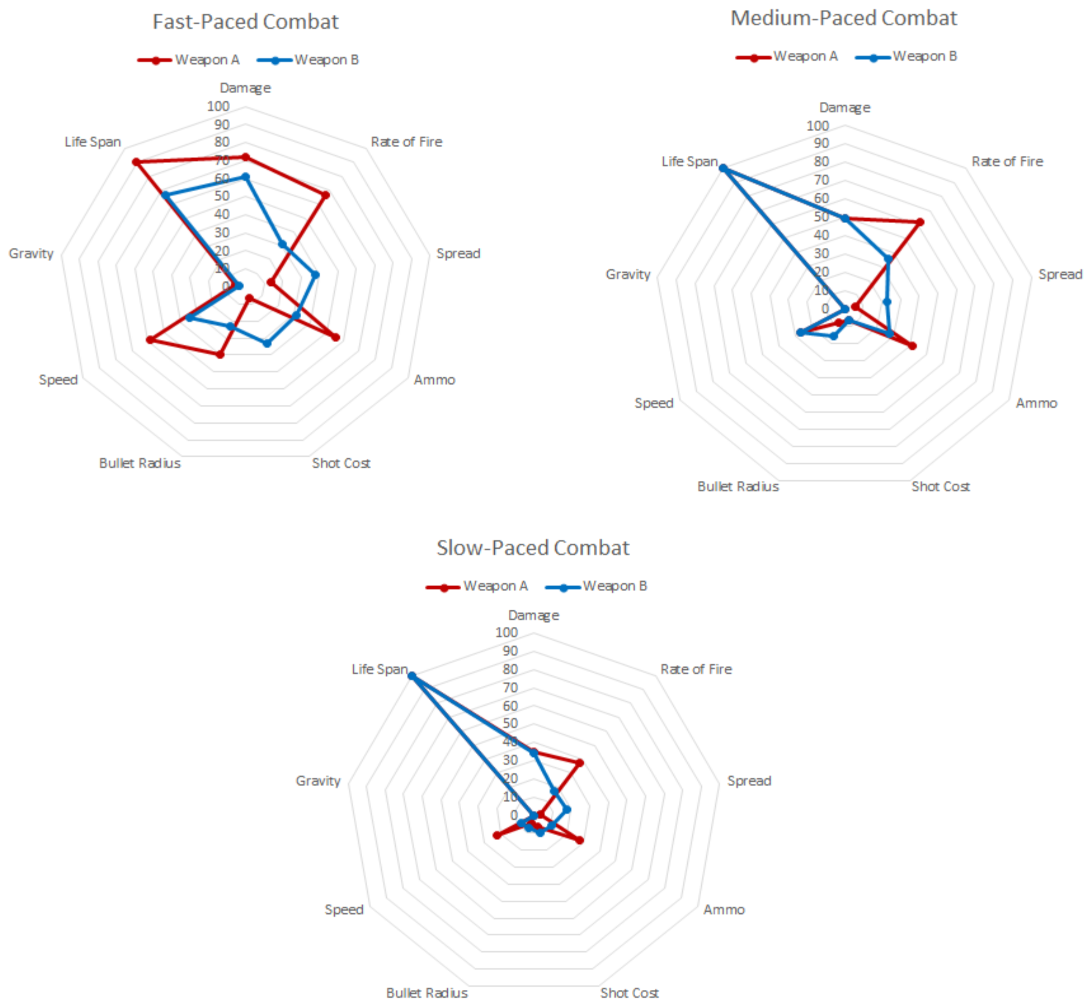


Figure 8. Radial Plots for Median Parameter Values of Fast-, Medium-, and Slow-Paced Combat

4.3 Questionnaire Responses and Observations

4.3.1 Positive Feedback

The first question on the post-study questionnaire asked participants to describe which aspects of the framework worked well. Many participants expressed that the user interface was easy to use and understand despite the large number of controls present. Many of the features of the framework were praised such as the ability to speed up the simulation, randomize parameter values, link parameter values, and reset the statistics within the simulation.

4.3.2 Suggestions for Improvement

The second and third questions on the post-study questionnaire asked participants to describe which aspects of the framework worked poorly and requested they provide suggestions for improvement. The majority of their responses to these two questions can be divided into two categories: user interface and AI.

4.3.2.1 User Interface

Many participants stated that the were unsure of the exact function of some of the parameters. Although the training video explained that tooltips for each of the parameters could be viewed by hovering over the parameter's name, one participant suggested adding tooltips to explain the function of each parameter. It is unclear whether the participants that suggested this change were similarly unaware of the existing tooltips or whether more extensive descriptions of the parameters were necessary. Another common suggestion was the addition of a field that allows the user to type in exact parameter values rather than exclusively using the sliders. This would allow users to more easily control parameter values and make small adjustments.

4.3.2.2 AI

There were a few bugs present with the AI that participants would occasionally experience. While these bugs were not game breaking, they were inconvenient, and many participants mentioned them on their questionnaires. The first of these caused the two AIs to spend a significant amount of time trying to find each other. Even when the simulation was running at twenty times speed, it could occasionally take up to twenty seconds for the AIs to find each other. Although this was not a constant problem, many of the participants experienced this at least once over the course of the study. Another issue happened when the two AIs were dueling at very close ranges. Sometimes at this range they would be unable to deal damage to one another. This could generally be fixed by pressing the randomize value to randomize the weapon parameters. This suggests that it was an issue with the parameter values selected by the participant rather than an issue with the AI. It is possible that a large spread value, high gravity value, or short life span value prevented the AIs from hitting one another.

Many participants also suggested more general improvements to the AIs such as making them behave more like human players by allowing them to strafe or by modifying their behavior based on the type of weapon they were using. One participant mentioned that the AI with the shotgun was attempting to snipe its opponent, which is not a behavior that shotguns are typically designed for because they are expected to be used at close range.

4.3.3 Other Feedback and Observations

When observing participants while they completed the study, it was noted that the majority of participants seemed comfortable using the weapon balancing framework with only minor questions being directed at the researcher during the course of the study. However, one participant was noted struggling with the framework and attempting to utilize keyboard controls to move the in-game characters, even after viewing the training video. After the researcher intervened to explain that the characters were controlled by AIs and only the on-screen UI elements could be utilized to balance the weapons, the participant continued to attempt to control the characters with keyboard controls. This seemed to be an outlier, as no other participants were observed attempting to control the on-screen characters.

Although all participants in the study were recruited from game development courses, it is important to note that not all participants were familiar with games in the first-person shooter genre, and not all participants were game designers. The courses that participants were recruited from contained students that were primarily focused on a variety of different game development disciplines including art and programming in addition game design. One participant requested the addition of starting parameter values for guns so that users who are not familiar with how different types of guns work have an example of what values are expected for each gun type. This response suggested that the participant was not familiar with the first-person shooter genre of video games, since shotguns and pistols are common weapons that are featured in a variety of games in that genre. Another participant expressed that they had difficulty understanding how to change the parameters to make the game balanced. While this could indicate that a more extensive tutorial is needed for the framework, it could also indicate that the user had not had much game design or first-person shooter experience.

Participants also noted that changing some parameters had more impact than changing others. This was by design, as parameters such as damage and spread will influence the KPM much more than bullet radius or life span. Although the intention was to give users as much control over the design of the weapons in the simulation as possible, one participant noted that they would primarily use the parameters that had the greatest impact on KPM in order to reach the goal. Since participants were focused on achieving KPM and balance goals, they likely did not see a reason to tune parameters that may influence the feel of a weapon such as speed or gravity. These two parameters can influence the trajectory and speed of the projectile that is being shot, and while tuning them could increase or decrease the difficulty of hitting a target with the weapon, human playtesters would be needed to see the full impact of these changes since AIs are able to be accurate regardless of how difficult a weapon is to aim.

4.4 Conclusion

This chapter provided descriptions, visualizations, and analysis of the data collected during the study. Observations regarding the participants' ability to meet pacing and balance design goals were discussed and the creativity of the parameter values selected for each weapon was explored. In addition, qualitative data from the post-study questionnaire was described and analyzed.

CHAPTER 5. DISCUSSION

This chapter summarizes this research and provides suggestions for future work, such as improvements to the weapon balancing framework and other possible applications.

5.1 Discussion

5.1.1 Summary

This research documented the development and testing of a framework used to teach junior game designers how to balance weapons in first-person shooter deathmatch games. The testing served as a pilot study for the weapon balancing framework. Due to the small sample size and other limitations of this research, nothing conclusive can be drawn from the results, however, some interesting trends can be noted. The majority of the participants seemed to understand the framework's interface and praised its features. Since the number of participants that met their assigned goal increased significantly after the first goal, it is possible that the framework served its intended purpose of increasing the user's understanding of weapon balance, however, further research would need to be conducted to reach more definite conclusions.

This research and the framework developed by it could serve as the foundation for future research in the area of educating junior game designers about game balance. By expanding upon this research, methods and curricula could be developed for teaching both game design students and junior game designers in industry about game balance. As mentioned previously, this would prove quite valuable since balance is essential to well-made games but is rarely present in game development curricula.

5.1.2 Future Work

5.1.2.1 Improvements to the Framework

The following are a list of improvements that could be made for future iterations of the weapon balancing framework, collected from both participant feedback on the post-study questionnaire and the researcher's observations.

- The addition of an input field for typing in numerical parameter values would allow for more precise control when tuning parameters, especially for some of the parameters that offer a large range of values.
- An interactive tutorial within the framework would allow users to gain a better understanding of the features of the tool and the exact purpose of each parameter.
- A variety of improvements could be made to the AIs in order to make them behave more like human players. Behaviors such as strafing and using cover could be implemented and the AI's behavior could change according to the type of weapon equipped. For example, an AI using a shotgun should prefer close range combat, while an AI using a sniper rifle should prefer long range. There were also a few minor bugs present in the current AI implementation that could be fixed.
- Adding more visualizations, such as bullet trails, would help the user to be able to visualize how the changes they make affect the simulation.

After making improvements to the framework, further testing could be conducted with both junior and senior game designers serving as participants in order to determine industry professional's thoughts on the framework. While the design goals used in this study focused exclusively on KPM and score, future testing could include more varied design goals that explore some of the other metrics tracked by the framework such as distance, TTK, and DPS.

5.1.2.2 Other Applications

While the weapon balancing framework developed for this study was designed exclusively for balancing weapons in deathmatch modes of first-person shooter games, future research could focus on developing similar tools for team deathmatch or objective-based game modes, such as capture the flag. Similar tools could be developed for other game genres, such as third-person shooters or fighting games. Developing a similar tool for balancing maps in first-person shooters may also yield interesting results.

5.2 Conclusion

This chapter provided a discussion of the research. A summary of the findings and their applications was presented, as well as suggestions for future work. Improvements to the weapon balancing framework developed for this study were explored as well as other applications for similar frameworks that could be developed in the future.

APPENDIX A. CONSENT FORM

RESEARCH PARTICIPANT CONSENT FORM

The Development of a Framework for Weapon Balancing in Multiplayer First-Person Shooter Games

Dr. David Whittinghill
Department of Computer Graphics Technology
Purdue University

Key Information

Please take time to review this information carefully. This is a research study. Your participation in this study is voluntary which means that you may choose not to participate at any time without penalty or loss of benefits to which you are otherwise entitled. You may ask questions to the researchers about the study whenever you would like. If you decide to take part in the study, you will be asked to sign this form, be sure you understand what you will do and any possible risks or benefits.

This study seeks to develop a framework for balancing weapons in multiplayer first-person shooter video games. The framework will allow the user to tune parameters to achieve desired design outcomes. This will give junior game designers a method of gaining experience balancing games, a valuable skill that is traditionally only gained through industry experience.

This research project will take place over the course of five months.

What is the purpose of this study?

Individuals who participate in this study will provide valuable feedback about the framework through quantitative data collected by the framework and qualitative data collected through the post-study questionnaire.

We would like to enroll 30 people in this study.

What will I do if I choose to be in this study?

Participants in this study will be asked to achieve up to five design scenario goals using the weapon balancing framework. This design goal will be related to the game's pacing, intensity, symmetry, or some combination of these. The time it takes the participant to achieve their goal and their selected parameter values will be collected. The game statistics and weapon performance metrics produced by the selected parameter values will also be collected. Once the participant has achieved their assigned design goals, they will complete a short post-study questionnaire.

How long will I be in the study?

Participation in this study will consist of a single session that will take 20 minutes or less to complete.

What are the possible risks or discomforts?

Participation in this study will include viewing of on-screen video game violence similar to that seen in popular modern-day first-person shooter games such as *Call of Duty* or *Counter-Strike*. Participation in this study is strictly voluntary, and if you feel uncomfortable at any time, you may opt-out of the study without penalty.

The risks of participating in this study are no greater than the participant would encounter in daily life or during the performance of routine physical or psychological exams or tests.

Breach of confidentiality is always a risk with data, but we will take precautions to minimize this risk as described in the confidentiality section.

Are there any potential benefits?

There are no anticipated direct benefits to participants.

If the weapon balancing framework is determined to be effective, it will provide junior game designers with a method for gaining experience with game balance, which is an important area of game development. The framework may also serve as the foundation for future projects aiming to provide designers with a method of learning how to balance other aspects of games, such as maps.

Are there costs to me for participation?

There are no anticipated costs to participate in this research.

This section provides more information about the study

Will information about me and my participation be kept confidential?

The project's research records may be reviewed by departments at Purdue University responsible for regulatory and research oversight.

No individually identifiable information will be collected from participants.

What are my rights if I take part in this study?

You do not have to participate in this research project. If you agree to participate, you may withdraw your participation at any time without penalty.

Upon completion of the participant's session, they will no longer be able to withdraw their data from the study since a lack of individually identifiable information makes it impossible to identify and remove a participant's data.

Who can I contact if I have questions about the study?

If you have questions, comments or concerns about this research project, you can talk to one of the researchers. Please contact Dr. David Whittinghill by phone (765) 494-1353 or by email (dmwhittinghill@purdue.edu) or Carly Fox by email (fox54@purdue.edu).

To report anonymously via Purdue's Hotline see www.purdue.edu/hotline

If you have questions about your rights while taking part in the study or have concerns about the treatment of research participants, please call the Human Research Protection Program at (765) 494-5942, email (irb@purdue.edu) or write to:

Human Research Protection Program - Purdue University
Ernest C. Young Hall, Room 1032
155 S. Grant St.
West Lafayette, IN 47907-2114

Documentation of Informed Consent

I have had the opportunity to read this consent form and have the research study explained. I have had the opportunity to ask questions about the research study, and my questions have been answered. I am prepared to participate in the research study described above. I will be offered a copy of this consent form after I sign it.

______________________________________ ________________________

Participant's Signature Date

____

Participant's Name

______________________________________ ________________________

Researcher's Signature Date

APPENDIX B. SURVEY

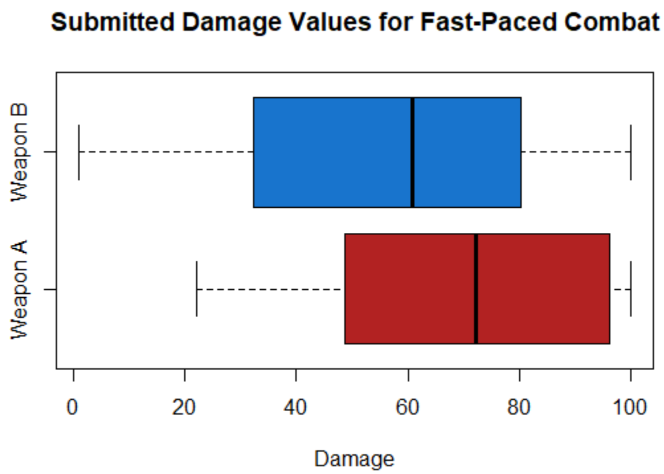
Post-Study Questionnaire

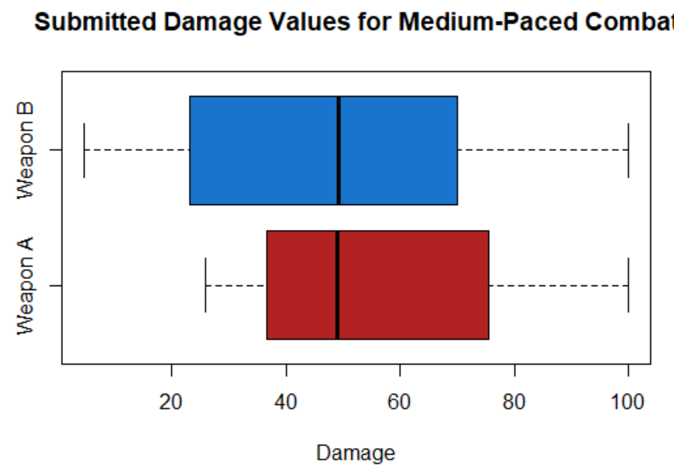
1. Which aspects of the framework worked well?

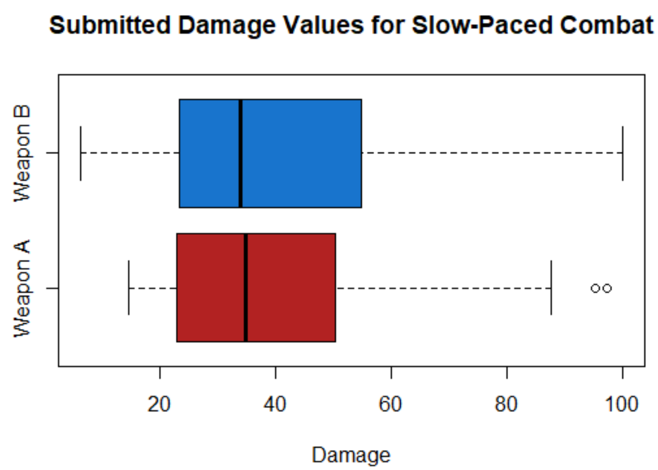
2. Which aspects of the framework worked poorly?

3. Do you have any suggestions for improving the framework?

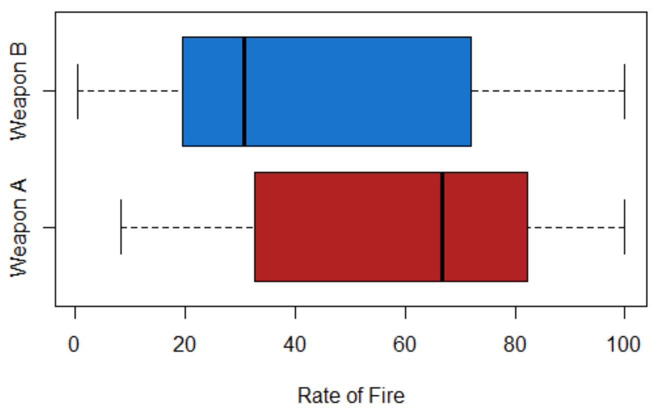
APPENDIX C. PARAMETER BOXPLOTS



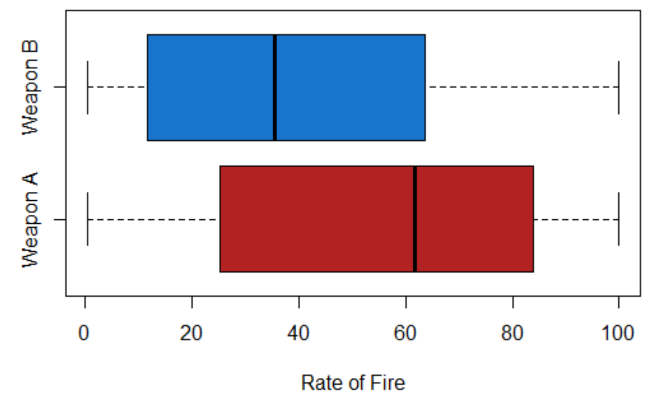




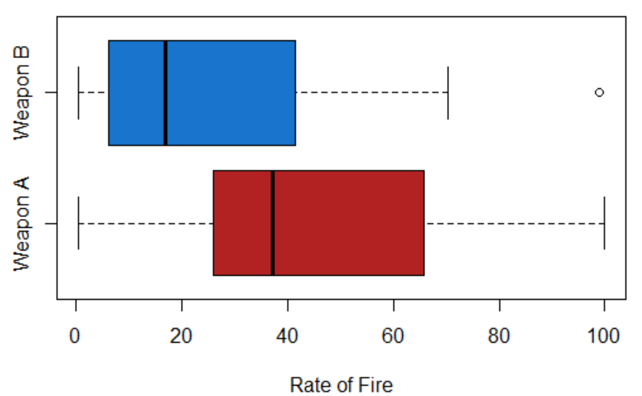
Submitted Rate of Fire Values for Fast-Paced Combat



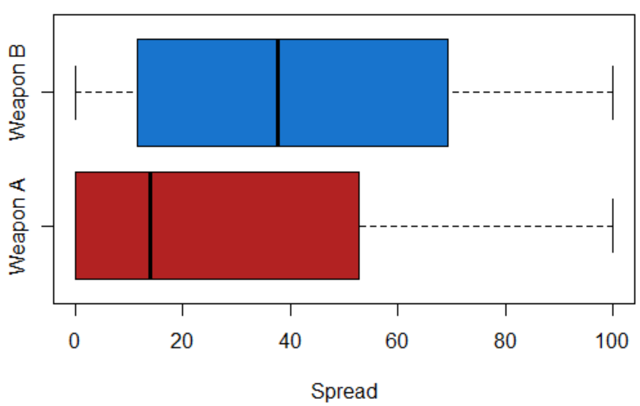
Submitted Rate of Fire Values for Medium-Paced Combat



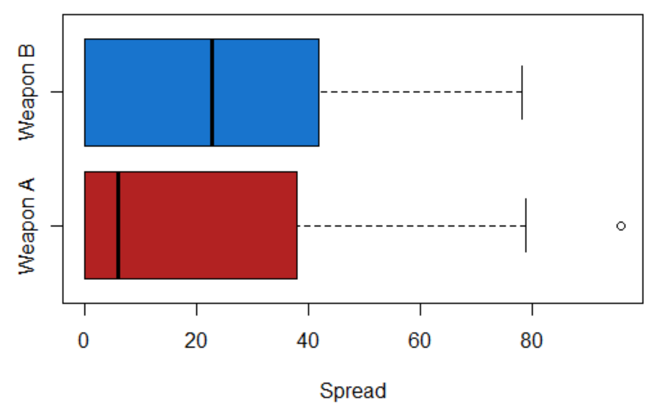
Submitted Rate of Fire Values for Slow-Paced Combat



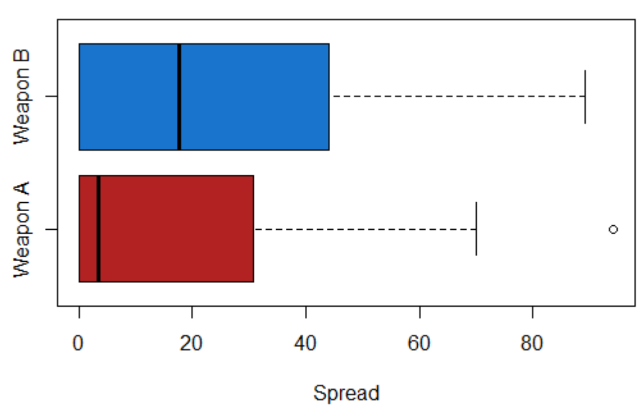
Submitted Spread Values for Fast-Paced Combat

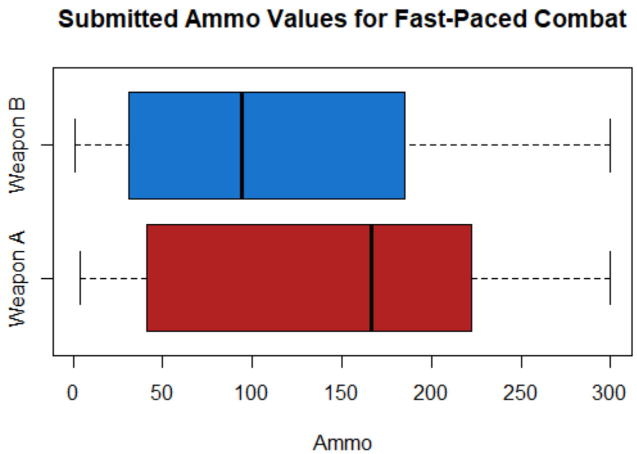


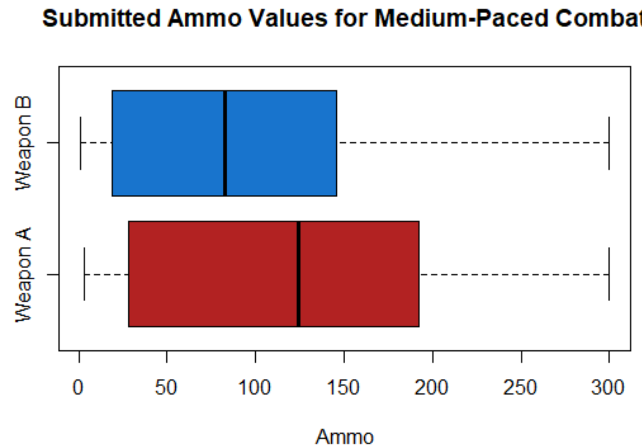
Submitted Spread Values for Medium-Paced Combat

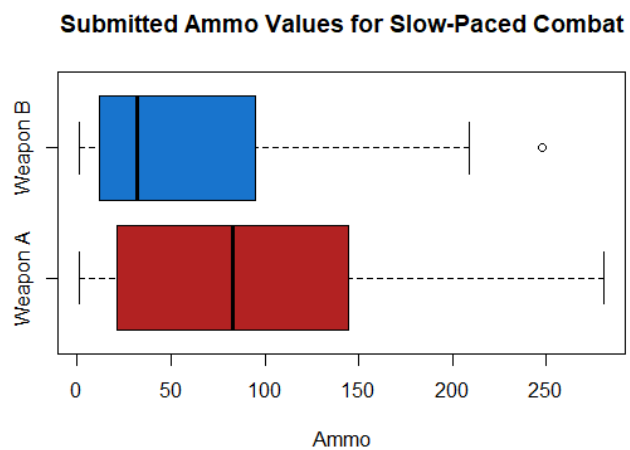


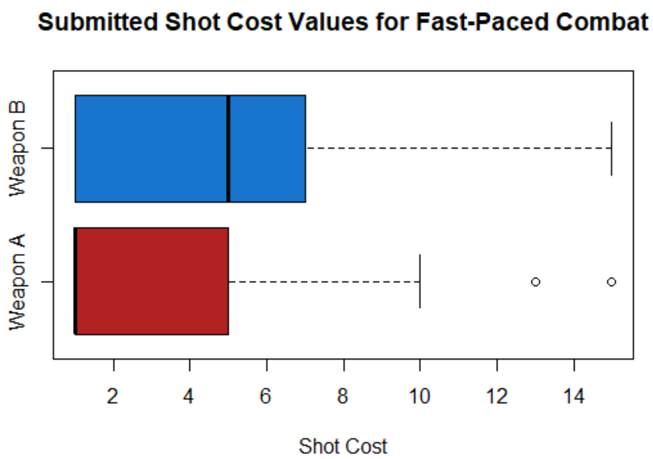
Submitted Spread Values for Slow-Paced Combat

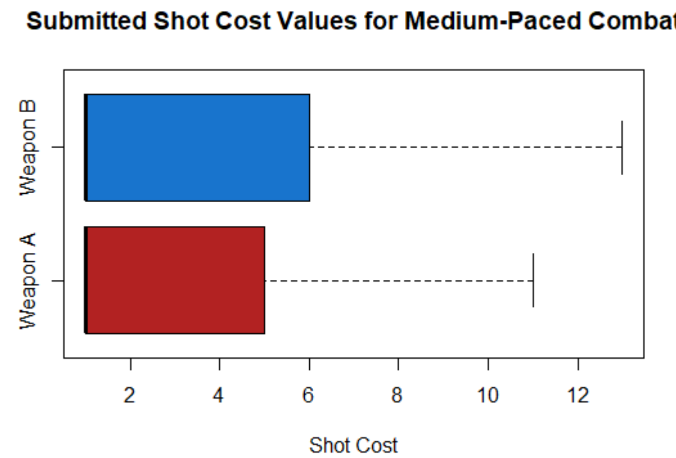


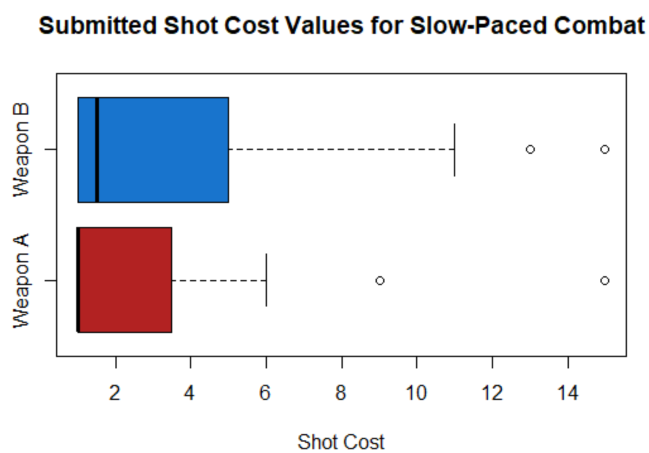

Submitted Ammo Values for Fast-Paced Combat
Weapon B
Weapon A
0
50
100
150
200
250
300
Ammo

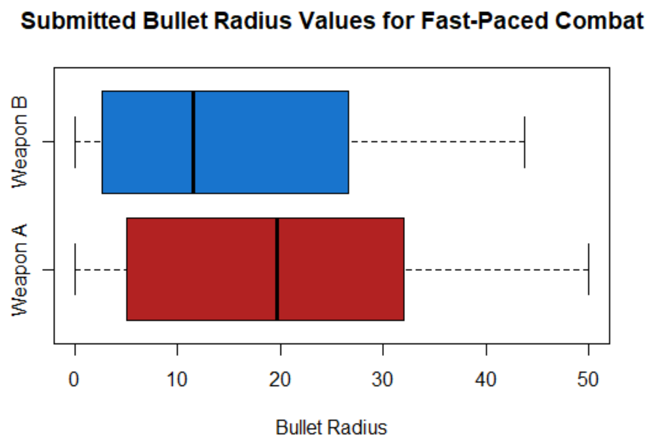

Submitted Ammo Values for Medium-Paced Combat
Weapon B
Weapon A
0
50
100
150
200
250
300
Ammo

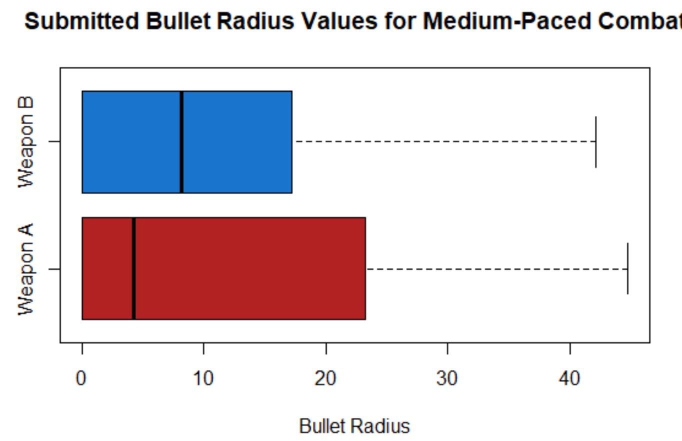

Submitted Ammo Values for Slow-Paced Combat
Weapon B
Weapon A
0
50
100
150
200
250
Ammo

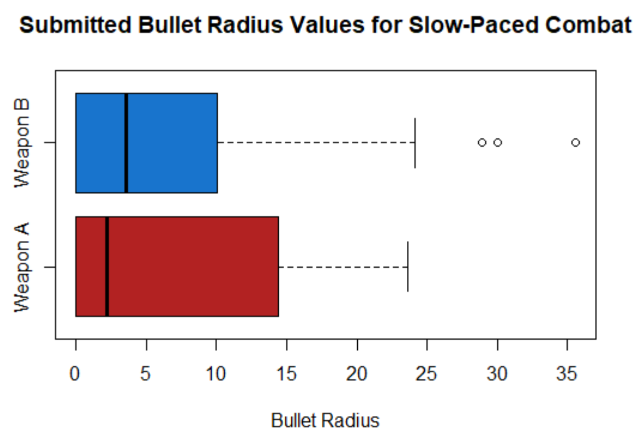

Submitted Shot Cost Values for Fast-Paced Combat
Weapon B
Weapon A
2 4 6 8 10 12 14
Shot Cost

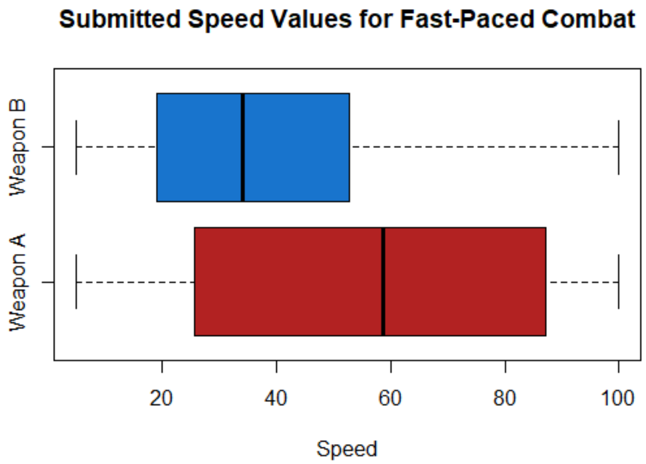

Submitted Shot Cost Values for Medium-Paced Combat
Weapon B
Weapon A
2 4 6 8 10 12
Shot Cost

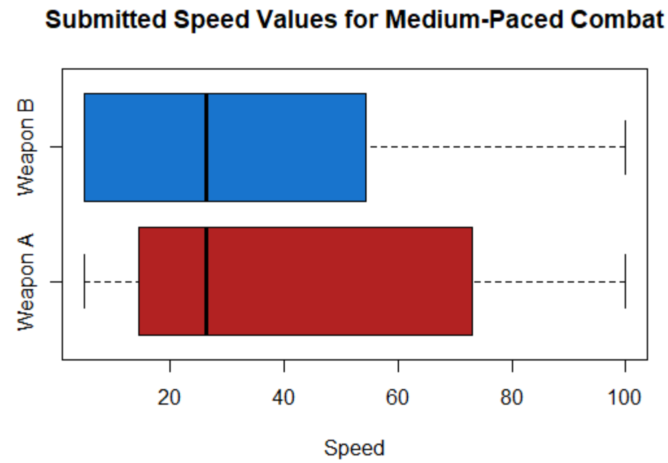

Submitted Shot Cost Values for Slow-Paced Combat
Weapon B
Weapon A
2 4 6 8 10 12 14
Shot Cost

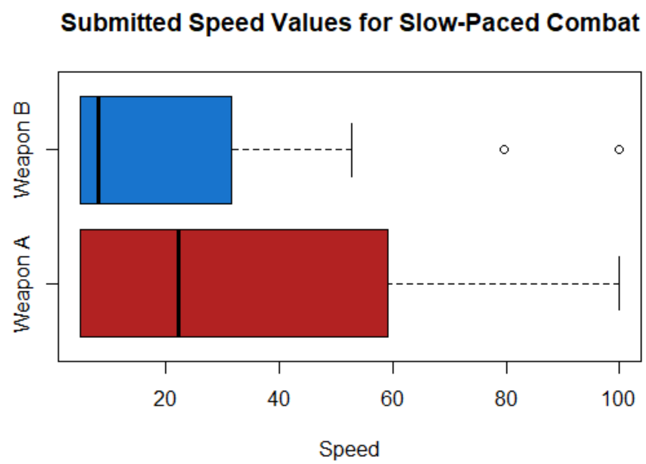

Submitted Bullet Radius Values for Fast-Paced Combat
Weapon B
Weapon A
0
10
20
30
40
50
Bullet Radius

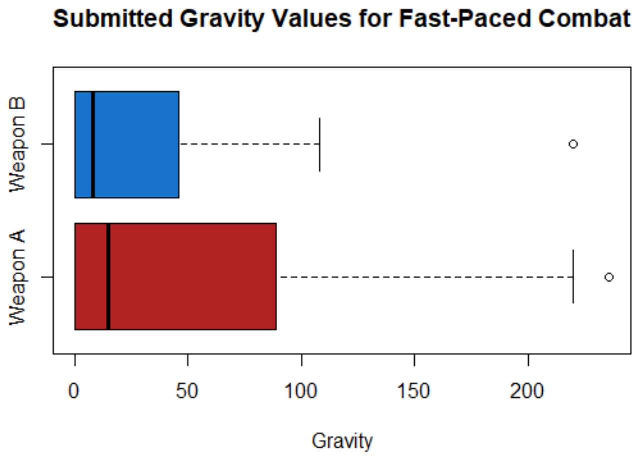

Submitted Bullet Radius Values for Medium-Paced Combat
Weapon B
Weapon A
0
10
20
30
40
Bullet Radius

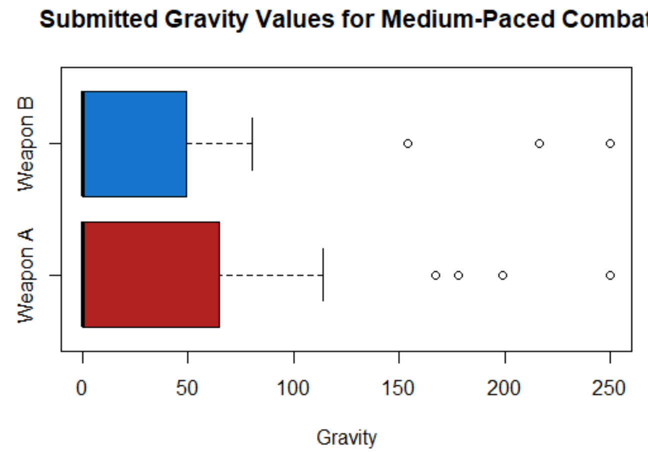

Submitted Bullet Radius Values for Slow-Paced Combat
Weapon B
Weapon A
0
5
10
15
20
25
30
35
Bullet Radius

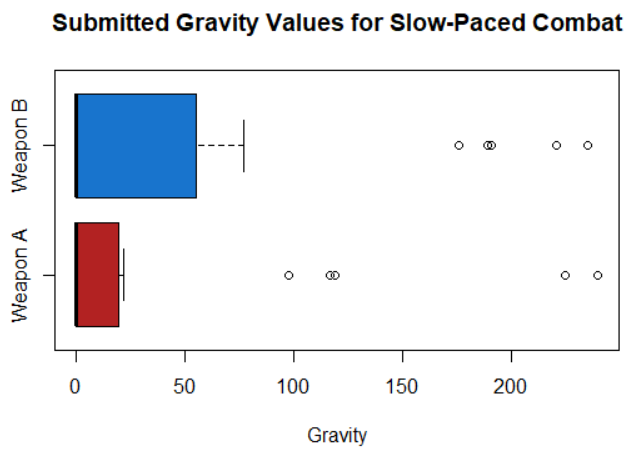

Submitted Speed Values for Fast-Paced Combat
Weapon B
Weapon A
20
40
60
80
100
Speed


Submitted Speed Values for Medium-Paced Combat
Weapon B
Weapon A
20
40
60
80
100
Speed

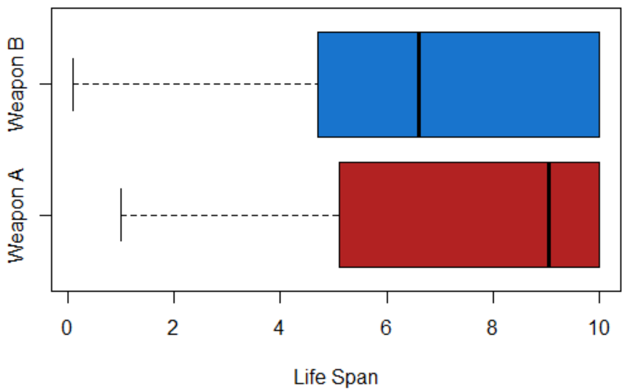

Submitted Speed Values for Slow-Paced Combat
Weapon B
Weapon A
20
40
60
80
100
Speed


Submitted Gravity Values for Fast-Paced Combat
Weapon B
Weapon A
0
50
100
150
200
Gravity

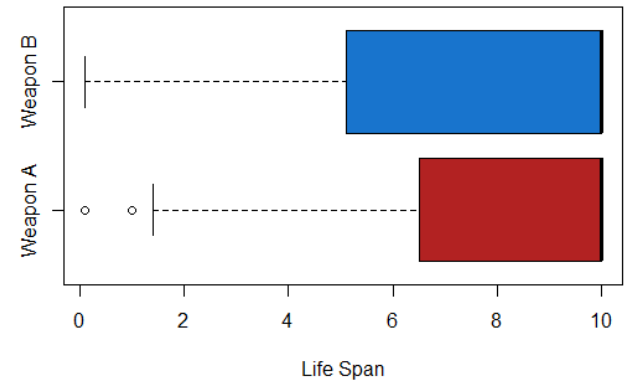

Submitted Gravity Values for Medium-Paced Combat
Weapon B
Weapon A
0
50
100
150
200
250
Gravity


Submitted Gravity Values for Slow-Paced Combat
Weapon B
Weapon A
0
50
100
150
200
Gravity

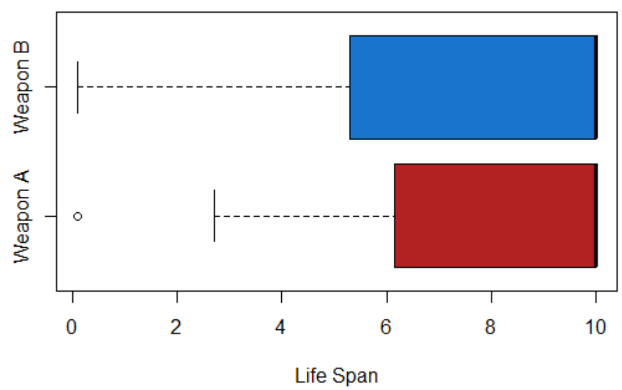
Submitted Life Span Values for Fast-Paced Combat



Submitted Life Span Values for Medium-Paced Combat



Submitted Life Span Values for Slow-Paced Combat



REFERENCES

Burgun, K. (2011). Understanding Balance in Video Games. Retrieved from https://www.gamasutra.com/view/feature/134768/understanding_balance_in_video_.php

Burgun, K. (2015). Asymmetry in Games. Retrieved from https://www.gamasutra.com/blogs/KeithBurgun/20151001/255058/Asymmetry_in_Games.php

Csikszentmihalyi, M. (1990). Flow: The Psychology of Optimal Experience (pp. 1–8). New York: Harper & Row. Retrieved from https://www.researchgate.net/publication/224927532_Flow_The_Psychology_of_Optimal_Experience

Davies, M. (2009). Examining Game Pace: How Single-Player Levels Tick. Retrieved from https://www.gamasutra.com/view/feature/132415/examining_game_pace_how_.php

Drachen, A., Canossa, A., & Sørensen, J. R. M. (2013). Gameplay Metrics in Game User Research: Examples from the Trenches. In M. S. El-Nasr, A. Drachen, & A. Canossa (Eds.), *Game Analytics: Maximizing the Value of Player Data* (pp. 285–319). London: Springer-Verlag. https://doi.org/10.1007/978-1-4471-4769-5

Elias, G. S., Garfield, R., & Gutschera, K. R. (2012). *Characteristics of Games*. Cambridge: The MIT Press.

Fullerton, T. (2014). *Game Design Workshop* (3rd ed.). Boca Raton: A K Peters/CRC Press.

Gackenbach, J., & Rosie, M. (2011). Presence in video game play and nighttime dreams: An empirical inquiry. *International Journal of Dream Research*, *4*(2), 98–109.

Gravina, D., & Loiacono, D. (2015). Procedural Weapons Generation for Unreal Tournament III. In *2015 IEEE Games Entertainment Media Conference (GEM)* (pp. 1–8). IEEE. https://doi.org/10.1109/GEM.2015.7377225

Griesemer, J. (2010). Design in Detail: Changing the Time Between Shots for the Sniper Rifle from 0.5 to 0.7 Seconds for Halo 3. In *Game Developers Conference*.

Jaffe, A., Miller, A., Andersen, E., Liu, Y.-E., Karlin, A., & Popović, Z. (2012). Evaluating Competitive Game Balance with Restricted Play. In *The Eighth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* (pp. 26–31).

Jennett, C., Cox, A. L., Cairns, P., Dhoparee, S., Epps, A., Tijs, T., & Walton, A. (2008). Measuring and defining the experience of immersion in games. *International Journal of Human-Computer Studies*, *66*, 641–661. https://doi.org/10.1016/j.ijhcs.2008.04.004

Karavolos, D., Liapis, A., & Yannakakis, G. (2017). Learning the Patterns of Balance in a Multi-Player Shooter Game. In *Foundations of Digital Games 2017*. https://doi.org/10.1145/3102071.3110568

Lombard, M., & Ditton, T. (2006). At the Heart of It All: The Concept of Presence. *Journal of Computer-Mediated Communication*, *3*(2). https://doi.org/https://doi.org/10.1111/j.1083-6101.1997.tb00072.x

Lynn, J. (2013). Combining Back-End Telemetry Data with Established User Testing Protocols: A Love Story. In M. S. El-Nasr, A. Drachen, & A. Canossa (Eds.), *Game Analytics: Maximizing the Value of Player Data* (pp. 497–514). London: Springer-Verlag. https://doi.org/10.1007/978-1-4471-4769-5

Medler, B. (2013). Visual Game Analytics. In M. S. El-Nasr, A. Drachen, & A. Canossa (Eds.), *Game Analytics: Maximizing the Value of Player Data* (pp. 403–433). London: Springer-Verlag. https://doi.org/10.1007/978-1-4471-4769-5

Mereu, M., Hudson, P., Nix, S., & Richards, J. (2013). Digital Distribution and Games as a Service: Symposium Panel I. *SMU Science & Technology Law Review*, *16*(25), 1–26.

Morosan, M., & Poli, R. (2017). Automated Game Balancing in Ms PacMan and StarCraft Using Evolutionary Algorithms. In *Applications of Evolutionary Computation* (pp. 377–392).

Morosan, M., & Poli, R. (2018). Lessons from Testing an Evolutionary Automated Game Balancer in Industry. In *2018 IEEE Games, Entertainment, Media Conference* (pp. 263–270).

Schell, J. (2019). *The Art of Game Design* (3rd ed.). Boca Raton: A K Peters/CRC Press.

Schreiber, I. (2016). A Course about Game Balance. In *Game Developers Conference*.

Seufert, E. B. (2013). *Freemium Economics*. Waltham: Morgan Kaufmann.

Sweetser, P., & Wyeth, P. (2005). GameFlow: A Model for Evaluating Player Enjoyment in Games. *ACM Computers in Entertainment*, *3*(3), 1–24.

Vilaseca, P., & Guardascione, P. (2019). Breaking down the metagame design in a mobile RPG. Retrieved from https://www.gamasutra.com/view/news/333792/Breaking_down_the_metagame_design_in_a_mobile_RPG.php

Volz, V., Rudolph, G., & Naujoks, B. (2016). Demonstrating the Feasibility of Automatic Game Balancing. In *Genetic and Evolutionary Computation Conference* (pp. 1–13).

Zook, A., Fruchter, E., & Riedl, M. O. (2014). Automatic Playtesting for Game Parameter Tuning via Active Learning. *ArXiv*.