

Relating Cognitive Models of Computer Games to User Evaluations of Entertainment

Paolo Piselli and Mark Claypool
Interactive Media and Game Development
Worcester Polytechnic Institute
100 Institute Rd, Worcester, MA 01609, USA
claypool@wpi.edu

James Doyle
Social Science and Policy Studies
Worcester Polytechnic Institute
100 Institute Rd, Worcester, MA 01609, USA
doyle@wpi.edu

ABSTRACT

As the interactive entertainment industry matures, a better understanding of what makes software entertaining is needed. A natural starting point is the application of traditional Human-Computer Interaction (HCI) tools to interactive entertainment software. HCI tools include cognitive models that researchers have used to model users' thought processes and evaluate interface design. This paper uses a simple cognitive model to investigate the relationship between the complexity of an interaction and the entertainment experienced by the user. We design a simple computer game, create a normative model for how a user plays this game, and build several variations of this game such that normative models of these variants differed across two factors: pace and complexity. User studies conducted on these variations allow comparison with these factors to user performance and self-reported user enjoyment. Users in-game enjoyment was found to be related to both the subject's performance and the game complexity.

1. INTRODUCTION

Over the last two decades, entertainment software has grown from a cottage industry to a multi-billion dollar industry. According to a January 2006 press release by NPD Research, U.S. retail sales of game consoles, handheld devices, games and accessories reached \$10.5 billion in 2005. As the industry matures, developers have a competitive interest in producing games that offer a more entertaining experience to the end-user; however there has been limited scientific research into what, exactly, makes a computer game entertaining.

Although specialized for the purpose of entertaining one or more people, a computer game is still software and thus provides the opportunity for study by computer scientists as software. The science of Human-Computer Interaction (HCI) is relevant to such a user-focused domain. One informal suggestion from HCI is that a game must have a certain computational complexity in order to be interesting [E1]. The problem with this theory is that it speaks of the complexity of an algorithm used to compute a solution to a game, rather than the thinking a human player does while playing a game. If humans go about playing games in different ways than comput-

ers compute solutions to games, then there may be more accurate measures as to what interests a human than just computational complexity.

An evaluation of the user experience must look at interest and complexity from the user's point of view. Studies indicate that there is a correlation between cognitive interest and enhanced learning [HM1997]. It is possible that the cognitive features of a game also correlate with enjoyment. Cognitive models, tools that model user thought processes, may take a variety of forms that capture thinking, such as decision trees or production rules. Cognitive modeling has been used in HCI research for over two decades, but to the best of our knowledge, has yet to be applied to the study of user enjoyment of computer games. As a quantifiable representation of a user's thought process, a cognitive model is an artifact that can be measured more directly than the user's actual thoughts.

This research investigates the relationship between a cognitive model of an interaction and the entertainment value that a user places on that interaction. We built four variations of a simple computer game, created normative rule-based models of how users play these variations, and conducted user studies to determine the relative entertainment value of each variation. The game was made simple enough so that the normative models for play could be constructed a priori. The variations were designed such that their models would differ by the number of rules in the model and by the rate at which these rules were used. User studies were conducted with these game variants on over 100 college-aged men and women. Users played a randomly chosen game variant then filled out a post-test questionnaire. The resulting data was then analyzed to determine if the size of the active-set of rules and rate of rule-use in the cognitive models can be related to user self-evaluations of entertainment.

An analysis of our data indicated that complexity variable had a statistically significant relationship with user perception of difficulty. Users' in-game enjoyment was significantly related to both the users' performance and the complexity variable. Our data shows evidence of an "inverted-U" phenomenon when comparing enjoyment to performance. There is also a noted difference between the enjoyment that users reported in-game and the enjoyment that is reported in the post-game questionnaire.

The rest of this document is organized as follows: Section 2 discusses work related to this research; Section 3 describes our approach to designing our game and our user study; Section 4 presents an analysis of our user study data; Section 5 discusses issues that arose and lessons learned; Section 6 summarizes our conclusions and presents possible future work.

2. RELATED WORK

Work related to our research primarily falls into three areas: cog-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ICFDG 2009 April 26-30, 2009, Orlando, FL, USA.
Copyright 2009 ACM 978-1-60558-437-9 ...\$5.00.

nitive Psychology, cognitive modeling of HCI, and science and entertainment software.

2.1 Cognitive Psychology

Nearly a century ago, Yerkes and Dodson's famous experiment predicted an "inverted-U" relationship between level of arousal and performance in a memory task [YD1908]. The Yerkes-Dodson law predicts that there is a level of emotional arousal that optimizes a subject's rate of learning on some task. With too little or too much emotional arousal, the subject's rate of learning will be diminished. Our research also investigates emotional arousal, but the specific type of emotional arousal that a subject judges to be "enjoyment". We also seek to investigate this type of arousal as a variable that is dependant on factors of an interactive experience.

Petty and Cacioppo present the elaboration likelihood model of persuasion that models how attitudes are formed and changed [PC86]. The model incorporates both thought and motivation, while our approach examines enjoyment based on input and an interactive experience.

Burns proposed the use of Bayesian-information theory to analyze enjoyment in a game of slots [BU2006]. Burns' work models fun as resulting from information gain from a violation of expectations. This model also predicts inverted-U functions, referred to as "Goldilocks Functions", relating fun to win-probability. Our research attempts to relate fun to cognitive complexity. When cognitive complexity impacts win-probability, there is evidence of a Goldilocks Function.

2.2 Cognitive Modeling in HCI

Cognitive models have been in use as evaluative tools in HCI for over two decades. One of the most popular ways of modeling has been GOMS, which codifies a cognitive task as Goals, Operators, Methods and Selection rules. John and Peck used Soar, a GOMS model, to create a computational model of the task of browsing a database [JP1992]. Their work shows that GOMS models can capture the cognitive elements of even a highly interactive task. John and Kieras, in their overview of GOMS models, described ten successful real-world applications of a GOMS model to user interface design issues [JK1996].

Another popular cognitive modeling tool is the ACT-R family of cognitive architectures. Belavkin applied ACT-R to model the inverted-U phenomenon described by the classical Yerkes-Dodson experiment [BE2001]. This work modeled emotional activation and found that a medium level of activation resulted in a higher simulated rate of learning. Byrne applied the ACT-R/PM architecture to modeling the interactive task of menu selection [BY2001]. ACT-R/PM was a cognitive modeling system which had an additional perceptual-motor module to enable modeling interactive tasks to a higher level of detail. Aside from GOMS and ACT-R, Chery and Farrel described HCI research involving Perceptual Control Theory [CF1998], which may be better suited to modeling tasks of continual adjustment.

2.3 Science and Entertainment Software

The scientific study of entertainment software is a relatively nascent field. Vorderer, Klimmt and Ritterfeld presented an integrated theory of media entertainment that accommodated for both the user and the media [VKR2004]. Their model of Complex Entertainment Experiences (CEE) was a conceptual model that defined enjoyment as a user's response to a media product based on the particular user, motives, and the particular media. In this model, it is suggested that all three of these must meet certain prerequisites in order for an experience to be viable as entertainment; however a precise enu-

meration of these prerequisites was not given.

Vorderer et al. divided the properties of a media product into four categories: technology, design, aesthetics, and content. They stated that a product with a certain level of interactivity may bore or interest a user depending on the particular content and the particular user, and that users seek a certain emotional complexity to their experiences. Their survey of other research indicated that human motivations for playing video games range from presence to self-efficacy to competition. Our research focuses on the complexity of the cognitive aspects of an interaction. In the light of Vorderer's integrated theory, this research asks the question: "given the same users and the same content, to what degree do the purely cognitive aspects of entertainment software affect the users' enjoyment?"

3. APPROACH

This research investigates how the cognitive complexity of playing a computer game influences how entertaining it is for a user. Human cognition of a game was modeled as a rule-based system. The factors of complexity that chosen to investigate included total number of rules in the model, and frequency of rule activation. These factors were investigated in the context of a simple "Punch-Out" style game. An original game in this genre was developed, including original artwork. Four variants of the game were made such that the normative cognitive models of these games would vary along the two chosen factors. User studies were then conducted using a between-subjects 2x2 randomized experimental design. Users gave self-evaluations of entertainment both during and after their testing sessions. Finally, data from the user studies was analyzed to see how different levels of game complexity affected user-reported levels of enjoyment.

3.1 Game Design

3.1.1 Design Choice

This research attempts to relate the complexity of thought needed to play a game to how entertaining it is by relating cognitive models to user enjoyment. For our results to be meaningful there needed to be some certainty that the cognitive models matched the thinking of the human subjects. Typically, a cognitive model is developed through Cognitive Task Analysis (CTA), a process which may involve observing subjects as they perform some task, asking subjects to "think aloud" and describe their thoughts as they perform the task, or interviewing subjects about the decisions they make under certain circumstances.

With limited time to complete this work, we instead chose to investigate a style of computer game which had gameplay straightforward enough that gameplay cognition could be modeled without lengthy CTA. Such a game must give the user limited choices, otherwise it could involve decision making that varies from user to user. A game where the proper action to take is clear for every condition that occurs would challenge the player's memory, dexterity, concentration, and reaction time rather than their decision making ability.

Electronic games such as Simon (©1978, Milton Bradley) are purely reactive. In Simon, the player is challenged to repeat a sequence of colors made by the device. For each sequence presented by Simon, there is only one proper reaction: to repeat the given sequence. By investigating a game with a reactive style of gameplay it is presumed that a model for play that captures the proper reactions to all conditions represents the normative play style.

In addition to being purely reactive, certainty was needed that subjects had an understanding of the game that matched the normative model. This meant that subjects would need to learn the

proper actions for all conditions before testing. By keeping the number of conditions small, subjects could be quickly trained, giving confidence that they had a complete understanding of how to play during the test.

3.1.2 Punch-Out Style Games

Our game was based on a reactive play style similar to that of the game Punch-Out (©1987, Nintendo). Punch-Out is a simple abstraction of a one-on-one boxing match. The user has control over one of the boxers, and can execute a limited set of punch, block and dodge actions. The computer controls the actions of the user's opponent. There is very little strategic planning on the part of the user. The game task is mostly reactive – the user must recognize a certain action executed by the opponent and respond with an appropriate sequence of actions within a limited amount of time. For example, if the opponent raises his right arm to punch, the proper action would be for the player to dodge to the opposite side then counter-punch to the mid-section. By using this style of gameplay, the number of conditions that could arise could be controlled as could the proper response to each.

3.1.3 Paolo's Kickboxing

Similar to Punch-Out, our game, called *Paolo's Kickboxing*, involves two on-screen avatars, one user-controlled and one computer-controlled (see Figure 1). The game consists of ten rounds of play, where each round is won by the first boxer to score ten hits in the round. Both avatars start in a neutral state and are able to execute any of their available actions from this neutral state. The user interface involves pressing keys that directly correspond to actions. Execution of an action causes the relevant avatar to transition to a different state, such as blocking or punching, followed by a return to the neutral state.

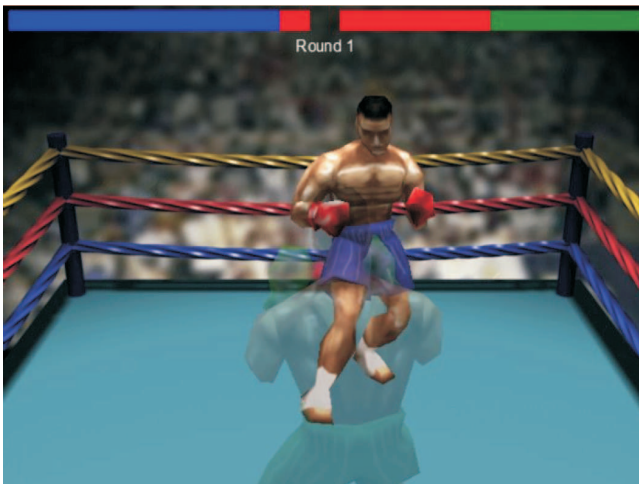


Figure 1: Paolo's Kickboxing

The proper action for the user in the neutral state is to wait for the opponent to take an action. If the user attacks while the opponent is in a neutral state, the opponent will automatically block the user's attack. When in the neutral state, the computer will delay for a set amount of time then randomly select an attack to execute. The rate of the computer's attacks was varied to control the pace of the game. Also, the number of different attacks which the computer could randomly select was varied to control the complexity of the game. When the computer makes an attack action, the user has a

one-second interval in which to recognize the attack and perform an appropriate block action. If the user successfully blocks the opponent's attack, the opponent is then vulnerable to a counter-attack for a one-second interval. These generous response intervals were selected because we did not want reaction-time to be a large factor in the difficulty of the game.

Hick's Law predicts that the choice response time given N equally probable choices is logarithmic in N [H1952]. This model of interaction is applicable to our game. However, empirical research is required to estimate the parameter in Hick's equation, and the parameter can differ from individual to individual. For our research, a response interval of one second was chosen for all test groups based on an informal pilot study that showed trained subjects could easily respond within this time frame.

3.2 Game Variants

We built four variants of the game such that normative models for game play would vary across two factors. These factors were the rate of decisions, or *pace*, and number of choices, or *complexity*, of playing the game. The first dimension of variation was the pace at which the computer player made attacks. The second dimension of variation was the total number of possible conditions i.e. the number of possible attacks executed by the opponent. Informal user testing was conducted as the game was developed to determine the two levels for each of these factors. The intent was to capture a level that was challenging and a level that was not challenging for each factor. For the pace factor, we chose an opponent that would attack every two seconds and an opponent that would attack every four seconds. For the complexity factor, we chose an opponent capable of executing four types of attacks and an opponent capable of executing eight types of attack. The attacks took the form of punches and kicks. The four-attack opponents had one punch attack and one kick attack on both the left and the right sides. The eight-attack opponents had high and low punch attacks and high and low kick attacks on both the left and right sides.

3.3 Cognitive Models

3.3.1 Rule Based Systems

A rule-based system represents decision making as a collection of IF-THEN rules that state that IF certain conditions are true, THEN perform certain actions. The system has a working memory that maintains assertions about the current state of the world called facts. A rule's conditions make logical statements about these facts, and when these statements become true, the rule activates, and its actions are performed. A rule-based system is sufficient for modeling our game because our game involves the user taking discrete actions in response to actions by the opponent. Rule-based systems may not be effective for modeling some types of games because rules are not good at modeling tasks involving continuous control.

By representing normative gameplay cognition as a rule-based system, the gameplay task can be quantified in various ways. The number of rules in the entire model can be counted. Also, the number of facts that must be kept in working memory can be determined. The frequency in which rules must be activated and how many rules might apply at any given time can be predicted. For each rule, the time taken for working memory to be updated can be timed, as well as the time for conditions to be matched and the rule executed. If our model accurately captures human thought, then these quantities represent the demands the game places on the user's cognitive faculties.

3.3.2 Modeling Assumptions

It is assumed the rules in a model executed in sequence collapse into a single rule with a concatenated sequence of actions. Because the opponent always becomes vulnerable after a successful block action, the two separate rules “if I am attacked, then block” and “if the opponent is vulnerable, then punch” collapse into the single rule of “if I am attacked, then block and then punch”. Thus, all “block, punch” action sequences are assumed to be the result of the activation of a single rule.

In measuring the features of a rule-based model, the way in which the rules are written influences the resulting numbers. For instance, the single rule “if A or B, then do C and D” is functionally equivalent to having the two rules “if A then do C and D” and “if B then do C and D”. For a more complicated model, a well-defined method of counting rules based on unique sets of conditions is needed, resulting in unique sequences of actions. However, our game is intentionally simple so that the arising conditions are mutually exclusive. Therefore, there is assumed to be exactly one rule in the model for each condition that results in a specific sequence of actions.

Our game also has the property that there is only one proper sequence of actions to take for each condition. Thus, there is exactly one rule in the model for each condition. Finally, the only conditions that arise that are appropriate for the user to respond to are attacks by the opponent; therefore there is exactly one rule in the model for each of the opponent’s attacks.

3.3.3 Characterization

Given the stated game design, the gameplay in Paolo’s Kickboxing can be described by a rule-based model which has a certain number of rules and time constraints for how quickly and how often these rules must operate. In all cases, the firing of a rule must take place within one second in order for its action to be successful. The number of rules in the model for a given variant is equal to the number of attacks the computer opponent can execute in that version of the game.

For the four-attack variants, there are four rules in the model:

```
IF opponent punches to my left
  THEN press `7`, press `space`.
IF opponent kicks to my left
  THEN press `1`, press `space`.
IF opponent punches to my right
  THEN press `9`, press `space`.
IF opponent kicks to my right
  THEN press `3`, press `space`.
```

For the eight-attack variants, there are four additional rules in the model:

```
IF opponent punches low to my left
  THEN press `4`, press `space`.
IF opponent kicks low to my left
  THEN press `0`, press `space`.
IF opponent punches low to my right
  THEN press `6`, press `space`.
IF opponent kicks low to my right
  THEN press `.` , press `space`.
```

Our models also differ across the rate at which the rules are activated. The slower variants have a delay of four seconds between attacks and the faster variants have a delay of two seconds between attacks. With these four variants, there are four conditions for our 2x2 experimental design, shown in Table 1.

Table 1: Game Conditions

	Simple	Complex
Slow	4 rules, 0.25/sec	8 rules, 0.25/sec
Fast	4 rules, 0.5/sec	8 rules, 0.5/sec

3.4 Game Development

Game Maker¹ was used software as the development platform for our game. Game Maker provides a visual interface for importing 2D images, managing game content, and building the various stages of the game. The built-in game engine has hooks for a wide range of user inputs, both keyboard and mouse. The scripting features of Game Maker gives access to libraries for 2D graphics and file I/O.

The game logic was parameterized such that a new variant of the game could be created by simply changing a constant in the code. This facilitated randomized testing, as the game can pick one of the four variants at random at the start of a new gaming session. The post-test questionnaire was implemented in the game as well, which gave the user’s testing experience a continuous flow and allowed for convenient logging of questionnaire answers to a file.

3D Studio (3DS) MAX was used to develop the pre-rendered sprites for both players, as well as the pre-rendered background. The use of pre-rendered 3D (using the 2D Game Maker engine) gave the game a crisp, 3D look without the difficulty of programming a full 3D scene-graph rendering engine or programming routines for importing 3D geometry and animations. Since the game had a point-of-view fixed behind the player’s avatar, and combatants that are unable to move about the ring, 3D scenes could be created in MAX with the boxers and the ring rendered from a fixed camera. Pre-rendering the graphics greatly sped up the process of creating and editing the animations. Upon completion, the game had over 150 frames of animation.

One boxer was modeled, texture mapped, and rigged with skeletal system for both the user and the opponent in 3DS MAX. One base-texture with two variations in the color of the trunks was created for this mode in Adobe Photoshop. Animations were created in 3DS MAX and rendered to image sequences for conversion into sprites. Various interface graphics and iconic representations of levels of enjoyment were created in the GIMP.

To match the game design, the attack animations were designed to be two-seconds long each, moving through several key poses. The opponent begins each animation in a neutral pose. Over 0.5 seconds, the opponent then transitions through a “tell” pose – a pose that allows the user to clearly recognize the incoming attack. The opponent then transitions to a “strike” pose at the midway point of the animation – a pose which indicates to the user that the interval to respond to the attack is over and that the opponent has made contact with the user. In the final second of the animation has the opponent returning to the “neutral” pose.

During user testing, data was logged to a text file marked with a unique, non-identifying session ID. The post-test questionnaire was implemented in-game, and questionnaire responses were also logged to this file. Following testing, data from each subject’s text files was aggregated into a single table for manual filtering and preliminary analysis in a spread sheet. Finally, the table was imported into SAS for more advanced analysis.

¹<http://www.yoyogames.com/make>

3.5 User Studies

Users were solicited, both orally, in writing and via email, to participate in the experiments. Extra credit was given to students in some classes in exchange for participation in the study. Most subjects were undergraduates who agreed to participate in exchange for the aforementioned extra credit, with an additional number of undergraduate and graduate students who volunteered in response to an announcement of the study. Students were given sign-up sheets where they could schedule a specific half-hour time slot to participate in the study.

All testing took place in a campus computer lab. The lab was not reserved for our exclusive use, but conditions were quiet and each subject was adjacent to vacant computer stations. Users wore ear-covering headphones to hear in-game sounds and muffle background noise. Users were tested three at a time, positioned at computer stations in three corners of the lab, with users not facing each other. The tester was present in the room during testing, but sitting in the fourth corner of the room, not facing the test subjects.

Instances of the game were launched over the campus network, with log files written to the remote directory. The remote execution of the game centralized the recording of log files to one location on the network without the use of additional networking code within the game itself. Remote execution also enabled testing to easily take place on any free PC in the lab, as no executable files, configuration files, or log files needed to be moved back and forth from the local machine to the centralized storage.

When each user arrived for their time-slot, s/he was given a waiver that explained the testing procedure and the voluntary nature of their participation. After reading and signing the waiver an instance of the game was started on a free lab PC and the student was left alone to undergo the in-game tutorial (see Figure 2), play the game, then complete the in-game questionnaire. After each round in the game, the user was asked how much they had enjoyed that round by clicking on one of five different icons (see Figure 3). After the completion of ten rounds in the game, a 20 question in-game questionnaire was administered. Once the questionnaire was completed, the subject was informed by the game that their testing session was over. The entire testing session took from 10 to 20 minutes depending on the speed of the variant and the proficiency of the subject.

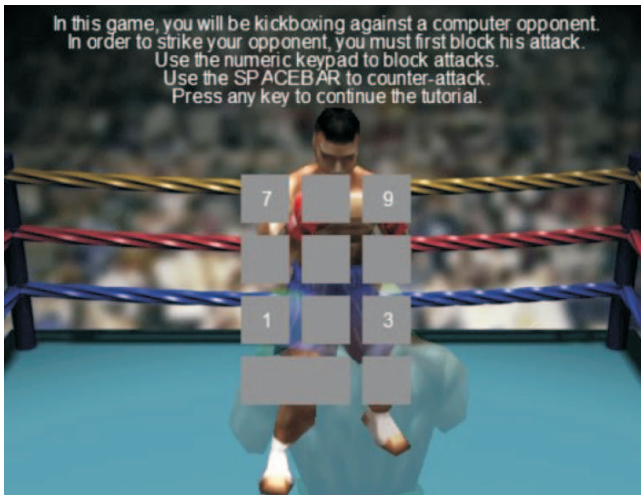


Figure 2: Game Tutorial

The questionnaire that followed the game asked demographic questions regarding age, gender, and computer usage and asked users to evaluate their level of interest, immersion and enjoyment of the game.² The answers to these questions enabled correlation of features of the user's experience with the features of the specific game variant played.

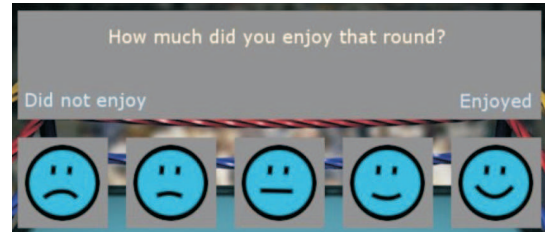


Figure 3: Post-round Assessment

4. RESULTS

In total, 107 subjects were tested. However, input lockup occurred during three of the testing sessions and one testing session was aborted early due to program failure. The data used in our analysis is from the remaining 103 subjects. Of these 103 subjects, 82 were male and 21 were female. With respect to computer experience, 82% of subjects reported 20 or more hours of computer use per week. With respect to gaming experience, 72% of subjects reported computer game play of 10 hours or fewer per week, and 50% of subjects reported 5 hours or fewer per week. Although the game's random selection process had an equal probability of selecting each game variant, the final sample sizes were 23 subjects for the slow and simple variant, 27 for the fast and simple variant, 25 for the slow and complex variant, and 28 for the fast and complex variant.

4.1 Questionnaire

A two-way analysis of variance (2-way ANOVA) was conducted on each questionnaire response using pace and complexity as the independent variables. The slow pace and low complexity levels of each variable were assigned numerical values of -1, while the fast pace and high complexity levels were assigned values of 1. The five different questionnaire responses were scaled on a -1 (strong disagreement) to 1 (strong agreement) scale. In almost all cases, there was not a significant difference across the two levels of the pace variable (at 0.05). Pace did have a small effect on response to the question of whether the user was engaged by the game ($F=4.36$, $p=0.0386$, $R^2=0.076$).

Across levels of complexity, users showed a marked difference in responses to the statement "this game was difficult to learn" ($F=63.69$, $p<0.0001$, $R^2=0.41$), and the statement "this game was difficult to play" ($F=77.2$, $p<0.0001$, $R^2=0.44$). Complexity also had significant difference in responses to the statement "this game held my attention" ($F=4.31$, $p=0.04$, $R^2=0.077$) and the statement "this game had a goal" ($F=5.88$, $p=0.0171$, $R^2=0.066$). Analyses using questionnaire responses for age, gender, computer usage and game usage as controls did not find significant results.

²A full copy of the questionnaire can be found at: <http://www.cs.wpi.edu/~claypool/papers/game-fun/>

4.2 User Performance

Users showed evidence of improving performance over the course of their ten-round play sessions. The user’s margin of victory over the opponent was used as a measure of individual round performance. This value had a maximum of 10 (perfect victory) and a minimum of -10 (utter defeat). When considering all rounds played by all users, the complexity of the variant showed a significant effect on the margin of victory ($F=661$, $p<0.0001$, $R^2=0.392$), and mean margin of victory increased based on round number ($F=3.17$, $p=0.0009$, $R^2=0.0272$).

Figure 4 illustrates the difference in average performance between the simple and complex variants over the course of 10 rounds of time. In this figure, the horizontal axis represents the round of the game and the vertical axis represents the point difference, or margin-of-victory, at the end of the round. There are four data series which represent the four game variants, and each data point is the mean margin-of-victory in that round for users who played that variant.

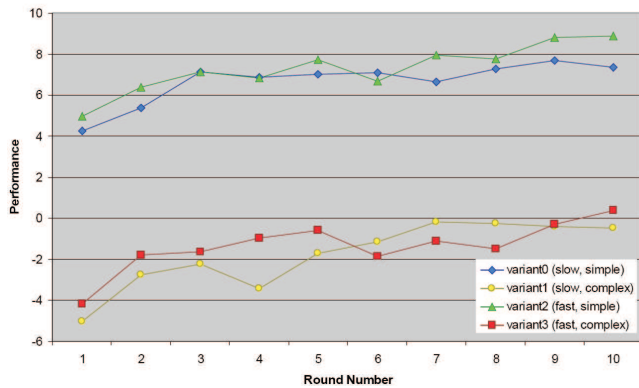


Figure 4: Mean Performance (point difference) over Time (rounds)

Figure 4 indicates that there was a large performance gap between players of the simple and complex variants. On average, users of all variants improved in their margin of victory by roughly 4 points over the course of testing. Notice that although improvement in performance was similar, average performance of users in the simple variants begins and ends in “winning territory”, while average performance in the complex variants begins in “losing territory” and approaches the win-loss threshold.

4.3 Enjoyment versus Time

Figure 5 shows a graph of in-game user enjoyment over the course of the ten rounds for each variant. The horizontal axis represents the round number, and the vertical axis represents the mean reported enjoyment in the $[-1, 1]$ range. In this graph, enjoyment appears flat with the simple game variants, but enjoyment appears to increase over time with the complex variants.

One could assume that the more complex variants were more difficult and therefore as the user’s performance improved, s/he felt a greater sense of accomplishment in overcoming the challenge. However, as observed in Section 4.2, the absolute improvement in mean margin-of-victory from round 1 to round 10 was similar for all variants. If the users were simply judging their increasing mastery of the game, enjoyment over time should be similar for all variants.

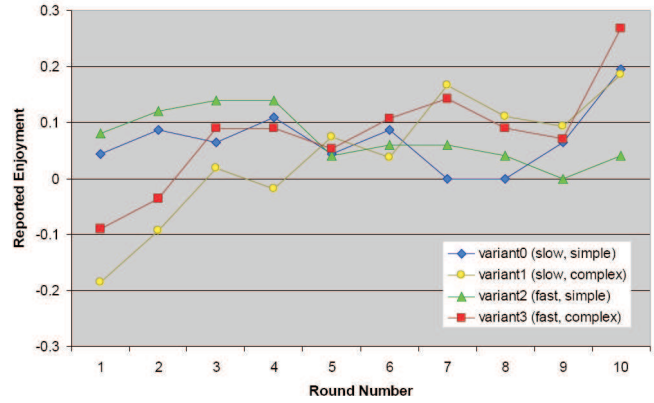


Figure 5: Mean Enjoyment over Time (rounds)

4.4 Enjoyment versus Success

With our chosen factors of pace and complexity not having much effect on user enjoyment, enjoyment with respect to user performance was examined next. First, the post-game enjoyment reported in the questionnaire was compared to measures of overall performance. For measures of overall performance, the fraction of rounds that the user had won and the user’s average margin of victory over all rounds were used. A regression on enjoyment was done with respect to pace, complexity, fraction-won and average-victory. In this model, fraction-won showed a significant effect ($p=0.044$, parameter estimate 1.33) on the user’s post-game enjoyment.

Next reported enjoyment on the individual-round level was examined. The variable “won” was assigned the value of 1 for a round that the user won and a valued of 0 for a round that the user lost. A regression on the reported enjoyment was done for each round across all users with respect to pace, complexity and won. In this model, won showed a significant effect ($p<0.0001$, parameter estimate 0.6) and complexity also showed a significant effect ($p<0.0001$, parameter estimate 0.14) on the user’s enjoyment of an individual round.

Finally, with success seeming to be a significant factor in user enjoyment, a regression on individual round enjoyment was done with respect to the user’s margin of victory in that round. In this model, margin of victory showed a significant effect on round enjoyment ($p<0.0001$).

Visually, this relationship can be seen by looking at a graph of the mean reported enjoyment for each margin of victory (see Figure 6). The horizontal axis represents the margin of victory and the vertical axis represents reported enjoyment. Each data point in this graph represents the mean of the reported enjoyment for all rounds that ended in the given margin of victory, with the vertical lines depicting the standard deviation. The number of rounds that ended in the respective margin-of-victory fall in the range [26,147]. The interesting feature of Figure 6 is the large jump in reported enjoyment between a margin of -1 (barely lost) and 1 (barely won). Also interesting is the falloff of enjoyment close to the rounds that were perfect victories. This is empirical evidence of an “inverted-U” phenomenon [YD1908]. However, unlike the Yerkes-Dodson function, which plots learning versus arousal, this inverted-U appears in a graph of performance versus enjoyment. This graph is similar to the Goldilocks Functions described by Burns that relate fun to win-probability [BU2006].

This exploration of mean enjoyment also elucidates the differences in enjoyment over time between the simple and complex game variants. Referring back to mean performance per round,

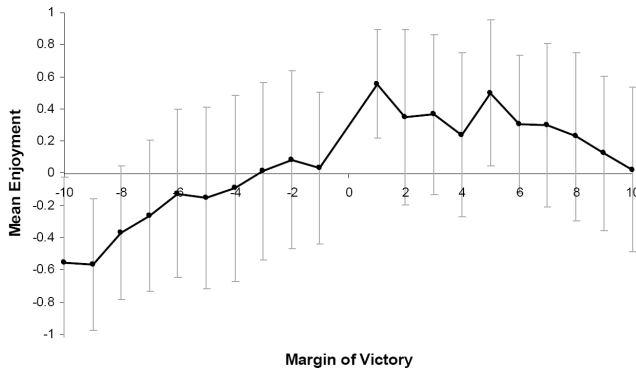


Figure 6: Performance versus Enjoyment

one can see that over the course of testing, the average play session involving a simple variant moves from moderate success to near perfect success, whereas the average play session involving a complex variant moves from moderate failure to the barely-lost/barely-won threshold. Thus the users with the simple variants are moving down the right decline of the inverted-U as their play improves, while users with the complex variants are moving up the left incline of the inverted-U as they improve. Both groups' performance may be improving to a similar degree, but what appears to be important is not how much performance improves, but how close the user's experience was to the "sweet spot" on the inverted-U of performance.

5. DISCUSSION

5.1 Development Postmortem

Although using pre-rendered 3D art saved a great deal of effort, the consequence was that the game had a very large memory footprint. Because of the Game Maker middleware development platform, there was no ability to optimize the memory usage of the sprite engine. It is likely that images for all frames of animation were held uncompressed in memory simultaneously, with the memory usage of the game becoming excessive. In order to save memory, the frame rate of the animations were halved, thus halving the number of images per animation. Even after this change, memory usage often exceeded 250 MB. The user studies could not be conducted on machines with slow hard-disks or insufficient RAM because disk caching effects slowed the pace of the game and lengthening the response interval for reacting to a punch. Fortunately, the campus lab had powerful enough machine machines that were able to play the game smoothly.

After testing, a few subjects reported difficulty in distinguishing between the high- and low-kick conditions until it was too late. The intent was to make each animation such that as the opponent transitions into the "tell" poses the task of recognizing each condition would be easy. In the case of the kick animations, the tell poses were similar, and early frames of the animations differed only slightly (see Figure 7). If the subject was unable to detect this subtle difference, then the condition would not be recognized until a later point in the animation, and the user would have effectively less time to react with a block action. It is possible that this problem was caused in part by the frame-rate reduction of the animations (described above), a step which was taken to save memory. To correct this, the kick animations could be reworked such that the difference is clear at an earlier time.

In a few cases, a problem arose due to the location of input keys

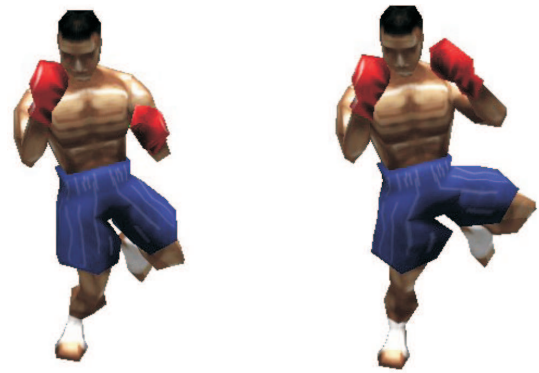


Figure 7: Differences in "Tell" Poses for Low-Kick and High-Kick

on the keyboard. The buttons for blocking were located on the numeric keypad of the PC keyboard. If the user tried to press the '7' key, but overshot their target by a full row of keys, they would hit the 'numlock' key, which would cause the game to stop recognizing key presses on the numeric keypad. The data for the few cases where this issue interrupted testing was discarded.

5.2 Problems with Pace

Our original intent was to have the enemy attack at a rate of one attack every two or four seconds, depending on the variant. During informal testing, we realized that how quickly a user countered the opponent affected how fast they perceived the pace of the game. For example, assuming a two-second attack interval, if the user blocks and counters in 0.5 seconds, the opponent's next attack would occur 1.5 seconds after the counter attack. If the user were less proficient in recognizing the condition and responding, their block and counter could take as long as 1.5 seconds, leaving only 0.5 seconds before the opponent's next attack and making the pace of the game seem much faster.

Instead, the pace was changed from a rate of attack to a delay until the next attack. This way, no matter how long the user took to execute their actions, the duration until the opponent's next attack would be constant. In doing this, the mistake was made of keeping two-seconds and four-seconds as time values. Following the user study, it became clear that the chosen levels for the pace factor resulted in little difference between questionnaire responses especially in reported level of difficulty, where the complexity factor showed a large effect. Prior to the change, the effective delay before the opponent's next attack following the completion of the user's response would have been the attack interval minus the time for the user to recognize and respond to each attack. This delay would have been close to one-second for the "fast" variants, and close to three seconds for the "slow" variants.

5.3 Cognitive Modeling

An underlying assumption of this research was that the user would recognize a certain condition, make a decision on how to respond and then respond within a certain amount of time. Within this interaction, only the cognitive task of decision making was modeled; however this process also involves the perceptual task of recognizing the conditions and the motor task of executing the responses. It became clear when certain users had difficulty differentiating between a few of the attacks, and when users overshot or missed the appropriate key, that the perceptual and motor tasks are also impor-

tant factors in at least the difficulty of the game, if not the entertainment value.

6. CONCLUSIONS

The original goal of was to determine if the complexity of a game impacts its entertainment value. However, our measures of pace and complexity did not show a direct relationship to post-game evaluations of enjoyment. The complexity variable did show a significant effect on subject performance, perception of difficulty, and reported in-game enjoyment. This result supports our hypothesis that complexity affects enjoyment, but also indicates an important difference between in-game reporting of enjoyment and post-game reporting of enjoyment.

Subject performance improved similarly across all variants; however in-game enjoyment did not follow improving performance. Post-game enjoyment was significantly related to the fraction of rounds won and in-game enjoyment was significantly related to whether an individual round was won. An analysis of in-game enjoyment with respect to margin-of-victory showed evidence of an inverted-U phenomenon. This differed from previous research in that performance was treated as the independent variable and enjoyment as the dependant variable. The difference in enjoyment over time across game complexity can be explained by observing that players of the simple variants were approaching perfectly victorious performance, while players of complex variants were approaching barely victorious performance. The experimental evidence showed that mean enjoyment peaked at levels of performance near barely victorious, and fell off towards perfectly victorious, thus players of complex variants were climbing up the enjoyment curve while players of simple variants were sliding down the side. This result is significant in that proximity to the victory-threshold is an important factor in the enjoyment of a game such as ours. This result also contradicts the idea that mastery of a game leads to enjoyment, because it indicates that an increase in performance can lead to a decrease in enjoyment.

Although the differences in our cognitive models did not explain user enjoyment very well, they were good for characterizing a task and speaking of its difficulty. Cognitive models provide us with tools to speak of the equivalence of games. Once the structural similarities in the gameplay of different games can be discussed, factors outside of gameplay that influence enjoyment can be researched. Would two games that have identical cognitive models be enjoyed differently by users if they are framed in different contexts (e.g. a competitive activity versus a cooperative activity) or had a different style of artwork (e.g. cartoon 2D sprites versus realistic 3D models)? If gameplay can be isolated, then it can be determined if there are demographic differences in enjoyment, and so determine to what degree factors such as gameplay, context and aesthetics contribute to these differences.

The motivational factors that help explain user enjoyment of an interactive game are also rich areas for future work. With success seeming to play heavily into user enjoyment, further research could investigate how goals play into enjoyment. The differences in enjoyment between a game with one clear victory condition and a game with multiple goals could be explored. Determining how many goals are necessary for enjoyment, or whether a user will create goals to enjoy overcoming can be undertaken. The effect of frequency of goal-satisfaction, and perceived probability of goal-satisfaction on user enjoyment can be examined. Social goals such as competition and cooperation with other people could also be important factors in user enjoyment to be explored.

References

- [BE2001] Belkavin, R.V. "Modeling the Inverted-U Effect in ACT-R", *Proceedings of the Fourth International Conference on Cognitive Modeling*, 2001. pages 275-276.
- [BU2006] Burns, K. "Fun in Slots", *Proceedings of CIG-06: IEEE Conference on Computational Intelligence in Games*, Reno, NV, USA, May 22-24, 2006.
- [BY2001] Byrne, Michael D. "ACT-R/PM and menu selection: Applying a Cognitive Architecture to HCI", *International Journal of Human-Computer Studies*, 2001. pages 41-84.
- [CF1998] Chery, Sandra; Farrell, Philip S.E., "A Look at Behaviourism and Perceptual Control Theory in Interface Design", *Defense and Civil Institute of Environmental Medicine*, Report Number 98-R-12, 1998.
- [DH2000] Douglas, Yellowlees; Hargadon, Andrew. "The Pleasure Principle: Immersion Engagement, Flow", *Proceedings of the Hypertext Conference*, 2000. pages 153-160.
- [E1] Eppstein, D., "Computational Complexity of Games and Puzzles", Online at: <http://www.ics.uci.edu/~eppstein/cgt/hard.html>
- [GA2001] Gray, Wayne D.; Altmann, Erik M. "Cognitive Modeling and Human-Computer Interaction", *International Encyclopedia of Ergonomics and Human Factors*, October 2001.
- [HM1997] Harp, Shannon F.; Mayer, Richard E. "The Role of Interest in Learning from Scientific Text and Illustrations: On the Distinction between Emotional Interest and Cognitive Interest", *Journal of Educational Psychology*, 89(1), March 1997. pages 92-102.
- [H1952] Hick, W. E. "On the Rate of Gain of Information", *Quarterly Journal of Experimental Psychology*, 1952. pages 11-26.
- [JK1996] John, Bonnie E.; Kieras, David E. "Using GOMS for User Interface Design and Evaluation: Which Technique?", *ACM Transactions on HCI*, 3(4), December 1996. pages 287-319.
- [JP1992] John, Bonnie E.; Peck, Virginia A. "BROWSER-SOAR: A Computational Model of a Highly Interactive Task", *Proceedings of the SIGCHI Conference on Human factors in Computing Systems*, 1992. pages 165-172.
- [PC86] Petty, R. E., and Cacioppo, J. T. *Communication and Persuasion: Central and Peripheral Routes to Attitude Change*, New York: Springer-Verlag, 1986.
- [SR2001] St. Amant, Robert; Riedl, Mark O. "A Perception/Action Substrate for Cognitive Modeling in HCI", *International Journal of Human-Computer Studies*, 2001. pages 15-39.
- [ST2004] Steiner, Karl; Tomkins, Jay. "Narrative Event Adaptation in Virtual Environments", *Proceedings of the 9th International Conference on Intelligent User Interfaces*, 2004. pages 46-53.
- [VHK2003] Vorderer, Peter; Hartmann, Tilo; Klimmt, Christoph. "Explaining the Enjoyment of Playing Videogames: The Role of Competition", *Proceedings of the Second International Conference on Entertainment Computing*, 2003. pages 1-9.
- [VKR2004] Vorderer, Peter; Klimmt, Christoph; Ritterfeld, Ute. "Enjoyment: at the Heart of Media Entertainment", *Communication Theory*, 14(4), 2004. pages 388-408.
- [YD1908] Yerkes, Robert M.; Dodson, John D. "The Relation of Strength of Stimulus to Rapidity of Habit-Formation", *Journal of Comparative Neurology and Psychology*, 1908. pages 459-482.