Testing Incremental Difficulty Design in Platformer Games

Rina R. Wehbe[1,2], Elisa D. Mekler[2], Mike Schaekermann[1], Edward Lank[1], & Lennart E. Nacke[1,2,3]
Cheriton School of Computer Science[1] & Games Institute[2] & Department of Drama and Speech Communication[3]
University of Waterloo
Waterloo, ON, Canada
rina.wehbe|emekler|mschaeke|lank|len@uwaterloo.ca

ABSTRACT
Designing difficulty levels in platformer games is a challenge for game designers. It is important because design decisions that affect difficulty also directly affect player experience. Consequently, design strategies for balancing game difficulty are discussed by both academics and game designers. In this paper, we study how manipulating the following design decisions, commonly found in platformers, moderates difficulty: Scroll Speed, Target Size, Jump Task Complexity, and Perspective. Results for Scroll Speed and Target Size indicate that errors increase as speed increases and platform size decreases. However, results for jump task complexity demonstrate a separation of errors from task complexity. Specifically, while double-jump tasks are harder than single-jump tasks, triple-jump tasks appear to be as difficult as double-jump tasks. Additionally, the study demonstrates how changes in perspective affect the errors made by players in gameplay. The study results are applicable both to automatic level generation and dynamic difficulty adjustment in platformer games.

ACM Classification Keywords
H.5.m. Information Interfaces and Presentation: Miscellaneous; K.8.0 Personal Computing: Games

Author Keywords
Games User Research (GUR); Difficulty; Game Design

INTRODUCTION
Difficulty in platformer games or platformers is easily adjusted because it is influenced by only by a few dimensions like speed or complexity of jumps. The main challenge in these games is to jump onto platforms while avoiding the holes that separate them. Popular platformers like Super Mario Bros. [G2], are well-known by a variety of different players with different skill levels, age groups, and hardware generations. There are currently few guidelines and parameters to aid game designers in the creation of difficulty levels, because we lack a thorough understanding how difficulty is generated and perceived in platformers. The result is that evaluation and manipulation of difficulty is frequently accomplished through continuous player testing, a post-hoc, time-consuming, and resource-intensive mechanism [8] for tweaking difficulty to create an optimal player experience. We report a study of platformer difficulty, where we manipulated *Scroll Speed*, *Target Size*, *Jump Task Complexity*, and *Perspective* in a bespoke game. As we expected, our results show that errors increase as platform size decreases and speed increases. We also found that the relationship for errors and complexity might not be as linear as one could assume because jump task complexity showed triple-jump tasks to be as difficult as double-jump tasks. We also found that vertical and z-axis scrolling both had similar difficulty levels, but are both more difficult than horizontal scrolling (i.e., errors were most prevalent in the forward running scroll (z-axis) and the vertical (y-axis) scroll condition, compared to the horizontal scroll (x-axis)). Alongside error measurements, our measures of self-reported levels of confidence in performance correlates well with quantified measures. To the best of our knowledge, the characterization of the relative difficulty of these factors is novel in the literature, and the implications of these results are useful to level designers creating content for their game, researchers developing automatically generated game levels, and Dynamic Difficulty Adjustment (DDA) system designers. Thus, our study results contribute a better understanding of these parameters for difficulty adjustment and balancing in platformers.

DOI: *http* : //*dx.doi.org*/10.1145/3025453.3025697

RELATED WORK
During game design—while many factors can affect the overall playability of a game—Nacke and Drachen [15] have argued that intentional difficulty in game design (i.e., making tasks harder to accomplish while the interface still remains easy-to-use) is a distinguishing factor from other digital design domains. Malone [13] stated that difficulty is important because it can cause players to believe that their success is uncertain, contributing to in-game challenge, alongside hidden information, randomness, and multiple goals. It has even been argued that the overall game feel is affected [1]. While platformers are particularly affected by difficulty adjustments and are played by many players, balancing game difficulty is a time-consuming challenge for game designers [3]. Researchers have found that changing the levels of difficulty elicits different emotional responses in the player [5] and input

difficulty leads to different cognitive demands [14]. Following this, difficulty balancing has emerged as a subject of study of academics [2, 3, 11], and practising game designers [4, 7, 6]. As a result, academics have attempted to aid the design of games by creating automatically adjusting difficulty [9, 11, 12], exploring parameter adjustments [10] or by changing the shooting mechanism in a First-Person Shooter (FPS) to provide target assistance [2]. Most notably, Wheat et al. [17] collected data on 2D platformer difficulty by testing a self-made game and then analysing the data to inform their classification system for adaptation. The study compared many factors (e.g., slope of curves, enemy difficulty) and provided important pointers for the selection of the difficulty criteria in our study. Following Smith et al. [16], who defined that platformers have avatars, collectibles, movement aids, obstacles, and triggers, we focused specifically on obstacles and movement aids in our study. We refrained from adding triggers or puzzle elements due to their less deterministic nature.

STUDY DESIGN

Our independent variables were selected to coincide with commercial game elements and literature on the platformer genre [16]. These include Scroll Speed, Target Size, Jump Task Complexity, and Scroll Perspective. Dependent measures include different measures of difficulty in platformers such as different types of errors, time on task, and perceived difficulty. To analyze effects of independent variables, we created a bespoke autoscrolling Jump-N-Run platformer game in the Unity 3D game engine that isolates game design strategies that may be used to create difficulty in platformers. Simple platformer games like the one we created may be found on the mobile app stores (e.g., Temple Run [G4] or Super Mario Run [G3]). We created a bespoke game to allow careful control of experimental conditions; inserting new conditions into an existing game maybe confounded with existing expectations, e.g. Mario Bros. does not traditionally have double jump and may be a confound for participants familiar with Mario.

In our bespoke game, players played as a cat that had to run in one direction to avoid a black abyss chasing it (Scroll Speed). To traverse this autoscrolling world, the cat needed to amplify its jumping abilities by bouncing off trampolines (Target). Sometimes a single jump was not enough, and the cat needed to further its reach by bouncing off balloons (Jump Task Complexity), and found itself changing directions (Perspective). For clarity, the x-axis scrolls along the horizontal plane, (i.e. left-to-right), the y-axis on the vertical plane (i.e., up-and-down), and the z-axis on the foreground/background plane. In our game, an error occurred when a player failed to make a jump by missing the targets or fell too far behind the necessary scroll speed. Content in the game was procedurally generated (PCG) and fully randomized for each session. The platforms appear ‘just in time’, using a pseudorandom number generator. We pick platform length and distance within a playable range based on the constraints of the jump which is controlled. The PCG levels provides a test of endurance for players: How long can the player keep moving forward before they miss a jump? If the player failed to make a jump, they lost a life. The game session ended if the player lost all five of their lives or reached the maximum time. Arc length and momentum were controlled. If the jump was executed with error, immediate feedback was given and corrections were not allowed. Throughout the study, each level of the game served to present a different condition. Game levels were pseudo-randomized by the computer to minimize practice effects between levels.

Procedure

Sixteen participants' data were analyzed for our study, 8 male and 8 female. All participants were over 18-years-old, with a mean age of 24 years.

Participants completed nine trials of the game, which presented different game design decisions created by the modification of the game's elements. Each trial was composed of two phases: a practice phase that consisted of five lives, and a gameplay phase that consisted of one minute of play with unlimited lives. After each trial/condition, participants were asked to rate the perceived difficulty of the condition played on a scale from 1 (least difficult) to 9 (most difficult). After they played all nine game levels representing the different levels of the independent variables, participants were interviewed about their in-game experience.

Conditions were counterbalanced to control for ordering and learning effects. We used a partial-block design to test each independent variable. For each independent variable, there were three levels of difficulty: The easiest level was always the baseline level, and the same baseline level was used for all conditions. This baseline included the slowest scroll speed, the largest platform size, only single jumps, and horizontal scroll perspective. For each independent variable (i.e., speed, target size, jump complexity, and scroll perspective) there were two levels of difficulty. Each independent variable was varied independently from the baseline level (i.e., when varying target size to smaller targets, we used the slow rate of scrolling, horizontal scroll, and single jumps only). For example, to test the smallest target size condition, the factors were: smallest target size, slow speed, single jump, and x-axis scroll—in this example only the target size is different from the baseline condition. Without the partial block design, participants would have been required to play many more permutations of the variable resulting in an extremely long session.

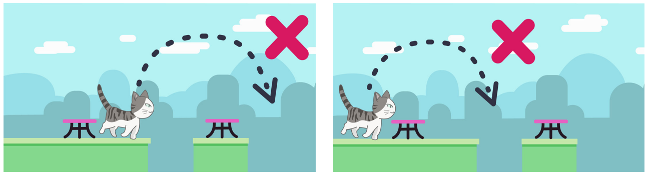
Parameters and Measures

The following values were used for levels of independent variables in game units (in Unity 3D): scroll speed in units/second (10[1], 15, 20), target size (4, 2, 1), jump task complexity (single jump, double jump, triple jump) where targets of additional jumps (balloons) were 1 unit, and perspective (horizontal scrolling, vertical scrolling, or z-scrolling).

We measured players' perceived difficulty rating of each condition on a scale of 1 to 9, easy to difficult. Additionally, the following in-game events were captured:

- Start of a jump (target acquisition)
- Continuation of a complex jump combination
- Errors, distinguished by their cause:

[1] Vertical scrolling was set to 7.5 units/second because 10 units/second was found too difficult during player testing.



(a) Overshooting (b) Undershooting

Figure 1: Concept illustration of the principles of overshooting and undershooting in the bespoke game used in our study.

– Jump was missed (classified as overshoot or undershoot errors as shown in Figure 1)
– Player ran from a platform
– Player was outrun by the camera
– Jump was not continued due to insufficient timing

Hypotheses

Our experimental design tested the following four hypotheses stemming from our review of the literature and discussions with platformer players and designers:

1. Increasing scroll speed increases game difficulty as measured by error rate, time, and player subjective ratings.
2. Decreasing platform size increases game difficulty as measured by error rate, time, and player subjective ratings.
3. Increasing jump complexity increases game difficulty as measured by error rate, time, and player subjective ratings.
4. Scroll perspective does not have a statistically significant effect on game difficulty as measured by error rate, time, and player subjective ratings.

RESULTS

Errors, recorded as a count of the errors the participant made in each condition, are depicted in Figure 2. We see that faster scroll speed, decreasing target size, and changing perspective increase difficulty. Surprisingly, triple jumps were less error prone than double jumps.

Experiment Condition	*Wald* χ^2	*Exp(B)*
Baseline Scroll Speed	49.104*	0.341
Medium Scroll Speed	13.398*	0.635
Baseline Target Size	96.447*	0.236
Medium Target Size	28.916*	0.562
BJT Complexity	42.929*	0.363
DJT Complexity	2.446	1.185
Horizontal Perspective Scroll	69.895*	0.285
Vertical Perspective Scroll	2.959	0.835

Table 1: Parameter estimates of the Poisson Regression (df(1), *$p < 0.001$), BJT= Baseline Jump Task, DJT = Double Jump or Medium Task

The first question we explore is whether our dependent variables are linked to game difficulty. A Poisson regression (see Table 1) was run to determine whether Scroll Speed, Target Size, Jump Task Complexity, and Scroll Perspective were predictive of an increase in error rate. The goodness of fit test passed for the model for a Pearson χ^2 with $t(135) = 223.315, Value/df = 1.648$, indicating that our model fit the data well, that is, the model is predictive. Additionally, an analysis using a Likelihood Ratio χ^2 with $\chi^2(8) = 162.700, p < 0.001)$ indicated that the regression was statistically significant, i.e. that the tested variables were predictive of game difficulty. Test of individual variable effects on difficulty were also significant for each independent variable via Wald χ^2 tests: Scroll Speed (0-2) $\chi^2(2) = 51.461, p < 0.001$, Target Size $\chi^2(2) = 104.896, p < 0.001$, Jump Task Complexity $\chi^2(2) = 62.139, p < 0.001$, and Perspective Scroll $\chi^2(2) = 70.868, p < 0.001$.

A post-hoc analysis of independent variables' effects on dependent variables was conducted. Table 1 includes parameter estimates from the Poisson analysis (Column *Exp(B)*). Interpreting these result, our model predicts the following (based upon our 16 participants): Given that a user makes a certain number of errors in the most difficult (high) condition, then *Exp(B)* represents the fraction of errors for medium and baseline values of that independent variable. This means that for scroll speed, participants would make 0.635 errors in the fastest scroll speed condition and 0.341 errors in the slowest scroll speed condition (the baseline) for every error in the most difficult condition. Of particular interest in Table 1 are parameter estimates for Jump Complexity and Scroll Perspective. Consider, first, Jump Complexity: While the baseline jump is statistically easier than the triple jump, we see no statistically significant difference between the double jump and triple jump, though comparatively double jump results in 1.185 errors per error in the triple jump condition (double jump results in slightly more errors in our predictive model). Furthermore, the horizontal (x-axis) scroll perspective is statistically significantly different than the vertical or z-axis perspective but the difference between vertical and z-axis perspective falls just outside statistical significance (p = 0.085). We recognize that calling attention to comparative differences may be of concern to the reader at this point, but we highlight it here because the data triangulate well with other measures below.

Table 2 summarizes the results of participants' rating of game difficulty (9 = most difficult). We see, again, that the baseline condition (Control) was considered easiest by participants and parity between double-triple jump and between vertical-z-axis scroll perspective.

DISCUSSION

Overall, we find that scroll speed increases difficulty proportionately in a platformer game, decreasing target size results in increasing difficulty, increases in jump complexity initially raise the difficulty (but only until players adapt to what we believe is the rhythm of the game and complexity plateaus), and changes in perspective moderate game difficulty (horizontal/x-axi) scrolling is easier than forward-running/z-axis and vertical/y-axis perspectives).

Our data allows us to reject the null hypotheses for our first two hypotheses. Hypotheses three (jump complexity) is partially

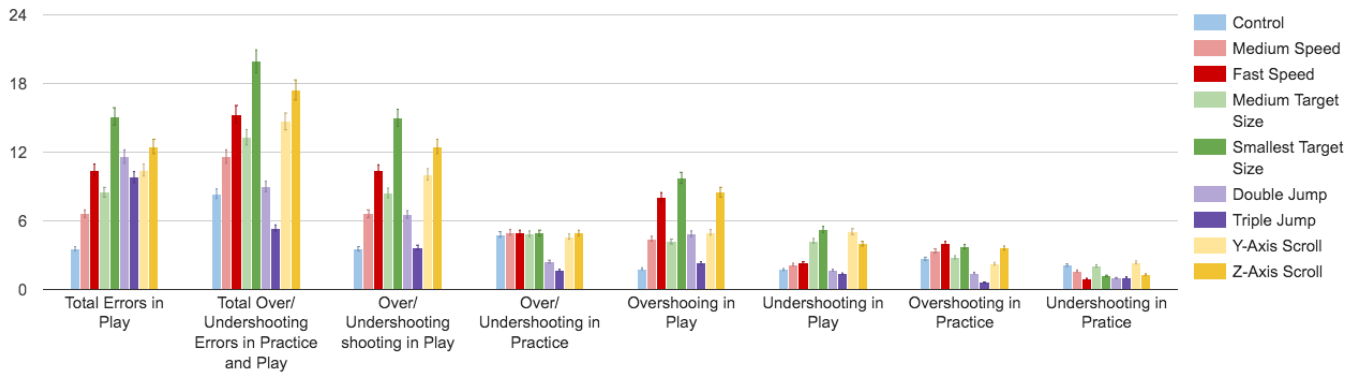


Figure 2: Error count in both practice and play conditions. Summary of errors recorded from the game log for each condition.

Condition	*M*	*SD*
Control	3.13	1.78
Medium Speed	4.06	2.44
Fast Speed	6.00	2.11
Medium Target	4.50	2.16
Small Target	7.09	1.42
Double Jump	6.31	1.85
Triple Jump	5.69	1.74
Vertical Scroll	6.28	1.57
Z-Axis Scroll	6.94	1.65

Table 2: Descriptive statistics of reported difficulty.

supported by our data, but our fourth hypothesis—the lack of effect of perspective—is not supported. Our first two hypotheses, the effects of scroll speed and platform size, were an expected result of speed-accuracy trade-off in task performance. The confirmation of these results validates our experimental design and baseline values for independent variables.

For our third hypothesis—increasing jump complexity increases difficulty—while double jump is, in fact, harder than single jump, triple jump showed no statistically significant increase in difficulty over double jump based on an analysis of error rate. Furthermore, comparatively in both our error rate and in self-reported difficulty, there may be triangulated evidence that triple jump is at least as easy as double jump, a result of lower error rate and lower rated difficulty in the average value of these statistics from our experimental measures. While we do not fully understand why triple jump is no more difficult than double jump, one explanation that merits further investigation is the notion of repeated key sequences becoming easier because players are able to get into an interaction rhythm. The idea of player actions falling into a rhythm, similar to the idea of a musical rhythm was discussed in Smith et. al.'s work on classifying platformer games [16], though not as it intersects with difficulty.

Finally, for our fourth hypothesis, we believed that horizontal scrolling and vertical scrolling would not be statistically significantly different from one another, and that z-axis scroll perspective might result in statistically significant differences. However, we formulated our hypothesis as scrolling having no statistically significant effect on difficulty, and, based on our data, we cannot support this hypothesis. We found that y-axis (vertical) scrolling and z-axis scrolling were statistically more difficult than horizontal scrolling, but, interestingly and counter-intuitively, z-axis and vertical scrolling had similar difficulty levels. Issues of scrolling and difficulty have implications for the design of both platformer games and other movement-based games. Scrolling perspective is, at heart, an issue of camera placement, so the increase in difficulty can be manipulated by allowing a user to control the camera; however, providing users with more variables to control could increase the difficulty, similar to perspective puzzler platformers (e.g., *Fez* [G1] or the 2D mobile game *Monument Valley* [G5]). Beyond scrolling perspective, our findings can apply to other game genres as well. For example, a shooter game can balance target size based on target-shooting difficulty, similar to work done by Bateman [2] that employed target assistance. Finally, our study results have interesting conclusions for both automatic level design and Dynamic Difficulty Adjustment (DDA). Currently DDA systems focus on mediating enemy characters, the appearance of items, the availability of resources (e.g., health resources), and other small tweaks to the design of the game. Another potential way to add or mediate difficulty may be by automatically manipulating the level design (or camera perspective) by incremental factors.

CONCLUSION

This paper is a focused exploration of how platformer game parameters can be used to manipulate difficulty in game design. Our data reveal that Scroll Speed, Target Size, Jump Complexity, and Scrolling Perspective all affect difficulty. Our study results have implications for the understanding of how these game design decision affect difficulty and player experience. Additionally, findings of this study may be useful for the design and selection of variables in dynamic difficulty adjustment systems and automatic level design.

ACKNOWLEDGEMENTS

The authors thank the support given by NSERC research grants and scholarships. Lennart Nacke also thanks SSHRC. Thank you Eddie Sheerer and Marim Ganaba, the artists.

REFERENCES

1. November 23, 2007. Game Feel: The Secret Ingredient. (November 23, 2007). http://www.gamasutra.com/view/feature/130734/game_feel_the_secret_ingredient.php
2. Scott Bateman, Regan L. Mandryk, Tadeusz Stach, and Carl Gutwin. 2011. Target assistance for subtly balancing competitive play. *Proceedings of the 2011 annual conference on Human factors in computing systems - CHI '11* (2011), 2355. DOI: http://dx.doi.org/10.1145/1978942.1979287
3. Glen Berseth, M. Brandon Haworth, Mubbasir Kapadia, and Petros Faloutsos. 2014. Characterizing and optimizing game level difficulty. *Proceedings of the Seventh International Conference on Motion in Games - MIG '14* 2 (2014), 153–160. DOI: http://dx.doi.org/10.1145/2668064.2668100
4. Daniel Boutros. 2008. Difficulty is Difficult: Designing for Hard Modes in Games. (2008). http://www.gamasutra.com/view/feature/3787/difficulty
5. Guillaume Chanel, Cyril Rebetez, Mireille Bétrancourt, and Thierry Pun. 2008. Boredom, engagement and anxiety as indicators for adaptation to difficulty in games. *Proceedings of the 12th international conference on Entertainment and media in the ubiquitous era - MindTrek '08* (2008), 13. DOI: http://dx.doi.org/10.1145/1457199.1457203
6. Greg Costikyan. 2013. *Uncertainty in games*. The MIT Press, Cambridge, Massachusetts.
7. Kris Graft. 2011. MIGS 2011: Redefining Challenge In Games Can Push Artistic Boundaries, Says Rohrer. (2011). http://www.gamasutra.com/view/news/128014/MIGS
8. Carl Gutwin and Saul Greenberg. 2000. The mechanics of collaboration: Developing low cost usability evaluation methods for shared workspaces. In *Enabling Technologies: Infrastructure for Collaborative Enterprises, 2000.(WET ICE 2000). Proeedings. IEEE 9th International Workshops on*. IEEE, 98–103. http://hci.usask.ca/publications/2000/mechanics-wetice00.pdf
9. Robin Hunicke. 2005. The case for dynamic difficulty adjustment in games. *ACE '05 Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology* (2005), 429–433. DOI: http://dx.doi.org/10.1145/1178477.1178573
10. Aaron Isaksen, Dan Gopstein, and Andy Nealen. 2015. Exploring game space using survival analysis. *Foundations of Digital Games* (2015).
11. Martin Jennings-Teats, Gillian Smith, and Noah Wardrip-Fruin. 2010. Polymorph: A model for dynamic level generation. In *Proceedings of the 6th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, AIIDE 2010*. 138–143. https://www.aaai.org/ocs/index.php/AIIDE/AIIDE10/paper/viewFile/2150/2558
12. Changchun Liu, Pramila Agrawal, Nilanjan Sarkar, and Shuo Chen. 2009. Dynamic Difficulty Adjustment in Computer Games Through Real-Time Anxiety-Based Affective Feedback. *International Journal of Human-Computer Interaction* 25, 6 (2009), 506–529. DOI: http://dx.doi.org/10.1080/10447310902963944
13. Tomas W. Malone. 1981. Heuristics for designing enjoyable user interfaces: Lessons from computer games. *Association For Computing Machinery* (1981). DOI: http://dx.doi.org/10.1145/800049.801756
14. Lennart E. Nacke. 2010. Wiimote vs. Controller : Electroencephalographic Measurement of Affective Gameplay Interaction. *Proceedings of the International Academic Conference on the Future of Game Design and Technology* (2010), 159–166. DOI: http://dx.doi.org/10.1145/1920778.1920801
15. Lennart E. Nacke and Anders Drachen. 2011. Towards a Framework of Player Experience Research. In *Proceedings of the Second International Workshop on Evaluating Player Experience in Games at FDG 2011, Bordeaux, France*. http://hci.usask.ca/uploads/230-NackeDrachenPXFramework.pdf
16. Gillian Smith, Mee Cha, and Jim Whitehead. 2008. A framework for analysis of 2D platformer levels. *Proceedings of the 2008 ACM SIGGRAPH symposium on Video games Sandbox 08* 1, 212 (2008), 75. DOI: http://dx.doi.org/10.1145/1401843.1401858
17. Daniel Wheat, Martin Masek, Chiou Peng Lam, and Philip Hingston. 2016. Modeling perceived difficulty in game levels. *Proceedings of the Australasian Computer Science Week Multiconference on - ACSW '16* (2016), 1–8. DOI:http://dx.doi.org/10.1145/2843043.2843478

LUDOGRAPHY

1. Polytron Corporation. 2012. *Fez*. Game [Computer, Play Station 3,4, Xbox 360]. (2012). Trapdoor, Montreal, Canada.
2. Shigeru Miyamoto. 1985. *Super Mario Bros.* Game [NES]. (1985). Nintendo, Kyoto Japan.
3. Shigeru Miyamoto. December 2016. *Super Mario Run*. Game [iOS, Android]. (December 2016). Nintendo, Kyoto Japan.
4. Kiril Tchangov. 2011-2013. *Temple Run*. Game [iOS, Android, Windows Phone 8]. (2011-2013). Imangi Studios.
5. Ustwo. 2014. *Monument Valley*. Game [Android, iOS, Windows Phone. (2014).