Smart Games:

Emergent Game Design

Keaton Johnstone
http://research.frontier-studios.net

2013

School of Design

Victoria University of Wellington

PO Box 600

Wellington 6140

New Zealand

A thesis submitted for a degree of

Master of Design of the Victoria University of Wellington

Abstract

Gaming is a rapidly growing recreational activity. Over the last two decades we have seen a proliferation of games across all genres and user demographics. The consumption of content in modern games has grown as a result of the advancing technology available to game designers, as well as the growing expectations of the audience. The need for increasing the value of the content and expanding potential audience grows every day. Using dynamic adjustments to the underlying game systems as the player experiences the game, this paper will demonstrate how using emergent principles combined with other foundational systems, such as evolutionary algorithms, can increase the value of the content created for any game. This added value is of benefit to both those who have made the game, and the users who experience the results.

Contents

Introduction

The purpose of this thesis is to explore the mechanics of and the effectiveness of an emergent dynamic game, created through the use of a distributed system that can be adapted to fit most computer games. This system analyzes the game being played, and modifies the rules of play according to the behaviours of the player. This particular system uses the power of emergence and evolution as the foundation, deployed through computer programming, for the platform.

To enable this experiment, a testing suite was developed to allow a test subject to play a series of games, while capturing data at specific moments in time throughout the testing, to reveal trends in game play. This same data is used integrated into the game the mechanics as it is played in a fashion that suits the player and their method of play at the time. This technique creates a system of mechanics that enhances gameplay, while also uncovering additional information that may be used for subsequent. This should aid in the ability of other researchers and game designers to integrate such methods into their own games.

This particular approach to game design could be of great interest to the current generation of games designers. Advances in technology and hardware drive the development of richer game environments, resulting in an ever increasing need for content - in both quantity and quality - to fill these environments. In maximising the usage of the content being created, as well as expanding a game’s appeal and thereby its potential audience, we introduce significant benefits. The end player will ultimately gain a better experience while playing the game, and overall this should result in a win-win situation for all parties involved.

Philosophy of Approach

Game Design

Games appeal to different audiences in different ways; just as there are different genres of movies and books, games also fall into different genres. These genres help differentiate types of games, such as action games which usually involve shooting objects, and sports games which usually try to emulate a particular real sport. When games are designed, the goal is usually to guide a player through a predetermined set of experiences, determining how players should play the game, and offering an interesting breadth of experiences to satiate the desires of the player. This focus on the player means that the design of the game is often dictated by the particular audience that game is intended for.

A recent development in game design is the ability to have games with no distinct goal presented to the player. These are usually called sandbox games. This lack of clearly outlined goals puts the player in control of how to play the game, where they would like to go, and to experience a myriad of immersive features. The loose nature of control in these particular types of games means the designer gives up much of the control regarding how players should be playing their game in exchange for controlling how the emergent systems interact. As the player is provided more choices, the predictability of what the player will choose to do next decreases with each choice; so focusing on the choices returns control back to the designer. The more control a designer has over the player, the less freedom the player has; just as the more freedom the player has, the less control the designer has. It would appear that moving to a design foundation that allows more control to the designer, and more freedom to the player, is an area of untapped potential for new game design.

Emergence

Emergence is a concept that is important to consider when building a game; games are often constructed using many underlying rules (systems), and when a player interacts with the game these underlying systems can create varying events and outcomes within the game. In 2000, Marc LeBlanc, in his GDC (“Game Developers Conference”) presentation titled "Formal Design Tools: Emergent Complexity, Emergent Narrative", discusses these important qualities for emergent game design (Marc LeBlanc, 2000). These ideas are further discussed and summarized in a short paper by Noah Falstein, titled “Emergent Complexity”. The systems discussed and used in this thesis are developed from these principles.

When many simple systems are constructed, each with their own behaviours and self describing rules, and subsequently they are amalgamated, they merge to form much larger complex systems. The complexity of the resultant system is a factor of how many subsystems are interacting and how complex each of these systems is. Thus this newly formed system is emergent - we see the appearance of new properties that can only result from the interaction of the individual systems. This type of behavior is represented in natural processes such as the origins of life, and our concept of physics, both of which emerge from the fundamental building blocks of our universe.

Flow & DDA

Flow, as coined by psychologist Mihály Csíkszentmihályi, is the entrance into a state of fully focused attention onto a particular activity. It is synonymous with being *in the groove*, where a person is fully immersed in what they are doing. This is of particular importance to game designers who desire to captivate their players and encourage them to continue playing their game. To achieve a state of flow game designers use a simple formula of "Challenge vs. Skill", where the challenges presented within the game need to match the player's skill level: too much challenge, and the player will be frustrated; not enough and the player will become bored; just enough of both however will allow the player to become fully immersed in their particular activity.

In Jenova Chen's popular thesis "Flow in Games", the relationship between video games and flow is demonstrated to be an important component for effective game design. Chen highlights the importance of using dynamic difficulty adjustment in combination with core elements of flow to augment the player experience (Chen, 2006).

Dynamic difficulty adjustment (DDA) is a system within a game's design that enables the game to modify the experience based on the player's current state. The purpose of DDA is to allow the player to continue to enjoy the game based upon their skill level. DDA often tries to challenge the player slightly to keep them more interested in play and to avoid boredom.

The concept of DDA is not a new one in game development, and has been used in many popular video game titles, such as:

- Elder Scrolls Oblivion (Bethesda, 2006): scaling each encounter based on player's current level.
- Homeworld (Sierra Entertainment, 1999): AI difficulty is based on the size of the player's fleet on each mission start.
- Mario Kart (Nintendo, 1992-2011): players at the back of the starting line receive more speed & better items; those at the front, less.

Each game offers a different approach towards using DDA; however they all employ the same concept of adjusting the game's difficulty to the player's skill level as they play the game. The use of DDA is an important tool used to encourage the player to be in a better position to achieve *flow*.

The challenge vs skill formula works extremely well when choosing to view games as simple *goals with obstacles*; however, an often overlooked component of *flow* is the intrinsic reward found within the activity itself. The player needs to feel rewarded by simply partaking in the activity, and not the just through the act of completing the activity. Thus, the obstacles become an important component of developing an intrinsic care for the activity. The intent is to encourage in the player an interest in the act of playing itself, independent of skill and challenge.

Evolution

The use of evolutionary algorithms in this experiment allows the system to benefit from the variety of attributes related to genetic design principles. Genetics can be described as the passing of particular traits from one generation to the next, with a feedback loop that enables selection of only the most optimal traits for the particular environment, and arresting those traits that are not beneficial.

The evolutionary principles used offer various effects such as: *Convergence*, *Adaptability, and Drift*.

- *Convergence* is the ability to have two separate paths independently arrive at the same conclusion;
- *Adaptation* is a trait in which the system that is evolving is able to change itself to be better suited to its given context;
- *Drift* enables the ability for the given system to naturally become variant.

All of these traits are enabled through the use of a Genetic Algorithm (GA), which copies the biological process of evolution and implements them into software design.

The algorithms, or procedures, used to execute this concept will contain a data structure that allows particular traits to be varied, independently of one another. These traits will then be analyzed and modified in each iteration of the algorithm. The process of analyzing the efficiency of a given permutation of the data is typically referred to as a fitness function. The fitness function created for this system requires extreme care and attention as it becomes the backbone for changes and alterations in the evolutionary system; thus making it a primary component of design.

Methodology

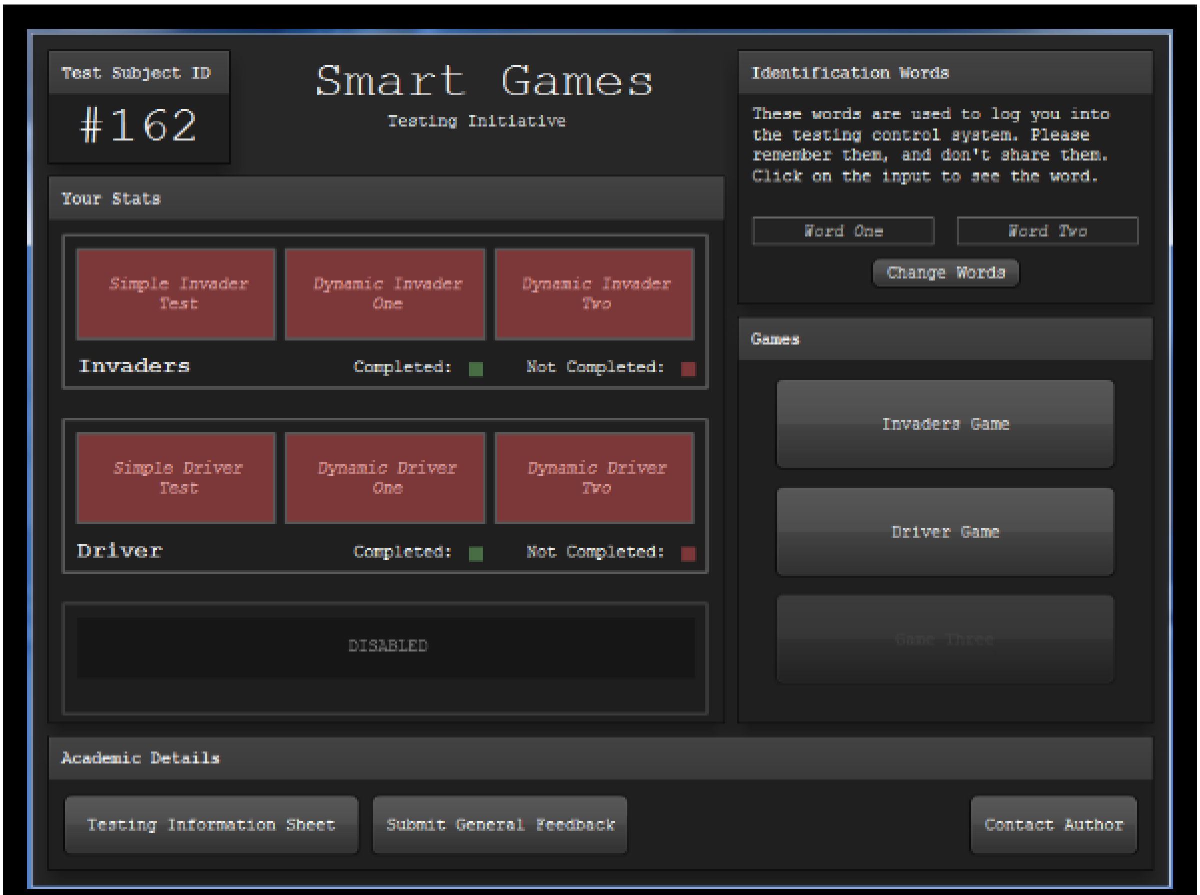
Testing Procedure

The testing procedure for this experiment involved allowing a group of volunteers (friends, family and colleagues) to download the software from the testing website. This application enabled anonymous registration, which assigned each tester a unique combination of words to allow them to re-authenticate at a later date if they desired. When the test subject was authenticated as a test subject, they were able to choose which game they wished to play (Fig. i).

Each test subject always performed the same amount of initial training by being given the initial introductory static test: a three-round static test, with pre-defined escalation of difficulty, typical of a normal game difficulty. After completing the static test, a second test was chosen at random; either a three-round dynamic test, or a six-round dynamic test. Each of these tests had the emergent system enabled. After a test was completed, it could not be completed again, until all other tests for that particular game were completed. The randomized administration of tests aids in removing the effect that play order (ie. player familiarity) could have on the final data.

At the end of each round the test subject answered three questions to determine if they had a greater focus on and more attention devoted to the game, indicating some of the qualities typical of approaching a state of *flow*.

After completing the test, all data was submitted to the research server and stored in a database; this included the data the game recorded during play, and the results of the questions. Once all three tests had been completed, the cycle of testing began again from the start.



Testing Application Dashboard

Fig i

The Games

The games created for testing needed to be generic in nature yet invoke a familiarity in the player by being reminiscent of historically popular games. Using the mechanics inherited from these 'traditional' games allows for more objectivity when it comes to evaluation, as it will allow evaluation of the modified mechanisms that differ from tradition. The game genres for the two designed games were chosen to provide a more diverse platform of mechanics; this is to demonstrate the effectiveness of the emergent mechanics across a greater variety of foundational game mechanics. This will help identify if the emergent mechanics operate similarly in two quite different games.

The target audience for this experiment was anyone who enjoys playing small/casual games. In addition, the style of the game's aesthetics, although important for engaging players throughout the testing session, was required to be simple enough as to not influence the evaluation. Often aesthetic opinion of a game can heavily influence the initial experience. With simple visuals and limited audio, the hope was to avoid any discrimination based on aesthetics.

The two games created for this study and a short description of how they play follows:

Invaders

Invaders is a simple shooting game where the player occupies the bottom of the screen, represented as a blue rectangle. The object is for the player to survive the duration of the round without losing all of their health. The player can move left or right, and also shoot straight upwards, using the left and right arrows and spacebar on the keyboard. The enemies are located above the player, moving from one side of the screen to the other, shooting straight down at the player.

The player starts with 100 health points, and loses points each time they are hit by an enemy shot. The player has the opportunity to regain points for each enemy they shoot. If the player’s health points reduce to 0, the player must restart that round.

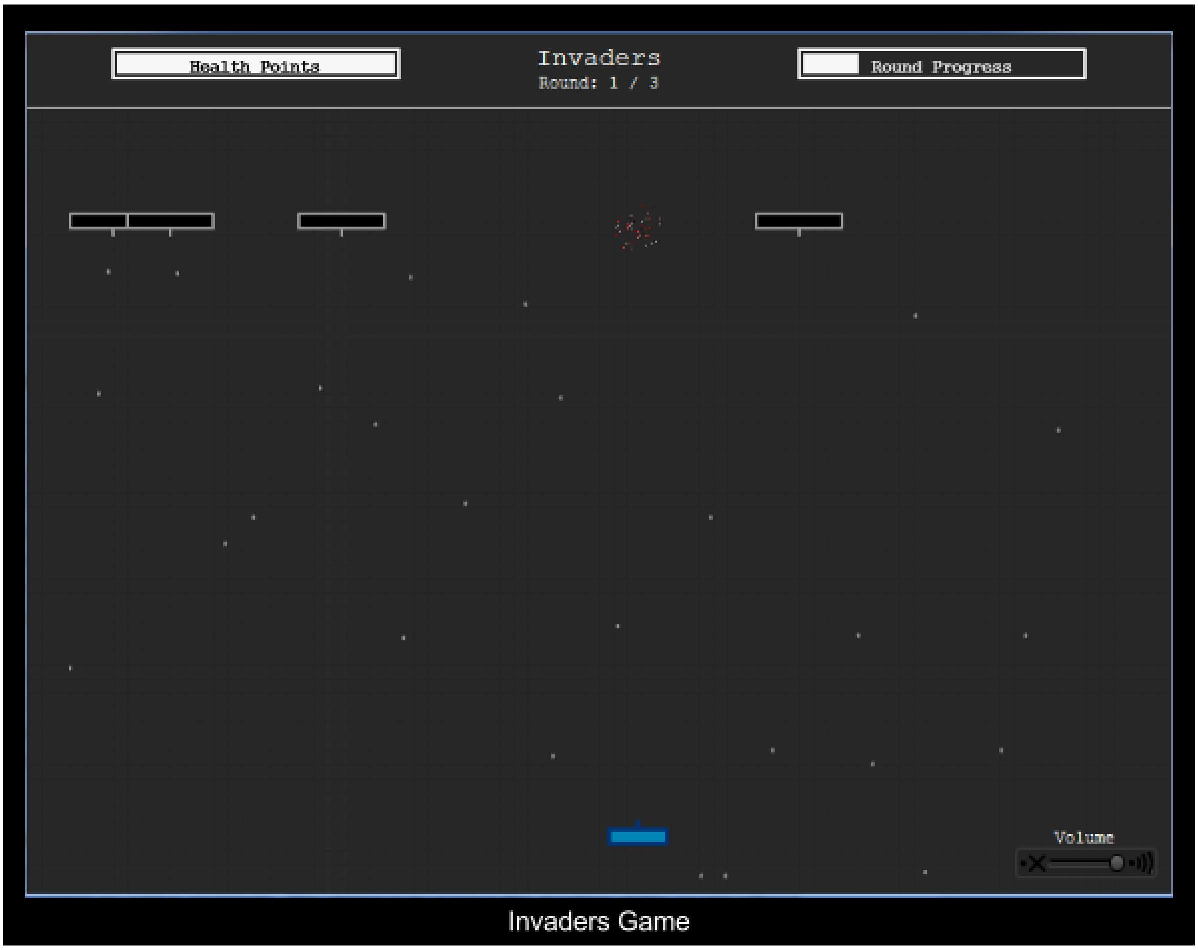


Fig ii

Driver

Driver is a simple driving simulation, where the player must travel around the track for 3 laps. The vehicle is controlled with the arrow keys: left & right arrow keys steer the vehicle left or right and the up & down arrow keys change the throttle setting. A high throttle setting will accelerate the car as fast as possible, a low setting will provide very little acceleration.

The player's vehicle is affected by the surface they are driving on: dirt, grass and oil will each affect the vehicle’s traction, the latter more so than the former. The vehicle must pass both checkpoints, indicated by the flags and the checkered pattern on the track, for a lap to be complete.

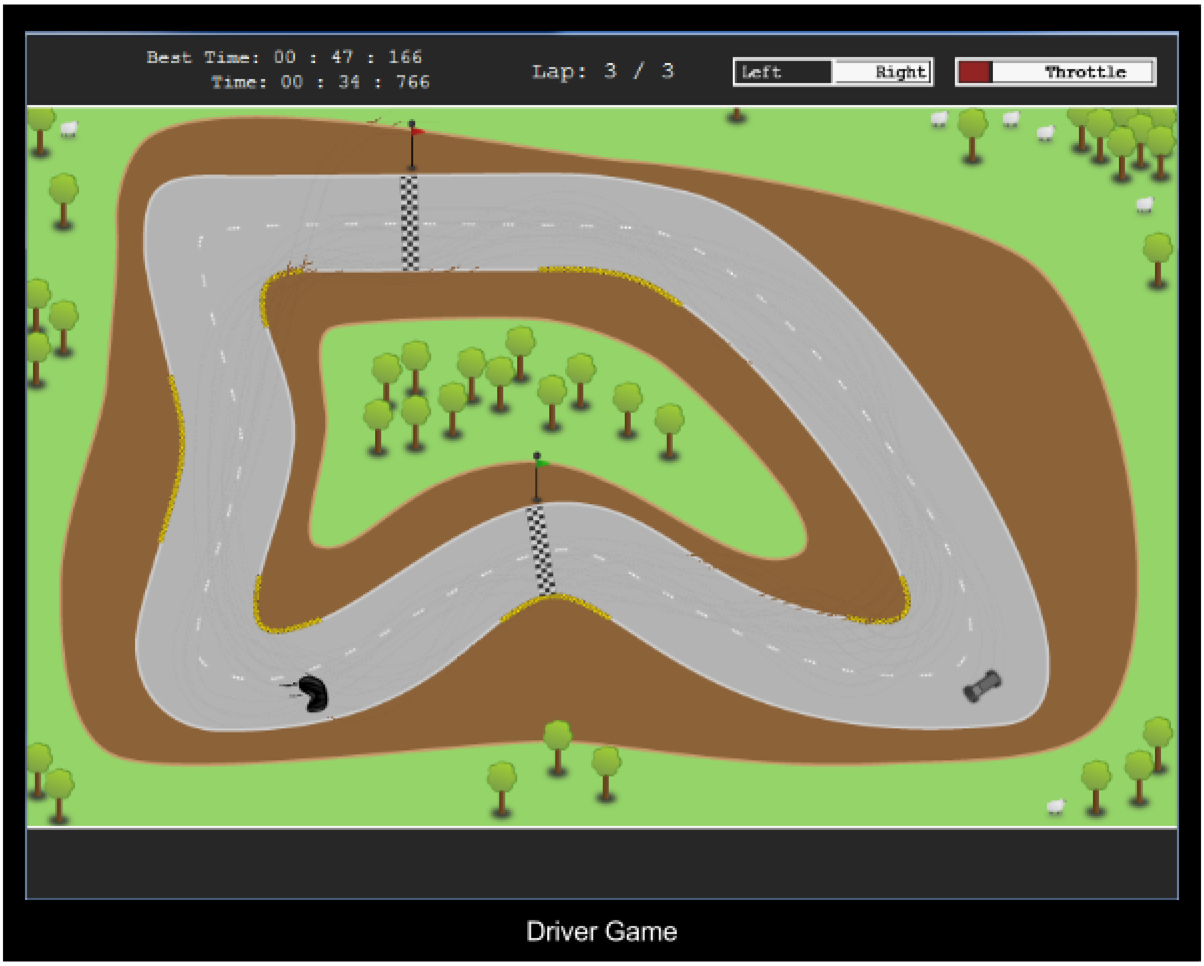


Fig iii

The games were designed by extracting the most common elements of original games within the particular genre. For *Invaders*, the original “Space Invaders” released by Taito (Taito, 1978) was used; for *Driver* the “Indy” series by Atari (Atari, 1975) was used as a foundation, incorporating elements also found in later top down racing games, such as “Grand Theft Auto” by DMA Design (DMA Design, 1997). The original games of each genre often offered little more than the core gameplay mechanics for that particular genre. These primary mechanics have resulted in very little actual design for the core gameplay experience.

A particular design constraint imposed on each game in the test suite was the need to employ round based game play to allow for specific intervals of testing/data collection. This allowed each round to be evaluated independently. The round based structure also played a role in facilitating the emergent mechanics, allowing them to be processed at discrete intervals. This mechanism acted as our generational interval for the designed system; this is when the emergent mechanics can be processed, and evaluated for effectiveness.

The Emergent Architecture

The underlying architecture upon which the games operate and create decisions needed to be consistent across both games to allow for a more comparable statistical evaluation during subsequent data analysis. This consistency also provides a clear framework for others to follow. During any given round of gameplay, all games stored their statistical data in the same format, at the same intervals. The same evaluation system was used in all games.

Each game inherited from the same code base; this exposed the same components within each game. Using the same base code in both games also enabled a quicker development process as inheritance allows for code reuse. This architecture used in each game represented the combination of all underlying ideas: emergence, flow, and evolution; all employed within the same particular task.

The feedback process that allowed for the emergent gameplay elements to be calculated and implemented is as follows: (Fig iv)

1. The player plays the game
2. The statistics are recorded
3. The statistics are analyzed by the emergent coding structures
4. The game rules are modified for next play

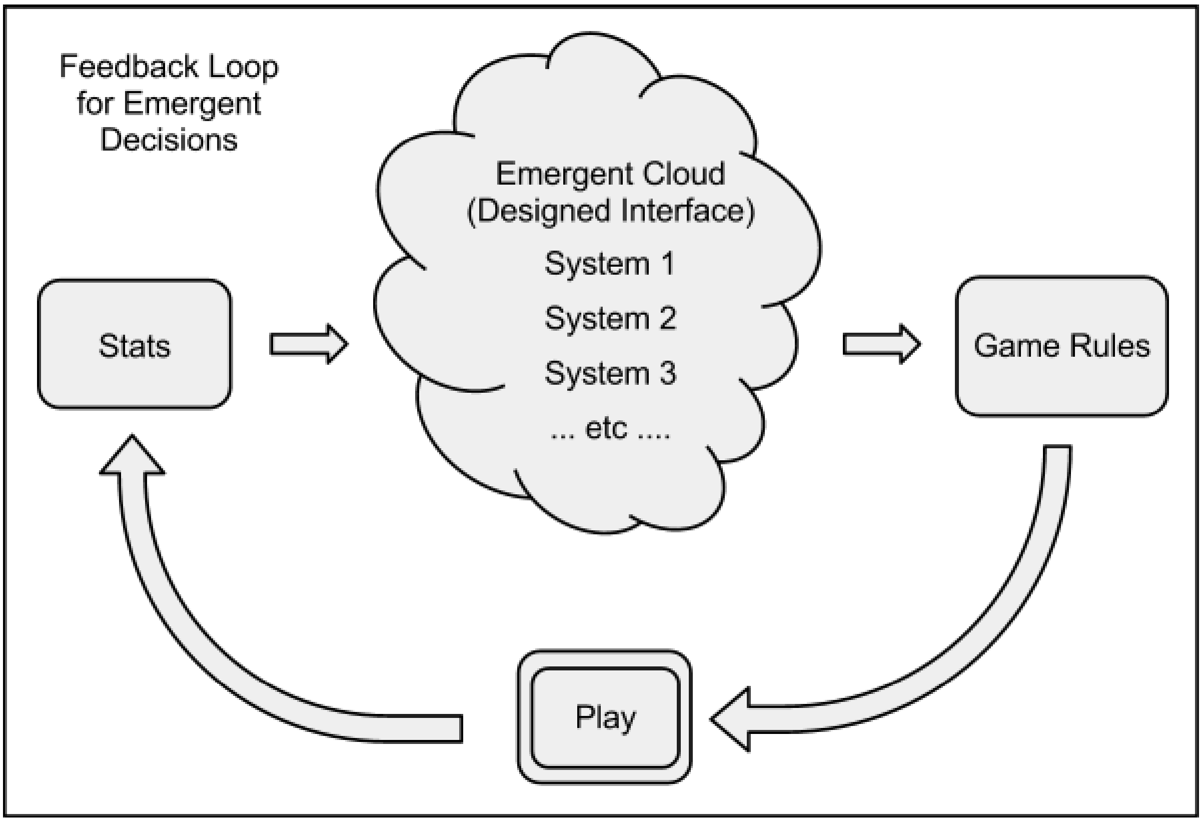


Fig iv

The process the game follows serves to evaluate a series of calculations that move the game state from one pre-determined period of time to the next. These calculations are comprised of several independent data measurements, such as for Invaders: player movement delta, invader spawn times and for Driver: player's position, speed and direction. Each of these measurements are taken at each point where the game evaluates its current state. This can be viewed as a derivative function as it simply moves the equation along from the previous state. Each of these calculations rely on many variables that influence the outcome of each "time step" of the game engine. The variables are exposed to enable modifications of sections of the calculations; this allows the game to modify its operations even as the game is being played. This procedural nature of the game allows for these variables to be tweaked or influenced before playing the game, or during.

Typically in game development, this process of modification is completed before shipping a game, and determines the feel and nature of the game mechanics - values such as the influence of gravity, or the speed a bullet flies through the air. During initial playtesting of the games, these values are individually tested to find their limits. This is done by increasing or decreasing the particular value repeatedly, until the game is found to be in a "broken" state. A "broken" state would be viewed as either completely "un-fun" or just not practical - such as moving too quickly, or too slowly. The lower end of the middle ground of these limits are then used to determine the static testing values for the game.

In contrast, each of the iterations of the emergent system modifies each of these values to change the feel and the nature of the next round of game play. The emergent nature of this design means that each system within the emergent decision cloud is created independently, and will only influence the potential outcome . The individuality of each system, as well as the potential influence upon the outcome of the new modified game rules, demonstrate the emergence of the system. The influence of each decision along with the nature of the decision being made ultimately becomes the new domain for the game design to occur.

In the testing suite, the methodology in regards to these emergent systems is to identify whether the system should either encourage the current activity or attempt to prevent it. The two sides of the coin represent the two aspects of flow that we desire to address in this thesis. The player either needs to have the difficulty of the current encounter modified to match their skill level; or they have decided to play the game in a particular fashion. Each system designed for these games fight a tough battle along both these lines. The assumption initially made is that any large ratios of difference represent intentional methods of play, where small ratios demonstrate only a variation in skill.

Based on these key assumptions, the specifications were then progressively modified for each system through player feedback, further analysis of the data recorded, and extensive tweaking. During this process, new systems had to be created to fill gaps that were found through initial playtesting, with some elements being split-up to allow for more simplicity and independence amongst systems. These areas represented aspects of the game that were either too complex or missing in entirety.

While designing each system, an analogy was created to simplify the design focus of each system. Some of the systems included within the Invader game are listed below:

- *Rapid Fire Control System:* Restrict wasted shots if the player is found to be firing bullets needlessly; analyse the accuracy to determine if the fire rate should be modified; improve player skills.

- *Pace Identification System:* Review player movement in reference to shots, to identify a defensive or aggressive player (defensive players avoid being hit, and don't fire back; offensive players move for the shot); accentuate the chosen play style.

- *Meat-shield Monitor:* If the player demonstrates the ability to be hit often, and still recover their health, the game will increase the values used to intensify the activity; lower the values if the player is having difficulty recovering health.

- *Farm Prevention System:* If the player is killing the invaders too quickly after they enter the screen, the system increases the number of invaders, as well as their distribution, to make it difficult for the player to stay on the edges of the screen and kill the invaders easily.

Having the decision cloud separate from the main game process, as well as utilizing round-based generations for the system, benefits the game by enabling the extraction of the studied system from the game in a way the enables the game to run entirely without any influence from the emergent interface. This technique was specifically chosen to explore the idea of attaching this system to an already existing game or game design, without large amounts of integration, but simply a modular approach to design.

ESM

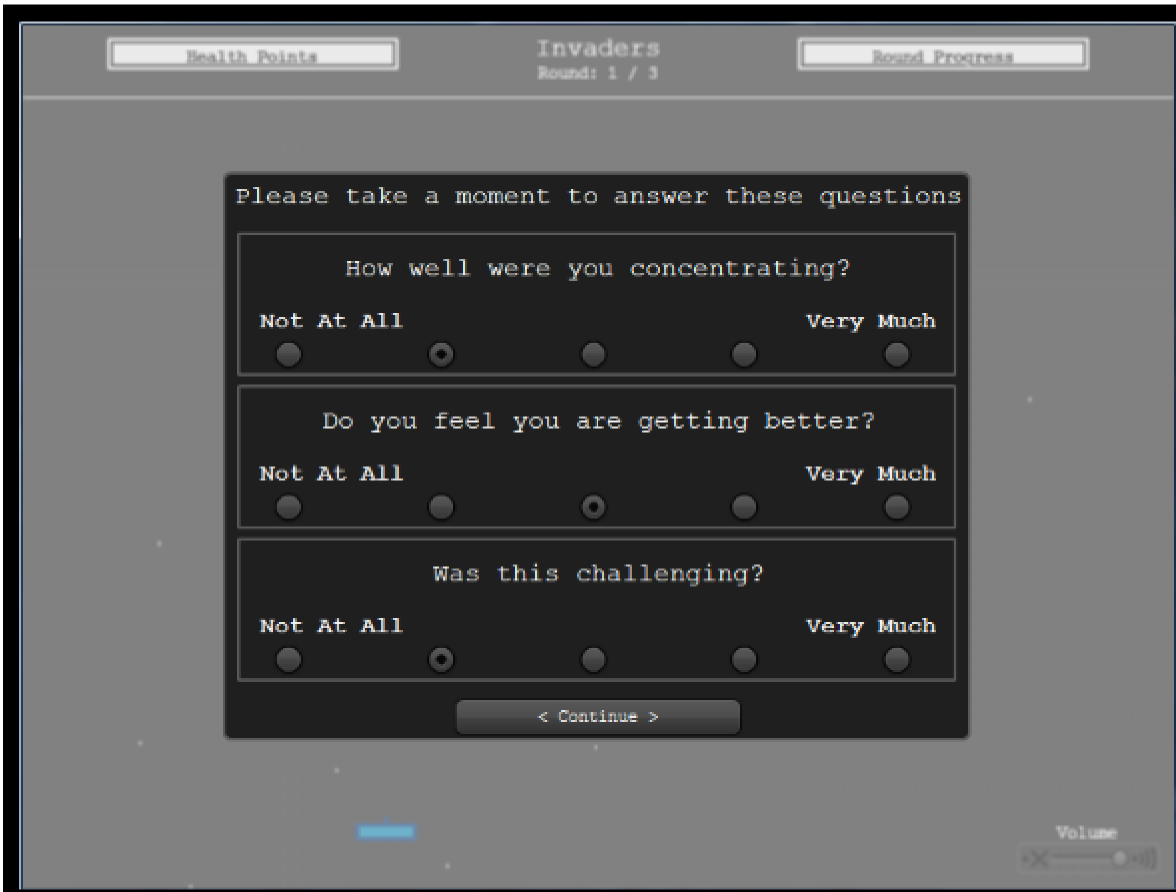
The Experience Sampling Method (ESM) was used as a basis for the design of the experiment by performing a random data sampling on a regular basis, attempting to capture as many player stats as possible. The sampling of data over-extends the breadth of data that is originally envisioned as being required, as it then allows more analysis at a later time, and exposes the relationships between unseen variables. After capturing data from initial play testing, the emergent mechanics were then built into the game - created in response to the patterns identified from the sampling data. The experience sampling method is typically used on a personal self evaluation basis; however in this case we have a focus on the games themselves, rather than the people playing them. With the technology available, we can implement this on a much greater scale within the games; sampling the current game state often and saving all data for analysis, with no discernible effect on the test subject.

Technology has lent itself to this expanded method of sampling; it allows us to redirect the sampling towards the application itself rather than the user. This enables the ability to use the power of ESM data analysis on subjects that were not originally intended to have an ESM administered. The questions presented to the test subject at the end of each round were taken straight from the ESM that was used in Mihály Csíkszentmihályi's flow research; each picked to demonstrate the test subject's experience of "fun", "caring" and "focus".

Analysis of Variance

The analysis of variance is a statistical method of comparing multiple data sets. The technique for our methods will be to capitalize on the ESM testing method and the data it records. Each successive round acts as a separate data set; these data sets are then compared. The difference between the data values in each successive round represents the statistical variance resulting from the changing variables used in the test; these are the differences in the mechanics evaluated by the game. These evaluated differences represent a delta of experience for each test.

Comparing each tests variance amongst rounds then provides the indicator of a successful positive change in the playing experience; When the player is presented with the questions at the end of the round (fig v), they are able to select on a scale of 1 to 5, 1 representing “Strongly Disagree” and 5 representing “Strongly Agree”; If a player chooses value 3 on the scaled evaluation at the end of the round, then chooses 4 in the next round, the variance is described as +1. This identifies when a particular sampled experience has increased in value. Overall, if these values show an increase for each round in the dynamic test and differ from the static control test, we are able to identify a positive result for the system. This would indicate that the player has had a more fulfilling experience. This can be used as our fitness function for the study.



End of Round Evaluation

fig v

Analysis

Creating the initial game platform framework was completed using an agile development environment, which allows for achieving short term milestones on a regular basis while always having a playable version of the game. A conventional game would sit as 'feature complete' at this playable stage, and would require only minor tweaking and adjustments to the design of the overall experience. However, in this experiment, this is the point at which the interesting design of the experience for the test games actually began.

The process of creating these simple games demonstrated an interesting shift in design focus. The design of linear games begins to inherit design concepts that are used everyday when doing sandbox style, or emergent games. In linear games the designer is often dictating how the player should experience each encounter; perhaps offering choices, but each resulting outcome is still a specifically designed experience within the game. The sandbox design shifts from asking “how *is* the player doing this?”, to “how *can* the player do this?” Within a sandbox, a player can choose how they would like to play the game, and the game designer is simply in control of the tools available to the player, while controlling the overall experience within the game. Using a more emergent game design process to create each individual experience, the designer is in charge of defining the limits of the design, and ensuring there are multiple systems that are independently capable of altering the experience.

The individual characteristics of each controlling subsystem within the emergent decision cloud is the dictating component of design. Throughout the development phase, through analysis of the data, the patterns of play were distinguishable. For each of these distinct patterns, the emergent systems (some of which were outlined in the previous section) were created as responses. After initial implementation of these systems, it was possible to measure a change in the patterns of play - and not always for the better. It was found that the fewer variables that the emergent system would modify, the more manageable it was to isolate bugs and other unwanted behaviours in each system; behaviors such as extremes of different types being reached, creating bad feedback in the system, thereby encouraging even more bias. In addressing this, the resultant benefit was that the simpler each system was, the easier it was to design. An effort was made to ensure the statistics being analyzed were of the same domain that the system would enact change within. This meant that each system could detect when a limit was being reached, perhaps by its own influence, and attempt to restrict any bad feedback.

An important conclusion that can be seen within the comparison of the data, is that more aspects of the game content are being consumed by the more dynamic systems. Although one could ensure that the static initial tests explored all parts of the game, each system was designed for all audiences, regardless of skill level. The initial tests had to accommodate the good players as well as the bad ones; however, the dynamic tests were able to adapt the game to the players play style, keeping it simpler for those who needed it simple, and making it more difficult for those who wanted more challenge.

Analysis of a collected data set seems to demonstrate the effectiveness of the system. The comparison of the average movement within the three Invader tests demonstrate a lengthening of time of each round by roughly 210% (Fig vi). The length of time of each round is determined based on when the player completes the objective. It seems apparent that with a more aggressive system modifying the game mechanics, we see a dramatic increase in the length of each play session.

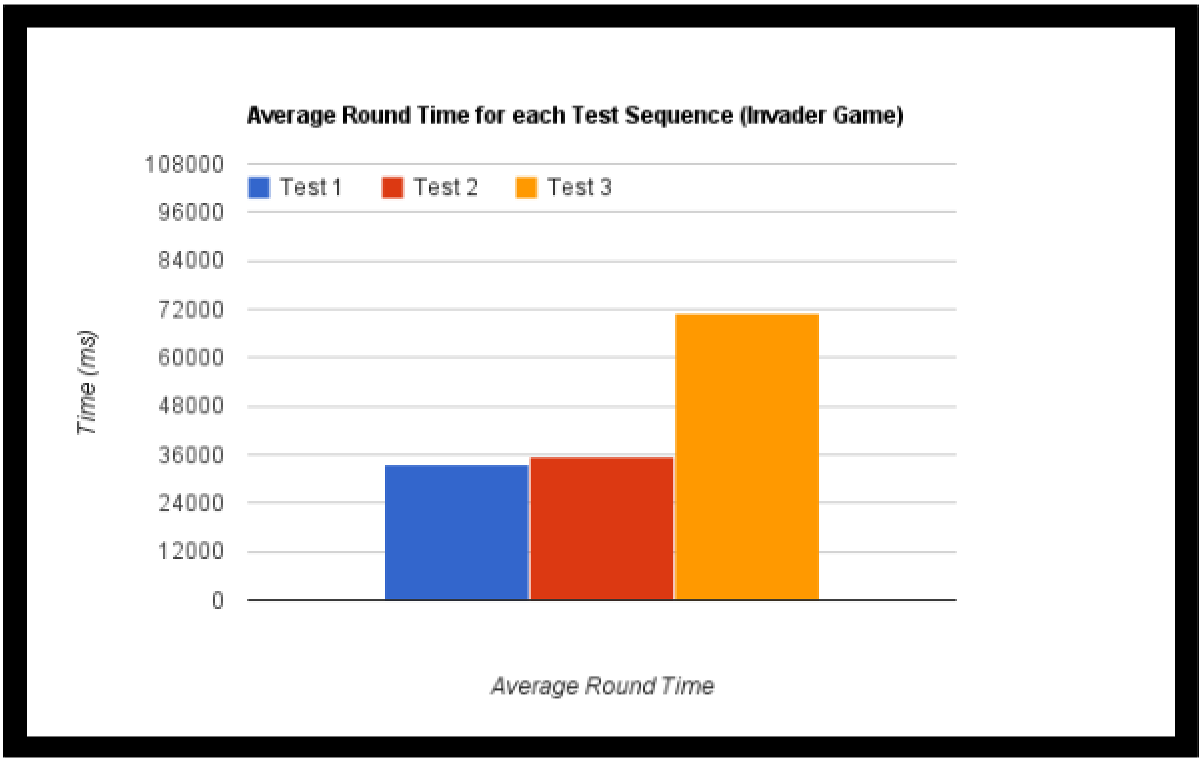


Fig vi

A look at the average movement of each player during each round, also demonstrates differences in the play for each test (Fig vii). Each sharp change in the data shows where players completed the round; this occurs because the data values these players contribute to the average are no longer valid once they complete the test. This accounts for the sharp upward turn for the static test “Test 1”. It is quite clear that the average movement shows large statistical differences for each test. The most interesting time period occurs shortly after 10000 milliseconds; the most stable commonly shared period of time demonstrates that activity is higher within both dynamic tests, with the 3rd test staying stable on high activity for the vast majority of the tests taken.

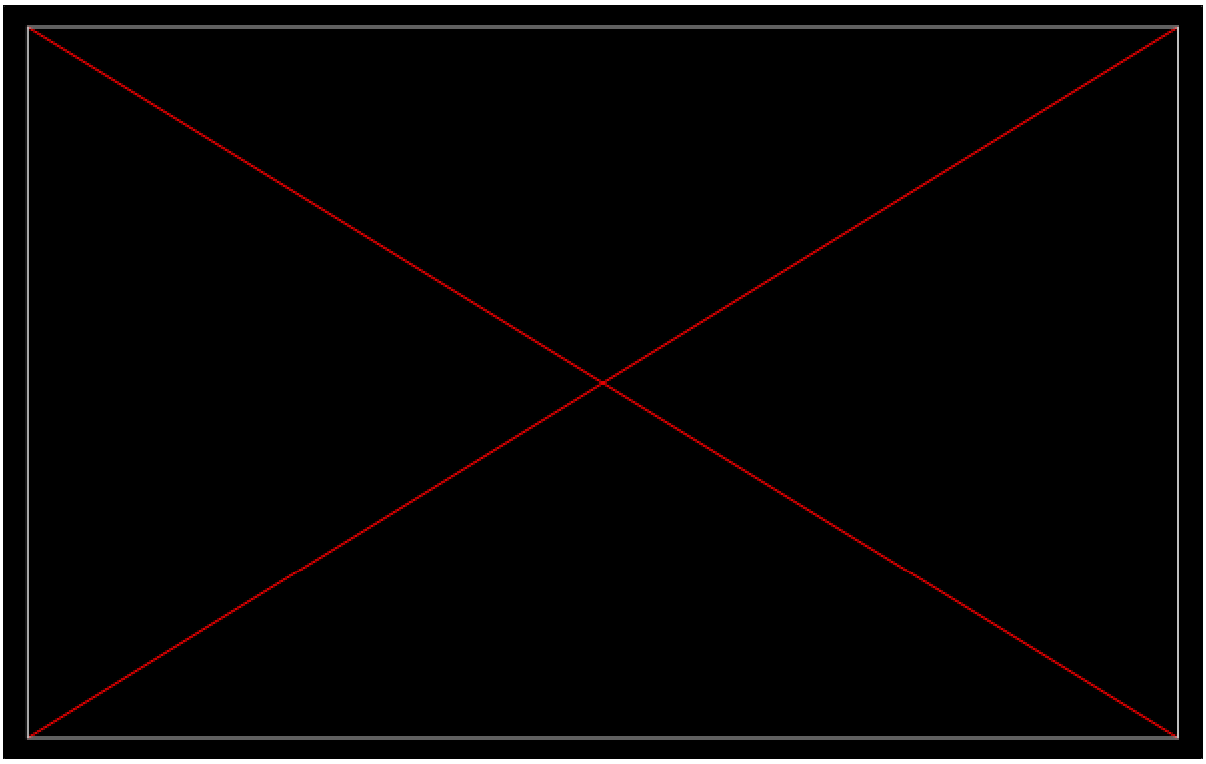
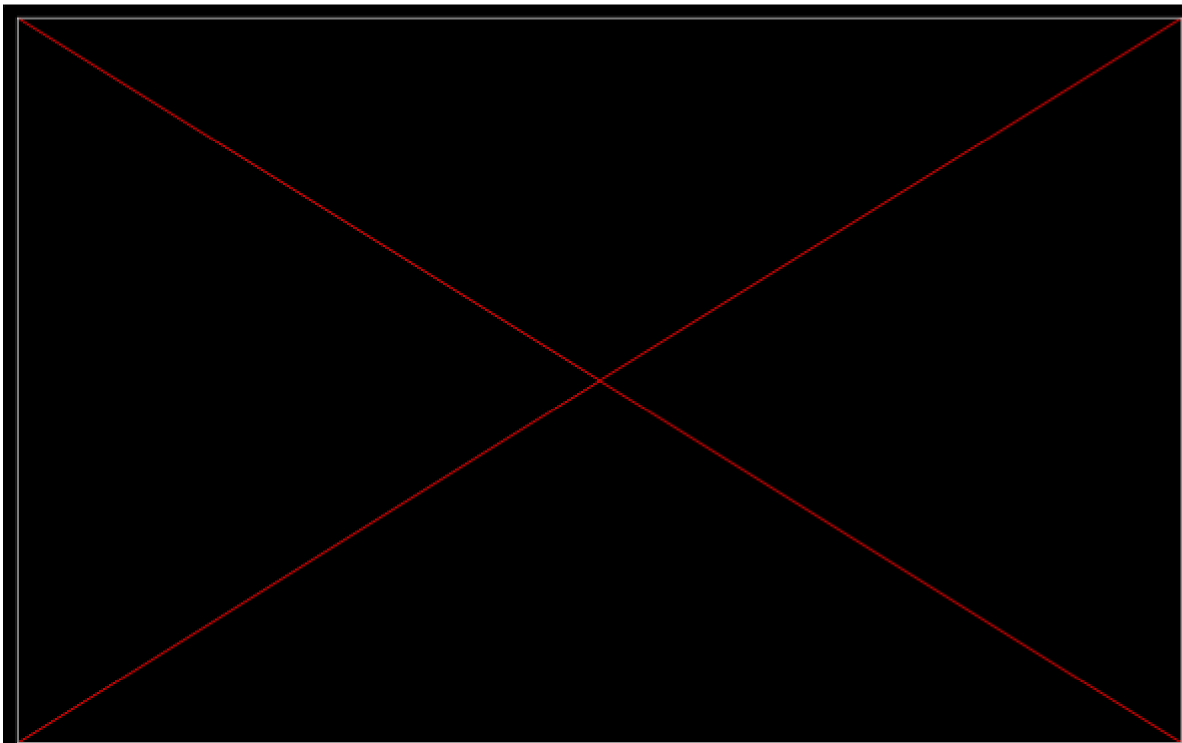


Fig vii

The comparison of each heat map for the driving game also shows differences between each test (Fig viii, ix, x). With each successive test more and more of the track is used; this is an indication of a greater utilization of the available content. We see better use of the content that is within the game; there was no need to add more content to these system, the game simply provides increased avenues to explore all of this content.

Heatmap of activity for the static test

Fig viii



Heatmap of activity for the 3-round dynamic test

Fig ix

Heatmap of activity for the 6-round dynamic test

Fig x

The analysis of variance for each of the tests reveal the overall success of the research. When comparing the data between each test, for each game, averaged across all participants, we can see a positive trend within the evaluations. The analysis for the Driver game has been included in the appendix, the Invader game evaluations have been included here (Fig xi, xii, xiii).

The first dynamic test (Fig xii) demonstrates that there is a positive impact for the average player; an increase of 0.8 of the 5 point scale provided for evaluation. This means that 4 of every 5 players found the system at least one point more favorable than the previous round, after only 2 generations. The 6-round dynamic test (Fig xiii) indicates that the effectiveness of the system decreases rapidly after the 3rd generation, dipping into the negative by the 5th. This could indicate issues with the balance of the system itself, or could indicate that as with other skill evaluation systems, it needs to adjust less aggressively for each subsequent generation.

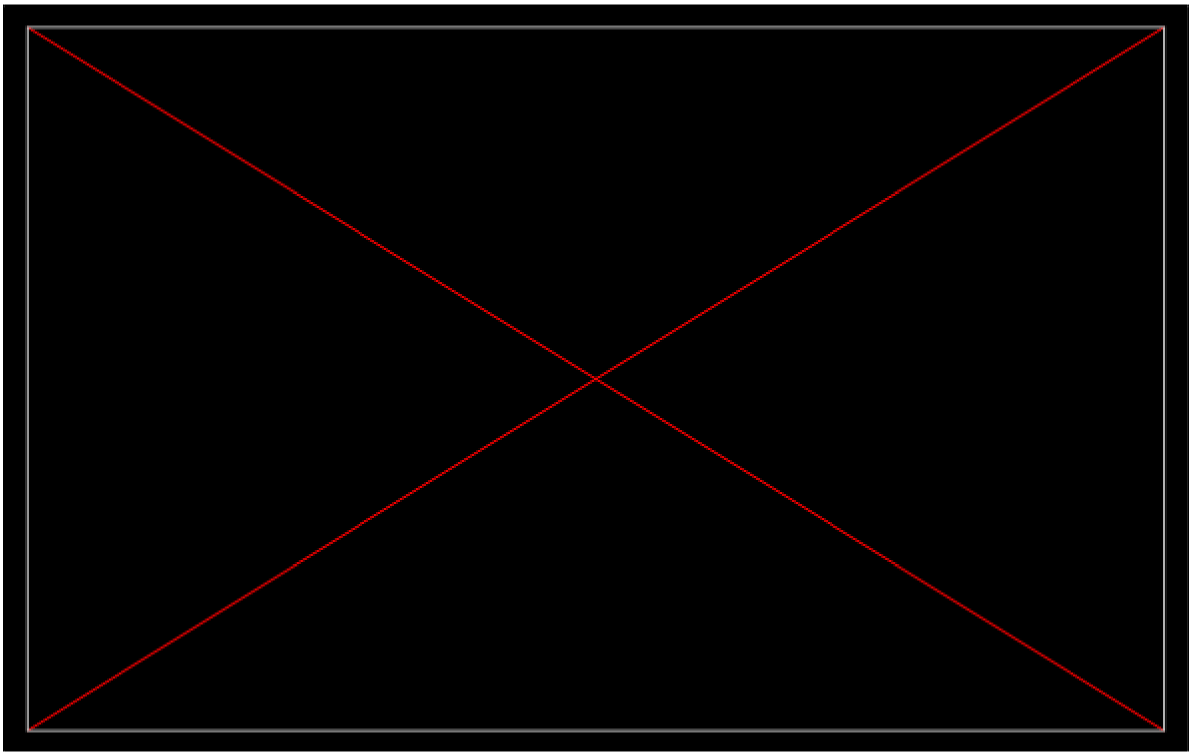


Fig xi

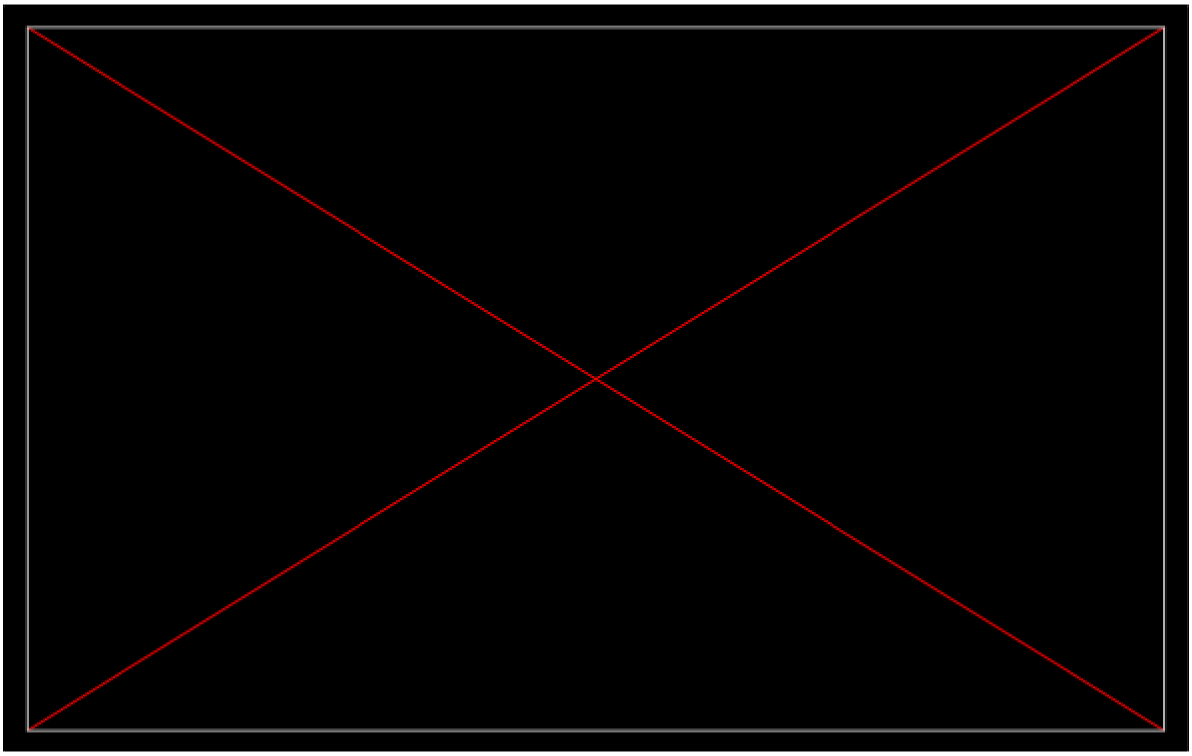


Fig xii

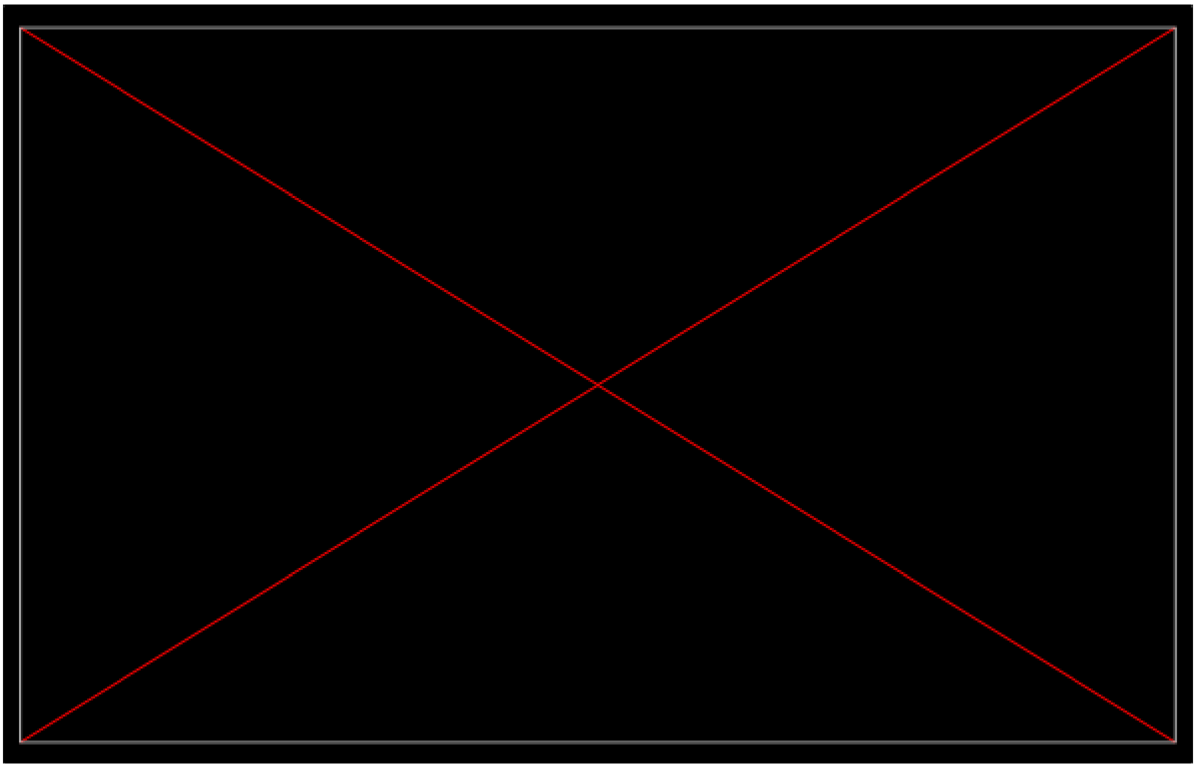


Fig xiii

Conclusions

The results of the testing indicate a positive change in experience for the average player. The emergent game that is built from the small independent modifications to the system appear to create a more interesting experience for the player. This demonstrates value in this area of game design. Being able to craft the experiences of the player; that is, to keep control of the game within the hands of the designer; while still allowing the player to have as much freedom as possible, is within the realm of possibility with this method of game design. Each change to the emergent system was specifically crafted and controlled by the design placed upon the system, however the outcome of the cumulative changes applied to this machine is rarely able to be predicted beforehand for each individual player. Each time the game is played it will appear to learn and build itself to make it more fun and interesting for the player, thereby increasing skill in a range of areas, rather than one simple difficulty slider.

This new area of game design still holds much to be explored, however it demonstrates how a series of smaller, simple systems can appear to have an emergent collective intelligence, when it is really just a series of feedback mechanisms built into the core mechanics of the game. This allows each design to still fall prey to a bad decision by the designer and does not prevent failure. As with any craft, the design of games is still a complex mechanism to master.

The simplicity of these test games demonstrate only the value of the system in a generic fashion, and would ideally be more suited to a game that has more depth. The simple nature of the games involved in the testing naturally limits the potential for exploration of the emergent mechanics - in a more extensive game, founded upon systems of greater complexity, we would see an increased number of outcomes, as the potential for variance is increased. This area of research could even reach into the realm of procedurally created content based upon the experiences of, and decisions made by, the player. This of course is not fully realised in this particular study, but should represent the ideal future of this work.

References

Adams, E., & Rollings, A. (2007). Fundamentals of game design. Upper Saddle River, N.J: Prentice Hall.

Ashmore, C & Nitsche, M. (2007). The quest in a generated world. In Situated Play, proceedings of DiGRA 2007 conference, (2007, The University of Tokyo, Hongo, Bunkyo-ku, Tokyo, Japan)

Atari. (1975). Indy. (Video Game)

Bethesda. (2006). The Elder Scrolls IV: Oblivion. (Video Game)

Bjork, S., & Holopainen, J. (2005). Patterns in game design. Hingham, Mass.: Charles River Media.

Calleja, G. (2009). Experiential narrative in game environments. In Breaking New Ground: Innovations in Games, Play, Practice and Theory, proceedings of DiGRA 2009 conference, (2009, Brunel University, West London, United Kingdom)

Castillo, T., & Novak, J. (2008). Game development essentials: game level design. New York: Delmar.

Chen, J. (2006). Flow in games (MFA Thesis). Retrieved from http://www.jenovachen.com/flowingames/Flow_in_games_final.pdf

Crawford, C. (2003). Chris Crawford on game design. Indianapolis, Ind.: New Riders.

Csíkszentmihályi’s, M. (1991). Flow: the psychology of optimal experience. New York, NY: Harper Collins Publishers.

Dahlskog, S. & Kamstrup, A. & Aarseth, E. (2009). Mapping the game landscape: locating genres using functional classification. In Breaking New Ground: Innovations in Games, Play, Practice and Theory, proceedings of DiGRA 2009 conference, (2009, Brunel University, West London, United Kingdom)

Dansey, N. & Stevens, Dr. B. & Eglin, Dr. R. (2009). Contextually-Ambiguous pervasive games: an exploratory study. In Breaking New Ground: Innovations in Games, Play, Practice and Theory, proceedings of DiGRA 2009 conference, (2009, Brunel University, West London, United Kingdom)

DMA Design. (1997). Grand Theft Auto. (Video Game)

Drachen, A. & Jhala, A. & Yanakakis, G. & Hitchens, M. (2009). Towards data-driven drama management: issues in data collection and annotation. In Breaking New Ground: Innovations in Games, Play, Practice and Theory, proceedings of DiGRA 2009 conference, (2009, Brunel University, West London, United Kingdom)

Eladhari, M. (2009). Emotional attachments for story construction in virtual game worlds. In Breaking New Ground: Innovations in Games, Play, Practice and Theory, proceedings of DiGRA 2009 conference, (2009, Brunel University, West London, United Kingdom)

Falstein, N. (2006). Emergent complexity. Game Developer, 13(4), 54-54. Retrieved from http://search.proquest.com/docview/219047077

Flanagan, M. & Fernandez-Vara, C. & Pearce, C. (2009). "Some assembly required": starting and growing a game lab. In Breaking New Ground: Innovations in Games, Play, Practice and Theory, proceedings of DiGRA 2009 conference, (2009, Brunel University, West London, United Kingdom)

Fox, B. (2005). Game interface design. Boston, MA: Course Technology PTR.

Hektner, J. M., Schmidt, J. A., & Csikszentmihalyi, M. (2007). Experience sampling method: measuring the quality of everyday life. Thousand Oaks, Calif.: Sage Publications.

Juul, J. (2010). A casual revolution: reinventing video games and their players. Cambridge, MA: MIT Press.

Koster, R. (2005). A theory of fun for game design. Scottsdale, AZ: Paraglyph Press.

LeBlanc, M. (2000). Formal Design Tools: Emergent Complexity, Emergent Narrative [PowerPoint Slides]. Retrieved from http://algorithmancy.8kindsoffun.com/gdc2000.zip

Lemay, Dr. P. (2007). Developing a pattern language for flow experiences in video games. In Situated Play, proceedings of DiGRA 2007 conference, (2007, The University of Tokyo, Hongo, Bunkyo-ku, Tokyo, Japan)

Liesegang, S. (2009, November 9). Disruptive construction of game worlds. *Scholars' Lab*. Lecture conducted from University of Virginia, Charlottesville.

Llopis, N. (2009). Procedural content creation. Game Developer, 16(7), 41-n/a. Retrieved from http://search.proquest.com/docview/219090465

Mateas, M. & Wardrip-Fruin, N. (2009). Defining operational logics. In Breaking New Ground: Innovations in Games, Play, Practice and Theory, proceedings of

DiGRA 2009 conference, (2009, Brunel University, West London, United Kingdom)

Nintendo. (1992). Super Mario Kart. (Video Game)

Reyno, E. & Cubel, J. (2009). A platform-independent model for videogame gameplay specification. In Breaking New Ground: Innovations in Games, Play, Practice and Theory, proceedings of DiGRA 2009 conference, (2009, Brunel University, West London, United Kingdom)

Ribbens, W. & Poels, Y. (2009). Researching player experiences through the use of different qualitative methods. In Breaking New Ground: Innovations in Games, Play, Practice and Theory, proceedings of DiGRA 2009 conference, (2009, Brunel University, West London, United Kingdom)

Roberts, D. & Riedl, M. & Isbell, C. (2009). Beyond adversarial: the case for game AI as storytelling. In Breaking New Ground: Innovations in Games, Play, Practice and Theory, proceedings of DiGRA 2009 conference, (2009, Brunel University, West London, United Kingdom)

Rogers, S. (2010). Level up!: the guide to great video game design. Chichester: Wiley.

Salen, K., & Zimmerman, E. (2006). The game design reader: a rules of play anthology. Cambridge, Mass.: MIT Press.

Sierra Entertainment. (1999). Homeworld. (Video Game)

Simons, I. (2007). Inside game design. London: Laurence King Pub.

Svahn, M. (2009). Processing play; Perceptions of persuasion. In Breaking New Ground: Innovations in Games, Play, Practice and Theory, proceedings of DiGRA 2009 conference, (2009, Brunel University, West London, United Kingdom)

Taito. (1978). Space Invaders. (Video Game)

Thorn, A. (2011). Game engine design and implementation. Sudbury, Mass.: Jones & Bartlett Learning.

Trefry, G. (2010). Casual game design designing play for the gamer in all of us. Burlington, MA: Morgan Kaufmann.

Zagal, J. & Mateas, M. (2007). Temporal frames: a unifying framework for the analysis of game temporality. In Situated Play, proceedings of DiGRA 2007 conference, (2007, The University of Tokyo, Hongo, Bunkyo-ku, Tokyo, Japan)

Appendix

Please visit http://research.frontier-studios.net to access the the testing station, the source code for the games and for further data analysis.