ENDURE ADVENTURES

Measuring player experience enhancement in an interactive procedural-adaptive storyworld

Aalborg University CPH
Mikki Vitus Rasmus Jordan

Aalborg University Copenhagen

Semester:

10^{th} semester, Medialogy (MSc)

Title:

Measuring player experience enhancement in an interactive procedural-adaptive storyworld

Project period: July 8, 2021 - November 7, 2021

Semester theme: adaptivity integration in video games and simulations

Supervisor: Henrik Schønau Fog

Group members:

Mikki Vitus Rasmus Jordan (mjorda15@student.aau.dk, 20156904)

Deadline: 7^{th} of November 2021

Pages: 86

Abstract

Procedural content generation can potentially extend the interactive adaptive storyworld and its affiliated user experiences without succumbing to loss in narrative control. It seems that the research on combining these to achieve procedural-adaptative systems could be researched more in depth. This project aims to bridge procedural content generation and interactive adaptive real time storytelling by including the user into a game state evaluation of the storyworld and thus explore how user's behaviour and interactions can be used for adapting the storyworld and thus potentially enhance the experience for the user. The game state interprets a meaning from an interaction, and from this, it makes an adaptive steering encoding and feed the encoded data back into the game state so that the storyworld changes its narrative. Open worlds aim to provide both player freedom and author control at the same time, which creates a narrative paradox; and the possibility to change the storyworld may further complicate this challenge. However, a space-time drama manager can help the narrative events to be temporal and spatial independent, allowing the narrative to appear in front of the user at the time needed. A demo has been developed as a proof of concept and evaluated with a between-subjects design (n = 3). In the control condition the participant experienced a game with a linear narrative without a procedural-adpative storyworld. For the experimental condition an interactive procedural-adaptive storyworld was presented. A mean comparison indicates that the proposed procedural-adaptive framework enhances the player experience.

Keywords

Procedural content generation · Interactive adaptive real time storytelling · Digital game experience · Storyworld

This page is intentionally left blank

Table of content

List of Figures

Definitions

Term	Description	Page
Personalised game	A game that strives to automatically adapt itself from a current state to an enhanced state based on player's intention, play style and behaviour.	6
Literal experience	The experience generated from the game itself and not the storyworld.	7
Holistic experience	The experience that is generated from the storyworld.	7
Perceived agency	A subjective measure to how actions are recognised and matters in the storyworld.	9
Theoretical agency	An objective measure to which degree actions can change the storyworld.	9
Adaptive representation problem	When an adaptation representation in a storyworld cannot be fully exposed or hidden as it would be ignored.	13
Narrative paradox	The conflict between authored narratives and the freedom that is offered by the storyworld.	16
Emergent storytelling	The player personally constructs a narrative, through improvisational play and interactions in the storyworld.	16
Combinatorial explosion	The difficulty in maintaining a meaningful story depth when agency can exponentially explode the story in an interactive storyworld.	16

Term	Description	Page
Experience adaptivity	Adaptivity in a storyworld is a temporal interpolation between its experiences and the interactions performed by the player.	18
Real time interactive adaptive story telling	The storyworld allows a prior state to become convoluted into a new enhanced state, based on either player style, behaviour and intention or an intelligent NPC	19
Storytelling adaptivity	When the narrative discourse changes such that it is presented differently or through different means to preserve the narrative flow.	20
Procedural content generation	An algorithm that is automatic, self-organisational, based on rule-sets and constraints to generate content.	21
Procedural-adaptive NPC	A character that perceives the storyworld and the player, from which it makes its own decisions.	59

Preface

There are ten appendices included for this project. They are ordered in chronological order in accordance to the sections. The page numbering indicates which pages the appendices regard. All appendices will be located digitally, where they will be referred to according to their appendix article as given beneath.

Appendix article	**Page**
Game and video	4 - 14
Affiliated work	19 - 20
Questionnaires for case study	38 - 41
Procedure and questions low-fi mock-up test	44 - 48
Optimisation	48 - 68
Participatory accessibility	48 - 68
Optimisation	48 - 68
Story	48 - 68

Delimitations

A few deliminations have been made throughout this project because they are beyond the scope for this project and the current COVID-19 situation. There will be no attempt to give a low-level definition on artificial intelligence, however since artificial intelligence has relevance to procedural content generation a high-level description will be given for the areas necessary. Procedural content generation will be limited to a storyworld; by this statement it is acknowledged that procedural content generation extends itself outside what will be covered throughout this project. The implementation has been limited to what can be considered as the most common game aspects, that being the narrative, game and level design. Areas not covered in this project should be expected to be eligible to the proposed framework. Due to the COVID-19 situation throughout the period of this project, the implementation has excluded the development and player experience testing for virtual reality. However, it is acknowledged that this technology may enhance the player experience due to better presence and immersion. This also applies for the overall lack of appropriate measures, access to the Aalborg University Copenhagen labs, proper equipment for the development of the project. For the same reason, the project software criterion has been limited to work on less powerful hardware, and is specifically designed for low specification systems. This also implies that the development have had to consider participatory accessibility options to access a broader audience. This options regards, language compatibility for both English and Danish language and graphical settings to degrade the visuals and complexity in the environment. The author has used the baseline system, HD 620 graphics, i7-7500u and 16GB of ram.

Introduction

Most often in the context of procedural content generation (PCG) the question has become how to achieve a potential ad infinitum, without the need for author and narrative control (Blatz and Korn 2017). The shortcoming with ad infinitum for PCG is that it rarely results in great player experiences and narratives. Research has implied that player styles, behaviours and intentions can be fed to a PCG to enhance the player experience in games by using a generative and player dependent storyworld. However, a PCG cannot continuously change and align the storyworld experience with the player. A solution to this problem has been to introduce interactive adaptive real time storyworlds. Although, this introduction has lead to the narrative paradox and the combinatorial explosion. A space-time drama manager (Schoenau-Fog 2015a), has been proposed to help preventing this narrative paradox by respecting the player's spatio-temporal orientation instead of being fixated. It allows narrative events to be introduced both at a certain time and at a specific place in the storyworld. The narrative in a storyworld has thus far been adaptive based on player interactions.

This implies that the player role shifts, in that experiences in the storyworld can become dependent on the player styles, interactions and behaviours. In this way the experience are allowed to become highly personalised and centralised. However, this introduces the narrative explosion (Louchart and Aylett 2003) and problems for game flow (Csikszentmihalyi and Larson 2014) and agency (Mateas and Stern 2005). Accounting for flow and agency constitute to the exploding narrative (Mateas and Stern 2005) where the narrative becomes either boring because the player does not follow the narration or too overwhelming because the narrative is experienced too quickly. Game flow (Csikszentmihalyi and Larson 2014) in an adaptive storyworld becomes a concern in open world games and PCG implementations because the player has freedom (Louchart and Aylett 2003). However a common practice, have been to follow linear narratives, with a fixed spatio-temporal relation as achieved with a space-time drama manager. Some games, Shadow Of Mordor (Monolith 2021), Left for Dead (Valve 2021) and Façade (Mateas and Stern 2005), rely on both a PCG and adaptive framework and have shown potential solutions to the combinatorial explosion, narrative paradox (Stern 2008).

Both the player experience and emergent storytelling may affect the storyworld and its components such as, interactive materials, vegetation, growth, animations and environments, models, elements, characters and music. And the storyworld may afford a player experience through the storyworld and its components. These storyworld components are important because narratives and expressions can be extrapolated from them and changed depending on the player style and behaviour. For instance, the player may attack a character and this character may appeal to this action, by attacking the player or fleeing.

A procedural-adaptive has shown to be a promising approach to account for a storyworld that is dependent on player styles and behaviours in that it allows the storyworld to be generative and adaptive at the same time and thus enhance the player experience. It has shown to be a potential solution to the narrative paradox and the combinatorial explosion as narrative control can be achieved with semantics.

The aim for this report will be to bridge the gap between PCG and an interactive adaptive storyworld, creating a procedural-adaptive framework where the player steers the experience. This will be achieved by feeding player interactions, behaviours and play styles into the PCG algorithm that generates the storyworld. This project thus aims to contribute to the area of interactive procedural-adaptive storyworlds by relying solely on the player. The following initial research question will therefore be considered as the starting point to achieve this aim.

0.1 Initial research question

How can an interactive adaptive real time story telling be generated such that it accommodates for the player without missing the narrative.

Supplementary question #1 How can the storyworld continuously align itself with the player experience?
Supplementary question #2 How can the player direct adaptation in a storyworld so it emphasises on the player experience?

0.2 Author's motivation

The conducted research is motivated by the author's personal interests and ideologies. It has been an intriguing thought to play an engaging game that reflect the one's wants and desires; however most games does not respect the player's needs, and lack agency refraining from becoming truly interactive to accommodate the player. After playing the Witcher series, it has become disappointing to acknowledge that the game is not adaptive and disregards the player from changing the narrative. The game instead emphasises on the experiences that the player has felt and the journey to reach the end, as similarly to (Valve 2021). The Witcher prevents the player from alternating the narrative because it rely on pre-written branching narratives. Henceforth, the realisation that few games are not adaptive and linear have risen the curiosity that the Witcher originally evoked, to play and make games that are adaptive and which narrative changes accordingly to my desires and interactions as a player.

0.3 Research motivation

This research field is interesting because it extends an interactive adaptive storyworld without the player succumbing to a loss in narrative control. A potential solution to achieve narrative control is by allowing the player to have an interactive role in the game narrative. Examples of this are also apparent in (Mateas and Stern 2005; Peirce et al. 2008). A game system that can interpret player interactions are important in open world games and non-linear narratives because expression, virtual manifestation and play styles are important. The player may want to play these games in a particular way, which is not supported by the game system. In game structures like this, the combinatorial paradox and narrative paradox, both directly impact the player's experiences, in that narrative flow cannot be contained due to virtual freedom. For this reason, the two paradoxes should be omitted in open world non-linear narratives and the player's play style should be considered to enhance the experience.

1 Analysis

The following chapter will explore the current research for an interactive adaptive real time storytelling narrative in open world storyworlds. There are three components to potentially achieve this, player experience, interactive adaptation in storyworlds and procedural content generation (PCG). A player experience-driven approach is presented and is compatible with an interactive adaptive storytelling paradigm. An interactive and generative storyworld adaptation is needed to comply with the player experience. Hence, this imply the usefulness of a PCG as an algorithm that allow an interactive and generative adaptive steering towards the player experience to occur. This potential solution will be a procedural-adaptive framework that is based on the player's experiences. The theory needed to develop this framework will be thoroughly investigated, and potential problems and concerns will be addresses. In exploring this field, the chapter will introduce to an adaptive real time storytelling, as an approach to enhance the player experience, the player is included into the adaptive process such that the player can change the narrative and storyworld directly, and thus enhance the experience. It furthermore introduces, amongst others, to the drama manager as an approach to organise the narrative, whilst allowing freedom in open worlds and thus potentially prevent the narrative paradox. An investigation will be put into solving this narrative paradox but also how a game can account for player's agency, play style and behaviour in a storyworld such that it can dictate the pace of the narrative, and therefore be a potential solution to the Combinatorial explosion.

This chapter will present and investigate how control, organisation and expression can be derived from a PCG and used in conjunction with an adaptive real time storytelling as a method to achieve an enhanced player experience. The play style, behaviour and intention guides the procedural-adaptive framework so the narrative and storyworld can align with the player experience.

*

1.1 The player experience

An experience and how to enhance this during game play is a central concern for this project. An experience is rarely determined monotonously, because it can be associated with multiple entities. An experience may be the result of different emotional characteristics that the player takes during game play. To complicate this, the experience may change as different situations arise; such as when the game system contradicts the play style and intention. The aim is to provide an interactive real time adaptive storytelling that enhances the player experience. It therefore becomes necessary to understand what an experience is and which parameters that will allow an adaptive storyworld and the player to become dependent. This section will therefore address how an experience-driven can be injected into an interactive adaptive storyworld, such that the player is an interactive agent that steers the storyworld and their own experiences. The following sections will present methods to control and predict an experience, and which problems arises in this field of research.

A personalisation model has been proposed as a solution to capture player experiences from play styles, behaviours and intentions. A low-level description for what constitute to an experience will be given such that the personalisation model becomes more definitive and personalised; to achieve this, how a potential prediction should be generated and represented in the story world such that it aligns with the player will be considered. The personalisation model uses player modelling and adaptation for its experience predictions. For this reason a data-driven and model-driven approach have been investigated to feed the personalisation model.

1.1.1 Personalisation modelling

Personalisation dates back to trends in marketing, homogeneous segmentation to capture diversity and variation in consumers (Zhu and Ontañón 2020). Segmentation and personalisation (SAP) are vastly researched for automatic processes in different media and regard experiences, preferences, needs, behaviour in games and adds, recommendation systems and inspiration based on behaviour patterns on the internet (Churchill 2013; Kramer et al. 2000; Ontañón and Zhu 2021). A personalisation system essentially tries to build a model that characterises a player and then personalise an experience on this basis (Ontañón and Zhu 2021). Artificial intelligence can be used for this categorisation and help to capture preferences, needs, goals and behaviours to improve user enjoyment.

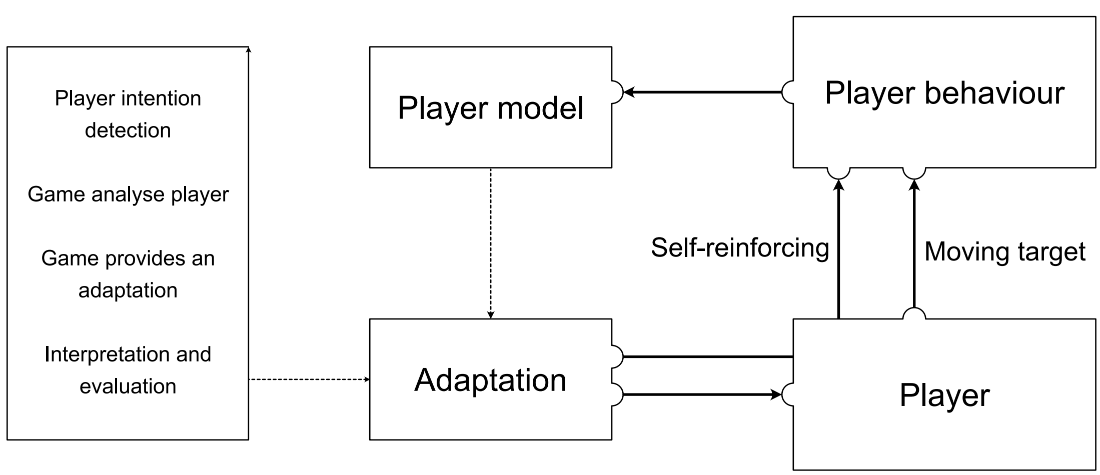


Figure 1 shows the dependencies of personalisation modelling and experience prediction. Inspired by (Zhu and Ontañón 2020; Ontañón and Zhu 2021)

SAP is pragmatised in human computer interaction (HCI) and cognitive fields of interactive structures as player centrism and affiliated experiences (Schell 2008). A SAP model becomes important for game design in that it aims to optimise an experience by capturing player traits such as characteristics, behaviour, player types, preferences, intentions. The SAP approach aims to model an experience based on the player as seen with marketing strategies. But sometimes, the SAP model, may confuse player intentions and provide a bad experience (Zhu and Ontañón 2020). This means that an experience prediction does not align with the player's experiences, creating a potential personalisation paradox (Ontañón and Zhu 2021). This paradox happens when there is a conflict between the predicting player personalisation and adaptation. The paradox is most prominent when combining personalisation with behavioural change. There are two elements to this paradox, self-reinforcing loops, and moving target. Self-reinforcing loops happens when a personalised system forces a player into a specific model category, regardless of its quality. Situations where this may occur is when there is no input from the player as a consequence of confusion. For instance, Bartle (Bartle 1996) categorises different player types in terms of Killers, Achievers, Explorers and Socialisers, these are also shown in. As shown in figure 2, it is an approach to categorise different behaviours, player types, preferences and intentions. The blue outline indicates that a player has a range of play styles. Although, the personalisation model may categorise the player to have a Killer play style despite being an Explorer. The model may also confuse a play style to be a Killer even after the play style has changed into an Explorer. As the goal for the SAP model is to steer preferences, behaviours and play styles into a particular direction this may become contradicting.

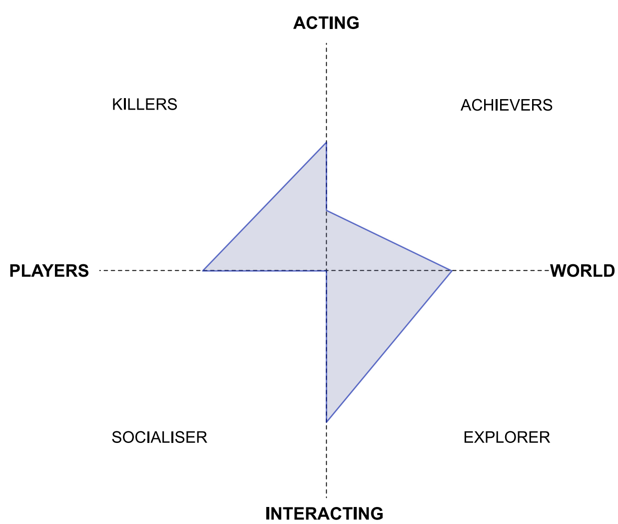


Figure 2 shows Bartles' different player types and associated preferences. The blue outline emphasises that a player has a range of play styles.

A personalised game is defined as a game that strives to automatically adapt itself from a current state to a new one based on the experience of the player's intention, style and agency (Hudlicka 2009). Bartles player types (Bartle 1996) is one possible solution to steer this automatic adaptation, as player types can be categorised and described from player experiences during game play, as also demonstrated in figure 2.

It is important to note that a player experience lacks a clear definition in literature; the reason for this is that an experience can be associated with multiple entities. A Presence-Involvement Flow Framework (Takatalo et al. 2010) that is grounded in quantitative data has been used to achieve perception and attention for the storyworld (Tamborini and Skalski 2006). In similar terms, (Calleja 2011) uses a affective motivational involvement continuum to increase engagement. Whereas the authors (Csikszentmihalyi and Larson 2014) use affective states to achieve game flow during game play. An experience has also been correlated with control, freedom, curiosity, interactivity, game enjoyment and fun (Hartmann and Klimmt 2006; Lucas and Sherry 2004; Takatalo et al. 2010).

These different descriptions for player experiences in games also indicates that a SAP becomes a problem. The following sub section will therefore address how a player experience can solve this problem.

1.1.2 Player experience modelling

In games, a player experience could be defined with respect to its perspective, as indicated in the exploratory work by (Oswald et al. 2014). Two different definitions that separate game experiences can be derived from this work. These are the literal and holistic experience views as described for definition 1.1 and 1.2. For instance, people keeps playing Tetris because the game itself is fun. Whereas for a holistic experience, what keeps people playing Bioschock, is not merely the game, but the narrative, mystique, and the experience of living in Rapture through Jack.

The advantage of a holistic experience is that all storyworld components can be quantified (Vorderer et al. 2003). This allow a game to enclose qualitative components of the storyworld during or prior to the game and determine a particular user experience, intention and style. A game component refers to '*all aspects of a game that affect gameplay but are not non-player character behaviour or the game engine itself*' Yannakakis and Togelius 2011, p. 148. This definition exclude intelligent NPCs. Excluding NPCs, hence partly the Socialiser (Bartle 1996) is somewhat puzzling because, this player preference empirically correlates to the experience (Tamborini and Skalski 2006; Strack and Deutsch 2004) and since the storyworld can become adaptive it can contribute to the player experience (Lopes and Bidarra 2011). It provokes emotions and changes to the storyworld, allowing the player to have social awareness (Cohen 2001) and games have shown player's capability of strong affective responses towards characters (Bordwell et al. 2017).

Definition 1.1 (Literal experience) *The experience generated from the game itself and not the storyworld.*

Definition 1.2 (Holistic experience) *The experience that is generated from the storyworld.*

The goal of a player experience model is to translate an intention, behaviour and play styles into a conceivable act by a game. In itself, a goal is abstract, because it is how a predicted intention is formed into a possible solution for the player experience. An apparent research problem in predicting the player intention is how strong a correlation there is between prediction and intention. Which actions should be taken in a quasi model and how prediction confusion should be interpreted (Zhu and Ontañón 2020). Identifying a player intention is a valid challenge for adaptive system because the player actions and intentions are unpredictable. In similar terms, preferences and intentions may contradict itself from previous states of the game and not only in the present.

The framework proposed by (Takatalo et al. 2010; Bernhaupt 2010) allows different features of an experience to be evaluated and determined in terms of their quality, intensity, meaning, value, and extensity. Different experiences are evoked by different impressions during game play and how the player is perceptually affected by these (De Kort et al. 2007; Sedikides 1992). It is a concern how to determine the quality of an experience and which experience features that are important. The quality of a player experience model is correlated to a potential great prediction, not a best case prediction. For instance, if the system acknowledges that the player has interest in the Killer type, the environment has to become smaller, and loot has to be prioritised because this experience is preferred (Zhu and Ontañón 2020). However, there are different levels to a player style, and therefore, a best case prediction becomes difficult (Yannakakis and Hallam 2009; Zhu and Ontañón 2020). Quantifying the expressive quality of a storyworld is a common topic for holistic experiences because it depicts how successful an experience prediction, PCG and adaptation is to the current game state. Since a PCG may theoretically produce any conceivable storyworld, it does not imply that the prediction fits the player preferences (Yannakakis and Togelius 2011). This suggest that such system needs a matrix to evaluate the storyworld and potentially have it representative to player's experience. The matrix used for the player experience is crucial for the adaptive development during game play and should support crosmodality to account for each component of the storyworld (Yannakakis and Togelius 2011). The storyworld can be interactive and passive to make an equilibrium in cases where player predictions are dominantly more diverse than a former state (Zhu and Ontañón 2020).

As illustrated by figure 3, players may have different experiences and succumb to different experiences despite narrative control. However, an adaptive storyworld experience allows these to be much more definitive. For instance Co-presence can be enhanced between other agents because the relationship and narrative can become dynamic. Player involvement like this is inherited by an interaction component to the storyworld as player interactions may impact the storyworld. A player that interacts with the storyworld can support the idea that two different players may initially play the same game, but have it unfold to become separable, because the experience may change. This may be a possible solution to enhance the experiences of the player, because the experience becomes individualised. In a participatory sense these may result in an overall average experience, and a potential evaluation of the user's experience would arguably be lower when the storyworld is controlled compared to an adaptive one as indicated in literature (Bosser et al. 2020; Schønau-Fog 2012; Hastings et al. 2009; Jennings-Teats et al. 2010; Shaker et al. 2010).

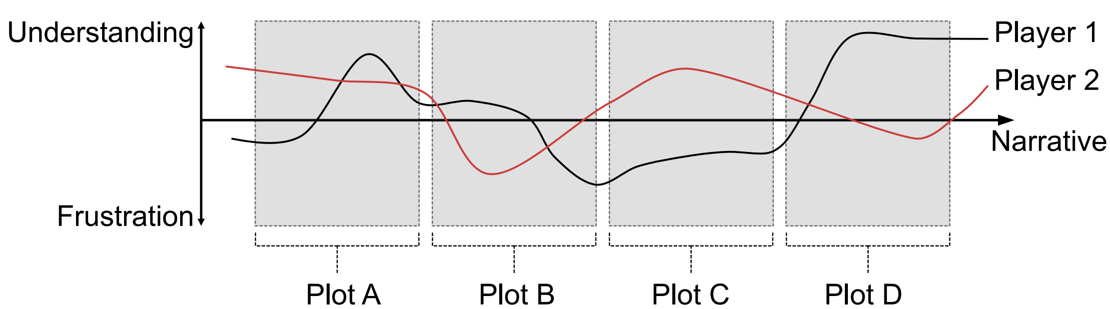


Figure 3 player's experiences fluctuates when dealing with tasks, thus two different players may have very different experiences even though they have been presented with the same story.

An experience can also be linked with a PCG to enhance the player experience (Yannakakis and Togelius 2011), and at the same time be based on the affective patterns and cognitive processes generated during game play. Using an experience to steer an adaptive storyworld and PCG may allow the two to become more meaningful to the player. One solution to achieve this is with a data- and model-driven approach and preference learning. The detection, modelling and synthesis of player experiences are not trivial since emotions are conceptual constructs and emotional states are entities with unclear boundaries (Yannakakis and Togelius 2011; Hartmann and Klimmt 2006; Lucas and Sherry 2004). A holistic experience is suggested to have a storyworld that is interactive and linked to a affection model so that it can be quantified. The storyworld should be responsive towards it inhabitants, although the author (Yannakakis and Togelius 2011) states 'fast' instead of responsive. However 'fast', is arguably misleading in that adaptation only needs to be as 'fast' as an alternation is needed in the storyworld. For instance, an adaptation can be withheld as long it is outside the observable space of the player.

It is difficult to confide intentions, motivations and behaviour in an experience-driven framework, where player interactions are meant to impact the experience provided by the storyworld. The reason for this is that intentions, motivations and behaviours are not necessarily correlated to their immediate interactions (K. Smith 2013; K. Smith 2017; Hastings et al. 2009). For instance, the two players' experience depicted in figure 3 can dynamically change by curiosity, uncertainty (Hamari and Tuunanen 2014; Bonser 2020) and dynamic player types. A possible solution to this problem could be when the player shows to have an Explorer player type. Then the storyworld could enhance experiences directed toward this player type by introducing, hidden passages, collectables and detailed storyworlds. This allow the player to become an integral part of the storyworld. In this manner, the storyworld afford the Achiever and Explorer player type as an attempt to enhance the player experience. In a rogue-like and role playing games (RPG), it is common that the player is given a limited range of play styles. A RPG may account for a play style by letting the player choose between either a magician class and barbarian class. However, being able to play a character that is a barbarian-magician is either not possible or the player has to prioritise one of the two. Allowing the player to play as a barbarian-magician, and at the same time have agency in a narrative and storyworld, would result in strong self identification (Curry and Arnaud 1974), thus enhancing the experience.

A main concern in modelling player experiences from intention is the model precision that determines the ideal player experience. This refers to the model's capabilities of individualism, recognition, data gathering and to distinguish between otherwise similar players. The precision of the prediction model depends on agency and the euclidean distance between actions quality and meaning to the player (Zhu and Ontañón 2020). Perceived agency, is a subjective measure to how actions are recognised and matters in the storyworld. Which thus implies the potential to enhance the player experience (Day and Zhu 2017; Cardona-Rivera et al. 2014). On the other hand, when a game for instance propose exactly one solution to solving a level for, say the Socialiser, but socialising with NPCs and other players will not result in a change in the storyworld then the euclidean distance increases and the model precision degrade. The lack of change from actions will cause agency to be theoretical and agency is questioned, degrading the overall experience (Day and Zhu 2017). Theoretical agency, is an objective measure to which degree actions can change the storyworld. Consider the euclidean distance to be the bias towards the choices the game presents and the player's agency, intentions, player styles and preferences (Zhu and Ontañón 2020). Data-driven approaches may not capture these features, partly because the features are flexible and players do not strive to achieve a goal most precisely, but a goal is achieved with features that causes a better representation of the player. For this reason, to determine which actions are meaningful and steering actions that are meaningless is an evident problem. It is important to note that theoretical agency does not yield in perceived agency as no player experience alignment is guaranteed. Emphasis has also been put on opponent modelling, to advance in intelligent NPCs (Magerko 2008; S. C. Bakkes et al. 2009). Opponent modelling can be based on the player and the storyworld through observations, clustering, feature data and scenario learning, so that NPCs can be incorporated in an interactive adaptive loop. In opponent modelling, a Finite State Automata has been used to recognise the player and storyworld (Lopes and Bidarra 2011). To close the gap between an experience prediction and player observable features can be applied (Zhu and Ontañón 2020). For instance, figure 3 showcases that two players may have fluctuating experiences in the same linear game, an prediction thus becomes difficult.

Advances to maintain proficient experience prediction has been proposed in research, classification, Bayesian models, Markov chains long-short term memory artificial networks (Yannakakis and Togelius 2011; Yannakakis and Hallam 2009; Zheng et al. 2020; Zhu and Ontañón 2020), player types, preferences, experiences and behaviour (Bartle 1996), feature filters (Tutenel, Van Der Linden, et al. 2011). Advances that specifically includes the player is oriented around PCG, automatic, adaptive mechanisms and play style specific storyworlds (Nitsche et al. 2006; Shaker et al. 2010; Yannakakis and Hallam 2009), evolutionary algorithms and artificial intelligence (Yannakakis and Togelius 2011; Hastings et al. 2009), that all situates or alternates the storyworld. A framework for optimising the quality for an experience and PCG driven experience are not explicitly and uniformly defined (Zhu and Ontañón 2020).

An experience is composed of different affections, player states and how the player perceives a game. However, the storyworld needs player data to interactively predict an experience that aligns with the player throughout the game. The following sub section will therefore, tend the focus towards how such data can be acquired.

1.1.3 Data-driven and model-driven approaches

There are two approaches from which an experience can be directed by the player, that is the bottom-up data-driven (subjective) and the top-down model-driven (objective) (L. F. Yu et al. 2011; Zhu and Ontañón 2020). The data driven approach is based on data that is accessible, either through studies, self-reports or the player during game play, whereas the model-driven is based on the assumption that the player behaviour can be predicted with the player's immediate or past actions. For the data-driven approach, the process is driven by some qualitative measures and theories that help map player experiences model and the quality of a psychological state (Yannakakis and Togelius 2011). A data-driven approach may collect data with psychological and cognitive measures with additional equipment, such as heart rate, cameras, physical interaction (K. Smith 2013; K. Smith 2017; Balakrishnan et al. 2013), electroencephalography, electrocardiography, galvanic skin response, motion caption (Yannakakis and Togelius 2011), pupillometry and gaze with the advance of virtual reality (Zheng et al. 2020; Palmero et al. 2021) to drive experiences. In (Yannakakis and Togelius 2011), an experience-driven PCG is based on the assumption that player emotions are associated by user self-reports, such as intrusive questionnaires and game context variables. A storyworld that adapts towards the player's experience may not necessarily arrange itself to these player descriptions. For instance, studies have shown that self-reports can guide A.I algorithms to predict player experiences accurately and provide personalised levels. An example of this is in Super Mario (Shaker et al. 2010; G. Smith and Whitehead 2010; Jennings-Teats et al. 2010).

One possible way to integrate a data-driven approach dynamically during game play, could be through Bartle's Socialiser. For instance, by following a similar intervention as seen in The Witcher Series, dialogue can function as an un-interrupted case where the player's desire is exactly to Socialise with a NPC. The player may through this interaction feed the game with data, through the NPC. A statement could be 'How was the travel?', 'You look battered would you like to buy some potions' The game would in these cases have to track the events leading up to these questions. Two [things will happen] NPCs will appear to be more intelligent, and individualised, actually integrating them into the storyworld. Agency and narrative involvement may be enhance because the NPC interaction towards the player intentions prior to the conversation (Magerko 2008).

Data-driven modelling is specific and accurate, but at the expense that games may not support these technologies directly, they may become impractical, such as pupillometry and gaze tracking being sensitive to light and distances, bodily placement and the methodologies are often not capable of real time. However, there are participatory uncertainty to a data-driven approach in that participants may be inclined to suffer from memory (Cowan 2014; Dehon 2012), memory contiguity (Cowan 2014; Lohnas and Kahana 2014), self-discrepency (Vorderer et al. 2003) and short term preferences may not match long term preferences (Zhu and Ontañón 2020). Whereas a model-driven approach alleviate analytical precision, the player experience tend to become better, however the decoction of specific experiences may in either case become difficult (Yannakakis and Togelius 2011). For an overview, the data-driven and model-driven approaches have been illustrated in figure 4. The black filled circles indicates a direct sample point, whereas the grey filled circles are predictions from a pool sampling points gathered with a data-driven approach.

A data-driven approach is pre-deterministic and thus in-capable of an interactive adaptive real time regime, a model-driven approach therefore seem to result in a better methodology. This is due to the idea that the player is unpredictable and experience predictions have to be interactive and provide continuous feedback to map experiences, play styles and intentions (Yannakakis and Togelius 2011). Since a storyworld and the player are interactive, it is difficult to identify the player during game play with a data-driven approach (Zhu and Ontañón 2020). A model-driven approach assumes that a construction between modalities of player input and an emotional state representation is plausible, making the approach speculative instead of specific (Yannakakis and Togelius 2011). In contrast a model-driven approach tend be well functioning in real time. In similar terms to a data-driven approach, both usually rely on classification algorithms, artificial intelligence predictions and action-to-experience mapping declarations.

Many approaches to experience driven approaches proposes the use of PCG, evolutionary algorithms or artificial intelligence, in that these can learn and automatically change the content in the storyworld or its components, such that these gradually align with the player's experience (Lopes and Bidarra 2011). A storyworld's components can be fed to the player vice versa. This means that an evaluation function is plausible, and grounds the idea that

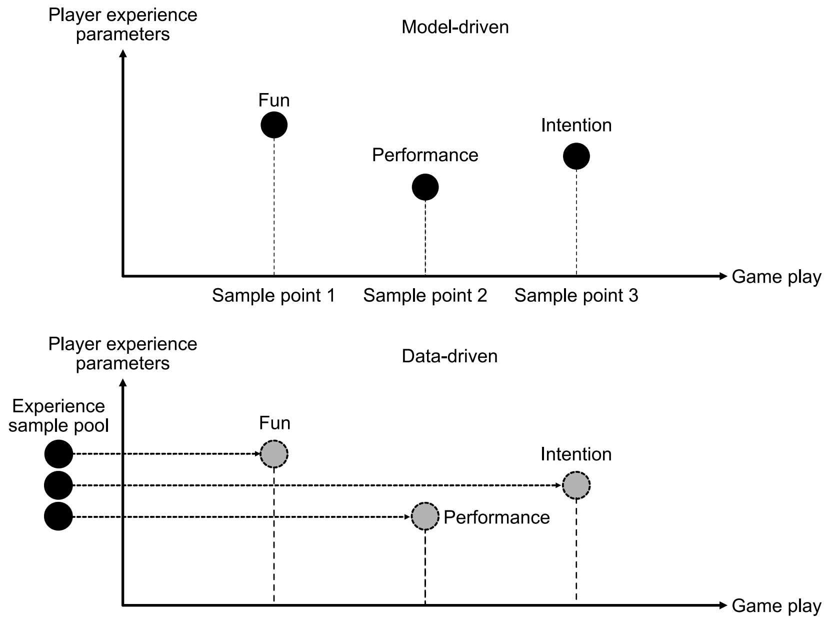


Figure 4 illustrates the difference for acquiring experiences prior to the game (data-driven) and acquiring experiences during the game (model-driven). Both approaches try to predict experiences during game play.

artificial intelligence and PCG can be a viable tool to predict player experiences (Yannakakis and Togelius 2011). The author (Yannakakis and Togelius 2011) have used an interactive evolutionary preference learning framework to predict player experiences in real time. Both an Artificial intelligence and PCG can predict an experience based on interactions during game play. Borderlands is a game that uses PCG to enhance the player experience for Achievers, Explorers and Killers, in that it creates weapons that changes how the player interacts with the storyworld and its inhabitants. The player is allowed to change the different components that makes these weapons to afford their play style (G. Smith, Gan, et al. 2011). Therefore, attention is directed to PCG and how it can potential provide a solution to an experience driven storyworld. Another approach is the experimental prototype called Charibitat, that uses the player to steer a PCG and adaptive storyworld in real-time (Nitsche et al. 2006). It presents four different approaches to this method, that is authored, random, adaptive to player interaction and procedural spaces. The project shows the potential in using an adaptive-procedural.

Both a data- and model-driven approach can acquire information about play styles, intentions, behaviours and interests. However, a 'data-to-perceiving' translation has to be considered for the storyworld to meaningfully align itself with the player. The following section will provide potential solutions, including address the issues that arises when the storyworld tries to present an adaptation prediction towards the player experience.

1.1.4 Experience representation and generation from data acquisition

Exploration has been used to guide a PCG for levels, where the player's active decisions determine which representation and generative direction the PCG is to take (G. Smith, Gan, et al. 2011). This model-driven approach rely player decisions which are given visually with six colour coded ladders. Each ladder represent different that is fed to a PCG. In platformers a model-driven approach could evaluate the player's dexterity in regards of exploration and engagement. Where the storyworld generation would afford dexterity through the representation of difficult jumps for instance (Shaker et al. 2010). In contrast, getting killed in Super Mario does not change the storyworld, but in (G. Smith, Gan, et al. 2011) the player is allowed to choose a different ladder. The data, through player choices can thus be used to represent and regenerate the storyworld. However, a confusion and player intention may difficult to account for. As demonstrated in the Walking Dead game, decisions do not always result in the intention of the player; the representation and generation from the player data therefore becomes inaccurate (G. Smith, Gan, et al. 2011). It is suggested that for the player to evaluate their experience they should understand how the storyworld adapt from their data. However, the proficiency for storyworlds that appear to have adaptive incognito is also suggested (Zhu and Ontañón 2020). The idea for this 'adaptive representation problem' is that when an adaptation representation in a storyworld cannot be fully exposed or hidden as it would be ignored.

The player's misconceptions of the storyworld's ability to adapt towards intentions can therefore sometimes impact the experience negatively. In contrast, a player that is fully aware of an adaptive storyworld choose to ignore it completely (Zhu and Ontañón 2020).

The representative quantity is also important. For instance, in Borderlands a player may use a wide range of the acquirable weapons, whereas others may use one weapon and modify this particular weapon. This task representation of the player comes down to how well a storyworld can identify a player through the data it has acquired as comparable to the SAP model. For a data-driven approach a concern is the lack of a priori. For instance, how would a storyworld accurately evaluate the quality of an experience representation before it has been generated. A player model may be a solution to this problem (Yannakakis and Togelius 2011). Another possible solution could be to evaluate the storyworld or its parts with reinforcement learning For a model-driven approach a concern may be the immediate intention of the player.

A concern with an experience driven game is how to align predictions with the player (content representation) and which features are most eligible to generate (content generation) (Hartmann and Klimmt 2006; Lucas and Sherry 2004; Yannakakis and Togelius 2011; G. Smith and Whitehead 2010). For example, the storyworld that dictates the experience may be based on emotions, performance and such as fun, challenge, frustration, precision, frequency, score et cetera. A system that can distinguish between understanding and frustration in two separate players as depicted in figure 3, should alternate features, when found, so the representation align with the player experience, preventing frustration. Equally important, is the feature evaluation model that makes up this negative experience. An experiences may be different prior to a climax as it would be directly hereafter, or indifferent as both may be equally satisfying. Thus, which 'experience sample' that should be considered may be crucial, for later adaptive predictions. Which sample and its best case steering direction seems in literature to be unclear and a study that concludes on this problem has not been found. An experience sample is difficult to confide and isolate because actions, intentions and motivations are not necessarily correlated (K. Smith 2013; K. Smith 2017) and may change accordingly to the game stimulus (Hamari and Tuunanen 2014; Bonser 2020). When the SAP model fails to identify a player type, Additionally, a play style may change monotonously or across several styles during game play (Hartmann and Klimmt 2006; Lucas and Sherry 2004). Which experience that results in specific emotions are difficult to determine as well. Likewise difficult to determine is the experience effect and no academic conformity has been established to account for this (Yannakakis and Togelius 2011). An attempt to control player diversity dictates why it is common for games to afford different play styles (Lopes and Bidarra 2011). And this diversity will likely change during game play (Hastings et al. 2009). However, capturing player diversity, and making a storyworld that emphasises the player, it becomes possible to construct an adaptive real time storytelling that may accurately depict the player (Magerko 2008).

1.2 Interactive adaptive real time storytelling

From the previous section, it is addressed how experiences and player modelling can yield in an quantification. A quantification makes it possible to feed player data to an interactive adaptive real time storyworld; and in this way steer the storyworld. However, this functionality implies that the story world can become adaptive. An adaptive real time storytelling is one potential solution to enhance the player experience in that it allows the player to become an interactive agent in the storyworld. These interactions can be injected to the SAP model from where player types, behaviours and intention can constitute to the adaptive alignment in the storyworld. In this way, the storyworld can be steered based on the player style, intention and perceived agency.

The following section will address the idea behind an interactive adaptive real time storytelling and how it can be used to enhance the user experience in digital games. It will discuss how the player can be correlated with the storyworld based on interactions and present potential concerns for this research area. It will present possible solutions to the personalisation paradox in that the player can make an initial generalisable storyworld state with a data-driven approach and conform it into a personalised one based on continuous evaluated through the model-driven approach. A semantic constrain approach can help the epxerience representation and generation to be controlled.

*

In games it is common to see small variations in the game design as an attempt to appear dynamic, where the player have static game content that can be interacted with (Lopes and Bidarra 2011). When considering an experience-driven game, an interactive adaptive real time storyworld has to resolve in a change that correlate to the perceived agency, player style and intentions of the player (L. F. Yu et al. 2011; Thue et al. 2007).

An adaptive behaviour has been defined by (Shoulson 2013; Mazeika 2016; Schoenau-Fog 2015b; Jenkins 2004) as an alternation to a former state based on a previous one. This means that when a play style, intention and preferences are indicated through interactions or behaviour, it may change the storyworld entirely in a later temporal state. The alternation may be evoked consciously or unconsciously (Lima 2018). As mentioned by (Zhu and Ontañón 2020) a good tool is an invisible tool, and paradoxically an impactful one is conscious one (Spronck et al. 2006). For instance, a virtual character may initially be talkative, providing the narrative to the player, but the dialogue may change this character's behaviour, treating the player where they actual people (Larsen and Schoenau-Fog 2019). A potential narrative shift could happen, and the narrative would have to be acquired differently, such as by being persuasive or doing favours. The player experience may enhance from this as the player can define their own personalised storyworld, conforming the world towards their play styles, intentions and interactions. The author (L. F. Yu et al. 2011) refers to adaptation as the minima requirement for meaningful agency consisting of an adaptive intelligence, player model and generator. The adaptive intelligence would allow player's intentions, play style and agency to be a key feature in this adaptation loop, which may align the adaptive prediction with the player. An adaptive intelligence uses a goal deterministic approach in an adaptive storyworld to reflect and predict player's immediate goals and experiences (L. F. Yu et al. 2011). The adaptive intelligence aligns the player type directly to the performed interactions where these different 'types' are clustered. This achieves a mapping of the player types and preferences similarly to the player model presented by (Bartle 1996), where preferences are steered to increase the perceived agency. Predictions are an average of previous interactions and current interactions and follows the idea that the storyworld allows a prior state to become convoluted into a new enhanced state, based on either player style, behaviour and intention or an intelligent NPC, the but ignores that adaptivity in a storyworld is a temporal interpolation between its experiences and the interactions performed by the player. The approach is susceptible to the prediction problems related to the SAP model addressed in the player experience section, as goals are defined from solely previous states of interaction.

There are two different response methods to tackle the storyworlds and scenarios in an adaptive game. These are run time and real time. Run time refers to adaptation prior to the game play, which means that this modality is required to be data-driven. Real time refers to adaptation during the game play, which can support a data-driven, model-driven and hybrid approaches. This imply that the expected processing flux would be considerably greater and depend on how 'high-level' an adaptive module in the storyworld is. A high level framework is proposed by (Charles et al. 2005) and suggests that player modelling and performance are comparable to an adaptation model. Each adaptation is evaluated in terms of its experience quality. A low experience quality will yield in an update to the adaptation or player model. However, some literature shows advantages by extending adaptation beyond performance, progression and storyworlds. An example of this could be to have an integrated storyworld methodology where causation results in a chain of changes, because they are integrated to each other. Each respective change can propagate to the root of causation (Lopes and Bidarra 2011). A player model can help an storyworld adaptive paradigm with this extension and steer among other curiosity, challenge, fun, boredom, frustration and anxiousness (Lopes and Bidarra 2011; Pedersen et al. 2010) to for instance make branching and adaptive narratives personalised.

Preferences, intentions and play style can be derived from a data- and model-driven matrix that contains information about the player (Lopes and Bidarra 2011). Given an arbitrary state, such matrix can be used to predict an experience that align with the player. A storyworld can likewise have a matrix that cause in an adaptation. An appropriate storyworld experience would at least have to consider the predicted player experience. 'At least' because emergent occurrences and dynamic storyworlds may as well enhance the player experience (Charles et al. 2005). For the adaptive behaviour to become meaningful, it becomes easy to relate to the personalisation model paradox, because it should identify what storyworld elements that have to change. This requires the game to understand which storyworld elements yield in what player experiences. And secondarily, what should trigger in an adaptive change (Lopes and Bidarra 2011). As similarly to an experience-driven approach an adaptive change should not be determined to find a best possible solution; but instead resolve in an applicable candidate, that align with the player experience (L. F. Yu et al. 2011; Lopes and Bidarra 2011; Hastings et al. 2009). A different direction is taken by the author (Lopes and Bidarra 2011). To prevent the self-reinforcing personalisation paradox, adaptation is not achieved with pre-determined player modelling. An adaptive game is a possible solution to prevent this issue because as a technology it allows games to provide a personalised experience, that is responsive to diversity, such as player types, preferences and intentions. Regardless of the model (data, model and hybrid-driven) it should dynamically adapt the storyworld according to the player, such that the experience becomes personal and thus enhanced (Lopes and Bidarra 2011).

For a system that uses an interactive adaptation to convey an individualised experience it becomes eligible to have narrative non-linearity, whilst the narrative is to be authored, to retain the story integrity, coherence and chronology (Louchart, Truesdale, et al. 2015). In interactive open world games this becomes a problem because the narrative cannot be authored resulting in a narrative paradox. The narrative paradox is the conflict between authored narratives and the freedom that is offered by the storyworld. It implies that non adaptive games as a cost of narrative temporal-spatial fixation, subsides player's agency and freedom from interactions due to its narrative linearity structure. It implies that non adaptive games as a cost of narrative temporal-spatial fixation, subside player's agency and freedom from interactions due to its narrative linearity structure. Improvisational play (emergent storytelling) is proposed to prevent this narrative paradox, a concept that could use the storyworld where the player personally construct and expand the narrative. The Sims franchise and Minecraft are both games that almost solely depends on the emergent storytelling paradigm. Non-linear narrative structures have to consider the narrative management and structure because the narrative may grow exponentially more complex. In contrast to linear narratives the presentation of each narrative component happens with relation to the players and a fixed position in the storyworld. However, for a complex narrative its organisational structure and hence agency and freedom becomes difficult to attain. An intelligent agent that dictates the organisational structure therefore has to be considered to prevent the Combinatorial explosion. The Combinatorial explosion is the difficulty in maintaining a meaningful story depth when agency can exponentially explode the story in an interactive storyworld. This makes it impossible to implement a meaningful interactive story. There has to be narrative flow and the different narrative event has to be presented with equilibrium to remain engaging. It relates to game flow, where an exploded narrative becomes overwhelming, vice versa (Stern 2008). Façade is a game that uses a narrative intelligence (NI) system to map players' interactions to narrative storytelling. It presents a real time Drama Performance Manager to map actions with causation to prevent the Combinatorial Explosion (Mateas and Stern 2005). The dramatic pacing is controlled by tension level variable that synthesises the expression of the narrative, such as gestures, emotion and presented information. In regards to agency, façade has two different types of agency, a local that steers the overall experience based on interactions and a global that states the actions that have ultimately lead to a particular ending or narrative direction. The global agency is a continuous function that allows the player to understand what actions have lead to the narrative direction.

In general advances for interactive adaptive real time storyworld is scarce. One reason for this is that control, accuracy and flexibility is difficult to achieve (Lopes and Bidarra 2011). Given 2D games, the idea of adaptative storyworlds is directed towards experience predictions and the player. Difficulty, challenge, fun and levels are typically derived from from model-driven, data-driven approaches and player modelling. Generative scenarios in 2D games have shown to become relevant and an useful approach to engage and enhance player experience. However, the abstraction of the scenario defines the adaptation expression. Arguably this comes down to context. For instance when the goal is to get a key to open a door, the 'key' as a non-abstract object and has few adaptive states that it may acquire. The context is a design problem (Lopes and Bidarra 2011). For instance this key, could be considered as an abstraction where the storyworld to be adjusted; then the key could have a range of functions, appearances and tasks coupled to its retrieval as similarly to boomerang key in Legend of Zelda (Nintendo 2021). The same key may become different and dependent on the player, but game coherence and assets have to be assured. Semantic modelling is proposed by (Tutenel, Bidarra, et al. 2008) as a solution to this problem. It relates to how particular objects relates to the storyworld, that is its interactions capabilities and the relation to other objects. Semantic modelling can guide the generation layout of a storyworld.

Another proposed approach is declarative modelling (Bidarra et al. 2010). This is a modelling approach that uses semantic modelling to declare 'what' to change instead of 'how' to change something. Given the 'key' analogue, these semantic approaches indicate how and what, challenges, obstacles, tasks the player has to do to attain or use the key. In similar terms and as an extension to the semantic approach, a semantic service paradigm is presented by (Kessing et al. 2009), where services are defined as the interaction capacity of a particular object. For instance a key that can open locked doors. The semantic could be mapped to the player experience, so for instance a challenge is not too challenging.

Other advances for real time adaptation is mostly considered to train A.I game elements (S. Bakkes et al. 2009), NPCs (Peirce et al. 2008), adaptive elements, simple and generative worlds (Westra et al. 2010; Shaker et al. 2010; Yannakakis and Togelius 2011; Zohaib 2018) and PCG (Hastings et al. 2009). A.I would have much valid relevance in the area of real time adaptation for high level approaches. Because features [yadda yadday] can be taught. In serious, healthcare, adaptive simulations and rehabilitation games an adaptive behaviour can help creating personalised scenarios with, learning, assessment, effectiveness and reflection (Lopes and Bidarra 2011). For instance, in serious games, the player may try to overcome situations that suits them the least as to learn, in simulations it may introduce different situations, such as blood loss and mistakes during a surgery, rehabilitation may introduce gradually more difficult mobility tasks conforming a patient's capabilities. A virtual reflective agent has been given player data allowing it to asses performance and conduct interactive deep reasoning (Lopes and Bidarra 2011).

There are three distinctions that categorises adaptation, that is in regards to agency (experience adaptivity), the narrative (storyworld adaptivity) and the game elements (interactive adaptive realtime storytelling). The three categories are defined in definition 1.2, 1.4 and 1.3, and will be described in the following sections.

1.2.1 Experience adaptivity

Experience adaptivity, requires perceived agency because it is dependent on afforded interactions (L. F. Yu et al. 2011). Traditional games tries to appear adaptive and dynamic through a branching narrative. However, the narrative itself is not adaptive, because these branched narratives are pre-authored. An example why this may result in poor agency can be explained in The Walking Dead game. The only significance of agency is to decide which branch in the narrative the player is currently progressing along. With this kind of narrative limitation, the game does not provide great replayability and personalisation. A possible solution to pre-authored narratives is to have the experience adaptivity that impact the storyworld. An interactive real time adaptation could be a potential solution to this problem by allowing a game to be steered with an experience-procedural A.I. An A.I can use a data-driven approach to predict an appropriate adaptation based on play style and preference learning (Magerko 2008). But it has to understand which elements in the storyworld that correlates to which experiences. As also presented in (Shaker et al. 2010) jumps and enemies may correspond to perceived difficulty. Allowing reinforcement learning to evaluate on elements such as difficulty, may provide better estimates for experience predictions. An interactive adaptive real time storytelling adaption may change and regenerate accordingly to the player and their interactions (Lopes and Bidarra 2011).

Definition 1.3 (Experience adaptivity) *Adaptivity in a storyworld is a temporal interpolation between its experiences and the interactions performed by the player.*

How interactions is represented adaptively to the storyworld can be done in two different ways, that is indirectly and directly. Indirect adaptive player control implies that a behaviour has an impact on the storyworld. For instance, the player dies a lot, therefore the storyworld introduces to more places where health can be picked up or perhaps the difficulty is lowered. Whereas, direct player control implies that specific actions causes a specific change to the storyworld. For instance, the player may choose to mine resources, which result in minerals to become more expensive, thus increased profit or perhaps the player become more proficient in mining with a specific tool (G. Smith, Gan, et al. 2011). High direct player control results in a search space to become more complex in the storyworld. A stochastic gradient optimisation and evolutionary algorithm can be used to improve the search space in cases where player control is direct (Yannakakis and Togelius 2011).

A state of the art has used Staged Areas with adaptive real time storytelling in an static enclosed storyworld to achieve narrative control with space-time Drama Managers (Jordan 2020). This was achieved by mapping interactions to causation, relating to the experience adaptivity definition and thus a strong focus on agency and individualistic experience. The experience adaptivity, is crucial for the proof of concept because it allowed the player to feel how schizophrenia could potentially be experienced. The correct narrative sequences for the schizophrenia experiment has been handled by the space-time drama manager. The Combinatorial Explosion has been handled with dramatic pacing, controlled by plot tension levels. The game system uses interactions to determine the position of the staged area narrative, eligible the narrative position may change as interaction do. The literal experience where shown to be enhanced, but the change in a holistic experience where not investigated.

Game mechanics and design are two ways to steer an experience based on player interactions. The two are common areas that sees adaptive attention, the reason for this is that these operations are cheap, easy to implement, and difficult to notice (Lopes and Bidarra 2011). Game mechanics refers to how game elements works, such as the speed of the player, a temporary shield activation, a purification spell that removes de-buffs and temporary slow motion. Some games uses difficulty adjustment, aim assist, the rubber band strategy in racing games, such as Mario Cart (Jennings-Teats et al. 2010). In game design adaptivity is usually considered NPCs as an approach to appear intelligent, which can become very memorable (Larsen and Schoenau-Fog 2019). Attention has also been directed towards adaptive narratives, game scenarios, storyworlds to achieve a holistic and personalised experience (Booth 2009). PCG extends this as it can both generate and regenerate storyworlds. Efforts have been put into procedural narratives (Hartsook et al. 2011) and narrative pacing with drama managers, reactive scripts and Finite State Networks; and there is a mixed use of model-driven, data-driven and hybrid approaches in literature. There is also a diversity in which approaches are used for player models, such as Bartles player types, player interactions, SAP from previous players, questionnaires, decisions and performance, for instance.

1.2.2 Adaptive real time storytelling

The definition for an adaptive real time story has been proposed by (Larsen and Schoenau-Fog 2019), which confine the terminology to the player and their previous, unconscious actions, which result in an alternation. However, an adaptive real time storyworld, does not have to constrain itself to the player, but extend itself to the storyworld as well. As presented by (Nack and Gordon 2016) an adaptive real time storytelling, relates to the authors mentioned space-time drama manager. It allows the narrative to be intact because it can appear in the storyworld where it needs to be for the player in a space time continuum. This means that there is potential for roaming freedom and role playing. However, for any dynamic terrain this becomes improbable because any given narrative proposition cannot be accounted for along the players trajectory. This also implies that Staged Areas (Jenkins 2004) are likewise improbable if the storyworld that is expressed which imply the potential for a PCG. The reason for this is that these are authored, and to work in an adaptive storyworld a Staged Area has to become interactive and adapt to the real time interactive story telling and storyworld.

Definition 1.4 (Real time interactive adaptive story telling) *The storyworld allows a prior state to become convoluted into a new enhanced state, based on either player style, behaviour and intention or an intelligent NPC*

A state of the art proof of concept has used Staged Areas with adaptive real time storytelling in a static semi-open storyworld (Jordan 2019). This was achieved by trajectory calculus, where the correct narrative sequences have been handled by the space-time drama manager. The game system uses proximity logic to determine the degree of interest the player has in the current staged area narrative, eligible the narrative position may change. The literal experience where shown to be enhanced, but the change in a holistic experience where not investigated.

Advances in research adaptive storyworlds have regarded drama managers (Riedl and Bulitko 2013), storyworld from data-driven player modelling and profiling, dynamic difficulty adjustment (DDA) (Jennings-Teats et al. 2010;

Zohaib 2018). However, a disadvantage of these approaches is that they are genre dependent (L. F. Yu et al. 2011). For instance, games such as the X-Com series (slow paced) is subtable to fewer temporal interactions than a FPS (fast paced), but the engagement, fun, boredom etc, may be the exact same. Therefore, adaptation may include different parametric changes such as opponent modelling, in contrast to weapon dynamism as in (Hastings et al. 2009). Adaptation and PCG to some extend have to function under the assumption that restrictions are necessary.

1.2.3 Storyworld adaptivity

The storyworld consists of several elements, which helps provide a holistic experience. These components may allow experiences to emerge from, for instance characters, the musical score and the atmosphere. An interactive adaptive real time storytelling paradigm could be a solution to steer the player experience through player interactions. These interaction may refer to the player styles, behaviours and self identification. These interactions can be mapped to the storyworld to accommodate for an adaptive behaviour. It may conclude to be a potential solution to the problem risen in figure 3 and the idea of an directed and thus individualised player experience. The storyworld may be an entity that intentionally, derive curiosity and uncertainty based on the player style, interaction and behaviour (Carvalho 2017). This could help to the SAP to prevent a true-negative where segmentation and personalisation are falsely predicted as mentioned by (K. Smith 2013).

Adaptation in for the storytelling implies that the storyworld can adapt to the player based on a contextualisation module. This module could track the player's trajectory in the world such that the narrative becomes spatially rearranged whilst the narrative is held intact; this also refers to the space-time drama manager which attempts to account for the player's freedom and agency and thus making an individual experience possible within the storyworld. Agency is important in interactive stories because the sense of perceived agency creates a substantial enhancement to the experience. However, it becomes a problem in stories, because, the game is required to have an adaptive and NI that can account for predictable and unpredictable actions (Mateas and Stern 2005). Furthermore, which interactions that should cause in an adaption should be considered. For instance, killing the first person the player sees in a village may cause the remain to become rebellious or have contempt towards the player as a dynamic causation to this action. Overtime this may be forgotten, and a different scenario may unfold, where the player to do the same action.

Definition 1.5 (Storytelling adaptivity) *When the narrative discourse changes such that it is presented differently or through different means to preserve the narrative flow.*

1.3 Procedural content generation

The immediate problem with both the SAP model and the interactive adaptive real time storytelling is the lack of a PCG. A PCG can help the adaptation and SAP model to persistently align with the player. One way to achieve this is to have the storyworld be PCG dependent. PCG can help support an adaptive real-time storytelling, because it can function as a generative storytelling and experience tool. The technology can enhance the game experience in that a player experience model and an interactive adaptive real time storytelling adaptation can be fed into the PCG. This is possible because PCG expect a data input to extrapolate a potential output from. A PCG allows both a data-driven and model-driven approach at any given point during game play. This means that the capabilities of the PCG, can make the storyworld player dependent as the player can be thought as a series of temporal interactions that correlates to the player experience.

The aim for the following section is thus to address how a personalisation model and an interactive adaptive storyworld can be situated in a PCG, such that it can generate a storyworld that is based on player experience predictions.

*

Procedural generation algorithms have been around since the 1800s and used as a technique to demonstrate stochastic processes. The Botanist Robert Brown observed an inexplicable motion paradigms such as pollen on water; later, Mandelbrot extended the Brownian motion as a self-avoiding random walk. Fractals where developed to accurately simulate nature such as landscapes, clouds and vegetation (Blatz and Korn 2017). PCG has since then been well capable for storyworld creation. Examples of PCG for storyworlds could be, building blocks, pointers to create graphics (fur, water, materials, vegetation), effects (smoke, fire, growth), animations (walking, behaviours, adaptations, Motion capture and artificial intelligence) (Bergamin et al. 2019; Booth 2009), and content (environments, models, compositions, characters) (Unity 2019; Houdini 2020), music (Games 2018), and storytelling elements with the compliance of replayability, aesthetics and experiences (Blatz and Korn 2017). PCG lend itself to *concepts, map generation, sequence generation, ontogenetic concepts* and *teleological concepts* (WikiDot 2020). PCG is used as a methodology to invite experiences to occur but it, itself is not the experience (Schell 2008). It creates the possibility to enhance an experience based on the player (Guckelsberger et al. 2017) to cause a change for how it describes the storyworld and narrative. PCG could be used to reduce the gap between the storyworld and the player experience, eventually have the storyworld align itself with the player's experience, as also demonstrated in (Jordan 2020).

Since then PCG has inspired certain game genres, such as RPGs and roguelikes (Hastings et al. 2009). These games have a high replayability value in that storyworlds are randomly generated, the Diablo franchise and Dwarf Fortress are great examples of this. Recent studies, have been directed towards how PCG can be influenced by the player (Hastings et al. 2009; Dormans 2010). Player behaviours are traced and fed into an procedural-adaptive storyworld generation (Nitsche et al. 2006). a generative and interactive stroy telling based on interactions in façade (Mateas and Stern 2005) and Drivers performance have been evaluated on a virtual race track where a fitness function is used to optimise the race track accordingly to the performance (Hastings et al. 2009).

Definition 1.6 (Procedural content generation) *An algorithm that is automatic, self-organisational, based on rule-sets and constraints to generate content.*

Game content, rules, narratives and storyworlds are commonly generated in run time. The reason for this is that content is easy to control and test (Lopes and Bidarra 2011). Although, providing these elements at run time does not yield great personalisation and game play may become predictable. In similar terms as adaptation in storyworlds, games rely on few variations that may depend on player modelling, such as difficulty adjustments. A major concern for a PCG algorithm is the sense of randomness and structure. A well designed PCG storyworld consist of a mission [scenario] and generative environment (space) to accommodate for the missions. Spaces can be described by a tree structure with nodes and edges. A mission refers to a narrative with a series of plots. Allowing a mission to be independent to the space allows both to be enriched, and thus enhance the player experience (Dormans 2010). Missions are commonly made prior to the storyworld and not the other way around, this way both may become detailed and expressive (Dormans 2010). Linear narratives that takes a non-linear, or branching narrative structure may be prone to face a parallel challenges, in that the player have to back travel. Although, this parallel challenge is not always a problem (Dormans 2011). An environment that is revisited can have a strong emphasis to the narrative and player's perceived agency when the environment reflects the two making the level design less linear (Dormans 2010). Additionally, it provides opportunities for *Kihon* and *Kihon-kata* stage, first training then mastering (Dormans 2011). PCG control is important when considering a *Kihon* stage as learning should happen in isolation and steering player experiences (Lopes and Bidarra 2011). A storyworld can be controlled, organised, follow narrative logic, grow and regenerate in terms with the player's decisions, play styles and introduce to novel content which provide a highly player-centric experience as shown in various research (Hastings et al. 2009; Smelik et al. 2011; Shaker et al. 2010). There are several possible ways to achieve control in adaptive storyworlds as well, such as, rewrite systems, context, context-free and shape grammar (Dormans 2011; Lopes and Bidarra 2011; Hartsook et al. 2011; Zhu and Ontañón 2020), formal grammar, in this case Lindenmayer systems (L-systems) (Martin et al. 2010; Prusinkiewicz and Lindenmayer 2012) and Generative Graph grammar is suited to create narrative structures (Dormans 2011). And the authors (Hartsook et al. 2011) present an adaptive real time linear story that allow control through the storyworld and NPCs behaviour.

An interactive adaptive real time storytelling could take advantage of the a real time approach, but PCG cannot distinguish between order and chaos, it merely structures components based on its ruleset et cetera, implying a potential costly and even an infinite iterative process to satisfy this. However, PCG may use artificial intelligence (AI) because this technology can learn this distinction. A Markov chain is a stochastic process where a future state depends on the previous state (Gagniuc 2017), this makes it highly efficient for speech synthesis, forecasts or likewise predictions. A Markov chain can happen in discrete and continuous to time. The Markov chain can support a continuous-time state for the player because interactivity is unpredictable, and a state for the storyworld, by which both becomes interdependent. An approach like this Markov Chain would allow the player experience to be mapped into the storyworld, equation 1.1 shows this interdependent relation between two systems.

$$\hat{y}_{x_1} = \begin{bmatrix} x_0 \times m_{11} \\ y_0 \times m_{12} \end{bmatrix} \quad \wedge \quad \hat{y}_{y_1} = \begin{bmatrix} x_0 \times m_{21} \\ y_0 \times m_{22} \end{bmatrix} \tag{1.1}$$

This indicates that PCG can be crossmodal and the principle can be extended to comply with entire storyworlds. The goal of PCG in this project therefore becomes to make a storyworld that allows different player types and experiences to interdependently co-exist. Each player type have associated player preferences to them. The achiever has interest in acting on the world. The killer has interest in acting to players. The explorer has interest in interacting with the storyworld. The socialiser has interest in interacting with other players. The following sections will respectively address how these could be achieved.

It has been highlighted that a PCG can be crossmodal which is an important feature in that the storyworld and the player have to be interdependent. It is furthermore important that there are methods to control the product for the PCG which implies that it is plausible to generate content within a controlled boundary. For instance, a player within a particular player style may deviate. The following sub sections will from this basis focus on how a player model and storyworld can be dependent on a PCG. This interconnection will allow the storyworld to become procedural-adaptive and therefore more capable of presenting and generating an experience that aligns with the player and thus enhance the player experience.

1.3.1 PCG with player modelling

Player experience modelling can be used to predict and steer personalised and procedural levels on the basis of the qualitative description of 'fun' (Shaker et al. 2010). A data-driven approach has been used for the preference learning, post game play. The levels have been sequentially build in Super Mario, with a single layered perceptron to approximate game design features. The quality of this game design feature composition is evaluated by a multilayered perectron to match player preferences. A game design feature is considered as features that defines the game play, such as, jump difficulty, deaths, enemies amount and type. This quantification allows the game design to be a prediction of how fun, frustrating and challenging a particular level is. Frustration has yielded in the most successful prediction (89.33%), followed by fun (69.66%) and challenge (74.66%). A function approximation can also be used to ensure environmental expression with level variability and quality through function parameters and fitness quantification (G. Smith and Whitehead 2010). An implication of the proposed approach is how to determine a level's expressiveness. A solution to this concern is to use a series of functions such that parts of a level may afford multiple skills such as dexterity, fighting and accuracy. To determine the quality of the game and level design, two parameters can be declared, linearity and leniency (G. Smith and Whitehead 2010). Linearity refers to how well a level fits a function. Whereas Leniency refers to game design parameters, such as the difficulty of a particular level. The difficulty is quantified on a scale (+1 to -1) where different elements are scored herein between, gaps (+1), moving platforms (-0.5). The expression of a level is important because it may relates to a particular player style. Several authors have already been addressed to have implemented a framework that steers experiences through game and level design.

A PCG in an adaptive context can be considered to have a pool that is representative to a range of different storyworlds. Which of these different storyworlds that is most appropriate to the player has to imply a cross evaluation of the entire pool. A pool of 10000 different states yield in approximately 100 well perceived levels (G. Smith and

Whitehead 2010). A potential evaluation would suggest a function similarly to a Fast Fourier Transform function (FFTT) (McClellan et al. 2017) where each temporal step is defined by an associated player actions. Different player behaviours and experiences may then be directly reflected in a matrix similarly to the FFTT (Shaker et al. 2010). An evaluation method per-player-style and preferences is a different approach to include the player (Shaker et al. 2010). However, this will not compel well with the idea that 'a player' is a dynamic actor in the player type and preference space.

1.3.2 Procedural-adaptive storyworlds

As similar to an interactive adaptive real time storytelling, a PCG can also be considered temporally, that is, discrete at run-time and continuous in real-time. Storyworlds or simply game elements are commonly created at run-time. Adaptation would suffice greatly from a continuous PCG framework because it allows the entire storyworld to change dynamically to the player (Lopes and Bidarra 2011). However, an adaptive-procedural algorithm is likely to become temporally tilted, in that latter levels are defined from past player experiences. This may become a problem because predictions will be indirect, although a quality degradation is not implied (Shaker et al. 2010). The narrative is commonly used to establish context, motivation through task in the storyworld. In an adaptive context, it can contain a bidirectional function to the player experience, in that it can be the catalysator for change or cause. The reason for a bidirectional function can be illustrated in RPGs. A RPG shows some concerns with the player and environment, in that it has a strong emphasis on exploration and progression. Players may show a diverse behaviours and preferences for the narrative content and progression. Examples of this could be the time exploring, engaging in combat and scavenging. The attempt to provide a potential great experience for diversity quickly becomes improbable, where these games' content may only be suitable for an insignificant range of the player behaviours. Search based PCG is commonly used in single aspects of the game design and level design (G. Smith and Whitehead 2010; Lopes and Bidarra 2011; Dormans 2010) instead of highly complex and entangled that may be more likely to align with a particular player experience.

A steering mechanism to steer difficulty through storyworld representation has been proposed (Jennings-Teats et al. 2010). The difficulty is steered by a DDA, although this framework can cause a generalisation problem in that players may refer to difficulty differently, such as jumps, enemies, environmental composition, resource management.Some games is build around the idea of difficulty and is a part of what makes the game fun and engaging, Kaizo Mario (Ryan 2021) and Dark souls (Namco 2021) are examples of this. An un-authored example is a data-driven and search-based PCG framework (Lopes and Bidarra 2011). This framework has been used to make personalised Super Mario levels based on the player's skill level. It uses a run time PCG and storyworld adaptation and the focus has been on storyworld generation rather than storyworld purposes. In regards to PCG storyworlds a good strategy is to generate the narrative components prior to the storyworld (Dormans 2011). This is to ensure fidelity in the storyworld while at the same time maintaining narrative congruity. The storyworld is dependent on the narrative, but not the other way around. An interactive generation allow the storyworld to change during game play as a result of narrative act. For instance, narrative elements or opponent types that are interacted with can diminish over time, whereas a novel generation of the narrative elements or opponent types are increases. A fractal interactive adaptive storyworld can help with this idea, in that more detail can be offered towards the player's interest (M.-L. Ryan 2015).

Games with a steep learning curve may also succumb to a difficulty adaptation problem. Adaptation can be controlled by ranking levels in a list based on the data-driven approach where each list inquiry is compared to the player performance and selected if it aligns well. Other games also use to adjust the difficulty dynamically, such as delimiting resources. Evolutionary algorithms could be used to personalise the idea of difficulty towards a play style and preferences. A model of the player and the difficulty is needed to align the appropriately. The authors (Jennings-Teats et al. 2010) uses a data-driven and machine learning build the player and difficulty models. The storyworld is constructed from a genetic algorithm (GA) and optimised with a fitness function and requires a comparison across several generations to find a potential great experience. However, environmental scalability, causes problems to this method. Narrative stagnation happens for small environments, vice versa. Nevertheless, as a storyworld is successfully made, the narrative may become conflicting due to adaptivity. As a result, NPCs may become omniscient in that the narrative may change as the storyworld tries to align itself to the player experience. This is a problem for acting, character traits and character personalisation. A similar approach has also been used by the authors (Dormans 2010) to generate level missions.

Reactive scripts can help make the narrative and storyworld to become dynamic and adaptive towards the player experience. A reactive script can be considered as an omniscient narratee that designs the world for the respective reader. The author uses an approach similar to Staged Areas, where each area functions as a plot point of a particular narrative type that refers to a semantic pointer to a narrative composition. What becomes adaptive is the temporal condition of these narrative compositions, not the content itself. As similar to (Hastings et al. 2009) a storyworld is initially presented pre-authored, but become more personalised as the player progresses in the game. Narrative elements are added and removed until the particular player preference and style potentially aligned. Alternating scenarios and branching narratives are proposed to account for the Explorer (Hartsook et al. 2011). The storyworld can become a model that is characterised and optimised by the player model, how the storyworld plaits the player model and how they are congruous with the storyworld. The author uses a data-driven approach to guide this storyworld coherence. Regardless of the used PCG-adaptive approach there may be cases where the PCG model and the storyworld are not negotiable. As a solution to this a priority system is proposed (Smelik et al. 2011). The environmental feature or semantic constrain with the highest priority determines process done to the environment. This is to prevent inconsistent and incoherent environment. Environmental consistency and coherence is achieved in the storyworld, in that the adaptation module is blended into the environment. It can be considered as a convolution operation that respect both the PCG model and the storyworld. Kernel operations such neighbouring-processing is a capable approach that can achieve such environmental consistency. This would also help to prevent the 'maintain coherence' problem as addressed in (Lopes and Bidarra 2011) where storyworld coherence is disrupted when it adapts towards the player.

A concern with an adaptive-procedural is its cross-platform generalisation capabilities, such as traversing from a 3D game to another, but different 3D game while utilising the same framework. The reason for this is due to game design, which may be different across games and thus be influential to player experience.

1.3.3 Procedural grammars

Grammars can define the transformation prediction in a logical way. Shape grammar is a step wise rule-set that is used to generate a product (Dormans 2011). Adaptative control can be achieved by selecting among weights or stochastic probability that potentially best fits the current generation module of the shape grammar. Grammars are commonly used to present low-level architecture (Martin et al. 2010). In similar terms to fractal geometry, it can be used to control level of detail, hence populate storyworlds, progressive difficulty and narrative events (Dormans 2011). Shape grammar has been used to interatively generate highly detailed cities (Müller et al. 2006). Context grammar is context-dependent where the left and right side can be covered by terminals and non-terminal symbols, figure 5 illustrates this with the key analogue. Context free grammar is a set of rules that recursively generate a series of strings. Formal grammar is similar to the context free grammar, but it uses a series of strings as production rules to generate or regenerate strings. These grammars are used to define change to a system from a starting point based on logic, variables, constraints and rules (Martin et al. 2010). Logic and rules could be variables that represent possible states that can be adapted towards the player. A rewrite system such as these grammars is very strict in that it allows only operations that are presented by rules. However, a rewrite system can create narrative events whereas another function as a drama manager for these narrative events.

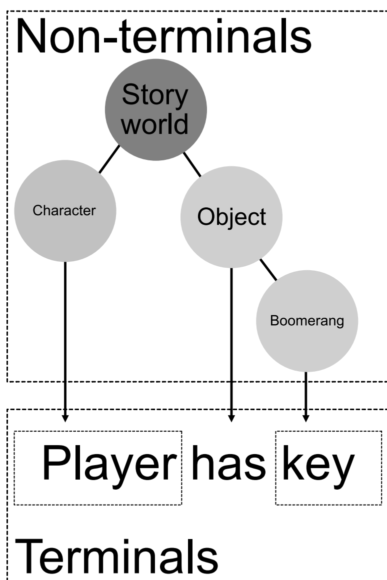


Figure 5 illustrates a string of non-terminal symbols that define a terminal output

Bringing up the key analogue a second time can lead to a research area concerning level design as model transformations. A model x can relate to model y such that y can be predicted by x, thus an experience transformation prediction p can be defined as $\hat{p} = [\ x \ \rightarrow \ y\ ]$ (Dormans 2011). This means that x can essentially dictate y entirely. An example of this idea is the boomerang in Zelda which. The boomerang can in a literal sense function as a weapon or symbolically as a key. This principle allows the game to be different to accommodate for this second function. It may as well relate to the 'first training then mastering' in that mastering the boomerang will lead the player to the door which it opens or objectives that will eventually lead to the door in a non-linear branching fashion. This also goes in hand with the game design perspective which is, the mission is created first, then the environment to accommodate for it (Dormans 2011). It is important to note that linearity does not dictate an interesting level, non-linearity tends to provide more interest when it represents the player and the narrative can be re-written.

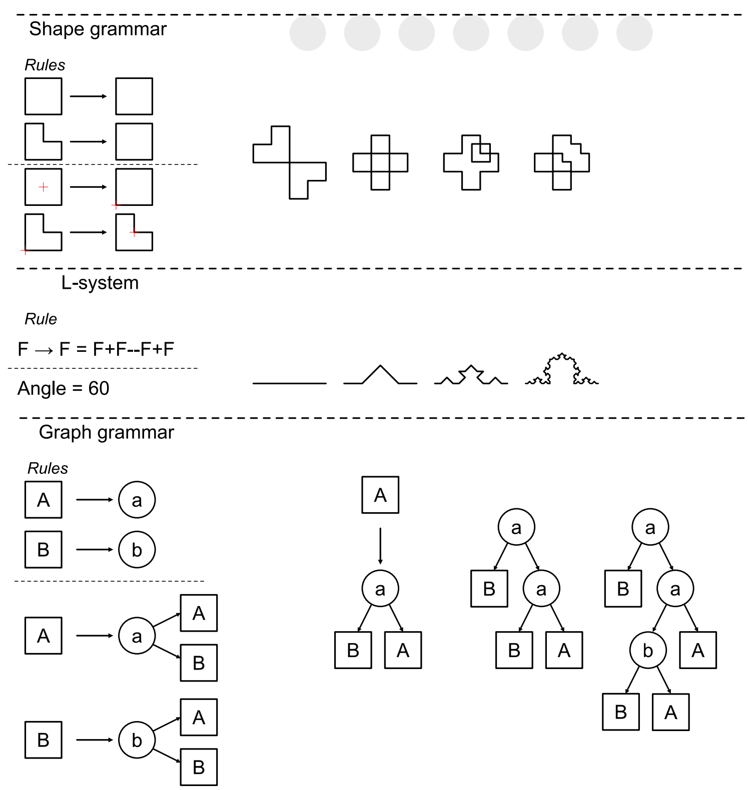


Figure 6 shows different grammar logic, that can help provide control in PCG storyworlds

L-systems are capable of generating environments and narratives from constraints and rules. common techniques includes generative trees, vegetation, landscapes and essentially a storyworld (Yannakakis and Togelius 2011); and it can guide its generation from player experience modelling. It uses a formal grammar variant that uses a parallel process instead of sequential, as similarly to shape grammars. This means that activation rules are performed simultaneously. A L-system that acquire the use of a terminal function is referred to as functional L-system. The terminal function can act like a mutation to the activation function. An example of this is when player interactions changes the storyworld or narrative. This terminal function can thus attain the player experience (Lopes and Bidarra 2011). A terminal function replaces poor predictions with predictions that is thought to be representative to the player. An example of this has been showcased in (Hastings et al. 2009). Terminal functions are appropriate for high-level features, whereas terminal symbols are appropriate for low-level features. Terminal function allows the rule-system to be more simplistic than terminal symbols (Dormans 2011). A terminal function may provide an adaptive intelligence such that they can align with the player (Yannakakis and Togelius 2011). The ideology can be extended to whole storyworlds. Commonly, a L-system is used for generative natural structures due to its 'growth approach', where as a sequential algorithm is suitable for structures and their characterisation (Martin et al. 2010). An adaptive semantic read-write file has been proposed to create different scenarios in simulations (Martin et al. 2010). This file is structured such that it defines structures, but not the execution as the progress is defined adaptively. However, the approach presented in (Martin et al. 2010) is domain specific, which has the potential yield in specific player desires. For simulations these desires may be knowledge, skills, attitude, training. A valid problem is to achieve the same 'desire quality' across several domains with the same framework. The read-write file can contain semantics for teammates, opponents, weather. A trigger can cause a series of events to occur. Bad weather may for instance results in a combination of wet roads, that causes bridges to collapse, thunder-storms, rain, little visual acuity. A dynamic change of such parameters will allow the player to change behaviour (Martin et al. 2010); allowing these parameters to be player dependent will enhance the player experience.

1.3.4 Advances in intelligent PCG

There have been advances in how artificial intelligence can be linked with PCG. The idea being that such system can become adaptive, real time and steered towards player experiences. A reinforcement learning variant has been already been used to achieve this (Ramstedt and Pal 2019). Normally in reinforcement learning the storyworld is separated into agent and environment. It is therefore, common for a reinforcement learning algorithm to assume that the storyworld is unknown and an A.I has to rely learning on data that the game can provide. This is one reason why reinforcement learning is not well suited for real time applications. For instance, the agent observes a given task, and then perform a temporal dependent task. A reward and Markov chain system can become useful in experience-procedural and adaptive paradigm because they are quantifiable and is capable of providing an agent-environment relationship (Ramstedt and Pal 2019; Hastings et al. 2009). This means that the agent (actions) and environment (states) are interdependent and thus changes simultaneously. Actions that take longer than real time are postponed til the next observation. The agent-environment dependency allows agents to take immediate actions (Ramstedt and Pal 2019).

One reason to change the storyworld procedurally is that immersion is directly impacted by the quality of the storyworld (Smelik et al. 2011). A semantic constrain approach where intention and context are mapped into the adaptation. The semantic constrain approach orient itself around high level features, shallow control and low level features, parametric attributes to attain specific components, both which can be layered. Context detection is used to test the capacity of the constraint and how it can be applied to different situations. An procedural-adaptive intelligence can use the player to evaluate the quality per-level (Shaker et al. 2010). The advantage of using an A.I is consistency and efficient experience evaluation. An A.I for a procedural-adaptive framework can occupy one particular play style, and evaluate this over a series of storyworlds with a run time data-driven approach. Although, a player may catch issues the A.I would otherwise not have. PCG can pragmatise a storyworld that constantly align itself with the player's experiences, intentions and behaviour, engaging them for longer (Hastings et al. 2009). To achieve this, the authors (Hastings et al. 2009) propose content-generating neuroEvolution of augmenting topologies (cgNEAT). The advantage and disadvantage of this approach is its extensibility. For instance, in linear narratives where the player acquire a weapon x, at some temporal time y, is not a good condition for the cgNEAT, whereas, games such as Borderlands or other semi- to open worlds provides an excellent condition.

Visual novelty can be achieved from an interactive evolutionary computation (IEC). An IEC tries to automatically generate graphics and game content from past player's preferences and evolve content of interest for subsequent players. A preference is indicated by how frequently an object is used, and object's choice of selection. The game is viable for multiplayer where the adaption is steered based on tendencies and usage and single player where adaptation is based on usage. In this manner the player replaces a fitness function (Hastings et al. 2009). The reason for this is that a fitness function cannot formalise aesthetic appeal that aligns with the player. For this reason an IEC is commonly used for generative graphical content. An IEC presents a generation queue with mutated members of an initial population state. The player determines which mutation in the generation queue will be considered in further mutated generations by showing interest in a particular queue item through interaction or selection. A L-system can also be used as an alternative approaches to achieve graphic novelty in 2D and 3D imagery and animations when coupled with linear and nonlinear functions, fractals, and Finite State Automata. The pattern generalisation system for the IEC can be biased towards tendencies, periodicity regularities with an activation function. For instance, a sinusodial may afford repetition, while symmetry is achieved with a normal distribution, linear functions yield in straight continuous lines and non-linear in dashed. Convolution can provide interesting patterns, combinations and visual novelty; since the IEC algorithm is based on function, infinite resolution can be attained (Hastings et al. 2009). Another approach to achieve visual novelty is to use an unsupervised neural view-dependent rendering algorithm has been used to procedurally generate a continuous photorealistic volumetric (voxels) landscape (Hao et al. 2021). It uses a pseudo pair, a segmentation mask (referent) and pseudo-ground truth images to generate environments. Two processes happen to the referent, it is fed to a Generative Adversarial Network (GAN) from where the pseudo-ground truth image is generated. A GAN is commonly used in image processing, such as image synthesis and generative images. Secondly it is sampled, per pixel and fed to a convolutional Neural Network (CNN). A CNN is commonly used for segmentation, classification and object recognition tasks. This CNN is used as a real time environmental render. The voxel has fragment and vertex properties that can be taught for composition. A style encoder allows the synthesised environment to attain a particular style, such as dawn and dusk. The approach separates itself from the state of the art in that it does not require real images to generate image-to-environment content. The principle in this work has been to generate highly detailed geometry from less detailed examples. This idea becomes interest for PCG because it allows a world-to-world generation from simplistic data (Hao et al. 2021). An example of this is image-to-world data to create PCG environments. Similar approach uses procedural filters (kernel) (Tutenel, Van Der Linden, et al. 2011) and voxel detail enhancement (K. Smith 2013). The procedural filters allow particular situations and attributes of objects to be mapped onto an environment. The approach uses semantics that describes the world, where a filter allow a particular change to the information herein. The procedural filters can define styles, textures, situations, effects and geometric and object infrastructures, independently of the environment. Applying a filter can become an automatic process based on pragmatic operations, such as object recognition.

The implementation addressed in (Hastings et al. 2009) highlights three problems with an intelligent PCG framework, the first two are, how fitness is to be computed and how sustain an equilibrium when some players are more influential than other by playing the game more. The solution to these problem is to positively change the fitness linearly for objects of interest and negatively for disinterest. Potential favouritism will in this way not disproportionate the equilibrium. The third problem is the immediate IEC prediction for the player, as there are no past data to base player preference from. Every player will be presented with authored content to choice from such that it is representative to the algorithm's search space and to prevent a poor impression. Since all players starts with the same content, it can be excluded from evolution process. However, this also imply that the algorithm needs a method to start the evolution. A stochastic probability pool that contains authored content, pseudo-random items and data from predecessors and player testing has been proposed. Another concern is that several PCG algorithms are needed to create the same world under different circumstances (K. Smith 2013). However there is evidence that this may simply not be the case (Yeh et al. 2012; Henderson et al. 2017). Voxel detail enhancement works by applying neighbouring processing (Moeslund 2012) on an image to determine which neighbouring pixels are similar. It is an image processing technique that allows kernel associated processing, in particular scaling. Recursive subdivision can produce a three dimensional kernel patch for each voxel to refine quality with resampling.

1.4 Conclusion

The analysis has sought to address the current research for an interactive adaptive real time storytelling and how a storyworld experience can be driven by the player. In the process a potential solution to the initial research question has been proposed. The research has highlighted different areas for interactive adaptive storytelling and storyworlds including how a PCG can help bridging the gap between the player experience and the storyworld. A PCG can be crossmodal which makes it a capable method to achieve a continuous experience alignment in the storyworld. Although, this requires that the player can somehow be predicted, as a solution to this a SAP model has been proposed. Based on agency and behaviour, this model can predict which player type and preferences that yield in an enhanced experience. Bartles player types has been proposed as a segmentation model where player intentions and behaviour can be mapped in regards to possible experiences. Two data elicitation models can be used to acquire player interactions and behaviours, these are a data- and model-driven approach. Both methodologies can yield in and personalised experience, however, the data-model approach can be used to set initial storyworld state and steering directions whereas the model-driven allow an experience alignment to happen during game play. A continuous player experience evaluation may also help prevent the personalisation paradox which is a concern for data-driven approaches. Both the combinatorial explosion and narrative paradox a concerns for generative storyworlds. However, semantic does show a potential solution to these problems in that they can control PCG and adaptation such that the narrative does not explode. The player may through intentions and behaviours determine narrative occurrences. Allowing the narrative to be based on the space-time drama manager and player makes it possible for storyworlds to prevent the narrative paradox. The reason for this is that the narrative can follow the player trajectory.

The following table summarises the key components from the analysis in descending order and how they respectively may help enhance the player experience in and interactive adaptive storyworld.

Research area	**Advantages**	**Concerns**	**Solution**
Personal modelling	Definitive and personalised experience	Confusion and precision	Player dependency and continuous evaluation
Bartles player types	Definitive player types and preferences, solution to content generation and representation	Player dependency and experience mapping	Player interactions
Holistic experience	Provide depth to player experiences	Player translation and storyworld quantification	Including affective remedies
Data- and model-driven approach	To make a PCG meaningful and feed the personalisation model	Player translation and storyworld quantification	player dependency
Agency	Player experience	Combinatorial paradox	Semantics and player dependency
Interactive adaptation	Player experience alignment	Generative method and player dependency, representation	
PCG	Generative player dependent model for continuous adaptation and alignment, cross-modality for player and adaptation	Player dependency and experience quality	
Procedural grammars	PCG and adaptation control, generative solutions within a player type space	Player experience quality	

This section has answered the initial research question, the table above shows the key components that answers this question. And from this stand point it is strongly indicated that a procedural-adaptive storyworld can be driven by the player experience. This means that such storyworld should be capable of providing a greater experience than in traditional sense. With these components it has motivated the following final problem statement.

Final problem statement

How can a procedural-adaptive storyworld be generated such that it enhances the player experience?

2 Case study

The analysis has explored the current research for an interactive real time story telling narrative, PCG, play styles and how these can be linked to provide an enhanced player experience. Thereby, provided a theoretical framework; a potential solution that uses a procedural-adaptive method to enhance an experience and at the same time potentially prevent combinatorial explosion and narrative paradox. Although, in practice the current field of study does not imply how a procedural-adaptive storyworld result in a great experience, but instead, a mere proof of functionality.

This case study therefore aims to extent the knowledge by introducing two areas that seem to be lacking from the current investigated research. The areas that has lead to the motivation of this case study are a high level description for what makes an procedural-adaptive storyworld in games a great experience for a player. The current research investigates what makes a potential great experience, but lacks how it can express a great experience. Seven games have therefore been investigated in regards to great and bad player experiences given their implementation of procedural-adaptive storyworlds and player experience. In general, a meta analytical approach to such games becomes difficult because game systems are rarely apparent, publicly announced or directly observable (Dormans 2011). The seven games are specifically chosen because they respectively orientates themselves around, player experiences and interaction without loss in narrative control, and the use of procedural and adaptive systems based on player interactions. Borderlands illustrates that an experience can be enhanced when the PCG aims to afford player styles and preferences; whereas No Man's Sky shows the consequence when this is not the case. In the Binding of Isaac novel storyworlds are procedurally generated with respect to player styles and preferences though choices. The Walking Dead shows that choices can steer the narrative adaptation in a storyworld. Spelunky shows that semi authored PCG can provide novel storyworlds whilst respected play styles. Left for Dead present a potential solution to the combinatorial paradox. Whereas Shadow of Mordor provides a solution to the narrative paradox.

Secondly, the current field of research addresses what will be referred to as a 'direct adaptation steering', but lacks an understanding for a potential 'indirect adaptation steering'. It may thus be an extension for how the player and the story would could become more interconnected, unpredictable and in this way enhance the player experience. This extension has been motivated by the absence for an indirect adaptation steering and the work done by (To et al. 2016; Carvalho 2017; Mazeika 2016; Shoulson 2013; Zhu and Ontañón 2020).

*

2.1 Games with procedural content generation

The following sections put emphasis on how PCG is used in popular games. The aim is an exploratory insight into how a PCG can be implemented and the player experience it resolves. Where academic research regards specific game design, level design and storyworld alternations, it may broaden the reasoning for how a PCG can provide an enhanced experience, in contrast to a theoretical player experience enhancement method. An exploratory research design has therefore been used to determine how PCG provides an enhanced experience. The four proposed games are interesting to analyse because they may succumb to the same technology but achieve different experiences. The communities, steam and meta-critic are used as a basis for how success these games have been. They will respectively be evaluated in regards to how great the experience is perceived to be. Although, it should be noted that a meta analytical approach is a concern as games commonly do not present the integrated technologies, such as adaptation or procedural paradigms.

*

2.1.1 Borderlands

Borderlands is a semi-open world looter-shooter game, which orients itself around the Explorer and Killer. It is a game that uses exploration to discover new weapon variations, and these variations support a highly personalised experience and different preferences for the Killer play style. The game uses PCG to create a vast variation of low level weapon variations without having the developers to exclusively design them (Blatz and Korn 2017). There are several parameters from which the algorithm can change a weapon's properties and appearances, such as firing speed, damage type, synergies. The PCG works by generating content for one of the replaceable components of a weapon as illustrated on 7. A replaceable part is indicated by the respective colourisation.

Though not procedural generated, Dead Space and the game Metro uses likewise weapon deconstruction, making this type of procedural generation unconscious at best. Borderlands shows that entire storyworlds does not have to be procedural generated for the game to be enjoyable and engaging, here it is the weapon variety that is.

Figure 7 Illustrates how a PCG can generate different types of weapons by interchanging between weapon components. Each coloured regions of the weapon can be generated and changes to afford a preference for the Killer player type.

2.1.2 The binding of Isaac

In contrast to Borderlands the focus is not on low level storyworld elements, but a high level generation of the storyworld itself. The binding of Isaac is a roguelike game that appeal to the Explorer, Achievers and Killer player types. The explorer is emphasised by introducing to a series of levels (depths) where each depth has several connected rooms and a boss at the end. This layout is unknown to the player although it becomes apparent as the player explores the level. Each layer contain exactly one secret, and treasure room that can be found; the two rooms present power-up items that gives the player an advantage. The Achiever is emphasised by the player's capability to find this secret room, gather all possible power-ups, characters and endings. At the end of each Layer a boss has to be defeated to progress in the narrative and by during so the player have the option to choose between three power-ups. The idea that the player has an option to motivates what is referred to as a 'clean run', where the player completes the game as it was initially started. The player can thus achieve to be powerful, 'clean' or build a specific personalised set of power-ups. The Killer is emphasised by the arena like rooms where the enemies inside these rooms have to be killed to progress.

The Binding of Isaac uses procedural level generation (McMillen and Himsl 2020), where symmetry and the sensation of 'human made' architecture are achieved by creating rooms that are decorated as of halves and then mirrored. Room, enemy and item variety are achieved with rules, a room decorator, population sparsity and stochastic list selection of ranked rooms and enemies, in terms of difficulty. The storyworld is generated in layered sequences, which means that the desired layout is built, then special rooms and then collectables. The PCG keeps track of the storyworld building, to prevent repeatability and impossible level designs. In The Binding of Isaac Rebirth, the levels designs are generated sequentially, where the algorithm selects from a list of predefined building blocks. This alternative approach resolves the bottleneck of infinite solutions and repeatability.

Figure 8 the same room in The Binding Of Isaac shows how procedural generation has been used.

2.1.3 Spelunky

The advantage that comes with a PCG is that a random number (seed) can be fed into the algorithm, reducing file sizes. Seeds is a proficient way to test different products of the PCG, in that it can function as an identifier. This has allowed the game Spelunky to be thoroughly tested, to achieve fairness and prevent unacceptable products such as difficulty spikes (D. Yu 2016). Spelunky is a two dimensional platformer that uses the same grid based system as in The Binding of Isaac. As similar to The Binding of Isaac Rebirth, all the levels appears to be pseudo-random and uses the same predefined building blocks to generate levels but the distribution is weighted. The PCG algorithm in Spelunky emphasises on its premise and creates fun and engaging levels through a fine balance between PCG and the authored content that helps the game not feel confusing, random and unfair; something that seem to be more valuable than realism when it comes to making an immersive experience (D. Yu 2016). It makes the player value rewards, risks and consequences much more. The explorer is emphasised in that all levels can be completed without the extensive use of tools, such as bombs, rope, however, the good items and secret areas require the use of these. Whereas the Binding of Isaac provides power-ups to make it easier to kill opponents, Spelunky orients this around exploration to make that easier.

Spelunky also emphasises on the Socialiser and Achiever player types. The game has a 'wanted' system that is intensified each time the player shoots or steal a shopkeeper. A social good standing with the shopkeeper circumvent this and the game supports multiplayer. In contrast to the Binding of Isaac, the game situates an arena like levels but without intent of the Killer player type, as the player does not have to kill opponents to proceed. As a consequence, opponents are unpredictable and difficult to overcome, which makes avoidance the better option. Much like The Binding of Isaac, Spelunky has several hidden levels, items and endings.

2.1.4 No Man's Sky

No Man's Sky is an excellent example to investigate because the experience has changed dramatically since its initial release.

No Man's Sky is a game that is, to large extend is (if not solely) based on a PCG which yield in an almost endless storyworld, and thus a strong emphasis is put on the Explorer player type. The goal achieves its premise, that is to use a PCG to create 'visual novelty', cosmos, planets, including and 'inhabitants'. Although, the game is so vast with few nuances in the PCG that this player type lacks purpose, because what is presented and generated becomes the same. This lack of player style, is an essential reason why people tend to dislike No Man's Sky. Essentially, the Explorer player style is degraded by the tool that is build to enhance it. This shows that an experience cannot be exclusively dependent on a PCG; but instead a PCG should allow to be fed substance, depth and potentially some sort of authored content in its evaluative paradigm. The case study in (Nitsche et al. 2006) indicates the same problems in No Man Sky.

Comparing the initial game state to the present version, the difference has become its capability to emphasise on player styles have improved. The storyworld has become much more dynamic, variant and complex as an attempt to depict reality and previously unexplored ecosystems. It hence, provide an experience where new occurrences may happen on every planet and solar systems. There are different resources, resource management and meaningful objectives. The game is now multiplayer compatible, implying the Socialiser play style. Customisation and companions to emphasise on the Achiever and personalisation.

The four investigated games indicates that PCG needs to afford play styles and relate to the game premise for the PCG to enhance the player experience. This implies that a PCG cannot merely rely on itself, but the storyworld has to appeal to the player. The following sections illustrates how a PCG can help an adaptive behaviour to enhance the player experience in similar terms and generate situations that afford play styles and the game premise.

2.2 Games with adaptive elements

The following sections put emphasis on how adaptation in a storyworld is used to enhance the player experience in popular games. The aim is an exploratory insight into how an adaptation in the storyworld can be implemented and the player experience it resolves. It highlights the potential for including a PCG approach in that the player experience can become more individualised as the PCG allows the adaptation to follow the SAP model with a model-driven approach. An exploratory research design has been used to determine how an adaptative storyworld provides an enhanced experience. The three proposed games are interesting to analyse because each provide different approaches to an adaptive storyworld. The communities, steam and meta-critic are used as a basis for how success these games have been. They will respectively be evaluated in regards to how great the experience is perceived to be.

2.2.1 Adaptivity by behaviour, Left For Dead

A procedural-adaptive structured unpredictability to populate the storyworld framework have been implemented in Left For Dead to afford replayability, and player-to-environment dynamics (Booth 2009). The adaptive system automatically evaluate individual performance, behaviour and pace in terms of a stress score during game-play. Player's respective or combined score is used to introduce situations to the player. In a procedural structured unpredictibility framework a list of both temporal and spatial functions that determines where and when the population function occur is generated.

Situations, collectables and population are introduced in correlation to an adaptive dramatic pacing where situations happen with respect to a frequency function. It creates uncertainty as to were enemies may come from, resulting continuous suspense and different narratives. As a consequence, weapons, health and special ammunition becomes more scarce, the better the player performs. The technique, utilises a navigation mesh, flow distance, potential visibility and active area set. Figure 9 illustrates this dramatic pacing. Alternatively, the game Residents Evil 4 links a DDA to player performance to manage game flow (Csikszentmihalyi and Larson 2014). Heavy Rain has an adaptive narrative behaviour where the narrative and endings are steered based on player interactions. This is to provide a personal gameplay experience. The dramatic pacing makes situational game flow with panic situations, mob rush, bosses, special infected, all which may happen concurrently. Based on these situations, the system introduces unpredictable peaks, none persistent influxes as these are less engaging. The dramatic pacing makes sure that the peaks are reliable, but not overwhelming. Since the player is dynamic and their skill may alternate, the world population is procedural and the timing and location of unpredictable peaks will therefore differ.

Left For Dead emphasises on the Killer and Socialiser. The killer is resolved through a 'run-and-gun' principle and mayhem. The game does provide multiplayer and where communication and team work becomes important, as the difficulty increases. Although, in game achievements does provide the game with elements for the Explorer and the Achiever.

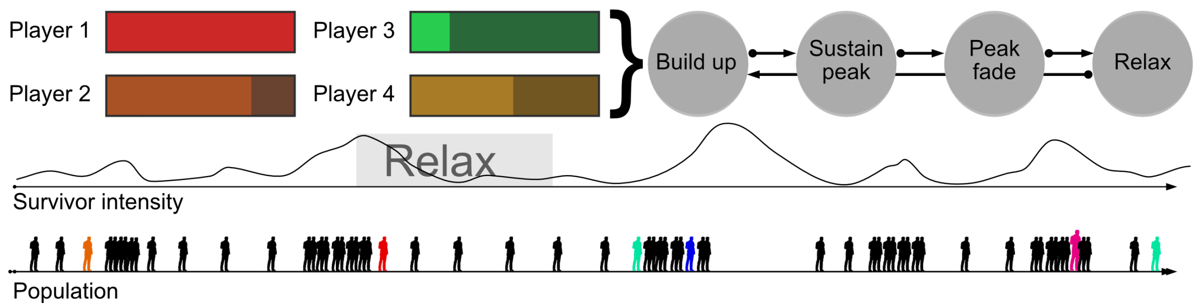


Figure 9, shows how the game L4D2 lend its players to an adaptive behaviour. The coloured units on the population curve are special infected.

2.2.2 Adaptivity by decision, The Walking Dead

The Walking Dead is a game that emphasises on the experiences that arises from decisions. The narrative is authored, but the storyworld is emotionally impacted by decisions, which from a player's perspective shapes personalisation and the storyworld. The player's personality is situated in terms of moral, being human, empathy, virtual status. Although, The Walking Dead has received conflicting opinions because the player adaptive control is indirect, which stands in contrast to player expected direct control, creating an adaptive expectation mismatch. For instance, amputating Lee's arm or keeping it whilst it is infected, does not changes Lee's faith, but Lee and the other characters' perspective does adapt correspondingly to what decision was taken. This means that the player can change the experience throughout the game, but not the storyworld itself, after all Lee dies either way. The player expects the storyworld to change not the emotional state. The game illustrates that there are two types of adaptive acquisitions for player's decisions in digital games. Although both, is a subset of definition 1.3, the storyworld allows a prior state to become convoluted into a new enhanced state, based on either player style, behaviour and intention or an intelligent NPC. There are decision that take part in the storyworld as perceived agency, and decisions that changes the narrative resolution. In The Walking Dead a change to the narrative does not imply that the narrative is rewritten, as also formally illustrated in figure 10.

In Until Dawn, performance, and decisions result in different endings, most of the time. 'Most of the time' implies that the game actually cannot alternate its authored narrative due to its strict progression. However, as is the case, providing the player with an ultimatum provides suspense and the ending has much greater impact. The walking Dead and Until dawn is essentially two comparable games, but the premise is different which help prevent a potential expectation mismatch.

2.2.3 Consistency in the narrative, Middle-Earth: Shadow of Mordor

Middle-Earth: Shadow of Mordor has a narrative system that extents its narrative based on player interactions. For instance, the player may retreat a battle, but entering the same battle later this retreat will be mentioned. The game uses PCG to make all the bosses in the game, and it is therefore important that these characters are memorised. This is achieved but generating a reference list, where mention-worthy events are stored, such as retreat, getting killed

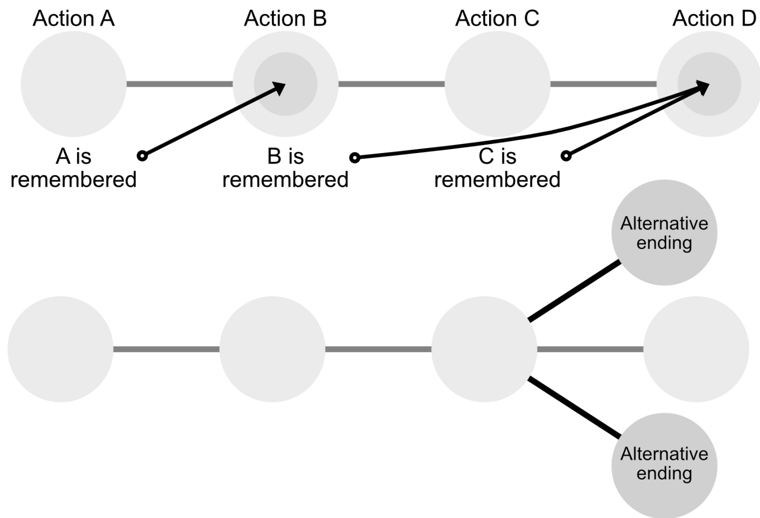


Figure 10, shows how a game system may adapt to a player's decision.

and actions that changed the narrative. All characters have access to this reference list, and it is the game intention to introduce emergent narratives and relations this way. For instance, an opponent can be persuaded into becoming a companion or spy. Both may later, help, get caught or betray the player. The narrative adaptation will be mentioned by encountered bosses, in a fashion that is in contrast to the player.

The narrative explosion is prevented by conserving the narrative components that are predicted to be important to the player. This is determined on the interactions the player makes. Bosses that are considered to be important to the story may cheat death, thus reappear later in the game. Some bosses may randomly get assigned tasks. The storyworld becomes a system that simulates a multiplayer game, where different people are playing together or against each other. These things helps the story to evolve unpredictably extending itself beyond the authored narrative and thus become individualised and enhance the experience.

2.3 Steering a procedural-adaptive storyworld

There is an immediate lack of research that tries to justify which elements in the storyworld that should be adaptive, and in what direction. For instance, (Carvalho 2017) propose a misleading system for emergent storytelling to improve player experience. Taken inspiration from this work, it then becomes naive to rely on an adaptive storytelling that steers experiences in a singular direction. The aim for the following section is to investigate a participatory adaptation desire. From this investigation it is indicated that the desirable adaptation in a storyworld varies.

A player model can be generated from data-driven approach to depict how a storyworld should adapt towards player intentions, behaviours and play styles in a procedural-adaptive storyworld. The research question to make this player model has been *how can a PCG-adaptive storyworld be generated such that it enhances the player experience?* A Likert scale in an explanatory and descriptive research approach (Bjørner 2015) have therefore been used to provide an overview for potential players opinions. The questionnaires will be structured around a multidimensional approach to attain a highly detailed representation of the player model. This case study will account for questions that regards a functional and methodological pragmatism.

Three questionnaires have therefore been generated. The aim for these questionnaires have been to acquire knowledge about peoples' storyworld adaptation preferences. For the first questionnaire, the hypothesis is stated as *There is a difference in how people wants an adaptive behaviour to function.* The question acknowledges that there is a need for a more elaborate adaptive paradigm in storyworlds. The second states that, *(None vocal) musical compositions can be categorised into different emotions* and thus constitute to a data-driven approach. The hypothesis for the last questionnaire states that *adaptation is a mixture of direct and indirect adaptations.* The intention is also to indicate that narratives occurrences may tend to realism, as similarly the work in (Shoulson 2013).

2.3.1 Methodology and indications

A total of twenty participants have been asked to participate in these questionnaires. The questionnaire presents questions that aim to map player's experience, understanding and expectancy patterns of an interactive real time adaptive behaviour. The interactive adaptive real time story telling is defined and exemplified to disclosure a potential knowledge gap. The results and findings for each questionnaire will be presented in this section, but the reader is directed to the digital appendix for the full datasets. The first questionnaire presents problems in a Likert's scale fashion, such that a player model can be generated from qualitative data. The data aims to evaluate and understand what an adaptive story telling is expected to be and how to steering the storyworld. It does this by stating problems that situates normal occurrences in games, such as weather, exploration and characters and which affection these may have in certain situations. It additionally, uses qualitative questions to witness whether the respondents knows what an adaptive behaviour is and when told, their ideologies towards this. Four respondents have answered this questionnaire, and the results are presented in figure 11.

The second questionnaire aims to determine how a set of sixteen pseudo-randomly chosen songs, can be categorised and afford different emotional situations. The questionnaire is structured around a Likert scale and the songs can be categorised in regards to five affiliated emotions such that it can be quantified for an adaptive storyworld. Each of the 16 songs have by the author been categorised to afford either tension, happiness, speed, exploration and action on a scoring system ranging from highly disagree, to highly agree (-2 to 2), with a neutral centre. A potential loss in musical expressiveness can be expected from this approach, although as a proof of concept this may be adequate.

A standard deviation from the mean set value is presented and plotted in figure 12. Nevertheless, six respondents have contributed to the categorisation of these songs; and each song in the questionnaire have been presented to prevent the participants to experience acoustic fatigue, letting two songs that are considered to have high tension not to appear in succession of one another. The quantitative data, showcases that the considered songs for an adaptive behaviour is flexible, as they collectively contribute to a wide emotional range across the five emotional categorisations. This gives the storyworld a range of emotions and situations that can reflect the player experience. The upper graph shows the averaged sum of each respective song, whereas the lower shows the mean function

Figure 11 shows how a game system may adapt to a player's decision.

across each song, and their distributions.

The third questionnaire aims to investigate adaptation in a storyworld that is dynamic. An image was drawn and states four rooms in alphanumeric order. The respondents had to answer which of these rooms they would choose to walk into, and what element that could potentially change this decision. The questionnaire sought to understand what change to a narrative that would result in a different room choice. Seven participants answered this questionnaire and the qualitative and quantitative data seem to point in the direction of film theory (Bordwell et al. 2017). The reveal and suspense and theme of the game were mainly pointed out.

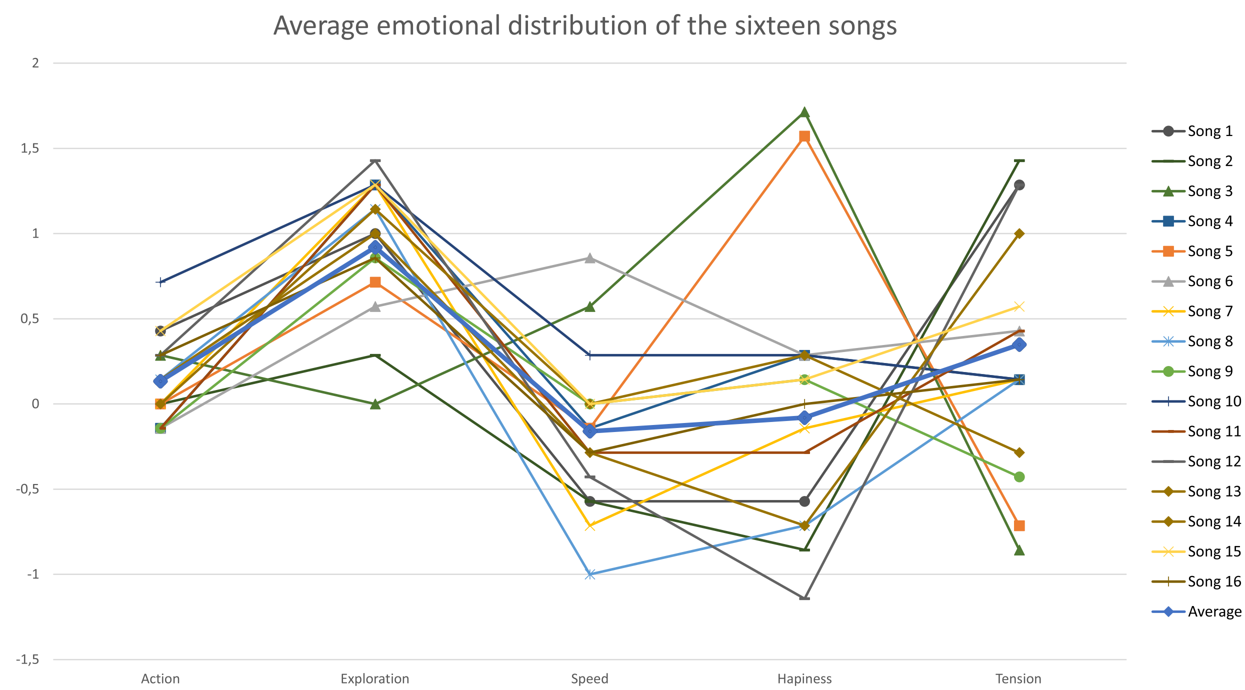


Figure 12 shows the distribution of songs, categorised by five respondents.

2.4 Conclusion

A case study has sought to explore how digital games uses PCG and adaptation in the story world to enhance the player experience. A total of seven games have been investigated in regards to how they respectively enhances the player experience. The tables below summarise how the procedural and adaptive oriented games compares including how they refer to the player experience. The exploratory case study indicates that an experience can be enhanced when the procedural and adaptive implementations correlates to play styles and the game premise. Furthermore, it seems that the PCG coverage is not as important as this correlation since the experience is only degraded when the play style and game premise does not align with the player.

A data-driven approach has been used to acquire participant's adaptation desire. It is indicated that there is a variety for which elements in a storyworld that is preferred for adaptation including the strength. This contrasts the adaptive representation problem addressed in the analysis. However, this contrast can be explained in that a desire may be different from exposure. The quantitative data for the adaptation desire can be used to steer a procedural-adaptive storyworld such that it aligns with the player. Sixteen songs have been categorised in terms of their emotional distribution. A wide emotional range can be extrapolated from the quantitative questionnaire as shown in figure 12. This allows a procedural-adaptive game system to rely on several emotions that can express a range of player styles, behaviours and interactions.

Game name	**No Man's Sky**	**Spelunky**	**The Binding of Isaac**	**Borderlands**
PCG coverage	Fully procedural storyworld	Procedural and authored storyworld	Procedural and authored storyworld	Procedural elements
Use case	To generate storyworlds	To generate a vast variety of environments	To generate a vast variety of environments	To generate a vast variety of weapons
Experience	Weakened	Enhanced	Enhanced	Enhanced
Player style and premise	Not clear	Clear	Clear	Clear

Game name	**Left For Dead**	**The Walking Dead**	**Shadow of Mordor**
Adaptive coverage	Storyworld and dramatic pacing	Authored narrative	Procedural and semi authored
Procedural coverage	Storyworld and situations	None	NPCs and narrative events
Use case	procedural-adaptive structured unpredictibility and player performance evaluation	Personalisation through player behaviour and interactions	Player interactions and the combinatorial explosion
Experience	Enhanced	Mixed	Enhanced
Player style and premise	Clear	Mismatch	Clear

3 Design and Implementation

This chapter aims to provide a solution to the problem statement that states *How can a PCG-adaptive storyworld be generated such that it enhances the player experience?*. It will therefore be parted into a design and the implementation section. The design section will address the design requirements, the development for the prototyping, the initial ideas and the implications during and prior to the implementation. The design will present how the different parts of the implementation enhances the experience. The framework that constitutes the procedural-adaptive storyworld will be presented, and summarised at the end of the design section. The implementation section will address the groundwork necessary to generate a similar experience. It will do this in a comprehensive, step-wise manner. There will be given an attempt to prevent the narrative paradox, combinatorial explosion and narrative loss. It will show how the storyworld can become interconnected and dependent on the player style, intentions and behaviours.

The data-driven approach sets the initial state for the storyworld, and the model-driven approach ensures that the storyworld experience will eventually align itself to the player. Although a vector structure has been necessary to conform a interdependent between the PCG the adaptive storyworld and the player. The personalisation model is build on the basis that the player interactions relate to play styles, behaviours and intentions. The storyworld and NPCs presentation and generation works under the assumption that player can be quantified and used in the PCG algorithm that defines these two.

*

3.1 Design requirements

The following section will put the design requirements listed in table 2 into practice.

Rely on PCG	real time and interactive
Player dependent storyworld	Open world
Solution to the combinatorial paradox	Solution to the narrative paradox
Play styles and behaviours have to steer the storyworld	No narrative loss
Data uniformity	

Table 2 design features

3.2 Design

The following section will introduce to the design considerations that have been put into game. In this regards three documents have been created, a game and level design, graphics and technical design. The design is mainly driven by what is commonly experienced in a digital game. For the level design this is buildings, vegetation, terrain and effects. For the game design NPCs and weapon. The graphics document illustrates the graphics. The technical design will address the initial ideas for how the procedural-adaptive storyworld can be generated.

3.2.1 Level design and game design

The aim for the level and game design is to have the player feel that the storyworld adapts to accordingly to their actions. This is to achieve agency. The player should feel freedom, in the storyworld. This implies that a passive behaviour carries over to other levels.

3.2.2 The ideal viable product

The player should be able to understand that their actions result in an effect on the virtual storyworld. It is not to say that the same action should result in the same behaviour, such as killing a robot, should not result in the robot in the next loop to flee every time, a weighted variable is therefore necessary to make sure that the robot may attack the player instead, so adapt in a different way. But how do we convey this?

3.2.3 The minimal viable product

3.2.4 Computer graphic design

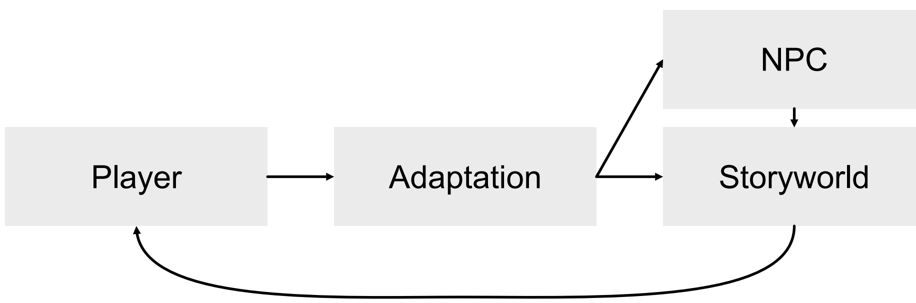
The graphical art style should be very minimalistic with few textures and shaders, similarly to The Falconeer and Superhot, see also figure 13. It should be atmospheric, yet clean such that adaptation in the storyworld can be acknowledged. The graphics should be procedurally generated such that it can be changed. The design should be driven by the player style and interactions.

Figure 13 shows the simplistic in Super hot (left) and atmospheric storyworld in The Falconeer (right).

3.2.5 Technical design

Initial procedural-adaptive framework

The following illustration shows the initial framework idea for the procedural-adaptive framework the storyworld.



14 shows the initial idea for the framework for the implementation.

Architecture

An object-oriented programming paradigm and scriptable objects are used to structure the code within the game. Unity's multi-scene editing will be utilised for the management of the different scenes in the game. The approach allows the project to have better performance, less build size and thus time. The multi-scene editing approach will be depended on distance for the open world and non-interior environments whereas for indoor and interior environments will depend on a trigger approach. In open world environments, fog or similar may hide the fact that this technique is used. Participatory accessibility can be achieved by having the narrative written down in two different documents, that is English and Danish. A switch can choose between the two languages.

Adaptive real time storytelling narrative

The adaptive system will automatically switch between four states depending on the player's location in the world space. Inside, outside (confined), outside (open) and seeking. The player may ultimately be idle in a certain room (inside a flat), location (outside), which may result in the game to have the narrative seek the player instead. Such events may utilise similar approach as seen in virtual reality films, where the viewer is guided by events that happens non- and diegetically. Narrative elements may also support this, such as seen with schizophrenia, where the player may start to hallucinate if within the same room too long. Within a confined space such as an apartment, story plots can be called and placed within the environment. This is possible because the game know where the player is at all time, by introducing gateway colliders at each door entrance as shown below. An open outside environment is where the player has free view of the scenery, perhaps in barren settings where vision is not necessarily obstructed. Here we would be interested to have events to be located inside the environment outside the players vision, and more importantly, at a certain distance ahead of them. However, if the player backtracks or traverses onto previous visited paths. The system should either A remember the previous location the narrative object was located or B find a new location by the visualisation should be different. A confined outside space, can be considered as being outside in a city, in a subway or likewise geometry that obscures the players view in a systematic manner. If no possible entrances are chosen the system should allow the seeking behaviour, allowing or likewise to start act as narrative agents.

Player data elicitation

The player modelling should be dependent on a data-driven and model-driven approach. The data for the data-driven has already been gathered in the case study. However, for the model-driven approach will rely on the player during the game play. When the player has finished the game, the command application.OpenURL can direct the player to the quantiative evaluation questionnaire. The aim for this questionnaire is to capture the player experience. It will do this with a Likert's scale questionnaire structure.

Destructible

Objects should destructive to reflect the player behaviour in the storyworld. These objects will be made in maya with the shatter component. A rigid body will account for the physics when the object breaks. The complexity for this will depend on how many items can be instantiated without effecting performance too much. The destructible objects can also be generated with voxels.

PCG

The PCG will be parted in two sections storyworld generation and NPCs. The storyworld regards vegetation both trees and grass, both will follow a L-system. All vegetation will be simplistically presented with a line render to accommodate for the graphics document. NPC will be generated procedurally, with a low fidelity appearance. The generated NPC will take the form of spiders, robots and humanoids. The player should start the storyworld based on the data-driven approach, and the storyworld should then align itself with the player style and behaviour. A 2D image should represent the storyworld.

The player may control the PCG with a map, which is a very simplified version of the storyworld which can be manipulated as they want. This storyworld should be procedurally generated as the player initially draws it. After each player action, the storyworld should in real time align itself with the play style and behaviour. The play style and interactions will be presented graphically, by the storyworld and NPCs. A destructive behaviour will result in the environment to take a hostile appearance, such as very sharp bulges, destroyed buildings and rotten vegetation. Non-destructive behaviour will result in a beautiful storyworld, such as grown trees, the environment has smooth curves. The PCG paradigm should account for this change.

Since the storyworld is generated based on the player, the animations should for a NPC should be able to adapt the these differences in the storyworld. NPC should be generated such that they reflect the player style and behaviours. Inverse kinematics can be used to acquire believable animations.

3.3 Implementation

An interactive procedural-adaptive storyworld has been developed based on the design requirements. This implementation will conjugate a player model together with a procedural-adaptive storyworld as an approach to achieve a potential enhanced experience. The following sections will thus introduce to one potential methodology that makes up an procedural-adaptive storyworld in a game where the player's experience is directed based on player styles, interactions and behaviours. The procedural-adaptive framework will be presented at the end of this section and the different components for this framework will be shown throughout this section. A solution has been proposed for the narrative paradox and combinatorial explosion by allowing the two to be dependent on the player. The player may dictate how much freedom and narrative growth is preferred based on the play style, behaviour and interactions.

3.3.1 Player modelling

A player model has been generated on a data-driven and model-driven approach. The data for the data-driven has been based on the data elicitation from the case study, and grounds the storyworld, the narrative and storytelling. Remember, that the case study have provided two data sets for the player modelling. That is the emotional correlation for sixteen songs and procedural-adaptive steering. To find the emotional weight of each respective songs, the averaged sum has been taken. Although, this causes a problem in that each weight value becomes non binary and thus difficult for the storyworld to account for. For instance the weighted sums for song 1, 5 and 15 are almost indistinguishable. A solution to this problem has been to convert each weight into an binary representation with a factor of two, such that the data becomes homogeneous with the data in the storyworld. This makes the procedural-adaptive framework easier to control. The conversion have revealed no ambiguity for the conversion method. Notice that negative values have been clamped to zero. The table below shows the first six song's and their representative data structures, in weighted sums (left) and vector bytes (right).

$$
\left(\begin{array}{ccccc}
Actions & Exploration & Speed & Happiness & Tension \\
\hline
0.4285714 & 1 & -0.5714285 & -0.57142 & 1.2857142 \\
0,2857142 & 0,8571428 & -0,2857142 & 0 & 0,1428571 \\
0 & 0,2857142 & -0,5714285 & -0,8571428 & 1,4285714 \\
0,4285714 & 1,2857142 & 0 & 0,1428571 & 0,5714285 \\
0,2857142 & 0 & 0,5714285 & 1,7142857 & -0,8571428 \\
0 & 1,1428571 & 0 & 0,28571 & -0,2857142 \\
\vdots & \vdots & \vdots & \vdots & \vdots
\end{array}\right)
\Longleftrightarrow
\left(\begin{array}{cccccc}
A & E & S & H & T & ID \\
\hline
0 & 2 & 1 & 1 & 2 & 1 \\
2 & 1 & 1 & 2 & 0 & 2 \\
1 & 1 & 2 & 0 & 1 & 3 \\
1 & 2 & 0 & 1 & 0 & 4 \\
2 & 0 & 1 & 0 & 0 & 5 \\
0 & 1 & 0 & 0 & 0 & 6 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots
\end{array}\right)
$$

After the song conversion, each song vector attain an additional byte sixth byte. The reason for this is that this sixth byte functions as an identification pointer. The pointer is directed to the song position in the assets folder, and is referenced when needed to be played in the storyworld. The emotional weights work as comparing values for the experience. However, when data about the player style, intention and behaviour is unknown a comparison becomes difficult to achieve. Therefore The most neutral emotional ranked song will be played when no data is available.

For the procedural-adaptive steering three adaptation problems where slightly disliked with a value of -1. These adaptation problems have been accounted for in different ways. For 'The game should adapt subtly' and 'adaptation caused by interactions should happen subtly' have resolved in a combination of invisible and visible adaptations (Zhu and Ontañón 2020; Spronck et al. 2006). For instance, the terrain may be an impactful way to visualise adaptation, but players destructive actions are not visibly indicated. For 'play styles that I do not like should reflect my preferred play style' has been removed from the implementation. Instead it should be taught as providing experiences that rely on the predicted experience. The remain adaptation problems have been considered for this implementation. Notice, that it has not been possible to treat the procedural-adaptive steering in similar terms as the song categorisation due to the adaptative representation problem.

The data-driven approach for the storyworld adaptation steers the initial generation for the storyworld when there is no captured data during game play to determine the player experience. It thus rely on an initial general player description where it during game play align itself to the player with the model-driven approach. The data for the model-driven approach is done directly through the interactions and behaviours of the player. Although, to prevent confusion and biased player modelling, (K. Smith 2013; K. Smith 2017) present a temporal and average behaviour calculus. It consists of parameters that each correlate directly to the actions that the player can perform whilst playing. These actions are mapped in figure 15. In preserving space for the player to be unpredictable, without changing the game experience drastically, inspiration has been taken from the drama pacing in (Booth 2009) for each action. This player modelling allows the player experience to be highly individualised and at the same time have initial presumptions about the average player. It allows the storyworld to conform itself towards specific player types, behaviours and intentions. Both the data-driven and model-driven approaches have been vectorised. Vectorisation is an proficient and cost efficient method to achieve a bi-directional data feed system.

3.3.2 The player experience

In a storyworld, the player is an interactive and unpredictable character. Experiences may be alternated based on the storyworld and the storyworld may change itself to accommodate to the player experience. Without the player, the game world becomes an empty container, with little meaning, because it is the perceived agency that gives the storyworld meaning. For this implementation the perceived agency can be categorised to different player types (Bartle 1996), following the SAP. These are the Explorer as 'cautious', Killer with 'Attacking and destruction' and Socialiser through being 'passive'. These interactions changes how the storyworld will predict the experience for the player modelling. The interactions changes the storyworld and how it predicts an experience in different ways. The storyworld will adapt accordingly towards these four and the principle is demonstrated in figure 15.

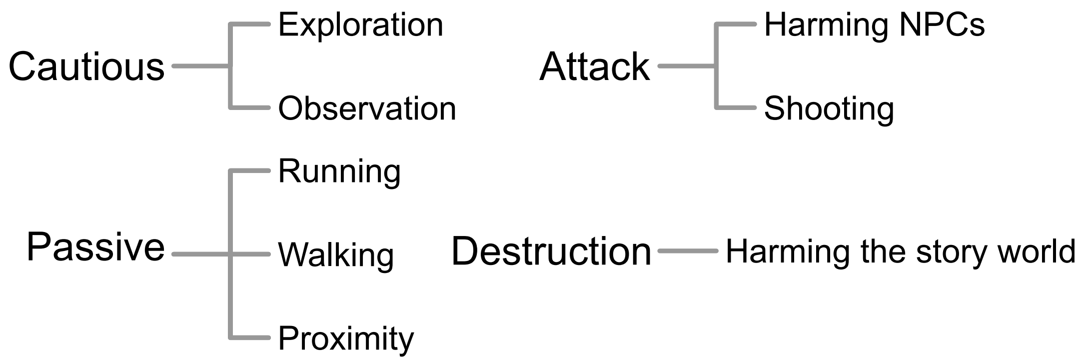


Figure 15 illustrates the different interactions the player can do.

These four interactions are stored as vectors, which value ranges from zero to two, where zero is a depleted interest for actions where two is considered as a high interest for actions. To prevent the exploration vector from growing indefinitely, a temporal data flush has been implemented. This happens when the vector exceeds 100 instances. To prevent data loss and fluctuations when the data is flushed the last ten indices are copied over to a new exploration vector.

3.3.3 Storyworld representation and generation

The player experience and model are depicted into the representation and generation for the storyworld. The player has been linked to the generation and representation for the storyworld. Figure 16 shows how the player actions are linked to the storyworld. The storyworld is generated in real time, although an initial representation of the storyworld is presented. When the player feed data into the storyworld through interactions, see figure 16, the storyworld will align itself towards these actions. Notice that a NPC is absent in figure 16, although being a part of the storyworld, this is because a NPC is considered as an individualised and intelligent entity to the storyworld, and therefore requires its own relationship framework to the player. This relationship will be addressed in section 3.3.9.

Figure 16 shows the relationship between the player and the storyworld

Code snippet 3.10 to determine which objects inside the camera frustum and thus manipulative.

```
foreach (GameObject isVisible in thisChildIsVisible){
   yPos = getHeightPosVisible((int) isVisible.transform.position.x, (int) isVisible.transform.position.
       z);
   Vector3 xsPos = isVisible.gameObject.GetComponent<LineRenderer>().GetPosition(0);
   Vector3 ysPos = isVisible.gameObject.GetComponent<LineRenderer>().GetPosition(1);

   isVisible.gameObject.GetComponent<LineRenderer>().SetPosition(0,
   new Vector3(isVisible.transform.position.x,
   Mathf.SmoothStep(xsPos.y, yPos, smoothTransition),
   isVisible.transform.position.z));

   isVisible.gameObject.GetComponent<LineRenderer>().SetPosition(1,
   new Vector3(isVisible.transform.position.x,
   Mathf.SmoothStep(ysPos.y, yPos + 1, smoothTransition),
   isVisible.transform.position.z));}
```

3.3.4 Procedural-adaptive environment

The environment in the storyworld is generated with Perlin noise series. This achieves a more compelling result in that the environment becomes smooth (McClellan et al. 2017). Three Perlin noise filters have been used to generated a smooth environment. Since Perlin noise can be thought as octaves (Moeslund 2012), a real time adaptive behaviour becomes possible to implement. A visual representation of this is presented in figure 17. Each respective octave has two parameters, these are, frequency and amplitude. Although, an alternative is to use kernels as the environment is represented as a two dimensional image (Moeslund 2012). The player may change these two parameter depending on their play style. The frequency is alternated by the player cautiousness and passiveness, whereas the amplitude is dictated by attacking and destruction. The frequency is thus used to emphasise on the Explorer experience whereas the amplitude for the Killer.

Code snippet 3.3 shows how the song selection algorithm.

```
void createGeometry(){
vertices = new Vector3[(xSize + 1) * (zSize + 1)];
   for (int j = 0; j <= zSize; j++){
      for (int k = 0; k <= xSize; k++){
         float y = Mathf.PerlinNoise(j * .4f, k * .4f) * 1.5f;
         y += Mathf.PerlinNoise(j * .2f, k * .2f) * 2f;
         y *= Mathf.PerlinNoise(j * .1f, k * .1f) * 2;
         y *= Mathf.PerlinNoise(j * .05f, k * .05f) * 2f;}}
```

A continuous smooth transition that aligns itself with the player experience has been achieved by storing the current environment, making a future prediction of the player experience, and then change the current environment in small increments towards this future prediction.

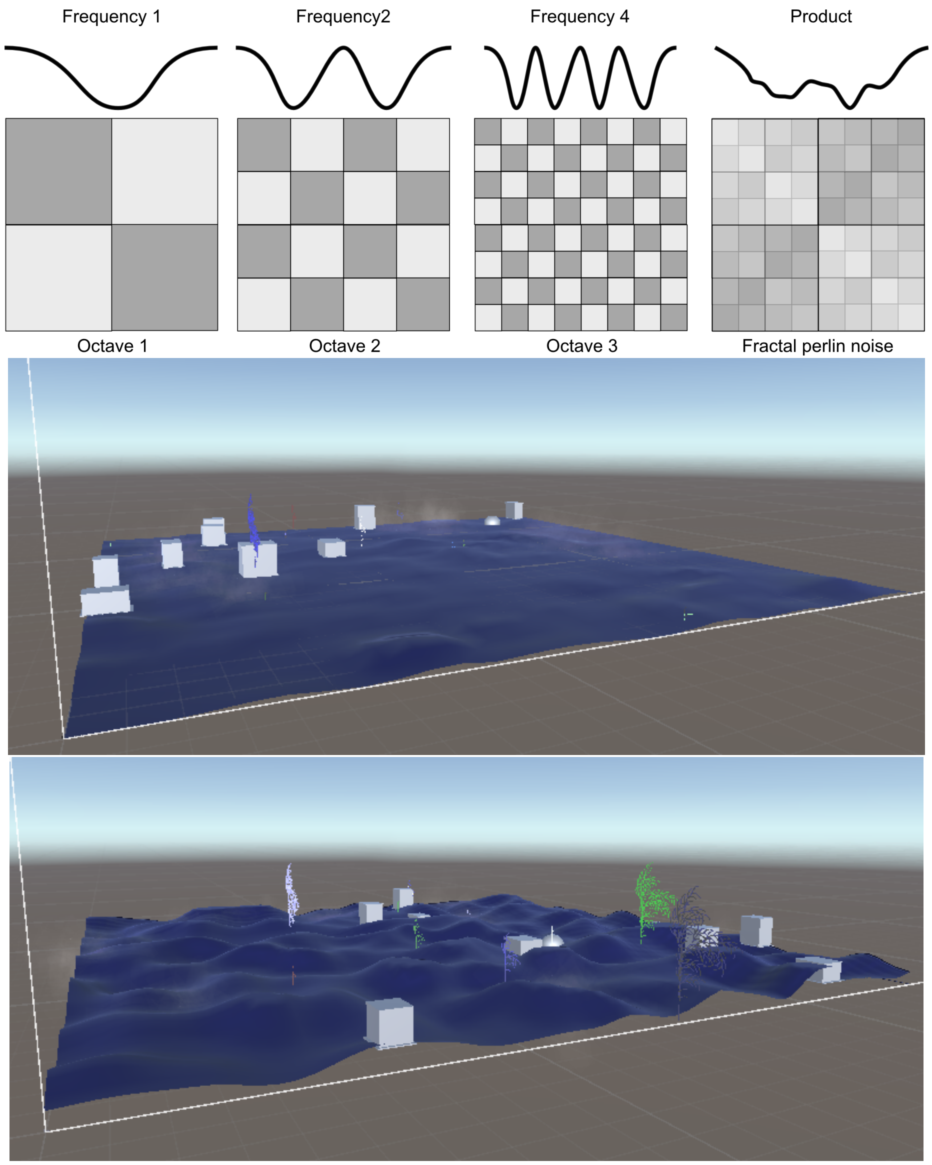


Figure 17 shows how smooth geometry can be achieved from Perlin noise (top image), and the result in game (lower images).

Code snippet 3.3 shows how the song selection algorithm.

```
private void setTerrain(){
  makeProceduralTerrain.clone = makeProceduralTerrain.cloneTerrain();
  makeProceduralTerrain.distDiff = makeProceduralTerrain.buildNextTerrain();
  makeProceduralTerrain.convolution = makeProceduralTerrain.adaptionTerrain();}

public IEnumerator callSmoothTransition(){
  for (byte i = 0; i <= makeProceduralTerrain.width - 1; i++){
    for (byte k = 0; k <= makeProceduralTerrain.height - 1; k++){
      makeProceduralTerrain.convolution[i, k] = Mathf.SmoothStep(makeProceduralTerrain.clone[i, k],
          makeProceduralTerrain.distDiff[i, k], makeProceduralTerrain.flow);}}
```

Since it is possible for the environment to transition between two experience states in a smooth manner, the storyworld has to follow this dynamic change as well. However, this process has become very demanding and a solution has been developed based on inspiration from (Yannakakis and Togelius 2011) where fast is preferred over responsiveness. This approach depends on a frustum check where all objects that are visible will be adapted towards the new experience smoothly, whereas invisible objects are adapted when the initial adaptation has happened or become visible. To determine which objects are visible and invisible a list of these objects are necessary. For this implementation the storyworld will align itself to the player experience when the player progresses in the narrative.

3.3.5 Procedural-adaptive population

Vegetation

Vegetation refers to grass and trees in the storyworld. The grass is generated with respect to the terrain. This means that since the environment is dynamic the grass becomes dynamic as well. For instance, a valley has more grass than on a bulge. Neighbouring distance processing ensures that the placement does not appear uniform as a consequence for the instantiation method. An alternative, Poisson-Disc Sampling method (Wu et al. 2020; Ebeida et al. 2012) has been developed, but has yield to be fragile for local minima and maxima and overall instantiation. Although a solution to this could be to increase the sample points and distance checks, but then the real time application seem questionable.

For the trees a semantic L-system has been used, in particular a bracketed OL-system, however it is acknowledged that various systems exists (Prusinkiewicz and Lindenmayer 2012). The Bracketed OL-system implementation requires some additional explanation; in this regard rule sequencing and branching will be introduced in the following sub sections.

Rule sequence

The following code snippet shows how the rule sequence for the bracketed OL-system is. Notice that the implementation uses a dictionary to store a collection of data in an un-organised manner. The advantage of using a dictionary over a list, is that a potential lookup has a linear complexity, lists is not necessarily linear because they are organised, however their structure is somewhat identical. The dictionary is used to store the actual rule string and key that is used to activate it. The currentCharacter variable creates a list of characters, in case multiple rules were present. Then this variable is presented to the dictionary which checks whether the certain key 'contains-key' has a rule applicable to it, so in the case of the key 'F', it will return FF+[+F+F+F]-[-F+F+F]. When the presented key is not 'F', then it will not be added to the list, but instead act as pointers, meaning that '+' and '-' results in rotations, '[' and ']' results in a propagation back to the path root, where a new path will emerge.

Code snippet 3.4 shows the rule set for a procedural tree generation.

```
private Dictionary<char, string> rules = new Dictionary<char, string>();
rules.Add('F', "FF+[+F+F+F]-[-F+F+F]");}

for (int i = 0; i < stringCharacters.Length; i++){
   char currentCharacter = stringCharacters[i];

   if (rules.ContainsKey(currentCharacter))
      newString += rules[currentCharacter];
   else
      newString += currentCharacter.ToString();}

      currentString = newString;
      stringCharacters = currentString.ToCharArray();
```

Branching

The branching algorithm interprets the string rule, and performs actions accordingly to the expression needed by iterating through a switch with each respective characters in the string. The process is simple and straight forward, however, notice the use of the stack, pop and push methods. A stack is to used to temporally copy an object and represents a variable structure of 'last-in-first-out'. The stack contains rotations and positions. This is handy in implementations where information is to be discarded after it has been retrieved, such as in this case of a branch. When a branch has finished the algorithm, it returns what constitutes the branch, and returns to its root by using the 'pop' method. The 'push' method, pushes its argument onto the stack, so the rotation and position of a given part of the branch.

Code snippet 3.9 illustrates how the rule set is interpreted.

```
for (int i = 0; i < stringCharacters.Length; i++){
   char currentCharacter = stringCharacters[i];
   switch (currentCharacter){
      case 'F':
         Vector3 initialPos = transform.position;
         transform.Translate(Vector3.up * length);
         drawLine(initialPos, transform.position);
      break;

      case '+':
         transform.Rotate(Vector3.left * angle);
      break;
      case '-':
         transform.Rotate(Vector3.left * -angle);
      break;

      case '[':
         ti = new transformInfo();
         ti.pos = transform.position;
         ti.rot = transform.rotation;
         transformStack.Push(ti);
```

```
        break;
        case ']':
            ti = transformStack.Pop();
            transform.position = ti.pos;
            transform.rotation = ti.rot;
        break;}
```

Form structure

As also shown in figure 18, several tree types can be made from one simple rule and procedurally change from mutations. One rule to introduce diversity is therefore used. All trees will be represented with Unity's Line Renderer object. The aim is to allow the tree structure to have different appearances that express the player experience. This has been done in regards to scalability and complexity as shown in figure 18. Small and less complex trees relates to the Killer whereas in contrast to the Explorer, complexity and scale is afforded.

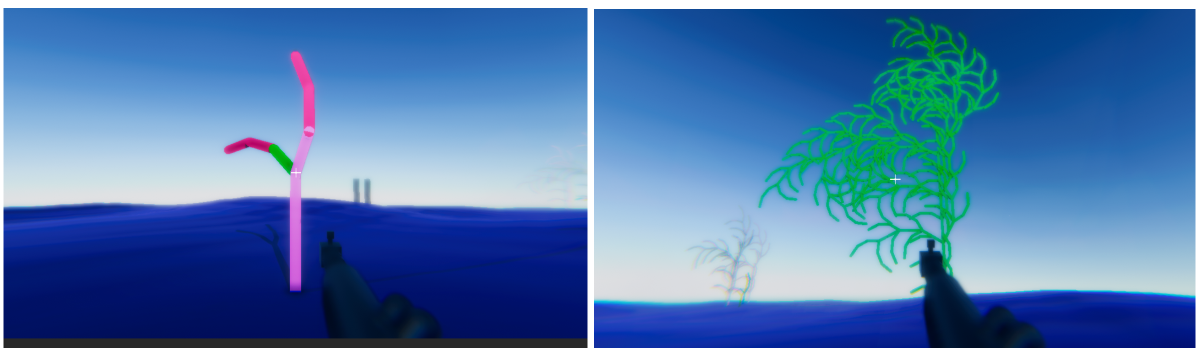


Figure 18 shows potential tree forms based on the code snippets 3.4 and 3.9

3.3.6 Interactive staged area

The problem with a Staged Area has been considered in regards to its integration in the storyworld. Staged Area therefore rely on strict geometric properties to accommodate for its placement in the storyworld. The information required for a staged area is lacking, in that the storyworld itself is not considered properly. Therefore to extent this idea, this implementation propose an Interactive Staged Area (ISA). The aim for the ISA is the capability to adapt to a wide range of geometric properties. To achieve this, inspiration has been taken from procedural filters and semantic feature constraints (Tutenel, Van Der Linden, et al. 2011; Smelik et al. 2011). An ISA allows the narrative container to become a coherent and consistent in the storyworld. Furthermore, it can be tightly integrated as a part of an adaptive and dynamic storyworld by relying on a PCG. For instance, a narrative is not dependent on the storyworld (Dormans 2010), therefore a narrative in a dungeon can have the same meaning in a forest, the ocean and sky. The ISA can thus change to comply with the storyworld.

A negotiation framework (Smelik et al. 2011) can be build such that the ISA and storyworld becomes blended. Two parts have been build to make an ISA, that is an environmental evaluation and 'emergent association'. The environmental evaluation has been done with the standard deviation calculation.

$$sd = \frac{\sqrt{\sum |x_i - \overline{x}|^2}}{n-1}, \quad \{n = detail\} \tag{3.1}$$

The variable x_i is the value a particular point in n and $\overline{x}$ is the mean value of x_i. When *sd* is large, this is due to a notch in the terrain, and some objects will not be considered here. Notice that the resolution is dependent on the entirety of a object cluster, and that the resolution is therefore flexible. Different clusters may therefore be actuated in different places. On the contrary, when *sd* is small, the cluster is predicted to be on even terrain. For the emergent association the ISA may either be blended with the storyworld or it may break into clusters at where this difference is still fairly large, providing a strong emergent storytelling paradigm, and enhanced experience for the Explorer. For instance, parts of a ruin may have sunk down a ledge. The code listing 3.6 shows the implementation for figure 19. For the blending, the resolution of the grid depends on the size of the narrative that is contained. The reason for this is that small structures require more detail to determine whether the object is at a notch or not. However, in Unity, for multiple objects this requires a boundary box operation that contains all the elements of the narrative. Notice that a tree search algorithm could have been used for the same result. The boundary box is calculated with Unity's axis-aligned bounding box (AABB). The centre of the AABB is found by the collective centre of the children's boundary boxes that constitute the narrative. The sum of each respective child boundary box is then grown to fit a singular *encapsulated* boundary. Other alternatives to this approach are relative positioning, ray casting, collision and neighbouring processing.

Code snippet 3.6 for the actuation algorithm resolution is given by gridRes.

```
for (int i = -gridWidth / 2; i < gridWidth / 2; i++){
   for (int k = -gridHeight / 2; k < gridHeight / 2; k++){
      grid = Instantiate(planeSquare, transform.position, Quaternion.identity, transform);
      grid.transform.position = new Vector3(transform.position.x + i * gridRes + gridRes / 2, transform
          .position.y, transform.position.z + k * gridRes + gridRes / 2);

if (calcPlaneStd && isPlayerDistant)
   for (int i = 0; i < collection.Count; i++){
      dist[i] = Vector3.Distance(player.transform.position, collection.ElementAt(i).gameObject.
          transform.position);
      tmp[i] = collection.ElementAt(i).gameObject;}

      for (int i = 0; i < dist.Length; i++){
         if (dist.Min() == dist[i])
            tmpClosest.Add(tmp.ElementAt(i).gameObject);
         else if (dist.Min() != dist[i])
            tmpFar.Add(tmp.ElementAt(i).gameObject);}

collection.ElementAt(i).gameObject.transform.position = getActuatedPos()[i].position;
```

Breaking the narrative container into clusters allows a storyworld to enhance itself in that it can provoke emergent storytelling. The approach may put emphasis on the Explorer, Killer and Socialiser with the appropriate mise-en-scene (Bordwell et al. 2017). These clusters can be generated in three ways. These are, by removing the object of consideration and at the same time add animated bits and pieces to simulate destruction as also demonstrated in the

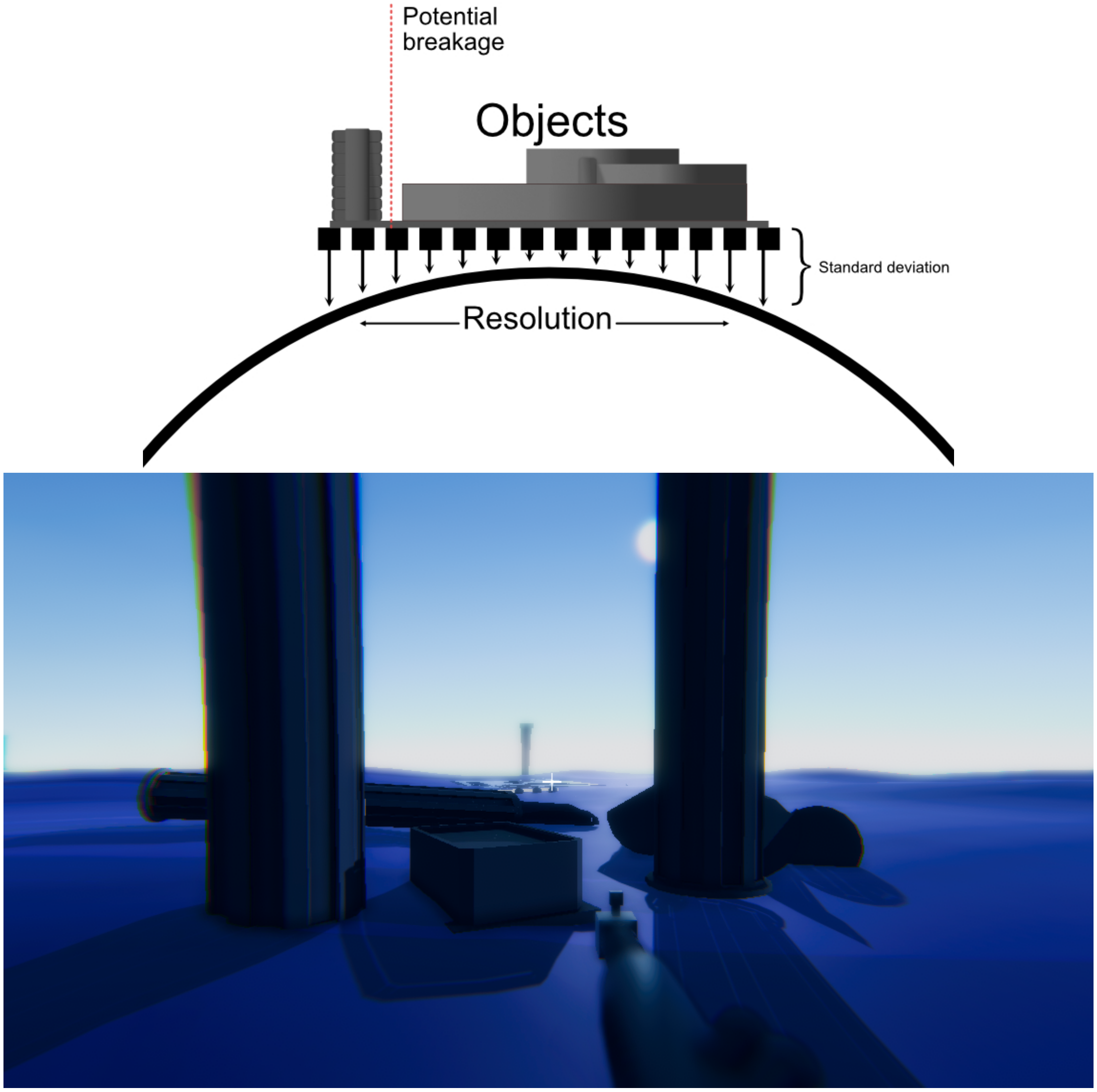


Figure 19 illustrates how the geometric trajectory is considered by the actuation algorithm (top image) and a possible result of the algorithm in game (lower image).

game Red Faction Guerrilla, secondly procedural and realistically, such as in Teardown, see parent(2) in figure 20, and thirdly simulated by having the object already in several pieces such as in Half Life 2, see parent(1) in figure 20.

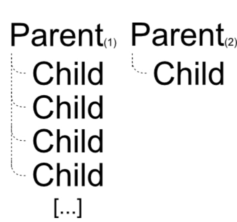


Figure 20, the object container which contains a series of a few to thousands of cut objects (left parent) and an object container that contains a conditioned series of cut objects (right parent).

3.3.7 Procedural colouring

The aim for the procedural colouring is to convey an experience that reflect the player. The colouring in the storyworld becomes vibrant when the player shows interest in the Exploratory play style, where as colours becomes matt for the Killer. The procedural colourisation uses colour theorem to find adjacent colours et cetera (Stone et al. 2008). Since, a monotonous colour scheme may fall flat, colours are therefore generated procedurally, based on the player style. The algorithm requires access to all materials and line renders in the game. An object search in the environment is therefore performed at run time see also code snippet 3.7, since this search may be as demanding as there are potential objects; textures and line renders are respectively stored in an array. The colourisation can then be changed in real time to afford the player experience. To prevent unnecessary search iterations, as to find these objects, unity's *script execution order settings* has been used.

Code snippet 3.7 shows how all objects in the scene are found and sorted.

```
allObjInScene = FindObjectsOfType<GameObject>();

for (int i = 0; i < allObjInScene.Length; i++){
   if (allObjInScene[i].GetComponent<MeshRenderer>() != null)
      allMeshes.Add(allObjInScene[i]);

   if (allObjInScene[i].GetComponent<LineRenderer>() != null){
      if (allObjInScene[i].transform.root.tag != "Player")
         allMeshes.Add(allObjInScene[i].GetComponent<LineRenderer>());}
   }
```

3.3.8 Procedural-adaptive real time weather

The procedural adaptive weather aims to simulate dynamic conditions based on the Killer play style. The weather system consist of thunder, rain, clouds, and will in real time. The killer type results in a cloudy, dark weather. A high interest in the Killer play style results in thunder and rain. The thunder strikes are created with a line-renderer. A particle system has been attached to it. A noise map is used to make the thunder appear dynamic with a root like structure as typically seen when observing thunder strikes. It behaves in the three dimensional space, and may rotate in any given direction, however to make sure that the thunder actually strike toward the ground, the down direction axis, is heavily weighted. The rain is created in the same fashion as the thunder strike system, but with different attributes, and with small transparent textures instead of line renderers.

The clouds are generated as clusters, and the density is determined by the Killer and Explorer play styles. Clouds can be generated in different ways. Half Life 2 uses image processing onto a box that surrounds the storyworld called 'skybox', see also figure 21. A skybox is commonly used to make the storyworld appear larger than it is. For instance, in Half Life 2, cities are built in the sky box and projection mapping allows objects to be less complex to simulate vast environments. Minecraft uses voxels and Crysis uses volumetric clouds and rain. This implementation uses procedurally generated voxels. The clouds are built in a billboard fashion to simulate depth with linear perspective,

Figure 21 shows the skybox in Half Life 2 (upper image) and the procedural used here (lower image).

and constancy scaling (Zanker 2010). Notice that since cloud voxels are demanding there have been given two options for its appearance in the setting's menu; these are complexity (layered clouds) and continuous motion.

3.3.9 Procedural-adaptive NPCs

A NPC is normally considered as an agent that interacts or acts with the player, and thus directly refers to the killer and Socialiser play style. Although, a companion may help afford Exploration and impact the player emotionally. However, as similarly to how the player may be unpredictable and change play styles, intelligent NPC may as well change play styles as in Metro Last Light, where the antagonist is initially the companion but becomes the opponent. In the Witcher series, doing certain quests for NPCs will make the player enemies with other NPCs that would otherwise have given the player quests themselves. In this fashion NPCs may appear to be human like, where choices matter much more to the player. An unpredictable play style for an intelligent agent has been achieved by correlating the Killer and Socialiser with how much the player is predicted to have interest in Killer play style.

Definition 3.1 (Procedural-adaptive NPC) *A character that perceives the storyworld and the player, from which it makes its own decisions.*

For this implementation, a NPC consist of two components, an intelligence behaviour and locomotion. To develop a framework where NPCs are included in a procedural-adaptive storyworld, further investigation has to be put into an intelligent behaviour, locomotion and generation. These will be addressed in the following sub sections.

Intelligent behaviour

To have the player think about their interactions, the NPC has four behaviours which are steered by the player type and behaviour, this relationship is demonstrated in figure 22. These behaviours are fleeing, attacking, regretting and being passive (Reynolds 1999). As a consequence for the player's Killer play style, the NPC may flee when it feels overpowered, due to high damage dealt to it. The NPC may also attack the player, both to emphasise the Killer play style. The regret behaviour refers to a change in behaviour based on a continuous stochastic evaluation. Although, when the player is predicted to be overpowering, such as by dealing a lot damage, the NPC attack behaviour turns into flee. Lastly, a passive behaviour refers to the Socialiser in that the player is predicted to be less aggressive in the storyworld. In this state, the NPC may provide the narrative. Additionally, the representation for the NPC can change accordingly to the behaviour, intentions and play style of the player in the storyworld. For instance, a NPC may grow and become stronger to enhance the experience for the player that is predicted to have interest in the Killer player type.

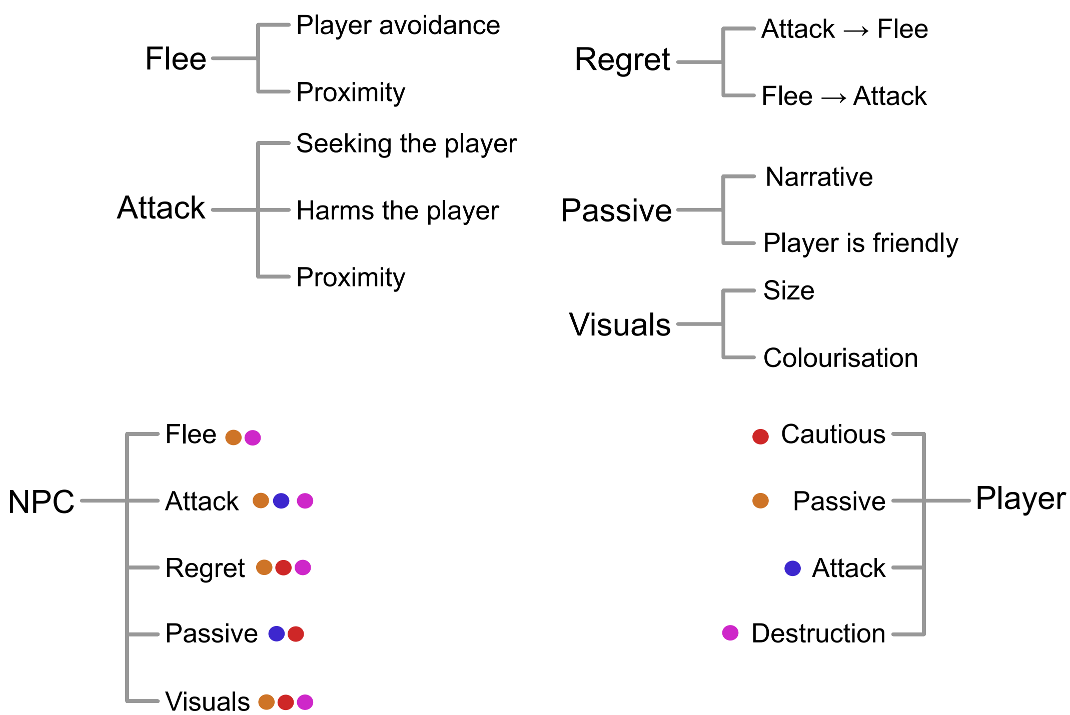


Figure 22 shows which actions the NPC can perform, and how the player behaviour relates to the NPC.

A Markov chain has been used to simulate this intelligent behaviour for the NPC. This allow the NPC to take an observation input, and have it dynamically change based on the player's play style as indicated by the coloured dots in figure 22. Although an alternative is a real time Markov Reward process that uses reinforcement learning

(Ramstedt and Pal 2019). However, this temporal approach depends highly on the storyworld complexity because it has to evaluate a texture map and determine navigation on this basis. In contrast, a procedural NPC does not have to build an environmental model for its navigation.

Locomotion

The state of the art for character's locomotion uses data driven approaches, such as animated clips, motion matching, and Phase functional Neural Network (PFNN). For animated clips, the character follows an animation state machine (Zadziuk 2016). Each animation clip may have different loops, nodes, rules, blend times, loop points and responsiveness can therefore become a problem. Actions can simply predetermined by an animator, which makes it difficult to correlate to an adaptive storyworld (Holden, Saito, et al. 2016). Motion matching uses animation clips to match a motion of a character. It is a locomotion synthesis that is designed to predict future motions from probability and real time applications. An disadvantage is that animations tend to drift, because animations are blended and excluded. The zombies in Left For Dead uses this technique. A procedural per frame PFNN as an extension has also been considered to guide motion and interactions (Holden, Komura, et al. 2017). The PFNN can change the weights of an AI model with respect to the pace of the player. A procedural motion generation (PMC) allows agents to adapt dynamically to the storyworld. Some PAC technologies uses motion matching based on motion capture data sets, but these lack precision in situations where the motion cannot be matched to the interaction. Raycasting can help the PMC to predict a future motion, allowing motions to be blended naturally. Figure 23 shows a character that walks on jagged terrain, but the motion considers the terrain. However, normally motion matching rely on situation that afford a wide range of nuance in the storyworld, such as specific slopes, heights, steps, corridors (Booth 2009). Animated clips are therefore impractical for interactive adaptive storyworlds.

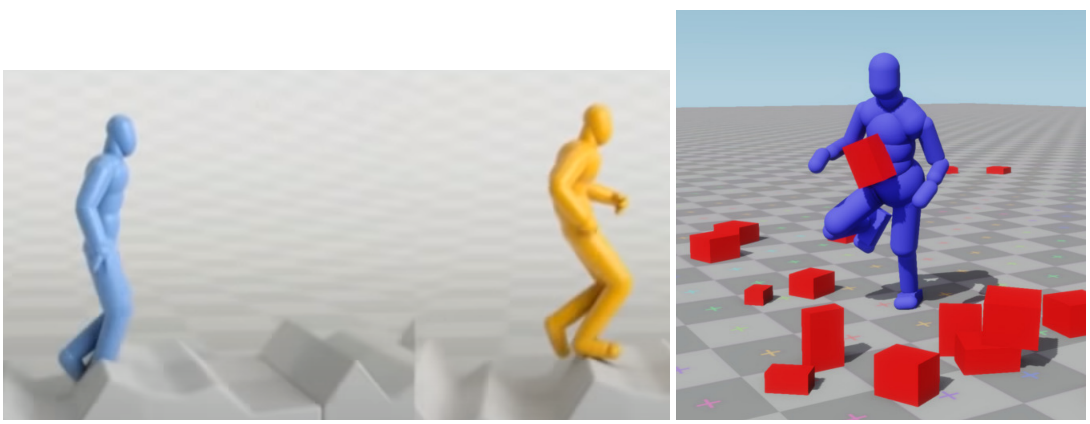


Figure 23 procedural animation on rough terrain (right) and manual animations (left).

Other animation dependent approaches are a per frame A.I algorithm called phase functioned neural network which allow agents to use procedural motions to match situations, by changing the network's weights with respect to character pace (Holden, Komura, et al. 2017). However, this approach is incapable of handling rough terrain. A PCG can be used to achieve physically simulated characters (Bergamin et al. 2019) and the solution is a direct take on the future works of (Holden, Komura, et al. 2017). Inverse kinematic has been used to generate realistic animations, compatibility for dynamic environments and thus adaptive storyworld (Bereznyak 2016).

Procedural generative motions and inverse kinematics can be considered as a potential solutions to achieve animations that can comply to an adaptive storyworld. A linear function to time defines the procedural generative motion as inspired from (Holden, Saito, et al. 2016; Hastings et al. 2009). This linear function is necessary for the NPC locomotion as locomotion is independent on animation clips. Since the storyworld is generated with Perlin noise functions, and thus numerically representative, it is possible for the NPC to directly interpret the storyworld. This technique requires environmental omniscience, for instance as illustrated in figure 3 when the NPC is at the coordinates (1, 1), it can make an environmental lookup, the corresponding height value may then be 1. The NPC then know that the its positioning has to change accordingly, which makes an adaptive storyworld possible.

(X, Y)		Z
(0, 1)	$\longrightarrow$	0
(1, 1)	$\longrightarrow$	1
(2, 2)	$\longrightarrow$	1.5

Table 3 represents a lookup table for a terrain where a given (x, y) coordinate correlates to a specific Z value

Code snippet 3.9 shows the core logic for the procedural character generation.

```
yPosLT = terrain.GetHeight((int)trackerLT.transform.position.x, (int)trackerLT.transform.position.z);
yPosRT = terrain.GetHeight((int)trackerRT.transform.position.x, (int)trackerRT.transform.position.z);
```

Inverse kinematics

The locomotion is a linear function to time, however constraints and extensions can be applied to this principle as illustrated in (Holden, Saito, et al. 2016; Hastings et al. 2009). The size of the NPC depicts the speed for the linear function. A sinusoidal temporal function has been used to simulate breath motion, such that the NPC appears neutral. Inverse kinematics can helps generate realistic movements as it simulate the movement through joints and thereby prevent the use for animation clips. The locomotion system predicts where the end node of the hierarchical system should be in the environment. This node is also referred to as the grandchild in figure 24.

To enhance the player experience through their interactions a simple dismemberment system has been build on the inverse kinematic structure. A collider is attached to each joint and detects where the NPC have been hit. Hit points are allocated to each joint, the prevent accidental dismemberment, where weak joints correlates to its depth in a hierarchy with joints. A three search is used to find which joints that exist on a lower depth should be detached from the NPC, see also figure 24. This implies that when the player attack the agent at these disembodiment locations as shown in figure 24, it will cause the NPC to loose this part of their body in the present.

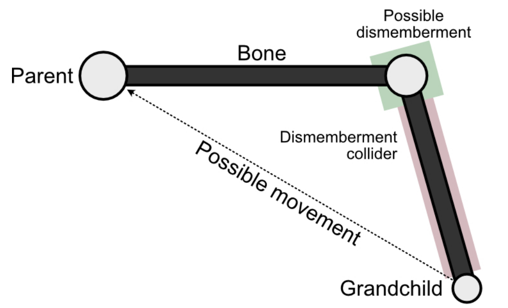


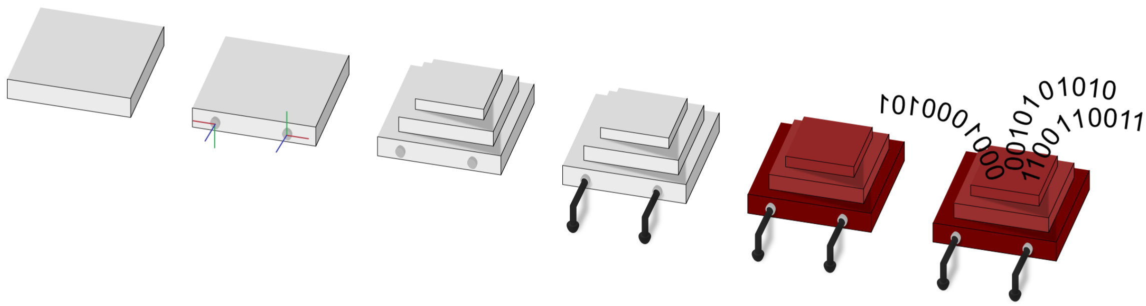
Figure 24 visually represents the decapitation model, including the character's leg structure.

NPC generation

A PCG has been used to generate the NPCs in the storyworld and is based on the play style up to that point. The generative procedure has taken inspiration from The Binding of Isaac and Spelunky and is generated accordingly to the illustration in figure 25. In a chronological order, this is, the body, back, head, legs, player relationship, and then behaviour and locomotion. Notice that each relationship state is based on the player style up to that point, however, it may change prior to a potential player confrontation.

Code snippet 3.9 shows the core logic for the procedural character generation.

```
private void makeBody(){
   for (int i = 0; i < parts; i++){
      buildComponentBody = agentConstruct.runAlogrithm(type, GameObject.CreatePrimitive(PrimitiveType.
          Cube));
      buildComponentBody.gameObject.AddComponent<Rigidbody>();
      buildComponentBody.gameObject.GetComponent<Rigidbody>().isKinematic = true;

      buildComponentFace = agentConstruct.runAlogrithm(type, GameObject.CreatePrimitive(PrimitiveType.
          Sphere));
      buildComponentFace.gameObject.AddComponent<Rigidbody>();
      buildComponentFace.gameObject.GetComponent<Rigidbody>().isKinematic = true;

private static void placeHead(GameObject head){
      head.transform.position = Vector3.zero;

      GameObject core = GameObject.FindGameObjectWithTag("AgentCore");
      head.transform.position += new Vector3(core.transform.position.x, core.transform.position.y, core
          .transform.localScale.z / 2 + core.transform.position.z);
      head.transform.localScale = Vector3.one * 1.5f * upScale();

private static void proceduralSpider(GameObject obj){
   if (tmpScale.x > 0.5f && tmpScale.y > 0.5f && tmpScale.z > 0.5f){
      scale = tmpScale * Random.Range(0.5f, 0.9f) * upScale();

      pos += new Vector3(obj.transform.position.x, tmpScale.y / 2 + scale.y / 2, obj.transform.position
          .z);
      obj.transform.position += pos;
      tmpScale = scale;}}
```



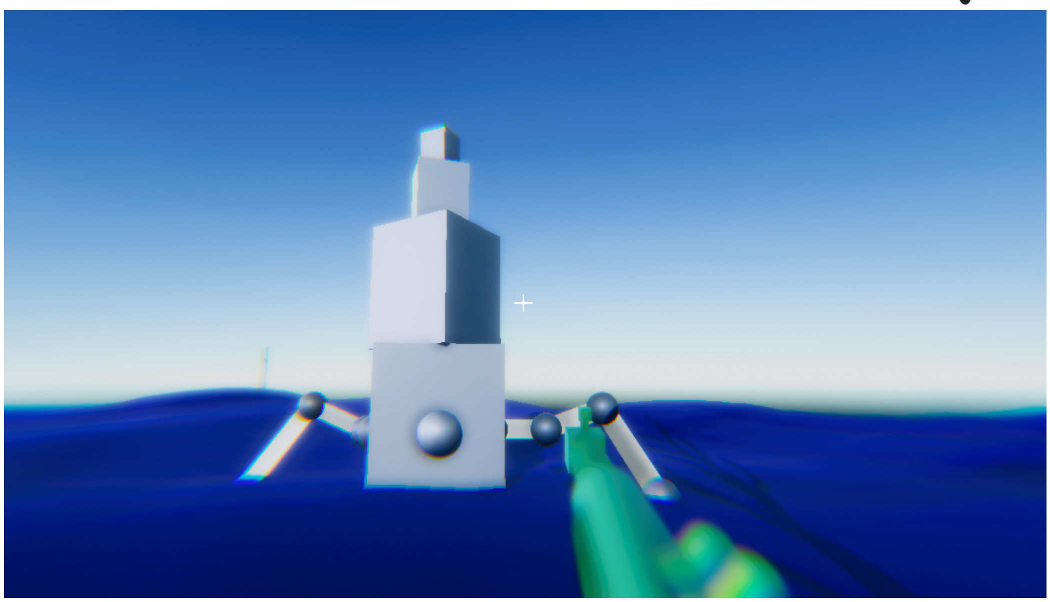


Figure 25 shows the procedural steps of the agent assembling algorithm in code snippet 3.9. This is the creation of a specific agent type that is chosen prior to its creation. The implementation is 500 lines of code, so only a high concept is presented; the reader is therefore referred to the source code in the digital appendix for further interest.

3.4 The narrative paradox

The storyworld helps prevent the narrative paradox, with the Drama Manager proposed in (Schoenau-Fog 2015b). It allows the player to freely roam the storyworld, as the narrative follows the player trajectory during game play. An ISA will be used as a narrative container that follows the player.

To prevent an ISA from being interruptive, it has to initially be located outside peripheral vision. However, this implies that the player follows an linear trajectory, and this is rarely the case, unless the player is driven by curiosity in one particular direction, or the storyworld provides linearity. A solution to this problem is to have the storyworld predict how interested the player is in the ISA. The player is considered to have interest in the ISA when in close proximity or the narrative is interacted with. Disinterest is predicted when the opposite is the case. Although, disinterest should not result in the initial ISA to be excluded from the storyworld. The reason for this is cohesion in the storyworld. This implies that various instances of a particular ISA is generated in the storyworld. This is not a problem if the two instances are not the same. Allowing the ISA to be procedural, and at the same time relying on the narrative to follow the storytelling structure could potentially be a solution to this problem.

To prevent narrative fatigue and cluttering may be a problem, in cases where the player ignores several interactive staged areas. An exhaustion variable is proposed as a solution to this problem. This variable will be addressed in the following section where it is more applicable.

3.5 Combinatorial explosion

The potential solution to prevent the combinatorial explosion is to introduce an exhaustion parameter as similarly to (Booth 2009). The variable dictates all the interaction that the player can perform. This means that the parameter is a vector where each index represents an action. The respective indices degrades over time at a constant linear rate due to the 'favouritism dominance' problem presented in (Hastings et al. 2009). However, each indices increases as the player interacts with the storyworld, which then automatically favours a particular player style and preference. Therefore, a high index value indicates an interest in a particular play style. In this way, the pacing of the narrative becomes a function that is determined by interactions in the storyworld. If the player is highly interested in the storyworld, then the narrative becomes exhausted, which results in a lack of narrative elements, whereas a lack of interest provides more narrative elements. An example of this idea is depicted in the frames below, where a player is in an observational state, and then gradually sets into motion. Notice that the IEC is spaced out in the storyworld due to this action and the spatial detail is increased, as the player becomes more unpredictable. The storyworld predicts the player to now have interest in the storyworld, increasing the agility parameter, whereas, passivity is degraded.

$$Frame\ 1\ : \left(\begin{array}{cccc} Aggression & Passivity & Agility & Destruction \\ \hline 0 & 2 & 0 & 0 \\ \text{IEC (sec)} & \text{IEC distance (m)} & & \\ \hline 5 & 1 & & \end{array} \right)$$

$$Frame\ 2\ : \left(\begin{array}{cccc} Aggression & Passivity & Agility & Destruction \\ \hline 0 & 1 & 1 & 0 \\ \text{IEC (sec)} & \text{IEC distance (m)} & & \\ \hline 3 & 5 & & \end{array} \right)$$

$$Frame\ 3\ : \left(\begin{array}{cccc} Aggression & Passivity & Agility & Destruction \\ \hline 0 & 0 & 2 & 0 \\ \text{IEC (sec)} & \text{IEC distance (m)} & & \\ \hline 2 & 10 & & \end{array} \right)$$

Since, storyworld considers interactions as an indication for change in player intentions, it may succumb to the personalisation paradox. A solution to this paradox has been attempted. A hypothesis element is introduced, which function as a measure to determine the current state of the player. This element is introduced in the storyworld when the player is highly in-mobile. Is the player to show interest towards this element, the storyworld has predicted correctly and the narrative may continue. A false prediction will result in the storyworld to wait for the player to interact, because they may simply be interested in something in particular.

Code snippet 3.10 to determine which objects inside the camera frustum and thus manipulative.

```
public static void manualExhastutionDepletion(byte reduction)
   _exhaustion -= reduction;

public static void deplete(int rate){
   for (int i = 0; i < state.getPlayerState().Length; i++){
      byte value = state.getPlayerState()[i];
      if (value == 0)
         value = 2;
      else if (value == 2)
         value = 0;

      _exhaustion += value;}

public static bool allowedStoryExhaustion(){
   return exhaustionHasToDeplete;}

public static void deplete(int rate){
   for (int i = 0; i < state.getPlayerState().Length; i++){
      byte value = state.getPlayerState()[i];
      if (value == 0)
         value = 2;
      else if (value == 2)
         value = 0;

      _exhaustion += value;}

   if (_exhaustion < 0)
      _exhaustion = 0;
   else if (_exhaustion > 80)
      _exhaustion = 80;

   if (_exhaustion > 10 * rate)
      exhaustionHasToDeplete = true;
   else exhaustionHasToDeplete = false;}

public static float[] getAgentState(){
   agentState[0] = setTmpState[0] + setTmpState[setTmpState.Length - 1];
   agentState[1] = setTmpState[1];
   agentState[2] = setTmpState[0] + setTmpState[setTmpState.Length - 1] - setTmpState[1];
   agentState[4] = setTmpState[0] + setTmpState[setTmpState.Length - 1];

   return agentState;}
```

3.6 The procedural-adaptive storyworld framework

The procedural-adaptive storyworld framework is a game system that aims to link a procedural-adaptive storyworld to player experiences. The framework regards an evaluation for player interactions that is mapped across the different player styles. It then aims enhance the player experience by allowing the player to change the storyworld based on their predicted play styles. Several authors have argued that a homogeneous data flow is important for a procedural-adaptive storyworld. A homogeneous data flow has also lead to a byte vector utilisation. The idea has been to allow a player 'experience' vector and interpolate this vector to the storyworld vector. These vectors are also listed in table 4 to give an overview. The advantage for a vector interpolation is that data structures does not have to be of identical sizes. Note that the data structure chosen here does not have to be exclusive to an procedural-adaptive storyworld, but for this implementation a vector byte structure is used. As evident in the game Minecraft, this data structure should be adequate enough to generate expressive procedural-adaptive storyworlds and validify an experience due to the mathematical capabilities (Edwards 2014). Another reason for using a byte data structure is that it only takes up the storage space needed for a particular byte, compared to integers and floats. The framework also introduces to an emergent storytelling 'side effect' in that the intention has not been to achieve this. The side effect is generated as the storyworld tries to align itself with the player and the ISA implementation. Although, this emergent storytelling can be controlled with semantic constraints.

Vector	**Bytes**	**Description**
Player	4	Aggression, Agility, Passive, and Destruction
Song	6	Action, Exploration, Speed, Happiness and Tension
Environment	6	Weather, Curvature, Sky, Vegetation (form and population), Colourisation and VFX
NPC	5	Aggression, Passive, Flee, Regret and Form

Table 4 shows how the different components in the storyworlds compares.

Figure 26 shows the framework for this implementation. It links the SAP model as proposed by (Zhu and Ontañón 2020; Ontañón and Zhu 2021) with a PCG that is fed into the adaptation module. In works by initially generated the entire storyworld, this is indicated by the open ended arrow connection. Then, afterwards allows the storyworld to generatively adapt towards the player style. The appeal for the NPC indirectly correlated to the storyworld is that the NPC is considered as an individualised character that can take their own actions. Although these actions are based on the storyworld. Notice that for the song selection, a few additional operations have been performed to make the correlation more logical. The average player speed is derived for the passive behaviour and is attributed to the Action, Exploration and Speed variables. Happiness correlates to low a Attack and Destruction behaviour, however, these are emphasised for the Tension. The player model is dependent on the data that can be provided for it. From the case study sixteen songs have been emotionally categorised providing data-driven information. And the player model can be generated from the data that describes how participants prefers an adaptive steering for the storyworld. To achieve a continuous adaptation between the storyworld and the player, a model-driven approach has been used where the play style is evaluated and fed to the adaptation module.

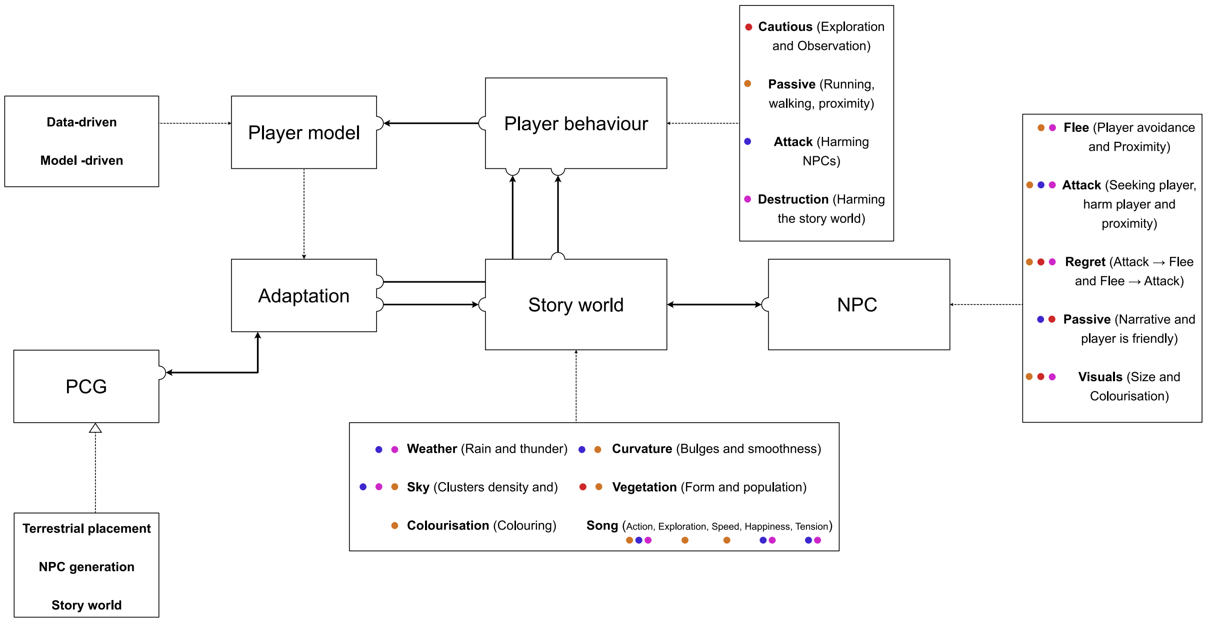


Figure 26 shows the framework for this implementation.

3.7 Conclusion

The presented framework in figure 26 is a proposal to the final problem statement, 'how can a procedural-adaptive storyworld be generated such that it enhances the player experience?'. The developed framework suggest a strong dependency towards the player. The player style and behaviour are defined through four different interactions. These are Cautious, Passive, Attack and Destruction. These interactions resolve in a change for the storyworld as it tries to align itself to the player experience. Player adaptation is based on the player model that uses the data- and model-driven approaches, such that interactions can be mapped appropriately to adaptation. The storyworld is generated in real time by the link between a PCG and adaptation as the player interacts with the storyworld. The solution for the narrative paradox and combinatorial explosion have been oriented around a dependency between these and the player behaviour. The player may depict through their behaviour in the storyworld how often narrative events should be generated and additionally where in the storyworld these narrative events are to occur.

4 Experiment, results and findings

This chapter aims to address the player experience in a procedural-adaptive interactive storyworld. The findings will indicate how successful the implementation has been in regards to develop a framework that aims to achieve a procedural-adaptive storyworld that is dependent on the player. A detailed description of the methodology, experimental and participatory design will be provided; and this chapter will conclude on the findings that have been found subsequently to the testing. The findings will be visually presented and evaluated to hold it against the hypothesis. Since 'an experience' is a term that covers several affective entities two authors are tended to. The PIFF model will be considered as it shows potential in evaluating user experiences in digital games (Bernhaupt 2010; Takatalo et al. 2010).

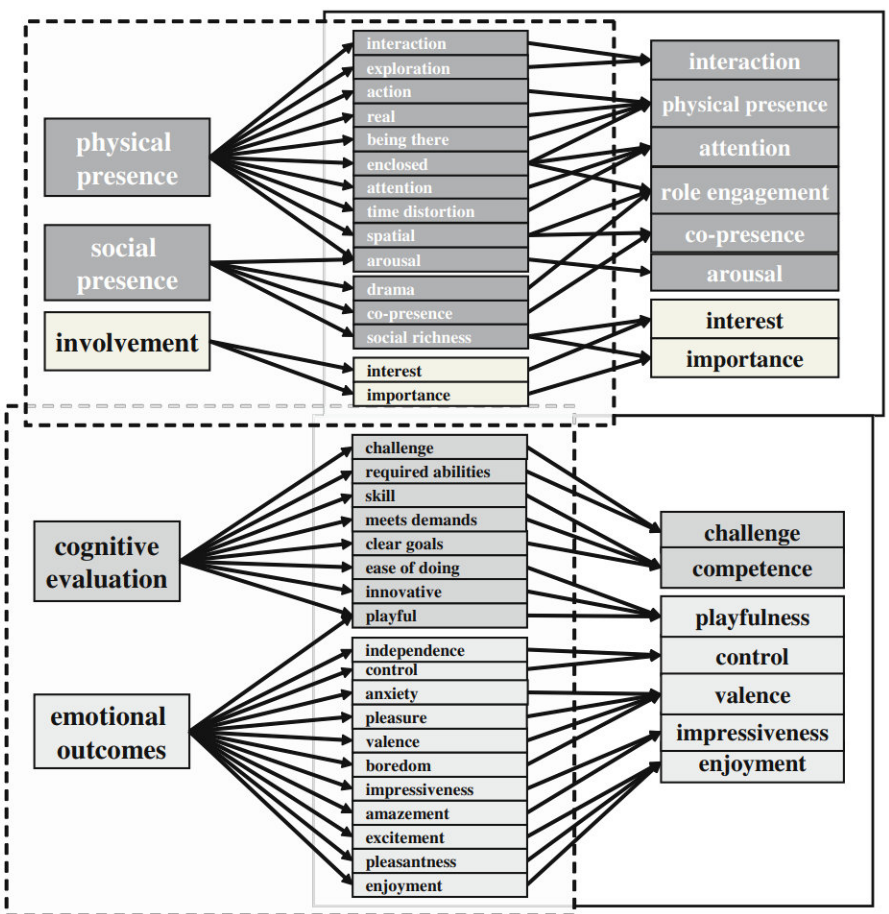


Figure 27 shows an evaluative list for player experiences

Inspiration has also been put from research studies that tries to evaluate an experience in similar setting (Larsen and Schoenau-Fog 2019). In contrast to (Bernhaupt 2010; Takatalo et al. 2010) the participants 'continuation desire' may also draw indications for an enhanced experience. In this regards the participant is asked after each narrative presentation how the experience was for that particular narrative presentation, and why. An example for this evaluation method has been illustrated in table 5.

Plot	Significantly decreased	Decreased	Neutral	Increased	Significantly increased
A			X		
B				X	
C			X		

Table 5 shows a potential continuation desire over a three event narrative.

4.1 Hypothesis

The aim for the implementation has been to develop a procedural-adaptive storyworld that is dependent on the player and to determine whether this potential solution to answer the final problem statement. The goal of the this experiment is then to accept the hypothesis. The following hypothesis has been formulated to test the proposed solution against the final problem statement, 'an interactive procedural-adaptive storytelling paradigm will provide a better experience in a digital game than the same without the two components.

4.2 Methodology

A procedural-adaptive storyworld has been designed to test the final problem statement with the test hypothesis. A traditional storyworld will be presented as a control condition and compared to the developed procedural-adaptive storyworld. The condition order is randomly chosen by the game system at run time, although a test run of 1000 yields in distribution that is skewed, around 48 percent and 52 percent for condition one and two respectively. The selected condition is shown on the screen and the participant is told to remember this condition number for the evaluative questionnaire. Both are explicitly shown at the centre of the screen to the player. The participant will play through the narrative which is composed of four plots. When the condition has finished the remaining condition is automatically chosen.

The control condition is traditional in that the narrative follows a linear narrative progression and the storyworld is not adaptive. Although, the storyworld for the control condition will be initially generated with a PCG. The reason for this is that the storyworld is dependent on the PCG. For the control condition, when the participant tries to progress in the story in a non linear fashion, the following text, *"I am too far ahead, I might return to the story"* will be presented. The procedural-adaptive storyworld uses the player to align itself towards the player experience. After each narrative presentation, this adaptation alignment occurs. It will use the Space-time Drama Manager to present the narrative elements along the player trajectory. The player style and behaviours indicates the pacing for the narrative presentations.

Both conditions rely on exploration in an open world setting, with no directional cues. Thus the player has to explore in the environment for the narrative to be presented. Prior to the game, the participant is told what the objective is and how to proceed in the game conditions. It states, 'The game is about exploring the world and to find artefacts (Pillar(s) with an alter, and two different bonfires). You will have to press *E* at these locations to progress in the story.', because this may not otherwise have been obvious.

4.3 Experimental design

The game experience have been publicly available on five different fora which are all listed in the digital appendix, for seven consecutive days. Four of these fora have specifically been chosen due to their specific orientation and the members' interest in games. Though not strictly limited, it ensures that people whom demographic candidates are related to gaming in general participates in the study. It is difficult to estimate how many people are reached in this way, but at worst, it becomes ≈ 29.8 million as members of these groups may carry over to the chosen groups. The fifth fora, is Itch.IO, which is an open digital platform where independent game developers can post games. The platform allows everyone to post their game content. Notice that this approach indicates that participatory cross testing cannot be directly in-assured, because they may play in two different languages, or different google accounts. Neither is it assured that participants whom test the game will actively participate in the questionnaire that follows.

Each of the five groups are asked to download a short prototype that takes approximately 10-20 minutes to complete. Furthermore, the participants are told that, 'the game is divided into two sections, and is about people's game experience and how game developers can enhance the experience in a game. The game is about exploring the world and to find artefacts, that is pillar(s) with an alter, and two different bonfires. You will have to press 'E' at these locations to progress in the story. You will after the game automatically be referred to a small online questionnaire (Gdocs). In advance thanks your contribution is greatly appreciated'.

4.4 Participatory affiliation design

The participant is automatically directed to a post analytical questionnaire when either the game has ended or the player closes the application window. The questionnaire can be found in the digital appendix. The game has two language settings that determine whether the questionnaire is presented in Danish or English, to reach a broader audience that is not linguistically confident. The questionnaire consist of three demographic questions, that may indicate a potential change in the experience based on play style, gender and the hours playing games. The rest of the questionnaire situates questions as based on the work of (Bernhaupt 2010; Larsen and Schoenau-Fog 2019) to evaluate whether this novel form of game design enhances the player's experience or not. The questionnaire is structured around a Likert's scale structure so that the data can be quantitatively evaluated and compared. A few qualitative questions are situated to gain an insight that may reveal the participant's quantitative opinions more in depth. For instance, by answering *can you elaborate why your continuation desire were as it was?* after they have ranked their continuation desire quantitatively.

4.5 Results and findings

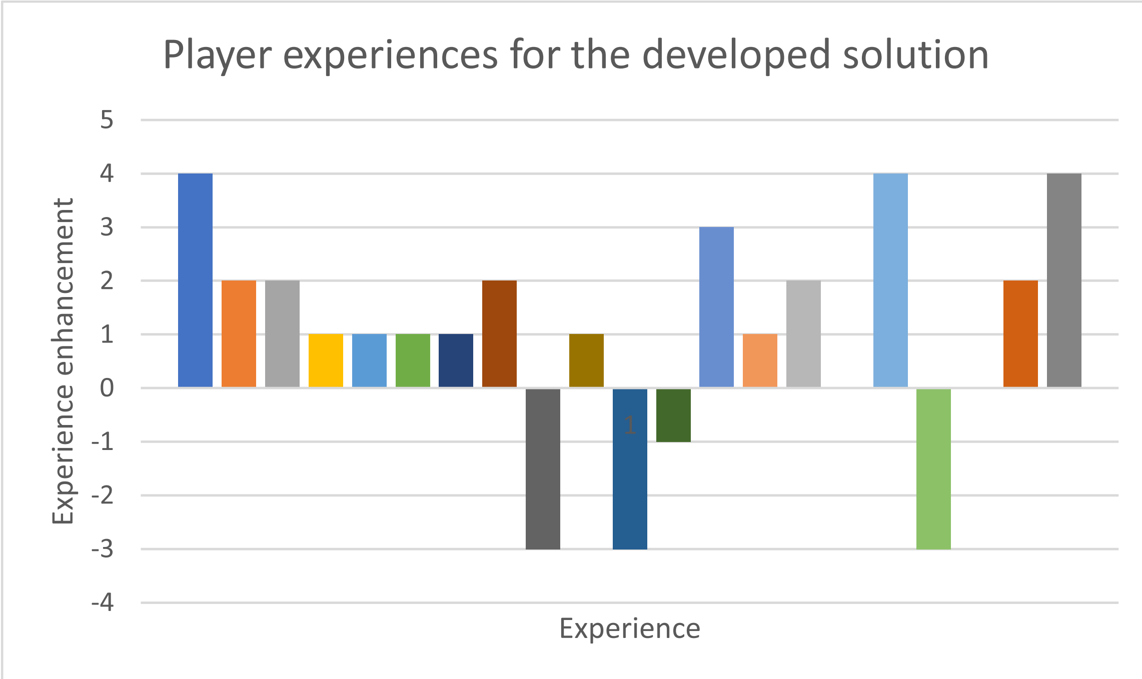
Three participants, one male and two females have participated in the following test. The average play time for these participants where 0-2 hours each week. The play style distribution for the participatory sample is shown in table 8 below. Notice that a player can range over several player types.

Player style	**Percentages**
Action	1 (11 %)
Social	2 (22 %)
Mastery	1 (11 %)
Achievement	1 (22 %)
Creativity	1 (11 %)

Table 8 shows the play style distribution for the population sample.

4.5.1 Player experience enhancement

Figure 28 shows the difference in perceived experience for both the control and proposed solution. The mean experience enhancement value for the non-control condition is (-0.91) compared to the procedural-adaptive storyworld which scored (1.14). The table 7 shows all the evaluated experience elements for this test in chronological order to figure 28.



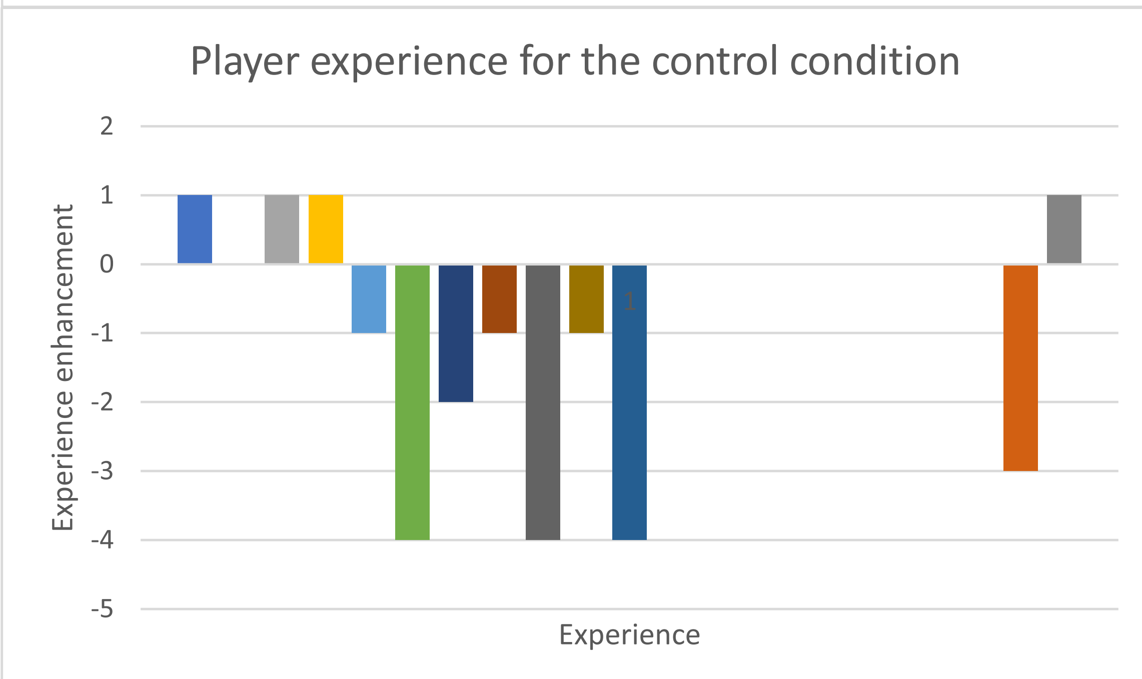


Figure 28 shows how the player experience compares for the two conditions.

Evaluated experiences
I felt like having an active role in the game
I felt that my curiosity was rewarded
I felt engaged
I felt motivated to continue.
I felt that my decisions mattered
I could make my own stories if needed
I felt that the experience was enjoyable
I felt that the experience was fun.
I felt empathy (understand character's feeling(s))
I felt emotional (anger, sadness, tension e.g.)
I felt like being connected (communicatively) with the characters
I felt what happened were exciting
I felt that the pacing was to my liking
I felt that I had control
I felt like having a role in the story
I felt lost into the story world (time flew by)
I felt motivated to explore
I felt that the game was innovative
I felt the goals were clearly defined
The game felt personal
I felt that the game gave me a sense of being absorbed into the world
The world tried to achieve the above to some extend

Table 7 shows the experiences that have been evaluated for this test.

4.5.2 Qualitative experience

A qualitative measure for the experience have been investigated to find potential remarks that may indicate why the non-control condition were preferred over the developed solution. The following table shows these nuances in detail. There is a strong indication that condition one is the more preferred option due to the storyworlds capabilities to be adaptive.

Participant	Comments	Preferred condition
1	I chose the non-control condition because the storyworld was more expressive. It engaged me in that it had more life, colours, music and varying storyworld, adaptation was rewarding and motivated me to continue.	1
2	I got a new experience in the non-control condition, the storyworld seem to be vast without being unreachable, the change to music, visuals was engaging.	1
3	None.	1

Table 8 shows the play style distribution for the population sample.

4.5.3 Continuation desire

The continuation desire has been evaluated for both the control condition and the developed storyworld. The mean continuation desire for the control condition is 0.25 whereas for the non control condition it is 0.83. Figure 29 shows the different continuation desires for the population sample. The mean tendency to continue is indicated by the thickened yellow line. It indicates that the overall desire to continue the experience is highest for throughout the entire narrative. The participants have been asked whether they would pick up the game had it been further developed and release. For the control condition the mean want for a future release was (-0.5) whereas for the proposed solution the mean want was 0.

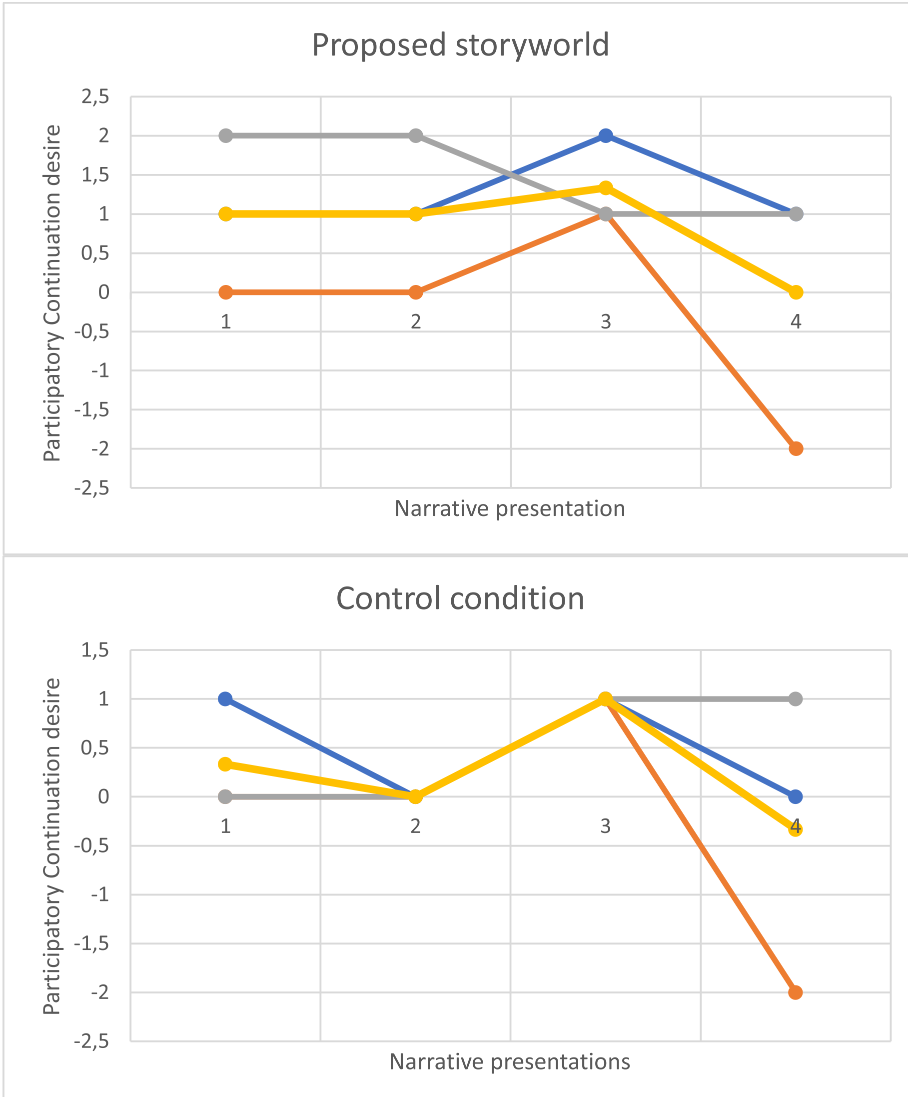


Figure 5 shows the participatory continuation desire for both the control and proposed solution

4.6 Conclusion

A test have been conducted to answer the hypothesis 'an interactive procedural-adaptive storytelling paradigm will provide a better experience in a digital game than the same without the two components'. Three participants have tested and answered a qualitative questionnaire to conclude on this hypothesis. The data indicates that this hypothesis can be accepted. The continuation desire scored highest for the proposed procedural-adaptive storyworld. The participants comments during the qualitative data elicitation suggest the procedural-adaptive storyworld provided in the best experience due to its adaptive capabilities. It is interesting to see that the procedural-adaptive storyworld have achieved this due to the variety of presented play styles. Based on this data indication it would be possible to answer the final research question in that this procedural-adaptive storyworld could be a solution to enhance the player experience.

5 Conclusion

The aim for this report has been to develop an interactive adaptive real time story telling that can help enhance the player experience. For this purpose a segmentation and personalisation approach has been used to characterise player types from interaction, behaviours and intentions. An investigation has yield that procedural content generation can become crossmodal and thus capable of linking the player with a generative storyworld. Player styles, behaviours and intentions helps the storyworld to align itself with the predicted player experience. The player style and behaviour are solutions that have been suggested to prevent the narrative paradox and the combinatorial explosion. The player style and behaviour may depict the pacing and how the storyworlds is presented and generated and where.

A case study has been made to investigate how games with procedural and adaptive paradigms enhances the player experience. This case study suggest that a procedural-adaptive behaviour have to afford player styles and the premise for the game to provide great experience. The second part for this case study have attempted to direct an adaptive steering based on a data-driven approach. Additionally, sixteen songs have been emotionally ranked and used in the game to afford emotional player experiences during game play.

A procedural-adaptive storyworld that depends on the player has been proposed as a solution to the final problem statement, 'how can a procedural-adaptive storyworld be generated such that it enhances the player experience?'. Three participants have therefore tested this procedural-adaptive storyworld. The data indicates that the procedural-adaptive storyworld solution enhances the player experience compared to a traditional game without the procedural and adaptive components to enhance the player experience.

Bibliography

Bakkes, Sander et al. (2009). “Rapid and reliable adaptation of video game AI”. In: *IEEE Transactions on Computational Intelligence and AI in Games* 1.2, pp. 93–104.

Bakkes, Sander CJ et al. (2009). “Opponent modelling for case-based adaptive game AI”. In: *Entertainment Computing* 1.1, pp. 27–37.

Balakrishnan, Guha et al. (2013). “Detecting pulse from head motions in video”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3430–3437.

Bartle, Richard (1996). “Hearts, clubs, diamonds, spades: Players who suit MUDs”. In: *Journal of MUD research* 1.1, p. 19.

Bereznyak, Alexander (2016). *IK rig: Moving forward*. URL: `https://www.gdcvault.com/play/1022984/IK-Rig-Procedural-Pose`.

Bergamin, Kevin et al. (2019). “DReCon: data-driven responsive control of physics-based characters”. In: *ACM Transactions On Graphics (TOG)* 38.6, pp. 1–11.

Bernhaupt, Regina (2010). *Evaluating user experience in games: Concepts and methods*. Springer.

Bidarra, Rafael et al. (2010). “Integrating semantics and procedural generation: key enabling factors for declarative modeling of virtual worlds”. In: *Proceedings of the FOCUS K3D Conference on Semantic 3D Media and Content*. Sophia Antipolis, Méditerranée France, pp. 51–55.

Bjørner, Thomas (2015). *Qualitative Methods for Consumer Research’?* Hans Reitzels Forlag.

Blatz, Michael and Oliver Korn (2017). “A Very Short History of Dynamic and Procedural Content Generation”. In: *Game Dynamics*. Springer, pp. 1–13.

Bonser, Sarah (2020). *World building with architecture*. URL: `http://www.sarahbonser.com/worldbuilding`.

Booth, Michael (2009). “The ai systems of left 4 dead”. In: *Artificial Intelligence and Interactive Digital Entertainment Conference at Stanford, 2009*.

Bordwell, David et al. (2017). *Film Art: An Introduction*. 11th ed. McGraw-Hill Education.

Bosser, Anne-Gwenn et al. (2020). *Interactive Storytelling: 13th International Conference on Interactive Digital Storytelling, ICIDS 2020, Bournemouth, UK, November 3–6, 2020, Proceedings*. Vol. 12497. Springer Nature.

Calleja, Gordon (2011). "Emotional involvement in digital games". In: *International Journal of Arts and Technology* 4.1, pp. 19–32.

Cardona-Rivera, Rogelio et al. (2014). "Foreseeing meaningful choices". In: *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*. Vol. 10. 1.

Carvalho, David B et al. (2017). "Simulated perceptions for emergent storytelling". In: *Computational Intelligence* 33.4, pp. 605–628.

Charles, Darryl et al. (2005). "Player-centred game design: Player modelling and adaptive digital games". In:

Churchill, Elizabeth F (2013). "Putting the person back into personalization". In: *interactions* 20.5, pp. 12–15.

Cohen, Jonathan (2001). "Defining identification: A theoretical look at the identification of audiences with media characters". In: *Mass communication & society* 4.3, pp. 245–264.

Cowan, Nelson (2014). "Working memory underpins cognitive development, learning, and education". In: *Educational psychology review* 26.2, pp. 197–223.

Csikszentmihalyi, Mihaly and Reed Larson (2014). *Flow and the foundations of positive psychology*. Vol. 10. Springer.

Curry, Nancy E and Sara H Arnaud (1974). "Cognitive implications in children's spontaneous role play". In: *Theory into practice* 13.4, pp. 273–278.

Day, Timothy and Jichen Zhu (2017). "Agency informing techniques: Communicating player agency in interactive narratives". In: *Proceedings of the 12th International Conference on the Foundations of Digital Games*, pp. 1–4.

De Kort, Yvonne AW et al. (2007). "Digital games as social presence technology: Development of the Social Presence in Gaming Questionnaire (SPGQ)". In: *Proceedings of PRESENCE* 195203, pp. 1–9.

Dehon, Hedwige (2012). "Illusory recollection: The compelling subjective remembrance of things that never happened. Insights from the DRM paradigm." In: *Psychologica Belgica*.

Dormans, Joris (2010). "Adventures in level design: generating missions and spaces for action adventure games". In: *Proceedings of the 2010 workshop on procedural content generation in games*, pp. 1–8.

Dormans, Joris (2011). "Level design as model transformation: a strategy for automated content generation". In: *Proceedings of the 2nd International Workshop on Procedural Content Generation in Games*, pp. 1–8.

Ebeida, Mohamed S et al. (2012). "A simple algorithm for maximal Poisson-disk sampling in high dimensions". In: *Computer Graphics Forum*. Vol. 31. 2pt4. Wiley Online Library, pp. 785–794.

Edwards, et al. (2014). *Mathematics for Multimedia Applications*. Aalborg University. ISBN: 978-1-78399-714-5.

Gagniuc, Paul A (2017). *Markov chains: from theory to implementation and experimentation*. John Wiley & Sons.

Games, Compulsion (2018). *Soundtrack reveal: The Make Believes*. URL: https://compulsiongames.com/en/news/181/soundtrack-reveal-the-make-believes.

Guckelsberger, Christian et al. (2017). "Predicting Player Experience without the Player. An Exploratory Study". In: *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, pp. 305–315.

Hamari, Juho and Janne Tuunanen (2014). "Player types: A meta-synthesis". In:

Hao, Zekun et al. (2021). "GANcraft: Unsupervised 3D Neural Rendering of Minecraft Worlds". In: *arXiv preprint arXiv:2104.07659*.

Hartmann, Tilo and Christoph Klimmt (2006). "Gender and computer games: Exploring females' dislikes". In: *Journal of computer-mediated communication* 11.4, pp. 910–931.

Hartsook, Ken et al. (2011). "Toward supporting stories with procedurally generated game worlds". In: *2011 IEEE Conference on Computational Intelligence and Games (CIG'11)*. IEEE, pp. 297–304.

Hastings, Erin Jonathan et al. (2009). "Automatic content generation in the galactic arms race video game". In: *IEEE Transactions on Computational Intelligence and AI in Games* 1.4, pp. 245–263.

Henderson, Paul et al. (2017). "Automatic Generation of Constrained Furniture Layouts". In: *arXiv preprint arXiv:1711.10939*.

Holden, Daniel, Taku Komura, et al. (2017). "Phase-functioned neural networks for character control". In: *ACM Transactions on Graphics (TOG)* 36.4, pp. 1–13.

Holden, Daniel, Jun Saito, et al. (2016). "A deep learning framework for character motion synthesis and editing". In: *ACM Transactions on Graphics (TOG)* 35.4, pp. 1–11.

Houdini (2020). *procedural software*. URL: https://www.sidefx.com/products/houdini/.

Hudlicka, Eva (2009). "Affective game engines: motivation and requirements". In: *Proceedings of the 4th international conference on foundations of digital games*, pp. 299–306.

Jenkins, Henry (2004). "Game design as narrative". In: *Computer* 44.53, pp. 118–130.

Jennings-Teats, Martin et al. (2010). "Polymorph: dynamic difficulty adjustment through level generation". In: *Proceedings of the 2010 Workshop on Procedural Content Generation in Games*, pp. 1–4.

Jordan, Mikki (2020). "Emulating and transforming the aesthetics and expressions of mental disorder into a interactive adaptive real-time game". Unpublished.

— (2019). "Using game mechanics for interactive adaptive real-time storytelling". Unpublished.

Kessing, Jassin et al. (2009). "Services in game worlds: A semantic approach to improve object interaction". In: *International Conference on Entertainment Computing*. Springer, pp. 276–281.

Kramer, Joseph et al. (2000). "A user-centered design approach to personalization". In: *Communications of the ACM* 43.8, pp. 44–48.

Larsen, Bjarke Alexander and Henrik Schoenau-Fog (2019). ""Well, That was Quick"–Towards Storyworld Adaptivity that Reacts to Players as People". In: *International Conference on Interactive Digital Storytelling*. Springer, pp. 204–213.

Lima, Edirlei Soares et al. de (2018). "Personality and Preference Modeling for Adaptive Storytelling". In: *2018 17th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames)*. IEEE, pp. 187–18709.

Lohnas, Lynn J and Michael J Kahana (2014). "Compound cuing in free recall." In: *Journal of Experimental Psychology: Learning, Memory, and Cognition* 40.1, p. 12.

Lopes, Ricardo and Rafael Bidarra (2011). "Adaptivity challenges in games and simulations: a survey". In: *IEEE Transactions on Computational Intelligence and AI in Games* 3.2, pp. 85–99.

Louchart, Sandy and Ruth Aylett (2003). "Solving the narrative paradox in VEs–lessons from RPGs". In: *International workshop on intelligent virtual agents*. Springer, pp. 244–248.

Louchart, Sandy, John Truesdale, et al. (2015). "Emergent narrative: past, present and future of an interactive storytelling approach". In: *Interactive Digital Narrative*. Routledge, pp. 185–199.

Lucas, Kristen and John L Sherry (2004). "Sex differences in video game play: A communication-based explanation". In: *Communication research* 31.5, pp. 499–523.

Magerko, Brian (2008). “Adaptation in digital games”. In: *Computer* 41.6, pp. 87–89.

Martin, Glenn A et al. (2010). “The use of functional L-systems for scenario generation in serious games”. In: *Proceedings of the 2010 Workshop on Procedural Content Generation in Games*, pp. 1–5.

Mateas, Michael and Andrew Stern (2005). “Structuring Content in the Façade Interactive Drama Architecture.” In: *AIIDE*, pp. 93–98.

Mazeika, Jo (2016). “A rules-based system for adapting and transforming existing narratives”. In: *International Conference on Interactive Digital Storytelling*. Springer, pp. 176–183.

McClellan, James H et al. (2017). *Dsp first*. Pearson Education.

McMillen, Edmund and Florian Himsl (2020). *Level generation*. URL: https://bindingofisaacrebirth.gamepedia.com/Level_Generation.

Moeslund, Thomas B (2012). *Introduction to video and image processing: Building real systems and applications*. Springer Science & Business Media.

Monolith, Productions (2021). *Shadow of Mordor*. Last accessed 30 July 2021. URL: https://www.shadowofwar.com/shadow-of-mordor/.

Müller, Pascal et al. (2006). “Procedural modeling of buildings”. In: *ACM SIGGRAPH 2006 Papers*, pp. 614–623.

Nack, Frank and Andrew S Gordon (2016). *Interactive storytelling*. Springer.

Namco, Bandai (2021). *Dark souls*. Last accessed 16 October 2021. URL: https://en.bandainamcoent.eu/dark-souls.

Nintendo (2021). *Zelda breath of the wild*. Last accessed 30 July 2021. URL: https://www.zelda.com/breath-of-the-wild/.

Nitsche, Michael et al. (2006). “Designing procedural game spaces: A case study”. In: *Proceedings of FuturePlay* 2006.

Ontañón, Santiago and Jichen Zhu (2021). “The Personalization Paradox: the Conflict between Accurate User Models and Personalized Adaptive Systems”. In: *26th International Conference on Intelligent User Interfaces*, pp. 64–66.

Oswald, Christopher A et al. (2014). “The perceived meaning of the video game experience: An exploratory study.” In: *Psychology of Popular Media Culture* 3.2, p. 110.

Palmero, Cristina et al. (2021). "OpenEDS2020 Challenge on Gaze Tracking for VR: Dataset and Results". In: *Sensors* 21.14, p. 4769.

Pedersen, Christopher et al. (2010). "Modeling player experience for content creation". In: *IEEE Transactions on Computational Intelligence and AI in Games* 2.1, pp. 54–67.

Peirce, Neil et al. (2008). "Adaptive educational games: Providing non-invasive personalised learning experiences". In: *2008 second IEEE international conference on digital game and intelligent toy enhanced learning*. IEEE, pp. 28–35.

Prusinkiewicz, Przemyslaw and Aristid Lindenmayer (2012). *The algorithmic beauty of plants*. Springer Science & Business Media.

Ramstedt, Simon and Christopher Pal (2019). "Real-time reinforcement learning". In: *arXiv preprint arXiv:1911.04448*.

Reynolds, Craig W (1999). "Steering behaviors for autonomous characters". In: *Game developers conference*. Vol. 1999. Citeseer, pp. 763–782.

Riedl, Mark Owen and Vadim Bulitko (2013). "Interactive narrative: An intelligent systems approach". In: *Ai Magazine* 34.1, pp. 67–67.

Ryan (2021). *This is Kaizo Mario*. Last accessed 16 October 2021. URL: `https://kaizomario.techyoshi.com/`.

Ryan, Marie-Laure (2015). *Narrative as virtual reality 2*. Johns Hopkins University Press.

Schell, Jesse (2008). *The Art of Game Design: A book of lenses*. CRC press.

Schoenau-Fog, Henrik (2015a). "Adaptive storyworlds". In: *International Conference on Interactive Digital Storytelling*. Springer, pp. 58–65.

— (2015b). "Adaptive storyworlds". In: *International Conference on Interactive Digital Storytelling*. Springer, pp. 58–65.

Schønau-Fog, Henrik (2012). "Hooked!-Player Engagement in Digital Games as a Driver in Interactive Storytelling, Learning Scenarios and Teaching". In:

Sedikides, Constantine (1992). "Changes in the valence of the self as a function of mood". In: *Review of personality and social psychology* 14.27, pp. l–3.

Shaker, Noor et al. (2010). "Towards automatic personalized content generation for platform games". In: *Sixth artificial intelligence and interactive digital entertainment conference*.

Shoulson, Alexander et al. (2013). "Paste: A platform for adaptive storytelling with events". In: *Ninth Artificial Intelligence and Interactive Digital Entertainment Conference.*

Smelik, Ruben et al. (2011). "Semantic constraints for procedural generation of virtual worlds". In: *Proceedings of the 2nd International Workshop on Procedural Content Generation in Games*, pp. 1–4.

Smith, Gillian, Elaine Gan, et al. (2011). "PCG-based game design: enabling new play experiences through procedural content generation". In: *Proceedings of the 2nd International Workshop on Procedural Content Generation in Games*, pp. 1–4.

Smith, Gillian and Jim Whitehead (2010). "Analyzing the expressive range of a level generator". In: *Proceedings of the 2010 Workshop on Procedural Content Generation in Games*, pp. 1–7.

Smith, Kerri (2013). "Brain decoding: reading minds". In: *Nature News* 502.7472, p. 428.

— (2017). "How to map the circuits that define us". In: *Nature News* 548.7666, p. 150.

Spronck, Pieter et al. (2006). "Adaptive game AI with dynamic scripting". In: *Machine Learning* 63.3, pp. 217–248.

Stern, Andrew (2008). "Embracing the combinatorial explosion: A brief prescription for interactive story R&D". In: *Joint international conference on interactive digital storytelling*. Springer, pp. 1–5.

Stone, Terry Lee et al. (2008). *Color design workbook: A real world guide to using color in graphic design.* Rockport Pub.

Strack, Fritz and Roland Deutsch (2004). "Reflective and impulsive determinants of social behavior". In: *Personality and social psychology review* 8.3, pp. 220–247.

Takatalo, Jari et al. (2010). "Presence, involvement, and flow in digital games". In: *Evaluating user experience in games.* Springer, pp. 23–46.

Tamborini, Ron and Paul Skalski (2006). "The role of presence in the experience of electronic games". In: *Playing video games: Motives, responses, and consequences* 1, pp. 225–240.

Thue, David et al. (2007). "Interactive Storytelling: A Player Modelling Approach." In: *AIIDE*, pp. 43–48.

To, Alexandra et al. (2016). "Integrating Curiosity and Uncertainty in Game Design." In: *Digra/fdg.*

Tutenel, Tim, Rafael Bidarra, et al. (2008). "The role of semantics in games and simulations". In: *Computers in Entertainment (CIE)* 6.4, pp. 1–35.

Tutenel, Tim, Roland Van Der Linden, et al. (2011). "Procedural filters for customization of virtual worlds". In: *Proceedings of the 2nd International Workshop on Procedural Content Generation in Games*, pp. 1–8.

Unity (2019). *Procedural nodes.* URL: https://docs.unity3d.com/Packages/com.unity.shadergraph@6.9/manual/Procedural-Nodes.html.

Valve, Corporation (2021). *Left for Dead.* Last accessed 30 July 2021. URL: https://www.l4d.com/game.html..

Vorderer, Peter et al. (2003). "Explaining the enjoyment of playing video games: the role of competition". In: *Proceedings of the second international conference on Entertainment computing*, pp. 1–9.

Westra, Joost et al. (2010). "Keeping the trainee on track". In: *Proceedings of the 2010 IEEE Conference on Computational Intelligence and Games*. IEEE, pp. 450–457.

WikiDot (2020). *Algorithms for procedural generation.* URL: http://pcg.wikidot.com/category-pcg-algorithms.

Wu, Xingquan et al. (2020). "A Singular Value Decomposition based Maximal Poisson-disk Sampling for adaptive Digital Elevation Model simplification". In: *PloS one* 15.9, e0238294.

Yannakakis, Georgios N and John Hallam (2009). "Real-time game adaptation for optimizing player satisfaction". In: *IEEE Transactions on Computational Intelligence and AI in Games* 1.2, pp. 121–133.

Yannakakis, Georgios N and Julian Togelius (2011). "Experience-driven procedural content generation". In: *IEEE Transactions on Affective Computing* 2.3, pp. 147–161.

Yeh, Yi-Ting et al. (2012). "Synthesizing open worlds with constraints using locally annealed reversible jump mcmc". In: *ACM Transactions on Graphics (TOG)* 31.4, pp. 1–11.

Yu, Derek (2016). *Spelunky: Boss Fight Books# 11.* Vol. 11. Boss Fight Books.

Yu, Lap Fai et al. (2011). "Make it home: automatic optimization of furniture arrangement". In: *ACM Transactions on Graphics (TOG)-Proceedings of ACM SIGGRAPH 2011, v. 30,(4), July 2011, article no. 86* 30.4.

Zadziuk, Kristjan (2016). *Animation Bootcamp: Motion Matching: The Future of Games Animation.* URL: https://www.gdcvault.com/play/1023478/Animation-Bootcamp-Motion-Matching-The.

Zanker, Johannes (2010). *Sensation, perception and action: an evolutionary perspective.* Macmillan International Higher Education.

Zheng, Lim Jia et al. (2020). “Four-class emotion classification in virtual reality using pupillometry”. In: *Journal of Big Data* 7.1, pp. 1–9.

Zhu, Jichen and Santiago Ontañón (2020). “Player-centered AI for automatic game personalization: Open problems”. In: *International Conference on the Foundations of Digital Games*, pp. 1–8.

Zohaib, Mohammad (2018). “Dynamic difficulty adjustment (DDA) in computer games: A review”. In: *Advances in Human-Computer Interaction* 2018.