Adaptive Music Generation for Computer Games

Thesis

ADAPTIVE MUSIC GENERATION FOR COMPUTER GAMES

Anthony Prechtl

A dissertation submitted for the degree of

Doctor of Philosophy

Supervisors:

Robin Laney

Alistair Willis

Robert Samuels

Centre for Research in Computing
Faculty of Mathematics, Computing and Technology
The Open University, UK

Submitted September 2015

Abstract

This dissertation explores a novel approach to game music that addresses the limitations of conventional game music systems in supporting a dynamically changing narrative. In the proposed approach, the music is generated automatically based on a set of variable input parameters corresponding to emotional musical features. These are then tied to narrative parameters in the game, so that the features and emotions of the music are perceived to continuously adapt to the game's changing narrative.

To investigate this approach, an algorithmic music generator was developed which outputs a stream of chords based on several input parameters. The parameters control different aspects of the music, including the transition matrix of a Markov model used to stochastically generate the chords, and can be adjusted continuously in real time. A tense first-person game was then configured to control the generator's input parameters to reflect the changing tension of its narrative—for example, as the narrative tension of the game increases, the generated music becomes more dissonant and the tempo increases.

The approach was empirically evaluated primarily by having participants play the game under a variety of conditions, comparing them along several subjective dimensions. The participants' skin conductance was also recorded. The results indicate that the condition with the dynamically varied music described above was both rated and felt as the most tense and exciting, and, for participants who said they enjoy horror games and films, also rated as the most preferable and fun. Another study with music experts then demonstrated that the proposed approach produced smoother musical transitions than crossfades, the approach conventionally used in computer games. Overall, the findings suggest that dynamic music can have a significant positive impact on game experiences, and that generating it algorithmically based on emotional musical features is a viable and effective approach.

I wish to thank

Robin Laney,

Alistair Willis,

Robert Samuels,

The Open University,

and Jon, Margie, and Jessica Prechtl.

Related publications

An overview of this research was presented at the Convention of the Society for the Study of Artificial Intelligence and Simulation of Behaviour (AISB) in London:

> Prechtl, A., Laney, R., Willis, A., & Samuels, R. (2014). Algorithmic music as intelligent game music. In *Proceedings of the 50th Annual Convention of the Society for the Study of Artificial Intelligence and Simulation of Behaviour (AISB50)*, 1–4 April 2014, London, UK.

Methodological aspects of the research, particularly those discussed in Chapter 5, were presented at the Audio Mostly conference in Aalborg, Denmark:

> Prechtl, A., Laney, R., Willis, A., & Samuels, R. (2014). Methodological Approaches to the Evaluation of Game Music Systems. In *Proceedings of the Audio Mostly Conference (AM '14)*, 1–3 October 2014, Aalborg, Denmark.

CONTENTS

CHAPTER 1

INTRODUCTION

The technologies used in computer games have become increasingly advanced over the past few decades. Modern graphics engines allow games to look nearly photographic, and modern physics engines allow for surprisingly realistic interaction with game objects. Players can often complete tasks and interact with characters any number of ways, with their actions greatly influencing the course and outcome of the narrative. One aspect of games that has made little progress in the past ten to fifteen years, however, is their music. Although the recordings used for game music benefit from increasingly high-quality production values, the underlying systems that trigger, loop, and crossfade them remain largely unchanged. As will be discussed below, these systems ultimately limit the granularity of the music and its ability to provide more than simply background ambience.

This dissertation therefore investigates an alternative approach to game music in which a system automatically generates the music in real time based on a set of input parameters, the variation of which would enable it to dynamically support the changing emotions of the narrative. The research primarily involved the application of algorithmic music to computer games, drawing from prior studies examining the relationship between musical features and expressed emotions. This led to the development of an algorithmic music generator to serve as a proof of concept and basis for further examination of the approach. Empirical evaluation of the system was also an important aspect of the research, not only to validate the work but also because empirical studies of game music have thus far been scarce, and little is known about how it can influence the playing experience. Ultimately, the proposed approach will be shown to be not only technically viable, but also capable of enhancing the game playing experience.

1.1 Problem area

Music is used in multimedia settings such as films and computer games primarily to help convey the *narrative*, or the fictional account of events in the game world. However, the way music is implemented in these media differs quite greatly due to fundamental differences in their narrative structure. Film narratives are *linear* in that they progress exactly the same way every time they are shown—the events of a film always occur at the same times and in the same order. For most films, the music is composed during post-production to a list of cues with exact timings and descriptions of all the important narrative events and scene changes. This is convenient for the composer because the exact course of the narrative is known beforehand, and every aspect of the score can be tailored accordingly. Once the music is recorded, it is added to the audio mix, and, barring further edits to the film, is then permanently synchronized with the narrative. In this regard, film music, and indeed recorded music in general, is also linear in that it proceeds in the same way every time it is played. Computer games, by contrast, feature *nonlinear* narratives which are fixed neither in structure nor in timing. Instead, they rely on user interaction to help determine the course of events, which can often be unpredictable. This presents a problem for the composer, since there is no obvious way to synchronize music—a linear medium—with a nonlinear narrative.

Most modern game music systems address this by dynamically crossfading different music recordings in response to game events (Collins, 2008). In this approach, game developers assign audio files, each containing a recorded piece of music, to the major states of a game. Then, when the game transitions from one state to another, the first state's music fades out while the second state's music fades in. The new music then either loops, crossfades to a similar piece of music, or simply stops playing after a while, until another state is triggered and the crossfading process repeats. What the states actually represent depends largely on the structure of the game. For example, games divided into multiple levels often have different music for each level, whereas games in which the player can move freely about an open world often have different music for each type of geographical location. Many games also use different music depending on the current mode of gameplay, such as whether the main character is in combat, or when more specific narrative sequences are triggered.

This approach to game music has a few notable consequences. First, the individual pieces of music used in games typically lack dynamic and memorable features, which could confuse the player by incorrectly signifying changes in the narrative. Instead, game music tends to exhibit what Berndt et al. (2012) refer to as “structural diffusion”, with the only notable dynamic behaviour arising from crossfades. Another consequence is that during a crossfade, both pieces of music can be heard simultaneously and can easily clash with each other, which could be detrimental to the aesthetics of a narrative transition. Although this effect might be mitigated to some extent by ensuring the neighbouring audio files of any potential transition are harmonically and rhythmically compatible, as well as properly aligned—Müller and Driedger (2012) propose one potential method for doing so—this would restrict the range of musical variation that could be used.

Modern game narratives are much more dynamic and fluid than their older counterparts, and as a result, the problems associated with the crossfading approach are becoming increasingly prominent. One example demonstrating this is the shift in the use of music from the 2007 fantasy game *The Witcher* to its 2011 sequel *The Witcher 2: Assassins of Kings* (both developed by CD Projekt RED). The original features music throughout most of the game, crossfading different tracks mainly after loading screens are shown when the main character moves from one area of the game world to another. For example, if the player moves the main character to a cave entrance, the gameplay pauses and a loading screen appears. When the cave area then finishes loading, the music system crossfades to the cave music and the gameplay resumes from inside. In the sequel, however, many of these transitions progress more naturally, without the use of loading screens. Although this improves the flow of the narrative, the lack of clear boundaries between states means that there are no obvious points at which to crossfade the music. As a result, the sequel uses music much more sparsely than the original, instead often relying on non-musical soundscapes. For example, there is a scene in which the main character is exploring a lush forest on a sunny day, encounters the entrance to a dark cave surrounded by a pool of blood, and can walk directly inside depending on the player's interaction. Although one could imagine the music becoming increasingly ominous as the character moves closer and eventually into the cave, there is no music during this scene. Instead, the game triggers a crossfade between a generic forest soundscape (birds chirping and insects humming) to a cave soundscape (reverberant water droplets and wind sounds) whenever the main character crosses a certain point at the cave entrance.

To summarize, the crossfading approach to game music suffers mainly from a lack of granularity. It reduces the musical representation of the narrative to relatively high-level state transitions, ignoring the more intricate nuances that can arise within states. Of course, the overall granularity could be increased by adding more states and corresponding music, but this would require both greater compositional effort and more crossfades during gameplay, which, as noted above, are often unmusical and distracting. Although the crossfading approach could be appropriate for games with discrete narratives that have clear boundaries between states, many game developers have already begun to move beyond this model.

1.2 Research question and objectives

The research question that guided the present work is as follows:

> **How can music be generated automatically in such a way that it can represent a player's progression through a game narrative?**

Addressing this question involved the investigation of an alternative approach to conventional game music, with the primary aim of being able to better support the dynamic and nonlinear aspects of game narratives. Whereas existing game music systems use audio mixing techniques to adapt linear, pre-recorded music to a nonlinear environment, the proposed approach instead involves the use of a nonlinear music system. Broadly, the approach is to automatically generate a game's music in real time based on a set of input parameters corresponding to musical features with known emotional correlates. Then, by varying the input parameters in response to narrative events and parameters, the music can seem to continuously adapt to support the changing emotional content of the narrative. To explore the viability and efficacy of the approach, the following main objectives were pursued:

- To develop an algorithmic music generator whose input parameters represent emotional musical features
- To determine whether the system can express target emotions, and whether smoothly varying the input parameters results in smooth emotional transitions
- To configure the system to reflect the emotional narrative of a computer game
- To evaluate the system and its configuration in the game in terms of how it impacts the playing experience

1.3 Clarifications of scope

A few clarifications should be made about the scope of the present research. Perhaps most important is the role of the music generator that was developed in the investigation of the underlying approach that guided its design. The music generator is intended to be a robust implementation of algorithmic music, but not the only possible implementation or even necessarily the optimal one. The focus of this dissertation is on how to *use* algorithmic music to support a game narrative, rather than to compare different generation algorithms. Further to this point, the design of the music generator was guided primarily by its emotional expressivity, and additional features which may have enhanced its musicality in other ways were not implemented. Most notably, at present the generator only outputs a stream of chords, which was deemed sufficient to explore the research question.

The second main clarification is the definition of narrative that guided this research. While a *narrative* may be generally defined as any account of events, here the primary concern is the emotional charge of these events, which the music is usually intended to reflect in multimedia contexts. In particular, the goal of the research was for the music to be able to trace the "path" implied by a narrative's changing emotions over time, or its *emotion trajectory*. León and Gervás (2012) refer to such trajectories more generally as "narrative curves", citing the *tension arc*—the progression of the amount of tension in a story over time—as a familiar example. As will be discussed in subsequent chapters, this work involves a similar representation of narrative in which the music traces the emotional tension of a computer game.

Two other clarifications are worth noting. First, the research targets music in the Western classical style. This is mainly because prior studies investigating the relationships between musical features and emotion—upon which aspects of this research were based—have mostly targeted Western classical music. This style is also prevalent in many types of computer games. Although the ideas and findings presented in this dissertation may well be applicable to other musical styles, this is not known for certain. Second, the type of computer game in question is any computer, console (e.g., PlayStation, Xbox), or mobile game with a clear and dynamic narrative. Furthermore, the research targets music during gameplay as opposed to music during *cutscenes*, which are non-interactive, film-like scenes that some games show between major sections of the narrative. Cutscenes in games often feature their own specially-composed music, and because they are linear, there is no clear advantage to accompanying them with a nonlinear music system.

1.4 Dissertation overview

This chapter has briefly introduced and motivated the main ideas that will be discussed in this dissertation. Chapter 2 further contextualizes the proposed approach by reviewing related literature. First, the importance of music in narrative contexts is discussed in terms of its many potential functions and how it has been shown to be capable of impacting the broader narrative experience. Next, several existing music-producing systems are reviewed which are relevant to this research. These include music systems intended specifically for games as well as some more general ones with similar aims. Chapter 2 then concludes by revisiting the main research question and reviewing the research gaps this dissertation aims to address.

Chapters 3 and 4 describe how the proposed approach was implemented. Chapter 3 primarily details the design of the music generator, describing each of the input parameters and how they affect the output music. The generator uses a Markov model to stochastically choose chords, and a constraint-based algorithm to voice lead them. Chapter 3 also presents the results of a preliminary evaluation of an early prototype of the generator. Chapter 4 then describes a computer game that was developed to serve as a testing environment for the generator, and how they were configured to work together. The game uses the first-person perspective and has a simple narrative characterized primarily by a smoothly varying amount of tension. This was reflected musically by adjusting the parameters of the music generator to become more or less tense accordingly.

Chapters 5 and 6 both focus on empirical evaluation. As there are currently no standard methodologies for evaluating game music systems, Chapter 5 categorizes and discusses a number of potential methodologies, drawing from prior literature to provide concrete examples of each. Chapter 6 then presents the evaluative studies that were conducted for this research. In the first study, participants played the game described in Chapter 4 under three different musical conditions and compared them along several subjective dimensions. Their skin conductance was also recorded. Some key findings were that the dynamic music that is the focus of this dissertation made the game more tense and exciting compared to the other conditions, as well as generally more fun. A second study with music experts then demonstrated that the proposed approach produces smoother musical transitions than crossfading.

Chapter 7 concludes the dissertation by revisiting the research question, clarifying the main conclusions drawn from the research, and summarizing its main contributions. Several directions for future work are then proposed.

CHAPTER 2

MUSIC AND THE EMOTIONAL NARRATIVE

Chapter 1 introduced this research primarily from a practical perspective; this chapter further contextualizes it from a more academic perspective. As computer game music is a new and largely unexplored subject of research, this chapter borrows from work in other areas, especially film music. Although games and films differ greatly in some respects, one of the main goals of this research is to enable game music to better support a narrative, much like film music can. Additionally, there seems to be a consensus among game music researchers that game music can perform at least the same functions as film music (Jorgensen, 2006; Berndt and Hartmann, 2007; Collins, 2008).

Section 2.1 first discusses several potential functions of music in films and games, then reviews a number of empirical studies demonstrating different ways that music can impact the perception and experience of a narrative. Section 2.2 then reviews several existing music systems that are either directly or indirectly relevant to games, identifying the main research gaps this dissertation addresses. Finally, Section 2.3 further contextualizes the proposed approach to game music and revisits the main research question guiding this dissertation.

2.1 Theory and research on music in games and films

2.1.1 Potential functions of music

As noted in Section 1.1, it is common for game music to remain mostly static except at major changes in game state, such as transitioning between levels or particular types of gameplay. In this way, the music usually helps to set the broad tone of the different aspects of the game. However, game music could potentially serve a number of other functions. Film music theorist Annabel Cohen (1998) identifies eight functions of film music, each of which could be relevant to games as well:

- Providing continuity
- Directing attention
- Inducing mood
- Communicating meaning
- Providing memory cues (e.g., *leitmotifs*, which are short musical segments associated with a certain character or place)
- Increasing arousal and absorption
- Contributing to the aesthetics of the film
- Masking the sounds of projection equipment

The first five pertain to supporting the narrative in specific ways, whereas the latter three involve enhancing the overall experience of the film. Interestingly, the more narrative-targeted functions distinctly overlap with most of Wingstedt's (2004) six classes of the narrative functions of film music:

- The *emotive* class communicates emotion.
- The *informative* class communicates meaning and values, and establishes recognition.
- The *descriptive* class describes setting and physical activity.
- The *guiding* class directs and diverts viewer attention.
- The *temporal* class provides continuity and defines structure and form.

Table 2.1: Comparison of Cohen's and Wingstedt's categorizations of the narrative functions of music in films. Similar functions are arranged by row.

Cohen (1998)	Wingstedt (2004)
Directing attention	Guiding class
Inducing mood	Emotive class
Communicating meaning	Informative, Descriptive classes
Providing memory cues	Descriptive class
Providing continuity	Temporal class
	Rhetorical class

- The *rhetorical* class "steps forward" and "comments" on the narrative (i.e., the music functions as a quasi-narrator).

Table 2.1 shows the two classifications side-by-side, with similar functions arranged by row. Cohen's three functions primarily concerning the overall film experience rather than supporting the narrative in particular are left out.

Berndt and Hartmann (2008) argue that although the above functions of film music are equally applicable to games, most have not been adequately utilized thus far in games. As discussed in Section 1.1, this is likely due to the fact that in current implementations of game music, fine-grained control over the structure and dynamics of the music is usually not feasible. The music's ability to support the narrative is therefore limited to relatively long-term, high-level changes which may be adequate for conveying the broad tone and setting but not more intricate narrative details. For example, in many games, pre-recorded, high intensity music is faded in when the main character enters combat, then sustained and eventually faded out when the combat ends. In such cases, the music communicates the high arousal (the *emotive* class) one might expect of combat sequences, and directs the player's attention to the enemies (the *guiding* class), but does so in a generic, mostly uninformative way. That is, during these combat sequences the music simply remains at the same level of intensity without reflecting the actual intensity implied by the specific course of the combat. In reality, the emotions implied by a game's combat sequence would likely change throughout its course, depending on multiple factors such as the health status of the main character, and how strong and how many the enemies are. The lack of resolution in the control over the music, however, prevents the music from conveying these, which compromises its ability to fully support the narrative. Indeed, Berndt and Hartmann (2008) argue that game music often only provides a “superficial dramatization” of action sequences (p. 130). The same can usually be said for other types of game sequences as well, especially when the music operates purely in the background.

Berndt and Hartmann (2008) also extend Wingstedt’s (2004) functions of film music to account for the interactivity of game music. They note that in games, unlike films, the player has a degree of control over the narrative; by providing cues to the player, game music can actually affect the shape and outcome of the narrative itself. From a human-centred perspective, the music thus has the potential to function as part of the game’s interface—in other words, as a point of interaction with the player rather than simply a source of information about the narrative. However, they note that this has not been fully explored in games, as with the other functions mentioned above.

2.1.2 The influence of music

Music can have a strong impact on games and films, but defining and measuring this impact is not trivial because music can be employed in a number of different ways and can serve several functions, as outlined in the previous section. One effect that has been well documented in empirical research, however, is music’s ability to influence the perception of a narrative. For example, Marshall and Cohen (1988) found that the amount of strength and activity perceived in two contrasting pieces of music affected the strength and activity perceived when they accompanied a film, as well as the perceived activity of the film’s characters. Similarly, Bullerjahn and Güldenring (1994) found that participants perceived different emotions in a film clip depending on the style of the accompanying soundtrack (melodramatic, crime/thriller, or “indefinite”). With the melodramatic soundtracks the film was perceived as more sentimental and sad, whereas with the crime/thriller soundtracks it was perceived as more thrilling and mysterious; by contrast, the indefinite soundtracks were not rated consistently. The soundtracks also influenced how the participants interpreted the actions and intentions of the film’s main character. The authors conclude that “...film music polarizes the emotional atmosphere and influences the understanding of the plot” (p. 116). These findings were later supported by

Parke et al. (2007a,b), who used multiple regression to model the emotions perceived in film clips with accompanying music as functions of the emotions perceived in the film clips and music separately. More recently, Bravo (2012), showed participants a short film clip accompanied by either a consonant or dissonant[1] chord progression, then had them make several judgments about the narrative and the main character. He found that the difference in dissonance greatly influenced many of the judgments, including the following:

- Whether the character felt confident or was scared
- Whether the character was trying to create or destroy something
- Whether the end of the film would be hopeful or tragic
- Whether the genre of the film was drama or horror

The above findings are perhaps unsurprising, as filmmakers have long been aware of music's ability to provide emotional subtext in films. For example, in the textbook *Directing: Film Techniques and Aesthetics*, Michael Rabiger (2003) notes that "An indifferently acted and shot sequence may suddenly come to life because music gives it a subtext that boosts the forward movement of the story" (p. 542). Perhaps a more important question is whether music can influence the actual experience of a narrative beyond simply helping to clarify it—in other words, whether it can contribute to the experience in a way that another aspect such as the visuals or the dialogue could not. Although narrative experiences are complex and not particularly well understood, a few studies have nonetheless demonstrated some of the different ways music can influence them.

In an early film music study, Thayer and Levenson (1983) had participants view an industrial safety film, which involved graphic depictions of workplace accidents, with either constant documentary-style music, dynamic horror-style music, or no music. They found that the participants had the highest levels of skin conductance in the horror music condition, suggesting that the music actually elevated the intensity of the viewing experience. More recently, Nacke et al. (2010) found that the inclusion of sound and music in a first-person shooter game increased participants' self-reported tension and *flow*—an enjoyable state characterized by deep concentration (Csikszentmihalyi, 1990;

[1] *Consonance* and *dissonance* are discussed in Appendix A.

discussed in Section 5.1.1)—during gameplay. However, they did not find any significant effects of the inclusion of sound or music on skin conductance or facial muscle tension. Interestingly, the music that was used was similar to Thayer and Levenson's documentary music condition in that it did not follow the narrative—it was simply background music. The fact that Thayer and Levenson's horror music condition was the only one between the two studies to produce a significant increase in skin conductance suggests that when music follows the narrative it may be able to have a stronger impact on the experience.

Along these lines, Gasselseder (2014) had participants play an action game under three conditions: one with low arousal music, one with high arousal music, and one with dynamic music that alternated between low and high arousal depending on whether the main character was in combat. He found that the dynamic music resulted in the highest self-reported *immersion*—the feeling of being a part of a virtual world—during gameplay, although the low arousal music resulted in the highest self-reported flow. Regarding the latter, he notes that the low arousal music was the quietest and thus allowed the game's sound effects, which provide important real-time feedback to the player, to be heard the most clearly (one of Csikszentmihalyi's conditions for the occurrence of flow is immediate feedback). He argues that dynamic music in games should therefore reflect not only relatively long-term aspects of the narrative but short-term ones as well. However, this seems to be beyond the capabilities of conventional game music systems.

2.2 Prior game music and emotional music systems

Section 1.1 described the conventional approach to computer game music and discussed how it lacks the granularity necessary for the music to be able to fully support the narrative. This section reviews several alternative music systems and system architectures relevant to this problem which have been proposed by other researchers. These include systems intended specifically for games as well as those with other intended purposes but which could be relevant to games due to their focus on real-time emotional control of the produced music.

One of the earliest proposed alternatives to conventional game music is MAgentA (Musical Agent Architecture; Casella and Paiva, 2001), a high-level system architecture in which the goal is to enable music to follow the mood of a game narrative. It consists of three main components: one that periodically evaluates the mood implied by the narrative, one that chooses and runs a corresponding "mood algorithm"—a music generation algorithm tagged with a particular mood—and one that interfaces with the sound API of the game engine. Though potentially viable as a system architecture, it leaves the implementations of the three modules and the mood algorithms as open problems. Additionally, it does not address how to musically transition between different moods, which compromises its practicality compared to using pre-recorded music. The authors acknowledge the omission, stating that "The transition between moods has to be smooth. The responsibility for such smoothness relies on each mood algorithm's particular implementation" (p. 231). However, no further guidance is provided, and overall it is unclear how different music generation algorithms could smoothly transition between each other.

Another high-level game music architecture is Mind Music (Eladhari et al., 2006), in which the idea is to use music to represent a particular character's emotional state. Notably, this state may be comprised of multiple simultaneous emotions rather than only a single one. In an example implementation of the architecture, the authors describe how a character's overall emotional state (the "inner mood") and how the character emotionally relates to the game world (the "outer mood") could each be represented by the harmony and the time signature of the music, respectively. The inner and outer moods are each divided into five discrete steps with corresponding musical scales and time signatures—for example, the outer moods include *angry* (5/4), *annoyed* (7/8), *neutral* (4/4), *cheerful* (6/8), and *exultant* (3/4)—yielding a total of 5 × 5 = 25 mood combinations. Although this potentially allows for a rich representation of the character's emotional state, it is unclear how these combinations would be perceived in practice, as they were not evaluated. The architecture is also based on pre-composed music, requiring one composition per mood combination. The authors note that the size of the combination space would thus need to be managed to avoid the need for a large number of compositions. Additionally, like MAgentA, Mind Music also does not address how to smoothly transition between moods.

A few concrete game music systems have involved actual generation algorithms. One such system is AMEE (Algorithmic Music Evolution Engine; Hoeberechts et al., 2007; Hoeberechts and Shantz, 2009), which uses a combination of algorithmic methods to generate music from pre-composed patterns. It generates and outputs blocks of music one at a time, and provides control over ten emotions which can each be set individually. When a mood change is made, the current block of music is discarded, and a new one with the target mood is then generated and output. The change is made abruptly, however, and the authors note that "It would be desirable to allow the user to request a gradual change" (Hoeberechts and Shantz, 2009, p. 7). Another concrete game music system is Mezzo (Brown, 2012a,b), in which pre-composed *leitmotifs*[2] are assigned to characters and other narrative elements, and can then be triggered and mapped onto stochastically generated harmonic patterns. The system uses narrative cues to specify when *leitmotifs* should be triggered, suggesting a discrete event-based approach. As with the above systems, however, the question of how transitions could occur is not addressed. Additionally, none of the above systems and system architectures have been formally evaluated, and it is unclear how they would fare in an actual computer game.

A music system that provides some groundwork for transitions is CMERS (Computational Music Emotion Rule System; Livingstone, 2008; Livingstone et al., 2010), which can adjust several features of a given piece of music in real time in order to express a particular emotion. The emotion is specified as one of the quadrants of the two-dimensional valence/arousal space (Russell, 1980), where *valence* represents the extent to which the emotion is positive or negative (i.e., pleasurable or displeasurable), and *arousal* represents the extent to which it is excited or calm. When a new emotion is specified, the rule system adjusts six musical features accordingly, with the emotion–rule mappings drawn from prior studies investigating the emotional effects of manipulating musical features (see Section 2.3). Most of the rules in CMERS can be adjusted continuously over time, though some only allow discrete changes—for example, the mode of the music can only be set to either major or minor. Livingstone (2008) notes that while the continuous rules can inherently be varied smoothly, the discrete rules would instead need to be scheduled to shift in discrete steps at appropriate times such as phrase boundaries. However, this

[2] *Leitmotifs* are short, memorable musical themes often associated with characters or places in the narrative.

functionality was not implemented, and transitions were not tested.

Several other music systems have been developed that can adjust musical features in order to control emotional output (an overview is provided in Williams et al., 2013), though rarely with the level of detail applied in CMERS. CMERS itself was largely inspired by Director Musices (Bresin and Friberg, 2000; Friberg et al., 2006), which has similar aims but which can only adjust performance features of the given music, whereas CMERS can adjust compositional features as well. Emotion also plays a prominent role in a few existing algorithmic music systems. For example, Roboser is an algorithmic music system that uses the valence/arousal space to specify the broad mood of its generated music, and overlays different tracks representing more specific emotions (Wasserman et al., 2003). It was used in an interactive art installation, *Ada*, to help make a large room appear capable of expressing emotion. A more recent example is Robin (Morreale et al., 2013), a system that also uses the valence/arousal space to specify the emotion of its generated music and adjusts its musical features accordingly. However, it produces musical bars successively in a manner similar to AMEE and thus does not intrinsically allow for smooth transitions.

For the most part, prior game music and emotional music systems have not included support for smooth musical transitions. Transitions represent a significant research gap for music producing systems in general, and in the context of games seem to be critical to being able to support a dynamically changing narrative. Another research gap is that although most of the above systems are targeted directly or indirectly at games, none seem to actually have been implemented in a game, and there is little precedent for how a dynamic music system might actually be configured to best support a game narrative. Further to this point, none of the above systems have been empirically evaluated in the context of a game, and only a few have been evaluated in other contexts. Notably, CMERS, Director Musices, and Roboser have each been evaluated in light of their ability to reliably express different emotions.

2.3 Towards a new approach

Most of the music systems described in the previous section rely on the manipulation of musical features as a means to control emotional expression. This approach is well supported by current literature, as a large body of empirical research has investigated the relationship between musical features and perceived emotions. Gabrielsson and Lindström (2010) and Juslin and Timmers (2010) each review the main findings of this research, with the former focusing on structural musical features such as mode and rhythm, and the latter on performance features such as timbre and articulation. In general, the emotional correlates of these features are well understood, and the current view in the field seems to be that the emotions perceived in music are mainly determined by musical features rather than external factors such as the listener or the listening context (Gabrielsson, 2002; Eerola, 2012). Overall, this suggests that musical features could form an effective interface for controlling emotional expression. This has been further supported by systems like CMERS (discussed in Section 2.2), the evaluation of which demonstrated that adjusting the features of a given piece of music can reliably influence the emotions people perceive in it (Livingstone et al., 2010).

Most of the recently proposed music systems intended for games (discussed in Section 2.2) have relied on *algorithmic composition*, or simply *algorithmic music*, which entails the formalization of a set of rules for generating music. An advantage of algorithmic music in the context of supporting a game narrative is that it intrinsically provides full control of the features of the generated music, at least within the confines of the algorithms being used. This could be useful or even necessary to manipulate certain musical features, especially structural ones. For example, Livingstone (2008) notes that an originally planned feature control for CMERS, *harmonic complexity*, could not be implemented without the use of algorithmic music to generate new harmonic structures for the given music. Such features have been shown to have significant effects on emotional expression—Gabrielsson and Lindström (2010) cite nine studies demonstrating the emotional effects of harmonic complexity, for example.

Thus, the central focus of this dissertation is the application of algorithmic music to computer games. The driving research question, *“How can music be generated automat-*

ically in such a way that it can represent a player's progression through a game narrative?" (discussed in Section 1.2), asks how this can be done effectively. The remainder of this dissertation investigates a new approach in which emotional musical features are used as an interface for an algorithmic music system, enabling a game to control the emotional output of the system in a way that reflects its emotional narrative. In doing so, this dissertation aims to address the main research gaps discussed in Section 2.2. First is the issue of how to produce music that can smoothly transition from one set of musical features to another over an arbitrary path and time scale. Connected to this is the broader issue of emotional granularity—specifically, how to express emotions that may lie "between" others in order to support transitions and further dynamic behaviour. Second is the issue of how to configure such a system to effectively support a game narrative. Currently there is little precedent for this since conventional game music systems are limited in their dynamic capabilities, and most recently proposed alternatives have not been implemented in a game. Finally there is the issue of evaluation, which to date has not been explored in depth in game music research.

CHAPTER 3

GENERATING EMOTIONAL MUSIC

The previous chapters motivated and contextualized the approach of generating music algorithmically for computer games. An important aspect of the approach is the idea that controlling features of the generated music with known emotional correlates will allow the emotions it expresses to be controlled in turn. Such musical features could thus provide an intuitive and practical interface by which to support the changing emotions of a game narrative. To test this idea, a stochastic music generator was developed whose input parameters represent desired musical features, and whose output is a stream of chords routed through a third-party synthesizer. At the core of the generator is a Markov model that defines the probabilities of all possible chord transitions. The input parameters control the tonality, voice leading, and performance features of the generated music. Those related to tonality are sent to a mapping function that converts them to the Markov model's probabilities, while the voice leading and performance parameters each control modules that determine how the generated chords should be played. The input parameters can each be varied in real time, with the generator responding as smoothly or sharply as needed.

This chapter focuses on the development and implementation of the music generator. Development was an iterative process consisting of two main phases. The first phase involved the design of a prototype that had a small subset of the planned functionality, and concluded with a preliminary study evaluating its musical and emotional output. The second phase involved refining the generator and preparing it for use in an actual computer game. Section 3.1 first provides an overview of Markov models and how they can be used to generate music, focusing on the general case of the model remaining static over time. Section 3.2 describes the prototype and preliminary study. Finally, Section 3.3 describes the refined version of the music generator, focusing on its architecture and each of its input parameters.

Much of this chapter assumes a basic understanding of a few core concepts of Western music theory, especially intervals and chords. For readers unfamiliar with the subject, a brief summary of the relevant concepts is provided in Appendix A.

3.1 Using Markov models to generate music

A *Markov model* is a type of stochastic model of a system's state transitions. The main component of a Markov model is a *transition matrix* describing the probabilities that each state in the system will transition to each other state. In the context of music, the states typically represent musical events such as occurrences of notes, chords, or phrases (Ames, 1989), and the transitions represent sequential pairs of these events. For example, in the key of C major, a Markov model might define a probability of 0.75 that a G major chord would be followed by a C major, and a probability of 0.25 that it would instead be followed by an A minor. This would make sense because G major is a V chord[1] in the key of C, and in Western music V chords tend to "resolve" to I chords (C major), but instead are sometimes followed by VI chords (A minor) to create tension. Another Markov model might define a lower probability of a V→I transition in order to reflect a more unstable, surprising system.

Much of Western music theory is concerned with which events lead to which others, reflecting the high importance of the relationship between sequential pairs of events. The V→I tendency is only one example. Music theorist Walter Piston summarizes the transition tendencies of common chords in his "Table of Usual Root Progressions" (Pis-

[1] Roman numeral chord notation is discussed in Appendix A.

ton, 1959) in the following way:

> I is followed by IV or V, sometimes VI, less often II or III.
>
> II is followed by V, sometimes VI, less often I, III, or IV.
>
> ...
>
> (p. 17)

which quite strongly implies a Markov model. Similar tendencies arise in other aspects of music as well. In melodies, for example, the leading note of a scale (B, in the key of C major) tends to resolve to the tonic (C), and a leap in a melody tends to be followed by a step in the opposite direction. Such behaviour can be captured quite intuitively with Markov models, and with relatively low complexity, ultimately making them an attractive foundation for a music generation algorithm.

Prior research has also supported the idea that Markov models can be used to encode many of the rules of Western music theory (Farbood and Schoner, 2001; Tanaka et al., 2010). For example, Farbood and Schoner (2001) used Markov models to generate sixteenth-century-style counterpoint melodies (melodies that harmonize with a given melody). The authors were able to encode several of the rules traditionally used in counterpoint writing, including, for example, preferences for certain intervals between the original melody and the harmonizing melody, and preferences for certain intervals between successive notes in the harmonizing melody. Referencing the generated music, they conclude that "Not only does the composed line comply with the strict rules of sixteenth-century counterpoint, but the results are also musical and comparable to those created by a knowledgeable musician" (p. 4). Perhaps unsurprisingly, Markov models have been used extensively in music computing, for example in algorithmic composition (Ames, 1989; Verbeurgt et al., 2004; Eigenfeldt and Pasquier, 2009), style imitation (Farbood and Schoner, 2001; Collins et al., 2011), and interaction with a live performer (Pachet, 2002).

Figure 3.1 shows a simple Markov model with three states representing different musical chords: C major, F major, and G major, which in the key of C are I, IV, and V chords, respectively. The arrows represent chord transitions, while the corresponding numbers indicate their probability of occurrence given the originating chord. For example, the

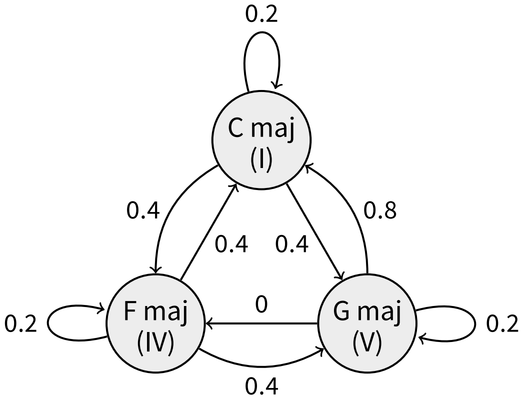


Figure 3.1: Example of a Markov model (a starting state must also be specified)

Current chord	Next chord		
	C maj (I)	F maj (IV)	G maj (V)
C maj (I)	0.2	0.4	0.4
F maj (IV)	0.4	0.2	0.4
G maj (V)	0.8	0	0.2

Table 3.1: The same model in matrix form

model specifies that if the current chord is a G major, there is no probability of the next chord being an F major; instead, the G major will either repeat or, more likely, be followed by a C major. This reflects the tendency for V chords to resolve to I chords, as mentioned above. The same model is shown in matrix form in Table 3.1, where the row i represents the current chord, the column j represents a possible following chord, and the element $p_{i,j}$ gives the probability that j will follow i, or the conditional probability $P(j|i)$. Each row of a transition matrix is, more specifically, a *probability vector*—a vector whose elements are between 0 and 1 (inclusive), and sum to 1. Further to this point, the matrix must, by definition, account for all possible states and state transitions in the system.

Although Markov models can be used for different purposes, in music they are typically used to randomly generate states in a semi-controlled way. Given a specified initial state s_1, new states $s_2, s_3, \ldots, s_n$ can be generated quite efficiently using the following algorithm:

1. Generate a weighted random integer r between 1 and the total number of states, with weights equal to the transition matrix's row representing the current state.
2. Choose the state corresponding to the r^{th} element of the row.
3. Assign the new state to the current one.
4. Return the new state.
5. Repeat.

A more complex question is how the probabilities should be determined. One approach is to train the model from a data set such as a corpus of music by counting the number of occurrences of each state transition in the data set, then converting the counts into probabilities. This is normally appropriate when the end goal is to generate music in the same style as the corpus (see, for example, Collins et al., 2011). However, music tends to exhibit different features from composer to composer, piece to piece, and even section to section within an individual piece, and encoding them together in a single model would make it difficult to distinguish or manipulate particular ones. A more general, "top-down" approach is to use a set of formal rules to define the probabilities. For example, the probabilities of the Markov model shown in Figure 3.1 and Table 3.1 were derived from the following rules:

- Repeated chords should be allowed with a probability of 0.2.
- The V chord should not transition to a IV chord.
- All other transitions should have their probabilities distributed equally.

The main advantage of this approach is that the compositional basis of the model is defined and laid out explicitly, and the rules presented independently. Adjusting the musical characteristics of the resulting Markov model would thus be a matter of adjusting the rules used to define it. This will be discussed further in the following sections.

3.2 Prototype

The first development phase of the music generator involved the design of a prototype to serve as a proof of concept. It was designed with the intention of being controllable in the two-dimensional valence/arousal emotion space (Russell, 1980) via a small set of input parameters. In the valence/arousal space, the *valence* represents the extent to which the emotion is positive or negative (i.e., pleasurable or displeasurable), and the *arousal* represents the extent to which it is excited or calm. Ultimately, the prototype responded to three parameters and produced entirely diatonic output. A preliminary study suggested that despite the prototype's overall simplicity, manipulation of its input parameters led to mostly consistent emotional variations. The following subsections describe the design of the prototype, the preliminary study, and reflections on the prototype that led to changes which were later implemented in the revised version.

3.2.1 Design

The prototype music generator was primarily based on a Markov model of the possible transitions between seven diatonic chords, and thus used a 7 × 7 transition matrix. Given a specified starting chord, it stochastically generated new chords using the algorithm previously described on page 24 (choosing the next chord based on a weighted random number). However, the probabilities of the transition matrix were varied over time in response to an input parameter, *mode*, which specified whether the output should be more consistent with the major or minor mode. Two other input parameters, *tempo* and *velocity*, specified how frequently and how strongly the chords should be played. All three parameters could be varied continuously and in real time during generation.

A new chord was generated every four beats, and a separate arpeggiated[2] version of the chord was played simultaneously, with a new note on every beat. The arpeggio was added mainly to provide a better indication of tempo changes—without it, tempo changes would have only been perceptible every four beats (when a chord was generated), which could have been up to several seconds. For simplicity, the chords were not voice led, but simply played in root position.

The prototype was created with the Max visual programming language, which is commonly used in sound and music computing. It produced a MIDI stream which was routed externally to TASCAM's *CVPiano*, a sample-based virtual piano, for real-time audio synthesis.

The following subsections describe each of the prototype's three input parameters and how they controlled the generated music.

The *mode* parameter (Range: 0–1)

In music theory, a *mode* refers to a type of musical scale. Although scales can be classified in different ways, in this case *mode* refers specifically to the distinction between *major* and *minor* scales. In Western music, the major mode is typically associated with happi-

[2] Arpeggios are explained in Appendix A.

ness and the minor mode with sadness. While other factors can certainly contribute to this distinction, evidence has shown that the major mode is perceived as at least more happy than the minor mode (Hevner, 1935; Temperley and Tan, 2013). This suggests that varying the mode of the generated music from major to minor or vice versa should influence the valence of the perceived emotion.

The *mode* input parameter was a continuous value with one extreme indicating “fully major” and the other “fully minor”. The prototype responded to changes in *mode* by linearly interpolating between two transition matrices—one for major and one for minor—element by element. If *mode* was at its midpoint, for example, then each element (i, j) of the generator’s transition matrix would have been equal to the average of the corresponding (i, j) elements in the two matrices.

The major mode was represented by the key of C major, and the minor mode by A minor, the relative minor of C major. Specifically, the *natural minor* was used rather than the *harmonic minor* so that the major and minor models could share the same chords,[3] even though the harmonic minor is more common in Western classical music. The matrix for each key was generated by starting with each row having a uniform distribution (i.e., all chord transitions being equally probable), then filtering it based on a set of rules. The two transition matrices, shown in Table 3.2, were generated from the following rules:

1. I, IV, and V chords, which each have strong tonal characteristics, were preferred in each matrix. Specifically, minor chords were not allowed in the major model and vice versa. B diminished was allowed in both matrices but to a lesser extent.
2. In C major, transitions from B diminished to C major were preferred, and in A minor, transitions from B diminished to E minor were preferred.
3. Transitions from V to I were strongly preferred in both matrices, though to reduce complexity the minor key did not use a major V chord as is common in Western music theory.
4. No chord could be repeated.
5. All other transition probabilities were equal.

[3] See Appendix A.

Table 3.2: The C major and A minor transition matrices

		C maj	D min	E min	F maj	G maj	A min	B dim
C major	C maj (I)	0	0	0	0.44	0.44	0	0.11
	D min (II)	0.31	0	0	0.31	0.31	0	0.08
	E min (III)	0.31	0	0	0.31	0.31	0	0.08
	F maj (IV)	0.44	0	0	0	0.44	0	0.11
	G maj (V)	0.8	0	0	0.16	0	0	0.04
	A min (VI)	0.31	0	0	0.31	0.31	0	0.08
	B dim (VII)	0.71	0	0	0.14	0.14	0	0
A minor	C maj (III)	0	0.31	0.31	0	0	0.31	0.08
	D min (IV)	0	0	0.44	0	0	0.44	0.11
	E min (V)	0	0.16	0	0	0	0.8	0.04
	F maj (VI)	0	0.31	0.31	0	0	0.31	0.08
	G maj (VII)	0	0.31	0.31	0	0	0.31	0.08
	A min (I)	0	0.44	0.44	0	0	0	0.11
	B dim (II)	0	0.14	0.71	0	0	0.14	0

The process of generating transition matrices using filters is described in greater detail in Section 3.3.1.

The *tempo* parameter (Range: positive)

Tempo refers to the speed at which music is played, and has been shown to be one of the strongest predictors of the arousal dimension of emotion (Ilie and Thompson, 2006; Gabrielsson and Lindström, 2010). The prototype accepted a tempo as a number of beats per minute (BPM), and updated its internal timer accordingly. It is worth noting that, in practice, tempo can sometimes be confounded by *note density*, which refers to the number of notes in a given time span. *Tempo*, by contrast, refers to the number of beats or “pulses” in a given time span, which may be heard or only implied by the rhythm. The prototype made no distinction between the two, however.

The *velocity* parameter (Range: 0–1)

In the MIDI standard, a note onset message consists not only of a pitch, but also a *velocity*, which traditionally refers to the speed at which a key is pressed on a MIDI keyboard.

More generally, it denotes the strength at which the note should be sounded. In most physical instruments, this greatly affects loudness, which has been shown to be a strong predictor of perceived arousal (Ilie and Thompson, 2006). It can also affect the timbre of the sound, however—for example, a strongly played piano note is usually not only louder but also brighter and more shrill than a softly played one.

In the prototype, the *velocity* parameter adjusted the MIDI velocity included with note onset messages. Most modern synthesizers, including *CVPiano*, respond to the velocity component by adjusting both the volume and timbre of the synthesized notes.

3.2.2 Preliminary study

The main objective of the preliminary study was to determine whether varying the prototype's three input parameters would allow it to reliably express different emotions. Accordingly, four sets of parameter values, shown in Figure 3.2, were created with the intention of expressing the four quadrants of the valence/arousal emotion space. These were used to test the following four hypotheses:

H1 The emotions that listeners perceive in the music would match the intended emotions.

H2 *Mode* would correlate positively with perceived valence (i.e., major mode with positive valence, and minor mode with negative valence).

H3 *Tempo* and *velocity* would correlate positively with perceived arousal.

H4 Linearly interpolating from one parameter set to another smoothly over time would result in smooth sounding transitions with distinct starting and ending emotions.

The study consisted of three sections, each following a similar format in which the participant listened to several fifteen-second segments of the prototype's real-time output and answered questions about the emotions they perceived in the music. Each section was preceded by on-screen instructions describing the task and reiterating that the study was investigating perceived rather than felt emotions, as well as one to two warm-up questions. The question and response formats for each section are provided in Figure 3.3.

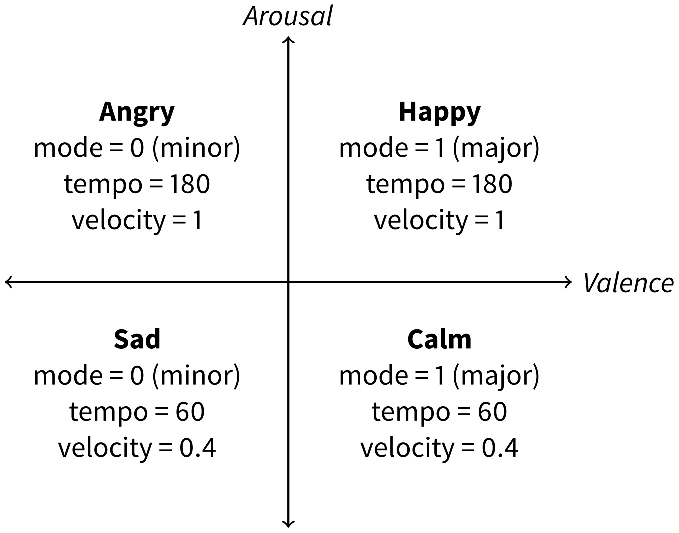


Figure 3.2: Parameter sets for the music generator, and intended emotions

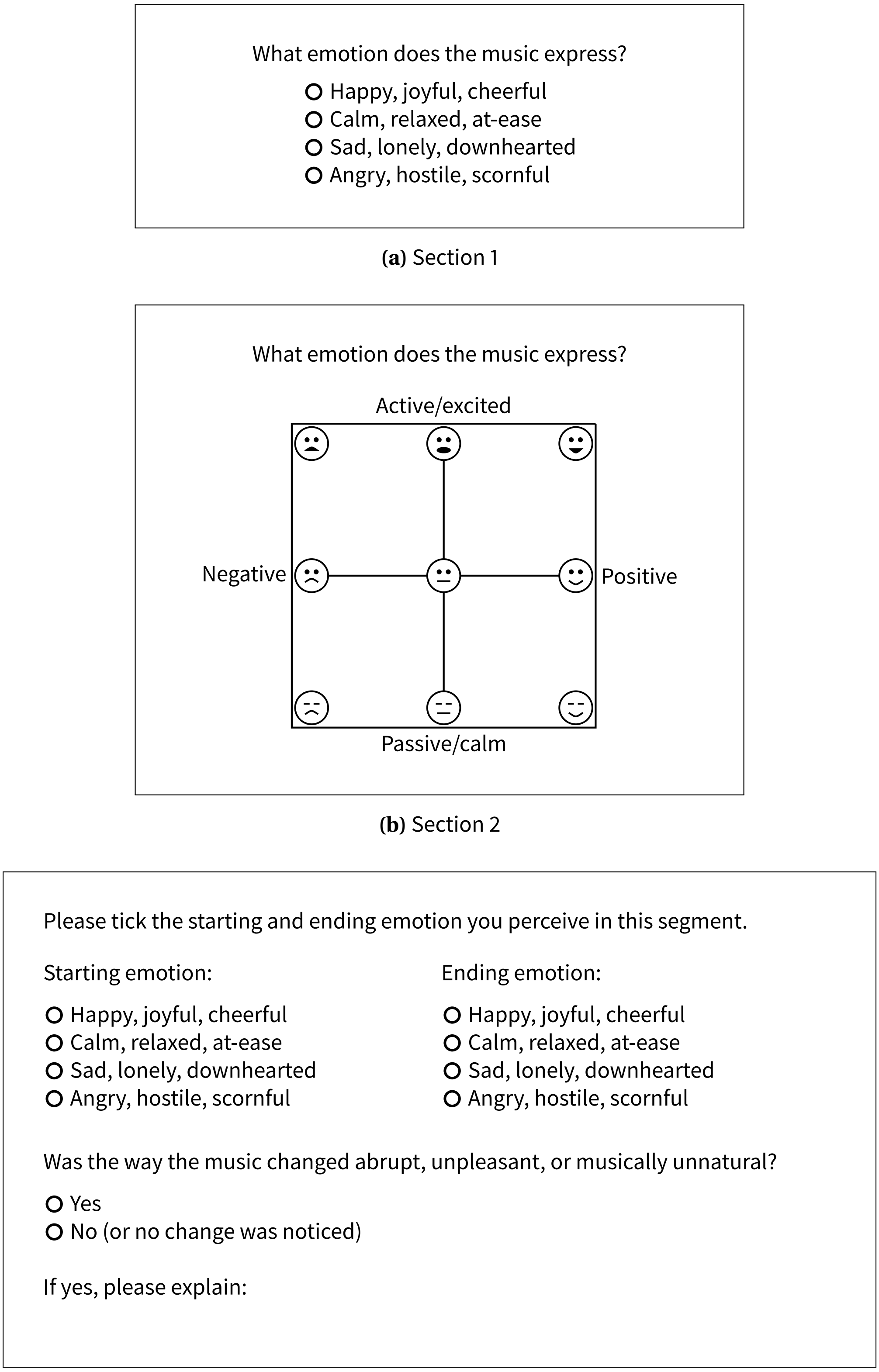
The participants were eight postgraduate students and lecturers from the Computing and Music departments at The Open University. In total, the study took approximately fifteen minutes for each participant to complete.

Section 1

Section 1 tested whether participants perceived the intended emotions of the four parameter sets shown in Figure 3.2, based on textual descriptions of their respective emotions. The participants listened to five segments of the prototype's output, each with a random one of the four parameter sets, and chose the perceived emotion from a list of the four possibilities (see Figure 3.3a). The labels used to represent each of the possibilities were taken from the *joviality*, *serenity*, *sadness*, and *hostility* groupings, respectively, of the Positive and Negative Affect Schedule – Expanded Form (PANAS-X; Watson and Clark, 1994), a discrete emotion classification system commonly used in music psychology studies.

Section 2

In Section 2, the participants used a graphical representation of the valence/arousal model of emotion (see Figure 3.3b) to specify perceived emotions. They listened to fourteen segments, each with a random one of nine different parameter sets, and chose the



What emotion does the music express?

○ Happy, joyful, cheerful
○ Calm, relaxed, at-ease
○ Sad, lonely, downhearted
○ Angry, hostile, scornful

(a) Section 1

What emotion does the music express?

Active/excited
Negative
Positive
Passive/calm

(b) Section 2

Please tick the starting and ending emotion you perceive in this segment.

Starting emotion:	Ending emotion:
○ Happy, joyful, cheerful	○ Happy, joyful, cheerful
○ Calm, relaxed, at-ease	○ Calm, relaxed, at-ease
○ Sad, lonely, downhearted	○ Sad, lonely, downhearted
○ Angry, hostile, scornful	○ Angry, hostile, scornful

Was the way the music changed abrupt, unpleasant, or musically unnatural?

○ Yes
○ No (or no change was noticed)

If yes, please explain:

(c) Section 3

Figure 3.3: Preliminary study question and response formats

perceived emotion by clicking on the valence/arousal graph to specify a point representing the perceived emotion. Four of the parameter sets were the original ones shown in Figure 3.2 and used in Section 1, representing the four quadrants of the valence/arousal model. The other five were averages of all possible combinations of any two of the four parameter sets: happy–calm, calm–sad, sad–angry, angry–happy, and happy–sad (or calm–angry, as both averages yielded the same parameter values). For example, the happy–sad average was *mode* = 0.5 (i.e., "neutral"), *tempo* = 120 BPM, and *velocity* = 0.7.

Section 3

In Section 3, there were three segments consisting of transitions between the original four parameter sets (for example, from happy to sad, from calm to angry, and so on), chosen at random. In each fifteen-second segment, the prototype would use one parameter set for the first three seconds, another for the final three seconds, and during the intermediate nine-second period, it would continuously linearly interpolate from the first set to the second. For each segment, the participants chose the starting and ending emotions from lists of the four possibilities used in Section 1 (see Figure 3.3c). They also indicated whether the transition sounded "unpleasant, abrupt, or musically unnatural".

Results

The analysis of the collected data primarily focused on determining how often the perceived emotion matched the intended emotion. This was also further broken down into how often the perceived and intended emotion's valence matched, and how often the perceived and intended emotion's arousal matched. For example, if the participant chose *positive valence, low arousal* ("calm"), and the intended emotion was *negative valence, low arousal* ("sad"), then the arousal dimension would matched, but not the valence dimension or the overall emotion.

The main results from the study are summarized in Table 3.3. Overall, participants chose the intended valence less often (68%) than the intended arousal (98%), but both were chosen above chance (50%). The intended overall emotion (67%) was also chosen above chance (25%). Interestingly, the participants chose the intended emotion much more often (77%) in Section 2, which used the graphical representation of the va-

Table 3.3: Percentages of cases in the preliminary in which participants chose the intended valence, arousal, and overall emotion

	Responses	Valence match	Arousal match	Overall match
Section 1	40	65%	95%	63%
Section 2*	39	77%	100%	77%
Section 3**	48	63%	98%	63%
Total	127	68%	98%	67%

*Excludes the averaged parameter sets, which were intended to represent intermediary emotions rather than to clearly fall into one of the emotion quadrants
**Includes both the starting and ending emotions

Table 3.4: Multiple linear regression results for the data from Section 2

	Valence (0–1) $R^2 = 0.22$		Arousal (0–1) $R^2 = 0.72$	
	Coefficient	*p*-value	Coefficient	*p*-value
Intercept	0.298	**<0.01**	0.056	0.13
Mode (0–1)	0.265	**<0.01**	0.016	0.726
Tempo/velocity (0–1)	0.128	**<0.05**	0.714	**<0.01**

lence/arousal emotion space (Figure 3.3b), than in Section 1 or 3 (both 63%).

Multiple linear regressions were performed on the data from Section 2 in order to model perceived valence and arousal as linear functions of the *mode*, *tempo*, and *velocity* parameters. These took the format

$$V = \beta_0 + \beta_1 m + \beta_2 t$$
$$A = \beta_0 + \beta_1 m + \beta_2 t,$$

where V is the perceived valence, A is the perceived arousal, m is the mode, and t is the tempo and velocity (because tempo and velocity were varied together in all cases, they were grouped together as one model parameter). The results are shown in Table 3.4. Both mode and tempo/velocity had a statistically significant effect on perceived valence, but mode had a much stronger effect. Tempo and velocity had a strong, statistically significant effect on perceived arousal, while mode did not.

Finally, in 22 out of 24 cases in Section 3, participants responded “No” to the question of whether or not the transition sounded “unpleasing, abrupt, or musically unnatural”. The two “Yes” cases were accompanied by the notes “Unnatural” and “A bit quick and with rather large variation”. They both involved transitions from low to high arousal.

3.2.3 Reflection

The preliminary study demonstrated that participants’ perceived emotions matched the intended emotions of the parameter sets well above chance. However, this occurred less often than would probably be expected to conclusively support H1. In particular, whereas the arousal dimension matched in 98% of cases overall, the valence dimension matched at only 68%. This is reflected in the regression results as well, which show that the arousal model provides a better goodness of fit than the valence model. Interestingly, the match percentages from Section 2 were notably higher than those from Sections 1 and 3. The textual labels used for the emotions in Sections 1 and 3 were intended to be simpler and more intuitive for participants to grasp than the graphical representation of the emotion space, but they were also less direct. That is, the parameter sets were originally designed to reflect the quadrants of the emotion space (*high valence–high arousal*, *high valence–low arousal*, and so on), whereas the textual labels for these quadrants were applied afterward. In any case, however, the identification of emotion is highly subjective, and the fact that neither representation of the emotions resulted in perfect or near perfect match rates was not particularly discouraging. To reiterate, there were only three input parameters, and in fact they all had statistically significant effects on the perceived emotion, as can be seen in Table 3.4. H2 and H3 were therefore supported. This suggests that the prototype was overall a step in the right direction in terms of generating emotional music.

In relation to CMERS (described in Section 2.2), a computational rule system that modifies a piece of music to express a particular emotion, the prototype performed quite well. In their evaluation of CMERS, Livingstone et al. (2010) found that the perceived emotion matched the intended emotion at 78% in one study and 71% in another study. The interface participants used to record the perceived emotion in the CMERS evaluation was a graphical representation of the valence/arousal emotion space nearly identical to the one used in Section 2 of the present study (Figure 3.3b). To recap, in Section 2 the perceived emotion matched the intended emotion at a comparable 77%. Even the match rate of 67% across all three sections is arguably comparable. However, in the CMERS evaluation, numerous musical features were manipulated, six of which were intended to influence perceived emotion, whereas the prototype in this study only manipulated three. Of course, this study also involved fewer participants and musical samples than the CMERS evaluation, and more data would have been needed to make a full comparison.

The match rates from Section 3 were nearly identical to those from Section 1, which implies that transitioning from one parameter set to another using linear interpolation did not confound the prototype's ability to express different emotions. This, together with the fact that only two of twenty-four transitions were rated as "unpleasing, abrupt, or musically unnatural", generally supports H4. The latter also suggests that the aesthetic quality of the prototype was not problematic or detrimental to its purpose. The two negatively rated segments involved transitions from low to high arousal, which entailed the tempo tripling and the velocity more than doubling, both in a relatively short amount of time. Such extreme shifts are uncommon in music, and without a wider narrative context it is easy to see how they could have been considered unnatural. As a result, this was assumed to be a consequence of the relatively large extent of the transitions and lack of context rather than an intrinsic problem with the prototype's ability to perform transitions.

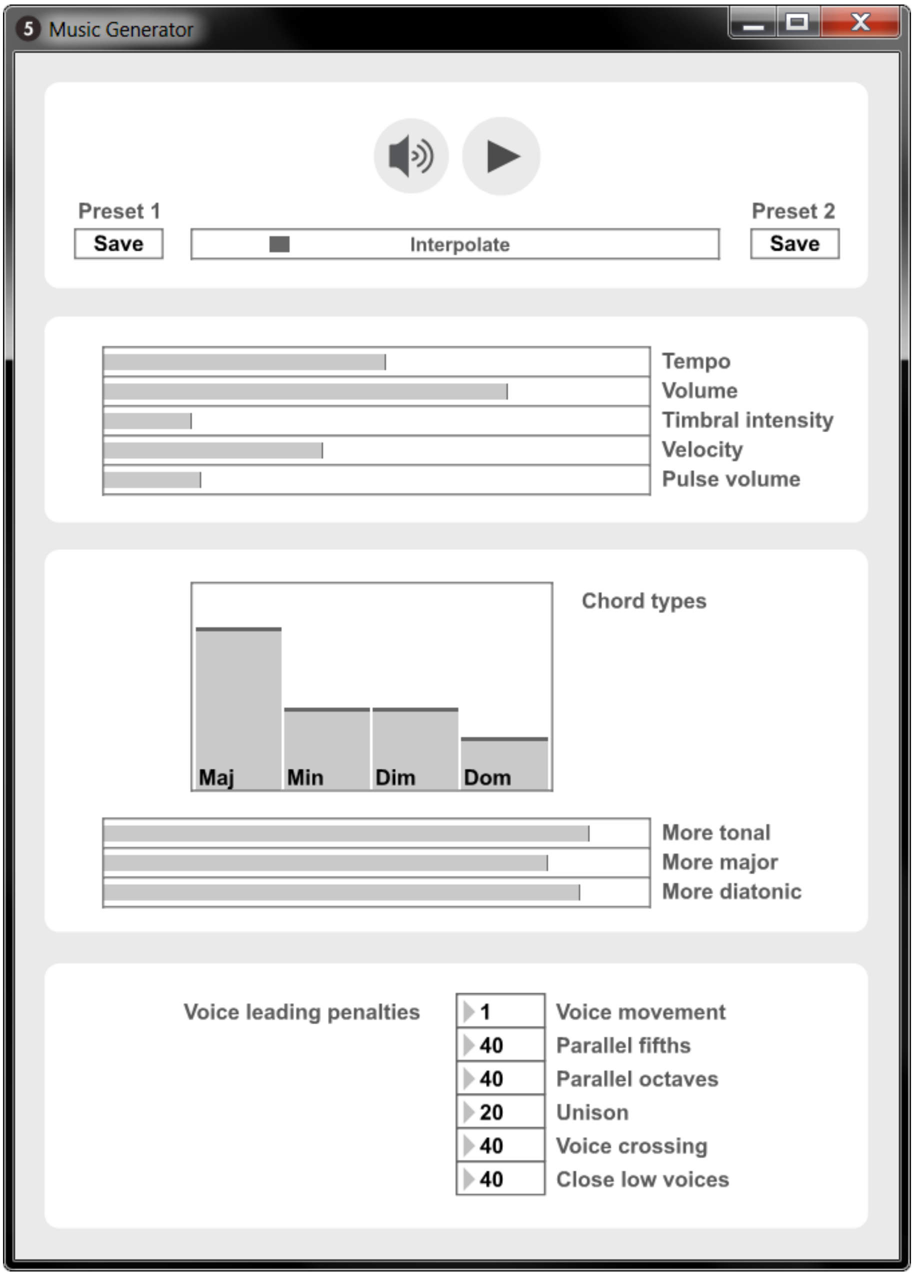
Nonetheless, there were a few points to consider in taking the development of the music generator forward. The first was the lower performance of the valence dimension compared to the arousal dimension. This was perhaps unsurprising given that only *mode* was hypothesized to influence valence—although all three parameters ultimately influenced valence, and with statistical significance, their effect sizes were relatively small. Additionally, as described in Section 3.2.1, the natural minor was used rather than the harmonic minor, which meant that the modality of the minor key may not have been properly emphasized. In practice, it could also be that the perception of valence is simply more ambiguous or personal than the perception of arousal, but in general it made sense to focus on valence as an area for improvement. Another point to consider was the relative simplicity of the prototype's output. Although it was not problematic for the fifteen-second segments used in the study, music in games is normally heard for much longer periods of time. As will be discussed in the next section, these points were addressed through the introduction of a larger transition matrix with more available chords, a more sophisticated method of building the matrix, and more input parameters providing finer granularity of musical control.

3.3 Revised music generator

Development of a revised version of the music generator began shortly after the preliminary study concluded. As noted in the previous section, one of the goals for the revised version was to improve its ability to represent the valence dimension of emotion. This was addressed primarily by updating the chord generation approach to allow for stronger reinforcement of the “majorness” or “minorness” of the music, and to provide control over chromaticism and dissonance. The revised music generator also includes a novel voice leading algorithm, which helps to ensure that the chords are arranged more naturally. Finally, three additional input parameters control the overall volume of the music, its timbral intensity, and the volume of a pulsing bass note.

Like the prototype, the revised music generator was created with Max, though the chord generation and voice leading algorithms were written in Java, which Max natively supports. Additionally, as in the prototype, the revised music generator does not synthesize audio on its own. Rather than sending MIDI messages to an external synthesizer, however, it hosts a dynamically loadable VST[4] instrument internally. The synthesizer can thus be chosen (or made) to suit a specific game. For testing purposes, the revised music generator also provides a GUI, shown in Figure 3.4, as well as a preset mechanism whereby two sets of input parameters can be stored and then interpolated, including during playback. The input parameters are summarized in Table 3.5 and detailed in the following subsections.

[4] VST (Virtual Studio Technology) is a popular audio plug-in format developed by Steinberg.



1 Voice movement
40 Parallel fifths
40 Parallel octaves
20 Unison
40 Voice crossing
40 Close low voices

Figure 3.4: GUI of the music generator

Table 3.5: Summary of the music generator's input parameters

Major chords	Scales the weights of all major chords
Minor chords	Scales the weights of all minor chords
Diminished chords	Scales the weights of all diminished chords
Dominant chords	Scales the weights of all dominant seventh chords
More tonal	Scales down the weights of chords that do not have a clear function in Western music theory (including non-diatonic chords)
More major	Controls the extent to which the above parameter pertains to C major or A minor
More diatonic	Scales down the weights of chords non-diatonic to the key C major or A minor
Voice leading penalties	Penalize note arrangements that violate the respective rule (the arrangement with the lowest total penalty is chosen)
Tempo	The speed at which chords are generated
Volume	The overall volume of the output
Velocity	The velocity with which chords are sounded
Timbral intensity	The timbral intensity of the output (depends on the synthesizer)
Pulse volume	The volume of a pulsing bass note

3.3.1 Mapping musical features to a transition matrix

Chord generation in the revised music generator differs from that of the prototype in two main ways. First, whereas the prototype used a 7 × 7 transition matrix of seven diatonic chords, the revised version uses a 48 × 48 transition matrix of major, minor, diminished, and dominant seventh (major chords with a minor seventh added) chords starting on each of the twelve chromatic notes. The complete list of chords is shown in Table 3.6. Second, whereas the prototype interpolated between major and minor mode matrices based on the *mode* input parameter, the revised version uses a set of filters to dynamically build its transition matrix according to the desired characteristics specified by several input parameters. As will be shown below, the main advantage of this approach is that the filters and their corresponding characteristics can be adjusted independently of one another.

In the filtering process, the elements of the transition matrix are treated as relative weights rather than absolute probabilities, and are initialized to 1. Here, the *weight* of a transition describes how probable it is in relation to the other weights in the same row.

Table 3.6: The forty-eight chords used in the music generator's chord transition matrix

Major chords	Minor chords	Diminished chords	Dominant chords
C maj	C min	C dim	C dom
C♯/D♭ maj	C♯/D♭ min	C♯/D♭ dim	C♯/D♭ dom
D maj	D min	D dim	D dom
D♯/E♭ maj	D♯/E♭ min	D♯/E♭ dim	D♯/E♭ dom
E maj	E min	E dim	E dom
F maj	F min	F dim	F dom
F♯/G♭ maj	F♯/G♭ min	F♯/G♭ dim	F♯/G♭ dom
G maj	G min	G dim	G dom
G♯/A♭ maj	G♯/A♭ min	G♯/A♭ dim	G♯/A♭ dom
A maj	A min	A dim	A dom
A♯/B♭ maj	A♯/B♭ min	A♯/B♭ dim	A♯/B♭ dom
B maj	B min	B dim	B dom

For example, if C maj→F maj has a weight of 1, C maj→G maj has a weight of 0.5, and the current chord is a C major, then the next chord is twice as probable to be an F major than a G major. Thus, initializing the weights to 1 means that all chord transitions initially have equal probability. A number of filters are then applied in sequence to the transition matrix, with each filter representing a particular desired musical characteristic and only adjusting weights relevant to that characteristic. The filtered transition matrix is then used to stochastically generate a chord with a weighted random number as discussed on page 24.

The music generator re-initializes and filters the weights whenever a new chord is requested. Because the current chord is always known, only that chord's corresponding row in the transition matrix actually needs to be built, since the other rows are irrelevant. This drastically reduces the number of required computations, although building the whole transition matrix only takes about 0.2 milliseconds on a modern computer with a 2.3 GHz processor.

The following subsections describe the input parameters and filters that are used to build the generator's transition matrix.

The *major*, *minor*, *diminished*, and *dominant chords* parameters (Range: 0–1)

The first set of filters applied to the transition matrix scales the weights of transitions to each chord type by the value of the respective input parameter. That is, the *major chords* parameter scales the weights of all transitions to major chords, *minor chords* scales the weights of all transitions to minor chords, and so on. Adjusting these parameters could be useful because different types of chords have different emotional connotations—for example, diminished chords tend to sound rather dark and unstable. To avoid any particular type of chord, its respective parameter could be set to a lower value than the others, or to prefer it, the parameter could be set to a higher value. Setting all four chord type parameters to the same value effectively disables them, since no chord type would be preferred over any other.

The *more tonal* parameter and *more major* parameters (Range: 0–1)

Tonality is a broad term used to describe the tendency of music to "resolve" to a particular note or chord called the *tonic*, which normally is the first in the scale. In the key of C major, for example, the tonic would be the note C or the chord C major. As in the prototype, the revised music generator targets the key of C major or its relative minor key (discussed in Appendix A), A minor. However, the prototype used the natural minor scale and only included a few rules to help reinforce the tonality of the music—most notably, the transition tendencies for the V chord and the diminished chord in each key (see p. 27). There are numerous other such tendencies which are well-documented in Western music theory (Piston's Table of Usual Root Progressions, discussed on p. 23, is one example), the use of which could help to further reinforce the extent to which the generated chords reflect the underlying key.

The second set of filters applied to the transition matrix thus provides control over the tonality of the music. The *more tonal* input parameter determines how tonal the generated chords should be, and the *more major* parameter determines whether the key of C major, A minor, or a combination of the two should be reflected. When *more tonal* is set to zero, neither parameter has an effect on the transition matrix. As it is increased towards its maximum value, however, there are two main effects. The first is

Table 3.7: Chromatic chords that are preferred over other chromatic chords (and alongside diatonic chords) as the *more tonal* parameter increases

Chord	In C major	In A minor
Major V	(diatonic)	E major
V7	(diatonic)	E7
Diminished VII	(diatonic)	G♯ diminished
Neapolitan	D♭ major	B♭ major
Tritone sub. for V7	C♯7	A♯7

Table 3.8: Conventional chord transitions which are preferred as the *more tonal* parameter increases

Key	Trans. from	Trans. to	Description
C maj	D min	G maj, G dom, or B dim	II to V, V7, or VII
	F maj	C maj, G maj, or G dom	IV to I, V, or V7
	G maj	C maj or A min	V to I
	G dom	C maj or A min	V7 to I
	B dim	C maj, G maj, or G dom	VII to I, V, or V7
	D♭ maj	G maj or G dom	Neapolitan to V or V7
	C♯ dom	C maj	Tritone sub. for V7 to I
A min	E maj	A min	Major V to I
	E dom	A min	V7 to I
	B♭ maj	E maj or E dom	Neapolitan to V or V7
	A♯ dom	A min	Tritone sub. for V7 to I

that the weights of all transitions to "non-tonal" chords are scaled towards zero. Here, tonal chords include the seven diatonic chords of C major and A minor as well as several chromatic chords that have conventional functions in tonal music, perhaps the most important of which are the major V and V7 (dominant seventh) chord—elements of the harmonic rather than the natural minor. The chromatic chords are shown in Table 3.7. The second effect is that conventionally tonal chord transitions become increasingly preferable by scaling the weights of all alternative transitions towards zero. These transitions are shown in Table 3.8.

As noted above, the *more major* parameter determines the extent to which the chords and transitions that are preferred are consistent with the key of C major (the second column of Table 3.7 and top half of Table 3.8), A minor (the third column of Table 3.7 and bottom half of Table 3.8), or a combination of both. At its minimum value, *more major* causes *more tonal* to only apply the A minor effects; as it is increased, the A minor effects are reduced and the C major effects increased.

The *more diatonic* parameter (Range: 0–1)

Diatonic chords tend to sound very consonant and harmonious, and thus it could be useful to prefer or avoid them. Thus, the *more diatonic* parameter specifies how often the output chords should be diatonic to the keys of C major or A minor (which share the same diatonic chords). To do so, it scales the weights of transitions to non-diatonic chords by a factor of one minus the value of *more diatonic*. This is distinct from the *more tonal* parameter, which includes both diatonic and non-diatonic chords, and for which the intent is to provide the ability to emphasize the tonic, whereas with the *more diatonic* parameter the intent is to control the consonance and dissonance of the chosen chords. At its minimum value, *more diatonic* has no effect. As it is increased, however, the weights of transitions to non-diatonic chords decrease towards zero, and eventually, when *more diatonic* reaches its maximum value, are not be allowed at all.

3.3.2 Voice leading

As demonstrated in Appendix A, normally the notes of a chord can be arranged in many different ways. *Voice leading*, the process of dividing a chord progression into multiple simultaneous melodic lines, or “voices”, aims to find the best arrangement, the selection of which normally depends on a number of stylistic rules and conventions. These can pertain both to how the voices should move from chord to chord (their “horizontal” arrangement) and how they should be arranged within the context of a single chord (their “vertical” arrangement). A canonical example of voice leading is the composition of melodies for the different parts of a choir, which harmonize together to form a sequence of chords. However, voice leading applies equally to instrumental music.

The music generator implements a novel algorithm that treats voice leading as an optimization problem with hard and soft constraints. *Hard constraints* must be satisfied for a particular note arrangement to be considered for selection, while *soft constraints* need not be satisfied, but their violation bears a specified penalty. For a given input chord, the algorithm finds the optimal arrangement by first iterating through all possible solutions as determined by the hard constraints, then penalizing each one according to which soft constraints it violates, and finally choosing whichever arrangement receives the lowest overall penalty. It then returns the arrangement as a list of four pitches.

The structure of the voice leading algorithm is analogous to the method that students are taught to perform voice leading. For example, in his *Preliminary Exercises in Counterpoint*, composer Arnold Schoenberg (1969) writes the following after reviewing a number of voice leading rules:

> But it must be admitted that the severe restrictions at this stage make it almost impossible to write many examples which do not violate one rule or another. [...] However, in spite of this it pays to try everything. There is the possibility of discriminating between greater and lesser 'sins'. (p. 14)

which is essentially what the algorithm does. As Schoenberg notes, in practice it is often impossible to arrange a chord without violating different rules and conventions—constraints provide an intuitive way to distinguish between different levels of severity in order to find the optimal solution. Unsurprisingly, constraint-based algorithms have been used to solve similar problems in music, most notably the harmonization of a given melody (Ebcioglu, 1988; Tsang and Aitken, 1991), in which one or more new melodies are added in order to create a sequence of chords. The present problem is distinct, however, in that the chord is given (by the chord generation algorithm described in Section 3.3.1) rather than the melody, and the task is to lead *all* voices from their arrangement for the previous chord to the optimal arrangement for the new chord.

Each of the four voices produced by the voice leading algorithm has a designated range in order to prevent it from moving too high or low in relation to the others. The ranges themselves are somewhat arbitrary, but in this case are consistent with the following standard vocal ranges taken from a modern music theory textbook (Aldwell et al., 2011, p. 94):

Soprano: C4–G5

Alto: G3–C5

Table 3.9: Voice leading penalties and input parameters

Input parameter	Penalty basis
Voice movement	Number of steps moved by each voice from its position in the previous chord to its position in the new chord (i.e., prefer less voice movement)
Parallel fifths	Instances of parallel fifths
Parallel octaves	Instances of parallel octaves
Unison	Any two voices performing the same pitch
Voice crossing	Any two adjacent voices swapping their pitch order (“crossing”)
Close low voices	The two lowest voices (bass and tenor) being less than a fourth apart

Tenor: C3–G4

Bass: E2–C4

These ranges are implemented as hard constraints in the voice leading algorithm. The algorithm also uses four other hard constraints:

- Only notes from the given chord may be performed.
- All notes from the given chord must be performed except the fifth, which is sometimes omitted in voice leading (Aldwell et al., 2011, p. 95) and is therefore optional.
- Only the root of the chord may be performed by two voices, or “doubled”.
- Chords may only be played in root position or the first inversion (the second inversion is normally avoided except during cadences).

Ultimately, the hard constraints serve mainly to limit the complexity of the voice leading task, as well as to help improve the algorithm’s efficiency, since any arrangements that violate a hard constraint do not require further testing. By contrast, the soft constraints manage violations of rules and conventions, the severity of which (if any) can vary from style to style. The penalties associated with the soft constraints are therefore provided as input parameters to the music generator, and can be varied over time, if desired. These are discussed in turn below and summarized in Table 3.9.

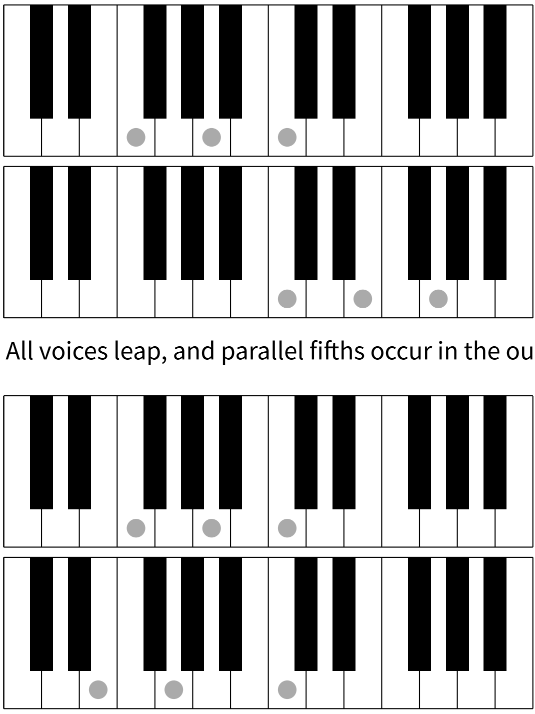
The *voice movement* parameter (Range: positive)

When voice leading there is usually a preference for the voices to move in steps or even repeat notes rather than to move in leaps (i.e., by more than one step). Aldwell et al. (2011) note that although leaps can add interest to a melody, if used too often they can prevent it from "holding together" (p. 103) as well as make it more difficult to perform. The voice leading algorithm addresses the preference for less voice movement by penalizing arrangements by the value of the *voice movement* input parameter for each step moved by each voice from its position in the previous chord's arrangement. Normally it makes sense for the *voice movement* parameter to be relatively small compared to the others, since the idea is more to prefer the simplest solution than to explicitly avoid leaps, especially not at the cost of violating other rules.

Figure 3.5 shows an example F major to C major transition in both a naïve arrangement and one in which voice movement has been minimized through voice leading. In the naïve arrangement, after playing the F major, all three voices move five steps up to play the C major chord. In the voice led arrangement, the two lower voices move down by only one step, while the upper voice does not need to move at all. The resulting chord is still a C major (it contains the notes C, E, and G), but its arrangement minimizes the distance that each voice needs to move.

The *parallel fifths* and *parallel octaves* parameters (Range: positive)

One of the core principles of voice leading is to maintain independence of the voices. Each voice should ideally have a distinct melody, and not seem to move together with any other voice. While the latter is sometimes unavoidable, two cases that are often strictly forbidden in Western music are parallel fifths and parallel octaves. *Parallel fifths* occur when two voices a fifth apart in one chord each move up or down by the same interval (i.e., in parallel), thus remaining a fifth apart in the next chord. *Parallel octaves* occur when two voices instead begin and remain an octave apart. To help avoid parallel fifths and octaves, the voice leading algorithm penalizes any arrangements containing them by the value of the of the *parallel fifths* and/or *parallel octaves* input parameters, respectively.



(a) *Naïve*: All voices leap, and parallel fifths occur in the outer voices

(b) *Voice led*: Voice motion is minimized, and the parallel fifths have been removed

Figure 3.5: Naïve and voice led versions of an F major to C major transition

Figure 3.5a shows an example of parallel fifths, since the outer two voices are a fifth apart in both the first and the second chords. In the voice led arrangement shown in Figure 3.5b, the outer two voices are a fifth apart only in the first chord, and a sixth apart in the second, thus avoiding the parallel fifths. It is worth noting that while the voice led arrangement in this particular example both reduces voice movement and avoids parallel fifths, in many cases there is more of a trade-off, where avoiding parallel fifths or octaves demands greater voice movement than would otherwise be necessary.

The *unison* parameter (Range: positive)

A *unison* occurs when two voices share the same pitch. It can create a sense of thinness in the chord compared to adjacent ones because it means that one fewer unique pitch is being sounded than otherwise could be. Unisons are therefore usually avoided when voice leading. The voice leading algorithm addresses this by penalizing any arrangements that contain a unison by the value of the *unison* input parameter.

The *voice crossing* parameter (Range: positive)

Although the ranges of the voices can and often do overlap, their pitch order is normally meant to be maintained when voice leading. In other words, the lowest voice should always sound the lowest pitch, the next lowest voice should always sound the next lowest pitch, and so on. *Voice crossing* occurs when the pitch order of the voices changes, even for one chord. To help avoid this, the voice leading algorithm penalizes any arrangements that contain crossed voices by the value of the *voice crossing* parameter.

The *close low voices* parameter (Range: positive)

Voices can be close together in pitch or more spread out. However, it is normally recommended that low pitches are not too close together, which can result in a "muddy" sound in which they are difficult to distinguish. The voice leading algorithm therefore penalizes any arrangements in which the lower two voices (the tenor and bass) are less than a fourth apart, based on the value of the *close low voices* parameter. This does not include unisons, however, which are penalized separately.

3.3.3 Other input parameters

Several other input parameters affect how the music generator performs chords. Two of them, *tempo* and *velocity*, remain unchanged from their implementation in the prototype, and thus are not discussed here (*tempo* is discussed on p. 28 and *velocity* on p. 28). The remaining parameters include *volume*, *timbral intensity*, and *pulse volume*.

The *volume* parameter (Range: 0–1)

The *volume* parameter controls the volume of the output music. It was implemented mainly to assist with audio mixing and fade ins and fade outs, but it also provides a means of controlling volume for its emotional effect. As noted in the description of the *velocity* parameter (p. 28), most synthesizers use the velocity component of a MIDI note onset message to control both the volume and the timbre of the output audio. However, in most synthesizers the velocity of a note cannot be changed after its onset. Thus, changes in the *velocity* parameter of the music generator are not perceptible until new notes are sounded. By contrast, changes in the *volume* parameter have an immediate effect.

Most synthesizers allow velocity mappings to be configured, in which case it could make sense to reduce the effect of the velocity→volume mapping within the synthesizer, and use the music generator's *volume* parameter as a supplement.

The *timbral intensity* parameter (Range: 0–1)

As mentioned above, the velocity component of a MIDI note onset message typically controls both the volume and the timbre of the output audio, but it cannot be changed after the note onset. Thus, there is a delay between when the *velocity* parameter changes and when its effect is perceptible. Similar to the *volume* parameter, *timbral intensity* therefore provides immediate control over the timbral intensity of the audio. Changes in *timbral intensity* are simply relayed to the synthesizer as a MIDI control change message. Most synthesizers provide a digital filter with a variable cutoff frequency, in which case the *timbral intensity* parameter could be mapped to the cutoff frequency. Another option could be a distortion or overdrive sound effect, if one is provided. In any case, it is up to the user and the synthesizer to handle the control change messages in the desired way.

The *pulse volume* parameter (Range: 0–1)

The *pulse volume* parameter controls the volume of a single pulsing note. The note is sounded once per beat at the same pitch as the bass note of the current chord. It is intended to mimic the sound of a heartbeat, which is often used to express distressing situations in films. Although the effect has not been empirically studied thus far, Winters (2009) argues that the use of heartbeat-like sounds in film music connects the physical body of the viewer with those of the on-screen characters, thus heightening the viewer's simulation of the danger felt by the characters. The idea of the *pulse volume* parameter is to be able to signal distressing narrative situations, especially in cases where the player actually controls a particular character.

3.4 Chapter summary

This chapter described an approach to algorithmically generating music based on input parameters representing emotional musical features. Chords are first stochastically generated using Markov models with variable probabilities, then voice led, and finally converted to audio by a synthesizer, with the input parameters controlling each step of the process. The input parameters and their respective musical features were chosen on the basis of having a known effect on the emotion of the generated music. The preliminary study described in Section 3.2.2 demonstrated that the expressed emotion of the music could be controlled reasonably well by manipulating the input parameters.

Of course, Markov models are not the only viable way to algorithmically generate music, as there are many well-established approaches—Nierhaus (2009) reviews at least eight, including algorithms based on Markov models, generative grammars, neural networks, and others. An advantage of Markov models, however, is that they can present the musical properties of a system quite clearly, as was demonstrated in Section 3.1. They also tend to be computationally tractable, requiring few computations to both model and generate music, which makes them attractive as a basis for real-time generation.

CHAPTER **4**

CONTROLLING DYNAMIC MUSIC IN A GAME

The music generator performed relatively well in the preliminary study, and was then further developed to provide greater musical and emotional control. The next major objective was to test it in the context of a computer game in order to determine whether varying its musical features would enable it to support and enhance the game's emotional narrative. This involved the development of a short game, *Escape Point*, and an interface and set of parameter mappings for it to control the music generator. In the game, the player navigates through a large 3-D maze, avoiding several patrolling enemies while attempting to find the exit. This is intended to be broadly representative of a common game scenario, as the task of navigating though mazes and large, complex spaces in general arises in several types of games, as well as the need to avoid enemies while doing so. During the game, the music becomes more or less tense depending on the enemies' proximity to the player, thus reflecting the changing amount of tension in the narrative.

Escape Point was created with Unity, a cross-platform game engine widely used for both commercial and independent games. Its programmatic functionality was coded in C#, and it controls the music generator remotely by sending parameter change messages via a UDP (User Datagram Protocol) connection. To help minimize development time, it was designed with only one main level and with minimal graphics and game mechanics. This also helped to make the game more accessible for novice game players and provided greater experimental control for the evaluative study that was later conducted (discussed in Chapter 6).

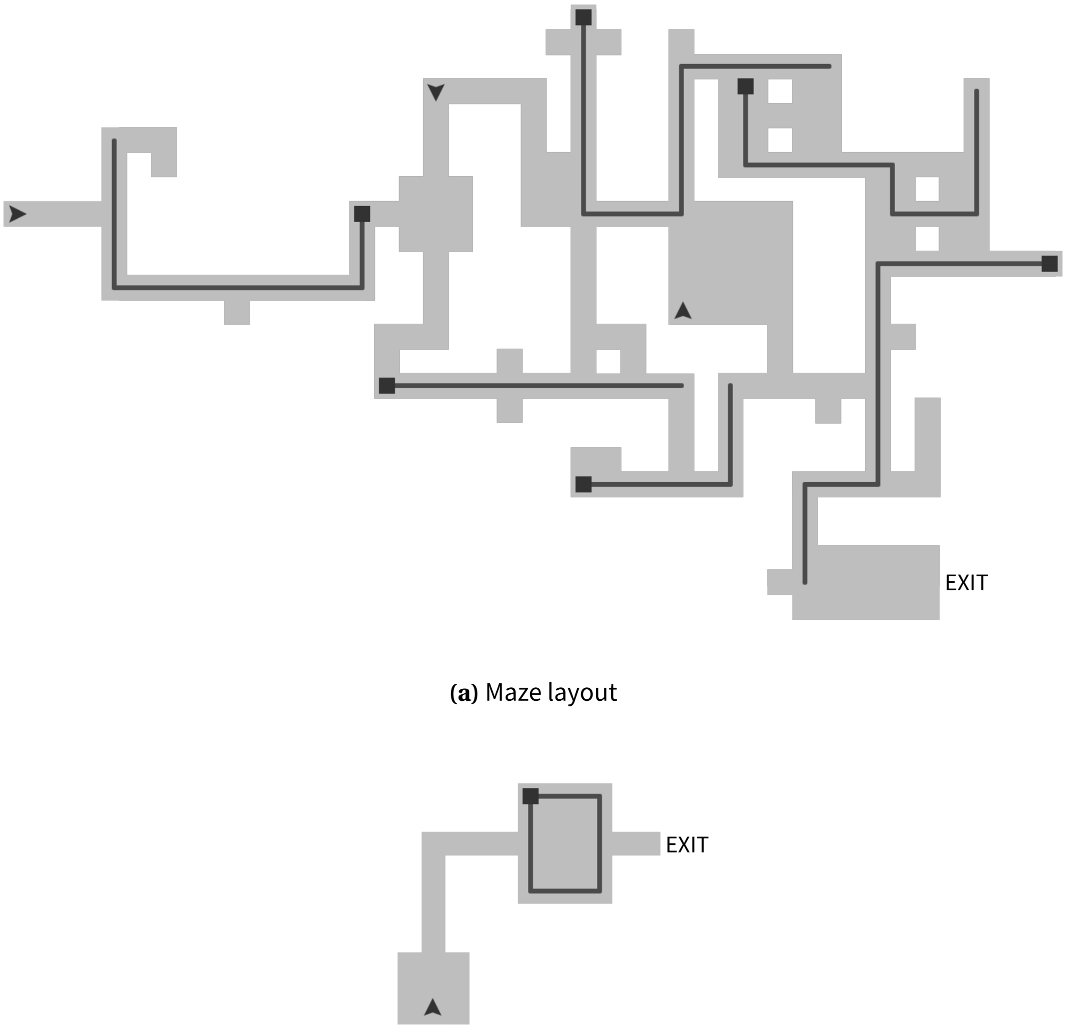
This chapter first provides a more detailed overview of *Escape Point*,[1] then describes how it was configured to control the music generator.

4.1 *Escape Point* overview

In *Escape Point*, the player has five minutes to guide a character (the "player character") through a large 3-D maze in an attempt to find the exit. A map of the maze is shown in Figure 4.1a. As the map shows, the player character begins the game in one of three possible locations, which helps to add novelty when replaying the game. The maze is populated by several enemies which move along predefined patrol paths. Although they have no awareness of the player character, they move relatively fast and can be difficult to avoid due to the placement of their patrol paths. If an enemy comes into contact with the player character, the character "dies" and the game is restarted from the beginning (except for the five-minute clock). A countdown timer appears on-screen for the final fifteen seconds if the game has not been completed by then. The game ends when the player finds the exit or the timer runs out.

Escape Point has the look and feel of a dark, horror-like science fiction game. The maze has glowing red wall panels arranged into right angles, and the enemies are large, hovering cubes that rotate and make futuristic humming sounds as they move. Figure 4.2 provides two screenshots from the game, one showing a more open area of the maze, and one showing a more confined area with a close-up of one of the enemy cubes. The game uses the first-person perspective, meaning the graphics show the maze from the perspective of the player character. The controls are standard among first-person computer games: moving the mouse rotates the character to the left or right and tilts the character's head up and down, while pressing the *W*, *S*, *A*, or *D* keys on the keyboard causes the character to run forward, backward, left, or right, respectively. The player can also

[1] *Escape Point's* Unity project files and source code can be downloaded from http://oro.open.ac.uk/view/person/ap22536.html



(a) Maze layout

(b) Warm-up layout

Figure 4.1: Layouts of *Escape Point's* maze and the warm-up version. Arrows indicate possible starting locations and directions, while squares and their adjacent lines indicate enemies and their patrol paths.

hear the enemies' humming sounds from a first-person perspective—that is, the volume and panning of the sounds change to reflect the enemies' locations in relation to that of the player character. Thus, the player can hear not only how close an enemy is, but also roughly which direction it is in relation to the player character. With the exception of the music, which will be discussed in the following section, the only other sounds in the game are those of the player character's own footsteps while running.

In addition to the main part of *Escape Point*, a short warm-up sequence was also created in order to help familiarize new players with the controls and game mechanics, especially players unfamiliar with first-person games. Many modern games begin with an introductory sequence or task for this purpose, and it was particularly important for the study described in Chapter 6. The warm-up is essentially a simplified version of the game, with one enemy, a much smaller layout (shown in Figure 4.1b), and no music. It involves the player moving forward and around a corner, across a room containing the enemy, and through a final hallway to the exit. As in the regular version of the game, the warm-up restarts if the enemy overtakes the player character, and it ends when the exit is reached. There is no time limit, however.

4.2 Music design

One of the main goals for the music generator was for it to be able to dynamically support a game's emotional narrative. While the design of *Escape Point* does not encompass a "story" in the traditional sense, there is nonetheless a narrative implied in the player's course through the maze. In particular, the game becomes increasingly tense when the player character nears one of the enemies, which are not only ominous in appearance, but also the most prominent obstacles to completing the game. Nearness to an enemy represents both a fictional danger to the player character and an increased likelihood that the player will lose any progress made thus far. The emotional narrative of the game was therefore defined for musical purposes as the changing amount of tension arising from the proximity of the player character to the nearest enemy.

To musically reflect the changing amount of tension, appropriate parameter sets were needed to represent two endpoints of a continuum. The game also needed to be

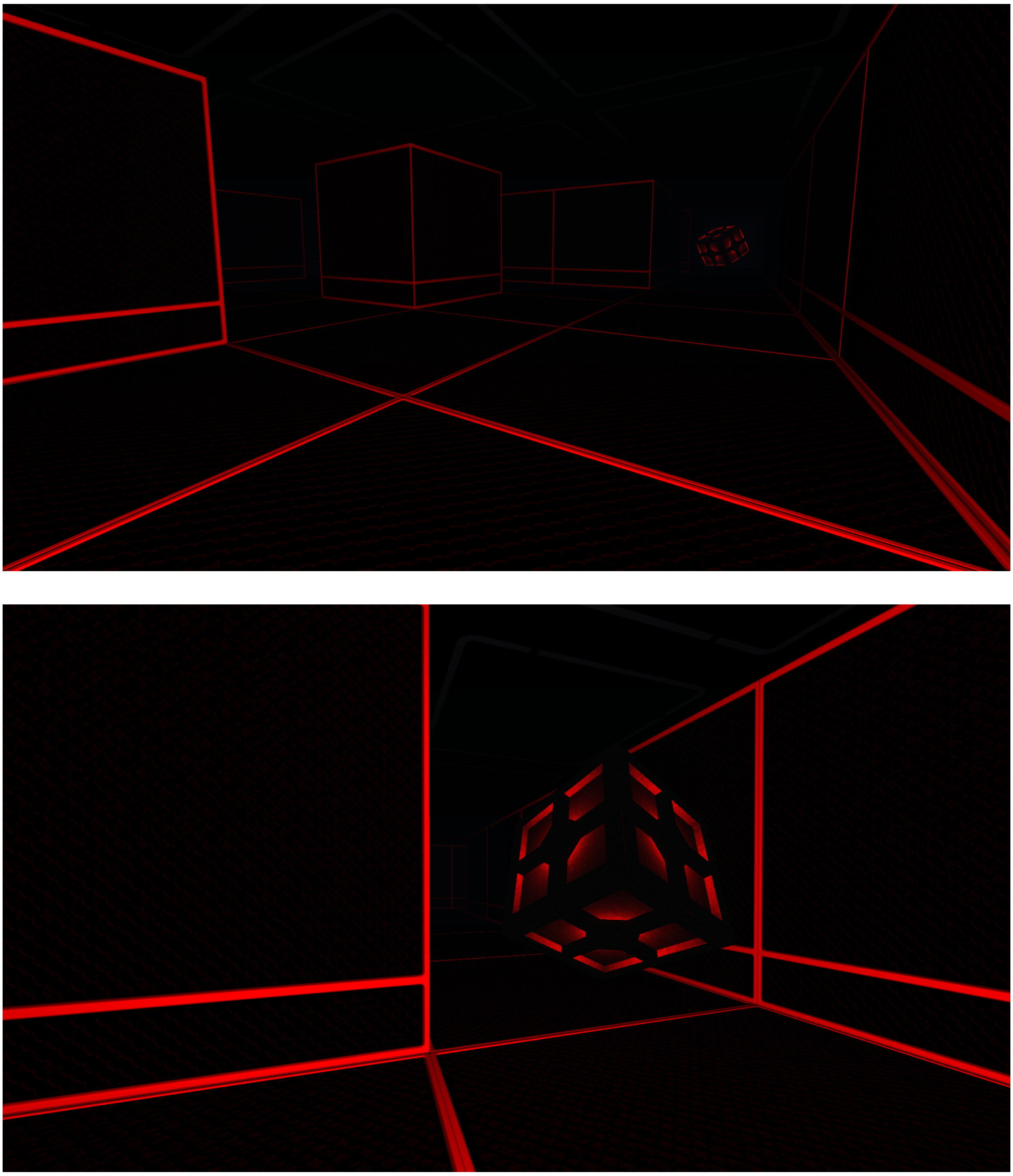


Figure 4.2: Two screenshots of *Escape Point*

configured to interpolate between the two parameter sets in accordance with player–enemy proximity, and update the music generator accordingly. The following subsections detail each of these steps in turn.

4.2.1 Characterizing musical tension

As discussed in Chapter 3, the development of the music generator was guided primarily by consideration of the valence/arousal model of emotion (Russell, 1980). Although this model does not explicitly account for tension, sometimes emotional arousal is divided into *energy arousal* and *tension arousal*, either as a two-dimensional model or together with valence as a three-dimensional model. However, Eerola and Vuoskoski (2011) found that tension arousal correlated significantly with both valence (negatively) and energy arousal (positively) in perceived emotion ratings of film music excerpts, and that the three dimensions could be reduced to two (valence and arousal) without significantly sacrificing goodness of fit. Musical tension in *Escape Point* was therefore assumed to be representable in the valence/arousal model, with an increase in tension being the equivalent of a decrease in valence and an increase in arousal.

Table 4.1 shows the two parameter sets representing low and high tension. The *low tension* set, which is intended for when the player character is a safe distance away from the enemies, specifies music that is consonant and quiet, with diatonic chords and tonal transitions, and low values for *tempo*, *volume*, *velocity*, and *timbral intensity*. The minor mode is favoured over major, since even in its least tense moments the game is still rather ominous. In the *high tension* set, which is intended for when the player character is very close to an enemy, the music is more dissonant and loud—non-tonal chords and chord transitions are allowed, the tempo is doubled, and the *volume*, *velocity*, and *timbral intensity* parameters are high. The *pulse volume* parameter is also at its maximum, so the player can clearly hear a pulsing bass note reminiscent of a heartbeat. The voice leading parameters (not shown) are left at their default settings in both parameter sets.

4.2.2 Controlling the music generator

To control the music generator, one of the main game logic classes periodically calculates the current amount of tension in the narrative, interpolates accordingly between the two

Table 4.1: *Escape Point's* parameter sets for the music generator

Parameter	Low Tension	High Tension
More major chords	0.3	0.1
More minor chords	1	1
More dominant chords	0.5	0.5
More diminished chords	0.5	1
More tonal	1	0.2
More major rules	0	0
More diatonic	1	0
Tempo	60 BPM	120 BPM
Volume	0.3	0.6
Velocity	0.4	1
Timbral intensity	0	1
Pulse volume	0	1

parameter sets shown in Table 4.1, and sends the interpolated set to the music generator. This is done at a rate of twenty times per second to ensure that gradual changes in narrative tension are reflected smoothly in the music. The same class also fades the music generator in and out at the beginning and end of the game, as well as whenever the player character dies and the game restarts.

As noted previously, the amount of tension is determined by the distance between the player character and the nearest enemy. Specifically, it is expressed as a value between 0 and 1, where 0 indicates that the distance is greater than or equal to a specified maximum distance to take into consideration—a constant of 40 metres was used—and a value of 1 indicates that the player character is colliding with the nearest enemy. It is calculated using the C# code:

```
float playerEnemyDistance = [ ... ];
float clippedDistance = min(playerEnemyDistance, maxDistance);
tension = 1.0f - (clippedDistance / maxDistance);
```

For example, if the distance between the player and the nearest enemy is 4 metres, `tension` would be calculated as $1-(4/40)=0.9$, indicating a relatively high amount of tension.

The resulting `tension` value is used to linearly interpolate between the low and high tension parameter sets, parameter by parameter. The interpolated parameter set is then sent via a UDP connection to the music generator, which updates its input parameters accordingly.

4.3 Chapter summary

This chapter provided an overview of *Escape Point*, a simple but representative computer game in which to test the music generator. Two sets of the generator's input parameters were created for the game: one that represents low emotional tension with consonant and quiet musical features, and one that represents high emotional tension with dissonant and loud musical features. At periodic intervals, the game's internal logic determines the amount of tension in the narrative, interpolates accordingly between the two parameter sets, and sends the resulting set to the music generator. Thus, the emotional expression of the music continuously adapts to reflect the amount of tension in the narrative, changing as smoothly or as sharply as needed.

CHAPTER **5**

METHODOLOGIES FOR EVALUATING GAME MUSIC SYSTEMS

Once the music generator was configured to reflect *Escape Point's* narrative, the next step was to empirically evaluate the approach. However, currently there is no commonly accepted methodology for evaluating game music systems, and in fact very little of the existing literature on game music has included an evaluation component at all. For example, none of the game music systems reviewed in Section 2.2 have been evaluated thus far. Since the field is nonetheless clearly maturing, it would be worthwhile to start thinking about how game music systems could be evaluated, especially under a set of common principles. This would enable researchers to better understand what works or does not work not only in an individual system, but, over time, in game music in general.

The absence of a standard approach to evaluating game music systems is likely due to a number of factors. Perhaps most important is that, barring commercial sales, it is not immediately obvious how the overall success of a game could be clearly measured, let alone that of an individual component like the music. Further complicating the matter is that music can serve different purposes in games, and game music systems are often designed with unique motivations that could warrant project-specific metrics. Finally, factors like a project's current stage of development might impose different restrictions on the kind of evaluation that could reasonably be carried out at a given time. Any single approach to evaluation is therefore unlikely to provide a comprehensive solution, and instead it would probably be more realistic to think in terms of a set of relevant approaches.

Thus, this chapter presents four broad methodological approaches to the evaluation of game music. The first two are borrowed from more general research on games; they include measuring the player's enjoyment or some other indication of the quality of the overall experience (Section 5.1.1), and measuring the player's psychophysiology during gameplay (Section 5.1.2). These are referred to as *player-oriented* approaches since they primarily focus on the player's experience. The other two approaches are borrowed from research on computer music systems outside the context of games; these include measuring the aesthetic quality or stylistic plausibility of the music (Section 5.2.1), and measuring how well it conveys a particular emotion or narrative (Section 5.2.2). These are referred to as *music-oriented* approaches since they focus on the intrinsic capabilities of the music system.

The following sections examine each of the four approaches in turn, characterizing them by a set of generalized research questions and reviewing and comparing specific methodologies that could be used to pursue them. My goal in this chapter is mainly to contextualize the evaluative study presented in Chapter 6, as well as to provide a theoretical and practical foundation for evaluation in future game music research.

5.1 Player-oriented approaches

As previously mentioned, it is not immediately obvious how the overall success of a game could be clearly measured, at least not prior to its commercial release. Traditional user experience metrics tend to focus on usability—for example, the speed or accuracy at which users can carry out certain tasks with a given interface. However, IJsselsteijn et al. (2007) argue that although usability is an important factor in games, it functions more as a "gatekeeper" for the quality of the overall experience, which itself is usually the central motivation for people to play games. In other words, while poor usability can certainly harm a gaming experience, strong usability will not necessarily guarantee a good one. Under this assumption, experience-based metrics for game evaluation would probably benefit from a broader interpretation of the experience.

In recent years, game researchers have measured different aspects of game playing experiences using several interesting methodologies, many of which could be relevant to game music research. These loosely fall into two categories: some aim to either directly or indirectly measure players' enjoyment, typically through the use of subjective questionnaires or interviews, while others focus on measuring players' psychophysiology during gameplay as an indication of their emotional responses. In the context of game music evaluation, the adoption of one of these *player-oriented* approaches would reflect an assumption that the success of the music could be determined by its impact on the player's experience. In the first case this would be an increase in the player's overall enjoyment, preference, or some other dimension of the experience taken to be indicative of the overall quality of the game. In the second case it would be for the music to emotionally affect the player in a certain controllable way as intended by the game or music designer—for example, to make the player feel more tense at certain times in the game. Accordingly, the approaches could be characterized by the following more general research questions, respectively:

- Does the music lead to a more enjoyable or otherwise better game experience?
- Does the music affect the player in the intended way during gameplay?

Perhaps the main drawback of player-oriented approaches compared to music-oriented ones (described in Section 5.2) is that the music system would actually need to be implemented in a game by the time of the evaluation. This means that both the music system and the game would need to be in a fully working state, which might only be feasible in later stages of development. Additionally, the evaluation would then be tied to that particular game, which might not necessarily be optimal for showcasing the music system's functionality. At the same time, a player-oriented approach could have much greater ecological validity than a music-oriented approach since by nature it would involve actual gameplay.

5.1.1 Player enjoyment and the overall experience

Player enjoyment is probably one of the most important considerations in game design. It would certainly be a strong indication of success if the music somehow led to an increase in player enjoyment. Simply asking participants how much they enjoyed a particular game condition, or which one they preferred, is certainly not out of the question, and has been done by Klimmt et al. (2007) and Weibel et al. (2008), for example. However, it could be difficult to encode the enjoyment of a gaming experience in such a way that is clear to participants yet also conducive to rigorous analysis. Additionally, the differences between experimental conditions could be subtle enough that a participant would not consciously notice them, or would have difficulty discerning a clear preference. Perhaps as a result, many game researchers have instead focused on other aspects of the game playing experience which are more easily or clearly measured, but still related to enjoyment or otherwise indicative of the overall quality of the game.

To date, the most common game enjoyment metrics relate to the concept of *flow*, first proposed by psychologist Mihalyi Csikszentmihalyi (1990). According to Csikszentmihalyi, flow is an "optimal" psychological state of strong enjoyment that is characterized by deep concentration and a loss of awareness of oneself and time. He notes that flow tends to occur during the performance of a task when a set of specific conditions are met. These include, for example, that the task's difficulty matches the person's skill level, and that the task has clear goals and provides immediate feedback. Flow has been empirically evaluated and measured in a variety of ways—an overview is provided by Moneta (2012).

Sweetser and Wyeth (2005) proposed one of the earliest formal strategies for evaluating game player enjoyment, named *GameFlow* and based entirely on flow. They brought Csikszentmihalyi's conditions for the occurrence of flow into the domain of game design, and later described a collection of 165 heuristics for how a game could satisfy the conditions (Sweetser et al., 2012). They argue that a game that can satisfy the flow conditions will be more likely to induce flow than one that does not, and therefore be more enjoyable to play. Although GameFlow does not encompass a method for identifying the actual occurrence of flow in game players, it could nonetheless be useful as a basis for heuristic evaluation.

The Game Experience Questionnaire, developed by IJsselsteijn et al. (2007),[1] aims

[1] Although discussed in IJsselsteijn et al.'s paper, at present the questionnaire itself remains unpublished. However, it can be obtained by contacting the Game Experience Lab at the Eindhoven University of Technology.

to actually measure flow in game players as well as the related dimensions of immersion, positive and negative affect, tension, and challenge, using a set of Likert scales. The questionnaire is available in both a full-length version designed to be completed after playing a game, and a condensed version designed to be completed in the middle of a game. These have since been used in several studies (e.g., Gajadhar et al., 2008; Nacke and Lindley, 2008a,b; Nacke et al., 2010) to assess the effects of different game conditions on flow and the other dimensions. Brockmyer et al. (2009) take a similar approach with their Game Engagement Questionnaire, in which they examine flow, presence, and absorption through several Likert-type items. The questionnaire was designed specifically to assess people's engagement with violent computer games, although the end result is similar in both content and format to the Game Experience Questionnaire. A comparison of the two questionnaires is presented by Norman (2013). It is worth noting that *presence*—the feeling of actually being in the fictional game world—may be a useful metric in its own right for game music. A discussion of presence as related to other concepts (involvement and immersion), as well as a questionnaire that aims to measure presence in virtual reality environments, is presented by Witmer and Singer (1998).

The applicability of these general, experience-based questionnaires to game music evaluation would largely depend on the availability and relevance of multiple conditions, as otherwise there would be no way to isolate the effects of the music on the experience. One potential alternative would be to have a single condition, but allow the participants to determine whether and how the music contributed to their experience. This could be achieved using a more targeted questionnaire or an interview, for example. Paterson et al. (2010) took this approach to evaluate the sound design in an augmented reality game. They used a questionnaire with both open-ended, free-response questions (e.g., "Which part of the game was immersive and why?"), and more specific Likert scales (e.g., "The sound made the game feel scary"). This gave the participants the freedom to devise and articulate their own opinions about the game, while still directing their attention to the specific items of interest to the researchers. An important drawback to an approach like this, however, is that it would rely almost exclusively on participants' subjective opinions. Whereas the Game Experience Questionnaire and the others mentioned above involve participants reporting aspects of their experience, this approach would additionally rely on them to speculate about what contributed to it. This is essentially a second layer of subjectivity that could conceivably be highly personal and prone to individual bias.

Of course, questionnaires and interviews are subjective in general, and using them to examine enjoyment, flow, and related aspects of a gaming experience could have its own limitations. Perhaps most important is that the psychological processes of interest may operate largely in the subconscious. For example, as previously mentioned, Csikszentmihalyi (1990) characterizes the state of flow in part by a loss of awareness of oneself. Under this assumption, asking participants to consciously reflect on a gaming experience, possibly including a state of flow, might not lead to an accurate representation. Nonetheless, these approaches have been used successfully by many game researchers, and arguably offer an important perspective for evaluation.

5.1.2 Player psychophysiology

Psychophysiological methods have become increasingly common in game research over the past decade. These methods involve the study of psychological phenomena through the analysis of certain physiological signals. For example, skin conductance has been shown to be closely associated with emotional arousal (Peterson and Jung, 1907; Lang, 1995; Dawson et al., 2007), so measures of players' skin conductance could provide a good, objective estimate of their arousal over time. Common psychophysiological measures in game research include skin conductance, heart rate variability, facial muscle tension, and a few others. A comprehensive review of their history and practical use in empirical research is provided by Cacioppo et al. (2007).

Kivikangas et al. (2011) note three main advantages of psychophysiological measures in the context of game research: First, since they are based on mostly involuntary responses, they are objective and unaffected by participant bias, limitations of the participant's memory, or limitations of the participant's ability to accurately reflect on the experience. The latter is of particular importance here since presently it is not actually clear at what level of consciousness game music primarily operates. Second, data can be recorded in real time, without disturbing the player and potentially disrupting an important psychological state. Third, psychophysiological measures are usually sensitive enough to reveal even very subtle responses. Another advantage worth mentioning is that they allow responses to be measured over time, so that they can be analyzed in relation to specific game or musical events of interest.

In one of the first game studies to utilize psychophysiological measures, Mandryk et al. (2006) examined correlations between players' subjective responses to a game and their average skin conductance, heart rate, respiration amplitude and rate, and facial muscle tension. They found several statistically significant correlations between the subjective and psychophysiological responses—for example, self-reported "fun" was positively correlated with average skin conductance (that is, a participant's mean skin conductance for a given condition). This finding was later supported by Tognetti et al. (2010), who modeled player preferences for different game conditions based on their psychophysiological responses.

It is perhaps unsurprising that averaged psychophysiological responses would be positively correlated with player enjoyment and preference. A stronger overall psychophysiological response would generally indicate a stronger overall emotional response, which in turn would suggest a more enjoyable experience. However, Nacke et al. (2010) examined the effects of sound (on or off) and music (on or off) in a game on players' subjective and averaged psychophysiological responses, and found significant effects on the subjective responses but not the psychophysiological responses. They go on to suggest that averaging psychophysiological data might be an imperfect approach for complex stimuli like games. The main alternative is to examine changes in psychophysiology over time—for example, in response to specific game events. Taking this approach, Ravaja et al. (2006) analyzed the effects of positive (e.g., scoring points) and negative (e.g., losing the game) game events on players' psychophysiology, and found statistically significant effects for both types of events. A similar method could be used to examine the effects of specific narrative and/or musical events to see if these effects reflect the music designer's emotional intention. For example, the music designer might want to see if the inclusion of music increases players' psychophysiological responses during fight sequences, or if players feel tense when they hear certain motifs.

The notion of an intended emotional effect is central to Mirza-babaei et al.'s (2013) idea of using Biometric Storyboards during game development. A *Biometric Storyboard* is essentially a set of graphs of psychophysiological data—one is the intended emotional response, manually drawn by a game designer, while the other is the actual response recorded from a player during gameplay. The game designer can then review the graphs in order to visualize discrepancies between the intended and actual responses, and adjust the game design accordingly, if desired. The authors posit that using Biometric Storyboards during game development can ultimately produce game designs that lead to better gaming experiences.

The use of psychophysiological measures in game music evaluation would generally reflect an assumption that the music should affect the player in a certain way. Success would therefore be measured in terms of how strongly or accurately the music does so. It certainly seems reasonable that a game developer would want to make use of a music system that has been shown to affect player psychophysiology in some controllable way. Perhaps the main limitation of the approach, however, is that psychophysiological measures can only provide insight into certain known psychological aspects of an experience—they cannot be assumed to represent the overall quality of the experience. However, for these known psychological correlates (e.g., skin conductance's association with emotional arousal; see respective chapters in Cacioppo et al., 2007 for other such associations), psychophysiological measures do offer much potential.

5.2 Music-oriented approaches

Although computer music systems have been developed for a wide range of purposes and applications, they usually share the goal of introducing novel functionality while simultaneously maintaining a certain aesthetic standard. Game music systems are no exception, and indeed many of the methods that researchers have used to empirically investigate other kinds of computer music systems could apply equally to game music. In the context of evaluation, two methodological approaches stand out as particularly relevant: measuring the aesthetic quality of the produced music, typically in terms of whether it could pass for ordinary human-composed music, and measuring the system's ability to accurately convey different emotions or a particular narrative. The main difference between these more *music-oriented* approaches and the player-oriented approaches presented in Section 5.1 is that these focus primarily on aspects of the music itself. In particular, the adoption of a music-oriented approach would reflect an assumption that the success of a game music system would be intrinsic, and able to be determined by an assessment of its musical output. These approaches could be characterized by the following more general research questions:

- Does the music sound aesthetically pleasing or stylistically plausible?
- Does the music convey the intended emotion or narrative?

One advantage of music-oriented approaches is that they do not require the music system to be implemented in an actual game. This means that they could be useful even in early stages of development, and in cases where tying the evaluation to any particular game would be undesirable. On the other hand, evaluating a game music system outside the context of a game would arguably limit the generalizability of the results.

5.2.1 Aesthetics and style conformity

Music as a form of art is closely tied with aesthetics, and game music is no exception. Aesthetics could therefore comprise a central focal point in the development and evaluation of a game music system. Indeed, perhaps the most basic requirement of any music producing system is that it reaches a certain aesthetic standard. However, the matter is somewhat complicated by the fact that music in games can provide functional value as well as aesthetic value, and the relationship between the two is not particularly well understood. For example, it is not actually clear at present whether players would need to like or even notice the music at all in order for it to improve their experience or otherwise function as intended. At the same time, regardless of any novel functionality a system may introduce, if the produced music was somehow unpleasant to listen to, it would probably be distracting and thus detract from the overall quality of the game. To simplify the matter it may be reasonable to assume that, for the purpose of evaluation, there is a minimum threshold over which the aesthetic qualities of a game music system should at least “suffice” in typical use cases.

A common goal in algorithmic music research is to be able to generate music that is indistinguishable from a given style of human composed music. The ability of a system to do so—sometimes referred to as *style conformity*—could be a good indication that it reaches a satisfactory aesthetic. Style conformity in algorithmic music has typically been evaluated using variations of the Turing test, first proposed by Alan Turing (1950). In the original Turing test, an "interrogator" uses a text-based interface to interact with two agents, one human and the other a computer program. If the interrogator cannot tell which is the human and which is the program, then the program passes the test. More recently, Pearce and Wiggins (2001) proposed a method for evaluating algorithmic music systems using a variation of the Turing test. In this method, the researcher first trains the parameters of an algorithmic music system from a corpus of music, then uses the system to generate new pieces of music, and finally has human participants listen to, and attempt to distinguish between, the training music and the generated music. If they cannot consistently distinguish the two, then the music system passes the test. Although clearly targeted at algorithmic music systems, such a test could potentially be adapted to target the more computational aspects of systems primarily using human composed music. This could include how they transition between different pieces of music, for example, which could be compared with similar, human composed transitions.

On the surface, the applicability of variations of the Turing test to the evaluation of game music may seem dubious. That is, there is no intrinsic need for game music to sound human composed. Additionally, Ariza (2009) criticizes their use in the evaluation of algorithmic music systems, noting that it would presume to measure machine intelligence and/or creativity, when in reality they are no more than "listener surveys". However, as listener surveys, they do provide a simple, objective way to tell whether the generated music demonstrates at least a façade of musical competence. In practice, responses to the question of whether the music sounds human composed would probably be informed more by the perception of the music's naturalness and coherence than the intelligence of its composer. Thus, while passing the test may not be indicative of machine intelligence in the way that Turing meant, it would suggest that the system is at least capable of following stylistic conventions, which for game music would probably be desirable.

Another promising methodological approach would be to measure people's aesthetic responses to a music system's output over time, which would allow researchers to examine individual musical events and functionality in relation to the rest of the music. For example, Madsen et al. (1993) had participants move a dial while listening to a piece of music in order to indicate their changing aesthetic response. Notably, all thirty of the participants specified in a post-study questionnaire that they felt that their movement of the dial roughly corresponded to the variations in their aesthetic experience, which suggests that the task was clear and accessible. After graphing the resulting recordings, the researchers were able to identify several consistent aesthetic "peaks" and "valleys" in the piece. An approach like this could be used to identify which functionality is more successful and which requires improvement. It could even be adapted for use during gameplay, for example by having players use a volume control for the music, or pressing a button whenever it becomes distracting.

Evaluating the aesthetics of a game music system is important because, in many ways, it could be a weak link in the system's overall impact. As mentioned above, though strong aesthetics would not necessarily guarantee the overall success of a music system, weak aesthetics would almost certainly hinder its success. At the same time, this means that success measured in other ways—for example, using one of the player-oriented approaches described in Section 5.1—could also be a good indication of the aesthetic qualities of a system. Any effort devoted to improving them beyond that point would reflect a more explicit focus on aesthetics, which would probably warrant a more individualized evaluation methodology.

5.2.2 Conveyance of emotions and narrative

In most games the music is used to support and help convey the narrative, typically by expressing the emotions that characterize narrative situations and events. The ability of a system to do so effectively and reliably would therefore be a relevant focal point in the evaluation of a game music system. As with the approach of measuring players' psychophysiology (Section 5.1.2), the underlying emphasis here is essentially controllability. In this case, however, it is about the controllability of the musical output rather than the player's experience. Specifically, if a system could be controlled to convey what the designer intends it to—whether a particular emotion, situation, or event—then this would be a strong indication of its practical value in the context of a game.

As discussed in Section 2.3, there is a large body of empirical research concerned with the relationship between music and emotions. Much of this work has involved participants listening to different pieces of music and reporting the emotions they perceive in them according to a certain model of emotion (see Eerola and Vuoskoski, 2011, for a comparison of the two main models). For game music, the concern would primarily be whether the emotion that people "hear" in the music (the *perceived emotion*) is the same as the emotion that the music was meant to convey (the *intended emotion*). Rutherford and Wiggins (2002) used this approach in their evaluation of an algorithmic game music system. They wanted to test whether varying the system's single input parameter, *scariness*, actually affected the amount of scariness that people perceived in the music, as well as the amount of tension. Accordingly, they created three audio clips of the music system's output—one with low scariness, one with medium scariness, and one with high scariness—and had participants rate how scary and tense the clips were on bipolar scales. They then analyzed whether the participants tended to rate the high scariness clip as more scary and tense than the medium scariness clip, and the medium scariness clip as more scary and tense than the low scariness clip. This provided a rough measure of how well the scariness of the music system could be controlled.

Livingstone et al. (2010) used a slightly different approach in their evaluation of a computational rule system that modifies a piece of music in order to express a given emotion, regardless of any emotions present in the original, unmodified piece. They conducted two experiments in which they played emotionally modified pieces of music, and asked participants to identify the emotion they perceived in each one. They were then able to calculate the percentage of cases in which the perceived emotion matched the intended emotion. This was roughly the approach used in the preliminary study described in Section 3.2.2, where the goal was to determine how reliably the music generator could convey different emotions. However, the preliminary study also examined emotional transitions. Specifically, the music generator was configured to transition from one emotion to another, and participants were asked to identify the perceived starting and ending emotion. Continuous response approaches like the one discussed in Section 5.2.1 (Madsen et al., 1993) could alternatively be used for a more detailed analysis of such transitions. For example, Schubert (2004) had participants move a cursor in a two-dimensional valence/arousal emotion space to indicate the changing emotions they perceived in several pieces of music. Although he was not actually comparing intended and perceived emotions as in the above examples—he was modeling the perceived emotions based on the changing musical features of the pieces—the method would nonetheless be similar.

Perhaps the main limitation of comparing intended and perceived emotions is that the perception and identification of emotions is a highly subjective task which seems particularly prone to individual bias. One potential alternative would be to take a broader interpretation of narrative that is not necessarily characterized explicitly by emotion. Wingstedt et al. (2008) used a methodology that could be adapted for this purpose. Their study was based on a piece of software they designed that allows the user to adjust several features (e.g., tempo, harmonic complexity) of a piece of music in real time via a set of graphical sliders. In the study, they showed participants three videos depicting still scenes—a dark urban area, a view of space, and a picnic by a lake—and for each one had them adjust the sliders in order to make the music best fit the scene. Although the study focused on the participants' understanding of the narrative functions of music, the methodology could potentially be adapted for game music evaluation. For example, participants could adjust a game music system's parameters so as to best fit different narrative scenes, and then rate how satisfied they are with the system's ability to do so. By circumventing the question of emotion and focusing more directly on the game narrative, this approach would arguably not only be less prone to personal bias, but would also more closely measure the actual use case.

5.3 Chapter summary

This chapter has provided an overview of four methodological approaches that could be relevant to the evaluation of game music systems. The first two, which are referred to as *player-oriented approaches*, focused on the player's experience as the main indication of the possible success of a game music system. These included measuring the quality of the player's overall experience, and measuring the player's emotional response during gameplay. The latter two, which are referred to as *music-oriented approaches*, focused more on the system's actual musical output. These included measuring the aesthetic quality of the music, and measuring how well it can convey a particular emotion or narrative. While not intended to be all-encompassing, these should serve as a basis from which to examine some of the more common motivations and design goals driving the development of these systems. Of course, in practice an evaluation could easily draw from more than one of the approaches—for example, it could involve both psychophysiological measurements of the players during gameplay, and questionnaires about their experience and the music afterwards. This was the approach used in the main study presented in the following chapter, in which the goal was to determine how the music affected different aspects of the experience of playing *Escape Point*.

Evaluation should be a critical step in the development of game music systems. In reality it can only increase the value of game music research, as without it there would be little from which could be learned. In addition to providing an indication of whether a system is successful in the achievement of its design goals, an evaluation can provide clear paths to improvement in future research, not only for the system in question but also for game music overall.

CHAPTER **6**

EMPIRICAL EVALUATION: DESIGN AND RESULTS

Chapter 5 outlined some key research questions that could guide an effective evaluation of game music, and different methodological approaches to answering them. This chapter narrows and applies those ideas, presenting the design and results of two empirical studies: a main user study and a second study with music experts. The main study primarily evaluated the music generator and its configuration in *Escape Point*. It used a repeated measures design in which participants played three versions of the game: one with the original dynamic music as described in Chapter 4, one with static, unchanging music, and one with no music. They then responded to questionnaires comparing each version to each other along six subjective dimensions: *preference/liking*, *tension*, *excitement*, *challenge*, *enemy scariness*, and *fun*. To provide context for these responses, the participants were asked whether they generally like horror games or films, since the game overall is quite tense and similar in style to some horror games. Finally, their skin conductance was recorded during each condition in order to identify potential effects of the music on their emotional arousal.

In the second study, which is described in the final section of this chapter, the approach of interpolating sets of parameters for the music generator was compared with two variations of the approach of crossfading audio clips. Music experts first listened to transitions created using the three approaches and ranked them by perceived smoothness, then listened to clips representing the midpoint of each transition and rated their expressed emotion, how discordant they were, and how listenable they were.

6.1 Main study overview

In terms of the evaluation approaches presented in Chapter 5, the main study was primarily *player oriented* in that it examined the net effects of the music on the player's experience with the game. The task of identifying specific aspects of the music to improve (a more *music-oriented* approach) had largely already been addressed in the preliminary study (described in Section 3.2.2), and was later addressed in another study (Section 6.7). In this study the intention was rather to evaluate the musical approach as a whole, specifically by comparing it to some plausible alternatives and determining which was the most effective. The two player-oriented research questions from Chapter 5 were therefore used to guide the design of the study. Adapted to the current context, these were as follows:

RQ1 Does the music make the game more fun or otherwise enjoyable?

RQ2 Does the music make the game more emotionally arousing?

The questionnaires were designed to address both research questions—RQ1 was addressed through the *preference/liking* and *fun* dimensions, while RQ2 was addressed through the *tension* and *excitement* dimensions. RQ2 was also addressed through analysis of the skin conductance measurements, particularly changes in skin conductance immediately following deaths of the player character. This was because deaths in *Escape Point* are particularly tense moments, and they occur at discrete, unambiguous times, and usually several times per game. Accordingly, the study tested three main hypotheses:

H1 Participants will prefer the dynamic music condition to the static music and no music conditions, and find it more fun.

H2 Participants will rank the dynamic music condition as more tense and exciting than the static and no music conditions.

H3 The dynamic music condition will elicit larger skin conductance responses than the static and no music conditions.

6.2 Skin conductance overview

This section reviews the physical basis, measurement, and significance of skin conductance in order to contextualize some of the data analyses and results presented later in this chapter. A more detailed overview of skin conductance can be found in the chapter *The Electrodermal System* by Dawson et al. in the *Handbook of Psychophysiology* (Cacioppo et al., 2007).

Skin conductance, also known as *electrodermal activity*, refers to the skin's ability to conduct electricity. It varies depending on the amount of sweat present in the skin, which is controlled by the body's sympathetic nervous system (the "fight-or-flight" response), a part of the autonomic nervous system. Skin conductance is therefore closely associated with physiological and emotional arousal, and is regulated involuntarily.

Skin conductance is typically measured by passing small, physically imperceptible electrical charges through the skin via a pair of electrodes, and calculating the inverse of the resistance measured between them. Although sweat glands are located all over the human body, the most common locations for electrode placement are on the palm sides of the hands and the bottoms of the feet. When possible, Dawson et al. (2007) recommend using the pads of the fingertips since they have a particularly high concentration of sweat glands, resulting in stronger readings (Freedman et al., 1994). In such cases, even very small amounts of moisture can be detected, and the subject does not need to be actively "sweating" in the conventional sense for the measurements to be accurate.

Skin conductance measurements are conventionally expressed in micro-Siemens (μS). Usually, a change in skin conductance in response to a stimulus is of particular concern, which is referred to as a *skin conductance response* (SCR), or sometimes an *electrodermal response* (EDR) or *galvanic skin response* (GSR). SCRs not elicited by a specific stimulus can also occur, and are referred to as *non-specific SCRs*. Figure 6.1 provides a visual representation of an SCR which denotes the main components. Typically, an SCR begins a few seconds after the presentation of a stimulus; this window is referred to as the *latency*.

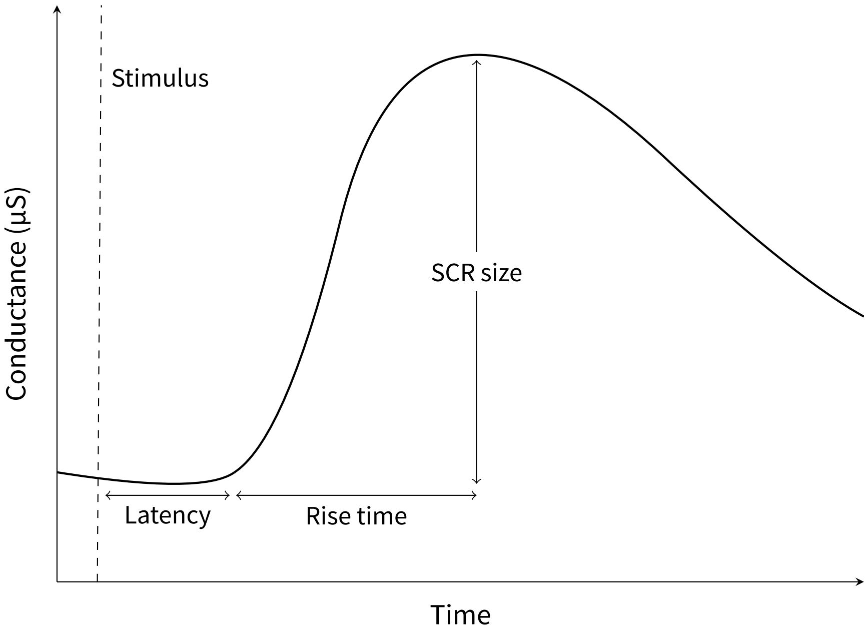


Figure 6.1: Shape and main components of a typical skin conductance response (SCR)

Conductance then increases for a few seconds—the *rise time*—until it reaches a peak and then slowly begins to fall. The *size* of the SCR, which is the most important component, is the (positive) difference between the conductance at the peak of the rise and the conductance at the onset of the rise. If there is no rise in conductance within a few seconds after the presentation of a stimulus, then the common understanding would be that no SCR occurred, not that one occurred with a zero or negative size. However, sometimes a size of zero is used to represent such cases in statistical analyses of multiple SCRs.

Although the skin conductance analysis in this study focused on SCRs, it is worth noting that another important aspect of skin conductance is the *tonic* or *background skin conductance level* (SCL). These refer to the base amount of skin conductance upon which SCRs can be considered to be "superimposed". Whereas SCRs occur in a matter of seconds, tonic SCL drifts slowly up and down over periods of up to a few minutes. It is therefore more a suitable metric for longer-term effects beyond the scope of this study, since *Escape Point* only takes about five minutes to play.

6.3 Method

The study was conducted at The Open University's main campus in Milton Keynes, UK. The following subsections detail the methodology that was used. The final subsection concludes by reviewing and clarifying some of the key design decisions that were made during the planning process.

6.3.1 Participants

Thirty-two postgraduate students and members of staff at The Open University participated in the study. They were recruited through university mailing lists, and received no financial compensation for their participation. A brief questionnaire (available in Appendix B) asked about their background and previous experience with games, including the study's main task of playing a three-dimensional game with a keyboard and mouse. The results are summarized in Table 6.1. The majority of the participants reported either liking computer or console games ($n = 18$) or otherwise having an interest in them ($n = 9$). Most also reported having played a three-dimensional game before with a keyboard and mouse ($n = 19$), or having played one with another interface such as a game controller ($n = 4$).

6.3.2 Conditions and stimuli

The stimuli used in the study were three different versions of *Escape Point* presented in a randomized order. The *dynamic music* condition was exactly as described in Chapter 4—it had music that varied in the level of tension depending on the distance between the player character and the nearest enemy. The *static music* condition was modified to ignore this distance and instead remain at a constant tension level of 15%. This level was obtained by calculating the dynamic music condition's mean tension level across approximately twenty minutes of gameplay, recorded with two participants before the study. The motivation for using this value was to experimentally control the overall level of tension between the dynamic music and static music conditions. Finally, the *no music* condition was modified to have no music at all. All three conditions included the game's sound effects—the player character's footsteps while running, and the humming sounds

Table 6.1: Participant summary ($N = 32$)

		Count	%
Gender	Male	19	59%
	Female	13	41%
Age group	18–24 years	0	0%
	25–29 years	7	22%
	30–39 years	16	50%
	40–49 years	8	25%
	50–59 years	1	3%
	60+ years	0	0%
Likes games	Yes	18	56%
	No	1	3%
	Little or no experience, but interested	9	28%
	Little or no experience, and not interested	4	13%
3-D game history	Have played with keyboard and mouse	19	59%
	Have played, but not with keyboard and mouse	4	13%
	Have only played other kinds of digital games	8	25%
	Have not played a digital game	1	3%

made by the enemies.

In addition to the musical variations, in each condition the player character was placed at a different starting location in the maze, the order of which was also randomized. As noted in Chapter 4 and shown in Figure 4.1a, there were three possible starting locations, each roughly equal in difficulty. This helped to ensure that the conditions each began with similar novelty, at least in terms of the spatial layout.

6.3.3 Collected data

Questionnaires

In the questionnaires, the participants compared the three conditions along six dimensions: *preference/liking*, *tension*, *excitement*, *challenge*, *enemy scariness*, and *fun*. Specifically, a pairwise comparison approach was used in which, for each possible pairing of the conditions, the participant specified which condition, if either, was stronger on each dimension (i.e., which one they liked more, which one was more tense, and so on). Ties were allowed so that the participants would not have to make arbitrary decisions in cases where they did not find a noticeable difference.

Thus, there were three mostly identical questionnaires: one comparing the first and second conditions, another comparing the second and third conditions, and a final one comparing the first and third conditions. The questions for each dimension were worded as follows (the names of the dimensions were not shown):

Preference/liking In terms of the music (or lack of music, if there was none), which did you like more?

Tension Which version of the game felt more tense?

Excitement Which version of the game felt more exciting?

Challenge Which version of the game felt more challenging?

Enemy scariness In which version of the game did the enemies seem more scary?

Fun Which version of the game was more fun?

Using the first questionnaire as an example, which compares the first and second conditions, the response choices were "The first version", "The second version", and "I did not notice a difference." For the *preference/liking* dimension, the latter choice was worded "I did not notice a difference, or have no preference."

Finally, because there would have been at least ten minutes between the completion of the first and third conditions, at the end of the final questionnaire there was an extra question asking how confident the participant was in those particular responses. The response choices were "Confident", "Somewhat confident", or "Not confident", with some minor clarifications for each.

The exact questionnaires, including accompanying instructions, are available in Appendix B.

Skin conductance

Participants' skin conductance was recorded using the ProComp Infiniti system by Thought Technology, Ltd., and its accompanying skin conductance sensor, the SC Flex/Pro. The sensor consists of electrodes sewn into separate velcro straps designed to be attached to the tips of two fingers. The ring and little fingers of the participants' mouse hand

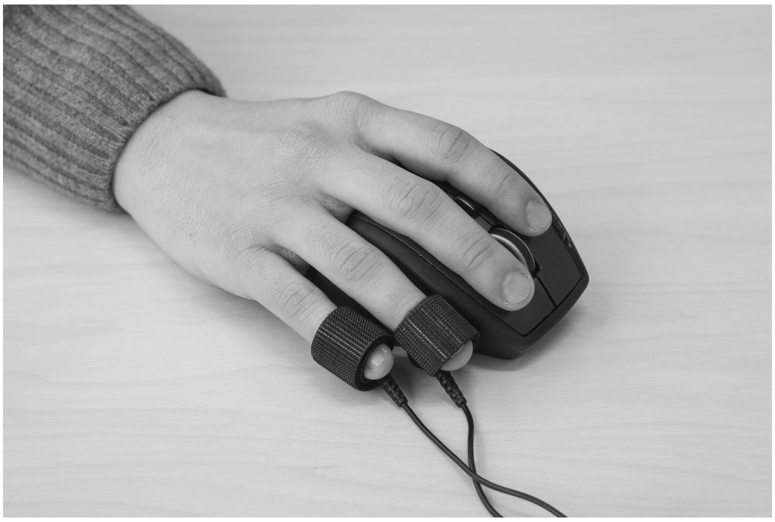


Figure 6.2: Skin conductance sensor configuration

were used because these tended to simply hang off the edge of the mouse during gameplay, thus minimizing the risk of disturbing the sensor. The configuration of the sensor is shown in Figure 6.2.

The sensor connected to a multi-channel encoder—the ProComp Infiniti itself—which in turn was connected to a separate computer (i.e., not the one the participants were using) via USB. The skin conductance signal was sampled at a rate of 256 Hz, and recorded using the software *BioGraph Infiniti*, also by Thought Technology, Ltd.

Game and music logs

Escape Point and the music generator were configured to keep extensive timestamped logs during gameplay, and save them as text files afterward. Using the logs it is possible to reconstruct various aspects of the game, including the parameter curves of the music generator, and the exact musical audio that was heard during the game. The logs include major game events such as when the player character dies, when fade-ins and fade-outs begin and end, when musical notes begin and end, and many others. They also include the curve over time of the distance between the player character and the nearest enemy, sampled at a rate of approximately 20 Hz—the same rate at which *Escape Point* updates the music generator in the dynamic music condition.

Interviews

At the end of the study there was a short videotaped interview that was mainly intended to help interpret the participants' questionnaire responses. There were two questions:

1. *Is there anything you want to clarify about your questionnaire responses?*
2. *Do you like horror games or films?*

The first question was asked mainly to ensure that the participants had understood the questionnaires, and to allow them to clarify whether any of their responses bore a non-obvious significance. However, the majority of the participants affirmed that the questions and their responses were quite clear. A few asked about the phrasing of some of the questions, but this never led to them changing, or requesting to change, any of their responses.

The second question was asked to help control for participant bias toward or against the horror genre. This was important because *Escape Point* is similar in style to some horror games, and such a bias could conceivably affect a participant's subjective responses to the different conditions. For example, one might expect people predisposed to the horror genre to generally prefer conditions considered to be more tense, and others to prefer conditions considered less tense. Thus, in the questionnaire analysis, the participants were divided into those who responded more positively to this question (the *Horror+* group; $n = 14$) and those who responded more negatively (the *Horror–* group; $n = 18$). A few participants noted that they only like certain horror games or films, or only on certain occasions. However, rather than creating an intermediary group, which would have been too small for significance testing, these were instead included in the Horror+ group on the grounds that they at least had an interest in the genre.

6.3.4 Procedure

The study was carried out in the Gaming & Future Technologies Lab at The Open University. The lab was designed to look and feel like a familiar gaming environment, with a couch, gaming consoles, game posters, a desktop computer, and a large flat screen television. Participants completed the study individually, with sessions lasting about 30–35 minutes. All sessions took place on week days between 10:00 am and 5:00 pm.

Upon arriving in the lab, the participants first signed a consent form and completed a background questionnaire, after which the premise of *Escape Point* and the study procedure were briefly described. The participants then completed *Escape Point's* warm-up, which took anywhere between thirty seconds and a few minutes, depending on how much assistance was needed. Extra time was spent with participants who initially struggled with the controls, to ensure that by the end of the warm-up they were able to navigate comfortably on their own.

The electrodes were then attached to the participants' fingers, and the skin conductance recording began. The participants then completed each of the three conditions in a random order. They completed the first questionnaire (which compared the first and second conditions) after the second condition, and the other two questionnaires (which compared the second and third conditions and the first and third conditions) after the third condition. Finally, the semi-structured interview was conducted.

During the study, the participants sat in a comfortable chair at a desk and used an HP desktop computer with a 3.1 GHz Intel Core i5 quad-core processor. Peripherals included a 19-inch Samsung SyncMaster monitor, a Logitech Wave keyboard, a Logitech MX Revolution cordless mouse, and Sony MDR-XB600 over-ear headphones. The questionnaires were administered on paper.

6.3.5 Design decisions

Participants

Participation was open to both gamers and non-gamers alike. This was in part because the distinction is not always obvious or practical to make—for example, it would not be obvious in which group to include people who used to play games, people who only rarely play games, or people who only play a particular type of game. It was also justifiable particularly because the game mechanics were relatively simple and did not require knowledge of the conventions of any particular game genre—there were no guns to reload, hidden keys to find, or anything else that might come naturally to a gamer but not a non-gamer. The only exception to this was the controls. Since the controls were standard among first-person games, those who had played first-person or similar kinds of three-dimensional games were usually able to learn the controls with little or no assistance. However, the warm-up helped to ensure that all participants had a firm grasp of the controls before proceeding to the main part of the study.

Conditions

As previously mentioned, the main purpose of the study was to evaluate the success of the overall musical approach used in *Escape Point* by determining whether it was an improvement over a few plausible baseline conditions. In the vein of a traditional control condition, an obvious choice was to also include a condition without music. This seemed reasonable because many modern games forgo music at certain times, and most provide the option to disable the music altogether; thus, in general there is no real reason to assume that including music would necessarily be preferable to not including it.

Another plausible baseline condition was one that would broadly represent conventional, non-dynamic music. Although music in modern games is quite diverse, it is fairly common to simply have an ambient music track playing in the background, more to set an appropriate tone than to follow the actual events of the game. The static music condition was intended to represent this approach. The use of a pre-composed piece of music was initially considered for this condition, but an alternative that provided greater experimental control was to simply use the music generator with its parameters kept constant. The main advantage of this approach is that the same musical style and synthesizer was used as for the dynamic music condition, and the only difference between the dynamic music and the static music was that the former varied in tension while the latter did not. This minimized confounding that could have arisen if composed music was used.

Questionnaires

The pairwise comparison approach used in the questionnaires was selected over Likert scales and other kinds of rating scales mainly because it was more closely aligned with the research questions of the study. Additionally, by having the participants compare the conditions to each other along each dimension rather than rate them individually on arbitrary scales, the questions were less prone to personal interpretation and bias. This was particularly important because the participants had diverse backgrounds. For example, a game player who normally plays high-budget, professionally developed games might not have considered *any* of the conditions fun by comparison, but could still have been able to formulate preferences between them. With the pairwise comparison approach, the intention of examining differences (if any) between the conditions was more clear.

6.4 Data analysis

The following subsections detail the statistical analyses that were performed on the skin conductance data and questionnaire responses. The analyses were performed mainly in Python using the *pandas* data analysis library (McKinney, 2011), as well as in R.

6.4.1 Questionnaire analysis

Analyzing the questionnaire data mainly consisted of converting the comparisons between conditions to rankings, then testing for consistency in the rankings. This was done separately for all six dimensions. Additionally, the analyses were performed three times: once for only the participants who said they liked horror games or films, once for only those who said they did not, and once for all participants.

A simple scoring system was used to convert a given participant's comparisons between conditions to a set of rankings for each condition: For each comparison, the condition that was rated higher received one point, while the other received no points. In the event of a tie, both conditions received half of a point. For example, if the participant rated dynamic music as more tense than both static music and no music, and the latter two equally tense, then dynamic music would have received two points, and static music and no music half a point each. These scores were then reordered into fractional rankings between 1 and 3, where 1 was highest (i.e., "first place") and 3 was lowest. Thus, in the previous example, dynamic music would have received a ranking of 1 for tension, while static and no music would have received rankings of 2.5.

Once the rankings were computed, a Friedman test was performed on each dimension to determine whether any of the conditions were ranked consistently higher or lower than any others. The Friedman test is a non-parametric version of the repeated measures ANOVA, and is appropriate for ranked data (in fact, it orders data by rank internally). In cases where the test did not return a statistically significant *p*-value, the null hypothesis that the rankings were not consistently different could not be rejected, and thus no further analysis was pursued. However, if the returned *p*-value was significant ($p < 0.05$), post hoc analysis was performed in order to determine which rankings in particular consistently differed. Because multiple statistical comparisons were being performed (which increases the likelihood of a Type I error) the *p*-values were adjusted using the Benjamini-Hochberg method (Benjamini and Hochberg, 1995), which controls the false discovery rate.

Post-hoc analysis involved pairwise comparisons of the conditions using the Wilcoxon signed-rank test, a non-parametric version of the paired *t*-test. Thus, one test would compare dynamic music and static music, another would compare dynamic music and no music, and a third would compare static music and no music. The *p*-values resulting from these tests were also adjusted for multiple comparisons using the Benjamini-Hochberg method.

6.4.2 Skin conductance analysis

The skin conductance analysis consisted of calculating the sizes of the SCRs at the times of the player character's deaths, then determining whether and how the SCR sizes differed between conditions using a generalized linear mixed model. As mentioned previously, deaths of the player character are particularly tense moments in *Escape Point*, and they occur at discrete times automatically logged by the game.

A Python script was first written which would search each game log for the times of the player character's deaths, read in the corresponding skin conductance files, and compute the sizes of the SCRs at those times. The size of an SCR was calculated as the maximum skin conductance within five seconds after the time of death minus the minimum skin conductance within one second before the time of death. Sizes of zero were allowed, indicating that no SCR had occurred. The script then arranged the SCRs in the "stacked" format (one row per SCR) with columns for the SCR size, participant, music condition, and the order of the condition.

The next step was to model the data to determine whether the music condition or the order of the condition had a significant effect on SCR size. Ordinary linear regression was initially considered, but the data violated some key assumptions of linear models. First, it violated the assumption of independent observations. This was because individual participants had multiple SCRs in all three conditions, and a given SCR could not be considered independent from other SCRs from the same participant. Second, it violated the assumption of normally distributed errors. This was true even after logarithmic and square root transformations were applied to the SCR sizes, which are sometimes recommended in skin conductance analyses. Finally, it violated the assumption of constant variance. In particular, the magnitude of the errors increased as SCR size increased. A *generalized linear mixed model* (GLMM) was ultimately used since they elegantly address these concerns. GLMMs are a type of regression model that combine the features of mixed models with those of generalized linear models, both of which are extensions of the ordinary linear model. As will be explained below, both sets of features were necessary to appropriately model the data.

Mixed models are similar to ordinary linear models in that the response variable (SCR size) is modeled as a linear combination of a set of predictor variables (music condition and order) and an error term. However, mixed models break down the error term into variance between specified sub-groups (in this case, individual participants) of the observations, and variance within them. They are normally useful when the variance between sub-groups should be accounted for—as is the case with repeated measures studies—but is not of primary interest. This was true for the SCR data since different participants had vastly different ranges of SCRs, yet the focus was on the effects of the music condition and the order of the condition, not on differences between participants. Nonetheless, accounting for such differences improves the reliability of the inferences that can be made using the model. Additionally, it effectively resolves the issue of non-independent observations mentioned above. A more detailed overview of mixed models in the context of psychophysiological research is presented by Bagiella et al. (2000).

Generalized linear models (GLMs) extend the ordinary linear model in two ways. First, they allow the response variable to be related to the linear predictor via a given *link function*, which is similar in principle to a transformation of the response variable. Second, they allow the specification of an arbitrary probability distribution of the response variable. If the specified distribution is a member of the exponential family (which is usually the case), then an additional benefit of GLMs is that the amount of variance in the observations can be expressed as a function of their predicted value. As an example, it is worth considering that ordinary linear models are in fact special cases of GLMs with an identity link function and normal probability distribution, and that their assumption of constant variance arises from the fact that normal distributions are characterized by constant variance. By using a different link function and probability distribution, GLMs can effectively resolve the issues of non-normally distributed errors and non-constant variance mentioned above. A more detailed overview of GLMs is provided by Fox (2008).

A generalized linear mixed model (GLMM) is simply the combination of a mixed model and a generalized linear model. The `glmer()` function of the *lme4*[1] R package (Bates et al., 2015) was used to model the SCR data. The participants were specified as the varying subgroups (the mixed model aspect), and the inverse square link function and inverse Gaussian probability distribution were used (the generalized linear model aspect). The inverse Gaussian distribution is appropriate when the variance increases rapidly with the predicted values (Fox, 2008), which is true in this case, as mentioned above. The inverse square is the inverse Gaussian distribution's so-called *canonical link function*, the use of which has some mathematical and practical advantages, particularly that the mean of the probability distribution can be expressed as a function of the predicted value. A gamma distribution was also tested, which is appropriate when the variance increases linearly with the predicted values (Fox, 2008), as well as other sensible link functions. However, the combination of the inverse Gaussian probability distribution and the inverse square link function resulted in the lowest Akaike information criterion (Akaike, 1974; lower is better), which has been recommended for model selection of GLMs, for example, by Lindsey and Jones (1998).

6.5 Results

The results from the questionnaire analysis are presented in Table 6.2. A concise visual representation is provided in Figure 6.3. Across all participants, Friedman tests indicated statistically significant ranking differences between conditions for the *tension*, *excite-*

[1] Version 1.1-7

ment, and *enemy scariness* dimensions. Among the participants who said they did not enjoy horror games or films ("H–" in Table 6.2), there were significantly different rankings for *tension*, *excitement*, and *enemy scariness*. Finally, among the participants who said they did enjoy horror ("H+"), there were significantly different rankings for *preference/liking*, *tension*, *excitement*, and *fun*. Post hoc analyses revealed that in all of the above cases, the statistically significant distinction was that dynamic music was ranked higher than either static music or both static music and no music. The only dimension for which there were no statistically significant findings was *challenge*.

Table 6.2: Analysis of rankings by condition and dimension

								*Post-hoc analyses**								
		Mean ranks			Friedman test			Wilcoxon NM/SM			Wilcoxon NM/DM			Wilcoxon SM/DM		
		NM	SM	DM	χ^2	*p*	Adj. *p*	*T*	*p*	Adj. *p*	*T*	*p*	Adj. *p*	*T*	*p*	Adj. *p*
Preference	H-	2.06	1.92	2.03	0.2	0.904	0.904	-	-	-	-	-	-	-	-	-
	H+	2.64	1.93	1.43	11.23	**0.004**	**0.008**	12	**0.026**	**0.042**	7.5	**0.004**	**0.01**	22.5	0.174	0.237
	All	2.31	1.92	1.77	5.37	0.068	0.106	-	-	-	-	-	-	-	-	-
Tension	H-	2.47	2.42	1.11	22.96	**<0.001**	**<0.001**	64	0.819	0.877	4	**<0.001**	**0.002**	0	**<0.001**	**0.001**
	H+	2.5	2.18	1.32	11.41	**0.003**	**0.008**	31	0.276	0.345	11	**0.012**	**0.023**	5.5	**0.005**	**0.011**
	All	2.48	2.31	1.2	33.58	**<0.001**	**<0.001**	178.5	0.354	0.425	32.5	**<0.001**	**<0.001**	10.5	**<0.001**	**<0.001**
Excitement	H-	2.33	2.25	1.42	10.09	**0.006**	**0.013**	60.5	0.683	0.759	18.5	**0.009**	**0.017**	25.5	**0.014**	**0.024**
	H+	2.64	2.21	1.14	16.71	**<0.001**	**<0.001**	30	0.109	0.155	3	**0.001**	**0.004**	12	**0.008**	**0.017**
	All	2.47	2.23	1.3	25.82	**<0.001**	**<0.001**	177.5	0.218	0.285	38	**<0.001**	**<0.001**	74	**<0.001**	**0.002**
Challenge	H-	2.22	2.06	1.72	2.9	0.235	0.317	-	-	-	-	-	-	-	-	-
	H+	2.32	1.89	1.79	2.8	0.247	0.317	-	-	-	-	-	-	-	-	-
	All	2.27	1.98	1.75	5.3	0.071	0.106	-	-	-	-	-	-	-	-	-
Enemy scariness	H-	2.33	2.44	1.22	17.94	**<0.001**	**<0.001**	52	0.629	0.725	12	**0.002**	**0.005**	0	**<0.001**	**0.001**
	H+	2.21	2.04	1.75	2.46	0.293	0.351	-	-	-	-	-	-	-	-	-
	All	2.28	2.27	1.45	18.2	**<0.001**	**<0.001**	147	0.927	0.927	50.5	**0.001**	**0.004**	31.5	**<0.001**	**0.001**
Fun	H-	1.94	1.94	2.11	0.43	0.807	0.855	-	-	-	-	-	-	-	-	-
	H+	2.25	2.29	1.46	6.26	**0.044**	0.079	44	0.912	0.927	23.5	0.063	0.095	15	**0.027**	**0.042**
	All	2.08	2.09	1.83	1.65	0.437	0.492	-	-	-	-	-	-	-	-	-

*Post hoc analyses were only performed if the respective Friedman test returned a significant *p*-value.

Regarding the question about confidence in the third questionnaire, only one participant responded "Not confident". Of the others, twelve responded "Confident" and nineteen responded "Somewhat confident". In order to determine whether there were any major differences between the latter groups, their mean condition rankings for each dimension were compared (similar to the bar graphs shown in Figure 6.3, except with "Confident" and "Somewhat confident" instead of "Horror–", "Horror+", and "Overall"). The mean discrepancy in rankings was only 0.16, and the largest was 0.36. Thus, there

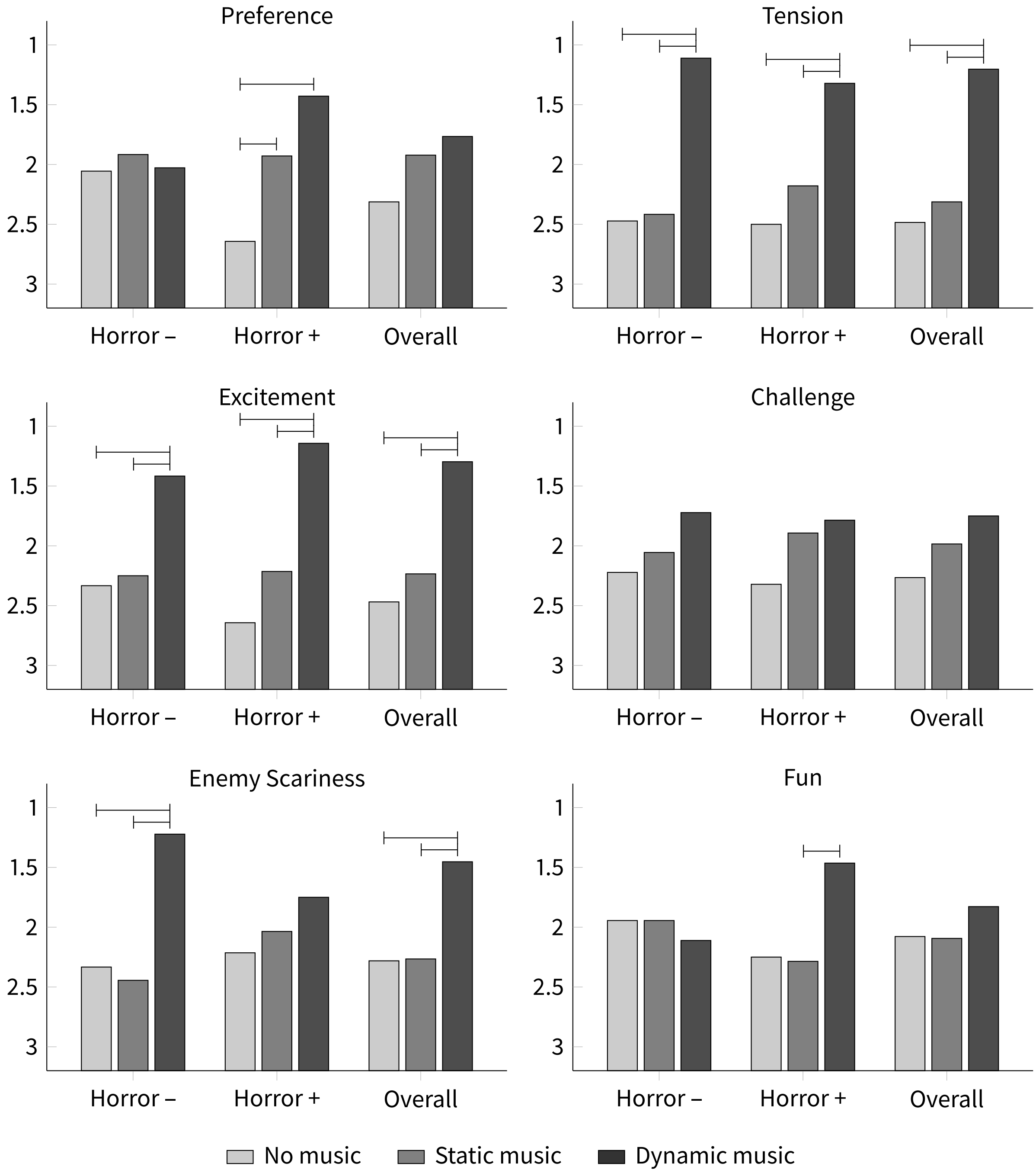


Figure 6.3: Mean rankings for each questionnaire dimension. Horizontal connectors indicate statistically significant ($p < 0.05$) differences, calculated using the Wilcoxon signed-rank test and corrected for multiple comparisons. There were eighteen participants in the Horror– participant group and fourteen in the Horror+ group.

was relatively little difference in how these groups responded.

In total there were 558 deaths of the player character and corresponding SCRs, with a mean of 17.4 per participant and 5.8 per individual condition. Table 6.3 summarizes the SCRs by condition. The mean SCR size was much larger in the dynamic music condition than in the other two, for which the mean sizes were relatively close. However, the standard deviations for all three conditions were quite large.

Table 6.3: SCR summary by condition

	N	Mean size	SD
No music	179	0.314	0.423
Static music	191	0.303	0.341
Dynamic music	188	0.426	0.451
Total	588	0.348	0.41

In this case the three conditions are represented by two dummy variables, one for static music and one for dynamic music; the no music condition is indicated when both static music and dynamic music are set to 0

The main results of the GLMM analysis of the SCRs are summarized in Table 6.4. The music condition was encoded as two dummy variables[2] with the no music condition as the reference, so the coefficients for static and dynamic music indicate deviations from the no music condition. It is important to note that because the model actually predicts the inverse square of SCR size (as described in Section 6.4.2), negative coefficients indicate an *increase* in SCR size, while positive coefficients indicate a *decrease*. Thus, the results indicate that the static music SCRs were generally slightly smaller than the no music SCRs, but without statistical significance, while the dynamic music SCRs were larger than the no music SCRs, with statistical significance ($p < 0.001$). The model does not directly compare the static music and dynamic music conditions, but because the dynamic music SCRs were significantly larger than the no music SCRs and the static music SCRs were smaller than the no music SCRs, the dynamic music SCRs were therefore also significantly larger than the static music SCRs. Running the model again with dynamic

[2] A *dummy variable* is a variable that takes a value of 0 or 1 in order to indicate the absence or presence of an effect in a regression model. Multiple dummy variables can be used to encode categorical variables with more than two levels.

music as the reference condition confirmed this with $p < 0.001$.

Table 6.4: GLMM fixed effects summary

	Coef*	Std. err.	*t*	*p*
(Intercept)	0.691	0.056	13.116	**<0.001**
Static music	0.011	0.021	0.502	0.615
Dynamic music	-0.07	0.02	-3.503	**<0.001**
Order	-0.018	0.01	-1.731	0.083

*Positive coefficients indicate a negative effect on SCR size, while negative coefficients indicate a positive effect (see main text).

6.6 Discussion

To recap, the study tested the following hypotheses:

H1 Participants will prefer the dynamic music condition to to the static music and no music conditions, and find it more fun.

H2 Participants will rank the dynamic music condition as more tense and exciting than the static and no music conditions.

H3 The dynamic music condition will elicit larger skin conductance responses than the static and no music conditions.

As can be seen in Table 6.2 and Figure 6.3, participants who said they like horror games or films (the “Horror+” participants) tended to rank the dynamic music condition as both the most preferable and the most fun to play, which supports H1. However, H1 was not supported for participants who did not like horror games or films (the “Horror–” participants), who ranked the three conditions roughly equally preferable and fun. By contrast, there was consistency between the groups in the *tension* and *excitement* rankings, for which the dynamic music condition was ranked the highest almost unanimously, which supports H2. Similarly, there was strong evidence to support H3, since the mean SCR size for the dynamic music condition was larger than for the others, and the GLMM results showed that the effect was statistically significant.

These findings support two main conclusions. The first is that the dynamic music condition was the most emotionally arousing of the three. This is particularly noteworthy because, as discussed in Section 6.3.2, the overall amount of tension implied in the musical features was controlled between the dynamic music and static music conditions. The increased arousal that the participants experienced in the dynamic music condition can therefore be attributed to the music's dynamic behaviour rather than its overall amount of tension. In particular, it seems to be due to the emotional cues the music provided about the changing narrative. The dynamic music did not just indicate the player character's proximity to the enemies—their humming sounds clearly indicated their proximity in all three conditions—but more importantly that proximity to an enemy constitutes a tense narrative situation. This is reflected in the relatively high rankings for the dynamic music condition in the *enemy scariness* dimension, especially for the Horror– participants, who may have been more sensitive to the scariness of the enemies than the Horror+ participants. Interestingly, there were no statistically significant differences between the no music and static music conditions for the *tension* rankings, the *excitement* rankings, or the SCR sizes. Overall, it is unclear what the static music contributed to the game playing experience, but the dynamic music clearly made the game more emotionally arousing to play.

The second main conclusion is that, for most of the Horror+ participants, the dynamic music improved the subjective quality of the experience compared to the other conditions, which is evident in their rankings in the *preference/liking* and *fun* dimensions. In practice, the Horror+ group would likely constitute the target audience of a game like *Escape Point*, as it uses a dark visual style and relatively tense gameplay. However, while one might expect a player who enjoys the horror genre to favour the increased tension arising from the dynamic music, this would not necessarily hold for players who do not enjoy the genre. Thus, although the original hypothesis (H1) was that the dynamic music would improve the quality of the experience for all participants, it makes sense that this was only true for the Horror+ participants.

The study and its potential implications are further discussed in Chapter 7.

6.7 Transition study

The main study evaluated the proposed game music approach by comparing it with two variations of a control condition, neither of which were dynamic. Although both the static music and no music conditions were arguably plausible reflections of what might be expected in a real-life parallel of *Escape Point*, conventional game music systems do allow some degree of dynamic behaviour, primarily by crossfading different audio tracks. However, as noted in Section 1.1, this approach is normally used to reflect broad state transitions, which *Escape Point* lacks. The narrative as it was defined instead changes constantly and quite rapidly, and it is difficult to say whether and how a game music composer might have attempted to use the crossfading approach. Nonetheless, crossfading is used in many modern games and certainly warrants consideration in the evaluation of the proposed approach. A second study therefore compared the proposed approach—in this section referred to as the *interpolated parameters* approach—with two variations of the crossfading approach, focusing primarily on their ability to produce smooth musical transitions. The stimuli were audio clips of the output of the music generator using variations of the two parameter sets used in *Escape Point* (see Section 4.2.1), designed to express low and high emotional tension. In the study, music experts first listened to low-to-high and high-to-low tension transitions created using the three approaches and ranked them by perceived smoothness. They then listened to clips representing the midpoint of each transition and reported whether they expressed roughly a “medium” amount of tension, as well as how discordant and listenable they were.

For the interpolated parameters approach, the audio clips were created by interpolating the input parameters of the music generator between the low and high tension parameter sets, and recording the generator’s audio output. The crossfaded clips were created by first recording separate low and high tension clips, then mixing them together using a digital audio editor. However, the crossfading approach was further divided into “synced” and “unsynced” variations intended to represent more ideal and less ideal crossfades, respectively. As noted in Section 1.1, although crossfading between two clips can lead to harmonic and rhythmic clashing, this might be mitigated to some extent by ensuring that the clips are harmonically and rhythmically compatible as well as properly aligned. Thus, in the synced crossfades, the low and high tension clips were temporally aligned during the mixing process such that a chord onset in the low tension clip occurred at the same time as every other chord onset in the high tension clip. This was possible because the tempo of the low tension parameter set (60 BPM) was precisely half that of the high tension parameter set (120 BPM). By contrast, in the unsynced variation, the clips were not aligned, and their tempos were also shifted “inwards” by 10 BPM (60 became 70 BPM for low tension, and 120 became 110 BPM for high tension), which ensured that they were rhythmically incompatible. No adjustments were made to ensure or prevent harmonic compatibility in any of the crossfaded clips, as it was unclear how this could have been achieved systematically. However, it is worth noting that the low and high tension parameter sets did not imply different keys and thus were not overtly harmonically incompatible.

The study tested the following hypotheses:

H4 Continuously interpolating the music generator’s input parameters from one set to another over time will produce smoother transitions than crossfading between audio clips generated with the two parameter sets.

H5 Using an interpolated parameter set equivalent to the average of two parameter sets will express an intermediary emotion more accurately than overlaying audio clips generated with the two parameter sets.

6.7.1 Method

Eight postgraduate students and lecturers with musical backgrounds participated in the study, with means of 8.9 years of musical training and 18.8 years of experience playing or composing music. The study consisted of two sections and took about fifteen minutes to complete. Responses were collected via an online form.

In the first section, the participants ranked two groups of three transitions from least to most smooth. One group included transitions from low to high tension, and the other from high to low tension. For each group, one clip used parameter interpolation, one used a synced crossfade, and one used an unsynced crossfade. The clips were each twenty-two seconds long and consisted of five seconds of the first parameter set, twelve seconds of a transition to the second set, and five seconds of the second set.

In the second section, the participants evaluated ten-second clips of the music produced by each approach at the midpoint of a transition between low and high tension. The purpose of this section was mainly to determine how well each approach could express a “medium” amount of tension in relation to the low and high tension parameter sets. For the interpolated parameters approach, the clips were created by setting the music generator’s parameters to the average of the low and high tension parameter sets, whereas the crossfades were created by mixing separate recordings of the music generator using each of the two sets. Six clips were evaluated in total; each approach was represented by two clips differing slightly in their harmonies due to the music generator’s stochastic chord algorithm. The participants first listened to recordings of the low and high tension parameter sets as points of reference, then responded to the following multiple-choice questions for each of the six clips:

- How emotionally tense is the clip in relation to the “Low tension” and “High tension” clips?

  a) Close to or more than the “High tension” clip

  b) About in the middle

  c) Close to or less than the “Low tension” clip

- How discordant did you find the clip?

  a) Discordant

  b) Somewhat discordant

  c) Not discordant

- How listenable did you find the clip?

  a) Listenable

  b) Somewhat listenable

  c) Not listenable

In both sections, the participants were allowed to listen to the clips as many times as desired, including the low and high tension reference clips. The order of the clips was randomized in both sections.

Table 6.5: Mean smoothness rankings for the three transition approaches

	Low to high	High to low
Interpolated parameters	1	1.125
Synced crossfade	2.375	2.125
Unsynced crossfade	2.625	2.75

Table 6.6: Tension, discordance, and listenability ratings for the transition midpoints

	Tension			Discordant			Listenable		
	Low	Mid	High	No	Some	Yes	No	Some	Yes
Interp. parameters	2	14	0	10	4	2	1	6	9
Synced crossfade	0	6	10	4	5	7	2	8	6
Unsynced crossfade	0	3	13	3	5	8	2	9	5

6.7.2 Results

The participants' mean smoothness rankings for the transitions in the first section are shown in Table 6.5. The interpolated parameters clip was ranked the smoothest low-to-high tension transition by all participants, and the smoothest high-to-low tension transition by all but one participant. The synced crossfade was most often ranked the second smoothest, and the unsynced crossfade was most often ranked the least smooth.

The results from the second section are shown in Table 6.6. In all but two cases, the tension of the interpolated parameters clips was rated as about in the middle (the "Mid" column) of the low and high tension clips, whereas ten of the synced crossfades and thirteen of the unsynced crossfades were rated as close or more tense than the high tension clip. The ratings for discordance and listenability were less consistent, but the interpolated parameters approach was rated as discordant and not listenable less often than either two of the crossfade approaches, which were rated similarly.

6.7.3 Discussion

The results from the first section of the study (Table 6.5) indicate that interpolating the parameters of the music generator from one set to another produced smoother transitions than crossfading recordings of its audio output (H4). Indeed, the participants ranked the interpolated parameters clips as the most smooth of the three approaches in all cases except one, in which it was ranked the second most smooth. The results from the tension ratings in the second section (Table 6.6) further indicate that parameter interpolation also expressed an intermediary emotion more accurately than the crossfading approach (H5).

The fact that the interpolated parameters transitions were ranked the most smooth can likely be explained by the fact that the amount of tension at their midpoint was usually rated between those of the low and high tension clips. In other words, the interpolated parameters transitions started at one end of the tension scale, passed through a medium amount of tension at the midpoint, and continued to the other end of the scale. By contrast, the amount of tension at the midpoints of the crossfaded transitions was usually rated about the same or higher than that of the high tension clip. The emotional trajectory from one end of the scale to the other was therefore unclear, but the movement was not constant and was likely not what would be expected in a smooth transition.

This lack of smoothness may arise from the amount of discordance perceived during the crossfades—while some discordance was to be expected since the clips were intended to express a "medium" amount of tension, the crossfaded clips were rated as both more discordant and less listenable than the interpolated parameters clips. This seems to support the previously mentioned idea that crossfading two audio clips can lead to clashing between them, especially when they are harmonically and rhythmically incompatible. Although none of the crossfaded clips were adjusted to prefer or avoid harmonic compatibility, it seems unlikely that chord progressions expressing different emotions could be made harmonically compatible while maintaining their individual emotions. Interestingly, the synced crossfades, which used compatible rhythms and thus minimized or avoided rhythmic clashing, outperformed the unsynced crossfades, which provides further evidence of the impact of clashing. However, the differences measured between the two were relatively minimal compared to the differences measured between them and the interpolated parameters approach. Although a more thorough examination with clips specifically composed for crossfading would be needed to confirm this, overall the interpolated parameters approach seems to outperform the crossfading approach in terms of its dynamic capabilities.

CHAPTER 7

CONCLUSIONS

Unlike in films and other linear media, narratives in computer games are unpredictable both in structure and timing, relying on player interaction to help determine their exact course. To address this, most conventional game music systems use audio mixing techniques such as crossfading to piece together pre-recorded, linear pieces of music. In practice, however, this limits the ability of the music to support the narrative mostly to high-level state transitions, and the relevance of these systems is waning as game narratives become increasingly dynamic. Recently proposed alternatives to conventional game music have laid some groundwork for how music might better support such narratives, but still have been limited in their dynamic capabilities. In particular, the problem of how to produce smooth transitions has not been addressed. Additionally, most of these alternatives have not been implemented in games, and none have been evaluated in a game. Overall, despite growing interest in the field, there is still little concrete evidence of how it can progress. This dissertation has addressed the above concerns through the rigorous investigation of a novel approach to game music.

In this final chapter, the original research question is first revisited and discussed, then the main conclusions and contributions of the research are summarized. Several possible directions for future work are then outlined, and some final remarks are offered about the future of game music research.

7.1 Insights from the research

The research question that guided the present work was *"How can music be generated automatically in such a way that it can represent a player's progression through a game narrative?"* To address this question, a novel approach to game music was investigated in which a system generates the music algorithmically based on a set of input parameters corresponding to emotional musical features, the variation of which would enable the music to continuously adapt to reflect a game's emotional narrative. An algorithmic music generator was developed to demonstrate the feasibility of the approach; its input parameters correspond to musical features with previously demonstrated emotional correlates, and are all continuous and can thus be varied smoothly over time. The generator was then configured to dynamically support the narrative of *Escape Point*, a 3-D maze game, by becoming more or less tense depending on the proximity of the player character to the game's enemies.

The music generator and the underlying approach were evaluated in three empirical studies:

- The **preliminary study** (Section 3.2.2) evaluated the ability of the prototype music generator to express different emotions as well as to perform smooth transitions between emotions. In terms of the evaluation approaches outlined in Chapter 5, it was a music-oriented study in that the generator's musical output was the main focus.

- The **main study** (Sections 6.1–6.6) evaluated the impact of *Escape Point's* music compared to conditions with static music and no music. Both psychophysiological and subjective responses were collected from participants. It was a player-oriented study in that the participants' experiences were the main focus.

- The **transition study** (Section 6.7) compared the proposed approach with cross-fading in order to determine which produced better transitions. Like the preliminary study, it was a music-oriented study.

The results from these studies suggest two main conclusions that can be drawn regarding the proposed approach:

> **Conclusion 1**: Algorithmically generating music based on a set of input parameters corresponding to emotional features is a viable approach to producing music with controllable, dynamic emotional expression.

Conclusion 1 was supported by the preliminary study, which demonstrated that varying the three parameters of the prototype music generator successfully influenced the emotions participants perceived in the music. This was especially true for the arousal dimension of the valence/arousal model of emotion (Russell, 1980). However, only one of the prototype's input parameters was intended to influence valence, and both valence and arousal were nonetheless influenced with statistical significance. Furthermore, transitions between different expressed emotions were reported as smooth and musically natural in the vast majority of cases. The transition study also supported Conclusion 1 because the participants consistently reported the music generator's medium tension setting as successfully expressing a medium amount of tension. This was not true when using the crossfading approach, in which the low tension and high tension audio were simply overlaid. The transition study also demonstrated that transitions were more smooth when produced by varying the music generator's parameters than when using crossfading. These results suggest that the proposed approach is capable of dynamically expressing different emotions in a controllable and musical way.

> **Conclusion 2**: Dynamically supporting a game narrative by varying the emotional expression of the music can positively impact a game playing experience.

Conclusion 2 was supported by the main study, which demonstrated that controlling the parameters of the music generator to reflect the changing amount of tension in the narrative of *Escape Point* made the game overall more subjectively tense and exciting than the static music and no music conditions, and also elicited larger skin conductance responses in the participants. For participants who reported that they enjoy horror games and films, which would likely be the target audience of a game like *Escape Point*, the dynamic music condition was also rated the most preferable and fun of the three conditions.

Together, these two conclusions suggest that the proposed approach successfully addressed the research question, and is a promising solution to the problem of implementing music in computer games. The main contributions of this research can thus be summarized as follows:

- A novel approach to game music was proposed and rigorously investigated, which was shown to be both viable and effective. Overall, the approach seems better equipped to support dynamic narratives than conventional game music systems.
- An original methodology was developed to empirically evaluate game music in the context of an actual game. A range of other evaluation methods were also reviewed, which will hopefully contribute to the adoption of standard evaluation techniques in the field.

7.2 Future directions

Research on different aspects of games and game playing experiences has gained considerable momentum in recent years. Although game music research has attracted some attention, it nonetheless remains a largely unexplored subject. Although this dissertation has addressed some key gaps in the literature, a number of questions remain. This work provides a starting point for several directions of research, some of the most prominent of which are described below.

More complex narratives

This research focused on the question of how to generate music that can support a game's emotional narrative, and proposed that varying emotional musical features of the music would enable it to do so. However, the emotional narrative of *Escape Point* was quite simplistic in the way it was defined, as it only involved a varying amount of tension. Controlling the music generator was thus a matter of defining parameter sets to represent the states of low and high tension, and interpolating between them accordingly. Many games involve more complex, multilayered narratives, and it remains to be seen how music generated using the proposed approach could be used effectively in these situations. One direction for future work could thus be to focus on the proposed approach from a higher level—how to control the music rather than how to generate it. This research could address any or all of the following:

- How to encode a game's emotional narrative using a multi-dimensional model
- How to support different layers of states—for example, as noted in Section 2.1.2, Gasselseder (2014) suggests that game music should reflect both long-term and short-term aspects of the narrative
- Whether game designers prefer to interface with musical features or expressed emotions—if both, then how they could be consolidated

Empirical research with games

The main study described in Chapter 6 used a novel methodology for measuring game playing experiences. The three conditions used in the study were chosen in order to best address this dissertation's main research question, but a number of related research questions could be addressed by comparing the dynamic music condition with any of the following, for example:

- A condition in which the music is dynamically varied using the crossfading approach (although crossfading was evaluated in the follow-up study described in Section 6.7)
- A condition in which the amount of musical tension is held statically at its maximum
- A condition in which the amount of musical tension is varied randomly, for example using a random walk
- A condition in which the amount of musical tension is only sometimes varied to reflect player/enemy proximity, for example only when the enemy is visible onscreen

- A condition in which only certain musical features are varied
- A condition in which player/enemy proximity is reflected visually rather than musically, such as via a minimap or a bar displaying the current amount of tension

Most of the above were implemented as options in *Escape Point.*[1]

Personalized game music

One interesting aspect of the proposed approach is that it would enable the music to be adjusted to suit individual preferences. For example, in a game like *Escape Point*, the optimal amount of musical tension might vary from player to player—some might be content to experience the full range of tension, while others might prefer to decrease the maximum amount of tension. Future work could therefore investigate whether and how game music can be personalized to help achieve a desired effect in spite of individual differences.

Relative importance of different musical features

The present research did not assess the relative importance of different musical feature manipulations in the expression of emotion. This would be useful to know because providing control over certain musical features in an algorithmic music system could require more development time or runtime processing than for other features. For example, implementing the *tempo* parameter of the music generator was simply a matter of assigning a variable rate to the timer used to trigger beats, whereas the harmony parameters involved the development of transition matrix filters. At the same time, tempo might have a more prominent effect in the expression of certain emotions. Depending on the particular range of emotions implied by a game's narrative, it might be sufficient to only implement these less complex feature controls. Thus, while much previous work has examined the emotional correlates of different musical features, future work could focus on determining the relative importance of different musical features in emotional expression.

[1] *Escape Point* and its source code can be downloaded at http://oro.open.ac.uk/view/person/ap22536.html.

7.3 Final remarks

While the field of game music is still quite young, its relevance is ever increasing as games continue to thrive in modern society. We are only beginning to understand how music functions in broader narrative contexts, but the potential for dynamic behaviour seems to be central to many of its capabilities. Conventional game music systems have changed very little in the past ten to fifteen years despite their limitations in this regard. Although the leap from using pre-composed music to algorithmic music is quite large, algorithmic music offers much more dynamic control. This dissertation has demonstrated not only that algorithmic music can serve as a viable foundation for a game music system, but also that its unique dynamic capabilities can strongly and positively impact the playing experience. Overall, the proposed approach seems to be a promising future direction for game music, and hopefully will continue to be investigated.

REFERENCES

Akaike, H. (1974). A New Look at the Statistical Model Identification. *IEEE Transactions on Automatic Control*, 19(6):716–723.

Aldwell, E., Schachter, C., and Cadwallader, A. (2011). *Harmony & Voice Leading*. Schirmer, Cengage Learning, Boston.

Ames, C. (1989). The Markov Process as a Compositional Model: A Survey and Tutorial. *Leonardo*, 22(2):175–187.

Ariza, C. (2009). The Interrogator as Critic: The Turing Test and the Evaluation of Generative Music Systems. *Computer Music Journal*, 33(2):48–70.

Bagiella, E., Sloan, R. P., and Heitjan, D. F. (2000). Mixed-effects models in psychophysiology. *Psychophysiology*, 37:13–20.

Bates, D., Maechler, M., Bolker, B., and Walker, S. (2015). Fitting Linear Mixed-Effects Models Using lme4. *Journal of Statistical Software*, 67(1):1–48.

Benjamini, Y. and Hochberg, Y. (1995). Controlling the False Discovery Rate: a Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society*, 57(1):289–300.

Berndt, A., Dachselt, R., and Groh, R. (2012). A Survey of Variation Techniques for Repetitive Games Music. In *Proceedings of the 7th Audio Mostly Conference - a Conference on Interaction with Sound*, pages 61–67, New York. ACM Press.

Berndt, A. and Hartmann, K. (2007). Strategies for Narrative and Adaptive Game Scoring. In *Proceedings of the 2nd Audio Mostly Conference - a Conference on Interaction with Sound*, pages 141–147, Ilmenau.

Berndt, A. and Hartmann, K. (2008). The Functions of Music in Interactive Media. In Spierling, U. and Szilas, N., editors, *Proceedings of the International Conference on Interactive Digital Storytelling*, pages 126–131, Erfurt.

Bravo, F. (2012). The Influence of Music on the Emotional Interpretation of Visual Contexts: Designing Interactive Multimedia Tools for Psychological Research. In *Proceedings of the 9th International Symposium on Computer Music Modeling and Retrieval*, pages 600–610, London.

Bresin, R. and Friberg, A. (2000). Emotional Coloring of Computer-Controlled Music Performances. *Computer Music Journal*, 24(4):44–63.

Brockmyer, J. H., Fox, C. M., Curtiss, K. A., McBroom, E., Burkhart, K. M., and Pidruzny, J. N. (2009). The development of the Game Engagement Questionnaire: A measure of engagement in video game-playing. *Journal of Experimental Social Psychology*, 45(4):624–634.

Brown, D. (2012a). Mezzo: An Adaptive, Real-Time Composition Program for Game Soundtracks. In *Proceedings of the Eighth Artificial Intelligence and Interactive Digital Entertainment Conference*, pages 68–72, Palo Alto.

Brown, D. L. (2012b). *Expressing narrative function in adaptive, computer-composed music*. PhD thesis, University of California, Santa Cruz.

Bullerjahn, C. and Güldenring, M. (1994). An empirical investigation of the effects of film music using qualitative content analysis. *Psychomusicology*, 13:99–118.

Cacioppo, J., Tassinary, L. G., and Bernston, G. G. (2007). *Handbook of Psychophysiology*. Cambridge University Press, Cambridge.

Casella, P. and Paiva, A. (2001). MAgentA: an Architecture for Real Time Automatic Composition of Background Music. In *Proceedings of the Third International Workshop on Intelligent Virtual Agents*, pages 224–232, Madrid.

Cohen, A. J. (1998). The Functions of Music in Multimedia: A Cognitive Approach. In *Proceedings of the Fifth International Conference on Music Perception and Cognition*, pages 13–20, Seoul.

Collins, K. (2008). *Game Sound*. The MIT Press, Cambridge, MA.

Collins, T., Laney, R., Willis, A., and Garthwaite, P. H. (2011). Chopin, mazurkas and Markov. *Significance*, 8(4):154–159.

Csikszentmihalyi, M. (1990). *Flow: The Psychology of Optimal Experience*. Harper Perennial, New York.

Dawson, M. E., Schell, A. M., and Filion, D. L. (2007). The Electrodermal System. In Cacioppo, J., Tassinary, L. G., and Berntson, G. G., editors, *Handbook of Psychophysiology*, pages 159–181. Cambridge University Press, Cambridge.

Ebcioglu, K. (1988). An Expert System for Harmonizing Four-part Chorales. *Computer Music Journal*, 12(3):43–51.

Eerola, T. (2012). Modeling Listeners' Emotional Response to Music. *Topics in Cognitive Science*, 4(4):607–24.

Eerola, T. and Vuoskoski, J. K. (2011). A comparison of the discrete and dimensional models of emotion in music. *Psychology of Music*, 39(1):18–49.

Eigenfeldt, A. and Pasquier, P. (2009). Realtime Generation of Harmonic Progressions Using Controlled Markov Selection. In *Proceedings of the First International Conference on Computational Creativity*, Lisbon.

Eladhari, M., Nieuwdorp, R., and Fridenfalk, M. (2006). The Soundtrack of Your Mind: Mind Music - Adaptive Audio for Game Characters. In *Proceedings of Advances in Computer Entertainment Technology*, Los Angeles.

Farbood, M. and Schoner, B. (2001). Analysis and Synthesis of Palestrina-Style Counterpoint Using Markov Chains. In *Proceedings of the International Computer Music Conference*, Havana.

Fox, J. (2008). Generalized Linear Models. In *Applied Regression Analysis and Generalized Linear Models*, pages 379–424. Sage Publications, Inc., Thousand Oaks, 2nd edition.

Freedman, L. W., Scerbo, A. S., Dawson, M. E., Raine, A., McClure, W. O., and Venables, P. H. (1994). The relationship of sweat gland count to electrodermal activity. *Psychophysiology*, 31:196–200.

Friberg, A., Bresin, R., and Sundberg, J. (2006). Overview of the KTH rule system for musical performance. *Advances in Cognitive Psychology*, 2(2-3):145–161.

Gabrielsson, A. (2002). Emotion perceived and emotion felt: same or different? *Musicae Scientiae*, Special is:123–147.

Gabrielsson, A. and Lindström, E. (2010). The role of structure in the musical expression of emotions. In Juslin, P. N. and Sloboda, J. A., editors, *Handbook of Music and Emotion: Theory, Research, Applications*, pages 367–400. Oxford University Press, Oxford.

Gajadhar, B. J., de Kort, Y. A. W., and IJsselsteijn, W. A. (2008). Shared Fun Is Doubled Fun: Player Enjoyment as a Function of Social Setting. In *Proceedings of the Second International Conference on Fun and Games*, pages 106–117, Eindhoven.

Gasselseder, H.-P. (2014). Dynamic Music and Immersion in the Action-Adventure: An Empirical Investigation. In Grimshaw, M. and Walther-Hansen, M., editors, *Proceedings of the 9th Audio Mostly Conference*, Aalborg.

Hevner, K. (1935). The Affective Character of the Major and Minor Modes in Music. *The American Journal of Psychology*, 47(1):103–118.

Hoeberechts, M., Demopoulos, R. J., and Katchabaw, M. (2007). A Flexible Music Composition Engine. In *Proceedings of the 2nd Audio Mostly Conference*, pages 52–57, Ilmenau.

Hoeberechts, M. and Shantz, J. (2009). Real-Time Emotional Adaptation in Automated Composition. In *Proceedings of the 4th Audio Mostly Conference - a Conference on Interaction with Sound*, pages 1–8, Glasgow.

IJsselsteijn, W., de Kort, Y., Poels, K., Jurgelionis, A., and Bellotti, F. (2007). Characterising and Measuring User Experiences in Digital Games. In *Proceedings of 4th International Conference on Advances in Computer Entertainment Technology (ACE '07)*, Salzburg.

Ilie, G. and Thompson, W. F. (2006). A Comparison of Acoustic Cues in Music and Speech for Three Dimensions of Affect. *Music Perception*, 23(4):319–330.

Jorgensen, K. (2006). On the Functional Aspects of Computer Game Audio. In *Proceedings of the Audio Mostly Conference*, Piteå.

Juslin, P. N. and Timmers, R. (2010). Expression and communication of emotion in music performance. In Juslin, P. N. and Sloboda, J. A., editors, *Handbook of Music and Emotion: Theory, Research, Applications*, pages 453–489. Oxford University Press, Oxford.

Kivikangas, J. M., Chanel, G., Cowley, B., Ekman, I., Salminen, M., Järvalä, S., and Ravaja, N. (2011). A review of the use of psychophysiological methods in game research. *Journal of Gaming and Virtual Worlds*, 3(3):181–199.

Klimmt, C., Hartmann, T., and Frey, A. (2007). Effectance and Control as Determinants of Video Game Enjoyment. *CyberPsychology & Behavior*, 10(6):845–847.

Lang, P. J. (1995). The Emotion Probe: Studies of Motivation and Attention. *American Psychologist*, 50(5):372–385.

León, C. and Gervás, P. (2012). Prototyping the Use of Plot Curves to Guide Story Generation. In Finlayson, M. A., editor, *Proceedings of the Third Workshop on Computational Models of Narrative*, pages 152–156, Istanbul.

Lindsey, J. K. and Jones, B. (1998). Choosing among generalized linear models applied to medical data. *Statistics in Medicine*, 17:59–68.

Livingstone, S. R. (2008). *Changing Musical Emotion through Score and Performance with a Computational Rule System*. PhD thesis, The University of Queensland.

Livingstone, S. R., Muhlberger, R., Brown, A. R., and Thompson, W. F. (2010). Changing Musical Emotion: A Computational Rule System for Modifying Score and Performance. *Computer Music Journal*, 34(1):41–64.

Madsen, C. K., Brittin, R. V., and Capperella-Sheldon, D. A. (1993). An Empirical Method for Measuring the Aesthetic Experience to Music. *Journal of Research in Music Education*, 41(1):57–69.

Mandryk, R. L., Inkpen, K. M., and Calvert, T. W. (2006). Using psychophysiological techniques to measure user experience with entertainment technologies. *Behaviour & Information Technology*, 25(2):141–158.

Marshall, S. K. and Cohen, A. J. (1988). Effects of Musical Soundtracks on Attitudes toward Animated Geometric Figures. *Music Perception*, 6(1):95–112.

McKinney, W. (2011). pandas: a Foundational Python Library for Data Analysis and Statistics. In *Proceedings of Python for High Performance and Scientific Computing (PyHPSC 2011)*, Seattle.

Mirza-babaei, P., Nacke, L. E., Gregory, J., Collins, N., and Fitzpatrick, G. (2013). How Does It Play Better? Exploring User Testing and Biometric Storyboards in Games User Research. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1499–1508, Paris.

Moneta, G. B. (2012). On the Measurement and Conceptualization of Flow. In Engeser, S., editor, *Advances in Flow Research*, pages 23–50. Springer, New York.

Morreale, F., Masu, R., and Angeli, A. D. (2013). Robin: An Algorithmic Composer for Interactive Scenarios. In *Proceedings of the Sound and Music Computing Conference*, pages 207–212, Stockholm.

Müller, M. and Driedger, J. (2012). Data-Driven Sound Track Generation. In Müller, M., Goto, M., and Schedl, M., editors, *Multimodal Music Processing*, pages 175–194. Dagstuhl Publishing, Saarbrücken/Wadern.

Nacke, L. and Lindley, C. (2008a). Boredom, Immersion, Flow: A Pilot Study Investigating Player Experience. In *Proceedings of the IADIS Gaming 2008: Design for Engaging Experience and Social Interaction*, pages 103–107, Amsterdam. IADIS Press.

Nacke, L. and Lindley, C. A. (2008b). Flow and Immersion in First-Person Shooters: Measuring the player's gameplay experience. In *Proceedings of the 2008 Conference on Future Play*, pages 81–88, Toronto.

Nacke, L. E., Grimshaw, M. N., and Lindley, C. A. (2010). More than a feeling: Measurement of sonic user experience and psychophysiology in a first-person shooter game. *Interacting with Computers*, 22(5):336–343.

Nierhaus, G. (2009). *Algorithmic Music: Paradigms of Automatic Music Generation*. Springer-Verlag/Wien, New York.

Norman, K. L. (2013). GEQ (Game Engagement/Experience Questionnaire): A Review of Two Papers. *Interacting with Computers*, 25(4):278–283.

Pachet, F. (2002). The Continuator: Musical Interaction With Style. In *Proceedings of the International Computer Music Conference*, Gothenburg.

Parke, R., Chew, E., and Kyriakakis, C. (2007a). Multiple Regression Modeling of the Emotional Content of Film and Music. In *Proceedings of the 123rd Convention of the Audio Engineering Society*, New York.

Parke, R., Chew, E., and Kyriakakis, C. (2007b). Quantitative and Visual Analysis of the Impact of Music on Perceived Emotion of Film. *ACM Computers in Entertainment*, 5(3).

Paterson, N., Naliuka, K., Jensen, S. K., Carrigy, T., Haahr, M., and Conway, F. (2010). Design, Implementation and Evaluation of Audio for a Location Aware Augmented Reality Game. In *Proceedings of the 3rd International Conference on Fun and Games*, pages 149–156, New York. ACM Press.

Pearce, M. and Wiggins, G. (2001). Towards A Framework for the Evaluation of Machine Compositions. In *Proceedings of the 2001 AISB Symposium on AI and Creativity in Arts and Science*, pages 22–32, York.

Peterson, F. and Jung, C. G. (1907). Psycho-physical investigations with the galvanometer and pneumograph in normal and insane individuals. *Brain*, 30:153–218.

Piston, W. (1959). *Harmony*. Victor Gollancz Ltd, London.

Rabiger, M. (2003). *Directing: Film Techniques and Aesthetics*. Focal Press, Waltham, Massachusetts, third edition.

Ravaja, N., Saari, T., Salminen, M., Laarni, J., and Kallinen, K. (2006). Phasic Emotional Reactions to Video Game Events: A Psychophysiological Investigation. *Media Psychology*, 8(4):343–367.

Russell, J. A. (1980). A Circumplex Model of Affect. *Journal of Personality and Social Psychology*, 39(6):1161–1178.

Rutherford, J. and Wiggins, G. (2002). An Experiment in the Automatic Creation of Music Which Has Specific Emotional Content. In *Proceedings of the Seventh International Conference on Music Perception and Cognition*, Sydney.

Schoenberg, A. (1969). *Preliminary Exercises in Counterpoint*. St. Martin's Press, New York.

Schubert, E. (2004). Modeling Perceived Emotion With Continuous Musical Features. *Music Perception*, 21(4):561–585.

Sweetser, P., Johnson, D., and Wyeth, P. (2012). Revisiting the GameFlow Model with Detailed Heuristics. *Journal of Creative Technologies*, 3.

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

Tanaka, T., Nishimoto, T., Ono, N., and Sagayama, S. (2010). Automatic music composition based on counterpoint and imitation using stochastic models. In *Proceedings of the 7th Sound and Music Computing Conference*, Barcelona.

Temperley, D. and Tan, D. (2013). Emotional Connotations of Diatonic Modes. *Music Perception*, 30(3):237–257.

Thayer, J. F. and Levenson, R. W. (1983). Effects of music on psychophysiological responses to a stressful film. *Psychomusicology*, 3(1):44–52.

Tognetti, S., Garbarino, M., Bonarini, A., and Matteucci, M. (2010). Modeling enjoyment preference from physiological responses in a car racing game. In *Proceedings of the 2010 IEEE Conference on Computational Intelligence and Games*, pages 321–328, Copenhagen.

Tsang, C. P. and Aitken, M. (1991). Harmonizing Music as a Discipline of Constraint Logic Programming. In *Proceedings of the International Computer Music Conference*, pages 61–64, Montreal.

Turing, A. M. (1950). Computing Machinery and Intelligence. *Mind*, 59(236):433–460.

Verbeurgt, K., Dinolfo, M., and Fayer, M. (2004). Extracting Patterns in Music for Composition via Markov Chains. In *IEA/AIE'2004: Proceedings of the 17th International Conference on Innovations in Applied Artificial Intelligence*, pages 1123–1132, Ottawa. Springer-Verlag.

Wasserman, K. C., Eng, K., Verschure, P. F. M. J., and Manzolli, J. (2003). Live Soundscape Composition Based on Synthetic Emotions. *IEEE MultiMedia*, 10(4):82–90.

Watson, D. and Clark, L. A. (1994). The PANAS-X: Manual for the Positive and Negative Affect Schedule - Expanded Form. Technical report.

Weibel, D., Wissmath, B., Habegger, S., Steiner, Y., and Groner, R. (2008). Playing online games against computer- vs. human-controlled opponents: Effects on presence, flow, and enjoyment. *Computers in Human Behavior*, 24(5):2274–2291.

Williams, D., Kirke, A., Miranda, E. R., Roesch, E. B., and Nasuto, S. J. (2013). Towards affective algorithmic composition. In Luck, G. and Brabant, O., editors, *Proceedings of the 3rd International Conference on Music and Emotion (ICME3)*, Jyväskylä.

Wingstedt, J. (2004). Narrative functions of film music in a relational perspective. In *Proceedings of the International Society for Music Education World Conference*, Tenerife.

Wingstedt, J., Brandström, S., and Berg, J. (2008). Young adolescents' usage of narrative functions of media music by manipulation of musical expression. *Psychology of Music*, 36(2):193–214.

Winters, B. (2009). Corporeality, Musical Heartbeats, and Cinematic Emotion. *Music, Sound and the Moving Image*, 2(1):3–25.

Witmer, B. G. and Singer, M. J. (1998). Measuring Presence in Virtual Environments: A Presence Questionnaire. *Presence*, 7(3):225–240.

APPENDIX **A**

RELEVANT CONCEPTS FROM WESTERN MUSIC THEORY

The following is a brief description of a few core concepts of Western music theory that are referenced in the main text of this dissertation, particularly Chapter 3.

Scales, keys, and modulation

A *scale* is a collection of notes typically arranged in ascending pitch order. The C major scale, for example, can be written as

C-D-E-F-G-A-B-C

The term *key* is related, but more general. Music in a particular key (e.g., C major) can use the notes of the respective scale (the C major scale) in any order, and any notes outside the scale as well. However, the notes of a scale tend to have important characteristics that depend on their ordinal position in the scale—their *scale degree*—by which they are often referenced. In the C major scale, C is the first scale degree (sometimes called the *tonic* of the scale), D is the second, E is the third, and so on. Referencing notes in this way instead of by their names allows their characteristics and functions to be generalized across scales.

There are many different types of scales, but the most prominent ones in Western music are major and minor scales. Minor scales have the connotation of being darker than their major counterparts. Compared to major scales, they have “flattened” third, sixth, and sometimes seventh scale degrees. When the seventh is flattened, the scale is normally referred to as a *natural* minor scale, otherwise it is referred to as a *harmonic* minor scale. Thus, C (natural) minor can be written as

C-D-E♭-F-G-A♭-B♭-C

It is sometimes desirable to *modulate* between different keys—to transition from one to another within a piece of music. Certain pairs of keys have a “close” relationship in the sense that modulating between them is relatively easy and natural from a compositional standpoint. One such case is *parallel* major and minor keys, whose scales share the same tonic note. For example, C major and C minor are parallel keys because they both start on C. However, as mentioned above, minor scales have a few flattened notes compared to major scales. Thus, although parallel keys share the same tonic, not all notes are shared. By contrast, major and minor keys whose scales share the same notes (or all but one, in the case of the harmonic minor), but simply start on different notes, are said to be *relative*. For a given major scale, the relative minor starts on its sixth scale degree, and for a given minor scale, the relative major starts on its third scale degree. For example, A minor is the relative minor of C major (A is the sixth scale degree of C major), and C major is the relative major of A minor (A is the third scale degree of C major). Because their scales share the same notes, it is particularly easy to modulate between relative major and minor keys since chromaticism (described below) can be minimized during the modulation.

Intervals

One of the most important concepts in music theory is the *interval*, which describes the pitch distance between two notes. It is often measured as the number of consecutive scale degrees up or down a scale, inclusive of both the starting and ending note. For example, consider the G major scale:

G-A-B-C-D-E-F♯-G

The interval between G and C is called a “fourth” because four notes are encountered when counting up the scale from G to C. The interval between A and D is also a fourth for the same reason. Two specially named intervals are the *unison*, which is an interval of one (i.e., the notes have the same pitch), and the *octave*, an interval of eight. The interval between two G’s, for example, could be a unison or any number of octaves.

Sometimes intervals are instead specified in *steps*, or the number of scale degrees up or down a scale *excluding* the starting note. For example, an A could be described as one step above a G, and a B two steps above a G. In melodies, an interval of a step is often distinguished from a *leap* or *skip*, which both refer to moving to any note more than one step away.

It is worth noting that intervals can be further qualified—for example, *half steps* can be distinguished from *whole steps*—and they can be defined in other ways, such as ratios. However, these are more advanced topics, the command of which is not necessary to understand the main text of this dissertation, and thus are not covered here.

Chords

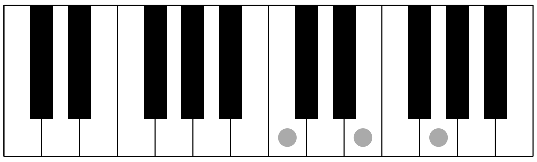
A *chord* is a simultaneous sounding of multiple notes, and can be defined partly by the intervals they form and partly by the root note of the chord. In Western music, the *triad* is the most common type of chord. It consists of three notes: a root note, a “third”, which is an interval of a third above the root note, and a “fifth”, which is a fifth above the root note. For example, a C major chord is a triad consisting of the notes C (the root), E (the third), and G (the fifth). A seventh is sometimes also included, in which case the chord would be considered a *seventh chord* rather than a triad.

The scale degree of a chord’s root note is normally of particular significance. In a general sense, a C major chord played in the key of C major will sound similar to a G major chord played in the key of G major, at least in the broader context. Chords are therefore often referred to by their root note’s scale degree, and notated as a roman numeral. For example, a G major chord is a V chord in the key of C major because G is the fifth scale degree of the C major scale. Similarly, an A major would be a V chord in the key of D major. As with the convention of referring to notes by their scale degree, an advantage of referring to chords by their root note’s scale degree is that their characteristics and functions can then be generalized across keys.

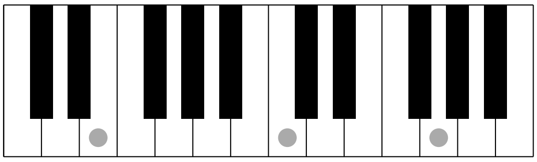
Finally, it is worth noting that chord notes can be broken up and played consecutively (instead of simultaneously) which is referred to as an *arpeggio.* Arpeggios are similar in principle to melodies except that they only consist of notes comprising the chord.

Inversions and voice leading

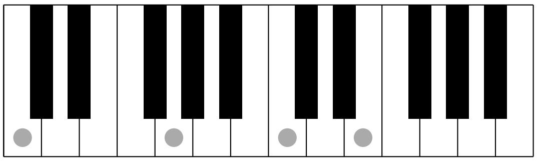
The notes of a chord are typically listed in ascending order starting from the root note. For example, a C major chord is comprised of the notes C, E, and G. However, they can be arranged in many different ways. Here, they are arranged in ascending order, and close together:



Here, they are more spread out, and with E as the lowest note rather than C:



Notes can also be doubled. Here, the C is played in both the lower and upper octaves:



When describing the arrangement of a particular chord, one of the most important aspects is which note is lowest, which is usually referred to as the “bass” note. When the lowest note of the chord is its root, it is said to be in *root position.* When the lowest note is its third, it is said to be in the *first inversion.* Finally, when the lowest note is its fifth, it is said to be in the *second inversion.* The *third inversion* is also possible for seventh chords.

There are certain conventions in Western music for how the individual notes of a chord should progress to the respective notes of the next chord. When arranging the notes of a chord progression, the chords are usually decomposed into multiple “voices” (either conceptually or literally) which should individually exhibit desirable aesthetic properties. The process of arranging chords in this way is referred to as *voice leading*, and is characterized by a set of common conventions. One such convention is that so-called *parallel fifths* are normally discouraged. Parallel fifths occur when two voices a fifth apart each move up or down by the same interval, thus remaining a fifth apart after the move. Parallel fifths draw attention to themselves and give the impression of upward or downward motion, which can be distracting.

Voice leading is discussed further in Chapter 3.

Diatonic and chromatic notes and chords

A note is considered *diatonic* if it is contained in the current scale. In the C major scale, for example, the notes C, D, E, F, G, A, and B are all diatonic. Any note not contained in the scale is a *chromatic* note. E, for example, is a diatonic note in the key of C major, but a chromatic note in the key of B♭ major (B♭-C-D-E♭-F-G-A-B♭). A chord can also be considered diatonic as long as it is comprised solely of diatonic notes; otherwise, it is considered chromatic.

Diatonic notes and chords are typically thought of as “consonant”, or harmonious. The connotation is that they are generally pleasing to hear—a diatonic note or chord will rarely sound wrong or out of place, for example. By contrast, chromatic notes and chords are typically thought of as “dissonant”, and are often used to create tension. Depending on how they are approached and resolved they *may* sound consonant, but if performed accidentally they will often sound wrong or out of place. In general, the use of chromaticism is relatively complex compositionally, and is usually considered an advanced topic in music theory studies.

APPENDIX B

MAIN STUDY QUESTIONNAIRES

Continued on the next page are the questionnaires used in the main study described in Chapter 6.

Your Background

Gender:

○ Male
○ Female

Age group:

○ 18–24
○ 25–29
○ 30–39
○ 40–49
○ 50–59
○ 60+

Do you generally like computer and/or console (Xbox, Playstation, etc.) games?

○ Yes
○ No
○ I have not played them much but am interested
○ I have not played them much and am not interested

Which of the following most accurately describes your past experience with digital 3D games?

○ I have played one with a keyboard and mouse
○ I have played one, but never with a keyboard and mouse
○ I have not played one, but have played other kinds of digital games
○ I have never played a digital game before

Questionnaire 1

If you wish, feel free to make comments anywhere on this page, or verbally to the researcher.

In terms of the music (or lack of music, if there was none), which did you like more?

○ The first version
○ The second version
○ I did not notice a difference, or have no preference

Which version of the game felt more tense?

○ The first version
○ The second version
○ I did not notice a difference

Which version of the game felt more exciting?

○ The first version
○ The second version
○ I did not notice a difference

Which version of the game felt more challenging?

○ The first version
○ The second version
○ I did not notice a difference

In which version of the game did the enemies seem more scary?

○ The first version
○ The second version
○ I did not notice a difference

Which version of the game was more fun?

○ The first version
○ The second version
○ I did not notice a difference

Questionnaire 2

If you wish, feel free to make comments anywhere on this page, or verbally to the researcher.

In terms of the music (or lack of music, if there was none), which did you like more?

○ The second version
○ The third version
○ I did not notice a difference, or have no preference

Which version of the game felt more tense?

○ The second version
○ The third version
○ I did not notice a difference

Which version of the game felt more exciting?

○ The second version
○ The third version
○ I did not notice a difference

Which version of the game felt more challenging?

○ The second version
○ The third version
○ I did not notice a difference

In which version of the game did the enemies seem more scary?

○ The second version
○ The third version
○ I did not notice a difference

Which version of the game was more fun?

○ The second version
○ The third version
○ I did not notice a difference

Questionnaire 3

Please think back to the first version of the game, and compare it to the one you just played (ignoring the second version). Feel free to make comments anywhere on this page, or verbally to the researcher.

In terms of the music (or lack of music, if there was none), which did you like more?

○ The first version
○ The third version
○ I did not notice a difference, or have no preference

Which version of the game felt more tense?

○ The first version
○ The third version
○ I did not notice a difference

Which version of the game felt more exciting?

○ The first version
○ The third version
○ I did not notice a difference

Which version of the game felt more challenging?

○ The first version
○ The third version
○ I did not notice a difference

In which version of the game did the enemies seem more scary?

○ The first version
○ The third version
○ I did not notice a difference

Which version of the game was more fun?

○ The first version
○ The third version
○ I did not notice a difference

Given that you played the first version of the game a little while ago, how confident do you feel about the above answers?

○ Confident (I remember the first version well)
○ Somewhat confident (I think I remember the first version)
○ Not confident (I barely or don't remember the first version)