

THE NLN-PLAYER: A SYSTEM FOR NONLINEAR MUSIC IN GAMES

Than van Nispen tot Pannerden, Sander Huiberts

Sebastiaan Donders, Stan Koch

Utrecht School of the Arts
Leader Research Group Music Design
Hilversum, The Netherlands
than@kmt.hku.nl

GreenCouch
Music Composition
info@greencouch.nl

ABSTRACT

Interactive music, in e.g. video games, often tends to be complex both in the creative and the technological part. Video games, that have any interactivity connected to the music, often have simplistic music and music technology. The sounding results vary greatly in quality, both in musical aesthetics and in interactive meaningfulness.

One of the interactive music strategies available is horizontal re-sequencing. In this paper experiences with a simple nonlinear music player (the nln-player), using this strategy, are being presented.

1. INTRODUCTION

Since its early beginnings in the seventies, video games have been equipped with interactive audio and music, often as part of the gameplay. These interactive sounds and music have played an indispensable role in many of these videogames, and reside in our collective memory [3].

The technological limitations of these early game platforms had impact on the way interactive music was capable of responding to the game experience, and this connection was often based on game parameters such as the remaining amount of time (e.g. Super Mario Bros, 1987)¹ or the number of enemies present on screen (as seen in Space Invaders, 1978)².

There are numerous ways of realising interactive music in games. Some of these methods are more complex in the part of the creation, and more demanding to the music engine, as is the case with some generative music. Other methods, like vertical re-orchestration, utilising 'stems' (or tracks) that are mixed according to a specific game-play variable, tend to consume more disk space [9, 6, 7]. This accentuates the notion that there is not an ideal method for using interactive music in games, as each method or combinations of methods can have specific value, but also disadvantages.

In this article various experiences with a practical and simple approach of horizontal re-sequencing, by means of a nonlinear music player, are discussed in the context of the video game *Shortburst*.

2. LEVEL OF INTERACTIVITY, LEVEL OF RESPONSIVENESS

The response to gameplay of some game music systems is almost immediate (e.g. Super Mario Galaxy³, Braid⁴). In other cases, musical form and gradual development of the composition are more important, causing the response to player interaction to be slower. When the music is implemented in such a way that it is not directly connected to or highly relevant for the gameplay, but merely present to support the narrative by adding a dimension of affect (more similar to the way film scores are used), a less responsive approach can be preferable.

Figure 1 shows a representation of the response in relation to the length of musical fragments. The shortest musical fragments (for example single notes) can shift from very small, for very responsive (almost immediate) interactive music to longer fragments with durations of a few seconds, for less responsive interactive music [10].

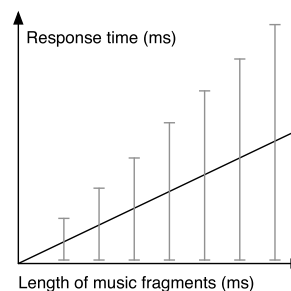


Figure 1. Duration of music fragments vs. response time

In general, and in parallel with film music, it is often not desirable that the music changes directly to the actions and on-screen narrative (the so-called 'Mickey Mousing') [8]. Still, in many games this immediate change of music as a result of gameplay events seems to be accepted even though this could be considered as musically inappropriate. In 2004 about 90% of all games, that has any interaction with the music involved, used an abrupt change in music, by fading in or starting a new sound-layer or -file [9].

An explanation for this fact can be that games have used these techniques for many years, and that players as

¹Super Mario Bros. (NES, 1987). Nintendo EAD, Nintendo.

²Space Invaders (1978). Taito Corporation, Midway.

³Super Mario Galaxy (Wii, 2007). Nintendo EAD, Nintendo

⁴Braid (Xbox 360, 2008, Windows, OS X, Playstation, 2009)

well as designers consider this way of using music apt in the game music tradition. Sometimes such direct transformation is also because of the direct, informative value the musical change can have, helping in the understanding of the gameplay. When, for example, the 1980 Pac-Man⁵, eats an 'energising pill' and becomes invulnerable, the 'music' is changed immediately up until the moment that the pill has worn out. This could be considered as the same effect as the direct fading in of music, that is seen in many more recent games, informing the player about the 'gamestate'.

In numerous video games the music is directly connected to user- or location-based events which can be perceived as rather abrupt. By connecting the music directly to the gameplay, part of the problem of maintaining the musicality while interactive, is solved.

Such a direct and, eventually, predictable music system can have the tendency of diminishing the immersion with the game [3]. The user can become aware of the system and what the outcome of the interaction will be, causing annoyance, often making the player mute all sounds or end the game.

3. HORIZONTAL RE-SEQUENCING

Horizontal re-sequencing, or cell based music, can be considered as a method that enables the reorganisation of the sequence of pre-composed musical fragments, instead of this sequence being fixed in advance. Some of the advantages associated with horizontal re-sequencing are considered to be its relatively limited requirements on data storage, the ability of limiting musical repetition and its reasonably fast responsiveness [7, 6].

The organisation of the musical material is often connected to, for example a certain variable or the timecode. In other cases the sequence can be rearranged randomly or by a performer, like the use in Karlheinz Stockhausen's 1956 Klavierstück XI [1], or the user and community as in Jason Freeman's Piano Etudes and Graph Theory [2, 4].

The precomposed musical fragments, that can be re-sequenced in the game music system during gameplay, can be connected to the game narrative as well. This can inform the player of a certain state in the game with a specific 'meaning'. This state or meaning can be explained as for example 'ambient', 'tense', 'romantic', 'a location' etc. In a same way this abstraction can be used not only for supporting a narrative but also to connect to a (gameplay) variable, making the musical meaning and narrative support 'interactive', as can be seen in Figure 2. The music can then be considered as being dynamically generated and interactive [7, 6].

Where the concept of horizontal re-sequencing can be considered to be very simple, practical applications of this technology are still rare. A complication with this technique in practice is the complex nature of the structuring of the musical material. A composer is not only required

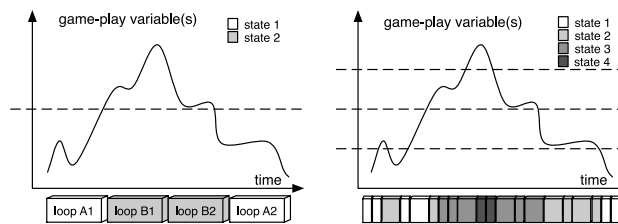


Figure 2. Cell-based music / Horizontal Re-sequencing, simple and a more complex example.

to create musical fragments, but, in order to make the system react in an appropriate manner to certain events, is also required to organise the musical material in such a way that alternative sequences of the fragments are supported. The organisation of possible pathways of the musical sequence should therefore be administrated by the composer. In the case of for example a Markovian approach this means that, for each musical fragment, possible successors are administrated in a separate data-file [5, 12]. When such an approach is undertaken, several problems are introduced, for example when a composer adds, changes or deletes musical fragments. The composer would then need to change the metadata of the musical elements involved and therefore needs to have an overview of all involved connections of successors and predecessors.

A musical cell-network also has the possible disadvantage of slowing down the responsiveness of the interactive music, when a fragment (or cell) that conforms to the requested state, is not available in the proximity of the current cell, as is depicted in Figure 3.

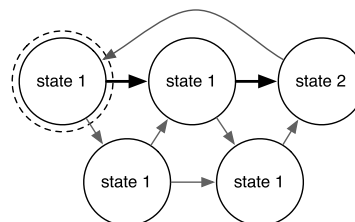


Figure 3. Simple cell-network in which more musical fragments are needed to proceed from state 1 to state 2, thus slowing down the responsiveness.

Next to this disadvantage, the search-engine that should find the most appropriate path towards a requested state is another risk of musical discrepancy [12].

Next to the already mentioned problems, the tools that are currently available for the creation, administration and testing of such musical cell networks are limited and not optimised for this specific approach.

4. INTRODUCING SIMPLIFIED CELL-NETWORKS

At the Utrecht School of the Arts, faculty Music and Technology, composition students were asked to create simple,

⁵PacMan (1980), Namco / Midway

interactive, compositions with a music system prototype⁶, based on Markov-based cell-networks [11, 12]. The best musical results were established when the students had a very good understanding of the metadata (or the 'architecture') of their composition or, in general, when the students had limited themselves in advance with the rule that any of their musical fragments should be able to connect to any other fragment.

The cell-network of a composition with this limitation would look rather complex in visualisations, but when such a limitation is introduced the cell-network data becomes redundant, since all musical fragments share the same possible successors. What remains in the metadata is the value of the interaction-variable or state, like the 'musical meaning'.

5. INTRODUCING A SIMPLE NONLINEAR MUSIC-PLAYER: THE NLN-PLAYER

As a step towards simple interactive music systems, but also to emancipate the use of, relatively easy to use, interactive music systems, a simple nonlinear music player for the web has been developed, with the working-title 'nln-player'. The web-version of the nln-player uses the concept of the 'simplified cell-networks' (as explained in Section 4) and was built with the Schillmania Soundmanager 2 library⁷, PHP and JavaScript.

The idea was to shift the focus from organising the musical material with - often complex - data-structures to a very simple model with higher limitations, but consequently requiring less administrative work and understanding of (meta-) data. This same framework has been used for the implementation of the interactive music in XNA⁸ for an Xbox game, Shortburst.

The only data that the composer is required to provide are:

- an id, or identifier, for example for the overall administration, or to identify the composer.
- a name, which normally would be a descriptive text, for the identification and understanding of the musical fragment.
- an exit-time, in milliseconds (ms), which is the time after which a new musical fragment should start. This value is implemented to allow the reverb tail of the music fragment to be played while the next musical fragment has already been triggered, preventing abrupt changes to be perceived by the player.
- one single variable, which is the master parameter that triggers the musical behaviour, for example a variable or state that connects to the musical meaning or gameplay-variable. In the online version of

the nln-player the requested variable is either controlled by the web-visitor by changing the value of the slider, or by the 'random-button', which randomises, in a brownian movement, the requested variable.

In the XNA, in-game, version, the requested variable is changed according to the progress of the game as explained in Section 5.2.

One of the aspects, that made working with this data-structure easy for the composer, was that the musical material had all required data stored in its filename. For example a filename would consist of: 007_ambient-outside-in-C_4000_1.mp3, the underscore '_' being the delimiter.

This data-structure is not proposed as the only data-structure for the nln-player, but as a simple default. More advanced features could be administrated in a configuration-file. In this configuration-file information and other metadata could be stored, regarding specific (groups of) musical fragments, making it possible to upscale the complexity of the music project and its system.

5.1. Shortburst

In this Section the music for the Xbox game Shortburst and its creation are discussed.

5.2. Shortburst - the Game

Shortburst is a simple one-level multiplayer game in which the game-variables can be summarised as follows:

- players first have to find each other in the game-world, for an 'encounter' to take place. This starting phase is the 'searching phase'.
- when an encounter is taking place, the players have to 'shoot' and 'hit' each other (and try not to be shot and hit). This is the 'encounter phase'.
- when a player is about to win, or a player is about to lose, this is considered as the 'action phase'.
- finally there is one winner that has taken the other players out, which is the 'victory phase'.

The narrative of the game is very simple, and the narrative direction is always very linear. The only non-linearity within the game is found within the timing of the narrative, which depends on the gameplay of the users. These aspects made this game very suitable for experimentation with the nln-player.

5.3. Shortburst - the Music

For the creation of the musical material three music composers worked together to create the entire game soundtrack. The goal was to create an interactive composition which matched the narrative variable of the game and would respond within a few seconds. Basically every level of narrative was scored by a different composer and for the

⁶2008 AMS-Toolkit developed withing the Adaptive Music Systems Research Group under Jan IJzermans

⁷<http://www.schillmania.com/projects/soundmanager2/>

⁸XNA is a set of tools with a managed runtime environment provided by Microsoft for computer game development

transitions between every of these levels transitions were composed.

When considering these transitions as a separate 'musical element', a sound fragment in this 'level' could only be played once, as is demonstrated in figure 4. It could also be considered that in this diagram the game variable goes from 1 to 3, 5 etc. and the music engine first plays a transition. Transitions are administrated by means of the configuration-file. Since this is an exception to the general concept of the simplicity of the nln-player it will not be discussed in depth.

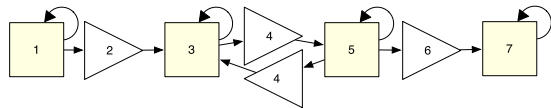


Figure 4. Diagram describing the difference between groups of musical fragments and transitions in the music of Shortburst.

The duration of the musical fragments varied from 6 to 13 seconds, which is also the maximum response time for the nln-player to respond to a changed variable.

6. CONCLUSION

The first experiments with the, simple nonlinear music player, the nln-player, sound very promising. The ease of use on the data-side of the interactive music makes the nln-player very accessible for composers with different skills and experiences with nonlinear music.

The constraint that all musical fragments have to be able to success any other fragment was found to be limiting for certain genres of music. For the music for Shortburst, however, this limitation was no problem, mainly because of the 'linearity' of the narrative (see figure 4).

The transitions, different from the regular musical fragments, were defined in a configuration file. This file contains extra information, other than the basic data extracted from the filenames. The idea of a configuration-file, which could be extended with more data regarding music and its interactivity, makes upscaling of the basic concept of the nln-player fairly simple. If a music system would for example require a Markovian approach on a cell-level, the configuration-file would also be logical for this metadata storage.

A single variable to control the interaction with the music can be quite limiting for certain applications. Because the nln-player is meant as a simple, interactive music-player, multi-variable approaches are not implemented in this version, but would technically be easy to realise.

When collaborating on one composition, the exchange of musical ideas and material between the composers can enhance the uniformity in the composition. By means of meetings and other communication between the composers musical ideas could very well be explained, but for the actual testing and experiencing of the musical material

(including the musical transitions) and the interaction, the web version of the nln-player proved very valuable.

For portfolio, demonstration, promotional and especially educational purposes an online version of such a simple interactive music player is considered as a very valuable tool.

The web-version of the nln-player, with the music of Shortburst, as well as more information and other nonlinear music, are available at www.nln-player.com and available for testing- and demonstration purposes. The source-code of the nln-player is available on request.

7. REFERENCES

- [1] W. de Ruiter, *Compositietechnieken in de twintigste eeuw*. De Toorts, Haarlem, 1993.
- [2] J. Freeman, "Graph theory: linking online musical exploration to concert hall performance," *LEONARDO MUSIC JOURNAL*, vol. 41, no. 1, pp. 92-93, February 2008.
- [3] S. Huiberts, "Captivating sound: the role of audio for immersion in games." Ph.D. dissertation, University of Portsmouth and Utrecht School of the Arts, Portsmouth, 2010.
- [4] *Graph Theory demonstration*. ICMC, 2009.
- [5] G. Loy, *Musimathics - The Mathematical foundations of music*. The MIT Press, 1997, vol. 1.
- [6] A. Marks, *Game Audio Development*. Delmar, 2009.
- [7] K. B. McAlpine, M. Bett, and J. Scanlan, "Approaches to creating real-time adaptive music in interactive entertainment: A musical perspective," in *The Proceedings of the AES 35th INTERNATIONAL CONFERENCE*. AES Royal Academy of Engineering, 2009.
- [8] R. M. Prendergast, *Film Music, a neglected art*. W. W. Norton, 1977.
- [9] G. A. Sanger, *The Fat Man on GAME AUDIO: Tasty Morsels of Sonic Goodness*. New Riders Publishing, 2004.
- [10] J. ter Borg, "Composing for a modern adaptive music system designed for use with games," Master's thesis, Utrecht School of the Arts, 2005.
- [11] B. van Dalen, *Adeptive Music v 0.9 beta introduction and manual*, Adeptive systems. [Online]. Available: <http://adeptive.nl>
- [12] T. van Nispen tot Pannerden, "Adaptable non-linear music for games," Master's thesis, Utrecht School of the Arts, 2008.