CBS: A Concept-Based Sequencer for Soundtrack Composition

Michael O Jewell, Mark S. Nixon and Adam Prügel-Bennett
Department of Electronics and Computer Science
University of Southampton, Southampton
SO17 1BJ, UK
moj|msn|apb@ecs.soton.ac.uk

Abstract

Existing methods of music composition vary from probability-based techniques to grammatical generation methods, yet the majority focus on the creation of independent pieces with no ties to other media. Our Concept-Based Sequencer (CBS) aims to use semantic markup of temporal media in combination with Genetic Algorithms to generate music that can fit scripted content such as film and radio. Genetic Algorithms offer an evolutionary approach which allows mutation but within a specified context. The concept here allows for linkage between media. This paper describes the techniques behind the CBS, and the construction of a web-based repository for concept mappings.

1 Introduction

Automatic composition of music has been an ongoing area of research since the 1950s, however focus has typically been directed toward the creation of independent works. Music is often created to express concepts, whether the motivation is an image (such as the imagery in Mussorgsky's Pictures at an Exhibition), an emotion (such as Mozart's Requiem), or a representation of the action present in a scene, such as in opera, musical, cartoon, or film. Current methods rarely take these conceptual influences into consideration, and so we have been developing a Concept-Based Sequencer (CBS) which aims to add the idea of concept to traditional automatic composing methodologies, hence adding a sense of direction that is otherwise missing from artificial music. Furthermore, the CBS framework is highly modular, and thus could be applied to temporal media other than music and film.

Current fields of music composition can be roughly split into four categories:

Rule-Based Systems that are rule-oriented provide a set of strict guidelines that the resultant music should follow. For example, when creating music with the same style as Bach or Mozart, parallel fifths and octaves are discouraged[9]. Schottstaedt used a penalty-based approach[10], which could be adjusted to be more or less lenient on specific rules.

Stochastic Covering a large portion of the early composition programs, stochastic algorithms typically use Markov Models to represent movements between states. While these have been used to produce very aesthetically pleasing music[2], Markov Models have the disadvantage that they require extensive training, and cannot synthesise beyond this configuration.

Microsoft have also made use of a stochastic technique in their DirectMusic system[7], specifically when describing chord progressions. This idea of connected chords is combined with a user-created collection of part variations to generate real-time music for games.

Grammar-based Grammatical approaches to composition encode musical elements as grammar symbols within a string, which is then manipulated using rewriting rules. McCormack[6] used a variation of this, known as L-Systems[5], to represent the repetition and self-similarity of musical phrases, which he then further extended to simulate Markov Models by assigning a probability to each production. Furthermore, hierarchical rules can be used to better represent musical structure, as there is often an element of similarity between piece, section, and phrase.

Genetic Based on the framework created by Holland[4] in 1975, Genetic Algorithms are modeled on Darwinian evolution, with selection handled by a 'fitness function' which evaluates each solution in the population and creates a subsequent genotype. Each solution consists of a collection of chromosomes, which are further made up of numerous 'allele'. These are the smallest structures in the system, and hence are often used

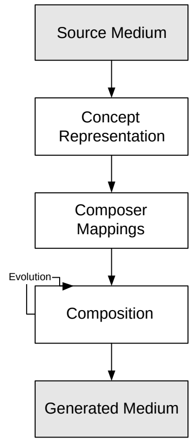

Figure 1. The structure of the Concept-Based Sequencer.

to represent notes when applied to music. Two basic operators, ‘crossover’ and ‘mutation’ are available, which swap and alter allele respectively. Genetic Algorithms have been applied to harmonisation[11] and jazz[8], but these do not synthesise with context and instead use static sets of rules in the fitness function.

The concept-based sequencer currently applies techniques from the stochastic style of composition, but also uses a genetic algorithm approach to diversify away from the Markov chains. We describe the three levels of our CBS system, and how these levels aim to tie conceptual information to a musical representation (See Figure 1).

2 Concept-Based Sequencing

2.1 Concept Representation

The ‘concept representation’ layer of CBS creates a semantic glue between the temporal media and the effects that the media should have at any point in time. The model is not restricted to a video-to-music system, and could be layered between many other types of media assuming a mapping layer is provided. For example, the system could essentially be reversed and used to manipulate video in response to music. Alternatively, the layer could be used as an indexing tool for temporal media, with the ability to search on items containing certain moods, objects, colours, or any other concept that can be described within the system.

The CBS concept model consists of 2 node types:

Event An event represents a significant occurrence within the media, typically involving several participant concepts. Each participant in an event and the event itself have a start and end time, and participants have a ‘presence’ parameter, which specifies the contribution of that participant. For example, if one actor is more significant than another, it will have a higher presence value. The presence values are later used by the composer mapping to select the influence of each concept on the synthesised music.

Concept A concept can represent an item, an attribute, a mood, or any other named entity within the system. It can be a specific instance of another concept (‘Civilian 1 `instanceof` Civilian’), a subclass of another concept (‘Civilian `isa` Man’), or an attribute that is applied to a concept (‘Civilian `hasattr` Nervous’). Concepts are purposefully generic, with a name being the only required information. A hierarchical organisation structure is planned, which will allow categorisation to avoid name clashes (for example, ‘blue’ the colour and ‘blue’ the mood).

Each event can be linked to subsequent event nodes using a temporal arc, and participants can specify a spatial envelope that represents the distance between two participants. This envelope is a fuzzy variable, and hence it is possible to describe ‘Actor 1 is near Knife’, or ‘Tortoise is far from Finish Line’. Each spatial arc can specify two concepts, thus allowing tweening from ‘relaxed’ to ‘intense’, or ‘happy’ to ‘sad’ based on the distance between two objects.

The final concept representation will be separated into two views: a sequencing view, which shows the timing and presence of participants in an event, and a semantic view, which shows how the participants in an event are constructed from individual concepts. There is also a possibility that the sequencing view could be abstracted further, and show more detailed information about the relationships between concepts. Figures 2 and 3 show examples of the two views, with the first being a typical sequencing view and the second showing a simple semantic view. The example is taken from Scene 2 in Casablanca, where two policeman question a nervous civilian about his papers, with the civilian running in panic when they are identified as forgeries. The graphs illustrated within each participant show an interpretation of the presence of that participant at that moment in time (typically when the individual is talking in this example).

Figure 3 shows the semantic view for the Civilian 1 concept, with the three types of relationship applied. The Civilian class inherits the functionality of the Man concept, as well as any composer mappings that were applied to Man. The `instanceof` operator is used here to create a specific nervous civilian (Civilian 1), but creating new instances would allow many civilians, all with different characteristics.

Figure 2. The sequencing view of a scene from Casablanca.

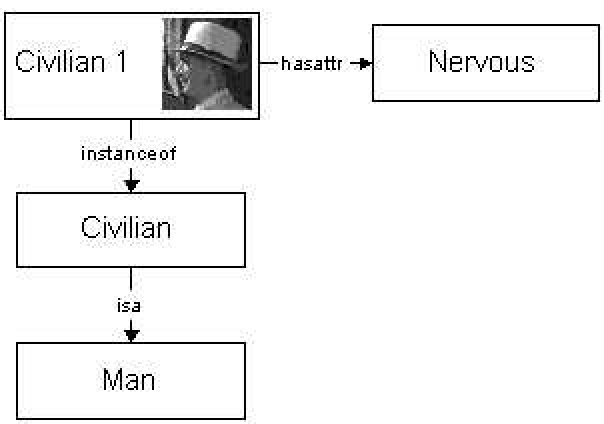

Figure 3. The semantic view of the Civilian from Figure 2.

2.2 Composer Mappings

Whilst a concept representation provides an abstract view of a scene, the interpretation of the specified concepts varies from composer to composer. By way of example, Angelo Badalementi (the composer for many of David Lynch's films) often uses styles that deliberately go against the concepts in a scene, such as adding a slight major feel to a death scene, whereas Hollywood composers use more traditional techniques, with John Williams' 'Jaws' employing a repeated chromatic motif to represent the threat of shark attack.

To preserve the idiosyncrasies of composers, the composer mapping is used. This is overlaid on the concept representation, and provides a set of 'musical modifiers' for the scene. For example, a 'happy' scene may be defined to use a major pitch mesh with violins and flutes providing a light texture, whereas an 'unhappy' scene may use a minor pitch mesh with 'celli and clarinets. If several pitch meshes are specified for an event, the presence level is used to weight the final composite mesh, which could, for instance, allow minor characteristics to come through the texture if a villain enters the scene. Each concept can contain pitch, key, instrumental, rhythmic, timing, texture, and dynamic information, as well as specifying whether it should have a motif (which is kept for the entire score, although it can be altered via the other parameters).

2.3 Music Composition

At the core of the CBS system is a genetic algorithm that aims to evolve music to fit the constraints passed down by the prior levels. The technique used is similar to that implemented by Wiggins [11], with each note corresponding to an allele within the chromosomal string. However, whereas Wiggins uses four strings of equal, fixed length, the CBS implementation stores the part and time of each note within the allele. This allows for polyphonic as well as homophonic music and is more efficient when creating large scores, as it is not necessary to store information when no notes are present.

The fitness function used by the GA is unique, as it is designed to be highly dependent on the modifiers provided by the composer mappings. Consequently the only required constraint is that the piece of music should fit within the time allocated. A few realistic defaults will be set to provide reasonable note densities and intervals, but these can be overridden entirely by the composer mappings. As a result, the system can be expanded to handle unconventional scales and techniques.

To encapsulate the musical requirements of the composer, several data structures are necessary. For pitch and key change transitions, Markov representations are used.

For the former, this provides the probability of moving to a new pitch from the current note, while the latter provides the probability of moving to a new key. Instrumentation, timing, and dynamics will be selected via a probabilistic representation, as these remain constant throughout individual concepts.

Texture and rhythmic information are more complex, as they depend on several parameters within the music itself. Benjamin[1] proposes that meter is created using not only accents and grouping (although these are the primary techniques), but also through melody and harmony. The texture of a piece of music can consist of a single melodic line (monophonic), several melodic lines (polyphonic), or a single melodic line with accompanying harmony (homophonic), but can also use a 'call and response' style (antiphony) or an echoing effect (heterophony). Whilst heterophony is generally used in non-Western music, allowing a representation of texture that is suitably generic to cover all variations is important, and this is an area of future research.

CBS uses a binary tournament selection operator, with two chromosomes from the genotype selected at random to be parents and the fittest of these inserted into the subsequent population. This selection is repeated over the genotype until the new population is created. Each child can also be affected by a mutation operator, which increases or decreases the pitch of a random note by one semitone, or by the crossover operator, which uses a multi-point technique to swap the allele between two bounds. While it is possible to use additional operators to provide constraints on the music, CBS aims to encapsulate these within the dynamic fitness function, thus allowing increased flexibility between composer mappings.

3 Web Interface

The initial testbed for the CBS project will consist of two Java applications; one designed for web-based use and the other as a desktop application. Both applications will provide functionality for the viewing and modification of the semantic and composer mapping levels, as well as for the generation of music, but only the desktop application will allow the creation of sequencing views, as video manipulation is too heavyweight a task for a web-based application. Individual components of these applications, such as the genetic approach, are already showing promise.

All information about semantics and composers will be held online in a global repository, and both applications will be able to make use of this as an extensive toolkit of conceptual information and styles. By allowing access to modify all of the information in the repository, it is hoped that the data will be refined and expanded by users as limitations are detected. As the system will store characteristics of many composers, it will also provide invaluable information for further research, such as how composing styles alter depending on genre, time, and location. Several XML schemata are currently under development to provide an easily processed representation of the semantic markups and models, and we aim to use MusicXML[3] as a final output format.

4 Conclusions

A new concept-based sequencer allows for an exciting fusion of media, which we believe will result in music that provides a more realistic interpretation of mood and emotion than existing techniques. Whilst genetic algorithms have been applied to music generation prior to CBS, the added level of semantic information will extend the model to allow highly flexible interpretative composition. The layer-oriented model also allows translation to other areas, including multimedia indexing, animation, and video creation. The applications for this new approach are wide-ranging, with cost-effective film scoring being the most obvious example, however the system could also be used to create independent music or backing scores for other media.

References

[1] W. E. Benjamin. A theory of musical meter. *Musical Perception*, 11(4), 1984.

[2] M. Farbood and B. Schoner. Analysis and synthesis of palestrina-style counterpoint using Markov chains. In *International Computer Music Conference*, 2001.

[3] M. Good. MusicXML: An internet-friendly format for sheet music. In *XML Conference and Expo*, 2001.

[4] J. H. Holland. *Adaptation in Natural and Artifi cial Systems*. Ann Arbor: University of Michigan Press, 1975.

[5] A. Lindenmayer. Mathematical models for cellular interaction in development, parts I and II. *Journal of Theoretical Biology*, 16(2), 1968.

[6] J. McCormack. Grammar based music composition. In *Complex Systems*, 1996.

[7] Microsoft Corporation. Composing music for interactive titles: An overview of DirectMusic Producer. 1998.

[8] G. Papadopoulos and G. Wiggins. A genetic algorithm for the generation of jazz melodies. In *STeP'98, Jyväskylä, Finland*, 1998.

[9] S. Phon-Amnuaisuk, A. Tuson, and G. Wiggins. Evolving musical harmonisation. 1999.

[10] W. Schottstaedt. *Automatic Counterpoint*. MIT Press, 1989.

[11] G. Wiggins, G. Papadopoulos, S. Phon-Amnuaisuk, and A. Tuson. Evolutionary methods for musical composition. *International Journal of Computing Anticipatory Systems*, 1999.