Algorithmic Composition in Contrasting Music Styles

Tristan McAuley, Philip Hingston

School of Computer and Information Science, Edith Cowan University
email: mcauley@vianet.net.au, p.hingston@ecu.edu.au

Abstract

The aim of this research was to automate the composition of convincingly "real" music in specific musical genres. By "real" music we mean music which is not obviously "machine generated", is recognizable as being of the selected genre, is perceived as aesthetically pleasing, and is usable in a commercial context.

To achieve this goal, various computational techniques were used, including genetic algorithms and finite state automata. The process involves an original, top down approach and a bottom up approach based on previous studies. Student musicians have objectively assessed the resulting compositions.

1 Introduction

The intention of this research is to provide a fully automated compositional tool enabling the generation of a song conforming to the desired genre. We created a system, Gen^3 (Genetic Genre Generator), that provides an alternative to hiring a costly human composer when acquiring quality music for use in short films, computer games, or multimedia presentations.

To achieve the said aim, music composed must exhibit the following qualities: it must be seemingly of human origin, i.e. not obviously "machine generated"; it must be recognizable as belonging to the desired genre; and it must be aesthetically pleasing (a very subjective concept). Further, generated compositions are required to be valuable in a commercial context. It is also intended that the software be of aid to composing musicians.

The structure of the paper is as follows. In section 2, we outline our two-pronged approach to the problem of automated composition, which models both local and global characteristics of songs. In the following sections, we describe the techniques employed by Gen^3 to generate songs in contrasting genres, Trance and Bossa Nova. Sections 3 and 4 detail the methods we use for local and global modeling respectively. In section 5, we describe how these local and global levels are brought together. We also describe, in section 6, initial evaluation results from a pilot study, in which we subjected the resulting compositions to evaluation by student musicians. The acquisition of an objective assessment of the software's capabilities is an important aspect of this research. In section 7 we discuss our findings and finally outline possible future work in section 8.

2 Background and Approach

Algorithmic control of music predates the invention of computers, however, since the introduction of computers, more elaborate projects have been undertaken (see Järveläinen 2000 for an overview of algorithmic composition or Papadopoulos and Wiggins 1999 for a greater focus on Artificial Intelligence methods). Hiller & Isaacson are credited with writing the first algorithmic composition on a computer, 'Illiac Suite for String Quartet', in 1956 (Music Library, U. of N.Y., Buffalo 2003). Approaches to algorithmic composition to date vary greatly and include rule-based, probabilistic, connectionist and evolutionary approaches. Interests within this field include the provision of automated tools for transforming, refining, or extending melodies, and generating partial or complete songs.

Our approach to this research views songs as comprising of both a local and a global abstraction. The local abstraction describes lower level characteristics, such as note sequences and note attributes. A grammar-based technique is utilized to represent these local characteristics. The global abstraction addresses the form of the song. A novel "instrument activity table" is evolved to capture the macro-structure of a song.

Extensive research exists that explores grammar-based approaches to modeling local characteristics. Clement (Clement 1998) has studied the ability of a Markov model to learn harmonic progressions in different musical styles. He forms the opinion that regular grammars are adequate for representing lower level musical characteristics. Therefore, in this study, we used finite state automata to model local characteristics found in the desired genre.

We also employed modification techniques to vary note sequences in a way that conforms to the musical constraints typical of the selected genre. We used techniques such as regeneration and morphing to emulate integral aspects of music such as improvisation.

Little success has been achieved, to date, in implementing a global structure within a song. The "learning" of a subgenre's distinctive macro-characteristics has proven to be difficult. To address this problem, we developed the concept of an "instrument activity table" and used simulated evolution to incorporate these characteristics. This table facilitates the integration of both local and global characteristics and orchestrates the application of note modification techniques. A critic, comprised of rules founded in the theory associated with the genre, guides the evolutionary process.

3 Local Characteristics

In this section, we describe how lower level characteristics common to a specific musical genre are "learned" from sample note sequences extracted from songs typical of the genre.

The technique we used to learn these local characteristics involves finite state automata. A separate finite state automaton was used to model each note attribute sequence contained in the sample data. Once the automata have been found, note attribute sequences such as pitch or velocity sequences, can be independently generated and combined to produce new note sequences for new compositions.

3.1 Finite state automata

A finite state automaton "is a mathematical model of systems which have a finite number of internal states and respond to external world just by changing their internal state" (Alberti, Marelli, and Sabadini 1993). The finite state automata used in this research were probabilistic, that is, a transition between states may not be uniquely determined, and a probability is associated with each possible transition between states. Output is generated for each state traversed (making these Moore machines). Figure 1 shows an initial segment of such a finite state automaton (further states and transitions extend out to the right). Starting with the leftmost state, the first symbol, -1, is emitted, and then a transition to the next state is chosen using the probabilities shown on the transition arcs (this initial symbol is ignored in subsequent processing). Supposing that the arc leading to the state labeled 104 is chosen, the symbol 104 is emitted, and the next transition chosen, and so on, until a sequence of the desired length is emitted. Examples given illustrate this process.

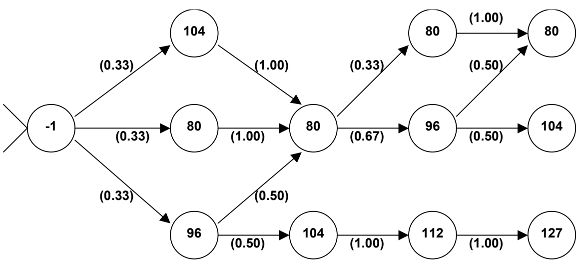

Figure 1 - Initial segment of a probabilistic finite state automaton

3.2 Deriving finite state automata from sample data

How are suitable finite state automata to be found? Sample data consists of note sequences of a fixed length (two bars) extracted from each instrument. These note sequences are decomposed into four components, a pitch sequence, a position (relative to the two bars) sequence, a sequence of velocities, and a duration sequence. Possible velocity sequences could be 104, 80, 96, 104, ... or 96, 80, 80, 80, ..., for example. A finite state automaton was derived for each note attribute for each instrument.

Each state holds a note attribute value, a sequence number within the original sequence, and a list of links to other states and their associated probabilities. Extracted note attribute sequences are processed in the following way.

A dummy state with attribute value -1 is created as position zero. The processing of each note attribute sequence commences from this state. Note attribute sequences are sequentially accessed and the position of each note attribute determined. When a new note attribute and position combination is encountered, a new state and a link to this new state from the current state are added. If the note attribute and position are not new, a link to the existing matching state is created. Figure 2 graphically depicts this process starting with the two attribute sequences listed above. Two states labeled -1 and two states labeled 80 are "merged" as they have the same attribute value and sequence number. Note that the automaton is capable of generating not only the two seed sequences, but others as well (e.g. 104, 80, 80, 80...). The same idea is used when there are more than two attribute sequences to work from.

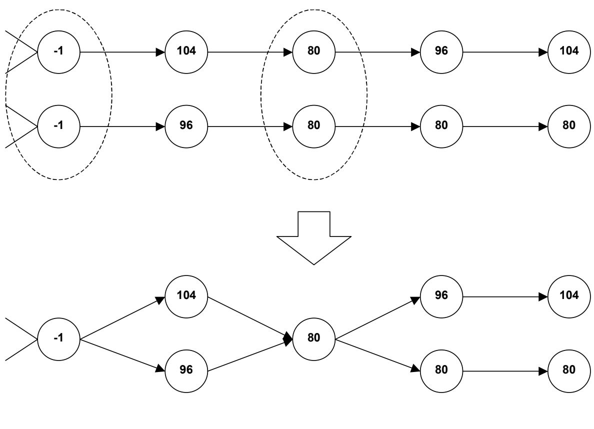

Figure 2 - Creating a finite state automaton from two attribute sequences

The probabilities associated with each link are determined last. The probability of a link being traversed is the inverse of the number of links from that state. Multiple links to the same state are possible.

3.3 Note sequence generation and modification

Note attribute sequences or “phases”, each two bars in length, are generated by the four finite state automata, which separately model the sample position, pitch, velocity, and duration sequences. These sequences are combined to form the initial note sequences for their respective instruments. The assumption is made that note attribute sequences, independently generated from the sample data, may be combined to produce note sequences representative of the sample data.

To effectively model characteristics of instrument voicings found within the genre, we subjected these initial note sequences to various transformations to generate successive phases or the song, after taking into account global characteristics. Note sequence transformations utilized include regeneration, partial regeneration and morphing.

Regeneration (using the same automata to generate a new set of four sequences) was applied to achieve a new note sequence still retaining characteristics of the sample data. Partial regeneration of a note sequence, that is, retaining at least one of the four note attribute sequences and regenerating the others, provides a more subtle variation. The regeneration of a note sequence was subjected to generic musical constraints, such as disallowing excessively large intervals, to provide continuity with the previous note sequence. Morphing is a process where “in-between” versions of two phases are calculated in the same way that “tweening” is done in animation.

4 Evolving Global Characteristics

As well as local sequences of notes, a song has an overall global structure, consisting of phases during which different instruments are active. We introduce the concept of an ”instrument activity table” to represent this global structure. Each column within the table denotes a phase, of fixed time length, of the song. Each row models the activity of instruments during this phase in the song. Thus, a cell contains information regarding an instrument's presence at the corresponding phase of the song.

4.1 Instrument groups

The instrument activity table represents the global structure of the song. To achieve a sufficiently abstract view, instruments sharing similar behavior traits are grouped together. The formation of these instrument groups results in an instrument activity table that is more responsive to the genetic algorithmic process, in that multiple, different instrument activity patterns were more easily formed. This instrument grouping also requires an additional processing stage involving the expansion of the table to include all instruments.

4.2 Genetic algorithms

Here we give a brief introduction to genetic algorithms (for a more complete treatments, see e.g. Goldberg 1989). A genetic algorithm models the natural biological evolutionary process, evolving a population of potential solutions to a complex problem.

A population consists of encoded representations (analogous to genes) of potential solutions. Ultimately, an individual representation is translated back into the problem domain as a possible solution. This population is evolved through the application of evolutionary operators, typically cloning, mutation and crossover, based on biological cloning, mutation and crossover. A critic that evaluates the *fitness* of each population member determines the survival prospects of a particular solution. Individuals receiving a high fitness value are favored in survival and reproduction. Over successive generations, fitter and fitter solutions are found.

In outline form, a genetic algorithm consists of the following steps:

1. Create an initial population of solutions
2. Evaluate each solution and assign it a fitness value
3. Select “parents” of the next generation of solutions based on these fitness values
4. Generate “children” from these parents using crossover and cloning, and mutation.
5. Repeat steps 2-4 until finished.

To specify our genetic algorithm, we must describe how solutions are represented (the coding scheme), how fitness values are calculated, how selection is done, how crossover, mutation and cloning are performed, how to decide when to stop, and how to translate the solutions back to the problem domain.

Coding scheme. A population of 40 instrument activity tables was created. Each table is represented as a 2 dimensional array of Boolean values. On construction, cells within the instrument activity table are set according to predefined “construction” rules. The rules, derived from analysis of the genre, instill a fundamental macro pattern common to the selected genre. These rules are enforced by the genetic algorithm and remain intact in the solution instrument activity table.

Fitness evaluation. A rule-based approach is used in the evaluation of an instrument activity table's fitness. The table is subjected to criteria formed from analysis of the genre. Criteria examine the instrument activity table for evidence of instrument conformance, typical song dynamics, and characteristic instrument inter-dependency. Activity arrangement was examined for fluidity, typical form, and instrument activity sequences conforming to the style. Implementation of the criteria is in the form of functions that return a bonus, or penalty, based on the arrangement of activity within the table. For example, in the Trance genre, each phase should involve both percussive and melodic instruments. If a generated song violates the rule given above, it is assigned a penalty of -1 each time the rule is violated. Other violations also attract penalties. After the population was subjected to 500 generations of evolution, the best individuals were found to have successfully satisfied all constraints.

Selection. An "elitist strategy" is employed to ensure survival of the best solutions - the top two fifths of the population is cloned and becomes part of the next generation. The parents for the remaining three fifths of the next generation are selected using "roulette-wheel" selection, in which parents are selected one at a time in independent trials, with selection probability proportional to fitness.

Crossover and mutation. After a pair of parents is selected, two-point vertical, and horizontal, crossover occurs between parents resulting in the production of two children. The diagram below illustrates the formation of each child during two-point horizontal crossover.

To further clarify, both point 1 and point 2 are randomly chosen but remain consistent between the two randomly selected parents. New points are randomly generated until a valid crossover, one that does not violate instrument activity table construction rules, is possible. In Figure 3, Child 1 is formed by the combination of Parent 1 - section A, Parent 2 - section B and Parent 1 - section C. The inverse applies in the formation of Child 2.

In addition to cloning and crossover, the genetic material of the children may also be subjected (with low probability) to mutation. Mutation involves the switching of the status of one instrument group within a phase. For example, if a phase with an active element group is mutated, its status will be switched to inactive. Mutation is only applied to children produced by crossover and the bottom half of the clones. To avoid corruption of the fundamental macro patterns instilled on construction, the mutation process will fail and retry on violation of the construction rules.

		Parent 1									Parent 2								
A	arpeggio	*	*	*	*				*	*	*	*					*	*	*
	melody					*	*	*	*				*	*	*	*	*	*	
Point 1																			
B	bass			*	*	*	*	*							*	*	*	*	
	bass drum	*	*	*	*	*	*	*	*				*	*	*	*	*	*	
	hi-hats		*	*	*			*	*	*	*	*	*	*	*				
	snare		*	*	*	*		*	*	*		*	*	*			*	*	*
	percussion	*	*	*	*				*	*	*	*	*						*
Point 2																			
C	strings	*	*			*	*	*	*	*					*	*	*	*	
	sound					*	*	*	*				*	*	*	*	*	*	
	build		*	*	*	*	*	*					*	*	*	*	*	*	

		Child 1									Child 2								
A	arpeggio	*	*	*	*				*	*	*	*					*	*	*
	melody					*	*	*	*				*	*	*	*	*	*	
Point 1																			
B	bass					*	*	*	*				*	*	*	*	*		
	bass drum			*	*	*	*	*	*		*	*	*	*	*	*	*	*	
	hi-hats	*	*	*	*	*						*	*	*			*	*	*
	snare		*	*	*			*	*	*		*	*	*	*		*	*	*
	percussion	*	*	*						*	*	*	*	*				*	*
Point 2																			
C	strings	*	*			*	*	*	*	*					*	*	*	*	
	sound					*	*	*	*				*	*	*	*	*	*	
	build		*	*	*	*	*	*					*	*	*	*	*	*	

Figure 3 - Horizontal two-point crossover on activity tables

Expanding the solution. The product of the genetic algorithm is a population of instrument activity tables that adhere to the construction rules and display characteristics formed by the process and guided by the evaluation criteria. The best (highest fitness) solution found after 500 generations is selected. This solution must undergo an expansion process to encompass all element activity and retain the instilled features. Instrument groups are expanded to specify activity of each composite instrument. Also at this time, note sequence modification techniques are integrated into the expanded instrument activity table.

An instrument group is expanded phase by phase. Rules, similar to those derived for the critic, determine which instrument(s) is active. Factors that effect the selection include the activity of the composite instruments in previous phases, the size of the current and previous phase, and the position of the current phase. For example, the logic behind determining which hi-hat(s) are selected is: If both hi-hat closed and hi-hat open were active in the previous phase and the following phase contained a significant drop in the number of active instruments then only the hi-hat closed would be selected.

5 Putting it all together

Here we describe the overall process of creating a song, starting from a set of examples of the genre. Music data supplied is processed into formatted, separate files. Four files are created for each instrument to be modeled, a pitch, position, duration and velocity file. Each file holds multiple sequences of each note attribute for that specific instrument.

Finite state automata are used to model the note attribute data contained in these files. Four finite state automata are linked with each instrument, one for each data attribute type. Output from these four finite state automata is independently combined to produce an initial note sequence for their corresponding instrument.

Using the critique measures discussed, an instrument activity table is evolved. This table describes the form of the song in terms of instrument group activity. Instrument groups are groups of logically related instruments. For each phase (two bars) of the song, a Boolean value indicates the presence of the corresponding instrument group. The resulting instrument activity table is expanded to include all instruments. Note sequence modification techniques are configured into the table indicating when and which technique is to be applied for each instrument.

The construction of the song involves the column-by-column traversal of the resulting expanded instrument activity table. If the instrument is active, the current note sequence is retrieved from this instrument, converted and stored in the appropriate track. Any note sequence modification techniques, such as morphing and regeneration, are applied directly. A computer can then play the resulting song.

6 Evaluation of the Results

To evaluate the resulting compositions, a pilot survey was designed and conducted with student musicians. The aim of this survey was to objectively assess the degree of success attained for each objective. Two questionnaires were designed to measure each aspect of the primary objective, fully automated composition, and the secondary objective, a compositional aid for composing musicians. The survey questions are listed in the Appendix below.

A sufficient amount of data (survey answers from 15 students) was obtained to provide an indication of how well Gen^3 fared in achieving its goals. The necessary conversion of generated songs into a machine-readable/playable format was both imprecise and induced a negative bias. However, results revealed crucial flaws and strengths within the algorithmic process.

6.1 Obtaining an objective critique

Auditioning of the resulting pieces required a translation from the internal representation of the song into a machine-readable format, MIDI. A faithful translation between formats proved to be complicated.

Decisions made, such as instrument designation, had profound effects on the rendition of the resulting song. The following issues prevented the achievement of complete accuracy in the translation. Performance, by instruments assigned, is imprecise. Instruments contain individual attack and decay settings, which affect the start and duration times assigned to each note. The translation process did not retain instrument settings, such as chorus and reverb. Additionally, pitch slurs and other nuances were not retained. Instruments assigned to each track were selected from a restrictive list. For example, the 'muted electric guitar' was assigned as the instrument representing a synthesizer.

Seven consecutively generated songs were generated in each musical style (Trance and Bossa Nova). These songs were auditioned to participants, who were required to provide instinctive responses to each question. The tactic of utilizing fewer songs and more participants was employed. It was anticipated a more accurate measure would be gauged from the common trends that emerge from the multiple evaluations.

The design of a survey addressing both our research questions and technical details of the software process was difficult. A compromise between musical comprehensibility and the distinct addressing of each technical aspect was needed. Certain technical features of Gen^3 could not be individually addressed. The resulting questions shared blurred boundaries and relationships with each other.

6.2 Trance

Trance styled songs achieved a greater level of success than those generated in the Bossa Nova style (and hence we have less to say about it). Results from the questionnaire, written comments, and verbal feedback all reinforce this conclusion. The aspect of the Trance songs that performed the least well was the aesthetic ratings.

Feedback regarding the form of the generated Trance song varied greatly between each song. This aspect of a song received the lowest mean value recorded for a song in either genre. However, success was evident in comments supplied with the song evaluation questionnaire. The large variations between songs might be due to large differences in the instrument activity tables produced by the genetic algorithm.

6.3 Bossa Nova

Songs generated in the Bossa Nova style were consistently rated lower then those generated in the Trance style. This was indicated through responses to Likert styled questions, open-ended questions and informal verbal feedback. Bossa Nova songs that were auditioned performed poorly in instrument interaction, musical phrasing, and the fluidity of the piece. Also, average aesthetical feedback was negative for three of the seven Bossa Nova songs generated.

The following statement, "*Instruments fulfill roles typical to the genre in terms of: Interaction with other instruments*", was designed to obtain critique regarding instrument interaction. No song generated in the Bossa Nova style received a 'Strongly agree' response to this statement. Under a third of the participants provided a positive response.

Musical phrasing refers to how musical sentences are performed. A response similar to that directed at instrument interaction was obtained from the participants. Additionally, verbal and written feedback criticized the melody as sounding "*very random and disjointed*". Similar criticism was directed at improvising instruments. Poor timing and rhythm, and lack of expected chordal relationships, were attributed as the cause.

The statement "*Instrument interactivity results in a smooth, continuous flow*" resulted in the lowest average rating assigned by participants to an aspect of the Bossa Nova songs. Again, under one third of responses were positive and no song received a 'Strongly agree' response. A song's aesthetical rating has a stronger relation to the measure of a song's fluidity than the measure of a song's technical conformance.

Despite these weaknesses, one of the seven songs achieved a reasonable appraisal. Rhythmic properties and recognisability to the genre were accomplished reasonably well overall, and very well in the song mentioned.

7 Discussion

7.1 Accomplishment of objectives

As discussed previously, the primary objective of this research is the automated composition of convincingly "real" music in specific musical genres. "Real" was clarified as meaning not obviously machine-generated, recognizable to the selected genre, aesthetically pleasing, and usable in a commercial context. A secondary objective of this research is to provide composing musicians with a valuable compositional aid. Due to the varying degrees of success by each genre, accomplishment of objectives by each genre is discussed separately.

Bossa Nova. Based on results from the survey, songs generated in the Bossa Nova style contain fundamental technical shortcomings. A Bossa Nova song must exhibit a warm, human-like quality. Both the translation to MIDI and the auditioning by MIDI instruments detract greatly from the song's potentiality.

Bossa Nova songs were judged by participants to be amateur. The overall, aesthetical response to these songs was negative. This said, one of the seven songs achieved a reasonable review. This may be attributed to the possibility that shortcomings were not as evident within this song. The diversity of responses indicated a varying ability, by participants, to perceive these shortcomings.

The primary objective of providing non-skilled people the ability to generate songs in the Bossa Nova genre was not successfully met. Fundamental deficiencies, though not distinctly obvious to the untutored ear, prevent the generated songs from being technically proficient.

The secondary objective of assisting musicians in the composition of music was more successfully accomplished. This was indicated by the responses addressing this intention in the survey. When used as a compositional aid, the identified shortcomings become less evident.

Trance. For several reasons the shortcomings identified in the Bossa Nova genre were less apparent in the Trance genre. Gen^3's approach lent itself to the generation of Trance style songs, perhaps because of the relative simplicity of Trance songs, and the machine-like precision of note placement that is typical of the genre.

The translation into MIDI did not adversely affect the song's evaluation to as great an extent as for Bossa Nova songs. However, the audition of the song by the restrictive GM instruments probably limited the song ratings.

In the case of automated Trance styled song generation, the primary objective of automated composition was successful to a degree. Gen^3 was often able to generate music that conformed to the specifications set - that is, not obviously machine generated, recognizable to the genre, aesthetically pleasing, and usable in a commercial context (the latter requirement not being directly tested). The secondary goal of assisting a musician in composing was also considered to be a moderate success.

7.2 Evaluation of techniques

A comprehensive assessment of the techniques employed was difficult to achieve due to the indirect nature of the survey questions evaluating them. A compromise between musical comprehensibility and distinctly assessing technical aspects resulted in blurred boundaries and relationships between the questions.

However, in both genres it seemed note sequences accurately reflected the characteristics found in the sample data. The independent generation and combination of note attribute sequences generated by finite state automata seemed to reflect characteristics found in the sample data. Comments and question responses pertaining to these aspects were generally positive. Problems arose from the lack of interaction between the note sequences rather than the note sequences themselves.

Note sequence modification techniques achieved varying levels of success. Regeneration or morphing techniques used in the generation of Trance songs did not attract any negative comments. A high appraisal of instrument conformance was received from participants. However, the simple technique of selective regeneration used in the Bossa Nova style did not effectively model an improvising instrument. The overriding message conveyed was that the phrasing was too random. Theory was not conformed to, ideas were not connected, and variation on the melodic phrases was not incorporated.

The effectiveness of the evolutionary process in producing a global structure accurate to the genre was somewhat erratic. It is expected that further time spent on refining the set of evaluation criteria, through trial and feedback, will improve the consistency and peak performance of the genetic algorithm. The assumption made that “two bars will provide enough precision to accurately model instrument activity” is invalid in the context of the Bossa Nova genre.

8 Future Work

The songs auditioned in the survey were the first seven songs generated in each genre. Some failings identified by participants of the survey may not appear in subsequent generations of songs. From this perspective, the project’s flaws may be classified based on the technical feasibility of achieving a “correct” solution with the current version of Gen^3. The following are aspects of the project that require improvement or correction to the software.

The software process does not learn, or instill, the important instrument interaction that is typical of the selected genre. The independent combination of instrument note sequences does not effectively model the intricacies of inter-instrument dependence. This interaction must be learnt from supplied song files and requires a dependency between note sequences being established.

Vital relationships between melodic phrasing, or improvisation, and the chord sequence are not present. This may be due to the inaccuracy of the note sequence transpositions, the segmentation and subsequent combination of melodic phrases, and an inability to learn this relationship from the source data.

The formatting of the source data, and the eventual translation of the song into a machine-readable format, results in a loss of fluidity and human-like subtlety. This problem can be addressed in two different ways.

Firstly, the precise state merging conditions that the finite state automata utilize may be altered to allow more flexibility in merging. A second approach is to reapply the subtle variations required in note sequences. This can occur after the generation of the note sequence by the finite state automata.

Participants considered the form of the songs generated in the Bossa Nova style to be lacking. The two bar precision assigned to one phase restricted the Gen^3’s capability to specify instrument interaction within the two bars of the phase. This stipulation also necessitates the separate generation and combination of note sequences to form musical phrases exceeding two bars in length. A possible solution is the use of a variable length phase that accommodates the length of the musical phrase. This is only a partial solution. Instrument interaction within a phase must also be addressed.

Feedback from the survey indicates that implementation of chord progressions, generation of chord voicings, and transposition of note sequences were musically inaccurate. These failings are more evident in compositions in the Bossa Nova style than the relatively simple Trance genre. The correction of these processes is vital in achieving a technically competent piece of music.

On reflection, the modeling of the warm, human-oriented Bossa Nova genre was a difficult challenge. The machine based translation and auditioning detracted from the potentially positive feedback it may have received. The auditioning of Trance songs with superior, genre-specific instruments will have a positive effect on aesthetical ratings received. It is anticipated a song generated in the Trance style will achieve both the primary and secondary objective.

References

Clement, B. 1998. “Learning harmonic progression using Markov models”. [Online]. Available WWW: `http://ai.eecs.umich.edu/people/bradc/papers/545proj.ps`
Goldberg, D. 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley.
Järveläinen, H. 2000. *Algorithmic musical composition*, Telecommunications software and multimedia laboratory, Helsinki University of Technology.
Lindemann, E., F. Dechelle, B. Smith, and M. Starkier. 1991. “The Architecture of the IRCAM Musical Workstation.” *Computer Music Journal* 15(3):41-49.
Mathews, M. V. 1969. *The Technology of Computer Music*. Cambridge, Massachusetts: MIT Press.
Morangelli, M. 1999. “Jazz, a short history”. [Online]. Available WWW: http://www.thereelscore.com/PortfolioStuff/PDFFiles/HistoryJazz.pdf
Music Library, State University of New York at Buffalo. 2003. *Chronology of Lejaren Hiller's Life*, [Online], Available WWW: `http://ublib.buffalo.edu/libraries/units/music/spcoll/lhlife.html`
Papadopoulos, G and Wiggins, G. 1999. *AI methods for algorithmic composition : A survey, a critical view and future prospects.* School of Artificial Intelligence, Division of Informatics, University of Edinburgh.

Appendix: Survey Questions

For each song, participants were asked to rate the following statements on a Likert scale:

1. *This song is recognisable as belonging to the musical genre.*
2. *This song conforms to the genre in terms of:*
   i. *Chord voicing*
   ii. *Rhythmic properties*
   iii. *Musical phrasing*
3. *Instruments fulfil roles typical to the genre in terms of*
   i. *Interaction with other instruments.*
   ii. *Displaying stylistic characteristics common to the genre.*
4. *The form of this song displays a realistic style.*
5. *Instrument interactivity results in a smooth, continuous flow.*
6. *This song exhibits a similar quality to that of a human-composed MIDI file.*
7. *This song is aesthetically pleasing to the ear.*

Are there any further comments about the piece you would like to express?

Also, the following questions were asked regarding the overall quality of the songs:

1. *Which one of the following classifications do you feel best describes the quality of these songs? (a. Novice/Beginner b. Amateur c. Professional d. Commercial quality e. Other)*
2. *Based on the songs you have evaluated, do you think this software may be of help to you in composing songs? (a. It would not help me at all b. It would be occasionally helpful as a compositional aid c. It would frequently be helpful as a compositional aid d. Other)*
3. *If you answered b. or c. to question 2., please explain how this software would be of help to you.*

Are there any further comments you would like to add?