Musical Composition with Stochastic Context-Free Grammars

Salim Perchy, Gerardo Sarria

Musical Composition with Stochastic Context-Free Grammars

Salim Perchy and Gerardo M. Sarria M.

Pontificia Universidad Javeriana - Cali, Colombia

{ysperchy,gsarria}@cic.javerianacali.edu.co

Abstract. In this paper we discuss the design of a generator of musical sequences for the purpose of aiding musicians in their compositional work. A series of steps were abstracted to reflect composition as a sequential flow of data refination; also each step involved in this process was modeled to gain each one of their functions. Specifically, in the case of the main structure of the suggestion a stochastic context-free grammar was developed to generate harmonic progressions, in the case of rhythms a probabilistic model based on accents was also developed, and finally, to state individual notes of the suggestion another probabilistic model based in the dissonance of intervals was used. Each one of these models were structured upon J.S. Bach's rules of homophonic composition as this was the target musical genre of the work.

Key words: Musical Composition, Stochastic Context-Free Grammars, J.S. Bach, Computer Music

1 Introduction

Music, although an art in all its sense, frequently follows a set of rules conceived by human being himself to transform sounds into pleasant experiences to the senses. These rules are mainly subjective and their interpretation is different in several cases, especially when there is time involved (in the sense of *age*), but this matter does not make them impossible to abstract and for this reason it is feasible the integration of computer science in the art of music creation.

The general idea of creating a music-composing machine is ambitious and rightly if you take into account all the musical corpus the human race has produced, for this reason we propose in this paper a tool to aid the musician in his compositional work through musical suggestions. The tool has an emphasis in a concrete music genre.

The computational background used here is very oriented to the repertory used in the area of speech-recognition due to a strong parallel between music and literature that the musicians themselves have underlined as *music literature* over the years. Concepts such as free grammars, stochastic models and training are cornerstones to the creation and development of this work, all of them aimed to create a model of musical generation based on the homophonic music principles proposed by Johann Sebastian Bach, possibly the most influential figure in the matter.

The rest of the paper is organized as follows: section 2 explains musical concepts relevant to the work done here, section 3 describes the context free grammars and its stochastic extension, section 4 shows the composition assistant proposed and also the abstracted models for the composition process (progression generator, rhythm generator and notes assigner), in section 5 we show and detail results generated by the composition assistant, and finally section 6 draws some conclusions and discusses related work

2 Musical Composition

The basic principle that guides and aims the primary objective in a musical composition is the creation of musical sequences that share an idea. Although this goal is the composer's intention, the form of communicating such idea and its internal details (called *harmonization*) vary in a high degree from age to age. The musical theory and work in this paper is mainly guided by the principles of composition that J.S. Bach followed and helped evolve as a establish composer. There is no text book by Bach to reference all these principles because all his theory went into his works (*Art of Fugue*, *The Well-Tempered Clavier*) but their analysis attest to these principles.

To abstract the meaning of the labor of composition we can cite *Shöenberg* in [1]:

> *A musical piece is the connection of a set of* ***phrases*** *that carry a visible or invisible structure called* ***progression*** *of* ***chords****, that itself may be adorned (****melody****) to personalize such piece. The progression and melody follow some form or set of principles that themselves establish the piece's musical genre.*

For the sake of clarity we describe the next musical terminology:

- A *phrase* is a fragment of a song that is able to independently communicate a musical idea. Phrases may be expressed in one or more *tonalities*.
- A *tonality* is an intrinsic characteristic of a phrase that indicates what scale was used to create it.
- A *scale* is a set of musical notes (all different) that, used in conjunction, creates a sense of harmony (pleasant to the human ear). Each note of a scale represents a degree and each one of these has a name and a harmonic function. Figure 1 shows the scale of *C major*.
- A *bar* or *measure* divides all the notes in the *score* (written notation of music) by groups of the same duration (each one of these groups are divided by a vertical line). The duration of a bar is represented by a fractional number just before the beginning of notes, where the numerator states the number of times t that a bar has (e.g. 4 would be $4t$) and the denominator tells the durations of this time t; there is a restriction on the denominator, it may

Fig. 1. The C major scale and the degrees' names

only be a power of 2 and this number indicates the accents of each bar; e.g. 1 (whole note), 2 (half note), 4 (quarter note), 8 (eighth note).
- A *chord* may be defined as more than one note played at the same time. A *progression* of chords is a sequence of chords to form phrases and musical ideas (the base concept of harmony). The tonality of a progression may be changed by means of a process called *modulation* .

Fig. 2. Chord Progression: C (I) - F (IV) - G (V) - C (I)

The most common and elemental harmonic progression is probably I - IV - V - I (tonic, subdominant, dominant, tonic). The figure 2 shows an example of this progression. This flow of chords, apart from containing the three most important degrees from a scale, establishes well a tonality due to the interval of I - V (tonic, dominant). This last interval soundly marks to the ear the tonality in which the whole song or part from it is developed.

J.S. Bach was a pioneer in a compositional technique called *counterpoint* that was later succeeded by common harmony but it served as basis of the harmonization theory. To summarize counterpoint proposes the next harmonic principles:

1. The most important degrees of a tonal scale are:
   - 1st - Tonic
   - 4th - Subdominant
   - 5th - Dominant
2. Musical phrases should finalize in the tonic degree to indicate the ear a resting point.
3. With the important degrees of a scale a progression can be constructed: I - IV - V - I (Tonic - Subdominant - Dominant - Tonic). Most other progressions may be derived from this base progression to construct phrases.
4. The most important modulations are:
   - To the dominant or subdominant.
   - To the relative (major or minor).
5. Each degree has an harmonic function expressed in terms of the three most important degrees of the scale:

– 1st degree: Tonic.
– 4th and 2nd degree: Subdominant.
– 5th and 7th degree: Dominant.
– 3rd degree: Tonic or Dominant (This function is frequently weak or ambiguous).
– 6th degree: Tonic or Subdominant (This function is frequently weak or ambiguous).

3 Context-Free Grammars

A *grammar* is a set of rules to describe strings that belong to a language and are syntactically valid; these strings are made upon an alphabet. Formally, a grammar is a tuple $G = \langle N, \Sigma, P, S\rangle$ where N is a set of non-terminal symbols, Σ is a set of terminal symbols (the alphabet), P is a set of production rules, and S is the initial non-terminal symbol.

To exemplify this concept we can take a grammar that generates all the powers of 10. This grammar has the set of non-terminals $\{S, P, C\}$, the set of terminals $\{1, 0, \emptyset\}$, the initial symbol S and the next production rules:

$$\begin{aligned} S &\rightarrow P \\ P &\rightarrow 1C \\ C &\rightarrow 0C \\ C &\rightarrow \emptyset \end{aligned}$$

A grammar is *context-free* if for each one of the production rules, its right side is made up of only one non-terminal symbol, meaning if the production rules follow the next form:

$$A \rightarrow E$$

where $A \in N$, and E is any string in $(\Sigma \cup N)^+$.

A *stochastic* context-free grammar is an extension of context-free grammars where each production rule has an associated probability representing the plausibility of the rule . Then, a stochastic context-free grammar can be seen as production rule with the following form:

$$A_i \rightarrow E \qquad [p_i]$$

where $A \in N$, E is any string in $(\Sigma \cup N)^+$, and p_i is the probability of the rule A_i. The probabilities must fulfill the next statements:

1. $0 \leq p_i \leq 1$
2. $\sum_i p_i = 1$, where p_i is associated with A_i

A very trivial example of a stochastic context-free grammar is one that defines a two-coins tossing result. The set of non-terminals is $\{S, T\}$, the set of terminals is $\{\text{head}, \text{tails}\}$, the initial symbol is S and the production rules are the following:

$$
\begin{aligned}
S &\rightarrow T\ T && [1, 0] \\
T &\rightarrow \text{head} && [0, 5] \\
T &\rightarrow \text{tails} && [0, 5]
\end{aligned}
$$

4 The Composition Assistant

The process of musical composition was abstracted as a sequence of steps to facilitate its transition to the computational theory. The steps can be observed in the left part of figure 3. The initial parameters the user must input are:

1. Tonality (e.g. G major, B-flat minor).
2. Measure (e.g. $\frac{4}{4}$, $\frac{9}{8}$).
3. Velocity (e.g. 120 BPM).
4. Number of bars (length of the song)
5. Output format (PDF, PNG or MIDI).

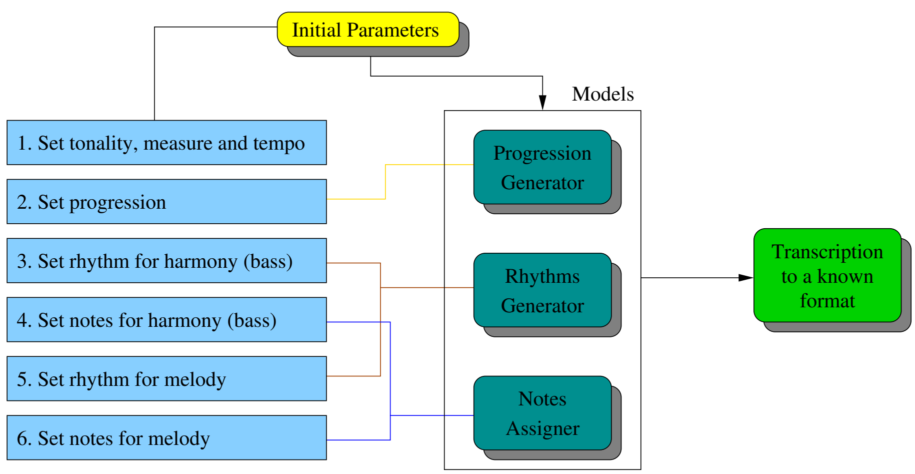

Fig. 3. Composition sequence and related models

In figure 3 the initial parameters are in charge of feeding the abstracted models with the appropriate initial data, the *progression generator* covers step No.2 in the composing sequence, the *rhythms generator* implements steps No.3 and No.5 of the sequence, and lastly the *notes assigner* deals with steps No.4 and No.6. Each model is an independent part and is solved as shown below.

4.1 Progression Generator

As said before, the cornerstone of harmony is the progression underneath the musical content, this is the main structure and foundation of the song. It can be represented by a string of sequential chords over time. Each chord is a degree used in the tonality of the musical phrase.

A very convenient way to model this is with a stochastic grammar with each rule meaning the possible flow the main structure of the song may take. Each symbol may represent a chord or an expandable progression that itself may transform again in a chord or another progression (depending on the length of the song the user suggested).

We modeled a stochastic context free grammar that generates the harmonic progression the song has. Also the grammar reflects the harmonic basis of the genre we wished to model (e.g. ending the song in the tonic degree) and giving appropriate probabilities to each rule so the result may sound consistent and coherent. These probabilities were found from a deep study of J.S. Bach pieces.

Next, we present part of the grammar for major tonalities (we confine ourselves to show just a segment of the grammar due to lack of space):

Degrees representation

Tonic	Supertonic	Mediant	Subdominant	Dominant	Submediant	Subtonic
i	ii	iii	iv	v	vi	viidis

Grammar for major scales

Start:

$S' \rightarrow A\ S\ F$	$[0,3]$	# Suggestion
$\rightarrow S\ F$	[0,7]	

Anacrusis:

$A \rightarrow ana\ v$	$[0,7]$	# Anacrusis
$\rightarrow ana\ i$	$[0,3]$	

Phrases:

$S \rightarrow T\ S_1\ D\ T\ S$	$[0,35]$	# Basic Phrases
$\rightarrow T\ D\ T\ S$	$[0,25]$	#
$\rightarrow T\ M\ D\ T\ S$	$[0,07]$	# Arpeggios
$\rightarrow T\ S_1\ S_2\ T\ S$	$[0,07]$	#
$\rightarrow S_1\ T\ D\ T\ S$	$[0,07]$	# No Tonic Start
$\rightarrow D\ T\ S_1\ T\ S$	$[0,07]$	#
$\rightarrow T\ S_1\ T\ S$	$[0,02]$	# Misc
$\rightarrow T\ S_1\ D\ S_2\ T\ S$	$[0,02]$	#
$\rightarrow T\ M\ S_1\ D\ T\ S$	$[0,02]$	#
$\rightarrow MOD$	$[0,06]$	# Modulation

Degree functions:

$T \rightarrow i$	$[0, 8]$	# Tonic
$\rightarrow iii$	$[0, 1]$	#
$\rightarrow vi$	$[0, 1]$	#
$S_1 \rightarrow iv$	$[0, 7]$	# Subdominant
$\rightarrow ii$	$[0, 2]$	#
$\rightarrow vi$	$[0, 1]$	#
$D \rightarrow v$	$[0, 4]$	# Dominant
$\rightarrow v7$	$[0, 3]$	#
$\rightarrow viidis$	$[0, 2]$	#
$\rightarrow iii$	$[0, 1]$	#
$M \rightarrow iii$	$[1, 0]$	# Mediant
$S_2 \rightarrow vi$	$[1, 0]$	# Submediant

End:

$F \rightarrow T\ D\ T$	$[0, 7]$	# End Suggestion
$\rightarrow T\ S_1\ T$	$[0, 3]$	

The grammar is fed to the program as a plain text file, so the rules and their probabilities may be changed by the user to reflect other rules or his own experimental ones. There was also the need to separate a grammar for major and minor tonalities due to their degrees differing in subtle aspects.

4.2 Rhythms Generator

Rhythm is the key aspect that brings music a sense of order because it (music) is a physic phenomena (sound) observable through time (duration). Scientifically speaking, rhythm is the disposition of the elements (notes) inside a basic unit of order (measure).

Therefore a measure or bar might present a rhythmic idea that can be executed to giver a sense of order to the human ear in the arriving of notes, making the whole phrase comprehensible. A musical piece frequently has a complex idea turning the notion of rhythm in a collective union and making way to a wider and superior idea (thus having order itself).

Having just a complex rhythm for a musical piece may transmit an idea but lacks some properties (color, feeling, climax), on the contrary, just having a progression with no dynamic rhythm will mark a harmonic structure also lacking important properties (climax, velocity, marking). Just the mixing of these two concepts is able to deliver fully the idea of music.

Because the duration of a note depends on the overall velocity (BMP) and the song's measure, considering all the possibilities a note may posses is cumbersome. To achieve an independence on the concrete time a note may have we needed to do a normalizing process to manage a duration relatively to the initial parameter the user input. Therefore, the duration of a note is expressed as a factor of the basic time unit the measure has. For example, if the measure is $\frac{4}{4}$, 1.0 is equal to a quarter note, 2.0 to a half note, 0.5 is equivalent to an eighth note and so on. The reader may refer to [2] to a formal study of rhythm metrics and structures.

The proposed model for generating rhythms is then a probabilistic model of events, where each event is considered the duration of a note. Each one of these durations has a probability of happening in a single measure, these probabilities are tuned accordingly to which part of the suggestion is being generated (harmony or melody) and also must fulfill the probability rules. Because of the normalization, the managed probabilities are the ones given to each musically possible factor of the basic duration (e.g. probability for 2.0, 1.0, 0.5 and so on).

The model also differentiates the importance of each measure based on its portion of the main harmonic progression and takes different probabilities for every event according to each case, clearly each case has probabilities adjusted to each event based on its function. Theoretically, we can discriminate measures based on their functionality as follows:

1. Normal measures: Bars from within the progression with no radical meaning other than advancing the piece.
2. Special measures: Pivotal bars or with special meaning. Frequently are bars which mark the tonic or dominant degree and/or serve as bridges to change tonality
3. Final measures: Final bars of phrases or the whole theme. They clearly mark the resting point to the ear.

A possible distribution of the probabilities of durations in a melody is as follows:

Duration	Probability		
	Normal Measure	Special Measure	Final Measure
0,5	0.10	0.05	0.05
1,0	0.30	0.20	0.10
1,5	0.15	0.20	0.05
2,0	0.20	0.20	0.30

All of these probabilities are again fed to the program by plain text files so the user may change the values or add more events.

In order to avoid rhythmic chaos (because the model may generate a different rhythm in each bar), there exists a decision tree for the generating of a new rhythm used in the following bars. Each leave of the tree models the course of the actual rhythm and again is accompanied by a probability. A decision tree to generate global cohesion of the musical piece is illustrated in figure 4.

Mainly, the rhythm has two options: to be the same or to change. If the decision of changing is taken then the model can simplify it (use more basic durations or longer) or complicate it (use less basic durations or shorter), also it can reuse a rhythm used in the past or generate a totally new one.

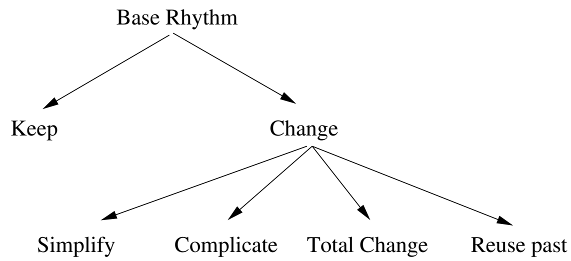

Fig. 4. Decision Tree for Rhythms

4.3 Notes Assigner

The assignation of concrete notes in the melody line is the most important step in the sense of personalizing a composition. It is here where the musical piece takes a defined form and reflects the idea or motive that the composer wanted to imprint. Theoretically, as said before, the melody line is an adorn to the main harmonic structure; the primary goals of the melody are to bring tensions, climax and dynamics to the song.

To achieve these aforementioned properties the composer must bear in mind the intervals used in the melody line. To assign a note we set a probability for each diatonic interval possible within the first octave, the process consists of setting the first (accented) note of the bar as the root of the first chord used in the progression and then setting the other notes of the rest of the bar.

One possible set of probabilities of the upward and downward intervals may be as:

Interval	1st	2nd	3rd	4th	5th	6th	7th	8th
Probability	0.05	0.30	0.20	0.10	0.20	0.05	0.05	0.05

The biggest probabilities may be given to the 2nd, 3rd and 5th interval; this is because the 3rd and 5th intervals are the other notes (together with the 1st) that make a basic chord with no dissonance involved, the 2nd interval is frequently used in tonal music to provoke a sweep of the scale that belongs to the tonality in use.

The intervals 6th, 7th and 8th are distant intervals and may be aggressive to the ear excluding special cases. The 1st interval means the same note (no interval) and is given a low probability to keep from repeating notes.

These probabilities are set to both upward and downward intervals and also the exact space between notes are relative to the scale being used. The data is again in a plain text file for manipulation.

5 Results

The algorithm for generating strings from a grammar was created due to the important parsers like Earley [3], Eugene Charniak's parser [4] or CKY (Cocke-Kasami-Younger [5]) being only to determine if a string belongs to a language, not to generate a string from *that* language, also these algorithms do not take into account stochastic grammars.

We implement the algorithms in Python using Gtk+ as a graphical interface. For the transcription to a known format we use Lilypond (`http://lilypond.org/`).

Next, we show a suggestion given by the model with the following initial parameters:

- Tonality: G major
- BPM: 110
- Measure: $\frac{4}{4}$ (binary)
- Length: Aprox. 25 bars

Sugerencia

Generado por Musi-K

The song's initial tonality is G major modulating to its relative minor (G minor) at measure 6 and keeping this tonality until finishing. Harmonically the song is marking the progression I-VI-I and the progression I-III-I with an intersection containing the common progression I-II-V7-I and finally finishing in the tonic degree (G minor).

Melodically the song is overall simple having upward intervals to make a tension or climax at various points (measures 3, 8, 10, 13, 16, 21) and then with downward intervals to relieve the ear of this. It is to note that the climax at bar 16 coincides with the chord V7 (this chord has a bit of dissonance).

Rhythmically there were two basic rhythms generated for the bass and the melody line, the reader may see that the trivial rhythm for the base (just one whole note) has a more complex counterpart in the melody each time it is played. The other two rhythms are fairly normal that make way to the following more complex rhythm. Also the global rhythm is far from chaotic.

There are more results, the reader may take a look at some of them at `http://cic.javerianacali.edu.co/~gsarria/suggestions`. On the same webpage the reader may download a beta version of the assistant.

6 Concluding Remarks and Related Work

The automated generation of music has been a research area since mid 1900's. Several advances from this time are still used today. The main focus was to generate consistent musical sequences; for this purpose statistical methods were established that made use of models and algorithms of artificial intelligence. The reader may see [6] to get a complete survey of music generation from statistical models. On the other hand some structural view of music and its analogy to natural language grammar have been well discussed in [7].

We have shown a model based on J.S. Bach's harmonic rules that can generate a consistent musical sequence with the purpose of aiding the musician in his labor of composing. Although the generated results are no match to the great works of Bach, they fulfill their purpose as suggestion when the composer is looking for inspiration or ideas to integrate into his works.

The overall consistency of the suggestions are well maintained, avoiding common pitfalls in the automatic music generation area, pitfalls like rhythm chaos, repetitive themes and lack of motif are shown not to be present.

The disadvantages in this approach are well known. When using the random method, although fast, there is no possible way to assure consistency inside the sequence and may produce chaos, when using markov-chains the complexity rises exponentially and this is a big problem considering long musical pieces.

There have been also other methods proposed for assisted composing, for example we can see the use of fractals [8] and process calculi [9]. Another important novelty is the use of *factor oracle* [10] that is simply a finite automaton trained by the composer's music and then swept to generate new sequences within the space the composer's music had, the big difference is that its basic units(each node in the automaton) are notes and in the grammars they may be notes or motives or big structures. Because of this the *factor oracle* adjusts itself with ease in atonal systems like the twelve-tone technique [1] proposed by Schöenberg, and jazz music.

The generation of music through computational models is not new, in fact there are a variety that have been used conform the interest in the subject has grown. Actually the most used approach is the probabilistic one (see [6] [11] [12] for instance) and the models used are for example random, Markov-based, stochastic generation, and pattern-based generation. The novelty of our approach in contrast to the mentioned above is the abstraction of compositional work in the three modules (progression generator, rhythms generator, and notes assigner) and their structure upon J.S. Bach's rules of homophonic composition.

We plan to pursue this work in two directions: (a) by including other progressions (I-IV-I for instance) and taking a look at other important composers like Jean-Philippe Rameau (Bach's contemporary composer), and (b) by using more specific techniques of parsing, for instance we wish to use the new context-free approach in [13] for producing new progressions from the grammars.

References

1. Schöenberg, A.: Fundamentals of Musical Composition. Faber and Faber (1967)
2. Nicolas, F.: Le feuillet du tempo. Entre Temps, 9:51-77. (1990)
3. Earley, J.: An efficient context-free parsing algorithm. Comm. ACM. 13(2):94–102. (1970)
4. Charniak, E.: Statistical Language Learning. Cambridge: MIT Press. (1993)
5. Kasami, T.: An efficient recognition and syntax-analysis algorithm for context-free languages. Scientific report AFCRL-65-758, Air Force Cambridge Research Lab, Bedford, MA. (1965)
6. Conklin, D.: Music Generation from Statistical Models. In Proceedings of the AISB 2003 Symposium on Artificial Intelligence and Creativity in the Arts and Sciences. Aberystwyth, Wales (2003)
7. Lerdahl, F., Jackendoff, R.: Generated Theory of Tonal Music. The MIT Press. (1982)
8. Wright, P.: Generating Fractal Music. PhD Thesis, University of Western Australia. (1995)
9. Perchy, S. and Sarria, G.: Dissonances: Brief Description and its Computational Representation in the RTCC Calculus. In SMC2009, Porto, Portugal. (2009)
10. Assayag, G and Dubnov, S.: Using Factor Oracles for Machine Improvisation. Soft Comput. 8(9):604–610. (2004)
11. Ortega de la Puente, A., Sánchez Alfonso, R., Alfonseca, M.: Automatic composition of music by means of grammatical evolution. ACM SIGAPL APL Quote Quad 32(4):148–155 (2002)
12. McCormack, J.: Grammar-Based Music Composition. Complexity International Vol. 3 (1996)
13. Alonso, M., Cabrero, D., Vilares, M.: Construction of Efficient Generalized {LR} Parsers. Springer-Verlag. (2009)