Unsupervised Generation of Chord Sequences from a Sound Example

Author: Katerina Kosta

MASTER THESIS UPF / 2011

Master in Sound and Music Computing

Master Thesis Supervisor: Dr. Hendrik Purwins
Department of Information and Communication Technologies
Universitat Pompeu Fabra, Barcelona

UNIVERSITAT POMPEU FABRA

Abstract

In this thesis project a system is developed for the analysis of a chord sequence given as an audio input with the aim of generating arbitrarily long musically meaningful and interesting sound sequence using the input characteristics. The procedure that is followed includes the transcription of a harmonic sequence into a shuffled multilevel representation, utilizing a tempo estimation procedure and identifying the most regular subsequence in order to guarantee that the specific harmonic structure is preserved in the generated sequence. In the final synthesis, the system recombines the audio material derived from the sample itself and it is able to learn various concepts of harmony and individual styles from the data. Also it can show how practice deviates from the *ideal* form. First of all, the sound is split into chord segments, then a clustering model is applied for grouping the chords, taking into account their harmonic structure through their Constant Q Profiles. Following this procedure, a Variable Lenght Marcov Chain (VLMC) model is used in order to predict and re-shuffle these elements, maximizing simultaneously the cluster resolution, in order to avoid musical discontinuities. The system is evaluated objectively as well as subjectively by musicians and non-musicians, showing that the automatically generated chord sequences maintain the key features of the original and that they can be musically interesting.

Keywords: *chord sequences, Music Information Retrieval, clustering, statistical models, automatic generation*

Acknowledgements

First of all I would like to thank Marco Marchini for the inspiration he gave me through his work on his last year's thesis project, and of course Panos Papiotis who introduced me at this very interesting field. In general I' m indebted to my supervisor and all MTG community which constituted a very fruitful environment. Special thanks to Srikanth Cherla for his support during coding and exploring this field.

Also I should not forget to thank Bruno Rocha, Lina Maria Villa Betancourt, Jan Van Balen and Emilio Molina for their help at the recording sessions and the great time we had at ESMUC studio. And again special thanks to Lina for her help on developing Matlab codes needed for the course "audio and music analysis": they gave me a "strength forward" for developing this project.

It is hard to be productive without good company, so I want to say that I was lucky to be surrounded by all these people I met this year in Barcelona. Also I want to thank my good friend Alexandra Tselepi for her help, especially at the begining of this year and Stayroula Tsolakou for her mentions during the harmonic analysis. At the end I would like to mention Pelopidas for giving me few but useful days of relaxation during summer.

Katerina Kosta
1/9/2011

Musical Terms that are used

⋇ **Legato:** Smooth and connected playing style in which the notes seem bound together.

⋇ **Staccato:** Detached, jerky playing style, the opposite of *legato*.

⋇ **Augmented Triad:** A modified major triad obtained by raising the fifth note one semitone (half step). It is two major thirds, one on top of the other. An augmented triad is denoted by appending a "+" to the major triad name.

⋇ **Diminished Triad:** A triad consisting of an outer interval of a diminished 5th and a bottom interval of a minor 3rd, for example C-Eb-Gb.

⋇ **Cadence:** The melodic pattern just before the end of a sentence or phrase, for instance an interrogation or an exhortation. More generally, the natural rhythm of language depending on the position of stressed and unstressed syllables. Cadence is a major component of individual writers' styles.

⋇ **Theme:** The musical subject of a piece (usually a melody), as in sonata form or a fugue. An extramusical concept behind a piece.

⋇ **Variation:** Restatement of the theme in different guises.

([26], [27])

Contents

Abstract **i**

1 Introduction **1**

2 Goals **3**

3 Problem domain **5**

3.1 Chord definition and grouping psychological background 5

3.2 Style of composition - Bach chorals 6

4 Methodology **7**

4.1 Input Information 7

4.2 Onset Detection 11

4.3 Sound Clustering 12

4.4 Statistical Model for generating sequences 14

5 Evaluation and Validation **18**

5.1 Evaluation of the Clustering part 18

5.2 Evaluation Questionnaire 21

6 Conclusions - Future Work **28**

6.1 Conclusions 28

6.2 Future work 28

7 Appendix A **30**

8 Appendix B **32**

9 Bibliographic references **37**

September 14, 2011

Chapter 1

Introduction

”The computer can’t tell you the emotional story. It can give you the exact mathematical design, but what’s missing is the eyebrows.”
Frank Zappa

Since the antiquity, music was approached by nations in terms of causality and determism. Firstly, through the school of Pythagoras, relationships between numbers and the natural environment have been discovered so that a whole system of overtone series, which is the base for the western music as an entity, could be constructed. Then the school of Plato insisted on the principle of causality, ”for it is impossible for anything, to come into being without cause” (Timaeus) [1]. Since the nineteenth century, music was surounded by these two concepts, but then a brutal and fertile transformation as a result of statistical theories in physics was emerged and played a crucial role in music construction and composition. From this point of view, from which we wish to examine and make music, ”primary time appears as a wax or clay on which operations and relations can be inscribed and engraved, first for the purposes of work, and then for communication with a third person” [1].

By this fundamental stand, we presume the way music can be formalized, described and analyzed is based in pure logic. This fact has helped harmonic rules to be used by computers for producing and composing music. During the last two decades much effort has been devoted to build computational architectures of musical sequence learning [2]. Searching for the basis of this field, we can begin our time travel many year earlier, when Mozart wrote the measures and instructions for a musical composition dice game in 1787, the famous ”Musikalisches Würfelspiel”: the idea was to cut and paste pre-written measures of music together to create a Minuet. It was the first time when two 6-numbered dice were used to create music in an unsupervised way, consisting 272 musical measures and a table of rules used to select specific measures given a certain dice roll. The result of that procedure is a randomly selected 16 bar minuet and 16 bar trio [22].

Another good, more elegant example is ”The Continuator”; a study which has been conducted for the analysis and the generation of music sequences by a MIDI-based system for real-time musical interaction, yielding good jazz style music generation [3]. That can be generalized by constructing a basic formula for chord sequences that can be used for any musical styles.

Many preceding operations have been occurred, following their specific paths for this area's exploration and implementation. One remarkable research work is the "Harmonisation system: composition of new harmonisations using data set of chorales" [7]; It is a creation of a harmonisation system which learns from examples and which can compose new harmonisations. The data set that is used is chorals from J. S. Bach. Also another one research work which was an affective inspiration for this current project is the "Rhythm Continuator" [11], which deals with unsupervised generation of percussion sequences from a sound example. It is a system development for the analysis of the structure and the style of a percussive audio sequence with the aim of generating an arbitrarily long musically meaningful and interesting sound sequence with the same stylistic characteristics as the original. Our thesis profile can be defined as the extension of the latter significant work.

Also some interesting applications based on the same philosophy are the following: The "Audio Oracle" which is an algorithm for fast learning of audio structure - indexing of audio data in terms of repeating sub-clips of variable length "audio factors". It is a new method representation for indexing of audio data. The structure that is followed allows fast retrieval and recombination of sub-clips in a manner that assures continuity between splice points. The resulting structure accomplishes effectively a new method for texture synthesis, where the amount of innovation is controlled by one of the synthesis parameters [16]. In addition, some commercial products, such as the "Band-in-a-Box" should not be missed. It is an intelligent automatic accompaniment program "from which you can hear and play along to many song ideas, and go from "nothing" to "something" in a very short time. Just typing the chords and choosing the composition style, it automatically generates a complete professional-quality arrangement of piano, bass, drums, guitar, and strings or horns" [17].

The combination of scientific, technological and artistic methodologies through computational approaches is the basic line for the idea of inverting a system that creates music in an unsupervised way, thus the "machine learning" field, a branch of artificial intelligence, is a scientific discipline which is worth of being discovered. The influence in memory, context and prediction of time ordered sequential events during the audition of tonal - western music are the key-words for this current research.

Chapter 2

Goals

"There's nothing remarkable about it. All one has to do is hit the right keys at the right time and the instrument plays itself."
Johann Sebastian Bach

The main goal of this thesis project is the development of a system for the analysis of a chord sequence given as audio input, with the aim of generating arbitrarily long musically meaningful and interesting sound sequence using the input characteristics. The automatically generated chord sequence has to maintain the key features of the original and it has to be musically interesting. This can be done by elaborating the following fields:

1. Deciding the kind of input:

Our database includes audio guitar and piano chord sequences, Bach's chorals played on the piano and Chopin pieces.

2. Taking the input information:

Some of the useful information we want to extract from the audio files is basically the key estimation and the tempo. For succeeding this, the MIR toolbox (University of Jyvaskyla), the Constant-Q Transform Toolbox for Music Processing (Queen Mary University), a New Method for Tracking Modulations as it is presented at [14], and some implementations through the MATLAB codes from the course "Audio and Music Analysis" (lab3) assist in yielding the information needed.

3. Segmenting and labeling the existing chords:

The basic idea for applying this step is to distinguish which notes are played simultaneously, by taking into account also the suspended notes played separately; in that point knowledge of harmonic theory is needed.

4. Grouping the existing chords using clustering models:

An unsupervised procedure is needed for grouping the existed samples. That is why hierarchical clustering is used, having the special features of the agglomerative (bottom-up, clumping) type of procedure, the creation of n singleton clusters and the formation of their specific merge.

5. Using variable length Markov chains for making more complex sequence structures:

Using Markov models we assume that future predictions are independent of all but the most recent observations. VLMCs are Markov chains with the additional attractive structure that their memories depend on a variable number of lagged values, depending on what the actual past looks like. There are also tree-structured models for categorical time series.

In order to construct a musically interesting chord sequence, the stable degrees I, IV, V (more details in the next chapter at "Chord definition and grouping psychological background" section) should be more repeated in contrast of the secondary ones, but both following the harmonic theory rules.

6. Evaluating the procedure by creating interesting sound generations and asking mucicians and non-musicians to rate how much the automatically generated chord sequences maintain the key features of the original and how interesting they are.

Chapter 3

Problem domain

"You just pick up a chord, go twang, and you 're got music."
Sid Vicious

3.1 Chord definition and grouping psychological background

As defined by Justus and Bharucha at [4], "a chord is the simultaneous sounding of three or more notes, and the Western system is built particularly on the triads within the major and minor keys. A triad is a chord consisting of three members of a scale, with each pair spaced by the interval of a major or minor third. Thus there are four types of triad: major, minor, diminished, and augmented, depending upon the particular combination of major and minor thirds used. In a major or minor key, the kind of triad built upon each scale degree will depend upon the particular series of semitones and whole tones that make up the scale. For example, in the key of C Major the seven triads are C Major (I), d minor (ii), e minor (iii), F Major (IV), G Major (V), a minor (vi), and b diminished (vii). The tonic (I), dominant (V), and subdominant (IV) are considered the most stable chords in the key by music theorists, followed by ii, vi, iii, and vii".

This hierarchy of harmonic stability has been supported by psychological studies as well. One approach involves collecting ratings of how one chord follows from another. For example, Krumhansl, Bharucha, and Kessler (1982) used such judgments to perform multidimensional scaling and hierarchical clustering techniques. The psychological distances between chords reflected both key membership and stability within the key; "chords belonging to different keys grouped together with the most stable chords in each key (I, V, and IV) forming an even smaller cluster. Such rating methods also suggest that the harmonic stability of each chord in a pair affects its perceived relationship to the other, and this depends upon the stability of the second chord in particular (Bharucha Krumhansl, 1983). Additionally, this chord space is plastic and changes when a particular tonal context is introduced; the distance between the members of a particular key decreases in the context of that key (Bharucha Krumhansl, 1983; Krumhansl, Bharucha, & Castellano, 1982)" [4].

The chords grouping method above is based on each scale they belong; Dahlhaus(1967) understands function theory (Riemann 1913) to be a further abstraction of the seven scale degrees to the three tonal functions [5].

3.2 Style of composition - Bach chorals

Like every art form, the choral is clearly articulated. As Schoenberg mentions at [12], "articulation (Gliederung) is necessary for every idea, the moment it is expressed; for, although we think an idea at once, as a whole, we cannot say it all at once, only little by little: we arrange the different components in succession components into which we divide up the idea differently from the way we put it together, and thereby reproduce more or less precisely its content". In that way, harmonic progressions are used as the components of an idea. The choral is articulated by a pause at the end of each musical phrase and the pauses divide the thought up into parts. The individual parts in such simple art forms relate to one another by the simplest forms of contrast or complement. What binds them together is the uniformity of the rhythmic movement, the straightforwardness, the simplicity, and above all the key [12].

Following the same philosophy, the re-generated chord sequence from the random sequence input that is used for the code implementation using audio input, has the basic characteristics depending on the articulation of the original one.

Chapter 4

Methodology

"Mathematics build systems of symbols to satisfy needs that are partly practical, partly aesthetic."
Henri Poincaré

4.1 Input Information

In order to take the information needed from an audio example, especially the frequencies that are used and can be shown through spectral coefficients, we have to take into acount the Constant Q profiles which offer a specific description of the spectral shape of the sound.

Cq-profiles are 12 - dimensional vectors, each component referring to a pitch class and they are calculated with the constant Q filter bank [18]. Basically they are a new concept of key profiles, which advantages are the following:
(1) Each cq-profile has a simple interpretation, since it is a 12 - dimensional vector like a probe tone rating. The value of each component corresponds to a pitch class.
(2) A cq-profile can easily be calculated from an audio recording. Since no complex auditory model, or other time consuming method is used, the calculation is quick and can be done in real time.
(3) The calculation of the cq-profiles is very stable with respect to sound quality. [14]

The calculation of the cq-profiles is based on the constant Q transform; As it is well decribed by Schorkhuber and Klapuri at [13], "it refers to a time-frequency representation where the frequency bins are geometrically spaced and the Q-factors which are ratios of the center frequencies to bandwidths, of all bins are equal. An inverse transform that is proposed is used so that we can enable a reasonable-quality (around 55dB signal-to-noise ratio) reconstruction of the original signal from its CQT coefficients". We could use this resynthesis audio for making some meaningful changes, for example transforming the major chords to minor ones and the inverse. CQTs with high Q-factors, equivalent to 36 bins per octave, are of particular interest and that is why this value is applied to our implementation, minimizing in that way the spectral leakage. Thus the method is flexible with regard to the applied window function, and the Q-factors. In our case, a BlackmanHarris window is used and hop size 50 %.

Like the Fourier transform, a constant Q transform is a bank of filters, but in contrast to the former it has geometrically spaced center frequencies, which are calculated by the following equation:

$$\mathrm{f}_k = f_0 \cdot 2^{\frac{k}{b}} \tag{4.1}$$

where k := the specific CQ bin and b := number of filters per octave.
Thus the $k - th$ filter having a spectral width some multiple of the previous filter's width is equal to:

$$\delta f_k = 2^{\frac{1}{n}} \cdot \delta f_{k-1} \tag{4.2}$$

where $n \in \mathbb{N}$ are the filters per octave.

From the above, the following recursive function is deduced:

$$\delta f_k = (2^{\frac{1}{n}})^k \cdot \delta f_{min} \tag{4.3}$$

Q is the integer number of cycles processed at a center frequency f_k and is defined as:

$$\mathrm{Q} := f_k/\Delta f_k = (2^{\frac{1}{b}} - 1)^{-1}, \forall k \in \mathbb{N} \tag{4.4}$$

where b dictates the number of filters per octave. This is achieved by choosing an appropriate window length N_k individually for each component of the constant Q transform (cq-bin) [14].

As it is mentioned at [14], "Krumhansl observed a remarkable correlation between the probe tone ratings and the total occurrences of the twelve chromatic scale tones in musical compositions. In order to establish a direct correspondence between probe tone ratings and profiles of a cq-reference set, one fact has to be taken into consideration: in the cq-profiles not only the played tones are registrated, but all harmonics. For piano tones the strongest frequency contribution falls (modulo octaves) on the tonic and on the dominant keynote in an approximate average ratio 3:1. That is why the harmonic spectrum of the analyzed tones is accounted for".

As we can see at the following figure, the constant Q transform is useful in establishing a direct correspondence between filters and musical notes by identifying appropriate center frequencies.

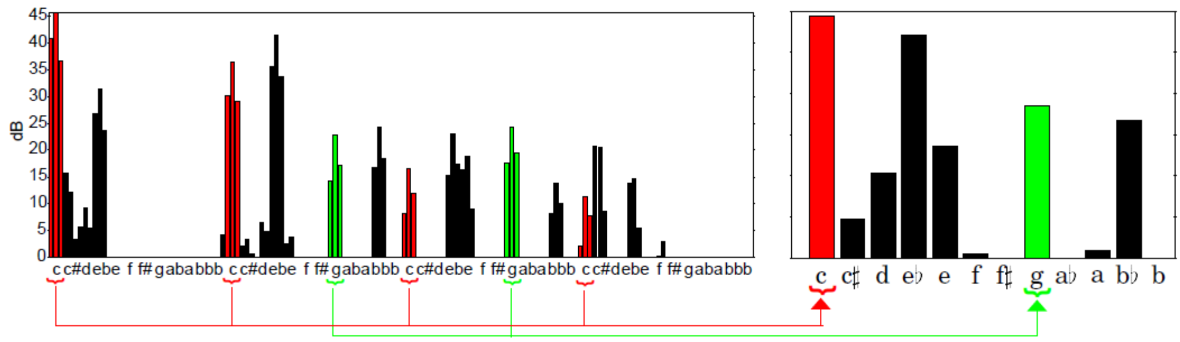


Figure 4.1: The constant Q transform is calculated from a minor third c e (played on piano) with three bins per half-tone (left figure). We yield the constant Q profile (right figure) by summing up bins for each tone over all octaves [14].

A cq-reference set is a sequence of 24 cq-profiles, one for each key. Every profile should reflect the tonal hierarchy that is characteristic for its key.

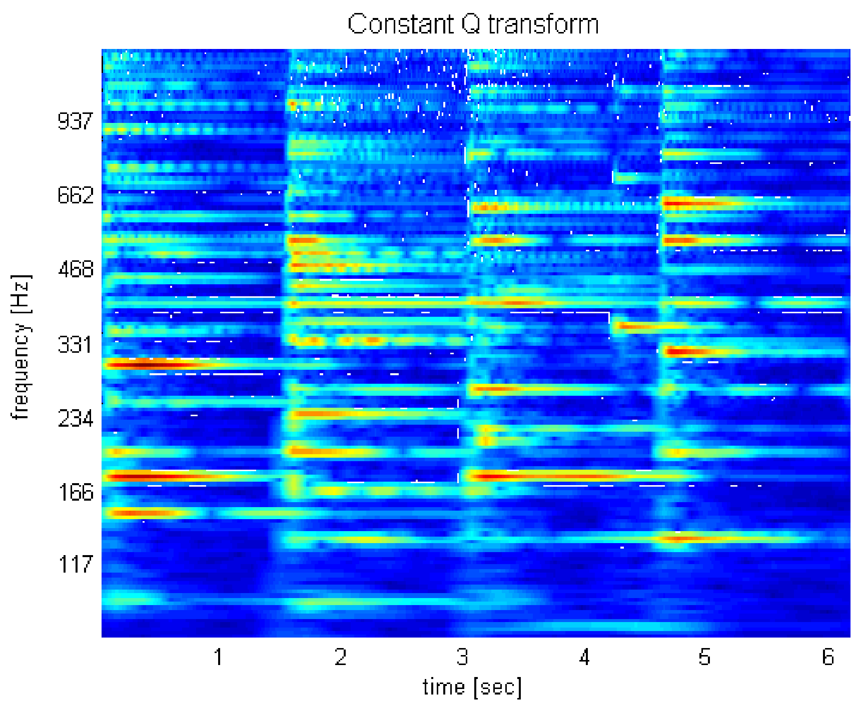


Figure 4.2: The first four chords of Chopin sonata - spectrum view through Constant Q Transform MATLAB code.

In order to achive a way of chord categorization, *sumCompress* function from CQ toolbox by Blankertz and Purwins has been used. This implementation is based on the idea of capturing the information of the audio signal and transferring this into one octave. For every input, even for the Bach choral examples that are applied at the system for evaluating the clustering part, we use the default parameters of *SumCompress*. The idea is that possible first and second inversions of one chord should be *categorized* together. Also the value of the parameter *attacklen*, which determines the duration in seconds of the event that take into account for taking the input information, is 0.6.

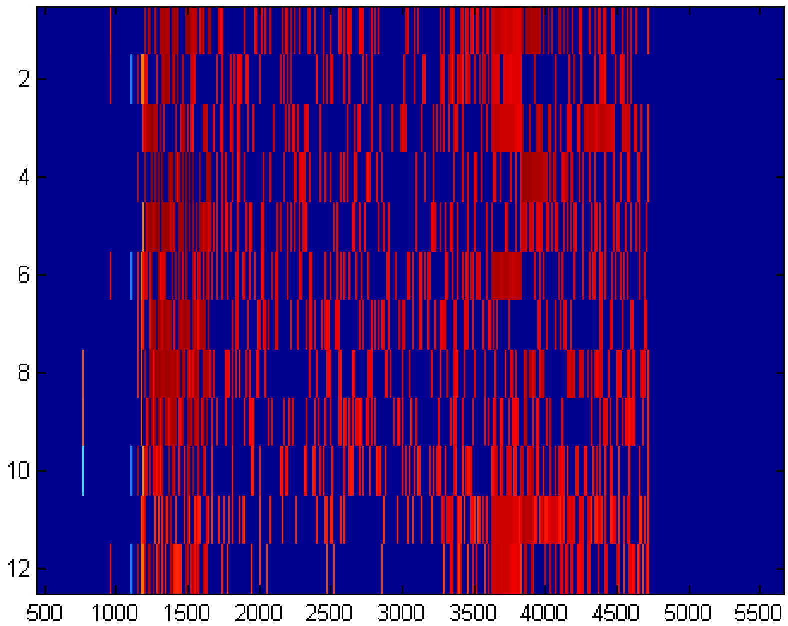


Figure 4.3: Frequencies that are heard from a guitar example compressed in one octave through time.

One important issue is a way of labeling each segment of the input sound, so that even a non musician could have a more spherical perspective, during the analysis of the evaluation of the system. That is why *find_key* function of Constant Q toolbox was applied. This kind of algorithm is based on the pitch class profiles created for each one and most corresponds to their clearness. It is somehow a "portrayal" of them. At this point an experiment was considered interesting; what should have happened if instead of the use of the Constant Q profiles, we also conceive other profiles having the same purpose but different configuration.

Following that philosophy, part of the chord estimation code, using Shepard Resynthesis Tones (more details in [28] and [29]), for analysing the audio input, has been implemented on piano chord excerpts having short duration (average 1 second). Comparing these two approaches, using 89 audio excerpts, the results were intriguing enough: Using the Constant Q profiles we had just 2 of 89 wrong responses (2.25%), instead of using the chord estimation of Shepard tones, where the wrong responses were 7 (7.86%). But the interesting part of this enquiry is the kind of these appearing errors: 66.6% of them had to deal with major keys and their relative minors, as the first ones are shown as result, instead of the second ones. This can be explained by their relatedness (fig. 4.6). Also 11% of the wrong responses have shown the dominant instead of the root (V and not I) and an explanation for that should be the way the harmonics are produced by the sounded notes.

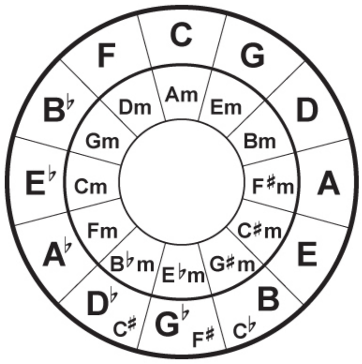


Figure 4.4: Major and minor keys relatedness through the "circle of fifths".

All the above just show the difficulty of the task we have to face: the possible error when the system deals with the segments' structure and how much it affects their grouping.

4.2 Onset Detection

In order to select the chords so that they can successively constitute the audio segments that we will take into account for the clustering and generation part, many methods have been tested. For the guitar audio inputs the function *mirsonsets* from MIRtoolbox has been implemented, using the method "attack"; this option estimates the beginning of the attack phase of each note by searching for the local minimum before each peak of the signal.

Having piano as an input the results using the same MATLAB function were not as they have been foreseen. The precision was so low, that we had to regard different approaches; the Aubio Tool [19] has been applied for that purpose and specifically the "complexdomain" method.

We are able to realize how important is a meaningful onset detection for the system, by implementing it with audio Bach chorals, played by piano. One problem that we had to face was how *sensitive* should the detection be, as we don't have every time four notes played simultaneously and clearly separated from the other elements. To be more specific, we have to take into account the parts of the piece, where single notes are played, but in the same time, other notes where played before, are still sounded. These kinds of single notes have three entities: They can just be consequence notes of a melodic pattern of a specific voice, or they are part of the chord that is heard before (so, that chord is usually a diminished or augmented triad), or they identify another chord (when the bass voice is changed, most of the time).

In our case, the sensitivity that is applied can be seen at fig. 4.8, where the first five segments of a Bach choral are presented. There, the fifth excerpt includes two different kinds of important chords that we have to take into account separately, an identification that couldn't be provided automatically.

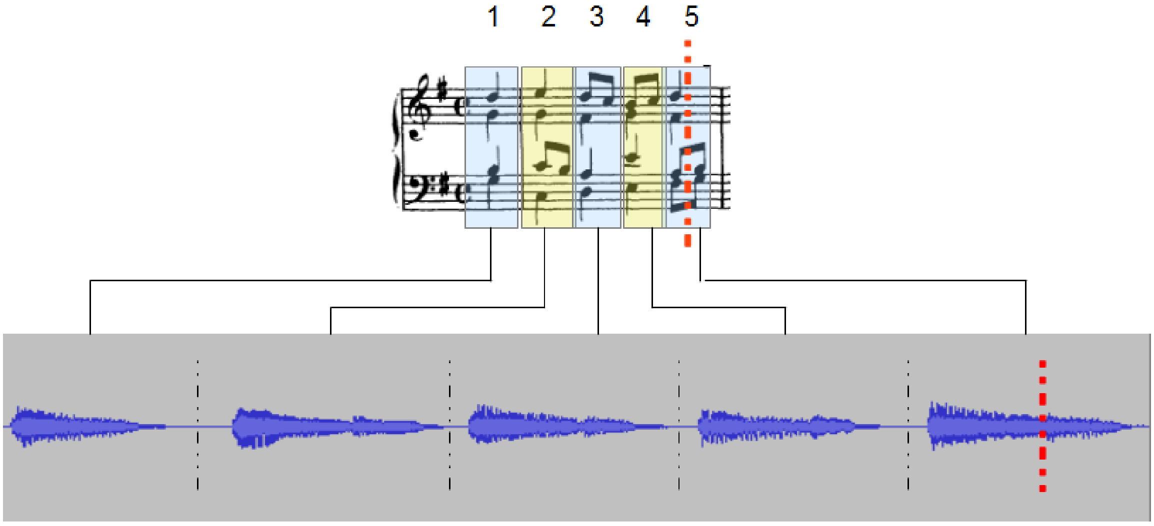


Figure 4.5: The first 5 segments of Bach "choral4.wav", using aubio for onset detection.

This unexpected grouping and chord identification error value, through the onset detection procedure, was 8.3% , based on analysis of four chorals.

4.3 Sound Clustering

At this processing stage, each event is characterized by a 12-dimensional vector. Events can thus be seen as points in a 12-dimensional space in which a topology is induced by the Euclidean distance. The single linkage algorithm has been used to discover event clusters in this space. This algorithm recursively performs clustering in a bottom-up manner. Points are grouped into clusters. Then clusters are merged with additional points and clusters are merged with clusters into super clusters. The distance between two clusters is defined as the shortest distance between two points, each in a different cluster, yielding a binary tree representation of the point similarities. The leaf nodes correspond to single events. Each node of the tree occurs at a certain height, representing the distance between the two child nodes [11].

The major steps in agglomerative (bottom up) clustering are contained in the following procedure, where c is the desired number of final clusters:

begin initialize c, $c' \leftarrow n, D_i \leftarrow \{x_i\}, i = 1, ..., n$

  do $c' \leftarrow c' - 1$

    find nearest clusters, say D_i and D_j

    merge D_i and D_j

  until $c = c'$

 return c clusters

end

This procedure terminates when the specified number of clusters has been obtained and returns as set of points, rather than as mean or representative vectors [9].

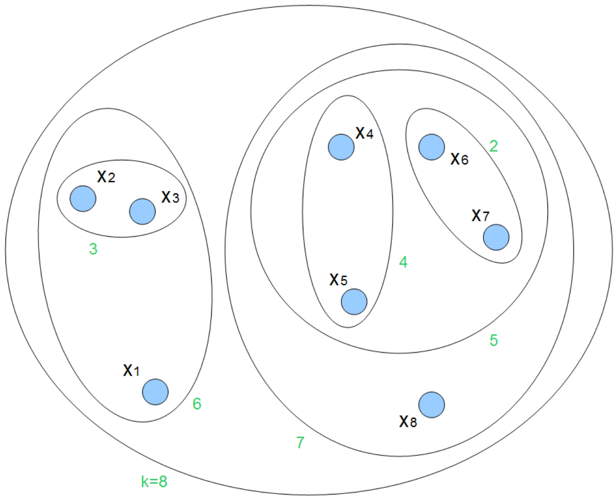


Figure 4.6: Venn diagram representation of two-dimensional data - each dot represent our events.

In our case, ideally, each cluster should involve the chords characterized from the same tonality degree and their merge, depending on the threshold level, should obtain the grouping of the chords representing the stable degrees. At the evaluation part, we analyze in detail what happens when our input is based on Chorals by Bach. We can see there how crucial is the appearance of the "important" clustering levels, i.e. the ones that have the biggest amount of elements, something that plays a key role at detecting the elements for the generation part.

4.4 Statistical Model for generating sequences

Having already the segments of the input sound and categorized properly, the next step is to re-generate them in a different sequence than the original one, taking into account that they are not independent and identically distributed; in contrary, the sequential patterns that will be established will follow the rule which base is that every new choice of segment should consider past observations. For implementing this idea it would be impractical to consider a general dependence of future observations on all previous observations because the complexity of such a model would grow without limit as the number of observations increases.

Let's assume $x_i \in [1, N]$ as the events which are elements of an infinite categorical space $\mathcal{X}$. In that point we can mention this product rule from which we can express the joint distribution for a sequence of observations in the form:

$$p(\mathrm{x}_1, ...x_N) = \prod_{n=1}^{N} p(x_n|x_1, ..., x_n - 1) \tag{4.5}$$

This leads us to consider Marcov models in which we assume that future predictions are independent of all but the most recent observations; for example let's assume that we have N observations. Then obtaining a *first-order Marcov chain* means that the contitional choice of a particular observation x_n depends on the value of the previous observation x_{n-1} and the joint distribution is given by

$$p(\mathrm{x}_1, ...x_N) = p(x_1) = \prod_{n=2}^{N} p(x_n|x_{n-1}) \tag{4.6}$$

A *second-order Marcov chain* means that the conditional choice of a particular observation x_n depends on the value of the two previous observation x_{n-1} and x_{n-2} and the joint distribution is given by

$$p(\mathrm{x}_1, ...x_N) = p(x_1)p(x_2|x_1) \prod_{n=3}^{N} p(x_n|x_{n-1}, x_{n-2}) \tag{4.7}$$

and so on.

In most applications of such models, the conditional distributions $p(x_n|x_{n-1})$ that define the model, will be constrained to be equal, corresponding to the assumption of a stationary time series. The model is then known as a *homogeneous* Markov chain. For instance, if the conditional distributions depend on adjustable parameters, then all of the conditional distributions in the chain will share the same values of those parameters [10].

Now we have to see what the *Variable Lenght Marcov Chains* are and why their evaluation to our system is special, having the additional attractive structure that their memories depend on a variable number of lagged values; that means that the transition probabilities are determined by looking back a variable number of lagged values, depending on how such a lagged-value history looks like (i.e. the index values of the last observation).
This characteristic is clear through the following definitions:

Definition 1.
Let $(X_t)_{t\in\mathbb{Z}}$ be a stationary process with values $X_t \in \mathcal{X}$, $|\mathcal{X}| < \infty$.
Denote by c: $\mathcal{X}^\infty \to \mathcal{X}^\infty$ a (variable projection) function which maps an infinite sequence (the infinite past) to a possibly shorter string (the relevant past). c: $x^0_{-\infty} \longmapsto x^0_{-\ell+1}$,
where $\ell = \ell(x^0_{-\infty})$ =min$\{k; \mathbb{P}[X_1 = x_1|X^0_{-\infty} = x^0_{-\infty}] = \mathbb{P}[X_1 = x_1|X^0_{k+1} = x^0_{-k+1}]\ \forall x_1 \in \mathcal{X}\}$,
and $\ell \equiv 0$ corresponds to independence.
Then $c(\cdot)$ is called a context function and $\forall t \in \mathbb{Z}$, $c(x^{t-1}_{-\infty})$ is called the context for the variable x_t.

By the projection structure of the context function $c(\cdot)$, the context-length $\ell(\cdot) = |c(\cdot)|$ determines $c(\cdot)$ and vice versa.

Definition 2.
Let $(X_t)_{t\in\mathbb{Z}}$ be a stationary process with values $X_t \in \mathcal{X}$, $|\mathcal{X}| < \infty$ and corresponding context function $c(\cdot)$ as given in *definition 1.* Let $0 \leq p \leq \infty$ be the smallest integer such that
$|c(x^0_{-\infty})| = \ell(x^0_{-\infty}) \leq p$, $\forall\ x^0_{-\infty} \in \mathcal{X}^\infty$.
Then $c(\cdot)$ is called a context function of order p, and if $p < \infty$, $(X_t)_{t\in\mathbb{Z}}$ is called a stationary VLMC of order p.

Clearly, a VLMC of order p is a Markov chain of order p, with the additional structure of having a memory of variable length. Such a structure implies that some of the transition probabilities are the same for various states of (the embedding) Markov chain [21].

In order to obtain an intuitively view of the VLMCs, we can represent them as a tree, following this definition:

Definition 3.
Let $c(\cdot)$ be a context function of a stationary VLMC. The context tree τ is defined as
$\tau = \tau_c = \{w; w = c(x^0_{-\infty}),\ x^0_{-\infty} \in \mathcal{X}^\infty\}$.

So now we can use the example located at [11] and [21], just to clarify the above determinations:

Let's assume that we have the space $\mathcal{X} = \{0, 1\}$ and order $p = 3$ then the context function can be defined as:

$$c(x_{-\infty}^{0}) = \begin{cases} 0, & \text{if } x_0 = 0, x_{-\infty}^{-1} \text{ arbitrary} \\ 1,0,0, & \text{if } x_0 = 1, x_{-1} = 0, x_{-2} = 0, x_{-\infty}^{-3} \text{ arbitrary} \\ 1,0,1, & \text{if } x_0 = 1, x_{-1} = 0, x_{-2} = 1, x_{-\infty}^{-3} \text{ arbitrary} \\ 1,1, & \text{if } x_0 = 1, x_{-1} = 1, x_{-2}^{-\infty} \text{ arbitrary} \end{cases}$$

That can be presented through this tree construction:

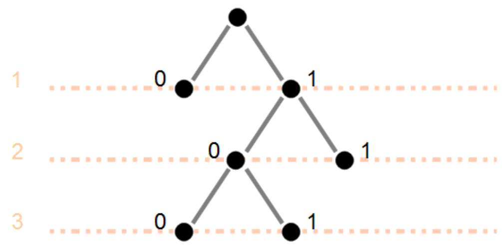


Figure 4.7: Tree representation of the context function $c(\cdot)$

From this representation, we can follow the nodes from upside down and we get all the selections from the context function. It is clear also that the options from each node is the number of elements of the space, $|\mathcal{X}|$.

The next step is to see how the VLMCs are implemented in our system, through a simplified example. Let's assume that we have, as an input, two sequences of events - elements of a categorical space having length $\ell = 4$. Let it be (A,B,C,A) and (B,C,C,D), which are parsed from right to left. At [3], context trees are created where a list of continuations encountered in the corpus are attached to each tree node. The "continuations" are integer numbers which denote the index of continuation item in the input sequence. At figures 4.8 and 4.9, we can see the procedure of the context tree creation of the sequence (A,B,C,A) and (B,C,C,D) respectively, where the green index numbers simply show in which element you can go next.

Exploring the final graph at figure 4.9, where the trees above are merged, we have all the possible sequence situations, following each path that is created from bottom to up and considering the index number of the first element. For example, if we want to find which is the next element of the sequence (A,B,C), we follow this specific path from the bottom of the tree and then we see the index number of the first element, A, which is 4, so we take the element with this index number, which is A and the sequence now becomes (A,B,C,A). As "e" we consider a random selection of any event.Also the length ℓ can be variable. For more details, the generation strategies are described at [11].

Fitting VLMCs from data is a nontrivial computational task (O(n log(n)) operations for an efficient algorithmic implementation).

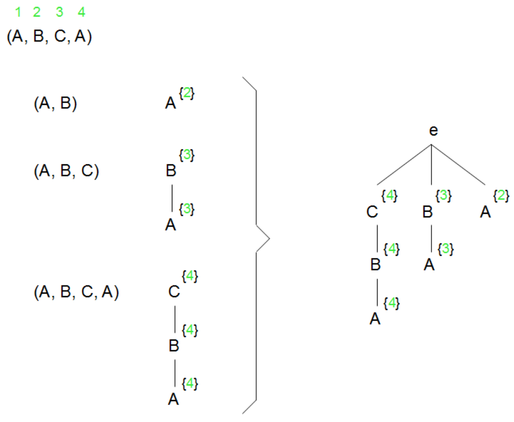


Figure 4.8: Tree representation of the context function $c(\cdot)$

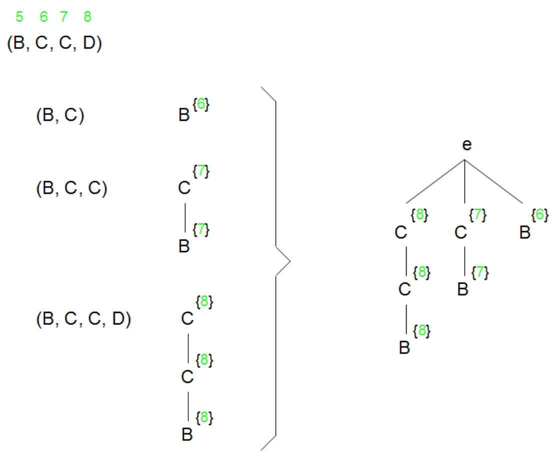


Figure 4.9: Tree representation of the context function $c(\cdot)$

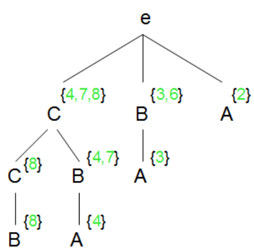


Figure 4.10: Context tree built from the analysis of the sequences (A B C A) and (B C C D).

Chapter 5

Evaluation and Validation

"Oh, people can come up with statistics to prove anything. 14% of people know that."
Homer Simpson

In order to evaluate the method, the first option was to find pieces of music, properly fitted to the onset detection and tempo estimation philosophy selected. That means, the chord sequences of the original sound should be properly strict in tempo thus the notes should be played more staccato than legato or sustain as well, since in this thesis the focus is not on advanced onset detection methods. Having that in mind, some piano recording sessions have taken place at ESMUC studio by me, so that we could have four Bach's chorals in a special piano version, following the restrictions above. Also some random played chords have been recorded, from piano and guitar as well.

5.1 Evaluation of the Clustering part

At this point, we present the clustering results we have got, having as input the audio "choral4.wav" in G major (for more details, the score, as well as its harmonic analysis can be found at the Appendix B). The system builds 10 clustering levels, each one having specific groups of elements and we will follow the procedure as some of them are merged, mounting the cluster hierarchy.

The cluster levels 1-9 will be examined one-by-one so that we can have more precise view. We only consider the clusters having more than one element, so that a meaning can be deduced. At the first column each cluster is defined by a number and at the second one we can see which elements - segments are inside that cluster. Finally, at the third column we *recognize* these segments through the score' s harmonic analysis, appeared at Appendix B, for each one separately (for example: "2 G I" means "2 of the elements are the root of G major", "1 d IV" means "1 of the elements is the subdominant of d minor" and "5 a V" means "5 of the elements are the dominant of a minor). That means that in this part we don' t rely on the key approximation code, so that we can clarify the specific errors we have only from this procedure.

⋇ Level 1.

Cluster	No. of Elements	Recognition
1	4, 64, 66	2 G I, 1 G V
2	33, 47, 50	1 G I, 1 a V, 1 d IV
3	2, 36	2 G IV
4	35, 37	1 G I, 1 G V
5	14, 18, 21, 23, 49, 52, 53, 54, 59, 67	5 G V, 1 a I, 1 d I, 1 d VI, 1 d V, 1 G I
6	11, 46	1 G IV, 1 a I
7	6, 41, 43, 56	1 G II, 1 a V, 1 a I, 1 d V
8	7, 29	2 G V

⋇ Level 2.

Cluster	No. of Segments	Recognition
3	2, 36	2 G IV
8	7, 29	2 G V
9	(cl. 5) + 55, 60	6 G V, 1 a I, 2 d I, 1 d VI, 1 d V, 1 G I
10	(cl. 2 + cl. 7) + 61	1 G I, 2 a V, 1 d IV, 1 G II, 1 a I, 1 d V
11	(cl. 4) + 12	2 G I, 1 G V
12	(cl. 1 + cl. 6) + 20	2 G I, 1 G V, 1 G IV, 1 a I

⋇ Level 3.

Cluster	No. of Segments	Recognition
8	7, 29	2 G V
12	(cl. 1 + cl. 6) + 20	2 G I, 1 G V, 1 G IV, 1 a I
13	(cl. 11) + 5	3 G I, 1 G V
14	(cl. 3 + cl. 9 + cl. 10) + 22, 45	2 G I, 1 G II, 2 G IV, 6 G V, 2 a I, 2 a V, 2 d I, 1 d IV, 2 d V, 1 d VI

⋇ Level 4.

Cluster	No. of Segments	Recognition
12	(cl. 1 + cl. 6) + 20	2 G I, 1 G V, 1 G IV, 1 a I
15	(cl. 8 + cl. 13 + cl. 14) + 57, 58	9 G V, 5 G I, 1 G II, 2 G IV, 2 a I, 2 a V, 3 d I, 1 d IV, 2 d V, 2 d VI

⋇ Level 5.

Cluster	No. of Segments	Recognition
16	(cl. 12) + 13	3 G I, 1 G V, 1 G IV, 1 a I
17	(cl. 15) + 26, 42, 44	9 G V, 6 G I, 1 G II, 2 G IV, 3 a I, 2 a V, 1, a IV, 3 d I, 1 d IV, 2 d V, 2 d VI

※ **Level 6.**

Cluster	No. of Segments	Recognition
18	(cl. 16 + cl.17) + 24	10 G V, 9 G I, 1 G II, 3 G IV, 4 a I, 2 a V, 1, a IV, 3 d I, 1 d IV, 2 d V, 2 d VI
19	15, 48	1 a I, 1 G V

※ **Level 7.**

Cluster	No. of Segments	Recognition
19	15, 48	1 a I, 1 G V
20	(cl.18) + 8, 27, 36, 47	10 G V, 10 G I, 1 G II, 4 G IV, 4 a I, 3 a V, 1 a IV, 3 d I, 1 d IV, 2 d V, 2 d VI, 1 C I
21	1, 25, 63	1 G I, 1 G IV, 1 d V

※ **Level 8.**

Cluster	No. of Segments	Recognition
19	15, 48	1 a I, 1 G V
22	(cl. 20 + cl. 21) + 9, 17, 20, 63	11 G V, 12 G I, 1 G II, 5 G IV, 4 a I, 3 a V, 1, a IV, 3 d I, 1 d IV, 4 d V, 3 d VI, 1 C I

※ **Level 9.**

Cluster	No. of Segments	Recognition
23	(cl. 19 + cl. 22) + 3, 16, 19, 25, 40, 46	13 G V, 13 G I, 1 G II, 6 G IV, 6 a I, 4 a V, 1 a IV, 3 d I, 1 d IV, 5 d V, 3 d VI, 1 C I

After that process the following comments are deduced. It seems crucial the issue that a segment contains single notes, for its recognition. That is why we possibly have the cluster No. 4. Also at cluster 12 we notice the merge of the stable degrees I, IV, V of G Major for the first time. In the cluster 14 we have a *rich* group, containing not only a big amount of G Major dominant chords, but also the stable degrees of its minor fifth. Representatives of the stable triads for a minor (which is fifth of the minor fifth of G) appears in cluster 17. At fig. 5.1, there is a tree representation, clarifying visually the procedure presented above, for the levels 1-6. At the base line we can see the clusters generated at Level 1 as circles; the blue ones are some of them containing one single element, which number is mentioned inside and the numbers under the yellow and red ones are the numbers of the clusters they represent. As we follow the graph from bottom to up, we clearly notice how some specific clusters are merged. The height of each cluster representation lines should not be taken into account.

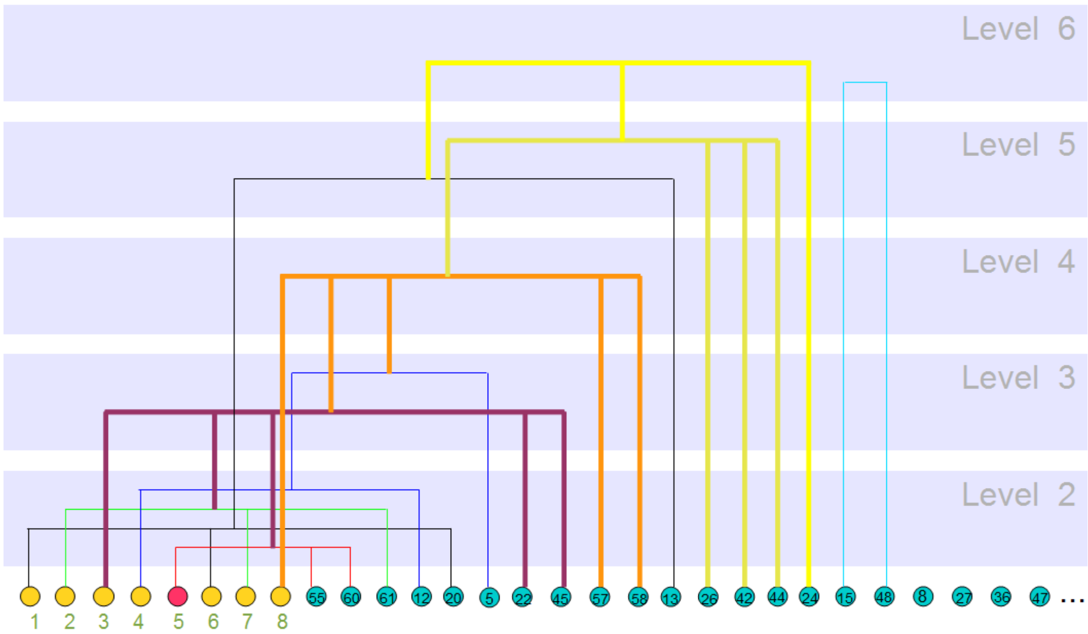


Figure 5.1: Tree representation of the clustering procedure, for levels 1-6.

5.2 Evaluation Questionnaire

The next step of the validation procedure was to create some interesting generations, using various audio inputs. We wanted an evaluation, based on subjective opinion, so a questionnaire for each input and its generations has been created and given to four musicians and four non-musicians. To be more specific, for that purpose we have five different kinds of inputs, followed each one by generations of one minute duration and in each case it is not clarified which audio is the original and which one is the generation.

Every questionnaire follows the same philosophy: the subjects have to listen to each audio and answer to the questions appeared inside the table above. At the forth question briefly we pose the subject to compare two audios; each time the combinations are variable.

1. Does this piece seem familiar to you (structure/rhythmic patterns/chord sequences)?
-Please rate on scale of 1 (not at all) to 5(very much)-

2. How interesting is it for you (in terms of chord sequences/rhythmic patterns)?
-Please rate on a scale of 1 (very uninteresting) to 5 (very interesting)-

3. If your answer in the previous question was 4 or 5, please mention max. 10 seconds of the audio that triggered your interest most.

4. Does xxx.wav sound similar in style to the yyy.wav? (put a + next to your reply)
Not similar.
Somewhat similar.
Very similar.

When the input is a classical piece (cases 2 and 3) there is another question added:

Rate how clear does the structure (theme, variations etc.) of the piece is.
-Please rate on scale of 1 (not at all) to 5 (very much)-

Each generation of an input is a different version, depending on some specific values to some parameters. Firstly we choose each time if we define which clustering level is selected or not, using the parameter *clust_level*. Then with the *maxcontextlen* we control how much an excerpt of the generation can be the same as the original (in duration) -the bigger the value of this parameter, the bigger the duration that is identical to the original. Finally, the parameter *recombinance* is used, which counts how many times an event segment is followed by a non-contiguous segment (i.e. how many "jumps" the generated sound contains, in the 30 second generation) [11]. This parameter' s values range from 0 to 1: if it is close to 1, we don 't take the same order as the original. For more details, tables with these values for each generation can be found at the Appendix A.

Then the responses from both musicians and non musicians for each input are provided.

Example 1.
Guitar chord sequence, based on the song "If I fell in love" by Beatles

Here, we have presented three audios, following this sequence:
⁕ Audio 1 - Generation 1 - Characteristics: The last 3 chords of the generated audio are used at the end, following the same sequence of the original (cadence).
⁕ Audio 2 - The original.
⁕ Audio 3 - Generation 2 - Characteristics: big repetition of the same type of chord as a non-endless loop.

The responses of our subjects are shown in the following table:

	Musicians			Non-musicians		
	Familiarity	Interesting	Seconds	Familiarity	Interesting	Seconds
Original	2,1,3,1	2,2,4,4	22-30	2,2,3,2	3,4,4,2	38-42
			11-16			22-32
Generation 1	2,1,3,1	2,2,3,5	4-12	2,3,2,2	4,3,3,2	1-11
Generation 2	5,1,3,1	2,2,3,2		3,3,4,2	2,5,4,2	48-58
						45-55
Similarity	Org.-Gen.1	Org.-Gen.2		Org.-Gen.1	Org.-Gen.2	
Not similar						
Somewhat similar	+	+		+	+++	
Very similar	+++	+++		+++	+	

Through the seconds that triggered their interest, we can see that this specific section selection in generation 1 contains an excerpt which is different from the original input and the section mentioned in generation 2, which is after the big loop of the same chord, it is part of the same section of the original which was defined as "interesting" by the same group of subjects - non musicians.

Something else that is clear, through the responses, is that the similarity is more emphasized by musicians than by non-musicians. We have also the following comments:
"I noticed that the three pieces were interesting in terms of chord sequences (except maybe from 3.wav because of the repeated chords). They were also quite similar to each other".
"The structure of the first one is more "complicated" and of the third one really simple".

Example 2.
Bach choral played on the piano.

Here we examine how the system copes with a classical piece as an input (”choral1”, which score is located at the appendix B). The audios that were presented to the subjects are the following:
⋇ Audio 1 - The original.
⋇ Audio 2 - Generation 1. Characteristics: pretty shuffled.
⋇ Audio 3 - Generation 2. Characteristics: big repetition of the same segment as an endless loop at the end.

The responses of our subjects are shown in the following table:

	Musicians			Non-musicians		
	Familiarity			Familiarity		
Original	4,4,4,5			3,3,4,2		
	Clearness	Interesting	Seconds	Clearness	Interesting	Seconds
Generation 1	4,5,5,3	3,5,4,2	30-40	4,5,4,4	4,5,4,4	1-11
			30-40			1-11
						45-55
						30-40
Generation 2	5,4,3,2	1,4,3,3	23-32	4,4,3,3	2,3,4,2	26-36
Similarity	Org.-Gen.1	Org.-Gen.2		Org.-Gen.1	Org.-Gen.2	
Not similar		+				
Somewhat similar	+	+		++	++	
Very similar	+++	++		++	++	

The ”interesting” excerpts mentioned above are all varied from the original audio. Especially for the first generation, we can notice from both groups of subjects that the dissimilar variations which are marked can make the example worth noticing. We have also one comment that punctuates the special characteristic between these two generations: ”The lack of interest in the third piece is due to the chords appeared in the last half of it (i.e the repeated one)”.

Example 3.
Funeral March by Chopin.

The original audio for this example is the first 2.38 minutes of the third movement of piano sonata No.2 in B flat minor by Chopin. We can devide it in two parts, A and B, where the first one, which tonality is different from the other, is from the beginning of the excerpt until 1.13' and it is repeated after the second one, which is from 1.14' until 2.27'. So we have produced two generations, qualified by the following characteristics:
⋇ Audio 1 - The original.
⋇ Audio 2 - Generation 1. Characteristics: uses only excerpts of the part A.
⋇ Audio 3 - Generation 2. Characteristics: there are excerpts from both parts and it is pretty shuffled in part B.

The responses of our subjects are shown in the following table:

	Musicians			Non-musicians		
Original	Familiarity 5,5,4,5			Familiarity 4,5,5,5		
	Clearness	Interesting	Seconds	Clearness	Interesting	Seconds
Generation 1	5,5,5,3	5,5,3,3	0-10	5,5,3,4	5,5,3,3	33-43
			43-53			43-48
Generation 2	5,4,4,2	5,4,4,3	34-44	4,5,3,5	5,5,4,4	17-24
			43-51			34-44
						45-52
						20-30
Similarity	Org.-Gen.1	Org.-Gen.2		Org.-Gen.1	Org.-Gen.2	
Not similar						
Somewhat similar	+	+++			+	
Very similar	+++	+		++++	+++	

We can discern that the first generation appears more similar to the original, than the second one, although the part B is not used at all during this generation. This can be based on the fact that we have some tempo variations during the second one, a senctioned element that makes it perceived as not very similar to the original. That is why we had some comments like the following: "At 0.46' of the third audio (i.e. generation 2) I couldn' t estimate the tempo".

As for the "interesting" point of view, the next comment captures the main meaning coming through the "Seconds of interest" section: "Those three pieces were not interesting in terms of chord sequences but they actually were captivating because of the variations, the volume changes and the low notes".

Example 4.
Guitar Flamenco

Here, we have presented three audios from a guitar example, playing this specific kind of music, following this sequence:
⋇ Audio 1 - The original.
⋇ Audio 2 - Generation 1 - Characteristics: pretty close to the original.
⋇ Audio 3 - Generation 2 - Characteristics: more shuffled than the previous one.

In the following table we can see the responses of our subjects:

	Musicians			Non-musicians		
	Familiarity	Interesting	Seconds	Familiarity	Interesting	Seconds
Original	1,2,1,5	4,2,4,4	0-10	3,2,3,2	4,3,3,1	1-8
			34-38			
			28-38			
			1.05-1.07			
Generation 1	1,2,1,5	4,2,3,4	0-10	4,1,4,2	3,3,3,1	
			8-14			
Generation 2	1,2,1,5	1,2,5,3	7-15	2,1,3,2	3,3,4,1	32-42
						45-55
Similarity	Org.-Gen.1	Org.-Gen.2		Org.-Gen.1	Org.-Gen.2	
Not similar						
Somewhat similar		+++		+	+++	
Very similar	++++	+		+++	+	

Phase errors have occured during the generations of this example, especially at the second one, having as a result some rythmic pattern discontinuities at some points. This fact governed by two kinds of approaches: some of the subjects considered these sections as ”confusing” and some others as ”intriguing expertise”. This can been seen through the following comments, respectively:
”It was hard for me to follow the rythmic pattern of the third one (i.e. Generation 2)”.
”Piece No.3 (i.e. Generation 2) had a more complicated rhythm that was quite hard to follow as a listener but made it more interesting than the previous two pieces”.

We can also distinguish through the similarity part that the subjects indeed realized the relation, based on the parameters' values choice, which are shown at Appendix A, between the original and the first generation.

Example 5.
Piano Chord sequence by a non-musician.

We want also to examine how the system acts when the original input is not based on a specific piece of music but it is produced by a non musician. Here, we have presented two audios, following this sequence:
⋇ Audio 1 - Generation - Characteristics: as far from the original as we could, using specific parameters' values mentioned at Appendix A.
⋇ Audio 2 - The original.

The responses of our subjects are in the following table:

	Musicians			Non-musicians		
	Familiarity	Interesting	Seconds	Familiarity	Interesting	Seconds
Original	1,2,1,3	1,2,2,2		1,1,3,3	2,2,2,2	
Generation	1,2,1,4	1,2,3,3		1,1,2,3	2,2,3,2	
Similarity	Org.-Gen.			Org.-Gen.		
Not similar						
Somewhat similar	++++			+++		
Very similar				+		

The average of both musicians and non musicians agreed that the sound examples are not very similar to each other, as it is shown at the following comment as well: "The chord sequences in both pieces were not very common and that made them more interesting on the one hand but less pleasant to the ear on the other". It is mentioned through their responses also that the generation is more interesting than the original. Another important issue is the quality of the generation, in terms of harmonic structure. The next comment shows a positive step in that field, as the *wrong* chord sequences weren' t repeated in that. "In the second audio (i.e. the Original) I could hear many harmonically false sequences".

Gereral results

We can highlight that only 1.25% of the responses found the generation example as *not similar* to the original input. Also through the examples 1, 4 and 5 we notice that 20% of the responses found the generation example more *interesting* than the original,when the average rate of them is 2.25, 30% of the responses found the generation example the same *interesting* as the original and 50% of the responses found the generation example less *interesting* than the original, having the deviation of rate at only 0.375.

Chapter 6

Conclusions - Future Work

"I never use a score when conducting my orchestra... Does a lion tamer enter a cage with a book on how to tame a lion?"
Dimitris Mitropoulos

6.1 Conclusions

The system that is described in this thesis, based on harmonic ambient, combines machine learning techniques, signal processing, together with music theory knowledge, in order to resynthesize sequences and to finally construct a new sound from a given example. This can have many applications. As the questionnaire results highlight, the generation is similar to the original input, mainting the key features of the latter. Also it is possible for the generation to be more interesting than the original, that is why it could be a helpful composing material. The feeling of creating a nice musical piece from any kind of this specific imput provides a new character to interdisciplinary applications and a big range of their capabilities.

6.2 Future work

An immediate evaluation should be the exploration of different ways in order to contemplate the composition style of the generated sound. One thought could be to consider the resynthesis audio signal from ConstantQ and change the chords from major to minor and vice versa. Also the tempo alternation, not only for the whole piece, but in some specific parts, as well as the change of the playing style, from legato to staccato for example should be two important modifications which create a different expression "id".

Another different point of view of the generated sound could be accomplished by implementing our code only in some specific variations of the original piece. That means this selective system has to be able to recognize the theme, its location inside and also to keep it intact. Some crucial work in the field of interval recognition has been held at [24] and [25], where musical patterns have been analysed through psychological research. To be more specific, at the first one, properties of joint accent structures involving accent couplings and time symmetries are used to address standard psychological issues of pattern similarity and pattern simplicity as they are realized in musical tasks. At the second one, an approach for fast discovering of all non-trivial repeating patterns in music objects is proposed. In that way the theme, which is defined by them as the longest most repeated excerpt, can be habituated for content - based retrieval of music data.

The next step for the system is to become a real time configuration, as it is well described at [11]. A fact that can improve the evaluation process should be the acceptance and automatic analysis of more complicated harmonic structures. Another interesting evaluation should be the implementation of the system having as an input an audio example with more than one instruments.

Chapter 7

Appendix A

Evaluation Questionnaire Audio Basic Parameters

Example 1.

Generation audios:

	Generation 1	Generation 2
clust_level	1	random
maxcontextlen	2	4
recombinance	0.5	0.9

Example 2.

Generation audios:

	Generation 1	Generation 2
clust_level	random	5
maxcontextlen	2	4
recombinance	0.5	0.9

Example 3.

Generation audios:

	Generation 1	Generation 2
clust_level	random	random
maxcontextlen	2	2
recombinance	0.5	0.8

Example 4.

Generation audios:

	Generation 1	Generation 2
clust_level	random	1
maxcontextlen	4	2
recombinance	0.2	0.8

Example 5.

Generation audios:

	Generation 1
clust_level	random
maxcontextlen	1
recombinance	0.1

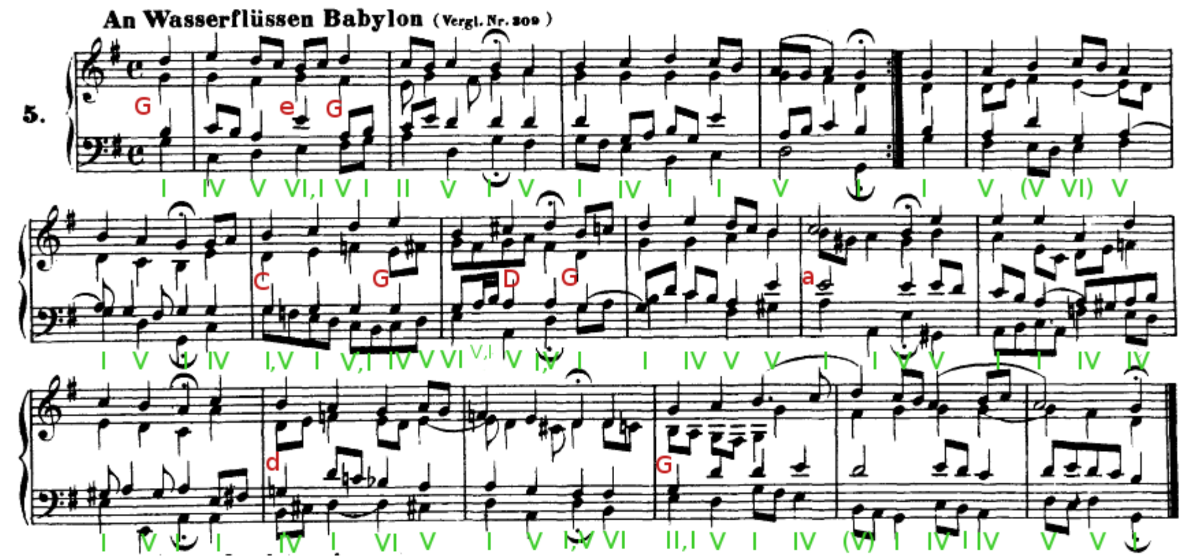
Chapter 8

Appendix B

Bach Chorals Scores

In that section we present the scores of the four Bach chorals which were used as part of the evaluation of the system. At the first one, the harmonical analysis is marked by me.

"Choral4.wav"



"Choral1.wav"

BWV 2.6

Harmonized by J.S. Bach

"Choral2.wav"

BWV 5.7

Harmonized by J.S. Bach

"Choral3.wav"

BWV 7.7

Harmonized by J.S. Bach

→

BWV 7,7

Harmonized by J.S. Bach

Chapter 9

Bibliographic references

[1] Iannis Xenakis, Formalized Music, Thought and Mathematics in Composition, Indiana University Press, Bloomington London, pg. 1-4.

[2] S. Dubnov, G. Assayag, and R. El-Yaniv. Universal classification applied to musical sequences. In Proceedings of the International Computer Music Conference, pp. 332-340, 1998.

[3] Francois Pachet. The continuator: Musical interaction with style. In ICMA, editor, Proceedings of ICMC, pp. 211-218. ICMA, September 2002.

[4] In S. Yantis (Volume Ed.) and H. Pashler (Series Ed.), Stevens Handbook of Experimental Psychology, Volume 1: Sensation and Perception (Third Edition, pp. 453-492), Timothy C. Justus, Jamshed J. Bharucha, pp .463-465

[5] Hendrik Purwins, Profiles of Pitch Classes Circularity of Relative Pitch and Key Experiments, Models, Computational Music Analysis, and Perspectives, pp. 51-55, Berlin Institute of Technology, 2005.

[6] Dominik Hornel and Wolfram Menzel, Learning Musical Structure and Style with Neural Networks, Computer Music Journal, 22:4, pp. 44-62, Winter 1998

[7] Moray Allan, Christopher K. I. Williams, Harmonising Chorales by Probabilistic Inference, Edinburgh, 2005.

[8] Amelie Anglade, Rafael Ramirez, Simon Dixon, First-order logic classification models of musical genres based on harmony, Proceedings of the SMC 2009 - 6th SMC conference

[9] Richard O. Duda, Peter E. Hartl, David G. Stork, Pattern Classification (2nd edition), pp. 517-557

[10] Christopher M. Bishop, Pattern recognition and machine learning, pp. 605-615

[11] Marchini M., Purwins H., Unsupervised Generation of Percussion Sound Sequences from a Sound Example, SMC master thesis, 2010

[12] Arnold Schoenberg, Theory of Harmony, translated by Roy E. Carter, University of California Press, Berkeley, Los Angeles

[13] C. Schorkhuber, Anssi Klapuri, Constant-Q transform toolbox for music processing, Queen Mary University, paper submitted to the 7th Sound and Music Computing Conference, Barcelona, Spain

[14] H. Purwins, B. Blankertz, K. Obermayer, A New Method for Tracking Modulations in Tonal Music in Audio Data Format, CCRMA, Stanford
[15] M. Leman. Schema-based tone center recognition of musical signals. Journal of New Music Research, 23:169204, 1994
[16] S. Dubnov, G. Assayag, A. Cont, Audio Oracle: A New Algorithm for Fast Learning of Audio Structures, in Proceedings of the International Computer Music Conference (ICMC), 2007
[17] `http://www.pgmusic.com`, band-in-a-box, 2011
[18] Judith C. Brown and Miller S. Puckette. An efficient algorithm for the calculation of a constant Q transform.,J. Acoust. Soc. Am., 92(5):26982701, 1992
[19] `http://aubio.org`
[20] C. Krumhansl, Cognitive Foundations of Musical Pitch. Oxford University Press, Oxford, 1990
[21] M. Machler, P. Buhlmann, Variable Length Markov Chains: Methodology, Computing and Software, Seminar fur Statistik, Eidgenossische Technische Hochschule (ETH), CH-8092 Zurich, Switzerland, Research Report No. 104, March 2002
[22] http://www.amaranthpublishing.com/MozartDiceGame.htm , 25/7/2011
[23] MIDI standards, a brief history and explanation, J. Pisano, 2006, http://mustech.net/2006/09/midi-standards-a-brief-history-and-explanation/ , 25/7/2011
[24] Mari R. Jones, Dynamic pattern structure in music: Recent theory and research, Attention, Perception and Psychophysics, Vol.41, No. 6, 621-634
[25] Chih-Chin Liu, Jia-Lien Hsu, Arbee L. P. Chen, Efficient Theme and Non-Trivial Repeating Pattern Discovering in Music Databases, International Conference on Data Engineering 1999, p. 14-21, Sydney. Current Version August 2002
[26] `http://www.dolmetsch.com/musictheorydefs.htm`, 4/9/2011
[27] `http://library.thinkquest.org/2791/MDOPNSCR.htm` 4/9/2011
[28] `http://labrosa.ee.columbia.edu/matlab/chroma-ansyn/` 23/5/2011
[29] `http://www.iua.upf.edu/~egomez/teaching/smc/AMA/` 23/5/2011