Funchal: a System for Automatic Functional Harmonic Analysis

Ricardo Scholz, Vítor Dantas, Geber Ramalho

Centro de Informática – Universidade Federal de Pernambuco (UFPE)
Caixa Postal 7.851 – 50.732-970 – Recife – PE – Brazil

{reps,vcd2,glr}@cin.ufpe.br

Abstract. *Functional harmonic analysis is an important task in music composition, accompaniment, arrangement and others. However, the solutions are still not satisfactory. The proposed process is divided into two levels: the first one extends one of previous works in the domain to carry out a richer analysis of chord grids and is where the very analysis is performed, and the second one is devoted to correct some conceptual inconsistencies concerning enharmonic spelling of chords. Both levels use an engine to make inferences on some rule bases, which can be easily improved by addition of new rules.*

1. Introduction

Functional harmonic analysis consists of attributing functions to each chord of a given chord grid, considering the context in which the chords are inserted. The main targets criteria for evaluating a system that performs functional harmonic analysis are correctness, extensibility and ability to recognize most of the patterns used.

The previous works studied did not reach some of these criteria. Our proposed process is divided into two levels: analysis and chord names correction. The first extends the work of Pachet [Pachet 1991], reused by Ramalho [Ramalho 1997]. We extended the approach to deal with more complicated structures, such as modal borrowing and secondary dominants and added a level for correcting chord spelling mistakes.

Section 2 will present the state of the art on this subject. Next, section 3 discusses in depth the approach that we used to do the automatic functional harmonic analysis. Section 4 shows the results obtained and conclusions of this work.

2. State of the Art

The perception of harmony in music is a task that is done very naturally by humans, but has been proven to be a complex task to describe algorithmically. The modeling of this task was the target of many psychologists [Krumhansl, 1990] and computer science researchers. For instance, Bharucha have done experiments with connectionist networks to simulate harmonic perceptions [Bharucha, 1991].

Some works assumed basic harmonic structures as part of the input, in form of rule-bases, such as Lerdahl and Jackendoff's generative theory of tonal music [Lerdahl and Jackendoff, 1983] and Narmour's implication/realization model [Narmour, 1990]. Temperley did recently extentions to these works [Temperley, 2001], implemented his solutions and did extensive tests. Other authors used case-based reasoning to avoid the flaws of assuming wrong predefined structures [Sabater, Arcos, and Mántaras, 1998].

Besides, a number of studies deal with what is known as roman numeral harmonic analysis, that is the way it is taught in music schools. Some authors proposed context-free grammars to realize the analysis iteratively, generating complex harmonic structures [Ulrich, 1977] [Steedman, 1984]. Pachet's approach is different and consists in dividing the analysis in three steps: search for recurrent patterns, overlapping patterns removal and finally unclassified chords are given a function. [Pachet, 1991].

In our work, we used Pachet's division in levels, and we have made a number of extensions to recognize more complex harmonic structures. We also added new a step to correct the names of chords that can be misspelled in the input. Our analysis level deals only with pitch information, without considering the spelling, such as Temperley's analysis did.

3. Automatic Functional Harmonic Analysis

The approach proposed works like growing islands, so it deals with modulation automatically, since neighbor islands are independent and can have different tonalities. In addition to this, the rules proposed deals with complex harmonic structures, as modal borrowing, secondary dominant and composite functions. Some of then are not covered by any of the cited works. The rules that will be shown were identified based on interviews with senior musicians with a good knowledge on functional harmony.

3.1. Chord Pattern Matching

This step consists in searching for recurrent and unambiguous chord sequences. We consider simple patterns sequences of chords with no complex harmonic structure. Unambiguous patterns are chord sequences that cannot have two possible harmonic analysis. Examples of such patterns are II-V, IV-V-I, etc.

We use a rule base coupled with a first order inference engine, called JEOPS [Figueira 2000] that works on forward chaining. Most of the rules in this base were adapted from Ramalho [Ramalho 1997]. The others were created based on the result of a manual analysis of many songs with Jazz and Bossa Nova harmony.

All the rules apply to consecutive chords, considering only the number of steps between these roots. Once a rule is fired on a set of chords, a new pattern is created with the chords' functions. Then, the pattern is inserted in the patterns list. To any set of chords, if more than one rule apply, all will fire. We treat the ambiguous cases in the third step of analysis.

3.2. Overlapping Patterns Removal

After applying the first rule base to match the predefined patterns, the pattern list may contain overlapping patterns. The goal of this analysis step is to eliminate all the overlaps. To reach this goal, the rules of this base eliminate from the knowledge base the patterns considered to have low priority or to be redundant.

The rules were defined from the possible overlaps generated by the first analysis step. In a deeper analysis, we can notice that there are two types of overlaps: total, if a pattern is contained by another, or partial with a common function, if the last function of the first pattern matches the first function of the second. Only one rule fires on each set of overlapping patterns, according to the rule priority.

3.3. Gaps Classification

Finally, the third rule base has the goal of identifying the functions of the chords that didn't match any pattern. Once a rule has been fired to a chord, no other rule can fire to the same chord. This step is responsible for recognizing the most complex harmonic structures, as modal borrowing, secondary dominants and composed functions.

First, we try to classify the chords according to the harmonic fields of previous and following patterns. If the chord belongs to one of these harmonic fields, it is inserted in the pattern with its respective function in that harmonic field. Otherwise, we try to classify it using a composite function by considering the previous and following chords as roots of an harmonic field and trying to find a function that belongs to one of these harmonic fields and can be applied to the gap. If the rule fires, the gap is inserted in the pattern to which its neighbor chord belongs.

The next attempt is to classify the gap as a modal borrowing of its neighbor patterns. Then, a specific rule was created to classify substitute dominants, since this function is not identified in any other rule. Analogously, when it is not possible to classify the gap as substitute dominant of its neighbor patterns, we try to classify it as a substitute dominant of its neighbor chords.

Pachet [Pachet 1991] and Ramalho [Ramalho 1997] do not consider some of these cases, as modal borrowing and composite functions.

3.4. Chords Names Correction

There is a big amount of song lyrics mixed with chord notation available on the web. The approach was designed to make use of these inputs. However, web songs often have enharmonic spelling mistakes, as in the sequence IIm^7 V^7 I^{7M} that can be found as Ebm^7 $G\#^7$ Db^{7M}, while the correct option should be Ebm^7 Ab^7 Db^{7M}.

It is a good feature for the analysis level not to distinguish enharmonic notes, such as $G\#^7$ and Ab^7, since this way the functions may be identified even if the chord grid has mistakes. Since the system knows what the correct chord names are, based in the analysis made, it corrects the pitch name of the misspelled chords. All other works cited do not regard to this question.

The rules of this base works as follows: first, we try to use tonalities that had been used before within the same song. Then, if the first attempt is not possible, we use a tonality transition table. Finally, each pattern tonality is corrected locally, based in its last chord.

The use of the tonalities transition table has the goal of making it easier to read and execute the music, avoiding the use of complex tonalities or rough tonality changes. The table specifies what tonality, between all enharmonic options, the following pattern may use, according to the current pattern's tonality. The transition table is based on the fifths cycle and avoids double sharpened and double flattened tonalities.

4. Conclusions

Our system was built aiming to analyze music which harmony is functional. The tests were realized with Jazz and Bossa Nova songs, because of their rich harmony. We used a set of twelve harmonic analysis hand-made by senior musicians for comparison with the automatic analysis. The twelve songs of the test set comprise a total of 810 chords. In our experiments, while the specialist analysis left 3,52% of the chords unclassified, the automatic analysis left 1,76%. However, considering that the specialist analysis is the ideal analysis, 9,16% of the chords were classified by the system with alternative functions. Considering this context, the framework has reached good results with this small set of songs.

We have taken advantage of previous works, mainly Pachet [Pachet, 1991] and Ramalho [Ramalho, 1997], and we have made some significant extensions. More cases are now covered by the pattern matching level, for instance composite functions and modal borrowing. We also added a step for correcting the chord names after the analysis, which did not exist in previous works.

References

Bharucha, J. J. (1991) “Pitch, harmony, and neural nets: A psychological perspective”, Music and Connectionism, The MIT Press.

Figueira, C. S. F. (2000) “JEOPS: Integração entre Objetos e Regras de Produção em Java”, Centro de Informática, UFPE, Brazil.

Krumhansl, C. L. (1990) "Cognitive foundations of musical pitch". New York, USA, Oxford University Press.

Lerdahl, F. and Jackendoff, R. (1983) “A Generative Theory of Tonal Music”, Cambridge, MA, USA, The MIT Press.

Narmour, E. (1990) “The Analysis and Cognition of Basic Musical Structures: The Implication-Realization Model”, Chicago, USA.

Pachet, F. (1991) “A Meta-Level Architecture Applied to the Analysis of Jazz Chord Sequences”, Institute Blaise Pascal – Laforia, Université Paris VI, France.

Ramalho, G. L. (1997) “Construction D’un Agent Rationnel Jouant Du Jazz”, PhD Thesis. Université Paris VI, France.

Sabater, J. and Arcos, J. L. and Mántaras, R. L. (1998) “Using Rules to support Case-Based Reasoning for harmonizing melodies”, Multimodal Reasoning, AAAI Press.

Steedman, M. J. (1984) “A Generative Grammar for Jazz Chord Sequences”, University of Wariwik and University of Edinburgh, Scotland.

Temperley, D. (2001) “The Cognition of Basic Musical Structures”, Cambridge, MA, USA, The MIT Press.

Ulrich, J. W. (1977) “The Analysis and Synthesis of Jazz by Computer”, Computing and Information Science Department. University of New Mexico, Albuquerque, New Mexico, USA.