An Alignment Based System for Chord Sequence Retrieval

Pierre Hanna, Matthias Robine and Thomas Rocher*
LaBRI - Université de Bordeaux
F-33405 Talence cedex, France
firstname.name@labri.fr

ABSTRACT

Music retrieval systems for Western tonal music digital libraries have to consider rhythmic, timbral, melodic and harmonic information. Most existing retrieval systems only take into account melodies. Melody comparison may induce errors since two musical pieces can be very similar whereas their melodies may differ in a significant way. In this paper, we propose to investigate and experiment a retrieval system based on the comparison of chord progressions. The definition of chords may be ambiguous but their properties can be precisely described and represented. We detail the adaptations of alignment algorithms, successfully applied for the estimation of symbolic melodic similarity, for chord progression retrieval. Several experiments, performed on symbolic databases, show that the system described is robust to variations and outperforms a recent chord retrieval system.

Categories and Subject Descriptors

H.5.5 [**Information Systems**]: HCI—*Sound and Music Computing*

General Terms

Experimentation

1. INTRODUCTION

Research works in the domain of the musical information retrieval generally concern Western music libraries [7]. The properties of this kind of music imply the consideration of four main characteristics: rhythm, timbre, melody and harmony. One of the main open problem is the automatic estimation of music similarity. The applications are numerous and consist of browsing, retrieving or recommending music from large digital databases. Existing systems generally consider timbral similarities [2]. Recent studies investigate important properties related to tonal information [8].

*This work is part of the SIMBALS project (JC07-188930), funded by the French National Research Agency. This research is also supported by the Aquitaine Regional Council.

JCDL'09, June 15–19, 2009, Austin, Texas, USA.

Other researches restrain tonal information to melody and propose retrieval systems based on the melodic similarity [13, 10]. Limitations of such systems have been experimented when considering polyphonic music [16]. Furthermore, complexities of algorithms handling polyphony prevent the application to large databases [11]. It is also important to note that musical pieces may be very similar whereas their melodies are different. Therefore, considering the melody as the only criterion for music comparison seems to be limited for polyphonic retrieval. All these studies lead to development of retrieval systems based on chord progression instead of melody. Experiments have to be proposed in order to show if a chord progression based retrieval system is robust and discriminant enough to be applied to large databases, or if it can be combined with a system based on melodic similarity. In this paper, we propose a system for chord progression retrieval from symbolic music and experiments showing its robustness.

The algorithm we propose is based on alignment and is currently applied in string matching or bioinformatics [9]. This choice is justified by the results of experiments that show the importance of representing music with sequences [5]. Moreover, retrieval systems based on alignment algorithm have been experimented as very accurate, in particular systems considering sequences of notes [15]. But settings of such algorithms highly determine the precision of the system [10]. Our main contribution is adaptation for applying alignment algorithms for the estimation of the similarity between chord progressions. The representation of chord progressions are discussed in Section 2. Then the algorithm is detailed in Section 3. We propose experiments with the same database considered in a recent study presented in [6], in order to compare the two systems in Section 4. We conclude and propose future works in Section 5.

2. CHORD REPRESENTATION

We begin by reminding what are the main properties of a chord in Western tonal music. This notion can actually be ambiguous.

Firstly, a chord could be simply defined as the result of two notes or more sounding in the same time. It's a "sound production" point of view. However, if we consider jazz notation with a chord per bar for example, all that the notes of this chord are not necessarily sounding in the same time along the bar. Another example is given by an unique notation of a figured bass for a whole beat of four sixteenth notes, or for an arpeggiated chord. It's the "perception" point of view of a chord. Figure 1 illustrates these different points. To unify the different points of view, we consider here a chord as a group of notes sounding in a time window which constitutes its length. In this case, the notes composing the chord are not necessarily sounding at the same time. We

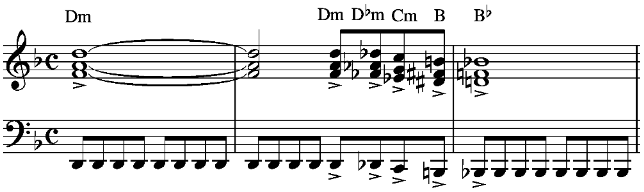

Figure 1: Example of the ambiguity of the chord notion in a polyphony excerpt of *The Phantom Of The Opera* (A.L. Webber). Are there 1 or 8 chords in the first bar ? It depends on the considered time window, and both could be answered.

introduce then the notion of note-chord which is a part of a chord, a group of notes sounding in the same time during a chord (it can be the chord itself).

Secondly, another ambiguity can appear between the instance of a chord, for example (C E Bb), and its type. The type of the chord may describe exactly the instance (in this case C7-5), or not (in this case the chord type could also be noted as C7). Moreover, a note present in an instance of a chord but analyzed as a melodic ornament could sometimes be ignored to give the type of a chord. There is then no bijection between the type of a chord and its instance, both providing different information.

This preliminary part allows us to introduce an unified definition of the main parameters of a chord. Thus, partially following the definitions of [12], a musical chord may be represented using its root (the note upon which the chord is built), its bass note (or its inversion defined by the degree of the chord played as its bass note), its type (defined by the component intervals that make up the chord relative to the root), its mode (which may be Major, minor or undefined), its instance (the list of the notes composing the chord), its duration, its degree regarding the key, and its tonal tension regarding the key.

Once these different parameters listed, a choice may be made for the representation of a sequence of chords for retrieval purpose. Regarding the method used, it is also possible to use different kind of representation for the chords in a sequence: absolute, relative to the precedent chord in the sequence or relative regarding the key. We illustrate these differences by representing the chord progression of the Figure 1. An absolute representation of this sequence could be here composed of the sequence of roots, modes and lengths in beats:

$$(D, m, 6)(D, m, \frac{1}{2})(Db, m, \frac{1}{2})(C, m, \frac{1}{2})(B, M, \frac{1}{2})(Bb, M, 4)$$

A relative representation could be achieved using the successive differences of roots in semitones and ratios of successive lengths in sixteenth notes:

$$(0, \frac{1}{12})(-1, 1)(-1, 1)(-1, 1)(-1, 4)$$

And the sequence of chords' degrees in the key (if it exists, -1 otherwise) with the Lerdahl's distance [14] of its instance regarding the triad of the key constitutes a key relative representation:

$$(1, 0)(1, 0)(-1, 11)(7, 9)(-1, 12)(6, 7)$$

3. SIMILARITY BETWEEN CHORD PROGRESSIONS

Chord progressions can be represented as a sequence of symbols. Several algorithms have been proposed to compare two sequences of notes, based on N-grams [18] or adaptations of string-matching algorithms [15]. They compute a measure which indicates the degree of similarity between a pair of sequences. Robust chord retrieval systems must take into account variations between the sequences of chords compared. This assumption leads to the consideration of approximate string matching techniques. One of these techniques is local alignment [17]. The adaptation of this algorithm for the comparison of chord sequences is presented in this section.

3.1 Alignment algorithms applied to music

Among several existing methods, Smith and Waterman's approach [17] consists in detecting local similar areas between two sequences. This *local alignment* or *local similarity* algorithm locates and extracts a pair of regions, one from each of the two given strings, that exhibit high similarity. A similarity score is calculated by considering elementary operations transforming one string into the other. The operations between sequences include deletion, insertion of a symbol, and substitution of a symbol by another. This similarity measurement requires the use of the dynamic programming principle to achieve an algorithm with quadratic complexity.

Algorithms based on local alignment have recently been successfully adapted for melodic similarity comparison purposes [15, 10]. A first approach for audio chord sequences comparison has also been presented in [4]. We propose here different possibilities for computing operation costs and we compare them in Section 4.

3.2 Adaptation of alignment algorithms

Adaptation to the specific problem of estimation of similarity between chord sequences requires the definition of the elementary operations. Costs associated to insertion and deletion are set to a same constant value. We propose some different functions for calculating the substitution score between two chords. These functions can be related to the roots, the basses, the types, the modes, . . . , of the two chords compared.

The first function is binary: if two properties of chords are identical, it returns +2, else it returns −2. The insertion/deletion score is −1. Another option is to consider the number of fifths n_f between the roots (or the basses) of the chords compared. If n_f is null, the associated score is +2. If n_f is greater than 4, the score is −2. Otherwise, it gets decreasing values (depending on n_f) between 2 and −2. In a similar way, another possible function takes into account the consonance of the interval between roots (or basses) of the chords compared. Such approach has been studied in [15, 10] for melodic similarity. The substitution score can also be based on the Lerdahl's distance [14], which gives a distance value between two instances of chords. Such functions can be extended by considering the difference of modes between the two chords compared. If the two modes are defined and are identical, the score is slightly increased. If they are defined and different, the score is slightly penalized.

Chord retrieval systems must be robust to key changes. Two identical chord progressions transposed in different keys have to be estimated as similar. The usual way to deal with such an issue [18] is to choose a chord representation which is transposition invariant. The first possibility is to represent variations between successive chords. But such relative representations have been experimented as less accurate when applied with alignment algorithms [10]. Another option is to consider the variations of chords related to the global key of the musical piece. Such representation thus relies on a prior knowledge of keys, and also prevent the correct comparison of songs with key variations. In this paper, we propose to represent chords with absolute values and to deal with transpositions by applying an adaptation of the local alignment algorithm, proposed in [1]. The new dynamic algorithm proposed allows to take into account multiple local transpositions, and can perfectly be applied to representations of chord progressions. In Section 4, we propose experiments for comparing different representations for chords. The results show that applying this algorithm lead to a chord progression retrieval system that is transposition invariant. For now, the main disadvantage is the significant computation time added by the algorithm. Optimizations are currently under development.

4. EXPERIMENTS

In this section, we propose experiments in order to test the different chord representations and the different substitution functions for the chord progression retrieval system.

4.1 Databases

We consider two collections of symbolically encoded chord progressions. These sequences have been generated from Band-in-a-Box[1] files, collected on the Internet[2]. Chord parameters are defined for each beat of songs. The first collection, denoted as *Jazz*, is the same as the one used for experiments in [6] and is composed of 388 sequences of chord labels describing 242 jazz standards found in the Real Book [3]. This collection contains 85 different songs that have two or more similar versions. The differences between these versions involve transpositions, chord deletions or insertions, differences in introduction, ending or number of repetitions, and chord substitutions. These substitutions can be enriched chords, but also relative substitutions, tritone substitutions, etc.

The second collection, denoted as *Mix*, is composed of 578 sequences of chords, representing 275 different songs, with different styles (pop, folk, jazz, latin, etc.). This *Mix* collection is larger than the collection *Jazz*, and does not contain versions that are identical. We propose to use the collection *Jazz* for experimenting the system proposed with different settings and compare the results obtained with the results of the retrieval system proposed in [6]. Then, the system proposed is tested with the *Mix* collection. The results are expected to be worse than on the *Jazz* collection. Nevertheless, they may be more informative since different styles are represented in the *Mix* collection.

In the following, all songs with multiple versions are successively considered as queries and all the other songs of the database are ranked according to their similarity score. The corresponding versions of the query are expected to be retrieved at the first ranks. Evaluations are presented according to the average first tier, the average second tier and the average TOP 10. The first tier is the number of correctly retrieved songs within the best $(C-1)$ matches divided by $(C-1)$, where C is the number of versions of the same song [6]. The second tier is the number of correctly retrieved songs within the best $(2C-1)$ matches divided by $(2C-1)$. The TOP 10 is the number of correctly retrieved songs within the best 10 matches, divided by $(C-1)$.

4.2 Chord Representations

The first experiments concern the symbolic chord representations. We consider different key-relative (KR) representations. The key of each song is assumed to be known since it is labelled in Band-in-a-Box files. Each song chord is represented as the difference in semitones (KR tones) or in number of fifths (KR Fifths) between the root of the chord and the tonic of the song key. Another representation considers the chord distance based on Tonal Pitch Space [14]. Two variations of this chord distance are tested: the original distance (KR TPS), which is asymmetric, and a symmetric distance (KR TPS08) presented in [6]. We also experiment absolute representations (ABS), and test this representation with and without the algorithmic adaptation that allows transpositions (ABS Transpo), presented in the previous section. The results are presented in Table 1.

Representation	1st Tier	2nd Tier	TOP 10
KR Tones	0.712	0.756	0.793
KR Fifths	0.722	0.766	0.796
KR TPS	0.512	0.552	0.558
KR TPS08	0.531	0.578	0.650
ABS	0.615	0.660	0.707
ABS Transpo	**0.840**	**0.884**	**0.892**

Table 1: Results (grand averages of average first, second tiers and top 10) of experiments with the *Jazz* collection, considering different chord representations.

	1st Tier	2nd Tier
Avg	0.74	0.77

Table 2: Results presented in [6] for chord retrieval, considering the *Jazz* collection.

Key-relative representations allow the retrieval system to be transposition invariant under the condition that the song key is known and correct. The transposition invariance partially explains why the best results are obtained with key-relative representations instead of the absolute representation (0.712 or 0.722 for the first tier instead of 0.615 for absolute representation). But these results also show that considering transposed sequences significantly increase the accuracy of the retrieval system with the absolute representation (0.840). This score is higher than all the key-relative scores. This result may be explained by errors concerning key annotations in the Band-in-a-Box files, or variations of keys during a song. These two limitations lead us to choose absolute representation for chords and to consider the algorithmic adaptation proposed for transposition invariance.

It is important to note the low scores obtained with key-relative representations based on Tonal Pitch Space chord distance, whereas results presented in [6] with a chord retrieval system that considers this chord distance are far better (see Table 2). These differences may be justified by the different comparison algorithms. The algorithm applied in [6] is probably more adapted to such tonal distance. Furthermore, such representation, based on Tonal Pitch Space, is dedicated to musical analysis since it has been defined to highlight tension and release patterns.

4.3 Substitution Score

Methods for calculating the substitution scores between two chords are experimented. Table 3 shows the different results obtained by considering different functions.

We experiment a few different methods that consider essentially root, since the first experiments show that comparing chord type is too accurate for retrieval purposes and leads to poor results. This observation is illustrated by the low results obtained with a substitution function based on Tonal Pitch Space (*TPS*). We first test substitution functions based on binary comparisons (*Binary*). If the roots (*Root*) or basses (*Bass*) of the two chords compared are

[1] http://www.pgmusic.com/products_bb.htm
[2] http://www.biabgroup.com

Substitution Score	1st Tier	2nd Tier	TOP 10
Root Binary	0.840	0.884	0.892
Bass Binary	0.808	0.871	0.894
Bass Fifths	0.730	0.793	0.831
Root Fifths	0.851	0.888	0.897
Bass/Root Fifths	0.850	**0.895**	0.903
TPS	0.272	0.462	0.524
Consonance	0.870	0.894	**0.904**
Consonance/Mode	**0.873**	0.894	**0.904**

Table 3: Results of experiments with the *Jazz* collection, considering different scores for chord substitutions.

	1st Tier	2nd Tier
Avg	0.685	0.732

Table 4: Results of experiments with the *Mix* collection.

identical, substitution score is set to $+2$, whereas it is -1 if they are different. The retrieval results obtained with these binary score functions are better than the results obtained in [6]: 0.84 instead of 0.74 for the first tier.

We also test more complex substitution functions based on musical properties. We propose to extend these functions by considering the number of fifths between notes (*Root Fifths*, *Bass Fifths*), or by introducing a score depending on the consonance of the interval between roots [15, 10] (*Consonance*). We also try to consider information about chord type such as mode (*Consonance/Mode*). Improvements induced by these functions can be observed, even if they are small. The best results are obtained with a substitution function which takes into account consonance and mode (0.873 for the first tier).

In Table 4, we present results of experiments performed with *Mix* database considering absolute representation, algorithm for transposition invariance and substitution score based on consonance and mode. Results are fine (0.685 for the first tier) but lower than those obtained with the *Jazz* database. This difference can be partially justified by the lack of identical version in the *Mix* database. For 159 songs over 275, other versions of the query song are retrieved at the top ranks.

5. CONCLUSIONS

The alignment based chord retrieval system described in this paper has been experimented as very accurate for databases with different styles. Comparisons with results presented in previous work [6] seem to indicate that applying local alignment improves the quality of such systems. Justifications of these differences are certainly due to the choice of substitution score, but also the possible insertion or deletion of chords within the two chord sequences compared.

One other major conclusion of this study is the comparison of different substitution score functions applied. A simple substitution function, only based on binary root comparison, also leads to good results. This implies that the different song versions within the databases considered certainly involve enriched chords. Such cases lead to errors when dealing with chord distance such as Tonal Pitch Space, since these distances take chord types into account.

In the future, it may be interesting to investigate music retrieval (symbolic or audio) based on chord comparisons, in order to estimate how discriminating such a comparison may be. Such a study may lead to a robust system for polyphonic music retrieval.

6. REFERENCES

[1] J. Allali, P. Hanna, P. Ferraro, and C. Iliopoulos. Local Transpositions in Alignment of Polyphonic Musical Sequences. In *Proceedings of the String Processing and Information Retrieval Symposium (SPIRE)*, pages 26–38, 2007.

[2] J. J. Aucouturier and F. Pachet. Improving Timbre Similarity: How High is the Sky? *Journal of Negative Results in Speech and Audio Sciences*, 1(1), 2004.

[3] V. Authors. *The Real Book*. Hal Leonard Corporation, 6th edition., 2004.

[4] J. Bello. Audio-based Cover Song Retrieval using Approximate Chord Sequences: Testing Shifts, Gaps, Swaps and Beats. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 239–244, Vienna, Austria, 2007.

[5] M. Casey and M. Slaney. The Importance of Sequences in Musical Similarity. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 5–8, Toulouse, 2006.

[6] B. De Haas, R. Veltkamp, and F. Wiering. Tonal Pitch Step Distance: A Similarity Measure for Chord Progressions. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 51–56, 2008.

[7] J. S. Downie. Music Information Retrieval. *Annual Review of Information Science and Technology*, 37:295–340, 2003.

[8] E. Gómez. *Tonal Description of Music Audio Signals*. PhD thesis, Univ. Pompeu Fabra, Barcelona, 2006.

[9] D. Gusfield. *Algorithms on Strings, Trees and Sequences - Computer Science and Computational Biology*. Cambridge University Press, 1997.

[10] P. Hanna, P. Ferraro, and M. Robine. On Optimizing the Editing Algorithms for Evaluating Similarity Between Monophonic Musical Sequences. *Journal of New Music Research*, 36(4):267–279, 2007.

[11] P. Hanna, M. Robine, P. Ferraro, and J. Allali. Improvements of Alignment Algorithms for Polyphonic Music Retrieval. In *Proceedings of the Int. Computer Music Modeling and Retrieval Conference (CMMR)*, pages 244–251, Copenhagen, Denmark, 2008.

[12] C. Harte, M. Sandler, S. Abdallah, and E. Gómez. Symbolic Representation of Musical Chords: A Proposed Syntax for Text Annotations. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 66–71, London, UK, 2005.

[13] K. Lemström and A. Pienimäki. Approaches for Content-Based Retrieval of Symbolically Encoded Polyphonic Music. In *Proceedings of the International Conference on Music Perception and Cognition (ICMPC)*, pages 1028–1035, Bologna, Italy, 2006.

[14] F. Lerdahl. *Tonal Pitch Space*. Oxford University Press, 2001.

[15] M. Mongeau and D. Sankoff. Comparison of Musical Sequences. *Computers and the Humanities*, 24(3):161–175, 1990.

[16] B. Pardo and M. Sanghi. Polyphonic Musical Sequence Alignment for Database Search. In *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR)*, pages 215–222, London, UK, 2005.

[17] T. Smith and M. Waterman. Identification of Common Molecular Subsequences. *Journal of Molecular Biology*, 147:195–197, 1981.

[18] A. L. Uitdenbogerd. *Music Information Retrieval Technology*. PhD thesis, RMIT University, Melbourne, Australia, July 2002.