Motivic Pattern Extraction in Music, And Application to the Study of Tunisian Modal Music

O Lartillot*, M Ayari†

*Department of Music, PL 35(A), 40014 University of Jyväskylä, Finland
†Ircam, Place Igor-Stravinsky, 75004 Paris, France

ABSTRACT

A new methodology for automated extraction of repeated patterns in time-series data is presented, aimed in particular at the analysis of musical sequences. The basic principles consists in a search for closed patterns in a multi-dimensional parametric space. It is shown that this basic mechanism needs to be articulated with a periodic pattern discovery system, implying therefore a strict chronological scanning of the time-series data. Thanks to this modelling global pattern filtering may be avoided and rich and highly pertinent results can be obtained. The modelling has been integrated in a collaborative project between ethnomusicology, cognitive sciences and computer science, aimed at the study of Tunisian Modal Music.

KEYWORDS: pattern extraction, time-series data, closed pattern, periodic pattern, music analysis, tunisian modal music

1 INTRODUCTION

This paper introduces a new methodology for repeated pattern (or motif) extraction in symbolic sequences, and is applied particularly to the analysis of musical scores. Among the different approaches that can be considered for time-series data analysis, one domain of research that has received much attention is the problem of extraction of motives, i.e. the discovery of patterns appearing frequently in time-series data [1, 2, 3, 4]. Indeed, motives may characterize important aspects of the data, and help discovering new association rules. In music too, repeated sequences of notes are easily perceived by listeners as important structures, forming the words of the musical structure.

Lots of research have been carried out in this domain and numerous interesting solutions have been proposed. One major problem stems from the structural redundancies logically resulting from this task, which, if not carefully controlled, may provoke combinatorial explosion and infringe the quality of the results. Few researches have considered the pattern discovery problem within this general context. The approach presented in this paper follows this idea of closed pattern, which is defined here in a multi-dimensional parametric space. Another combinatorial redundancy problem, provoked by immediate succession of same patterns, is solved by introducing the concept of cyclic pattern. The model has been applied to the automated motivic analysis of musical scores, and in particular to the study of Arabic improvisations played by Tunisian masters.

Email: O Lartillot `lartillo@campus.jyu.fi`, M Ayari `Mondher.Ayari@ircam.fr`

Most music databases contain sound files of performance recordings, which correspond to the way music is commonly experienced. The underlying structure of music, on the other hand, is represented in a symbolic form – the score – that describes musical pieces regardless of the way they are performed. There exist numerous digital formats of symbolic music representation (MIDI, MusicXML, Humdrum, etc.). The pattern discovery system described in this paper is applied uniquely to symbolic representation. A direct analysis on the signal level would arouse tremendous difficulties. A pattern extraction task on the symbolic level, although theoretically simpler, remains extremely difficult to carry out, and its automation has not been achieved up to now. Indeed, computer researches on this subject hardly offer results close to listeners' or musicologists' expectations. Hence the pattern discovery task is too complex to be undertaken directly at the audio signal, and needs rather a prior transcription from the audio to the symbolic representations, in order to carry out the analysis on a conceptual level.

2 AN INCREMENTAL MULTIDIMENSIONAL MOTIVIC IDENTIFICATION

2.1 Definitions

Music is expressed along multiple parametric dimensions. This paper will focus on two main dimensions (Figure 1):

- Melodic dimension (*melo*) defined by pitch differences between successive notes. (In scores, pitches are represented by the vertical position of the notes.)

- Rhythmic dimension ($rhyt$) defined by durations between successive notes, and expressed with respect to metrical unit. For instance, in a 6/8 metric (whose metrical unit is the 8th note) a dotted 8th note correspond to the value 1.5.

Figure 1 shows a musical example in traditional musical notation. Below the score is indicated the corresponding numerical description of the musical example along the melodic and rhythmic dimensions. For instance, the first value of the melodic dimension ($melo = +1$) indicates that the second note of the score is located one step higher than the first note of the score on the vertical dimension. In musical term, this means that the second pitch is one step higher than the first pitch. The first value of the rhythmic dimension ($rhyt = 1.5$) indicates that the duration of the first note (or, equivalently, the temporal distance between the first and second note) is equal to 1.5 times the basic unit.
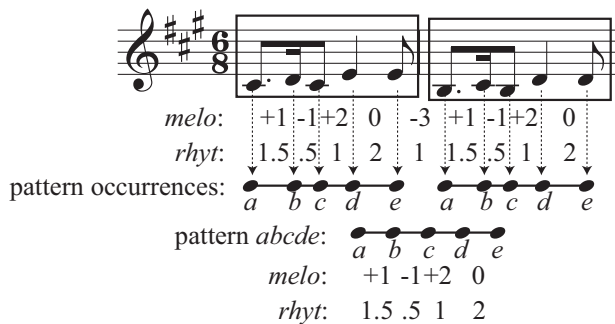

Figure 1: Multi-dimensional description of a musical sequence.

A repeated succession of descriptions forms a pattern, and the occurrences of the pattern correspond to its repetitions in the score. Figure 1 shows two repetitions of a pattern, which are squared in the score. Each pattern occurrence is represented as a chain of states, each successive state (a, b, c, d, e in figure 1) corresponding to each successive note of the occurrence. The pattern itself – i.e., the abstract entity that unifies all these occurrences – is represented at the lower part of the figure by the same chain of state. The transitions between successive states, indicated below the chain of states, describe the successive intervals between successive notes of the pattern.

2.2 Identification of similarities

Patterns are generally not exactly repeated but transformed in multiple ways. These patterns should therefore be detected through an identification of their different occurrences beyond their apparent diversities. Current approaches follow two different strategies. One is based on numerical similarity, and tolerates a certain amount of dissimilarity between compared parameters [5, 6]. The main drawback of this strategy arises from the impossibility of fixing precisely similarity thresholds, on which identification decision are based, and hence insuring relevant analyses.

Reference cognitive studies [7], on the other hand, assert that similarity does not come from numerical distance minimization, and propose instead an alternative strategy based on exact identification along multiple musical dimensions of various specificity levels. Several approaches to pattern discovery follow this second strategy of identification along different musical dimensions [8, 9] and search for repetitions along each different dimension and product of dimensions.

Nonetheless there exist patterns that are progressively constructed along variable successive musical dimensions. These *heterogeneous* patterns cannot be identified by traditional approaches. For instance, each line of the score in figure 2 contains a repetition of a same pattern: in the first half, both melodic and rhythmic dimensions are repeated whereas, in the second half, only the rhythmic dimension is repeated. The model presented in this paper is able to discover heterogeneous patterns.

3 COMBINATORIAL REDUNDANCY FILTERING

This section presents the basic problem of pattern discovery and introduces the notion of closed pattern.

3.1 Formalisation

Let $S = < a_1 a_2 \ldots a_N >$ be a sequence of elements of some set $a_i \in A$. A *subsequence* $S_{i,l}$ of index $i \in [1, N]$ and of length $l \in [1, N + 1 - i]$ is a sequence of the form:

$$S_{i,l} = < a_i a_{i+1} \ldots a_{i+l-1} > . \qquad (1)$$

A sub-sequence $S_{i,k}$ is *included* in another subsequence $S_{j,l}$, noted $S_{i,k} \subset S_{j,l}$ when $j \leqslant i$ and $i + k \leqslant j + l$. A *pattern* of *length* l, denoted $P \in \mathcal{P}(S)$, is defined as a repeated sub-sequence:

$$P \in \mathcal{P}(S) \iff \exists (i, j) \in [1, N]^2, P = S_{i,l} = S_{j,l}. \quad (2)$$

The *support* of a pattern P, denoted $\sigma(P)$, is the number of occurrences of the pattern, i.e.

$$\sigma(P) = \left| \left\{ i \in [1, N], S_{i,l} = P \right\} \right|. \qquad (3)$$

3.2 Maximal patterns and closed patterns

The task of discovering repeated patterns leads to combinatorial problems. Indeed each pattern of length l contains $\sum_{i=1}^{l} i = \frac{l(l-1)}{2} = O(l^2)$ sub-patterns. that would be considered as distinct patterns by any brute force algorithm. The problem can be avoided by restricting the search to the patterns of high support and/or to patterns of pre-specified length [4, 8, 9, 10, 11]. These constraints, in return, significantly reduces the richness of the analysis.

One common way to solve this problem consists in focusing on the *maximal patterns* P of the sequence S, denoted $P \in \mathcal{M}(S)$, which are patterns of S not

Figure 2: Repetition of a heterogeneous pattern.

included in any other pattern of S [12, 13]:

$$P \in \mathcal{M}(S) \iff \begin{cases} P \in \mathcal{P}(S) \\ \nexists Q \in \mathcal{P}(S), P \subset Q. \end{cases} \qquad (4)$$

For instance, pattern aij (which can simply be denoted by its last state j) in figure 3 is a simple suffix of pattern $abcde$ (or e). It does not need to be explicitly represented, since the set of its occurrences (or pattern class) can be directly deduced from the class of its superpattern e.

This heuristic enables a significant reduction of the number of discovered pattern, but leads also to a loss of information. Indeed, not all the sub-patterns may be immediately reconstructed knowing the maximal patterns. In figure 4, for instance, the pattern class of j cannot be directly deduced from the pattern class of e, and should therefore be explicitly represented in the final analysis. This corresponds to the concept of *closed pattern* [13]. A pattern P will be called *closed*, denoted $P \in \mathcal{C}(S)$, if and only if there exists no proper super-pattern Q of same support :

$$P \in \mathcal{C}(S) \iff \begin{cases} P \in \mathcal{P}(S) \\ \nexists Q \in \mathcal{P}(S), \begin{cases} P \subset Q \\ \sigma(P) = \sigma(Q). \end{cases} \end{cases} \qquad (5)$$

The concept of closed pattern offers a compact and lossless description of the musical piece: it allows an exhaustive description of the pattern configurations and avoids any global selection based, for instance, on minimum support threshold, as in traditional approach for mining association rules [1, 2]. Even pattern with only two occurrences are included in the description. Some constraints, however, have been added to the framework, that reduce the solution space following cognitive heuristics related to music perception. For instance, a constraint takes into account the limitations of short-term memory: namely, for a pattern to be detected, the temporal distance between at least two of its occurrences should not exceed a given threshold.

4 MULTI-DIMENSIONAL CLOSED PATTERNS

The model presented in this paper looks for closed pattern in musical sequences. For this purpose, the notion of inclusion relation between patterns founding the definition of closed patterns needs to be generalized to the multi-dimensional parametric space of music, defined in section 2.1.

4.1 Formalisation of the problem

Musical patterns are represented with the help of a conceptual framework that defines *objects* associated with different kinds of *attributes* [14]. These attributes consist not only of the different musical dimensions, but also of the different sub-patterns and super-patterns. The objects of the pattern descriptions are the successive notes of the musical sequence forming the set $\mathcal{N}(S)$. Each note $n_i \in \mathcal{N}(S)$ relates to a specific *temporal context*, defined by *the part of the musical sequence concluded by this note* n_i, that is to say, the subsequence $< n_1 \ldots n_i >$.

Each note n_i is described firstly by the different musical characteristics of the preceding interval: $\overrightarrow{n_{i-1}n_i}$.

$$\mathcal{D}_{\mathsf{desc}}^{0,p}(n_i) \quad : \quad \mathsf{desc}(\overrightarrow{n_{i-1}n_i}) = p, \\ \mathsf{desc} \in \{\mathsf{melo}, \mathsf{rhyt}\}, p \in \mathsf{desc}. \qquad (6)$$

Each note n_i is also described by the musical characteristics of the older intervals:

$$\mathcal{D}_{\mathsf{desc}}^{j,p}(n_i) \quad : \quad \mathsf{desc}(\overrightarrow{n_{i-j-1}n_{i-j}}) = p, \\ \mathsf{desc} \in \{\mathsf{melo}, \mathsf{rhyt}\}, p \in \mathsf{desc}. \qquad (7)$$

Then the pattern description of the sequence S may be expressed as a *formal context* $(\mathcal{N}(S), \mathcal{D}, I)$ [14] where :

- the set of *objects* is $\mathcal{N}(S)$: the set of notes in S,
- the set of *attributes* is $\mathcal{D}$: the set of elementary musical descriptions defined by equations 6 and 7,
- and I is the binary relation between $\mathcal{N}(S)$ and $\mathcal{D}$, called *incidence*, defined by:

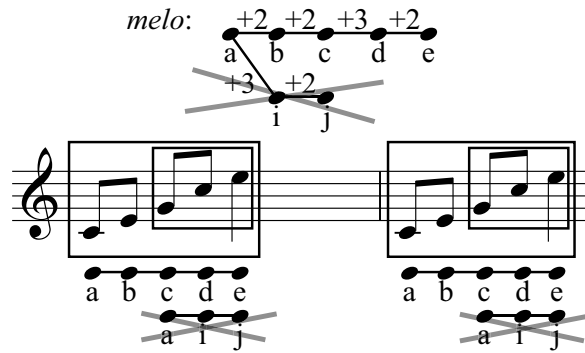$$(n_i, \delta) \in I \iff \delta(n_i) \text{ is true.} \qquad (8)$$

Figure 3: On the score are squared the occurrences of two patterns *abcde* and *aij*. The occurrences of these patterns are represented below the score. Pattern descriptions are represented over the score, using a tree representation justified in section 6.1. Pattern *aij*, suffix of *abcde* with same support (2 occurrences), is a non-closed pattern that does not need to be explicitly represented.
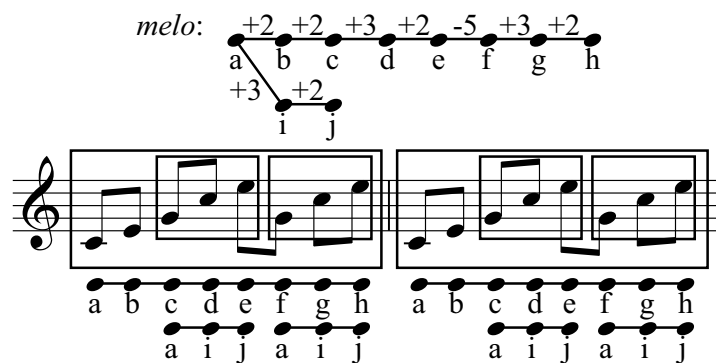

Figure 4: Same convention than in figure 3. Pattern *aij*, whose support (4 occurrences) is now greater than the support of *abcde* or *abcdefgh* (2 occurrences each) is a closed patterns and needs to be explicitly represented.

The *derived description* C' of a set of notes $C \subset \mathcal{N}(S)$ is defined as the common description of all these notes:

$$C' = \{\delta \in \mathcal{D} \mid \forall n \in A, (n, \delta) \in I\}. \quad (9)$$

The notes in C are therefore occurrences of a same pattern, which is maximally described by C'.

The *derived class* D' of a complex description $D \subset \mathcal{D}$ is dually defined as the set of notes complying with this description:

$$D' = \{n \in \mathcal{N}(S) \mid \forall \delta \in D, (n, \delta) \in I\}. \quad (10)$$

The pattern discovery task consists in finding exhaustive class D' sharing a same description D. The trouble is, lots of different descriptions D_i may lead to same classes D_i'.

4.2 A representation of patterns as formal concepts

The *derivators operations* defined by equation 9 and 10 establish a Gallois connection between the power set lattices on $\mathcal{N}(S)$ and $\mathcal{D}$ [14]. The Gallois connection leads to a dual isomorphism between two closure systems, whose elements, called *formal concepts* of the formal context $(\mathcal{S}(S), \mathcal{D}, I)$ corresponds exactly to the *close patterns* $P = (C, D)$, verifying:

$$C \subset \mathcal{N}(S), D \subset \mathcal{D}, C' = D, \text{ and } D' = C. \quad (11)$$

For a close pattern $P = (C, D)$, C is called the *extent* of D and D the *intent* of C. We may simply call C and D respectively the *class* and the *description* of P.

Hence, for a set of patterns $P_i = (D_i', D_i)$ of same class $D_i' = C$, the close pattern $P = (C, D)$ is described using the derived operator C' defined in equation 9: it contains all the elementary descriptions common to all notes of the class C. In other words, closed patterns are described as precisely as possible.

Closed patterns, or formal concepts, are naturally ordered by the *subconcept-superconcept* relation defined by

$$(C_1, D_1) < (C_2, D_2) \iff C_1 \subset C_2 \ (\iff D_2 \subset D_1). \quad (12)$$

This subconcept-superconcept relation can also be called *specificity relation*. In the equation above, for instance (C_1, D_1) is *more specific* than (C_2, D_2) and vice versa.

In particular, a pattern is more specific than its suffixes, since the description of the suffixes are strictly included in the description of the pattern.

4.3 Illustration

For instance, pattern *abcde* (in figure 5) features melodic and rhythmic descriptions, whereas pattern *afghi* only features its rhythmic part. Hence pattern *abcde* can be considered as more specific than

pattern $afghi$, since its description contains more information. The less specific pattern $afghi$ is a closed pattern, since its third occurrence is not an occurrence of the more specific pattern $abcde$, and is therefore not filtered out.

5   CYCLIC PATTERNS

In this section, we present another important factor of redundancy that, contrary to closed patterns, has not been studied in current general algorithmic researches.

5.1   Periodic sequences

Periodicity occurs when a given pattern is successively repeated, such as the repetition of the pattern $< 1, 2 >$ in figure 6. Periodicity leads to combinatory explosion, since all possible periods (i.e. all the possible rotations of one period, such as abc and pqr in the example) can be considered as patterns, as well as all possible concatenations of periods and their different prefixes, such as patterns $abcd$, $pqrs$, $abcde$, etc. on the left panel of the figure. These redundant structural artefacts should be replaced by a compact representation that would explicitly describe the structural properties of such configuration. For this purpose, we propose to model periodic sequence through cyclic graphs. A cyclic pattern chain (CPC) is constructed from an originally acyclic pattern chain (APC) representing one period of the cycle, where a transition is added from the last state to the first state. In this way, the whole local periodicity can be represented by a single pattern occurrence chain where each successive state is uniquely linked to the successive phases of the CPC. An example of CPC is given on the right panel of figure 6.

Each successive state of a pattern chain is related to each successive prefix of the pattern occurrence. For this reason, concerning the pattern occurrence chain representing the entire periodicity, the first states that represent the first period should not be associated with the CPC since they are already associated with the APC of that period. On the contrary, the states of the pattern occurrence chain following the first period will be associated to the CPC, since they represent a configuration that is actually specific to the periodicity. In this way, the CPC may be considered as a child of the APC, as can be seen in the figure.

This additional concept immediately solves the redundancy problem. Indeed, each type of redundant structure considered previously are *non-closed suffix* of prefix of the long pattern chain, and will therefore not be represented any more. For instance, in figure 6, pattern $abcd$ is a suffix of pattern b' with same support. Pattern $abcd$ is therefore non-closed and can be discarded. Idem for patterns $abcde$, $abcdef$, etc. Pattern pq, suffix of abc with same support, is also non-closed. Pattern pqr, suffix of b' with same support, is non-closed too. Idem for patterns $pqrs$, $pqrst$, etc. Hence only one pattern occurrence remains, that

cover the whole periodic sequence, as displayed on the right panel of the figure.

But this compact representation will be possible only if the initial period (corresponding to the APC) is considered and extended before the other possible periods. That is to say, in figure 6, the APC abc should be considered before pqr. This shows therefore that *the sequence needs to be scanned in a chronological way*. This justifies therefore the incremental approach followed by the algorithmic realisation of the modeling, presented in section 6.1.

5.2   Related works

Researches are dedicated to the automated discovery of periodic patterns in time-series data [15, 16, 17, 18]. But as the search is focused on periodic patterns only, no interaction is proposed with acyclic pattern discovery. Hence, although offering interesting descriptions of time series data, they cannot be used in order to solve the combinatory problem presented in the previous paragraph. In our approach, on the other hand, the periodic pattern problem is deeply articulated with the acyclic pattern discovery process, insuring the compactness of the results.

A simpler solution to the combinatory problem consists in forbidding overlapping between patterns [3]. But this heuristics presupposes that time-series data are segmented into one-dimensional series of successive segments. Time-series data do not all fulfil this requirement: musical sequences, in particular, may sometimes be composed of multi-levelled hierarchy of structures. Another solution is to control the combinatorial explosion by selecting, once the analyses completed, patterns featuring minimal temporal overlapping between occurrences [8]. But as the selection is inferred globally, relevant patterns may be discarded. Besides combinatorial redundancy remains problematic since the filtering is carried out after the actual analysis phase. Our focus on local configurations enables a more precise filtering.

5.3   The figure/ground rule

Another kind of redundancy appears when occurrences of a pattern – such as pattern $acd =< 1A, 2A >$ in figure 7 – are superposed to a cyclic pattern (b'), such that the pattern acd is more specific than the cycle period (b' simply representing the successive repetition of As). In this case, the intervals that follow these occurrences are identical, since they are related to the same state (b') of the cyclic pattern. Logically the pattern could be extended following the successive extensions of the cyclic patterns (leading to pattern e, and so on). This phenomenon, which frequently appears, leads to another combinatorial proliferation of redundant structures if not correctly controlled by relevant mechanisms. On the contrary, following the *Gestalt* Figure/Ground rule, the pattern acd can be considered as a specific figure that emerges above the periodic background. Following the *Gestalt* rule, the figure cannot be extended (into d) by a description
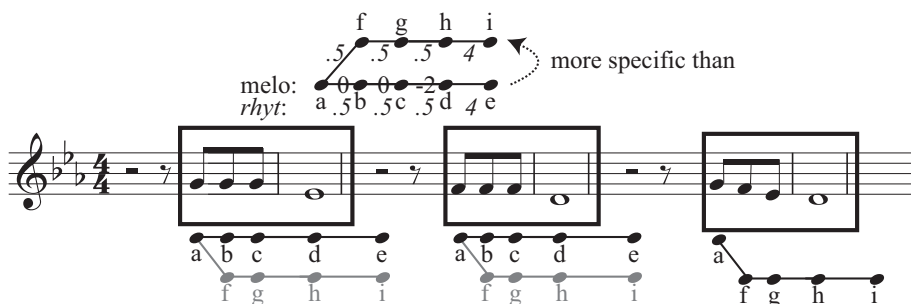
Figure 5: The rhythmic pattern $afghi$ is less specific than the melodico-rhythmic pattern $abcde$.
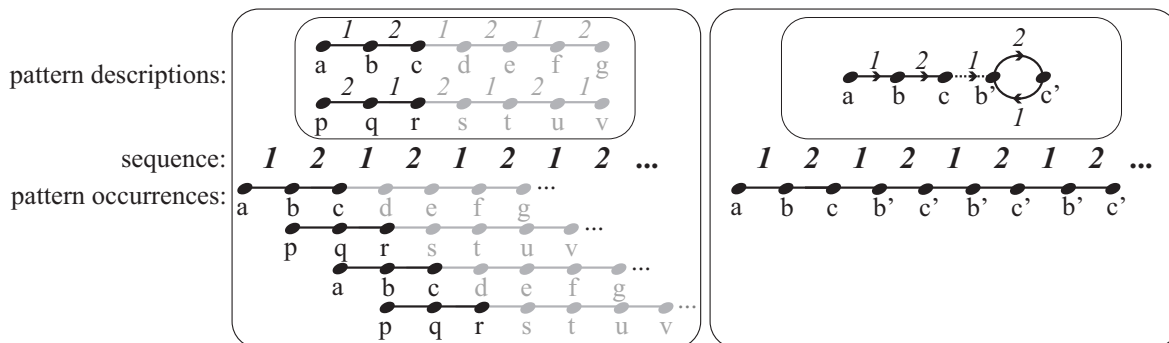

Figure 6: Multiple successive repetitions of pattern $abc =< 1, 2 >$ induce a complex intertwining of non-perceived structures (represented in grey, left panel) that can be filtered out using cyclic patterns (right panel).

that can be simply identified with the background extension. This rules shows the interest of integrating cognitive rules into the model, as these rules concern as much the perceptive adequacy of the results than the computational efficiency of the process.

6 IMPLEMENTATION DETAILS

6.1 Incremental pattern construction

This paragraph describes the basic mechanism of pattern extraction. As explained in section 5.1, it consists in an online process that progressively reads the musical sequence in a chronological order and that extracts a list of patterns. This list of patterns can be represented as a prefix tree, since two motives with same prefix can be considered as two different continuations of this prefix.

The basic principle of the algorithm refers to associative memory, i.e. the capacity of relating items that feature similar properties. The associative memory is modeled through inverted lists related to the different musical parameters (i.e. melodic and rhythmic dimensions). A first set of tables store the intervals of the piece with respect to their values along each different musical dimension. For instance, two tables (Figure 3, line a) store the intervals of the score according to their melodic and rhythmic values. The melodic table shows that the first interval of each bar shares same melodic value $melo = +1$, and, the rhythmic table indicates another identity $rhyt = 1.5$.

Intervals sharing a same value form occurrences of an elementary pattern that simply represents this

particular interval parameter. The elementary pattern is represented as a child (here b) of the root of the pattern tree (a). Each time a new pattern is created, new tables (at the right of node b) store all the possible intervals that immediately follow the occurrences of the new pattern (b). When any identity is detected in these new tables, a new pattern is created as an extension of the previous one (c, as an extension of b), and is represented as a child in the pattern tree, and so on. This algorithm enables a progressive discovery of the successive extensions of each pattern, either homogeneous or heterogeneous, as defined in section 2.2: the selection of musical dimensions defining each successive extension of a pattern may vary. For instance, in Figure 8, the last extension of pattern $abcde$ is simply melodic since the rhythm of the last interval in each occurrence is different. Besides, additional constraints have been integrated in order to insure a minimal continuity along these variable successive musical dimensions.

6.2 Avoiding redundant description of pattern occurrences

We have prolonged the attempt to optimize pattern descriptions by adding a principle of maximally specific descriptions of pattern occurrences: when a pattern occurrence is discovered (pattern e in Figure 5), all the occurrences of less specific patterns (pattern i) are not superposed on it, since they do not bring additional information, and can be directly deduced from the most specific pattern occurrence (e) and from the specificity relation (between e and i).
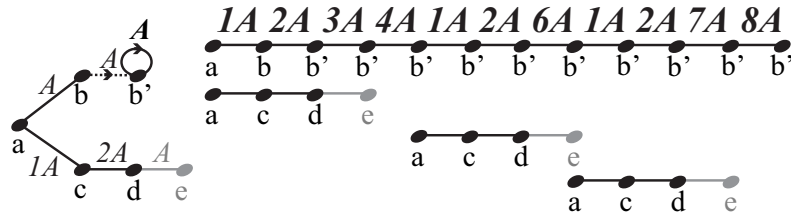
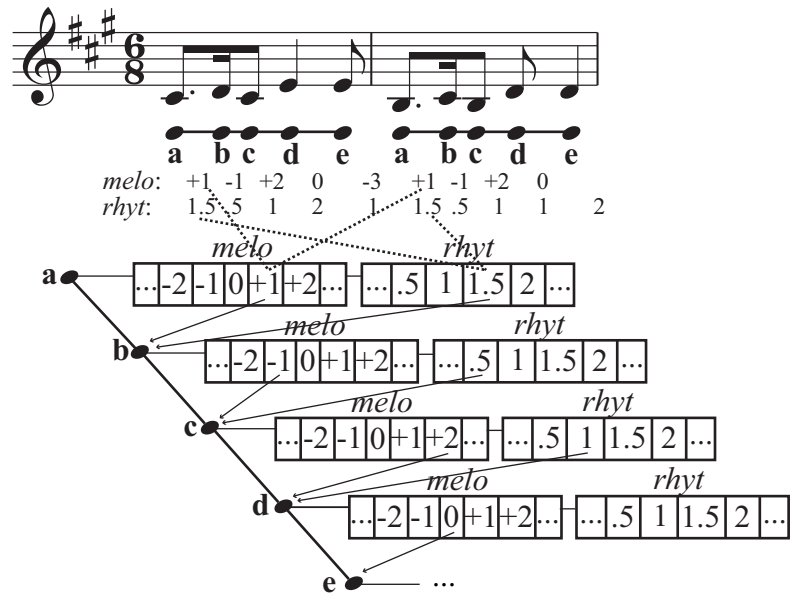*Figure 7:* Pattern c is a specific figure, above a background generated by the cyclic pattern b'.

Figure 8: Progressive construction of pattern abcde.

The less specific description should be taken into account implicitly though, because their extensions may sometimes lead to specific descriptions. For instance (Figure 9), groups 1 and 3 are occurrences of pattern h, and groups 3 and 4 are occurrences of pattern d. Since pattern d is more specific, the less specific pattern h does not need to be associated with group 4. However in order to detect groups 2 and 5 as occurrences of pattern l, it is necessary to implicitly consider group 4 as an occurrence of pattern h. Hence, even if pattern h, since less specific than d, was not explicitly associated with group 4, it had to be considered implicitly in order to construct pattern l. Implicit information is reconstituted through a traversal of the pattern network along specificity relations.

6.3   General and specific cycles

The integration of the concept of cyclic pattern in the multidimensional musical space requires a generalisation of specificity relations, defined in previous section, to cyclic patterns. A cyclic pattern C is considered as more specific than another cyclic pattern D when the sequence of description of pattern D is included in the sequence of description of pattern C. For instance, figure 10 displays four different cycles, the less specific cycle $d'' \circlearrowright f$ describes the alternation of 1 and 2, the most specific cycle $b'' \circlearrowright g'$ describes the alternation of 1A and 2B, and the two other cycles $b' \circlearrowright c'$ and $d' \circlearrowright e'$

are in-between in the specificy graph. All these four cycles forms therefore an oriented graph called *specific graph* (SG) whose root is the less specific cycle $d'' \circlearrowright f$.

7   GENERAL RESULTS

This model was first developed as a library of *Open-Music* [19], called *OMkanthus*. A new version will be included in the next version 2.0 of *MIDItoolbox* [20], a *Matlab* toolbox dedicated to music analysis. The model can analyse monodic musical pieces (i.e., constituted by a series of non-superposed notes) and highlight the discovered patterns on a score.

7.1   Experiments

The model has been tested with different musical sequences taken from several musical genres (classical music, pop, jazz, etc.). Table 1 shows some results. The experiment has been undertaken with version 0.6.8 of *OMkanthus* on a 1-GHz PowerMac G4. A musicologist expert has validated the analyses. The proportion of patterns considered as relevant is displayed in the table.

The analysis of a medieval song called *Geisslerlied* – sometimes used as a reference test for formalised motivic analysis – gave quite relevant results. The analysis has been actually carried out on a slight simplification of the actual piece presented in [21], exclud-
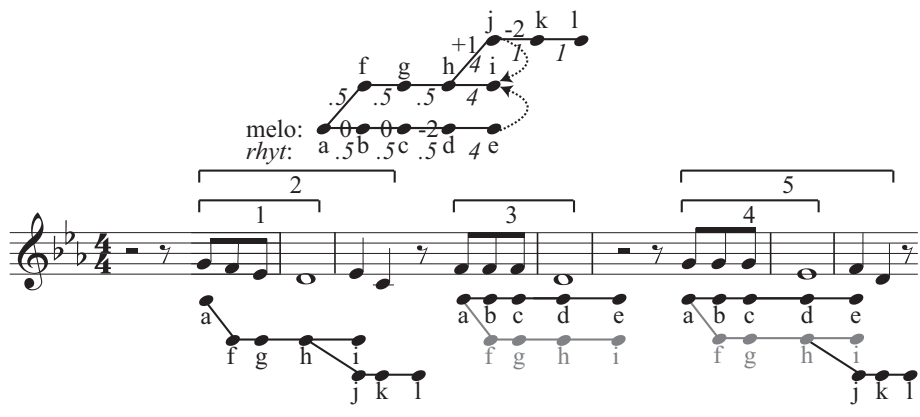
Figure 9: Group 4 can be simply considered as occurrence of pattern *d*. However, in order to detect group 5 as occurrence of pattern *l*, it is necessary to implicitly infer group 4 as occurrence of pattern *h* too.
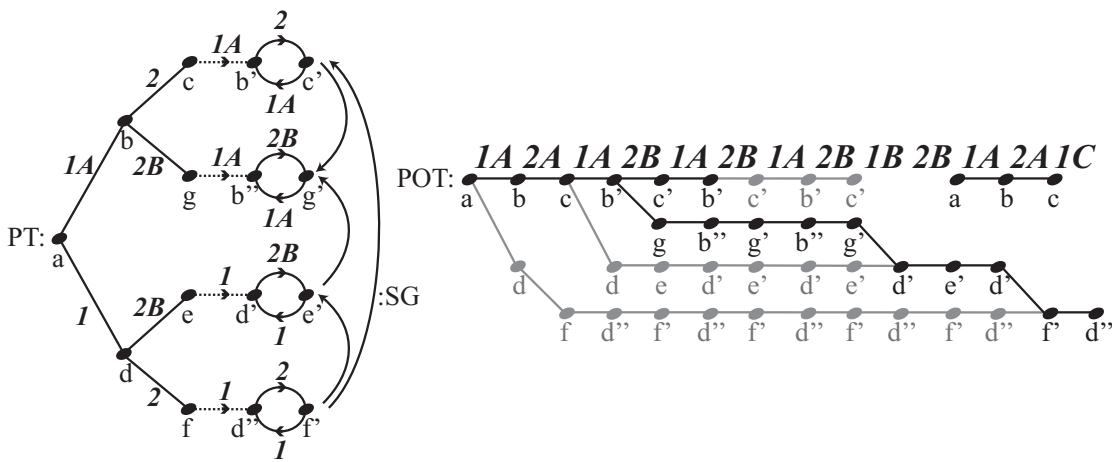

Figure 10: More detailed analysis of the perceived cyclic configurations.

Table 1: Results of analyses, either melodic (M) or melodico-rythmic (M+R), performed by *OMkanthus* 0.6.8.

Musical sequence		Anal.	Pattern classes			Comp.
Name	Notes	type	Disc.	Relv.	Succ.	time
Geisslerlied medieval song	108	M	6	5	83%	2.2 sec.
Au clair de la lune folk song	44	M+R	21	5	24%	5.6 sec.
Bach, *Invention in D minor* BWV 775	283	M	49	34	69%	37.6 sec.
Mozart, *Sonata in A* K331 1st theme, 1st half, melody	36	M+R	14	10	71%	0.8 sec.
The Beatles *Obla Di Obla Da*	390	M	14	10	71%	28.1 sec.

ing local motivic variations out of reach of the current modelling.

The melodico-rhythmic analysis of the French song *Au clair de la lune* posed problems: 21 patterns were discovered from a 44-note long sequence. This is due to the fact that the successive steps of progressive generalisation or specification of cycles are currently modelled using distinct intermediary cyclic patterns. The inference of these redundant cyclic patterns will be avoided in further works.

The algorithm has been successfully applied on a melodic analysis of a complete two-voice *Invention* by J.S. Bach. Figure 11 shows the analysis of the 21 first bars. The cyclic patterns are represented by graduated lines, the graduation representing each return of one possible phase. Due to the nature of the cyclic patterns, no preference is given by the model between different possible phases of the same cycle. The rhythmic analysis of the piece, on the contrary, failed, due to the alternation of sequences of either quarter notes or 8th notes, which will require a formalisation through hierarchical pattern chains (where successive states of higher-level patterns are linked to distinct lower-level patterns).

Figure 11: Automated motivic analysis of J.S. Bach's *Invention in D minor* BWV 775, 21 first bars. The occurrences of each pattern class are designated in a distinct way.

The analysis of The Beatles' *Obla Di Obla Da* melody shows 14 relevant pattern classes, representing the chorus, verses, phrases and motives inside each of these structures. The 4 irrelevant patterns are redundant patterns subsumed by the 14 relevant ones.

In all these pieces, some patterns are considered as irrelevant because they cannot be perceived as such by listeners. Additional mechanisms should be added to prevent these irrelevant inferences, based on short-term memory, top-down mechanisms, etc.

7.2   About algorithm complexity

The algorithm complexity may be expressed first in terms of discovered structures: proliferation of redundant patterns, for instance, would lead to combinatorial explosion, since each new structure needs proper processes assessing its interrelationships with other structures, and inferring possible extensions. Hence a maximally compact description insures in the same time the clarity and relevance of the results and the limitation of combinatorial explosion. Concerning technical implementation, the prototype needs further optimisations.

The overall computational modelling results in a complex system formed by a large number of highly dependent mechanisms. Without a real synthetic vision of the whole system, no general assessment of the global complexity of the modelling has been achieved yet. The complete rebuilding of the modelling currently undertaken should enable a better awareness and control of complexity.

8   COGNITIVE STUDY OF TUNISIAN MODAL MUSIC

The project, presented in this paper, of modelling of musical pattern discovery processes has been integrated into a more general collaborative project between computer science, ethnomusicology and cognitive sciences. The main objective of this project is to design a cognitive modelling, using a complex system, of the processes of music perception and understanding, in order to understand the perceptive, musical and computational aspects of sequence segmentation and patterns recognition.

This study has been focused on Tunisian Modal Music, and particularly on *Tba'*, a modal system that presents interesting configurations. A *Tba'* is based on a *musical scale* – i.e. a set of pitches, such as (C, D, E, F, G, A, B) for instance – subdivided into two or several sub-scales called *genres*. Each *genre* is characterised by pivotal notes (more important than others) and melodic profile. Hence in a specific genre, the pitches that it contains are played in a specific order. Genres themselves are hierarchically connected one with the others. In the musical scale of the *Tba'*, some of the notes play particular role in the mode: some are mostly played at the beginning of the improvisation, or at the end of phrases. Finally, to each *Tba'* is also associated a set of characteristic melodic patterns. Figure 12 presents an example of *Tba'* modal structure.

The study has been focused on one particular improvisation by the *Nay* flute player Mohamed Saada, along the *Tba' Istikhbar Mhayyer Sika*. First, the improvisation has been transcribed from an audio record into a musical score. Then the resulting symbolic sequence has been analyzed by the modeling.

8.1   Psychological experiments

Psychological experiments have been carried out in order to obtain a detailed description of listening strategies, and to assess the role of cultural schemes in particular. This study has been focused in particular on the determination of the patterns that form the basic structures of the musical *genres*. In order to understand the impact of cultural knowledge on this particular task, two groups of subjects have been considered: one group formed by European subjects unfamiliar to Arabic modal music, and another group formed by Arabic subjects of various degree of expertise in this music. Subjects have been asked to performed several tasks successively: First of all, after hearing the musical piece, they have to recognise the most salient musical structures and, using these structures, to reduce the whole improvisation in order to exhibit the dynamic macro-structure of the piece.

The experiments have been first carried out in Paris on European subjects, and then extended to
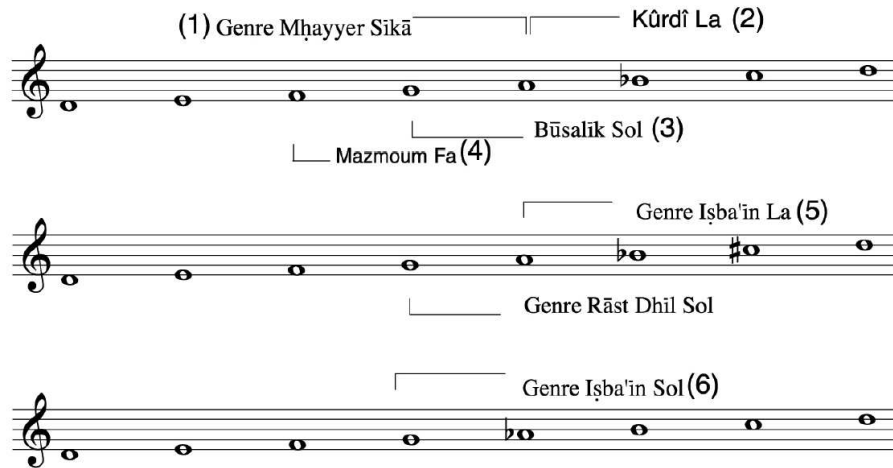
Figure 12: Description of the *Tba' Mhayyer Sika D(Ré)*, in terms of a sets of Genres and pivotal notes.

Arabic subjects in Tunisia. Results shows the relative variability and divergence of the judgements of European subjects. This is due to the fact that they cannot follow their own cultural scheme when analysing Arabic modal improvisations. They have to rely instead on the structural characteristics of the musical discourse, and in particular the discovery of repeated patterns. The experimental results of the listening tests have been used as a guideline in order to improve the model presented in the previous sections and to take into account the stylistic characteristics of Arabic modal improvisations.

8.2 Improvement of the modelling

One major limitation of the first version of the modelling, as presented in previous sections, is that only repetition of sequences of notes that are immediately successive could be detected. In music in general, and in modal improvisation in particular, repeated patterns are often ornamented: secondary notes can be added whose purpose is to emphasise the primary notes of the initial pattern. Figure 13 displays, for instance, a melodic phrase, and one possible ornamentation. To some of the notes of the original phrase are added secondary notes (displayed with smaller size in the score) that are located in the neighbourhood of the primary notes, both in time and pitch dimensions.

In order to take into account these ornamentation, a set of mechanisms have been added to the modelling. Solutions have been proposed [6] based on optimal alignments between approximate repetitions using dynamic programming and edit distances. We have developed algorithms that automatically discover, from the rough surface level of musical sequences, musical transformations revealing the sequence of pivotal notes forming the deep structure of these sequences. These mechanisms induce new connections between non-successive notes, transforming the *syntagmatic chain* of the original musical sequence into a complex *syntagmatic graph*. The direct application of the pattern discovery algorithm on this syntagmatic graph enables the detection of ornamented repetitions.

8.3 Results of the computational modelling

The analysis of Mohamed Saada's improvisation of *Istikhbar Mhayyer Sika* is displayed in figure 14. The discovered structures are represented below each line of the score. Each line represents an occurrence of the pattern, designated by a sign (1, 2, 3, 4, 5, + and -) on the left of the line. The notes actually considered by each pattern occurrence are represented by squares vertically aligned to the notes. These squares represents therefore the successive states along the pattern occurrence chain, as shown in Figure 1.

Pattern '-' represents a simple sequence of notes of continuously decreasing pitch heights, and pattern '+' represents a sequence of notes of continuously increasing pitch heights. Patterns 1 to 5 are sequences repeated several times in the improvisation. Each black square represents the beginning of a new occurrence, and each white square one successive state along the pattern chain. Grey squares corresponds to optional states that are not found in all the occurrences of the pattern. Finally, multiple branches designates multiple possible paths for one same pattern occurrence. The improvisation is built on the specific mode *Tba' Mhayyer Sika D*, characterised by the use of a specific set of notes (D, E, F, G, A, Bb, C) and a specific melodic figure, which corresponds exactly to the pattern 2. The beginning of the improvisation is also based on the successive repetition of pattern 1, which corresponds to a periodic melodic curve starting from note F and ending to the same note F, which is therefore a pivotal note of the improvisation. This

Figure 13: A melodic phrase and an ornamented version of it.

Figure 14: Analysis of the beginning of *Istikhbar Mhayyer Sika* improvised at the *Nay* flute by Mohamed Saada.

pattern correspond to the mode *Mazmoum F* shown in figure 12. The second line of the improvisation is characterised by the successive repetition of pattern 3, which is a little melodic line progressively transposed. Pattern 4 corresponds to another important melodic profile associated to pattern 2. Finally the two last lines of the improvisation are characterised by the repetition of pattern 5. Patterns 2 and 4 may be considered as stylistic characterisations of the mode *Istikhbar Mhayyer Sika* whereas patterns 1, 3 and 5 shows the characteristics of the individual style of the improviser.

The integration of these new mechanisms is not completely achieved. The application of the pattern discovery algorithm in the general syntagmatic graph leads to combinatorial explosion of redundant patterns not fully controlled yet, which will need further works.

9   CURRENT RESEARCHES

9.1   Addition of segmentation principles.

The structures currently found are based solely on pattern repetitions. Segmentation rules based on *Gestalt* principles of proximity and similarity [22, 8] need to be added. Although this rule plays a significant role in the perception of large-scale musical structures, there is no common agreement on its application to detailed

structure, because it highly depends on the subjective choice of musical parameters used for the segmentations. The study will focus in particular on the competitive/collaborative interrelations between the two mechanisms, in particular the masking effect of local disjunction on pattern discovery.

9.2 From monody to polyphony.

Our approach is limited to the detection of repeated monodic patterns. Music in general is polyphonic, where simultaneous notes form chords and parallel voices. Researches have been carried out in this domain [9], focused on the discovery of exact repetitions along different separate dimensions. Our model will be generalised to polyphony following the syntagmatic graph principle. We are developing algorithms that construct, from polyphonies, syntagmatic chains representing distinct monodic streams. These chains may be intertwined, forming complex graphs along which the pattern discovery algorithm will be applied. Pattern of chords may also be considered in future works.

9.3 Applications to musical databases.

The automated discovery of repeated patterns can be applied to automated indexing of musical content in symbolic music databases. This approach may be generalised later to audio databases, once robust and general tools for automated transcription of musical sound into symbolic scores will be available. A new kind of similarity distance between musical pieces may be defined, based on these pattern descriptions, offering new ways of browsing inside a music database using pattern-based similarity distance.

9.4 Applications to non-musical domains.

In future works, we plan to apply the algorithm to case studies other than music scores, such as DNA sequences.

ACKNOWLEDGMENTS

REFERENCES

[1] R. Agrawal, T. Imielinski and A. Swami. "Mining association rules between sets of items in large databases". In *Proceedings of the International Conference on Management of Data (SIGMOD 93)*. 1993.

[2] H. Mannila, H. Toivonen and A. I. Verkamo. "Efficient Algorithms for Discovering Association Rules". In *Proceedings of the AAAI Workshop on Knowledge Discovery in Databases*. 1994.

[3] Y. Tanaka, K. Iwamoto and K. Uehara. "Discovery of Time-Series Motif from Multi-Dimensional Data Based on MDL Principle". *Machine Learning*, vol. 58, no. 2–3, 2005.

[4] J. Lin, E. Keogh, S. Lonardi and P. Patel. "Finding Motifs in Time Series". In *Proceedings of the International Conference on Knowledge Discovery and Data Mining*. 2002.

[5] D. Cope. *Computer and Musical Style*. Oxford University Press, 1991.

[6] P.-Y. Rolland. "Discovering Patterns in Musical Sequences". *Journal of New Music Research*, vol. 28, pp. 334–350, 1999.

[7] W. Dowling and D. Harwood. *Music Cognition*. Academic Press, 1986.

[8] E. Cambouropoulos. *Towards a General Computational Theory of Musical Structure*. Ph.D. thesis, University of Edinburgh, 1998.

[9] D. Meredith, K. Lemström and G. Wiggins. "Algorithms for discovering repeated patterns in multidimensional representations of polyphonic music". *Journal of New Music Research*, vol. 31, no. 4, pp. 321–345, 2002.

[10] N. Pasquier, Y. Bastide, R. Taouil and L. Lakhal. "Discovering frequent closed itemsets for association rules". In *Proceedings of the International Conference on Database Theory*. 1999.

[11] J. Pei, J. Han and R. Mao. "Closet: An efficient algorithm for mining frequent closed itemsets". In *Proceedings of the ACM-SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*. 2000.

[12] R. Agrawal and R. Skirant. "Mining Sequential Patterns". In *Proceedings of the International Conference on Data Engineering*. 1995.

[13] M. Zaki. "Efficient Algorithms for Mining Closed Itemsets and Their Lattice Structure". *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 4, pp. 462–478, 2005.

[14] B. Ganter and R. Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer-Verlag, 1999.

[15] J. Han, W. Gong and Y. Yin. "Mining Segment-Wise Periodic Patterns in Time-Related Databases". In *Proceedings of the International Conference on Knowledge Discovery and Data Mining*. 1998.

[16] J. Han, G. Dong and Y. Yin. "Efficient Mining of Partial Periodic Patterns in Time Series Database". In *Proceedings of the International Conference on Data Engineering*. 1999.

[17] S. Ma and J. Hellerstein. "Mining partially periodic event patterns with unknown periods". In *Proceedings of the International Conference on Data Engineering*. 2001.

[18] W. W. J. Yang and P. Yu. "InfoMiner+: Mining Partial Periodic Patterns with Gap Penalties". In *Proceedings of the IEEE International Computer on Data Mining*. 2002.

[19] G. Assayag, C. Rueda, M. Laurson, C. Agon and O. Delerue. "Computer Assisted Composition at Ircam: From PatchWork to OpenMusic". *Computer Music Journal*, vol. 23, no. 3, pp. 59–72, 1999.

[20] T. Eerola and P. Toiviainen. "MIR In Matlab: The MIDI Toolbox". In *Proceedings of the International Conference on Music Information Retrieval*. 2004.

[21] N. Ruwet. "Methods of Analysis in Musicology". *Music Analysis*, vol. 6, no. 1–2, pp. 4–39, 1987.

[22] F. Lerdahl and R. Jackendoff. *A Generative Theory of Tonal Music*. The MIT. Press, 1983.