MODELING MUSICAL PATTERN PERCEPTION AS INDUCTION OF ANALOGIES INSIDE A SEMANTIC NETWORK

Olivier Lartillot
Ircam – Centre Pompidou, Paris, France

ABSTRACT

General methodologies for analyzing music — even structuralist ones — implicitly rely on perceptual principles. Indeed, music cannot be thoroughly understood without an appreciation of its communicative value. In fact, all limitations encountered by contemporary approaches of automated musical pattern discovery stem from an insufficient consideration for perception. It would be of high benefit, therefore, to develop a computational approach of automated music analysis based on a cognitive modeling of music perception. This first step towards a cognitive understanding of musical pattern perception aims at conceiving a general cognitive system that is able to produce expected results without combinatorial explosion. A new general methodology for Musical Pattern Discovery is proposed, which tries to mimic the flow of cognitive and sub-cognitive inferences that are processed when hearing a piece of music. Patterns have to be discovered along the branch of a syntagmatic graph, which generalizes the syntagmatic chain for polyphonic context. A musical pattern class is defined as a set of characteristics that are approximately shared by different pattern occurrences within the score. Moreover, pattern occurrence not only relies on internal sequence properties, but also on external context. Onto the score is build pattern occurrence chains which themselves interface with pattern class chains. Pattern classes may be inter-associated, in order to formalize relations of inclusion or repetition. The implemented algorithm is able to discover pertinent patterns, even when occurrences are, as in everyday music, translated, slightly distorted, slowed or fastened. Such an understanding of music perception agrees with subjective experience. Such a computer modeling may offer to musicology a detailed and explicit understanding of music, and may suggest to cognitive science the necessary conditions for a virtual perception of musical pattern.

1. SOME NECESSARY CONDITIONS FOR MUSICAL PATTERN DISCOVERY

1.1. Pattern Characterization

The concept of musical pattern may be characterized following three main criteria:

- Pattern may result from implicit knowledge that cannot be obtained directly from the score, such as: expected phrase length or metric (Lerdahl and Jackendoff 1983). The trouble is, musical motives may be structured in an ambiguous way, through a breaking of these rigid rules.
- Low-level structural properties of the musical surface may be obtained through local boundary detection (Cambouropoulos 1998). For instance, grouping boundaries may be introduced between entities that contrast one with the other according to their pitch, duration, etc. Although such heuristics may enable an understanding of metric phenomenon, for instance, such local segmentation does not contribute to the understanding to the idea of musical pattern itself. Indeed, a musical pattern is implicitly built through contrastive aggregation.
- Finally, a musical pattern may be defined as a set of characteristic that is shared by several sets of notes throughout the score. These sets of notes are said to be similar in a certain sense. Such concept of similarity has to be explicitly defined. This repetition-oriented criterion of pattern seems to remain the most relevant one, since music motives are classically defined in this way.

1.2. Set Paradigm Vs. String Paradigm

In previous definition, a pattern is a repeated set of notes. A pattern may in fact be either a general set of notes (Wiggins, Lemström and Meredith 2002) or a sequence, that is: a succession of notes, where successive notes in the sequence are successive notes in the score. If pattern is not constrained to be a sequence, temporal distance between successive notes within this set may be arbitrarily large. Even if limitations are set on temporal distance between successive notes, non-pertinent patterns may be found. On the contrary, if pattern is constrained to be a sequence, motives hidden inside rich polyphonic accompaniment may be out of scope. The problem may be solved by searching for strings along the branches of a syntagmatic graph (see paragraph 3.1).

1.3. Musical Similarity

The idea of successiveness should not be considered in a rigid way in order to enable deletions of notes or insertions of new ones in the pattern. Dynamic programming (Rolland 1999) is the most classical way to handle such operations. But music features other kinds of sequence transformation, such as passing notes or appoggiaturas, which should be also considered.

Now patterns may be subject to other kinds of transformation. Simply transposed patterns may be detected by considering each pattern in its own transposition reference. For example, if patterns are described not with absolute pitch, but with relative pitch whose reference is the absolute pitch of the first note of the pattern, then such descriptions of transposed patterns are exactly identical. In the same way, slower and faster patterns may be considered as identical one with the other if a relative temporal representation is considered. For this purpose, instead

ISBN 3-931852-67-9
ISSN 1617-6847

of considering the temporal interval between successive notes, the quotient between current temporal interval and first temporal interval is considered.

But real music features much more complex transformations. In particular, pitch and temporal distortions may appear locally inside patterns. To handle such plasticity, more relative viewpoints of the pattern may be considered, such as the contour representation in particular. However, such a crude representation is so loose that non-pertinent repetitions may also be detected. In fact, when considering such local distortions, there exist no viewpoint sufficiently loose for finding an exact repetition but in the same time sufficiently detailed for avoiding non-pertinent inferences. Therefore approximate repetition has to be tentatively inferred, to be *induced* from rough phenomenon, even if risks have to be taken.

1.4. Incremental Inference of Similarity

Lots of musical phenomenon deeply relies on the fact that music is progressively perceived, and that the listener itself progressively infers new knowledge about what he is currently hearing. Therefore, music listening should be considered as a kind of progressive reasoning. That is why some configurations are not detected and therefore not pertinent, simply because they cannot be caught during progressive listening. Hence, pattern cannot be defined solely along internal description, but also along external criteria, or context. It is senseless, therefore, to measure the similarity between sequences out of their context.

The incremental and logical thinking that builds human perception of music is ruled by fundamental principles, which are necessary for insuring a coherent process. For example, every time a sequence is considered as an occurrence of a pattern, every suffix could themselves be considered as occurrences of other pattern class, for simple mathematical reasons. But cognitively speaking, such inferences are not pertinent, since they do not correspond to inference human makes when listening to music. This is due to the fact that the first longest pattern was sufficient to explain the phenomenon, and that further inferences of suffixes would only infringe a clear analysis of the score. That is why suffix of pattern should not be explicitly represented.

1.5. Selection

As many patterns may be found, pertinent patterns are considered as those that feature a highest defined score (Cambouropoulos 1998). Such selecting mechanism is a classical and efficient way to extract important knowledge. It should be remarked, however, that this global selection, although enabling a quick characterization of a piece, infringes a thorough understanding of the complete score. We would like to retrieve also little detail at particular places, that may be of high relevance, and that may be taken into account by an active listening. The only necessary condition for a pattern to be considered as pertinent is that its score (here a degree of activation) has to exceed a certain minimal threshold. Therefore, to pattern *selection* we would prefer the concept of pattern *detection*.

Finally, since the process of pattern discovery proceeds itself through explicit characterization, there is no need to characterize a posteriori the patterns that have been discovered.

2. DATA REPRESENTATION

2.1. Pattern Class And Occurrence

The fact that several sequences are considered as similar in a certain sense means that they all belong to a same abstraction, which may be considered as a pattern class. These sequences are therefore occurrences of the pattern class. In this way, any new sequence sharing the same similarity will simply be considered as a new occurrence of this pattern class. The pattern class is not represented by a single prototype, but by all its occurrences that are effectively linked to it.

2.2. Pattern Class Chain

According to the incremental characteristic of music perception, patterns are progressively discovered, interval by interval, from initial interval to whole pattern. Pattern classes have to be represented following this cognitive constraint. In particular, all possible prefixes of a pattern may be considered as a pattern, uncompleted indeed, but still a pattern. Therefore, prefix of pattern classes are pattern classes. The set of all prefixes of a pattern class, ordered by growing order of pattern length, is a chain, called pattern class chain (PCC). Progressively, for each new note of the pattern that is being discovered, a new pattern class is added as a continuation of the previous pattern class. In this way, if a prefix of a discovered pattern class appears later, it will simply be considered as a new occurrence of the intermediary pattern class associated to this pattern.

2.3. Pattern Occurrence Chain

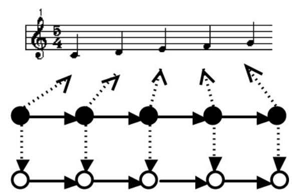

Figure 1: The POC (black circles) interfaces notes in the score with the corresponding PCC (white circles).

For each pattern occurrence, an additional interface should associate the sequence of notes inside the score that constitutes the occurrence with the pattern class. Such interface may also be described as a chain — called pattern occurrence chain — where each successive state within the POC represents at the same time a note in the sequence and a state in the PCC (see Figure 1).

2.4. Pattern Associations

The idea of segmentation may implicitly and dangerously suggest that score features only one level of pattern representation. On the contrary, patterns of different lengths may coexist and there may be inclusion or intersection relationships between them. Thus pattern cannot simply be characterized through an enumeration of similar intervals. The inner description as explained below should be made explicit too, and should be inferred by the machine.

We propose to represent such relationship between pattern and sub-pattern as follows. If occurrences of a pattern class feature a particular sub-pattern, a new POC, representing this sub-pattern, is linked to the PCC of the pattern itself. With such linking inside PCCs, a new association network is build between pattern classes. This high-level organization may help the recognition of basic pattern occurrences. In this way, expectations are generated by the system during the analysis: when a new occurrence of the pattern is discovered, sub-patterns are also *expected* (Meyer 1956).

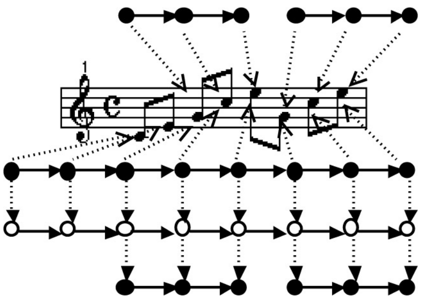

Figure 2: On the first bar of Bach's Prelude may be build a POC (below the score) that is associated to the 8-note PCC (white circles), and two POC for the 3-note PCC (over the score). These 3-note patterns, since they are repeated on different 8-note patterns, are represented directly on the 8-note PCC with two additional POCs (at the bottom).

2.5. Pattern Repetition

If a pattern is repeated successively several times, occurrences are themselves elementary objects that forms a meta-pattern. In particular, local intervals may be considered between such successive occurrences. Such a successive repetition of occurrences of a same pattern class may simply be represented by extending each pattern with the first note of the following pattern, and by associating to this added note the POC of this same pattern class.

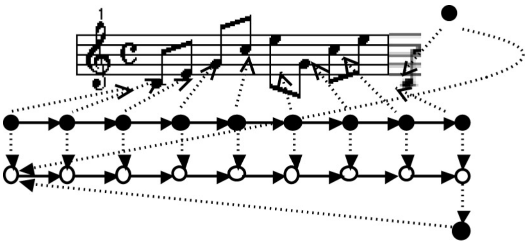

Figure 3: When a pattern is repeated more than twice, the last note of the pattern is linked to the first note through an additional POC on the PCC.

Such a mechanism is not as arbitrary as it may appear. When perceiving such successive repetitions of occurrences of a same 8-note pattern, we actually perceive 9-note pattern, where last note is in the same time the first note of a new pattern. Thanks to this mechanism, every time we perceive a whole occurrence of the 9-note pattern, we then expect a new occurrence again.

3. ALGORITHMS

3.1. Syntagmatic Graph

In paragraph 1.2, we explained that a pattern should be considered as a sequence. This means that successive notes in the pattern should have *syntagmatic relationships*. Let n be current note. Let m_1, m_2, … be the set of simultaneous notes that precedes n but in the same time are temporally closest to n. Let mi be the note that has the closest pitch compared to n. Then m_i is a *syntagmatic predecessor* of n. Now if there exist past notes p_1, p_2, … whose pitch is *particularly closer* (defined below) to the pitch of n than any other note already known to be close to n, then the p_i that is the temporally closest note is also a *syntagmatic predecessor*. This last operation is repeated as long as necessary.

Now the pitch of p_i is *particularly closer* to the pitch of n than any other close notes when the quotient between the pitch distance between p_i and n and the pitch distance between any close note and n is below a certain threshold. In this way, a current note may accept several possible previous notes. Such interval between any previous note and current note, which is the building element of pattern construction, will be called local interval. Thanks to this decomposing of polyphony into streams, overlapping patterns may be discovered.

We then obtain a *syntagmatic graph* — a generalization of *syntagmatic chain* — where nodes are notes and directed arcs link syntagmatic predecessor with successor. Patterns have to be discovered along the branches of this graph.

3.2. Pattern Class Discovery

In this section, we will show how our system is able to detect new pattern classes, that is, new abstractions. As told previously, a pattern is defined as an approximately (or exactly) repeated sequence. So pattern will be discovered only if a similarity relationship is inferred between a current sequence and a past one. Past sequence has to be recalled because of its similarity with current sequence. The trouble is: current sequence does not already exist as a sequence if repetition itself is not already detected. In our previous works, we alleviated the task by imposing a constraint, which can be expressed as follows: for a new pattern repetition to be detected, the repetition of each single interval of the patterns has to be explicitly and progressively discovered. In particular, the similarity between the first interval of each patterns has to be inferred before inferring the similarity of the remaining of the pattern. The trouble is, such a constraint can hardly be satisfied. A generalization of this algorithm will then be proposed, that can overcome previous limitation.

First Approach. First, every local interval has to be memorized in an associative memory that is able to retrieve any interval similar to a query. For this purpose, a hash-table associates for each interval parameter the set of its occurrences within the score. Now if the hash-table shows a similarity between current local interval i_1 and an old local interval i_1', a new pattern class is inferred (unless already discovered) associated to this single interval. Then if there exists any similarity between an interval i_2 that follows previous local interval i_1, and an interval i_2' that

ISBN 3-931852-67-9
ISSN 1617-6847

follows previous old local interval i_1', then a new pattern class extends previous pattern class. And so on.

If i_1 and i_1' have to be identical for being considered as similar, then pattern featuring a slight distortion on its first interval will not be detected. Therefore, a looser comparison between i_1 and i_1' should be tolerated. But in this case, lots of non-pertinent little patterns will be inferred too. Moreover, with such approach it is not possible to detect patterns with different speed, since i_1 and i_1' should have similar inter-onset value.

Second Approach. We propose to improve our first approach as follows. If current interval i_1 is particularly similar to an old local interval i_1', then a pattern class is inferred as previously. If, on the contrary, this similarity is not very high, previous local intervals i_0 that precede i_1 are considered, and compared to previous local intervals i_0' that precede i_1'. If the sequence i_0-i_1 is considered as similar to the sequence i_0'-i_1', then a pattern class is inferred, that consist of this succession of two intervals. In this way, a pattern may be detected even if its first interval was not a sufficient clue. Now such approach may be immediately generalized to *n* intervals instead of 2.

Pattern Class Extension. Once a new pattern class has been discovered, its extension is an easier task. Indeed, the new local interval that extends the discovered new pattern just have to be compared to possible continuations of the discovered old pattern, instead of comparing it to all possible intervals through the hash-table. Indeed, thanks to the previous pattern class initiation, two or more similar contexts have been discovered in the score. Pattern extension just consists of a deeper analysis of found contexts.

4. CONCLUSION AND FUTURE WORKS

This model has been implemented as a library of *Open Music*, a musical representation software developed at Ircam (Assayag and al. 1999). This new library called *OMkanthus* is able to find pertinent patterns in MIDI files, but also numerous non-pertinent ones. We surmise that such bad behavior may be avoided in the future through the integration of new general cognitive heuristics inside the framework.

The proposed model and implementation are still in an early phase, showing numerous limitations. Some further improvements include chord pattern discovery, comparison of sub-patterns associated to a pattern (inferring the similarity between sub-pattern themselves, comparing the relative pitch and temporal distance between sub-patterns). In a long term, such approach may try to go beyond pattern and catch higher-level concepts. Would a system be able to retrieve music theory?

Acknowledgments. My doctorate project[1] is supervised by Emmanuel Saint-James (LIP6, Paris 6) and Gérard Assayag (Musical Representation Team, Ircam). Some ideas arise from stimulating discussion with Professor Douglas Hofstadter and with my colleague Benoit Meudic.

7. REFERENCES

1. Assayag, G. and al. (1999). Computer Assisted Composition at Ircam: From *PatchWork* to *OpenMusic*. *Computer Music Journal 23(3), 59.*

2. Cambouropoulos, E. (1998). *Towards a General Computational Theory of Musical Structure*. Ph. D. thesis. University of Edinburgh.

3. Lartillot, O. (2003). Perception-Based Advanced Description of Abstract Musical Content. In *Digital Media Processing for Multimedia Interactive Services* (to appear). World Scientific Press: Singapore.

4. Lerdahl, F. and Jackendoff, R. (1983). *A Generative Theory of Tonal Music*. MIT Press.

5. Meyer, L. (1956). *Emotion and Meaning in Music*. The University of Chicago Press.

6. Nattiez, J.-J. (1990). *Music and discourse: towards a semiology of music*. Princeton University Press.

7. Rolland, P.-Y. (1999). Discovering Patterns in Musical Sequences. *Journal of New Music Research, 28(4): 334-350.*

8. Wiggins, G., Lemström, K. and Meredith, D. (2002). SIA(M)ESE: An Algorithm for Transposition Invariant, Polyphonic Content-Based Music Retrieval. *Proceedings of the International Conference on Music Information Retrieval* (pp. 283-284). Ircam: Paris.

[1] http://www.ircam.fr/equipes/repmus/lartillot

ISBN 3-931852-67-9
ISSN 1617-6847