



**HAL**  
open science

## Fragmentations with pitch, rhythm and parallelism constraints for variation matching

Mathieu Giraud, Ken Déguernel, Emilios Cambouropoulos

► **To cite this version:**

Mathieu Giraud, Ken Déguernel, Emilios Cambouropoulos. Fragmentations with pitch, rhythm and parallelism constraints for variation matching. International Symposium on Computer Music Multi-disciplinary Research (CMMR), Oct 2013, Marseille, France. pp.298-312, 10.1007/978-3-319-12976-1\_19 . hal-00906462v1

**HAL Id: hal-00906462**

**<https://hal.science/hal-00906462v1>**

Submitted on 19 Nov 2013 (v1), last revised 21 Jan 2015 (v2)

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Fragmentations with pitch, rhythm and parallelism constraints for variation matching

Mathieu Giraud and Emiliios Cambouropoulos

<sup>1</sup> LIFL, CNRS, Université Lille 1, France  
mathieu@algomus.fr

<sup>2</sup> Department of Music Studies, Aristotle University of Thessaloniki, Greece  
emilios@mus.auth.gr

**Abstract.** Composers commonly employ ornamentation and elaboration techniques to generate varied versions of an initial core melodic idea. Dynamic programming techniques, based on edit operations, are used to find similarities between melodic strings. However, replacements, insertions and deletion may give non-musically pertinent similarities, especially if rhythmic or metrical structure is not considered. We propose, herein, to compute the similarity between a reduced query and a melody employing *only fragmentation* operations. Such fragmentations transform one note from the reduced query into a possible large set of notes, taking into account metrical, pitch and rhythm constraints, as well as elementary parallelism information. We test the proposed algorithm on two prototypical “theme and variations” piano pieces by W. A. Mozart and show that the proposed constrained fragmentation operations are capable of detecting variations with high sensitivity and specificity.<sup>3</sup>

**Keywords:** melodic similarity, reduced melody, variations, fragmentation, musical parallelism

## 1 Introduction

Ornamentation, embellishment, elaboration, filling in are common strategies employed by composers in order to generate new musical material that is recognized as being similar to an initial or reduced underlying musical pattern. This way musical unity and homogeneity is retained, whilst at the same time, variation and change occur. This interplay between repetition, variation and change makes music “meaningful” and interesting. Listeners are capable of discerning common elements between varied musical material primarily through *reduction*, i.e. identifying “essential” common characteristics. Systematic music theories (e. g. Lerdahl and Jackendoff [19]) explore such processes, as do high-level descriptions [10, 22] or semi-Schenkerian computational models [20]. In this paper, we try to identify ornamentations of a given reduced melodic pattern. The proposed pattern matching algorithm employs not only pitch information but also additional rhythmic properties and elementary parallelism features.

---

<sup>3</sup> Preprint. Accepted to Computer Music Multidisciplinary Research (CMMR 2013), Marseille, October 2013.

Pattern matching methods are commonly employed to capture musical variation, especially melodic variation, and may be based on dynamic programming techniques. Similarity between melodies can be computed by the Mongeau-Sankoff algorithm [23] and its extensions, or by other methods for approximate string matching computing edit-distances, that is allowing a given number of restricted edit operations [7, 9, 13, 14]. The similarities can be computed on absolute pitches or on pitch intervals in order to account for transposition invariance [4, 11, 16, 25]. Note that some music similarity matching representations do not use edit-distance techniques [1, 8, 17, 21]. Geometric encodings also provide transposition invariance [18, 26, 27].

In edit-distance techniques, the allowed edit operations are usually matches, replacements, insertions, deletions, consolidations and fragmentations. However, edit operation such as replacement, insertions and deletions of notes are adequate for various domains (e.g. bioinformatics [12]) but present some problems when applied to melodic strings. In the general case, insertions or deletions of notes in a melodic string seriously affect metrical structure, and the same is true for substitutions with a note of different duration. Fragmentations and consolidations may be a further way to handle some aspects of musical pattern transformation [6, 23]. In [2], Barton et al. proposed to focus only on consolidation and fragmentation operations on pitch intervals: the sum of several consecutive intervals in one melodic sequence should equal an interval in another sequence. Their algorithm identifies correctly variations, including transposed ones, of a given reduced pattern, but incorrectly matches a large number of false positives, the consolidation and the fragmentation being applied only on the pitch domain.

In this paper it is asserted that identifying variations (that contain ornamentations) of a given reduced melodic pattern is best addressed using fragmentation operations, taking into account both *pitch* and *rhythm* information, along with other higher level musical properties such as *parallelism*. Apart from leaving aside replacement, insertion and deletion operations (only fragmentation is employed), this paper gives emphasis to rhythmic properties of melodic strings and other higher level structural features (e.g. similar ornamentations are introduced for similar underlying patterns) showing that such information increases both sensitivity and specificity of melodic variation detection.

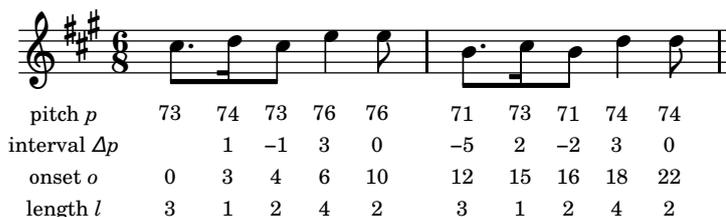
The current study is not meant to provide a general method for identifying variations of a given melodic pattern, but rather an exploration of some factors that play a role in some “prototypical” cases of musical variation. Two sets of variations by W. A. Mozart (K. 265 and K. 331) that are commonly used in composition as prototypical examples illustrating a number of basic variation techniques (extensive ornamentation, rhythmic variation, modal change) are used as the focal point of this research; these sets of variations are quite challenging for computational modeling as the number of notes can vary significantly between different versions (some variations may have 8 times or more notes than the underlying thematic pattern). Variations, however, appear in many guises and musical similarity is very difficult to pin down and define sys-

tematically in a general way [3]; two sets of variations by the same composer are hardly sufficient for studying the general phenomenon of musical variation – further research that takes into account a much larger variation dataset will be necessary. In the last section, limitations of the current proposal are discussed and future developments suggested.

The paper is organized as follows. Section 2 presents some definitions, Section 3 and 4 describe the algorithm and its results on two sets of variations by Mozart. On the *Andante grazioso* of Mozart K. 331, the variation matching with pitch, rhythm and parallelism constraint has an almost perfect sensitivity with more than 80% precision. Section 5 discusses some perspectives of this work.

## 2 Definitions

A note  $x$  is described by a triplet  $(p, o, \ell)$ , where  $p$  is the pitch,  $o$  the onset, and  $\ell$  the length. The pitches can describe diatonic (based on note names) or semitone information. We consider ordered *sequence of notes*  $x_1 \dots x_m$ , that is  $x_1 = (p_1, o_1, \ell_1), \dots, x_m = (p_m, o_m, \ell_m)$ , where  $0 \leq o_1 \leq o_2 \leq \dots \leq o_m$  (see Fig. 1). All the sequences used in this paper are *monophonic*: there are never two notes sounding at the same onset, that is, for every  $i$  with  $1 \leq i < m$ ,  $o_i + \ell_i \leq o_{i+1}$ . We do not handle overlapping notes.



**Fig. 1.** A monophonic sequence of notes, represented by  $(p, o, \ell)$  or  $(\Delta p, o, \ell)$  triplets. In this example, onsets and lengths are counted in sixteenthths, and pitches and intervals are represented in semitones through the MIDI standard.

*Approximate matching through edit operations.* Let  $S(a, b)$  the score of the best *local* alignment between two monophonic sequences  $x_{a'} \dots x_a$  and  $y_{b'} \dots y_b$ . This score can be computed by dynamic programming [23]:

$$S(a, b) = \max \begin{cases} S(a-1, b-1) + \delta(x_a, y_b) & \text{(match, replacement)} \\ S(a-1, b) + \delta(x_a, \emptyset) & \text{(insertion)} \\ S(a, b-1) + \delta(\emptyset, y_b) & \text{(deletion)} \\ S(a-k, b-1) + \delta(\{x_{a-k+1} \dots x_a\}, y_b) & \text{(consolidation)} \\ S(a-1, b-k) + \delta(x_a, \{y_{b-k+1} \dots y_b\}) & \text{(fragmentation)} \\ 0 & \text{(local alignment)} \end{cases}$$

$\delta$  is the score function for each type of mutation. If the last line (0) is removed, this equation computes the score for the best *global alignment* between  $x_1 \dots x_a$  and  $y_1 \dots y_b$ . Moreover, initializing to 0 the values  $S(0, b)$ , the same equation computes the score for the best *semi-global* alignment, that is the score of all candidate occurrences of the sequence  $x_1 \dots x_a$  (seen as a pattern) inside the sequence  $y_1 \dots y_b$ .

The complexity of computing  $S(m, n)$  is  $O(mnk)$ , where  $k$  is the number of allowed consolidations and fragmentations.

The image displays a musical score for the first two measures of the theme and six variations of the Andante, K. 331 by Mozart. The score is written in 3/8 time and D major. It includes a reduction 'R' of the theme, followed by the theme itself, and variations I through VI. The theme starts with a circled D in the first measure. Various annotations like '+' and '(+)' are placed above notes in the variations.

**Fig. 2.** The two first measures of the theme and variations of the Andante, K. 331, by Mozart, preceded by a reduction R of the theme. In the theme, the circled  $D$  of the first measure is a *neighbor tone*, as the  $C^\sharp$  of the second measure. This neighbor tone  $D$  can also be found in the variation VI, as an appoggiatura. The  $D$  that is present in the first measure of the other variations is better analyzed as a *passing tone* between  $C^\sharp$  and  $E$  (a similar role of passing tone can be also argued in a medium-scale interpretation of the theme). Finally, there are no such  $D$ s in variation I. A “note for note” alignment between the theme and variation III, IV and V, including some deletions and insertions, does not lead here to a satisfactory analysis. Less errors are done when considering fragmentations between the reduced pattern and the variations.

### 3 A fragmentation operation for variation matching

Allowing many fragmentations may produce many spurious matches: often fragmentations are thus restricted to only 2, 3 or 4 notes, of same length and pitch. However, fragmentation with more notes and with different pitches does occur in real cases, especially when a pattern is ornamented. Moreover, if we consider a reduced pattern, then almost any variation of the pattern can be seen as a fragmentation of this reduction. For example, the variations of the *Andante grazioso* of Mozart K. 331 (Figure 2) can be seen as a fragmentation between 2 and 6 notes of a reduced pattern, using chord tones but also ornamental tones.

More specifically, we take fragmentation to mean that a relatively long note is fragmented into shorter notes of the same overall duration (*length* constraint), and that the pitch of at least one of the shorter notes matches with the initial long note (*pitch* constraint). Additionally, metric structure is taken into account by assuming that patterns start only on beats. Finally, if the given reduced theme (query) comprises of repeating pitch and/or rhythm patterns, we assume that the same ornamentation transformations will be applied on the repeating pitch/rhythm patterns (this *parallelism* constraint is enforced in a post-processing stage).

We thus propose here to consider a semi-global pattern matching between a reduced pattern  $x_1 \dots x_a$  and a monophonic sequence  $y_1 \dots y_b$  (where  $y_1$  is on a beat, or preceded by a rest on a beat) with only fragmentations:

$$S(a, b) = \max_k S(a - 1, b - k) + \delta(x_a, \{y_{b-k+1} \dots y_b\})$$

The only operation considered here is the fragmentation of a note  $x_a$  into  $k$  notes  $\{y_{b-k+1} \dots y_b\}$ . We require that the score function  $\delta(x_a, \{y_{b-k+1} \dots y_b\})$  checks the following constraints:

- *length* constraint – the total length of the notes  $\{y_{b-k+1} \dots y_b\}$ , with their associated rests, is exactly the length of  $x_a$ ;
- *pitch* constraint – at least one of the pitches  $y_{b-k+1} \dots y_b$  must be equal to the pitch of  $x_a$ , regardless of the octave. To match minor variations, we simply use a “diatonic equivalence”, considering as equal pitches differing from only one chromatic semitone (or, when the pitch spelling is not known, allowing  $\pm 1$  semitone between the sequences, as in the  $\delta$ -approximation [5, 24]).

We are not interested here into fine-tuning error costs:  $\delta(x_a, \{y_{b-k+1} \dots y_b\})$  equals 0 when the constraints are met, and  $-\infty$  otherwise.<sup>4</sup>

Finally, we also propose a post-filtering that applies very well to the variations technique. Usually, inside a variation, the same transformation pattern is

<sup>4</sup> Note that with these simplified costs, a consequence of the length constraint is that, at each position, there is at most one fragmentation for each note  $x_a$  – so dynamic programming can be implemented in only  $O(mn)$  time.

applied on several segments of the theme, giving a unity of texture. In Figure 2, variation I could be described by “sixteenth with rest, using chromatic neighbor tones”, possibly with the help of some high-level music formalism [10, 19, 22]. We propose here a simple filter that will be very computationally efficient. The unity of texture often implies that the underlying base pitch is heard at similar places (+ marks on the Figure 2). We thus applied a refinement of the *pitch* constraint:

- *pitch position parallelism* filtering – when applying the pitch constraint on a pattern divided into equal segments, at least one matched pitch must be found at the same relative position in at least two segments.

For example, on Figure 2, all + marks, except the ones in parentheses in variation V, occur at the same relative position in both measures.

## 4 Results

### 4.1 *Andante grazioso*, Piano Sonata 11 (K. 331)

In order to evaluate the proposed algorithm we apply it on a set of “theme and variations” by W. A. Mozart, namely, the first movement *Andante grazioso* of the Piano Sonata 11 in A major (K. 331).

In a study on the recognition of variations using Schenkerian reduction [20], the author uses 10 sets of variations by Mozart; only the first four bars of each theme (10 themes) and variations (77 variations) are used for testing the proposed system. In the current study, the set of variations is used searching for a reduction of the theme in the whole melodic surface of the piece (144 bars).

We started from `.krn` Humdrum files [15], available for academic purposes at `kern.humdrum.org`, and kept only the melody (without acciaccaturas). The query is the reduced theme melody R (top of Figure 2), consisting of the four notes  $C^\# E B D$ . We choose this pattern, having three occurrences in each variation, instead of the full eight-notes pattern  $C^\# E B D A B C^\# B$  which has only one complete occurrence in each variation.

Results are summarized on Table 1, and all alignments corresponding to “length + pitch (diatonic)” constraints can be downloaded from `www.algomus.fr/variations`. In the theme and each variation, 3 occurrences have to be found. As our fragmentations can handle very large sets of notes, the 3 truth occurrences are always found, except for the variation III, in minor, when using pitch matching without diatonic equivalence.

The algorithm has thus an almost perfect sensitivity (recall), and should be evaluated for his precision. Allowing any fragmentation (even starting only on beats) leads to many spurious results. Adding only the pitch constraint does not help so much. Adding only the length constraint gives matching every sequence of two measures against the pattern.

As soon as both pitch and length constraints are enforced, the algorithm gives good results, with very few false positives: In the majority of the variations, only

m13@288 → m15@336 | Score = 4  
 Check pitch parallelism: [x2 ?] ++ Parallelism [x2] found.

m14@312 → m16@360 | Score = 4  
 Check pitch parallelism: [x2 ?] -- [x3 ?] -- [x4 ?] -- no common pitch position → discarding

**Fig. 3.** Two overlapping matches found on the minor variation of K. 331, at measures 13 and 14, including each one 4 fragmentations into 6 notes (F6), with length and pitch  $\pm 1$  constraints. The \* marks indicate the pitches that are identical to the query (with the approximation  $C = C^\#$ ). (Top.) The real match is confirmed by the parallelism of pitch positions (sixteenths number 1, 5, 7 and 10 inside each measure) (Bottom.) This spurious match is discarded, as the position of the matching pitches are different in the two measures.

m6@720 → m8@1008 | Score = 4  
 Check pitch parallelism: [x2 ?] ++ Parallelism [x2] found.

**Fig. 4.** A false positive in variation III of K. 331, that has also common pitch positions. This false positive occurs (with overlapping) one measure later than a true occurrence, at measure 5, that is correctly detected.

	theme	variations						sens	prec
		I	II	III	IV	V	VI		
number of notes	88	156	201	201	121	351	304		
no constraint (all frag. 1...20)	85	153	198	198	118	348	301	100%	< 2%
pitch	79	108	186	129	112	327	260	100%	< 2%
length	36	45	37	40	55	37	81	100%	6%
length + pitch	3	3	8	0	3	14	13	83%	41%
length + pitch + parallelism	3	3	6	0	3	3	4	83%	82%
length + pitch (diatonic)	3	3	8	5	4	14	14	100%	46%
length + pitch (diatonic) + parallelism	3	3	6	3	3	3	4	100%	84%
ground truth	3	3	3	3	3	3	3		

**Table 1.** Number of occurrences of the reduced pattern  $C^\# E B D$  found in the theme and variations of the *Andante grazioso* of the Piano Sonata 11 by Mozart (K. 331). Several fragmentation operations are tested. The columns “sens” and “prec” represents the sensitivity (recall) and precision of the proposed algorithm compared to the ground truth (3 occurrences in the theme and each variation). In all the cases, these 3 occurrences are found by the method (true positives), except for the variation III, in minor, when not using diatonic pitch matching. The “no constraint” line is directly related to the number of notes of the variation – there are matches everywhere.

the 3 true occurrences are found. The best results are here when using afterwards the “pitch parallelism” constraint (on two halves of the pattern), filtering out some spurious matches (see Figure 3). This method has an overall 84% precision. False (or inexact) positives can still happen in some situations (Figure 4), but they are very few: only 4 in this piece. Moreover, some false positives are overlapping with true matches, and could be discarded with a more precise scoring system.

#### 4.2 Variations on *Ah vous dirai-je maman* (K. 265)

We also tested the algorithm on another Mozart piano piece, *Twelve Variations on “Ah vous dirai-je, Maman”* (K. 265). The melody has been extracted manually and encoded in symbolic notation (363 bars in our MIDI encoding). The query consists here in the eight notes  $C G A G F E D C$ , this full theme appearing twice in each variation, totaling 26 occurrences. The parallelism constraint here applies on the three first bars of this pattern, requiring that at least two bars share common pitch positions.

Results are summarized on Table 2. With pitch, length and parallelism constraints, the algorithm outputs 20 true positive occurrences (sensitivity of 77%) with regular pitch matching, and 22 true positive occurrences (sensitivity of 85%) with  $\pm 1$  pitch matching. Note that the sensitivity is not as perfect as in K. 331: For example, on some variations, the length constraint can not be enforced (see Figure 5, bottom). Again, there are very few false positives, especially when the parallelism constraint is required.

m25@2304 → m32+1q@3024 | Score = 8  
 Check pitch parallelism: [x4 ?] ++ Parallelism [x4] found.

m25@2304 m26@2400 m27@2496 m28@2592

m29@2688 m30@2784 m31@2880 m32@2976

m40+1q@3792 → m48+1q@4560 | Score = 8  
 Check pitch parallelism: [x4 ?] ++ Parallelism [x4] found.

F5 F8 F8 F7

F8 F7 F6 F3

Fig. 5. Two matches found in the first variation of K. 265. (Top.) Good match and alignment. (Bottom.) Although the end of this match is a true positive, the alignment is wrong; the true occurrence should be shifted a quarter before, but the ties on the melody prevent a good alignment respecting the length constraint across the fragmentations.

## 5 Discussion : towards a unique *transformation* operation

In this paper, we have shown that a unique edit operation – a fragmentation – gives very good results in matching a reduced query against a theme and a set of variations. The key point in our approach is to focus on musically relevant fragmentations, allowing very large fragmentations, but restricting them with metrical, rhythm and pitch information along with some parallelism.

A very simplified matching procedure and error cost have been used in this study. This simple model has produced good results in two sets of variations that contain instances of extensive ornamentation. However, the model may be improved in many ways. For instance, fine-tuning scores for the  $\delta$  function could improve the results, allowing imperfect fragmentations and some other classical operations. The “parallelism” constraint that was tested here is also very

	occ (tp)	sens	prec
length + pitch	39 (20)	77%	51%
length + pitch + parallelism	29 (20)	77%	69%
length + pitch ( $\pm 1$ )	71 (22)	85%	31%
length + pitch ( $\pm 1$ ) + parallelism	36 (22)	85%	61%
ground truth	(26)		

**Table 2.** Number of occurrences (occ) of the reduced pattern  $C G A G F E D C$  found in *Twelve Variations on “Ah vous dirai-je, Maman”* by W. A. Mozart (K. 265, 1387 notes in our encoding). The ground truth has 2 occurrences in the theme and each variation, totaling 26 occurrences. The column “(tp)” shows the number of true positives found by each method, and the columns “sens” and “prec” give the associated sensibility and precision. As we encoded the files in MIDI, without pitch spelling information, we used here a  $\pm 1$  semitone pitch approximation to match the minor variation (but it also brings some spurious occurrences).

simple (same pitch position on several segments of a pattern), and the number of segments was manually selected for each piece. This parallelism constraint could be extended to become more generic, but its current simplicity makes it very suitable for efficient computation. Finally, these ideas could also be adapted to interval matching, to be transposition invariant.

Note that such an approach with fragmentations works because we start from a reduced query. Going a step further, we argue that relevant similarities between two melodies – and maybe even between polyphonic pieces – should be computed with a *unique transformation operation* of a group of several notes  $\{x_1, x_2 \dots x_\ell\}$  into another group of notes  $\{y_1, y_2 \dots y_k\}$ . The traditional edit operations of match/replacement/insertion/deletion, along with fragmentation and consolidations, can be seen as particular cases of this transformation operation, one set of notes being reduced to a singleton or to the empty set. In such a framework, computing  $\delta(\{x_1, x_2 \dots x_\ell\}, \{y_1, y_2 \dots y_k\})$  may require several steps, possibly including dynamic programming with the more classical operations. Seeing transformation as the basic operation could yield musical similarities that span a larger range than usual operations.

## References

1. Sven Ahlbäck. Melodic similarity as a determinant of melody structure. *Musicae Scientiae*, Discussion Forum 4A:235–280, 2007.
2. Carl Barton, Emiliós Cambouropoulos, Costas S. Iliopoulos, and Zsuzsanna Lipták. Melodic string matching via interval consolidation and fragmentation. In *Artificial Intelligence and Innovations Conf. (AIAI 2012)*, pages 460–469, 2012.
3. Emiliós Cambouropoulos. How similar is similar? *Musicae Scientiae*, Discussion Forum 4A:7–24, 2009.

4. Emiliós Cambouropoulos, Tim Crawford, and Costas S. Iliopoulos. Pattern processing in melodic sequences: Challenges, caveats and prospects. In *Arti Intelligence and Simulation of Behaviour (AISB 99)*, pages 42–47, 1999.
5. Emiliós Cambouropoulos, Maxime Crochemore, Costas S. Iliopoulos, Laurent Mouchard, and Yoan J. Pinzon. Algorithms for computing approximate repetitions in musical sequences. *Int. Journal of Computer Mathematics*, pages 129–144, 1999.
6. Michael Clausen. Modified Mongeau-Sankoff algorithm. <http://www-mmdb.iai.uni-bonn.de/forschungsprojekte/midilib/english/saddemo.html>.
7. Raphaël Clifford and Costas S. Iliopoulos. Approximate string matching for music analysis. *Soft. Comput.*, 8(9):597–603, 2004.
8. Darrell Conklin and Christina Anagnostopoulou. Segmental pattern discovery in music. *INFORMS Journal on Computing*, 18(3), 2006.
9. Tim Crawford, Costas S. Iliopoulos, and Rajeev Raman. String matching techniques for musical similarity and melodic recognition. *Computing in Musicology*, 11:71–100, 1998.
10. Diana Deutsch and John Feroe. The internal representation of pitch sequences in tonal music. *Psychological Review*, 88(6):503–522, 1981.
11. Mathieu Giraud, Richard Groult, and Florence Levé. Subject and counter-subject detection for analysis of the Well-Tempered Clavier fugues. In *Computer Music Modeling and Retrieval (CMMR 2012)*, 2012.
12. Dan Gusfield. *Algorithms on Strings, Trees, and Sequences*. Cambridge University Press, 1997.
13. Pierre Hanna, Pascal Ferraro, and Matthias Robine. On Optimizing the Editing Algorithms for Evaluating Similarity Between Monophonic Musical Sequences. *Journal of New Music Research*, 36:267–279, 2007.
14. Walter B. Hewlett and Eleanor Selfridge-Field, editors. *Melodic Similarity: Concepts, Procedures, and Applications*. MIT Press, 1998.
15. David Huron. Music information processing using the Humdrum toolkit: Concepts, examples, and lessons. *Computer Music Journal*, 26(2):11–26, 2002.
16. Tetsuya Kageyama, Kazuhiro Mochizuki, and Yosuke Takashima. Melody retrieval with humming. In *Int. Computer Music Conference*, pages 349–351, 1993.
17. Olivier Lartillot. Motivic pattern extraction in symbolic domain. *Intelligent music information systems: Tools and methodologies*, pages 236–260, 2007.
18. Kjell Lemström and Esko Ukkonen. Including interval encoding into edit distance based music comparison and retrieval. In *Symp. on Creative and Cultural Aspects and Applications of AI and Cognitive Science (AISB 2000)*, pages 53–60, 2000.
19. Fred Lerdhal and Ray Jackendoff. *A Generative Theory of Tonal Music*. MIT Press, 1983, 1996.
20. Alan Marsden. Recognition of variations using automatic Schenkerian reduction. In *Int. Society for Music Information Retrieval Conf. (ISMIR 2010)*, pages 501–506, 2010.
21. Alan Marsden. Interrogating melodic similarity: a definitive phenomenon or the product of interpretation? *Journal of New Music Research*, 41(4):323–335, 2012.
22. David Meredith. A geometric language for representing structure in polyphonic music. In *Int. Society for Music Information Retrieval Conf. (ISMIR 2012)*, 2012.
23. Marcel Mongeau and David Sankoff. Comparison of musical sequences. *Computer and the Humanities*, 24:161–175, 1990.
24. Veli Mäkinena, Gonzalo Navarro, and Esko Ukkonen. Transposition invariant string matching. In *Int. Symp. on Theoretical Aspects of Computer Science (STACS 2003), LNCS 2607*, pages 191–202, 2003.

25. Daniel Müllensiefen and Klaus Frieler. Modelling experts' notion of melodic similarity. *Musicae Scientiae*, Discussion Forum 4A:183–210, 2007.
26. Rainer Typke. *Music retrieval based on melodic similarity*. PhD thesis, Univ. Utrecht, 2007.
27. Esko Ukkonen, Kjell Lemström, and Veli Mäkinen. Geometric algorithms for transposition invariant content based music retrieval. In *Int. Conf. on Music Information Retrieval (ISMIR 2003)*, pages 193–199, 2003.