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

Mathieu Giraud, Ken Déguernel, Emilios Cambouropoulos

# Fragmentations with Pitch, Rhythm and Parallelism Constraints for Variation Matching

Mathieu Giraud[1], Ken Déguernel[1], and Emilios Cambouropoulos[2]

[1] Laboratoire d'Informatique Fondamentale de Lille (LIFL)
UMR CNRS 8022, Université Lille 1, France
{mathieu, ken}@algomus.fr

[2] 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 deletions 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 pitch and rhythm constraints, as well as elementary parallelism information. We test the proposed algorithm on four "theme and variations" piano pieces by W. A. Mozart and L. van Beethoven and show that the proposed constrained fragmentation operations are capable of detecting simple variations with high sensitivity and specificity.[3].

**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

[3] Author pre-print of the final version.
Citation: Mathieu Giraud, Ken Déguernel, Emilios Cambouropoulos, Fragmentations with Pitch, Rhythm and Parallelism Constraints for Variation Matching International Symposium on Computer Music Multidisciplinary Research (CMMR 2013), LNCS 8905, pp.298–312, 2014.
http://dx.doi.org/10.1007/978-3-319-12976-1_19

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]. We try here 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.

Pattern matching methods are commonly employed to capture musical variations, especially melodic variations, 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 operations such as replacements, 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 simple 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.

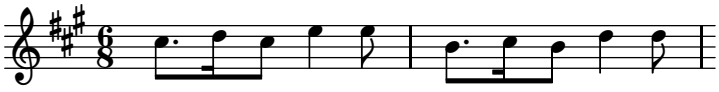
The current study is not meant to provide a general method for identifying variations of a given melodic pattern, but rather an exploration of some factors that play a role in some "prototypical" cases of musical variation. The chosen set of variations (W. A. Mozart K. 265, K. 331, K. 455 and Beethoven WoO 64) are

commonly used in composition as prototypical examples illustrating a number of basic variation techniques (ornamentation, rhythmic variation, modal change); these apparently simple sets of variations are already quite challenging for computational modeling as the number of notes varies 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 systematically in a general way [3]; 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 three sets of variations by Mozart and one set of variations by Beethoven, totaling 728 bars in 4 themes and 31 variations. The best results are obtained while combining pitch, length and parallelism constraints, with sensitivity between 70% and 85% and precision between 60% and 100%. 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 Figure 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.



| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| pitch $p$ | 73 | 74 | 73 | 76 | 76 | 71 | 73 | 71 | 74 | 74 |
| interval $\Delta p$ | | 1 | –1 | 3 | 0 | –5 | 2 | –2 | 3 | 0 |
| onset $o$ | 0 | 3 | 4 | 6 | 10 | 12 | 15 | 16 | 18 | 22 |
| length $l$ | 3 | 1 | 2 | 4 | 2 | 3 | 1 | 2 | 4 | 2 |

**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 sixteenths, 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]:

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

$\delta$ 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.

## 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 of a reduced pattern in 2 to 6 notes, 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). Finally, if the given reduced theme (query) comprises 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 $y_1 \dots y_b$ with only fragmentations:

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

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

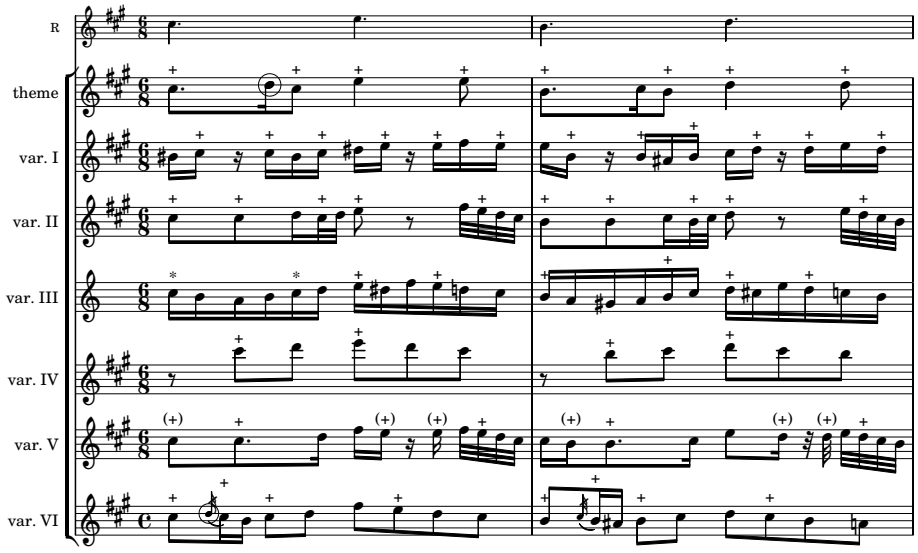



**Fig. 2.** The two first measures of the theme and variations of the Andante, K. 331, by Mozart, preceded by a reduction R of the theme. In the theme, the circled $D$ of the first measure is a *neighbor tone*, as the $C^{\#}$ of the second measure. This neighbor tone $D$ can also be found in the variation VI, as an appogiatura. The $D$ that is present in the first measure of the other variations is better analyzed as a *passing tone* between $C^{\#}$ 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 that would align all these $D$s but include 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.

– *length* constraint – the total length of the notes $\{y_{b-k+1}...y_b\}$, with their associated rests, is exactly the length of $x_a$;

– *pitch* constraint – at least one of the pitches $y_{b-k+1}...y_b$ 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: $\delta(x_a, \{y_{b-k+1}...y_b\})$ equals 0 when the constraints are met, and $-\infty$ otherwise. 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.

Finally, we also propose a post-filtering that applies very well to the variations technique. Usually, inside a variation, the same transformation pattern is applied on several segments of the theme, giving a unity of texture. In Figure 2, variation I could be described by "sixteenths 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 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 The Corpus

In order to evaluate the proposed algorithm, we apply it on the following sets of "theme and variations" (see Table 1). We started from either `.krn` Humdrum [15] or `.mid` files, keeping only the melody (without acciaccaturas):

– the first movement *Andante grazioso* of the Piano Sonata 11 in A major (K. 331) by W. A. Mozart,
– the variations on *Ah vous dirai-je maman* (K. 265) by W. A. Mozart,
– the variations on *Wilhelm von Nassan* (K. 25) by W. A. Mozart,
– and the *Six Easy Variations on a Swiss Song* (WoO 64) by Ludwig van Beethoven.

**Table 1.** Sets of variations used in this study. For Mozart K. 331, we started from the `.krn` Humdrum files available for academic purposes at `kern.humdrum.org`, and kept only the melody. For the other pieces, the melody has been extracted manually and encoded in `.mid` or `.krn` symbolic notation.

| | | | | | |
|---|---|---|---|---|---|
| Mozart | K. 331 | A major | 6/8 | theme + 6 variations | 144 bars |
| Mozart | K. 265 | C major | 2/4 | theme + 12 variations | 363 bars |
| Mozart | K. 25 | D major | 4/4 | theme + 7 variations | 144 bars |
| Beethoven | WoO 64 | F major | 4/4 | theme + 6 variations | 77 bars |

**Table 2.** Reduced themes used as query for the fragmentation matching, and number of ground truth occurrences of these themes in the considered sets of variations. The "parallelism" column display the number of segments used to check the parallelism constraint. Note that the length of the patterns is manually adapted for variations with another meter than the theme (variation 12 of K. 265, in 3/4, and variation 10 of K. 455, in 6/8).

| | | reduced theme | length | occurrences | parallelism |
|---|---|---|---|---|---|
| Mozart | K. 331 | $C^{\#}EBD$ | 2 bars | $7 \times 3$ | $\times 2$ |
| Mozart | K. 265 | $CGAGFEDC$ | 2 bars | $13 \times 2$ | $\times 4$ |
| Mozart | K. 25 | $DEF^{\#}EGF^{\#}$ | 4 bars | $8 \times 2$ | $\times 2$, $\times 2$ |
| Beethoven | WoO 64 | $CFACGFCCEFDC$ | 6 bars | $7 \times 1$ | $\times 2$ |

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 corpus has a total of 4 themes and 31 variations. This number may seem low, but the set of variations is used here searching for a reduction of the theme *in the whole melodic surface* of the piece (728 bars on the four pieces). Moreover, the search is not restrained to full bars – there could be an occurrence starting from any note. However, in the majority of cases, the length constraint will enforce the metrical structure of the pattern in the occurrences.

For each variation set, a reduced version of the theme (or part of the theme) is given as the query (Table 2). The algorithm then identifies matches of this theme in the piece. The following paragraphs details the results on the four sets. Some alignments corresponding to the best constraints can be downloaded from `www.algomus.fr/variations`.

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

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

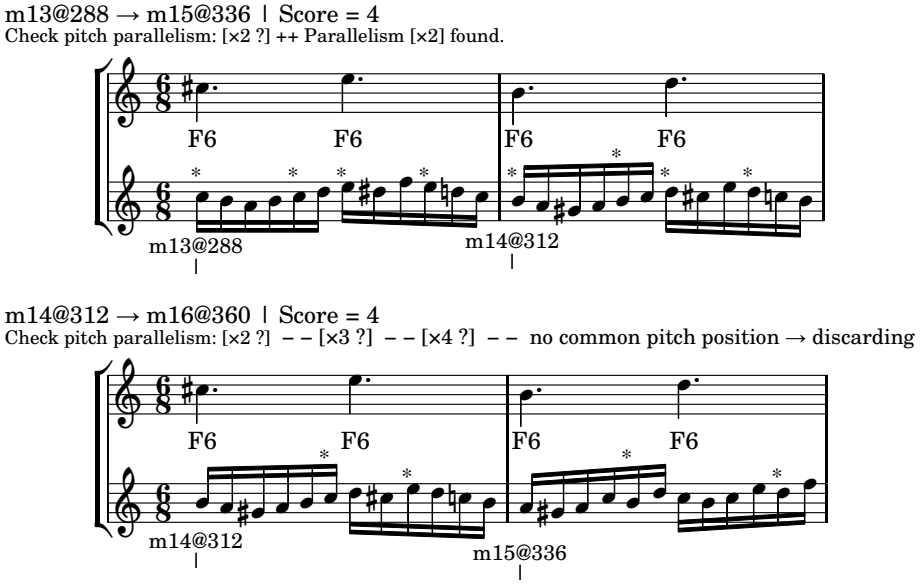



**Fig. 3.** Two overlapping matches found on the minor variation of K. 331, at measures 13 and 14, including each one 4 fragmentations into 6 notes (F6), with length and pitch $\pm 1$ constraints. The $*$ marks indicate the pitches that are identical to the query (with the approximation $C = C^{\#}$). (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.

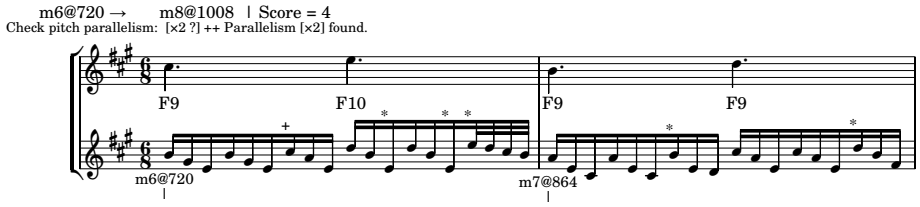



**Fig. 4.** A false positive in variation III of K. 331, that has also common pitch positions. This false positive occurs (with overlapping) one measure later than a true occurrence, at measure 5, that is correctly detected.

**Table 3.** 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.

| | 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 | | |

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 3. 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 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.

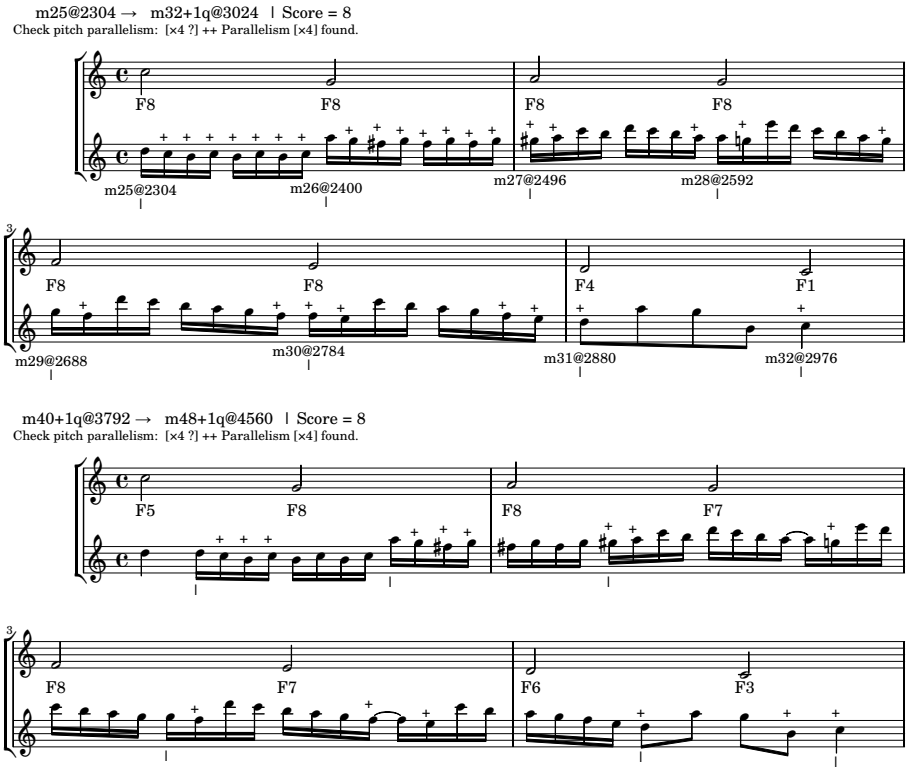



**Fig. 5.** Two matches found in the first variation of K. 265. (Top.) Good match and alignment. (Bottom.) Although the end of this match is a true positive, the alignment is wrong: the true occurrence should be shifted a quarter before, but the ties on the melody prevent a good alignment respecting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

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

For *Twelve Variations on "Ah vous dirai-je, Maman"* (K. 265), we selected a query as 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 four segments of one measure, requiring that at least two measures share common pitch positions.

Results are summarized on Table 4. 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, bot-

**Table 4.** 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 ±1 semitone pitch approximation to match the minor variation (but it also brings some spurious occurrences).

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

tom). Again, there are very few false positives, especially when the parallelism constraint is required.

### 4.4 Variations on *Wilhelm von Nassan* (K. 25)

On the variations on *Wilhelm von Nassan* (K. 25), the query appears twice in each variations, totaling sixteen occurrences (Figure 6). Since the reduced query is not symmetrical (two whole notes, then four half notes), we choose here to apply the parallelism constraint in two separated parts, first checking the first two bars together, and then the third and fourth bars.

With the pitch, length and parallelism constraint, the algorithm is able to find twelve true occurrences (sensitivity of 75%). The results are summed up in Table 5. Even with the parallelism constraint, eight false positives are found. Some of them are in fact true positives that are shifted in time, and therefore not correctly located.

However, fifteen false positives are discarded by the parallelism constraint, bringing the precision of the algorithm from 34% to 60%. This example shows again how a simple parallelism constraint discards many false positives, and therefore provides results with better precision.

### 4.5 Six Easy Variations on a Swiss Song (Beethoven, WoO 64)

On *Six Easy Variations on a Swiss Song* (L. van Beethoven, WoO 64), we use a longer reduced query with two anacrousis (Figure 7). The theme being eleven bars long, and the query six bars long, the query appears only once in each variations. Therefore, the pattern appears seven times in the whole piece. The parallelism constraint (see Figure 9) compares the pitch positions of the first three bars (with the anacrousis) and of the last three bars (also with the anacrousis).

**Table 5.** Number of occurrences of the reduced pattern found in *Wilhelm von Nassan* (K. 25) by W. A. Mozart. The ground truth has 2 occurrences in the theme and each variation, totaling 16 occurrences.

| | occ (tp) | sens | prec |
|---|---|---|---|
| length + pitch (± 1) | 35 (12) | 75% | 34% |
| length + pitch (± 1) + parallelism | 20 (12) | 75% | 60% |
| ground truth | (16) | | |

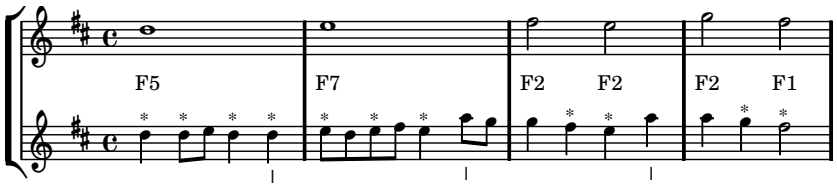



**Fig. 6.** Query used for the Variations on *Wilhelm von Nassan* (K. 25) aligned to the theme.

With the pitch, length and parallelism constraint, the algorithm is able to find five true occurrences (sensitivity of 71%). The pattern has not been found in the first variation, the pitch constraint not being respected (see Figure 8). In this variation set, no false positive has been found (precision of 100%), which can be explained by the length of the query. However, despite this length, the algorithm is still able to provide a very good sensitivity.

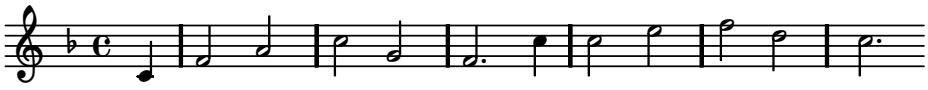


**Fig. 7.** Reduced theme of Beethoven's Six Easy Variations on a Swiss Song used as a query in the algorithm.

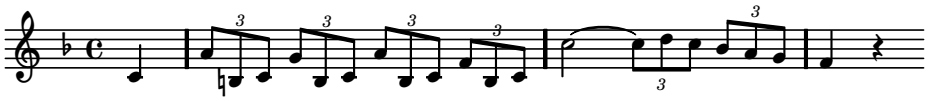


**Fig. 8.** First bars of the first variation of WoO 64. The A in the first bar does not respect the pitch constraint (F in the query), and therefore this occurrence is not found.

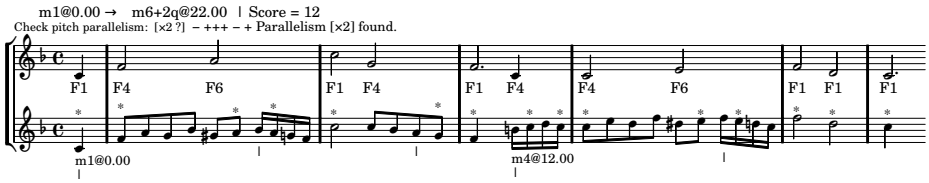



**Fig. 9.** A positive match found in the fifth variation of WoO 64. The parallelism constraint is enforced for at least one position in both segments.

## 5 Discussion

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 rhythm and pitch information along with some parallelism.

Such an approach with fragmentations works because we start from a reduced query. Moreover, a very simplified matching procedure and error cost have been used in this study. This simple model has produced good results in four 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 simple (one common pitch position on several segments of a pattern), and the number and the position of segments were 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, theses ideas could also be adapted to interval matching, to be transposition invariant.

### 5.1 Analysis of Elaborated Sets of Variations

An important limit of this present study is that the four chosen sets of variations are easy or intermediate piano pieces, and that their composition technique almost always respects the global layout of the theme. At the opposite end of the spectrum, one could look for example to the 15 "Eroica" Variations (op. 35)

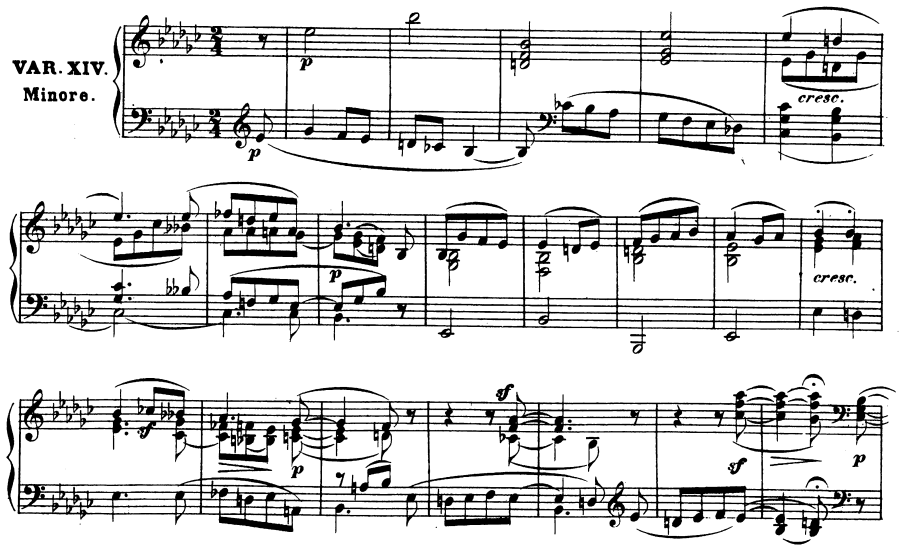



**Fig. 10.** Example of a more complex transformation on the "Eroica" variations by Beethoven (op. 35). In this set of variations, two melodies are transformed : the theme and the bass line. In the variation no. XIV, these melodies are tangled: During the first eight bars, the transformed bass line is played at the soprano, and the transformed is played at the bass. Then, the roles are exchanged.

or the 33 "Diabelli" Variations (op. 120) by Beethoven. Such pieces exhibit much complex transformations of the musical material. The proposed algorithm could extract *some* of these variations, but will be limited by the following facts:

- In many cases, there is not a single melody that can be extracted from the polyphonic texture, or the melody can alternate between several voices (see Figure 10);
- However, even in the cases where a melody can be extracted, further ornamentation and transformation sometimes do not respect the *length* constraint.

We used in this study only fragmentations, and not the usual insertion/deletion/substitution operations from [23], but, as stated in the introduction, these operations break the metrical structure as soon as they affect durations of individual notes. For us, a good generic solution based on the fragmentation operation could include some relaxation of the *length* constraints – thus allowing insertion of notes or group of notes – but at the same time shall include a *reinforcement of high-level constraints*, such as the parallelism operation.

### 5.2 Towards a Unique *Transformation* Operation

Going a step further, we argue that relevant similarities between two melodies – and maybe even between polyphonic pieces – should be computed with *a unique*

*high-level transformation operation* of a group of several notes $\{x_1, x_2...x_\ell\}$ into another group of notes $\{y_1, y_2...y_k\}$. The traditional edit operations of match/replacement/insertion/deletion, along with fragmentation and consolidation, 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 $\delta(\{x_1, x_2...x_\ell\}, \{y_1, y_2...y_k\})$ 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. Ahlbäck, S.: Melodic similarity as a determinant of melody structure. Musicae Scientiae Discussion Forum 4A, 235–280 (2007)
2. Barton, C., Cambouropoulos, E., Iliopoulos, C.S., Lipták, Z.: Melodic string matching via interval consolidation and fragmentation. In: Artificial Intelligence and Innovations Conf. (AIAI 2012). pp. 460–469 (2012)
3. Cambouropoulos, E.: How similar is similar? Musicae Scientiae Discussion Forum 4A, 7–24 (2009)
4. Cambouropoulos, E., Crawford, T., Iliopoulos, C.S.: Pattern processing in melodic sequences: Challenges, caveats and prospects. In: Arti Intelligence and Simulation of Behaviour (AISB 99). pp. 42–47 (1999)
5. Cambouropoulos, E., Crochemore, M., Iliopoulos, C.S., Mouchard, L., Pinzon, Y.J.: Algorithms for computing approximate repetitions in musical sequences. Int. Journal of Computer Mathematics pp. 129–144 (1999)
6. Clausen, M.: Modified Mongeau-Sankoff algorithm. http://www-mmdb.iai.uni-bonn.de/forschungprojekte/midilib/english/saddemo.html
7. Clifford, R., Iliopoulos, C.S.: Approximate string matching for music analysis. Soft. Comput. 8(9), 597–603 (2004)
8. Conklin, D., Anagnostopoulou, C.: Segmental pattern discovery in music. INFORMS Journal on Computing 18(3) (2006)
9. Crawford, T., Iliopoulos, C.S., Raman, R.: String matching techniques for musical similarity and melodic recognition. Computing in Musicology 11, 71–100 (1998)
10. Deutsch, D., Feroe, J.: The internal representation of pitch sequences in tonal music. Psychological Review 88(6), 503–522 (1981)
11. Giraud, M., Groult, R., Levé, F.: Subject and counter-subject detection for analysis of the Well-Tempered Clavier fugues. In: Computer Music Modeling and Retrieval (CMMR 2012) (2012)
12. Gusfield, D.: Algorithms on Strings, Trees, and Sequences. Cambridge University Press (1997)
13. Hanna, P., Ferraro, P., Robine, M.: On Optimizing the Editing Algorithms for Evaluating Similarity Between Monophonic Musical Sequences. Journal of New Music Research 36, 267–279 (2007), `http://hal.archives-ouvertes.fr/hal-00285560`
14. Hewlett, W.B., Selfridge-Field, E. (eds.): Melodic Similarity: Concepts, Procedures, and Applications. MIT Press (1998)
15. Huron, D.: Music information processing using the Humdrum toolkit: Concepts, examples, and lessons. Computer Music Journal 26(2), 11–26 (2002)
16. Kageyama, T., Mochizuki, K., Takashima, Y.: Melody retrieval with humming. In: Int. Computer Music Conference. pp. 349–351 (1993)

17. Lartillot, O.: Motivic pattern extraction in symbolic domain. Intelligent music information systems: Tools and methodologies pp. 236–260 (2007)
18. Lemström, K., Ukkonen, E.: 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). pp. 53–60 (2000)
19. Lerdhal, F., Jackendoff, R.: A Generative Theory of Tonal Music. MIT Press (1983, 1996)
20. Marsden, A.: Recognition of variations using automatic Schenkerian reduction. In: Int. Society for Music Information Retrieval Conf. (ISMIR 2010). pp. 501–506 (2010)
21. Marsden, A.: Interrogating melodic similarity: a definitive phenomenon or the product of interpretation? Journal of New Music Research, 41(4), 323–335 (2012)
22. Meredith, D.: A geometric language for representing structure in polyphonic music. In: Int. Society for Music Information Retrieval Conf. (ISMIR 2012) (2012)
23. Mongeau, M., Sankoff, D.: Comparison of musical sequences. Computer and the Humanities 24, 161–175 (1990)
24. Mäkinena, V., Navarro, G., Ukkonen, E.: Transposition invariant string matching. In: Int. Symp. on Theoretical Aspects of Computer Science (STACS 2003), LNCS 2607. pp. 191–202 (2003)
25. Müllensiefen, D., Frieler, K.: Modelling experts' notion of melodic similarity. Musicae Scientiae Discussion Forum 4A, 183–210 (2007)
26. Typke, R.: Music retrieval based on melodic similarity. Ph.D. thesis, Univ. Utrecht (2007)
27. Ukkonen, E., Lemström, K., Mäkinen, V.: Geometric algorithms for transposition invariant content based music retrieval. In: Int. Conf. on Music Information Retrieval (ISMIR 2003). pp. 193–199 (2003)