Gingras, B. & McAdams, S. (2011). Improved score-performance matching using both structural and temporal information from MIDI recordings. *Journal of New Music Research*, 41, 43-57.

Abstract

Automated score-performance matching is a complex problem due to the use of expressive timing by performers and the presence of notes that are unspecified in the score, such as performance errors and ornaments. Automated matchers typically use performance data extracted from MIDI recordings. For the most part, these algorithms use structural information, such as pitch and chronological succession, but do not use timing information. As a result, most matchers cannot deal satisfactorily with ornamented performances or performances that exhibit extreme variations in tempo. The matcher presented here relies both on structural and temporal information, allowing it to generate an accurate match even for heavily ornamented performances. Hand-made score-performance matches on a corpus of 80 MIDI recordings of organ performances of two pieces were used as ground truth data for a comparison with matcher results. The matcher achieved a nearly perfect accuracy rate. In addition, the matcher performed equally well or better than matchers previously described in the literature on a set of piano performances of two pieces by Chopin, thus demonstrating its versatility and robustness. We also propose a heuristic for the identification of ornaments and errors that is based on perceptual principles, and which could theoretically be amenable to empirical study. Finally, this matcher is designed to accommodate multi-channel MIDI recordings of performances from keyboard instruments with multiple manuals, such as organ or harpsichord. This feature makes it a potentially valuable tool for the investigation of ensemble performances of MIDI instruments.

1. Introduction

Music performance has been characterized as a component of a communication system in which composers code musical ideas in notation, performers transduce this notation into an acoustical signal, and listeners recode the acoustical signal into musical ideas (Kendall & Carterette, 1990). This model applies particularly to score-based music performance, which characterizes a significant proportion of classical Western musical practice. The score, written by a composer, generally specifies the pitch and duration categories of the notes to be played by the performer in an unambiguous manner, while conveying less specific information about exact tempo, articulation, dynamics and ornamentation (Large, 1993; Palmer, 1997). Depending on the repertoire, the performer has more or less freedom in deciding how to interpret the score, but pitches and nominal note durations are generally less subject to variation than other musical parameters, given that they can be categorically defined. Because the score provides an explicit benchmark with which the performance can be compared, score-based music performance has constituted the focus of research in music performance (Palmer, 1997).

In order to study score-based music performance quantitatively on a note-by-note basis, the researcher needs to determine the corresponding score note for every performance note, a process called *score-performance matching*. Although such matching can be done reliably by hand (Repp, 1996a), such a procedure becomes unwieldy for analyzing large databases of performances or performances of longer pieces. Fortunately, algorithms that automate this procedure have been developed. Such algorithms are called *matchers*. Automated matchers typically compare a representation of the performance (either audio or MIDI recording) to a symbolic representation of the score and try to seek the best match between both. In the last two

decades, several such matchers have been developed (Heijink, Windsor, & Desain, 2000b; Large, 1993; Puckette & Lippe, 1992). An important distinction should be made between matching algorithms whose main purpose is that of real-time accompaniment, often called *score following* (Dannenberg, 1984; Puckette & Lippe, 1992), and algorithms that are designed to find the best possible match for a performance, which we will call *offline matchers* (Heijink et al., 2000b; Large, 1993; Raphael, 2006). While the former are mostly concerned with efficiency and real-time responsiveness and are used in performance settings, the latter seek accuracy and are mainly used for research purposes (Heijink, Desain, Honing, & Windsor, 2000a).

The MIDI protocol does not provide an exact representation of the performance; MIDI records quantifiable data such as note onsets, note offsets, pitch, and velocity, but ignores other aspects such as timbre and spectral content. On the other hand, extracting performance information directly from the audio recording is a method that retains all sonic aspects of the performance and which can be used with non-MIDI instruments**.** However, until recently, direct matching of an audio recording of a performance to a score of a polyphonic piece has proven to be a challenging task, although researchers have addressed this problem (Dixon, 2005; Raphael, 2006). Altogether, for performance research focusing on timing, tempo, and articulation, MIDI does convey most, if not all, of the relevant information, and remains far easier to process than audio recordings, especially for polyphonic music and long performances. The present article will concern itself solely with MIDI recordings of keyboard performances.

Some authors have treated the problem of matching a performance to a score as a typical sequence-alignment problem (Large, 1993) and have sought to adapt solutions from other disciplines, such as nucleic acid or amino acid sequencing in molecular biology (Gotoh, 1982; Needleman & Wunsch, 1970). Thus, a number of matching algorithms define the best alignment

between two sequences *A* and *B* as the one for which the editing distance (usually defined as the number of changes such as deletions, additions, or substitutions) between *A* and *B* is the shortest (Mongeau & Sankoff, 1990). In cases where the performance closely matches the score, this model is generally adequate. However, even for expert performances, there is rarely a perfect one-to-one match between score and performance (Repp, 1996a). Discrepancies between score and performance can be attributed to three main factors: 1) performance errors, 2) temporal deviations brought about by expressive timing in performance, and 3) underspecification of scores (Heijink et al., 2000a).

A performance error can be defined in a very general way as an unintended deviation from the written score that occurs in performance (Palmer & Van de Sande, 1993). Most researchers have only considered errors that correspond to deletions (failure to play notes indicated in the score), additions (insertion of extraneous notes not indicated in the score) or substitutions (pitch errors or "wrong notes") (Repp, 1996a). Some researchers also take into account other error types that may be defined as "timing errors", or, to be more precise, chronological shifts between the succession of notes indicated in the score and that which was performed (Palmer & Van de Sande, 1993, 1995). This type of error should not be confused with temporal shifts caused by expressive timing (see below), although the boundary between them is necessarily subjective.

Because most matchers rely solely on a comparison between the chronological succession of notes and chords in the score and in the performance (Heijink et al., 2000b; Large, 1993), expressive timing in performance may affect the matching process by disrupting the order of the notes. For instance, a situation in which notes that should be played synchronously according to the score (for instance, notes belonging to the same chord) are played

asynchronously in performance can lead to wrong note assignments in the score-to-performance matching process. Such asynchronies are common occurrences in piano performance (Goebl, 2001; Palmer, 1989, 1996; Repp, 1996b).

Finally, scores generally indicate ornaments by means of symbols, which do not specify the exact timing of the ornaments, nor the number of notes that comprise them in the case of complex ornaments such as trills (Dannenberg & Mukaino, 1988). In addition, in certain musical genres, such as the Baroque repertoire, performers routinely add ornaments that are not specified in the score. This *underspecification* of the musical scores represents another obstacle for matchers in ornamented pieces, because editing-distance models assume an exact one-to-one mapping at the level of individual notes between score and performance (Pardo & Birmingham, 2001).

Indeed, in the case of performances that exhibit extreme expressive timing or heavy ornamentation, the analogy between score-performance matching and typical sequence-alignment problems does not apply: a performance may contain several additional notes not indicated in the score, and the order in which the notes are played in the performance may differ from the order in which they are notated. In this case, the score should be treated as a template that provides a more or less specific framework and indicates the key structural points, leaving several aspects of the performance, such as ornamentation and expressive timing, to be freely determined by the performer (Pardo & Birmingham, 2001).

Several authors have proposed using timing information to increase the accuracy of the score-performance matching process (Desain & Honing, 1992; Puckette & Lippe, 1992; Raphael, 2006). Hoshishiba and colleagues presented a matcher that uses temporal information (Hoshishiba, Horiguchi, & Fujinaga, 1996); however, the detailed implementation of this

matcher was not described. Vantomme (1995) developed a score follower that uses exclusively temporal information, reverting to pitch-matching only when the temporal matching fails, unlike most algorithms described in the literature. While attractive in the context of score-following, this approach is hardly suitable to offline matching given its disregard for pitch information.

Conversely, very few researchers have tackled issues related to the identification of ornaments. Dannenberg & Mukaino (1988) proposed an algorithm that can cope with specific ornaments, such as trills and glissandi, by relying on the fact that notes composing these ornaments usually have a much shorter duration than score notes, as long as they are indicated in the score. However, an algorithm that could handle all types of ornaments, regardless of whether they are specified in the score or not, would have a wider applicability to all kinds of musical situations.

Among the best-known offline matchers are those developed by Honing (1990), Large (1993) and Heijink and colleagues (Desain, Honing, & Heijink, 1997; Heijink, 1996; Heijink et al., 2000b). The *strict matcher* (Honing, 1990) takes the notated order of the notes in the score as a strict temporal constraint on the performance; the performance is processed note-by-note, and only one possible interpretation is considered at any point in time, which results in a high sensitivity to performance errors. In contrast, the matcher developed by Large (1993), which will be henceforth referred to as the *Large matcher*, is somewhat more robust because it divides the performance into clusters (notes played together) before trying to match it to the score and uses complete knowledge of the performance and score to find the globally optimal match. Furthermore, this matcher considers many possible alternative solutions at any point in time, and can analyze some performance errors, such as insertions, deletions, and substitutions. Indeed, it

has been used in the context of research on errors in piano performance (Palmer & Van de Sande, 1993).

In spite of their usefulness, these matchers present several limitations. The most important one is that they use only pitch and note order to find the optimal score-performance match, not taking into account voice structure or timing information. As a result, these algorithms cannot deal satisfactorily with ornamented performances or performances that exhibit extreme expressive timing such that the chronological succession of notes does not correspond to that indicated in the score. In an attempt to solve some of these problems, Heijink and colleagues (Desain et al., 1997; Heijink, 1996) proposed a *structure matcher*, which takes into account the voice information present in the score by assigning each score note to a voice. This matcher is able to cope with extreme expressive timing resulting in deviation in the chronological succession of notes. Nevertheless, the solution adopted by these authors is, by their own admission, "debatable" in that parallel events in different voices are considered to be temporally independent, a model which does not seem to accurately represent common musical practice.

Other problems encountered with the offline matchers discussed here involve a sensitivity to errors, and particularly errors involving repeated notes (Heijink et al., 2000b, p. 549). In addition, all MIDI-based offline matchers described in the literature were designed for the analysis of piano performance and cannot handle MIDI recordings of instruments with multiple manuals, such as the organ or harpsichord. Finally, most existing algorithms are designed to find a solution that maximizes the number of matched performance notes, regardless of the perceptual relevance of such an approach. However, a definition of the best match based solely on the number of matched notes is problematic, as it may ignore relevant structural and temporal information (Heijink et al., 2000b, p. 552).

In an attempt to overcome these limitations, we developed a matcher that relies both on structural information and on a temporal representation of the performance, which is obtained by sequentially tracking local tempo changes on a note-by-note basis and mapping performance events to the corresponding score events. This allows the matcher to generate an accurate match even for heavily ornamented performances. The best match is defined as the one that maximizes the number of matched performance notes, while minimizing the structural and temporal inconsistencies in the individual voices. Furthermore, this matcher is designed to accommodate multi-channel MIDI recordings. Finally, we propose a very general approach to the identification of ornaments. The second section of this article describes the algorithm used by the matcher, whereas the third section reports on the evaluation of this implementation. A final section discusses current limitations and possible improvements.

2. Description of the matcher

The matcher described here follows a three-step process; we will thus refer to it as the "three-step matcher". Before discussing each step in detail, we will outline this process. The first step is similar to the algorithm described by Large (1993) in that it decomposes the performance into note clusters (which we will subsequently refer to as "performance clusters") and establishes a preliminary match between performance clusters and score events by relying solely on the chronological ordering of events as well as pitch and note onset information. The second step takes into account both the results from the first step and the temporal information obtained from the MIDI data to construct a "temporal match" in which the onset times of score events are matched to corresponding performance clusters. Finally, the third step combines information from the first two steps to find the optimal note-by-note correspondence between score and

performance. Unmatched performance notes are identified as ornaments or errors at this stage. At each step, several possible alternatives are considered.

2.1 Symbolic representation of the score

As described by Schwarz, Orio, and Schnell (2004), the score is parsed into a time-ordered sequence of *score events*, where each score event corresponds to a change in the polyphonic texture (one or more note onsets or offsets). Each score note is thus bound in time by its *onset event* and its *offset event*. Score notes are also defined by their pitch, voice, and MIDI channel. Pitches are represented according to the standard MIDI format (Roads, 1996). Each voice and MIDI channel is represented by a unique number, and each score note is associated with a unique voice and MIDI channel. In this context, a *voice* is defined as a sequential collection of score notes played on the same instrument and MIDI channel, and usually associated with a specific pitch register and limb (left or right hand, or feet in the case of instruments equipped with a pedalboard) in the case of polyphonic instruments. A voice may run throughout an entire piece or span only a few measures. In addition, the matcher keeps track of embellishment markings in the score; this information is used for the identification of ornaments.

The use of voice information improves the quality of the match for polyphonic scores containing more than one voice, as it allows for a more refined representation of the musical structure of the score (Desain et al., 1997); likewise, notes that were played on different manuals on a MIDI-controlled organ, for instance, can be differentiated by taking into account the MIDI channel information. In contrast to the structure matcher, which treats parallel events in different voices as temporally independent (Desain et al., 1997), the temporal sequence of score events supersedes the voice information associated with each note in the case of the three-step matcher. Thus, the different voices are conceived as temporally related, so that notes in different voices

that share the same onset event are expected to have quasi-synchronous onsets, as is normally the case with common-practice music performance.

2.2 First step: cluster/event preliminary match

In the first step, performance notes are initially grouped into clusters according to the proximity of their onsets in time. Notes that are played quasi-synchronously are assumed to belong to the same score event (Schwarz et al., 2004). The three-step matcher initially groups together notes whose onsets can be found within a span of 40 milliseconds (this *maximum inter-onset interval* corresponds approximately to the maximal onset asynchronies observed in professional music performance; see Rasch, 1979) and whose onset times are closer to each other than to those of any other notes. This initial parsing is used to estimate the *average onset time distance* between adjacent clusters. This value is then used to generate a more refined parsing which adjusts the size of the maximum inter-onset interval according to the average onset time distance. One advantage of this two-step parsing is that it is more flexible than the procedure employed by matchers that use a fixed maximum inter-onset interval for the parsing of performance notes into clusters (Honing, 1990; Large, 1993). Moreover, while the parsing of the performance notes into clusters is a critical step in the strict matcher and the Large matcher, it does not determine the final results for the three-step matcher, because an erroneous parsing can be corrected in subsequent steps.

Once the second parsing is completed, *cluster/event ratings*, which range between 0 and 100, are computed for each performance cluster/score event combination, in order to evaluate potential matches between performance clusters and score events. These ratings are based on the following four criteria: (1) a comparison of the number of performance onsets and score notes (*NOO*), (2) a MIDI channel congruence rating *MID* which evaluates how closely the number of

performance onsets matches the number of score notes specified for each MIDI channel (for multichannel MIDI recordings), (3) the proportion of performance notes showing an exact pitch and MIDI channel match with at least one score note (*COR*), and (4) a pitch distance rating *PDR*. Given a performance cluster *P* and a score event *S*, such that $P = \{p_1, p_2, p_3, \ldots, p_m\}$ and $S = \{s_1, s_2, s_3, \ldots, s_n\}$, where p_i and s_j represent MIDI pitch values, $|P|$ = *size of set P* (number of onsets *p* in performance cluster) and $|S|$ = *size of set S* (number of onsets *s* in score event, counting unisons only once), the cluster/event rating *CER* for the pair (*P,S*) is defined as:

$$CER(P,S) = COR(P,S) + \max\left(0, 100 - \frac{\alpha NOO(P,S) + \beta MID(P,S) + \gamma PDR(P,S)}{\max(|P|,|S|)} | COR(P,S)\right) \quad (1)$$

where α, β, and γ are constants reflecting the relative weight of each rating, and:

$$NOO(P,S) = abs(|P| - |S|) \quad (2)$$

$$MID(P,S) = \sum_{c=1}^{m} abs(|P_c| - |S_c|) \quad (3)$$

where $P_c = \{p: p \text{ is played in MIDI channel } c\}$ and $S_c = \{s: s \text{ is notated so as to be played in MIDI channel } c\}$, and

$$COR(P,S) = \frac{|P| - |W|}{\max(|P|,|S|)} \times 100 \quad (4)$$

$$\text{where } |W| = \sum_{c=1}^{m} \left| P_c \cap \overline{S_c} \right| \qquad (5)$$

Finally, $PDR(P,S)$ is a function that computes the global pitch distance (in semitones) between P and S by mapping P_c onto S_c (as a surjection) for each MIDI channel c so as to minimize the pitch distance between P_c and S_c, with 0 representing an exact pitch match between P and S.

A table containing these cluster/event ratings for the entire performance is then built (Table 1). Note that more than one performance cluster may be perfectly matched to the same score event. Inversely, a performance cluster may not correspond perfectly to any score event (as shown in Table 1), a situation which may be caused by ornamentation or performance errors. It is normally unnecessary to compute values for the entire table, because it is unlikely that actual score event/performance cluster pairings will be located far from the main diagonal going from the top left to the bottom right part of the table. Such calculations are computationally expensive and time-consuming, especially for performances containing hundreds or thousands of events. On the other hand, if the matcher does not consider all possible solutions, there is a risk that the optimal solution will be missed. Therefore, there must be a trade-off between computational efficiency and finding the best solution. The three-step matcher uses a measure of structural discrepancy to evaluate how many score event/performance cluster pairings should be computed. This *discrepancy index* is defined as the maximum of the ratio of the number of performance clusters to the number of score events and the ratio of the number of performance onsets to the number of score onsets. When these ratios deviate considerably from a value of one, it suggests that the performance is heavily ornamented and/or that it contains several errors.

[Insert Table 1 around here]

The cluster/event ratings obtained at this stage are then used to generate a *cluster/event preliminary match*, which takes into account the chronological succession of events (but not the

timing information). This cluster/event preliminary match includes only unique events that are perfectly matched. Unique events are defined as being found only once in a span corresponding to approximately 20 events. The purpose of this preliminary match is to establish a set of *landmark events* that will be used in the following steps. This step may prove to be crucial in instances where substantial sections of the score were omitted in performance (such as when several chords or even entire bars were skipped in performance) or when a performance is heavily ornamented.

Scores that comprise a greater number of unique events will be conducive to good cluster/event matches, whereas pieces that have a small number of recurrent events, or that contain many similar events, tend to generate poor matches, regardless of the discrepancy index value between performance and score. This, of course, becomes increasingly relevant when the identical events are proximal in the score. The problem of repeated notes, as well as the larger issue of event similarity was mentioned by both Heijink et al. (2000b) and Large (1993), but they did not propose a coherent approach to this problem. The three-step matcher tackles this issue by computing an *event diversity index*, based on Shannon's diversity index (1948), and uses this information to estimate the number of solutions that should be considered in the following steps (temporal matching and note-by-note matching), so that a greater number of solutions are generated for scores that contain many similar or identical events. An event species is defined as the total population of score events (for a given score) that are structurally identical, that is, they contain the same number of note onsets, the same pitches, and are played on the same manuals (in the case of keyboard instruments). Under this definition, a score comprising a single recurrent event, such as a repetition of the same chord, will contain one event species, while a score for

which all events are different will contain as many species as they are events. The event diversity index *H'* is then defined as:

$$H' = -\sum_{i=1}^{S} p_i \ln p_i \qquad (6)$$

where S is the number of event species present in the score, and p_i denotes the relative abundance of event species i, calculated as the proportion of events of species i with respect to the total number of events in the score.

Finally, a performance with no errors or ornaments and a moderate amount of expressive timing will give a better *score-performance fit* than one that is either error-filled or that uses expressive timing deviations that create asynchronies between hands, such that the note order in performance differs from that indicated in the score. The quality of the score-performance fit F is quantified in the following manner:

$$F = \frac{\sum_{i=1}^{S} \frac{|n_i - m_i|}{\max(n_i, m_i)} \times n_i}{\sum_{i=1}^{S} n_i} \qquad (7)$$

where S represents the number of event species in the score, n_i the number of score events in species i, and m_i the number of perfectly matched performance clusters/score event pairings for species i. A large difference between n_i and m_i for a given event species i reflects a poor score-performance fit, which may be due to performance errors or to ornamentation. The value of S varies between 0, indicating a perfect fit, and 1, indicating a complete lack of fit.

Although very crude, this measure of fit provides a good assessment of the difficulty involved in matching a specific performance to a given score. Thus, the matcher takes into account the discrepancy between the number of performance clusters and score events, the score-performance fit, as well as the event diversity index to determine the number of solutions to be

computed. This approach has the advantage of tailoring the computational needs to the difficulty of the matching task.

2.3 Second step: temporal matching

The temporal matching is probably the feature that most significantly differentiates the three-step matcher from the majority of offline matchers described in the literature, and it proves to be crucial in determining the quality of the final match. During this step, the matcher initially uses information from the cluster/event preliminary match computed in the first step to predict the onset time associated with each score event, using onset times of landmark events as a starting point, and proceeding in a sequential way (that is, one score event at a time). The probable onset time of each event is estimated using a local tempo model, which attributes a greater weight to events closely following or preceding the current event than to events that are more distant in time (Vantomme, 1995).

A delicate issue associated with temporal matching is determining the size of the temporal window for which performance-cluster candidates corresponding to a given score event should be considered. Temporal deviations in performance may be due to motor noise (Desain & Honing, 1993) or abrupt changes in tempo such as *ritardandi* or *accelerandi*. However, it may also be that a score event was omitted in performance. An erroneous interpretation in such situations may lead the temporal matcher completely astray and negatively affect the quality of the match. Vantomme (1995) used a "window of belief" to estimate the maximum tolerance in onset time deviation, resorting to pitch information only when the deviation for an expected event was greater than this tolerance threshold. Conversely, the three-step matcher evaluates the *event rating* of performance-cluster candidates, which is defined as the sum of the cluster/event rating obtained in the first step and a temporal rating that is based on the distance between the

predicted onset time and the mean onset time of the notes belonging to the performance cluster. The relative weight ascribed to the cluster/event rating depends on the value of the score-performance fit (see Equation 7), so that the weight of the temporal rating increases when the score-performance fit is poor.

Moreover, the temporal matcher follows an iterative process, optimizing the quality of the match over several cycles: at each step, several solutions are considered, and only the ones with the highest ratings are selected. This step-by-step procedure increases the robustness of the matching process by making it less susceptible to errors brought about by local temporal deviations or score/performance mismatches. During the initial cycles, onset times of score events are predicted for both forward (proceeding from the first score event to the last) and backward (proceeding from the last score event to the first) passes. Solutions are obtained by pairing the forward and backward matches that show the highest agreement between onset times and retaining only the onset times that are common to both matches. The resulting match is then passed on to the next cycle, and onset times are computed for both forward and backward passes using information from the previous cycle until a stable solution is reached. Then, a new series of cycles is conducted, taking the match with the highest global event rating as the basis for the following cycle until a stable solution is reached (no distinction is made between backward and forward passes at this stage).

2.4 Third step: note-by-note matching

The third step consists of a specific note-by-note matching that uses information from the two previous steps and takes into account both voice and MIDI channel assignment for each note. As its name implies, the main difference between this note-by-note matching step and the

previous steps is that performance notes are considered individually instead of being grouped into clusters. It is during this final step that errors and ornaments are identified.

During this step, a temporal fit between individual notes and score events is first estimated by computing *onset difference ratings* as a function of the time difference between the onsets of performance notes and the predicted onsets of score events obtained from the temporal matching step. All performance notes whose onsets occur within 250 ms of a predicted score event onset are considered as possible candidates for a match; in addition, a minimum of three score events are considered as candidates for any given performance note, regardless of the onset time difference.

As with the temporal matcher, the note-by-note matcher follows an iterative process, optimizing the quality of the match over several cycles and considering several solutions at each step. For each score event e_i, a *match rating* is computed between every score note s belonging to this event and each candidate performance note p. This match rating $MAT(s,p)$ is based on the onset difference rating $ODR(p,e_i)$, which is a dimensionless value comprised between 0 and 100, and the *pitch-distance rating* $PDR(s,p)$, calculated from the pitch interval (in semitones) between s and p. In order to ensure that the relative weights of $ODR(p,e_i)$ and $PDR(s,p)$ are adjusted to the tempo of the performance, the value of $PDR(s,p)$ is divided by the maximal inter-onset interval (IOI) between e_i and adjacent score events e_{i-1} and e_{i+1}, resulting in the following equation:

$$MAT(s,p) = ODR(p,e_i) + \frac{PDR(s,p)}{\max(IOI(e_i,e_{i-1}),IOI(e_{i+1},e_i))} \qquad (8)$$

The note-by-note matcher preserves the order of the notes in a given voice: thus, to be considered as a potential match for a score note in voice v, the onset of p must occur later than the onset of the last matched note in v. During each cycle, performance notes are matched to score notes in a sequential way, proceeding both forward (from the first score event to the last)

and backward (from the last score event to the first). For the initial cycles, solutions are obtained by pairing the forward and backward matches that show the highest cumulative match rating on the notes for which they are in agreement, retaining only the notes that are common to both matches. The resulting match is then passed on to the next cycle, and onset times are computed for both forward and backward passes using information from the previous cycle until a stable solution is reached. Then, a new series of cycles is conducted, taking the match with the highest cumulative match rating as the basis for the following cycle until a stable solution is reached (no distinction is made between backward and forward passes at this stage).

This order constraint is based on the observation that notes belonging to a melodic line are not likely to be played in a different order from that indicated in the score (Desain et al., 1997). Moreover, only performance notes played in the appropriate MIDI channel may be considered as candidates. For instance, a note played on the pedal on a MIDI organ cannot be considered as a potential match for a score note meant to be played on the manuals, even if it matches the pitch of that note.

In most cases, the matching process is unambiguous: only one performance note *p* fits all the requirements in terms of onset time, pitch, and MIDI channel, to be matched to a given score note *s*. However, in cases where performance errors, expressive timing deviations, or ornaments introduce deviations from the score, a selection procedure must take place to find the optimal fit between score and performance. In such instances, the note-by-note matcher prioritizes exact pitch matches. Thus, in a situation in which only one of the candidate performance notes has the same pitch as *s*, this note receives the highest possible rating regardless of its onset time difference. If there is no such exact pitch match, the candidates are ranked according to their match rating.

Once the entire piece has been matched, the best solution is selected as the one that maximizes the cumulative match rating, that is, the sum of the match ratings computed for every score note. Because these ratings take into account structural as well as temporal information, the best solution is not necessarily the one that matches the highest number of notes. A solution that matches fewer notes, but preserves the structural and temporal coherence of the piece to a greater extent, may be favoured over one that matches more notes, but ends up distorting the temporal structure.

2.5 Identification of performance errors and ornaments

The final phase of the matching procedure consists of the identification and categorization of performance errors and ornaments. The matcher identifies two general types of errors: *score errors* and *non-score errors.* Score errors comprise pitch errors (also called substitutions), omissions (including "added ties" – repeated notes in the score that were not re-attacked in performance), and timing errors, whereas non-score errors include all performance notes that are extraneous to the score, such as intrusions and repetitions (re-attacked notes in performance that were not repeated in the score).[1] The matcher codes errors in a parsimonious manner; that is, in cases where an error could be analyzed as one error or as two distinct errors, the matcher prefers a solution that minimizes the number of errors (Palmer & Van de Sande, 1993).

The distinction between score errors and non-score errors is relevant to the identification of ornaments. Indeed, whereas the interpretation of score errors is generally unambiguous, because a score error represents, by definition, the omission or misplaying of a single score note, all non-score errors correspond to unmatched performance notes, which may be theoretically

[1] "Untied" notes (Repp, 1996a) are treated as repetitions.

interpreted as ornaments. The problem of ornament identification can thus be recast as an interpretation of the status of unmatched performance notes. The approach privileged here is to assume that, by default, all unmatched performance notes are non-score errors, unless there is substantial evidence that one or more of these notes represent an ornament. In practice, for each unmatched performance note *u*, the matcher evaluates the likelihood that it belongs to an ornament; if this likelihood is superior to a threshold value, *u* is treated is an ornamental note; otherwise, it is categorized as a non-score error. However, in order to implement this procedure, a general definition of what a performance ornament is needs to be developed. In the following paragraphs, we will introduce some rules and present their implementation in the matching algorithm.

2.5.1 Formal definition of performance ornaments

Musically speaking, ornaments are often referred to as embellishments of a score note. In other words, each ornament can be said to be hierarchically subordinated to a score note in a representation of the musical structure (Lerdahl & Jackendoff, 1983; Schenker, 1987; Desain & Honing, 1992). In the musical realization of a score, this subordination is reflected in the fact that the ornamental notes must occupy the temporal and registral space of the score note that they intend to embellish: a trill occurring in bar 29 cannot normally be associated with a note in bar 14. However, although this concept of *score anchoring* is a necessary condition for a note to be considered an ornament of a score note, it is not a sufficient one: non-score performance errors may also occupy the temporal and registral space of a score note. Another fundamental property of ornamental notes is their *intentionality*: in contrast to random errors, ornaments generally form characteristic melodic figures, which may or may not represent typical patterns such as trills or

mordents. This intentionality may be captured by well-formedness rules, elaborated in Gestalt principles.

To be perceived as part of a single ornamental figure, the individual notes that constitute an ornament should be organized temporally and perceptually so as to form a single-stream percept (Bregman, 1990). According to the proximity principle, notes whose onsets and/or pitches are close to each other will tend to be perceived as being connected to each other. Moreover, the percept of a continuous, single melodic line is enhanced if the offset of a note is close to the onset of the following note, so that there are no interruptions in the melodic activity, and if there is a limited overlap between successive notes (Huron, 2001, pp. 12-13). The belongingness principle may also be applied to the case of ornamental notes that are separated from the score note they are embellishing by a large pitch interval, but which belong to the same chord or harmony, as is the case with certain appoggiaturas.

2.5.2 Implementation in the matcher

The matcher first determines, for each score note s_j, whether there are unmatched performance notes p_i that occupy the temporal and registral space of s_j. The temporal space occupied by s_j is bound by the onset of the immediately preceding note in the same voice and the onset of the following note in the same voice, while its registral space is bound by the pitches of score notes that sound together with s_j (Figure 1).[2]

[Insert Figure 1 around here]

If there are unmatched performance notes that fit these criteria, they may be considered as potential embellishments to s_j. These notes then receive *ornamental ratings* $ORN(p_i,s_j)$, which are determined according to the rules of proximity and belongingness outlined above.

[2] Note that, according to this definition, the registral space of a monophonic melody is unbound.

Specifically, the computation of $ORN(p_i,s_j)$ involves the pitch distance (in semitones) between successive unmatched performance notes $PTD(p_i,p_{i+1})$, as well as the inter-onset interval $IOI(p_i,p_{i+1})$ and offset-to-onset interval $OOI(p_i,p_{i+1})$ between successive unmatched performance notes, and the duration of score note s_j in the performance, estimated using the inter-onset $IOI(s_j,s_{j+1})$ between successive score notes (all durations are given in seconds). Ornamental ratings are also influenced by the number of notes involved in the potential embellishment: because unmatched performance notes are more likely to be heard as errors if they occur in isolation rather than forming a coherent group, the matcher assumes that the likelihood of a group of unmatched performance notes $p_1, \ldots, p_n$ being an ornament anchored to s_j increases with the size of the group n. Furthermore, ratings take score indications into account: unmatched performance notes are more likely to be treated as embellishments to s if there is an indication in the score that s should be ornamented in performance. Equations 9 to 12 provide a formal definition of the ornamental rating $ORN(p_i,s_j)$ for performance note p_i and score note s_j. It is composed of a pitch component P, an interonset component I, an offset-to-onset component O, and a constant γ which reflects whether there is an indication in the score to the effect that s_j should be ornamented in performance. Constants α and β are used to adjust the relative weights of P and I (respectively) in relation to O.

$$ORN(p_i, s_j) = 1 - (\alpha P + \beta I + O) + \gamma \qquad (9)$$

$$\text{where } P = \max\left[(0, (PTD(p_i, p_i(i+1)) - \delta)\right] \qquad (10)$$

$$I = \max\left(0, \frac{IOI(p_i, p_{i+1}) - \varepsilon}{\max(\varepsilon, IOI(s_j, s_{j+1}))}\right), \qquad (11)$$

$$\text{and } O = \max\left(0, \frac{OOI(p_i, p_{i+1}) - \theta}{\sqrt{(n-1)}}\right). \qquad (12)$$

Component P (Equation 10) is equal to 0 unless the pitch distance in semitones between two consecutive unmatched performance notes is larger than a threshold δ, which is set to 2 semitones in the current implementation of the matcher. This limit corresponds roughly to the fission boundary observed by Van Noorden (1977, Fig. 1) for tempi in the range of 2.5 to 7 notes per second. This has the effect of penalizing the rating of consecutive notes separated by more than 2 semitones, under the assumption that notes separated by a large registral distance are less likely to constitute an ornament. Similarly, component I is equal to 0 unless the interonset distance between consecutive unmatched performance notes is larger than a constant ε, which is set to 0.1 s. This interonset distance is divided by the greater of ε or the duration of score note s_j in performance, as determined by the interonset distance between s_j and s_{j+1}. Finally, component O is equal to 0 unless the offset-to-onset distance between consecutive unmatched performance notes is larger than θ, which is set to 0.05 s. The value of O also takes into account the number of unmatched performance notes that occupy the temporal and registral space of score note s_j, represented by n.

The evaluation of potential candidates is an iterative process. Ornamental ratings are first computed for all unmatched performance notes associated with a score note s. Notes whose ratings are below a threshold value are treated as errors and excluded from the list of potential candidates. However, because the exclusion of a note may affect the ratings of the remaining notes, ornamental ratings are computed again for all remaining notes, until a stable configuration is reached in which either all the candidates have ornamental ratings above the threshold value or

no viable candidates are left. A final selection process excludes groups of unmatched performance notes whose mean ornamental ratings are below a minimal threshold.[3]

In some instances, an ornament could be potentially anchored to two or more score notes. In these cases, an additional selection step is undertaken to assign the ornament to a single score note. This step uses a hierarchical forced-choice procedure that first prioritizes ornament-score note couplings that contain the greatest number of notes (thus minimizing the number of unmatched performance notes treated as errors), then couplings that maximize the temporal-registral fit between score note and ornament, and, as a last resort, couplings that maximize the mean ornamental rating of the embellishment.

Finally, ornaments are classified into appoggiaturas, mordents, trills, scalar patterns, and "unidentified ornaments". Because the approach outlined here does not rely on the recognition of specific patterns, the matcher may recognize that certain groups of unmatched performance notes possess all the characteristics of an ornament (such as pitch and time proximity, as well as melodic continuity), even if they do not form a typical ornamental pattern.

2.6 Comparison with other offline matchers

To conclude this section, a summary of the principal features of the three-step matcher is provided in Table 2, along with a comparison with a few well-known offline matchers. Besides the use of temporal information, one of the main differences between the three-step matcher and other matchers is that it processes performances first at the level of clusters before moving down to the note level. It thus combines the advantages of both approaches, taking into account both voice structure and the grouping of score notes into events.

[3] These threshold values were adjusted empirically so as to optimize the categorization of unmatched performance notes into errors and ornaments. Generally speaking, increasing these threshold values will increase the proportion of unmatched performance notes identified as errors relative to those identified as ornaments.

[Insert Table 2 around here]

3. Evaluation

In order to evaluate the accuracy of the matching algorithm, it is necessary to compare its solutions to those obtained using an independent reliable process. Score-performance matches realized by hand by the first author (a music theorist) on a corpus of 80 MIDI recordings of organ performances were used as ground truth data for this purpose. These recordings consisted of 48 performances of the *Premier Agnus* by Nicolas de Grigny (1672-1703) and 32 performances of *Wachet auf, ruft uns die Stimme* by Samuel Scheidt (1587-1654), for a total of 27,168 score notes. It should be noted that these matches, which we will refer to as *hand matches* (Heijink et al., 2000b), were completed before the programming of the three-step matcher was undertaken (reference removed to protect anonymity). In fact, the amount of work involved in the completion of these hand matches was a primary motivation in the design of this matcher.

In addition, we sought to assess the improvement in matching accuracy brought about by taking into account the temporal information from the MIDI recordings. One way to evaluate this effect would be to compare two matching algorithms that are identical in all respects, except that one uses temporal information and the other does not. To that end, we implemented a version of the three-step matcher that does *not* take into account temporal information (the second step of the matching procedure uses only the chronological succession of the score events), but is otherwise identical to the original algorithm, and compared the results obtained by this implementation to the hand matches.

In order to test its ability to cope with heavily ornamented performances, the three-step matcher was also used to match 32 performances of the Fugue in D minor (BWV 538), also known as the "Dorian" fugue, by J.S. Bach (1685-1750), for a total of 86,432 score notes.

Several ornaments are marked in the score of this piece, including trills occurring simultaneously in the pedal and in the manuals. However, given the length of the piece, the task of matching the 32 performances by hand would have been prohibitively time-consuming; thus, only a comparison between the matches produced by the temporal and non-temporal implementations of the three-step matcher is presented for this piece.

Finally, we conducted a direct comparison between the solutions obtained by the three-step matcher and the structure matcher (Desain et al., 1997) on a series of piano performances of excerpts from the *Etude* in C minor, Op. 10, No. 12, and the *Fantaisie Impromptu*, Op. 66, both by Fryderyk Chopin (1810-1849). The *Fantaisie* is especially challenging because it features a polyrhythmic relationship between the left and right hands (Heijink et al, 2000b). The structure matcher was selected as a benchmark because this algorithm was the one that performed best on this particular set of performances according to the results reported in Heijink et al. (2000b).

3.1 Method

The scores for *Premier Agnus* and *Wachet auf* were entered by hand, and voice information was included. The score of the Dorian fugue was prepared from a MIDI file obtained from an Internet archive ("Classical music archives", 1994). The scores of the *Etude* in C minor and the *Fantaisie Impromptu* were prepared from MIDI files kindly provided by Hank Heijink. The MIDI data were hand-edited for errors so that it would match exactly the score of the pieces. Voice information was added by hand. Scores were then set up in a format suitable for the matcher.

The matcher was implemented in the MATLAB programming language and run under Windows XP. In this configuration, the time required to match a single performance ranged from 10 to 60 seconds for *Premier Agnus, Wachet auf*, and the *Etude*, and from 15 minutes to one

hour for the Dorian fugue and the *Fantaisie*. By comparison, it took approximately one hour to match a single performance of the *Premier Agnus* or *Wachet auf* by hand.

3.2 Results

3.2.1 Comparison between hand matches, temporal matches, and non-temporal matches for Premier Agnus *and* Wachet auf

For each performance of *Premier Agnus* and *Wachet auf*, the solutions provided by both versions of the three-step matcher were compared to the hand matches and discrepancies between matches were identified (Table 3). For each implementation, the percentage of discrepancies with the human matches to the total amount of score notes was computed. We note that whereas a total of 25 discrepancies (out of 27,168 notes) were observed between the hand matches and the solutions obtained using the non-temporal version of the three-step matcher, only 6 discrepancies were identified between the hand matches and those produced by the temporal version of the matcher, a fourfold improvement. This result clearly demonstrates that the use of temporal information substantially improved the matching accuracy.

An inspection of the discrepancies revealed that most of the disagreements between the non-temporal matches and the hand matches of *Premier Agnus* and *Wachet auf* involved repeated notes and timing errors. As mentioned previously, repeated notes pose a challenge to offline matchers that do not use temporal information. Likewise, timing errors cannot be properly resolved in the absence of temporal information. However, these discrepancies disappeared when comparing the temporal matches to the hand matches. In fact, after examining the six remaining discrepancies, we favour the matcher's interpretation over the hand matches in three of those six cases.

[Insert Table 3 around here]

3.2.2 Analysis of the discrepancies between temporal matches and non-temporal matches for Premier Agnus*,* Wachet auf*, and the Dorian fugue*

Discrepancies were further analyzed by categorizing them into three groups: Type 1 discrepancies refer to performance notes matched to a different score note in each of the two solutions under comparison (see left column, Table 3); Type 2 discrepancies correspond to performance notes unmatched in one solution and matched to a score note in the other solution; and Type 3 discrepancies designate performance notes matched to the same score note in both solutions, but that are identified as score errors in one case and not in the other. Comparisons between the solutions produced by the temporal and non-temporal implementations of the matcher are also included for the performances of the Dorian fugue.

Whereas the majority of the discrepancies observed between the temporal and non-temporal implementations for *Premier Agnus* and *Wachet auf* belonged to Type 3, most of the discrepancies for the Dorian fugue were classified as Type 1. In contrast to the recordings of *Premier Agnus* and *Wachet auf*, which contained very few ornaments, the performances of the Dorian fugue were heavily ornamented: the temporal implementation of the matcher identified 7.5% of all performance notes as ornamental. Upon close inspection of the matches generated by the temporal version, the authors found themselves in perfect agreement with the solutions provided by the matcher in practically every case. It is especially noteworthy that the matcher could successfully discriminate between ornaments and non-score errors. However, the non-temporal implementation was not nearly as successful, as the presence of ornaments specifically hampered the accuracy of the matches in the sections that were most lavishly embellished. Thus, it is likely that the abundant ornamentation affected the non-temporal implementation to a greater extent than the temporal one. Indeed, 244 (55.6%) of the 439 discrepancies observed for the Dorian fugue involved a note identified as ornamental by one or both implementations.

Moreover, nearly all discrepancies involving an ornament (242 of 244) were classified as Type 1, which correspond to mismatched score notes. The prominence of ornamental notes is thus largely responsible for the percentage of mismatched notes between temporal and non-temporal versions of the three-step matcher being 6 times higher in the Dorian fugue than in the *Premier Agnus* and *Wachet auf*. These results suggest that the use of timing information in automated matching procedures is especially important in the case of ornamented performances.

3.2.3 Comparison between the three-step matcher and the structure matcher on performances of Chopin's Etude *in C minor and* Fantaisie Impromptu

In an attempt to provide an empirical basis for the evaluation of different matching algorithms, Heijink and colleagues (2000b) compared the solutions obtained by revised implementations of the strict matcher (Honing, 1990) and the Large matcher (Large, 1993), as well as an implementation of the structure matcher (Desain et al.,1997), to hand matches of five performances of Chopin's *Etude* in C minor and two performances of his *Fantaisie Impromptu.* These seven performances were obtained from Yamaha Disklavier discs that were widely available at the time. Given that the structure matcher was, by far, the most accurate algorithm on this particular dataset, it afforded a suitable benchmark with which to compare our implementation of the three-step matcher. Using the MIDI files and hand matches provided by Heijink et al., we ran the three-step matcher on the same dataset. However, because we found ourselves in disagreement with some of the hand matches proposed by Heijink et al., we decided to compare the solutions with our own hand matches (realized by the first author) in addition to the hand matches from Heijink et al.. Table 4 lists the discrepancies between the solutions obtained by the structure matcher and the three-step matcher for each of the seven performances of the original dataset used by Heijink et al.

[Insert Table 4 around here]

When using the hand matches provided by Heijink et al. as ground truth, both the structure matcher and the three-step matcher misinterpreted 10 notes out of 2,821 score notes (0.354%).[4] In contrast, when using our hand matches as ground truth, only three notes (0.106%) were misinterpreted by the structure matcher, and only one (0.035%) by the three-step matcher. The discrepancy between these results is likely tied to methodological differences in the error coding procedure: as explained in section 2.5, the three-step matcher codes errors in a parsimonious manner, and we followed the same procedure in our hand matches. This did not seem to be the case with Heijink et al.

It is noteworthy that the three-step matcher yields results that are comparable to the structure matcher even on performances of the *Fantaisie*, bearing in mind that the structure matcher, which treats parallel events in different voices as temporally independent events (see Introduction), was designed primarily for handling pieces that exhibit a considerable degree of independence between voices, such as the *Fantaisie*. More generally, these results point to the versatility of the three-step matcher, considering that very low error rates were achieved for both organ performances of Baroque music as well as piano performances of Romantic music.

4. Discussion

[4] Heijink et al. (2000b) reported a total of 8 misinterpreted notes out of 5,642 notes for the structure matcher, for an error rate of 0.1%. The figure of 5,642 notes is computed by adding the total number of notes in the performances to the total number of notes in the score. However, in analyzing the data provided by Heijink et al. using our methodology, we arrive at a total of 10 misinterpreted notes for the structure matcher. All analyses are available upon request.

We have presented an offline score-to-performance matching algorithm that relies both on structural and temporal information, allowing it to generate an accurate match even for heavily ornamented performances. A comparison with score-performance hand matches on a corpus of 80 MIDI recordings of organ performances showed a near-perfect agreement between the solutions found by the matcher and the hand matches. Indeed, if the hand matches are treated as ground-truth data, our algorithm achieved an accuracy of 99.978%, which corresponds to approximately 1 mismatched note for every 4,500 score notes. Similar results were observed on a set of piano performances of two pieces by Chopin, thus demonstrating the versatility and robustness of the approach introduced here. As noted by Heijink et al. (2000b, p. 551), the highest possible matching accuracy is required in the context of music performance research, which is the typical domain of application of offline matchers. Thus, we believe that the improvements presented here are non-negligible and make this matcher suitable for large-scale performance studies.

In addition, this matcher is designed to accommodate multi-channel MIDI recordings of performances from keyboard instruments with multiple manuals, such as organ or harpsichord; it was actually used to study performances of complex organ pieces such as J.S. Bach's "Dorian" fugue, as well as harpsichord pieces, in the context of performance research (references removed to protect anonymity). This feature makes it a potentially valuable tool for the investigation of ensemble performances of MIDI instruments.

We have also proposed a heuristic for the identification of ornaments and errors that is based on perceptual principles, and which could theoretically be amenable to empirical study. It is worth noting that the approach described here does not rely on the recognition of specific patterns, in contrast to the technique pioneered by Dannenberg and Mukaino (1988); instead, it

proceeds from a very general definition of performance ornaments to the identification of typical embellishment figures.

As this description of the ornament identification heuristic suggests, the accuracy of automatic matching algorithms could greatly benefit from implementing a model of basic perceptual principles of music cognition. Indeed, as noted by Desain et al. (1997), the fact that human listeners have no difficulty in matching scores to performances implies that modeling perceptual processes might help in resolving remaining challenges associated with score-performance matching. As an example, we may note that the matcher does not take into account scale and chord structure in its current implementation. For instance, a series of notes that constitute an E major arpeggio are all part of the same harmony. They will be perceived as more similar to each other by a human listener familiar with this musical style than other notes which do not belong to the E major chord. Applying this to the analysis of performance errors, a B might be a more likely substitution error for a G# in the context of an E major arpeggio than an A#, even though the pitch interval between G# and A# is smaller than that between B and G#. However, our algorithm is insensitive to the notion of harmonic context. Moreover, the pitch distance rating used by the matcher is a simple measure of the interval in semitones between two notes.

The implementation of a hierarchical pitch-space model such as that proposed by Lerdahl (2001) might allow the matcher to arrive at more accurate solutions for tonal excerpts. Although this model is style-specific and could prove irrelevant, if not detrimental, to the processing of atonal music or music from non-Western styles, we nevertheless believe that the accuracy of matching algorithms would greatly benefit from the integration of concepts such as scale and chord structure, and perhaps of notions such as consonance and dissonance. While pointing out

the limitations of current algorithms, these suggestions underline the importance of issues related to the representation of musical similarity and to the larger question of the modeling of musical intelligence in the development of more effective matching paradigms.

Acknowledgements

This research was supported by fellowships (removed to protect identify of authors) to Author 1, as well as a grant (removed to protect identity of authors) to Author 2.

References

Bregman, A. S. (1990). *Auditory scene analysis: The perceptual organization of sound*. Cambridge, Mass.: MIT Press.

Classical music archives. (1994). Retrieved January 5, 2007, from http://www.classicalarchives.com/

Dannenberg, R. B. (1984). An on-line algorithm for real-time accompaniment. In *Proceedings of the 1988 International Computer Music Conference* (pp. 243-249). San Francisco: ICMA.

Dannenberg, R. B., & Mukaino, H. (1988). New techniques for enhanced quality of computer accompaniment. In *Proceedings of the 1988 International Computer Music Conference* (pp. 243-249). San Francisco: ICMA.

Desain, P., & Honing, H. (1992). *Music, mind, and machine: Studies in computer music, music cognition, and artificial intelligence*. Amsterdam: Thesis Publishers.

Desain, P., & Honing, H. (1993). Tempo curves considered harmful. *Contemporary music review, 7*(2), 123-138.

Desain, P., Honing, H., & Heijink, H. (1997). Robust score-performance matching: Taking advantage of structural information. In *Proceedings of the 1997 International Computer Music Conference* (pp. 337-340). San Francisco: ICMA.

Dixon, S. (2005). MATCH: A music alignment tool chest. In *6th International Conference on Music Information Retrieval* (pp. 492-497). London, UK.

Goebl, W. (2001). Melody lead in piano performance: Expressive device or artifact? *Journal of the Acoustical Society of America, 110*(1), 563-572.

Gotoh, O. (1982). An improved algorithm for matching biological sequences. *JournalMolecular Biology, 162*(3), 705-708.

Heijink, H. (1996). *Matching scores and performances.* Unpublished master's thesis, Nijmegen University.

Heijink, H., Desain, P., Honing, H., & Windsor, L. (2000a). Make me a match: An evaluation of different approaches to score-performance matching. *Computer Music Journal, 24*(1), 43-56.

Heijink, H., Windsor, L., & Desain, P. (2000b). Data processing in music performance research: Using structural information to improve score-performance matching. *Behavior Research Methods Instruments & Computers, 32*(4), 546-554.

Honing, H. (1990). POCO: An environment for analyzing, modifying, and generating expression in music. In *Proceedings of the International Computer Music Conference* (pp. 364-368). San Francisco: ICMA.

Hoshishiba, T., Horiguchi, S., & Fujinaga, I. (1996). Study of expression and individuality in music performance using normative data derived from MIDI recordings of piano music. In *Proceedings of the 4th International Conference on Music Perception and Cognition* (pp. 465-470). Montreal: McGill University, Faculty of Music.

Huron, D. (2001). Tone and voice: A derivation of the rules of voice-leading from perceptual principles. *Music Perception, 19*(1), 1-64.

Kendall, R. A., & Carterette, E. C. (1990). The communication of musical expression. *Music Perception, 8*(2), 129-164.

Large, E. W. (1993). Dynamic programming for the analysis of serial behaviors. *Behavior Research Methods Instruments & Computers, 25*(2), 238-241.

Lerdahl, F. (2001). *Tonal pitch space*. Oxford; New York: Oxford University Press.

Lerdahl, F., & Jackendoff, R. (1983). *A generative theory of tonal music*. Cambridge, Mass.: MIT Press.

Mongeau, M., & Sankoff, D. (1990). Comparison of musical sequences. *Computers and the Humanities, 24*(3), 161-175.

Needleman, S. B., & Wunsch, C. D. (1970). A general method applicable to search for similarities in amino acid sequence of 2 proteins. *Journal of Molecular Biology, 48*(3), 443-453.

Palmer, C. (1989). Mapping musical thought to musical performance. *Journal of Experimental Psychology-Human Perception and Performance, 15*(12), 331-346.

Palmer, C. (1996). On the assignment of structure in music performance. *Music Perception, 14*(1), 23-56.

Palmer, C. (1997). Music performance. *Annual Review of Psychology, 48*, 115-138.

Palmer, C., & Van de Sande, C. (1993). Units of knowledge in music performance. *Journal of Experimental Psychology-Learning Memory and Cognition, 19*(2), 457-470.

Palmer, C., & Van de Sande, C. (1995). Range of planning in music performance. *Journal of Experimental Psychology-Human Perception and Performance, 21*(5), 947-962.

Pardo, B., & Birmingham, W. (2001). Following a musical performance from a partially specified score, *Multimedia Technology Applications Conference*. Irvine, California.

Puckette, M., & Lippe, C. (1992). Score following in practice. In *Proceedings of the 1992 International Computer Music Conference* (pp. 182-185). San Francisco: ICMA.

Raphael, C. (2006). Aligning music scores with symbolic scores using a hybrid graphical model. *Machine Learning, 65*, 389-409.

Rasch, R. A. (1979). Synchronization in performed ensemble music. *Acustica, 43*(2), 121-131.

Repp, B. H. (1996a). The art of inaccuracy: Why pianists' errors are difficult to hear. *Music Perception, 14*(2), 161-183.

Repp, B. H. (1996b). Patterns of note onset asynchronies in expressive piano performance. *Journal of the Acoustical Society of America, 100*(6), 3917-3931.

Roads, C. (1996). *The computer music tutorial*. Cambridge, MA: MIT Press.

Schenker, H. (1987). *Counterpoint: A translation of Kontrapunkt* (J. Rothgeb, Trans.). New York, London: Schirmer Books; Collier Macmillan.

Schwarz, D., Orio, N., & Schnell, N. (2004). Robust polyphonic Midi score following with Hidden Markov Models. In *Proceedings of the International Computer Music Conference (ICMC)* (pp. 442-445). Miami, Florida: International Computer Music Association.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal, 27*, 379-423, 623-656.

Van Noorden, L. P. A. S. (1977). Minimum differences of level and frequency for perceptual fission of tone sequences ABAB. *Journal of the Acoustical Society of America*, *61*(4), 1041-1045.

Vantomme, J. D. (1995). Score following by temporal pattern. *Computer Music Journal, 19*(3), 50-59.

Table 1:

Score events

Performance clusters	1	2	3	4	5	6	7	8
1	100	0	0	25	0	0	0	15.625
2	0	100	0	0	0	81.25	56.25	25
3	0	0	100	0	37.5	0	0	25
4	25	0	0	100	0	25	0	50
5	0	0	100	0	37.5	0	0	25
6	0	0	62.5	25	50	0	0	0
7	0	0	37.5	0	100	0	0	0
8	0	81.25	0	25	0	100	50	25
9	0	56.25	0	0	0	50	100	0
10	15.625	25	25	50	0	25	0	100

Table 2:

	Strict matcher (Honing, 1990)	Large matcher (Large, 1993)	Structure matcher (Desain et al., 1997)	Three-step matcher
Processing unit	Note	Cluster / event	Note	Cluster / event (steps 1 & 2); note (step 3)
Uses voice information	No	No	Yes	Yes
Uses temporal information	No†	No†	No	Yes
Solutions considered	One	Several	Several	Several
Definition of best solution	Most matched notes	Most matched notes	Most matched notes, preserves voice structure	Best structural / temporal fit for events (steps 1 & 2) and for notes (step 3)

† The strict matcher and the Large matcher use a fixed maximum inter-onset interval for the parsing of performance notes into clusters.

Table 3:

		Premier Agnus (15360 notes)	*Wachet auf* (11808 notes)	Dorian fugue (86432 notes)
Hand matches/ temporal matcher				
	Type 1	0	0	
	Type 2	1	0	N/A
	Type 3	2	3	
	Total	3 (0.020%)	3 (0.025%)	
Hand matches / non-temporal matcher				
	Type 1	0	3	
	Type 2	1	4	N/A
	Type 3	13	4	
	Total	14 (0.091%)	11 (0.093%)	
Non-temporal matcher / temporal matcher				
	Type 1	0	3	295
	Type 2	0	2	49
	Type 3	11	5	95
	Total	11 (0.072%)	9 (0.077%)	439 (0.508%)

Note. Percentages refer to the number of discrepancies relative to the total number of score notes analyzed for each piece.

Table 4:

Piece	Disc No./ Track	Structure matcher/ HM Heijink et al.(2000b)	Structure matcher/ HM x & y	3-step matcher/ HM Heijink et al.(2000b)	3-step matcher/ HM x & y
Etude	YMM 900202/2	1	0	1	0
Etude	YMM 900148/12	0	0	0	0
Etude	YPA 1069E/1	1	1	0	0
Etude	YPA 1070E/27	2	2	3	1
Etude	YPA 1100E/8	0	0	0	0
Fantaisie	YPA 1100E/5	3	0	3	0
Fantaisie	YPA 1077E/3	3	0	3	0
Total		**10 (0.354%)**	**3 (0.106%)**	**10 (0.354%)**	**1 (0.035%)**

Note. Etude refers to the *Etude* in C minor, Op. 10, No. 12; *Fantaisie* refers to the *Fantaisie Impromptu*, Op.66 (both by Fryderyk Chopin). The performances were distributed on floppy discs from Yamaha Music Corp. HM: hand matches. Percentages are obtained by dividing the total number of discrepancies by the total number of score notes (2821).

Legends for tables and figures:

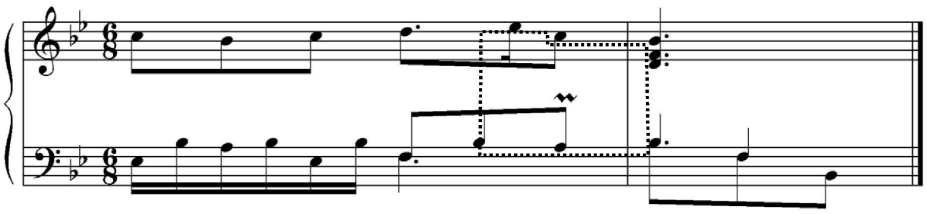
Table 1. Structural ratings for performance clusters / score events pairings. Highlighted cells correspond to perfectly matched pairings.

Table 2. Comparison between the three-step matcher and other matchers.

Table 3. Distribution of the discrepancies observed between different matching methods.

Table 4. Discrepancies observed between the structure matcher and the three-step matcher with hand matches from Heijink et al. and x & y.

Figure 1. Registral and temporal space associated with a given score note. The area bound by the dashed line represents the registral and temporal space for the ornamented note (indicated by the mordent sign •) in this excerpt from Couperin's *Les Bergeries*.



167x48mm (600 x 600 DPI)