ACOUSTIC SCORE FOLLOWING TO MUSICAL PERFORMANCE WITH ERRORS AND ARBITRARY REPEATS AND SKIPS FOR AUTOMATIC ACCOMPANIMENT

Tomohiko Nakamura, Eita Nakamura[†] and Shigeki Sagayama[†]
Graduate School of Information Science and Technology, The University of Tokyo
7-3-1, Hongo, Bunkyo-ku, Tokyo, 113-8656, Japan
{nakamura, enakamura, sagayama}@hil.t.u-tokyo.ac.jp

ABSTRACT

We discuss acoustic score-following algorithms for monophonic musical performances with arbitrary repeats and skips as well as performance errors, particularly focusing on reducing the computational complexity. Repeats/skips are often made arbitrarily during musical practice, and it is desirable to deal with arbitrary repeats/skips for wide application of score following. Allowing arbitrary repeats/skips in performance models demands reducing the computational complexity for score following. We show that for certain hidden Markov models which assume independence of transition probabilities from and to where repeats/skips are made, the computational complexity can be reduced from $O(M^2)$ down to $O(M)$ for the number of notes M, and construct score-following algorithms based on the models. We experimentally show that the proposed algorithms work in real time with practical scores (up to about 10000 notes) and can catch up with the performances in around 3.8 s after repeats/skips.

1. INTRODUCTION

Audio score following is the real-time alignment of acoustic signal of musical performance to the performance score, and has wide application such as automatic accompaniment, automatic score page turning and automatic captioning to music videos. It is particularly essential for automatic accompaniment, which synchronizes the accompaniment automatically to human performances in real time and helps music performers and lovers practice ensemble music by themselves.

Human performances have tempo fluctuation due to performers' physical limitation and their expression of musical ideas. Musical performers, both amateurs and professionals, also make performance errors such as pitch errors and note insertions and deletions. In addition to these, acoustic signals of musical performances are of complex nature because of possible noise and acoustic variation of musical instruments. According to these features of human performances and their acoustic signals, score following is a challenging task in musical signal processing and has been a field of research since [1, 2] and further explored in [3–12] (see [13] for a review).

Particularly during music practice, performers often repeat and/or skip sections for correcting errors or for practicing specific sections again and again, and it is desirable to handle such repeats/skips for application of score following in practical situations. In [5, 6, 12], score following algorithms allowing repeats/skips from and to specific score positions were studied. Although there are performers' tendencies on from and to where repeats/skips occur, estimation of the specific score positions is generically difficult, especially in practical situations where scores are prepared in musical instrument digital interface (MIDI) data or performances by various performers are necessary to be dealt with. Therefore it is attractive to have score following algorithms which can handle arbitrary repeats/skips from and to any score positions.

Allowing arbitrary repeats/skips leads to a large search space and results in two problems: (i) large computational complexity and (ii) a risk of lowering score-following accuracy. As we later discuss in detail, simply-generalized versions of algorithms in [3, 5, 6] are difficult to work in real time for practical scores with $O(1000)$ to $O(10000)$ notes,[1] and it is unavoidable to reduce the computational complexity.

Statistical approach to score following has advantages in handling acoustic variation of musical performances and was used in many previous works [13]. In this approach, one can either estimate the score position first and the tempo [3, 4], or estimate simultaneously the score position and the tempo [9, 10, 12]. Since the search space is too large in the latter case when dealing with arbitrary repeats/skips, we adopt the former method.

In the following, we discuss certain hidden Markov models (HMMs) for musical performance, which explicitly models performance errors and arbitrary repeats/skips. We show, when assuming independence of transition probabilities from and to where repeats/skips are made, the computational complexity can be reduced significantly, enabling us to construct acoustic score-following algorithms which handle arbitrary repeats/skips and work in real time. We experimentally evaluate the performance of the proposed algorithms for human performances in practice and also

[†] Presently with National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, 101-8430, Japan

[1] For example, there are around 1900 notes in the first movement of the clarinet part in Mozart's Clarinet Quintet.

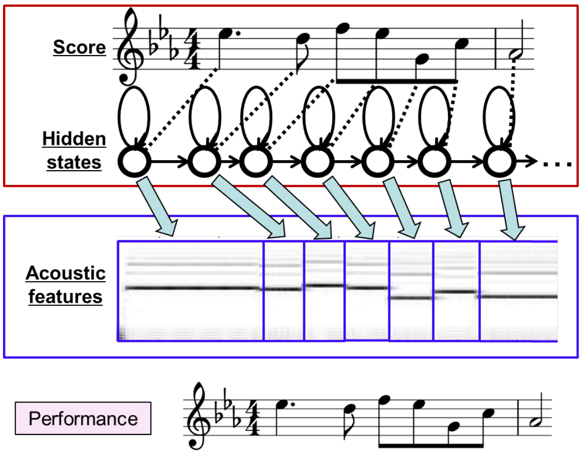


Figure 1. The performance HMM consists of states corresponding to notes, and the state emits acoustic features of the performed note.

examine whether there is any significant lowering of score-following accuracy. We confine ourselves to monophonic performances for the sake of simplicity.

2. HMM-BASED PERFORMANCE MODEL

2.1 HMM for Score Following

We regard score following as an inverse problem of estimating score positions from acoustic signals by modeling human performances. The human performance without errors and repeats/skips can be seen as a process of making a transition to the next note, and emitting an acoustic feature of the performed note. By associating the notes on score with hidden states, the performance is also interpreted as a state transition sequence. The performance often includes changes in tempos and note durations because of physical limitation and musical expression, and acoustic signals of the performance include noise and acoustic variations. These state transitions and emission of acoustic features are described as a stochastic process [3]. Assuming that the transitions depend only on the current state, the performance is represented by an HMM as shown in Fig. 1.

A performance with insertion/deletion errors are also described by an HMM [3]. Insertion is represented by a self transition and deletion is represented by a transition to the state after the next as shown in Fig. 2. These are described as

$$A_{i,i} = a_i + (1 - a_i)A_i^{(\mathrm{ins})}, \; A_{i,i+2} = (1 - a_i)A_i^{(\mathrm{del})}. \quad (1)$$

Here, $\{A_{i,j}\}_{i,j=1}^{M}$ is the state-transition probability matrix, and the durational self-transition probability a_i is determined by matching the expected staying time with the duration d_i of the i-th note, which yields

$$d_i = \sum_{k=1}^{\infty} k a_i^{k-1}(1 - a_i) = \frac{1}{1 - a_i}. \quad (2)$$

These errors are expressed as transitions to neighboring states and the HMM topology is left-to-right.

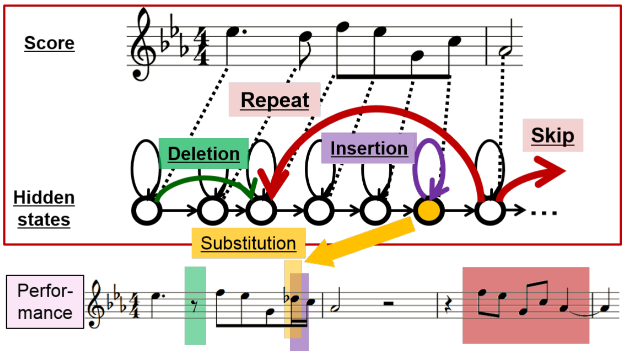


Figure 2. Representation of errors and repeats/skips in the performance model. Deletion (green arrows) is represented by a transition to the state after the next. Insertion (purple arrows) is described as a self transition, and substitution (orange objects) is represented by emission of CQF spectrum of incorrect pitch. Repeat/skip is expressed as a transition to a remote state (red arrows).

2.2 Feature Extraction from Acoustic Signal

The variation in acoustic signals of the performance is large even within the same pitch. For score following, therefore, features are preferred to be sensitive to pitch information and less sensitive to timbre and volume. As stated in [8], this requirement is matched by the normalized output of constant-Q filters (CQFs) with central frequencies at semitone intervals (CQF spectrum). For shorter calculation time, the CQF spectrum was calculated with a fast frame-wise algorithm [14]. Since a spectrum changes significantly at the onset time and is otherwise stationary, spectral flux is employed to distinguish successive notes of the same pitch [15].

2.3 Emission Probability

As shown in Fig. 2, substitution is represented by emission of CQF spectrum of incorrect pitch, and the corresponding probability is described as a mixture weight of a Gaussian mixture model for emission probability. The emission probability $b_i(y_t)$ at the i-th state of a CQF spectrum y_t at time t is thus

$$b_i(y_t) = \sum_{k \in \mathcal{K}} \omega_k(i) \mathcal{N}(y_t | \mu_k, \Sigma_k) \quad (3)$$

where $\mathcal{N}(\cdot|\mu_k, \Sigma_k)$ denotes a multidimensional normal distribution with mean μ_k and covariance matrix Σ_k, $\mathcal{K}$ is the set of all pitches, and $\omega_k(i)$ stands for the mixture weight.

3. MODELING OF ARBITRARY REPEATS/SKIPS AND THE COMPUTATIONAL COMPLEXITY

3.1 Topology of the Performance HMM

As discussed in Sec. 2, a performance with insertion/deletion/substitution errors is represented by left-to-right transitions to neighboring states and emission of acoustic features of incorrect pitch. On the contrary, repeats/skips from and to arbitrary notes are represented by transitions from each state to all the states, including remote ones (two examples are shown in Fig. 2.). Therefore, the topology of the performance model with arbitrary repeats/skips, which generalizes the models in [3, 5, 6], is complex, resulting in a large search space.

3.2 Computational Complexity of Score Following

The score position is estimated by calculating the most probable state given the CQF spectrums up to the time of estimation. In equations,

$$\hat{s}_t = \operatorname*{argmax}_{s_t} p(s_t|y_{1:t}) = \operatorname*{argmax}_{s_t} p(y_{1:t}, s_t) \tag{4}$$

where s_t and $\hat{s}_t$ denote the state random variable at time t and its estimated value, and $y_{1:t}=\{y_\tau\}_{\tau=1}^{t}$ stands for the CQF spectrum sequence. The second equation is derived from the Bayes' theorem.

(4) can be solved by applying the online forward algorithm, and its update rule is described as

$$\alpha_t(i) = b_i(y_t)\sum_{j=1}^{M}\alpha_{t-1}(j)A_{j,i} \tag{5}$$

where $\alpha_t(i){:=}p(y_{1:t}, s_t{=}i)$ is the forward variable. Here, the initial value $\alpha_1(i){=}b_i(y_1)\pi_i$ is calculated with the initial distribution π_i. (5) indicates that the computational complexity for score following is $O(M^2)$ since there are M summations over M states. As shown in Sec. 4, the $O(M^2)$ complexity is too large for the score follower to work in real time for scores with a number of notes larger than a few hundreds, and therefore it is crucial to reduce the complexity for processing practical scores.

3.3 Algorithms for Reducing Computational Complexity

In order to reduce the computational complexity, some constraints on the state-transition probability matrix are necessary. In this section, we propose two models and algorithms reduced the complexity to linear orders.

Human performers probably perform with their tendencies of pausing before repeats/skips and resuming after them. We can represent the tendencies at each state as the probabilities of pausing and resuming, or C_j and D_i. The distribution of where human performers resume is also probably dependent on where they pause. However, allowing the dependence results in $O(M^2)$ computational complexity as shown in Sec. 3.2, and thus we assume that the distribution of where human performers resume is independent of where they pause. With this assumption, the transition matrix can be written as

$$A_{j,i} = B_{j,i} + C_j D_i \tag{6}$$

where $B_{j,i}$ is a band matrix with bandwidth three representing the straight performance and deletion/insertion errors. Note that the normalization conditions $\sum_i A_{j,i}{=}1$ and $\sum_i D_i{=}1$ yield $C_j{=}1-\sum_i B_{j,i}$.

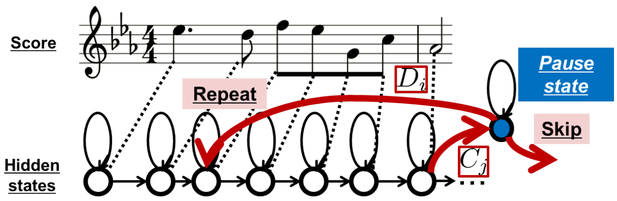


Figure 3. Representation of repeats/skips in the proposed performance model with the pause state (a blue disk) corresponding to pause sections at repeats/skips. Those are expressed as two-step transitions via the pause state (red arrows).

Substituting (6) into (5), we have

$$\alpha_t(i) = b_i(y_t)\left[\sum_{j=i-2}^{i}\alpha_{t-1}(j)B_{j,i} + \Big(\sum_{j=1}^{M}\alpha_{t-1}(j)C_j\Big)D_i\right]. \tag{7}$$

Since the sum in parentheses in the second term on the right-hand side is independent of i, it is sufficient to calculate this once at each estimation. The computational complexity of the sum is $O(M)$ and that of the rest of (7) is $O(M)$. Thus, we can reduce the computational complexity required for the estimation from $O(M^2)$ down to $O(M)$.

We obtain a similar model by focusing on a silent pause which is often made at repeats/skips before resuming performance. Such a pause can be represented by an additional state (the pause state). Since the repeats/skips are described as two-step transitions via the pause state as shown in Fig. 3, the tendencies of pausing and resuming the performances can be expressed as the transitions probabilities to the pause state and those from the pause state. In equations, the transition matrix of the model is

$$\tilde{A}_{j,i} = B_{j,i},\ \tilde{A}_{j,N} = C_j,\ \tilde{A}_{N,i} = (1-\tilde{A}_{N,N})D_i \tag{8}$$

for $i, j \in [1, M]$ where the N-th state is the pause state and $N{=}M+1$.

Naively, the computational complexity for updating the forward variable in the model is $O(N^2)$. However, since the transition probabilities to the note states except for those from neighboring notes and the pause state are zero, the complexity for updating the forward variable for the note states is reduced to $O(M)$. For the pause state, we must deal with transitions from all the states, and the complexity for calculating its forward variable is $O(N)$. Therefore, the overall computational complexity is reduced to $O(N) \simeq O(M)$.

While the above discussion of computational complexity is based on the forward algorithm, a similar discussion is valid for the Viterbi algorithm. With a slight modification, the discussion can also be generalized for Mealy-type emission probabilities of the form similar to $A_{j,i}$ in (6).

3.4 Comparison of the Two Models

The two models discussed in the previous section has a similar structure as seen in (6) and (8). In both models, one can describe tendencies of performance on the distributions of notes to which repeats/skips are made. Both the models rely on the independence of the distribution from the notes before them. The difference is the explicit modeling of the pause state in the latter model. In actual performances, silent pauses at repeats/skips often exist and their duration is long to some extent. Therefore, the latter model is expected to be more suited for score following. However, since quantitative comparison of both the models is difficult, we provide experiments for evaluating the performances of the models in Sec. 4.

4. EVALUATION OF COMPUTATIONAL COMPLEXITY AND SCORE-FOLLOWING PERFORMANCE

4.1 Experimental Conditions

4.1.1 Overall Conditions

To evaluate our algorithms, we conducted three experiments. The first experiment examines quantitatively whether the proposed algorithms works in real time with the practical scores, the second one evaluates the performance of the proposed algorithms in following repeats/skips, and the third one evaluates score-following accuracy and examines whether there is a lowering of accuracy in modeling arbitrary repeats/skips for performances without repeats/skips.

In all the experiments, we used acoustic signals of monophonic performances at 16 kHz sampling rate and the scores were prepared in MIDI format. CQF spectrums were extracted by using 128 ms frames with a 20 ms hop-size, and the emission probabilities of the performance models were trained by clarinet performances in RWC musical instrument sound database [16]. The parameters of the proposed algorithm without the pause state were set as $\boldsymbol{\pi}=[1,0,0,\cdots,0]^{\top}$, $A_i^{(\mathrm{ins})}=A_i^{(\mathrm{del})}=\exp(-500)$, $C_i=\exp(-1000)$, and $D_i=1/M$ for $i\in[1,M]$. For the other proposed algorithm, the parameters were set as $\tilde{A}_{N,N}=0.98$ in addition to the above. The probabilities of making errors of semitone, whole tone and perfect 12th were set as 0.001, 0.001, and 0.0001, respectively.

4.1.2 Condition on the First Experiment

Since the computational complexity mainly depends on the number of notes, and not on pitches and durations, artificially prepared scores with various numbers of notes were used in the first experiment. The machine had an Intel Core 2 Duo P9400 2.40 GHz with 6 MB of cache and 2 GB of RAM, and the operating system was Ubuntu 12.04LTS. The evaluation measure was the real time factor (RTF) defined as the ratio of the processing time and the hop-size, which is less than one if and only if the algorithms work in real time.

4.1.3 Condition on the Second Experiment

In the second experiment, for evaluating the score-following performance under practical situations, we used acoustic signals of 14 recorded performances (total 1687 s) by an amateur clarinet performer during his musical practice. Seven different songs were performed including classical and popular music pieces and nursery rhymes, partially from RWC music database [16]. 43 repeats/skips and 45 insertion/deletion/substitution errors were made naturally in the performances, and the ranges of repeats/skips were distributed from 0.1 s to 85 s in score time (0 bars to 43 bars). The performer did not waited the score follower's catching up with his performance. As evaluation measures, the detection rate of repeats/skips and the following time were employed. The following time is defined as the time interval (in units of seconds and notes) between the repeat or skip and the time when the score follower caught up with the performance within a range of Δ ms.

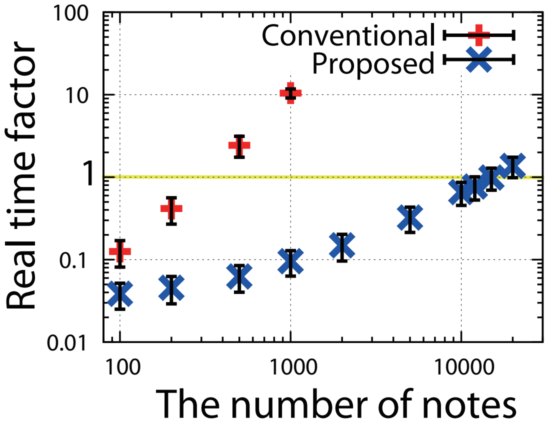


Figure 4. Real time factor (RTF) and its standard deviation of score following with the various number of notes in the performance score. The red points represent RTFs of the conventional algorithm, and the blue ones represent RTFs of the proposed algorithm with the pause state.

We compared the proposed algorithms with the algorithm without modeling of repeats/skips which corresponds to the previous work [3]. While Cano *et al.* used slightly different acoustic features of pitch and energy, CQF spectrums were employed as acoustic features in this experiment. The difference does not result in lowering the score-following accuracy, and rather improves it as stated in [8], and we believe that our choice of the acoustic features is adequate.

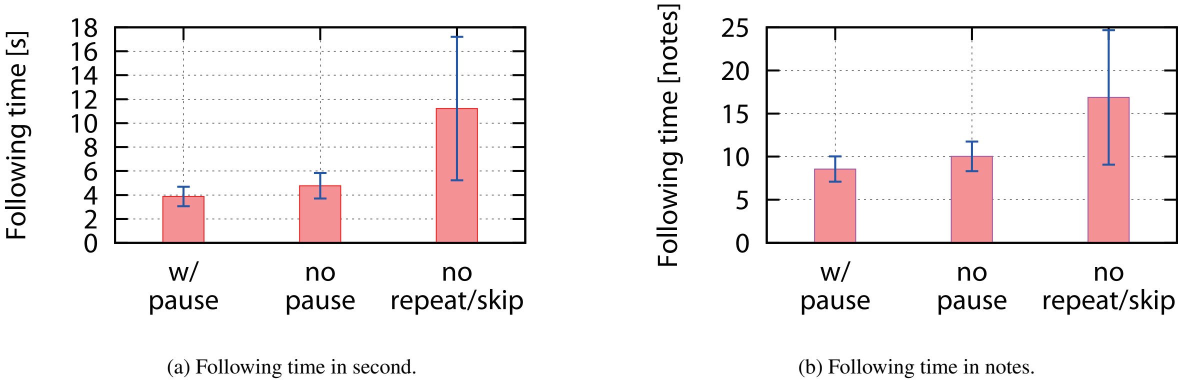
4.1.4 Condition on the Third Experiment

In the third experiment, a sufficient amount of real performances could not be prepared, and we used monophonic acoustic signals converted from MIDI signals. For the MIDI signals, the melody parts of 112 popular music pieces and royalty-free ones without repeats/skips in RWC music database were employed [16]. Evaluation measures were the piecewise precision rate and the overall precision rate used in the MIREX contest [17]. The piecewise precision rate (PPR) is the average of detection rates of notes in each piece, and the overall precision rate (OPR) is the detection rate of notes in all pieces.

4.2 Results and Discussions

4.2.1 First Experiment

The result of the first experiment is shown in Fig. 4, where the RTF was averaged over 95 calculations for each con-



(a) Following time in second.

(b) Following time in notes.

Figure 5. Following time (both (a) in second and (b) in notes) of the proposed algorithm with the pause state (w/ pause), that without the pause state (no pause) and the conventional algorithm (no repeat/skip) in left-to-right fashion.

Evaluation Measure	w/ pause	no pause	no repeat/skip
Detection rate of repeats/skips	$32/43$	$29/43$	$8/43$

Table 1. Detection rate of repeats/skips by the proposed algorithm with the pause state (w/ pause), that without the pause state (no pause) and the conventional algorithm (no repeat/skip).

dition. Only the result of the proposed algorithm with the pause state is shown, since the result was similar for the other. The figure shows that in the proposed algorithm, the RTF increases asymptotically in proportion to M and, in the conventional algorithm, asymptotically in proportion to M^2, which is consistent with the theoretical result in Section 3.3. The result shows that the score follower worked in real time on the computer up to around 10000 notes, and the conventional one up to around 300 notes. The conventional algorithm is difficult to handle the practical scores with over $O(100)$ notes, and for those with 10000 notes, the computation time is around 2 s, or ten times the hop-size. On the other hand, the computational time is reduced to around 0.02 s, or one hundredth, by the proposed algorithms, and almost all the practical scores can be used. Although the detail of the upper bound of the number of notes for real-time working may be changed on other computers because of difference in processing power, the reduction of the computational complexity by the proposed algorithms always remains effective.

4.2.2 Second Experiment

In the second experiment, the algorithm with the pause state detected 32 repeats/skips of 43, and its following time was 3.9 ± 0.8 s (8.0 ± 1.5 notes) for Δ=500 ms as shown in Fig. 5 and Table 1. On the other hand, the algorithm without the pause state detected 29 repeats/skips, and its following time was 4.9 ± 1.0 s (10 ± 2 notes) for Δ=500 ms. As we conjectured in Sec. 3.4, the proposed algorithm without the pause state followed repeats/skips later than that with the pause state. In contrast to those algorithms, the conventional algorithm corresponding to the one in [3] detected only eight repeats/skips, and followed those with 11 ± 3 s and 17 ± 8 notes delay. It is obvious that modeling repeats/skips significantly improves the performance in following repeats/skips.

The algorithm with the pause state had 11 undetected repeats/skips. Some of the undetected repeats/skips were caused by the existence of similar sections and phrases such as choruses in popular music. Others happened in the cases where only a few notes were performed between the repeats/skips. Such scores and performances are generally difficult to follow both for computers and humans. Because human accompanists would need comparable following time, the proposed algorithms are applicable to practical use.

4.2.3 Third Experiment

In the third experiment, all the PPRs were 0.839 ± 0.009, the OPRs by the algorithms except that without the pause state were $30073/36051$ and the other was $30070/36051$. There were only the slight difference between the proposed algorithms and the conventional one in PPR and OPR, and this result shows that the modeling of repeats/skips did not lower the accuracy significantly.

4.3 Implementation to Automatic Accompaniment

We also implemented the proposed score-following algorithms to automatic accompaniment. As an accompaniment playback module, a tempo estimation [18] and a playback speed conversion of acoustic signals of accompaniment [19] were employed. Fig. 6 shows the accompaniment result to the performances with repeats by the algorithm with the pause state, and videos for such performances are available at `http://hil.t.u-tokyo.ac.jp/~nakamura/demo/automatic_accompaniment.html`.

5. CONCLUSION

We have proposed two score-following algorithms for monophonic performances with both insertion/deletion/substitution errors and arbitrary repeats/skips. (i) Assuming the independence of transition

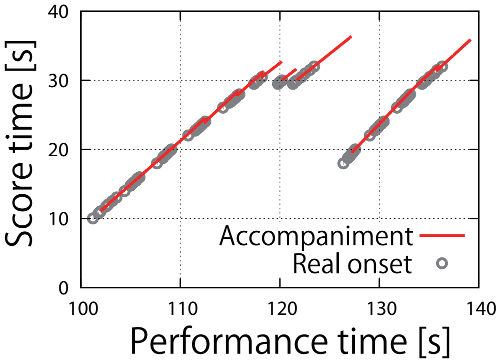


Figure 6. The automatic accompaniment result for a human performance with repeats by the algorithm with the pause state. The gray circle expresses a real onset, and the red line represents the played accompaniment.

probabilities from and to where repeats/skips are made, we have shown that the computational complexity is reduced from $O(M^2)$ down to $O(M)$. (ii) Focusing on a silent pause which are often made at repeats/skips before resuming performance, we have revealed that the computational complexity is also reduced down to $O(M)$ by explicit modeling of the existence of the pause. We have experimentally shown that the proposed algorithms work in real time for the practical scores up to 10000 notes and can catch up with performances in around 3.8 s after repeats/skips. The experiment has indicated that there is not a significant lowering of the score-following accuracy originating in modeling arbitrary repeats/skips.

As future works, an extension to polyphonic music is important to enable the score followers to process more scores and performances by other instruments as discussed in [7, 17]. Using tempo information is important to improve the performance of the algorithms and to help us to use beat information as discussed in [9–11].

Acknowledgments

We thank Naoya Ito for playing the clarinet, and Hirokazu Kameoka for useful discussions. This research has been partly supported by the Grant-in-Aid from JSPS (#23240021).

6. REFERENCES

[1] R. Dannenberg, "An on-line algorithm for real-time accompaniment," in *Proc. of ICMC*, 1984, pp. 193–198.

[2] B. Vercoe, "The synthetic performer in the context of live performance," in *Proc. of ICMC*, 1984, pp. 199–200.

[3] P. Cano, A. Loscos, and J. Bonada, "Score-performance matching using hmms," in *Proc. of ICMC*, 1999, pp. 441–444.

[4] C. Raphael, "Automatic segmentation of acoustic musical signals using hidden Markov models," *IEEE TPAMI*, vol. 21, no. 4, pp. 360–370, 1999.

[5] M. Tekin, C. Anagnostopoulou, and Y. Tomita, "Towards an intelligent score following system: Handling of mistakes and jumps encountered during piano practicing," in *Proc. of CMMR*, 2004, pp. 211–219.

[6] B. Pardo and W. Birmingham, "Modeling form for on-line following of musical performances," in *Proc. of AAAI*, vol. 2, 2005, pp. 1018–1023.

[7] A. Cont, "ANTESCOFO: Anticipatory synchronization and control of interactive parameters in computer music," in *Proc. of ICMC*, 2008.

[8] C. Joder, S. Essid, and G. Richard, "A comparative study of tonal acoustic features for a symbolic level music-to-score alignment," in *Proc. of IEEE WASPAA*, 2010, pp. 409–412.

[9] Z. Duan and B. Pardo, "A state space model for on-line polyphonic audio-score alignment," in *Proc. of ICASSP*, 2011, pp. 197–200.

[10] T. Otsuka, K. Nakadai, T. Takahashi, T. Ogata, and H. Okuno, "Real-time audio-to-score alignment using particle filter for coplayer music robots," *EURASIP JASP*, vol. 2011, p. 2, 2011.

[11] C. Joder, S. Essid, and G. Richard, "A conditional random field framework for robust and scalable audio-to-score matching," *IEEE TASLP*, vol. 19, no. 8, pp. 2385–2397, 2011.

[12] N. Montecchio and A. Cont, "A unified approach to real time audio-to-score and audio-to-audio alignment using sequential Montecarlo inference techniques," in *Proc. of ICASSP*, 2011, pp. 193–196.

[13] N. Orio, S. Lemouton, D. Schwarz, and N. Schnell, "Score following: State of the art and new developments," in *Proc. of NIME*, 2003, pp. 36–41.

[14] J. Brown and M. Puckette, "An efficient algorithm for the calculation of a constant Q transform," *JASA*, vol. 92, pp. 2698–2701, 1992.

[15] P. Masri, "Computer modelling of sound for transformation and synthesis of musical signal," Ph.D. dissertation, University of Bristol, 1996.

[16] M. Goto, "Development of the RWC Music Database," in *Proc. of ICA*, 2004, pp. l–553–556.

[17] A. Cont, D. Schwarz, N. Schnell, and C. Raphael, "Evaluation of real-time audio-to-score alignment," in *Proc. of ISMIR*, 2007.

[18] H. Takeda, T. Nishimoto, and S. Sagayama, "Automatic accompaniment system of MIDI performance using HMM-based score following," in *Proc. of the SIG Technical Reports on Music and Computer of IPSJ*, Aug. 2006, pp. 109–116, in Japanese.

[19] Y. Mizuno, H. Tachibana, and S. Sagayama, "Real-time Time-scale Modification of a Music Signal Using Phase Reconstruction for Synchronous Playback in Conducting/Accompaniment System," in *Proc. of ASJ Autumn Meeting*, Sep. 2011, pp. 897–898, in Japanese.