Modeling the Tempo Coupling between an Ensemble and the Conductor

Bridget Baird, Ozgur Izmirli

Center for Arts and Technology, Connecticut College
e-mail: bbbai@conncoll.edu, oizm@conncoll.edu

Abstract

An ensemble and its conductor are in constant communication about their respective tempi. During periods of acceleration the conductor may be ahead of the ensemble; likewise ritards may cause the ensemble to be in front. Modeling this coupling of tempi is the subject of this paper. Data from live performances are gathered, analyzed, and generalized to produce a model; this model is then used to produce a computer system that follows a human conductor. A position sensor is used to track the conductor’s beat and tempo.

1 Introduction

Ideally, the tempo of the performing ensemble and the tempo implied by the movements of the conductor should be very close to each other. In practice, a difference in tempi is observed, especially during parts that have accelerando and ritardando. This is due to the conceptualization of the conductor, who plans ahead, predicting the response latency of the ensemble, in order to change the performed tempo predominantly under her control. For example, during an accelerando, the conductor will conduct slightly ahead of the ensemble until the ensemble catches up. On the other hand, during a ritardando, the conductor stretches the inter-beat intervals in order to make the slowing down apparent to the players. Either way, there is a continual interactive process in which the conductor and the ensemble are communicating to obtain as coherent a performance as possible. Tempo coupling is defined as the measure of how closely the tempo of the ensemble corresponds to the tempo of the conductor.

There are several factors that affect the level of tempo coupling: the amount of experience of the players performing in an ensemble, familiarity with the piece, familiarity with the conductor, experience level and personal conducting style of the conductor, and both the effectiveness and number of rehearsals. The expectations regarding a particular performance of a piece are refined with each new rehearsal and a consensus of tempo variation is gradually established that rigidifies the final performance.

This paper describes a three-part system: an analysis component, a generalization of the results obtained in the analysis, and then a performance component which is informed by the generalization. The analysis component is used to analyze and model the tempo coupling between an ensemble and a conductor. Once the modeling is performed for the performances those results are generalized. Finally, the generalization is used in live performance to enable a computer ensemble to “follow” a human conductor.

Various computer systems that follow a human conductor have been reported in the literature. (Mathews 1989; Morita, Hashimoto and Ohteru 1991; Brecht and Garnett 1995). As well as the usefulness of the conductor’s baton in most of the reported work, more elaborate interfaces to extract gestural information from the conductor’s movements have been proposed. (Marrin and Rosalind 1998) Automated accompaniment systems also deal with the problem of interaction, acquiring information from the performer instead of the conductor. (Toiviainen 1998; Vercoe and Puckette 1985; Dannenberg 1984; Baird, Blevins and Zahler 1993) Also in this context, systems with adaptive gesture mapping strategies have been proposed to reduce the drawbacks of fixed gesture interpretation. (Lee, Garnett and Wessel 1992) A computerized conducting environment (Garnett, Malvar-Ruiz and Stoltzfus 1999) is of particular interest to us, because of its functionality as an educational tool.

2 The Analysis System

The analysis system uses inputs from the conductor’s hand and audio input from the ensemble in order to deduce both the tempo of the conductor and that of the ensemble. The current tempo, deduced from the conductor, is used in conjunction with a history of tempo information to represent the contour of the conductor’s tempo. The tempo contour and the extracted tempo from the audio input are fed into a system that stores the context dependent interaction of the coupling. Generalization is later performed by clustering the collected data from this stage. The analysis system is explained in more detail in the following.

A position sensor is tied to the conductor’s index finger or alternatively to the tip of a baton. For this, a single Bird...
tracker (from Ascension Technology Corporation) is used, which is capable of sampling position data at 100 samples per second with six degrees-of-freedom. The raw position data is then filtered to obtain a relatively smooth position variation curve from which vectorial velocity and acceleration information is obtained. Extrema in the magnitude of the acceleration vector are used in conjunction with the position extrema to determine the beat times of the conductor. Moreover, the direction of the acceleration vector is used to confirm the beat number, in case some beats are not detected or in fact were never made explicit by the conductor. An instantaneous “implied tempo” is calculated from the detected inter-beat times.

Next, an N+1-element context vector, C(i); i = -N..0, representing the tempo contour is calculated for each beat. N is the size of the history window in beats. The element at index 0 is the current implied tempo. The first element in the history window next to the center element, C(-1), represents the tempo in the previous beat and C(-2) represents the tempo two beats before the current one. A small amount of averaging of tempi is done to produce smoothness. The tempi are also normalized; this will have the effect of causing similar tempo changes to look alike, even if their absolute tempi are different. Thus, for example, a two measure ritardando which moves steadily from a tempo of 120 to a tempo of 100 would look similar to one that slows from 180 to 150. The context vector (N=5) in this case might look like (1.1207, 1.0954, 1.0708, 1.0466, 1.023, 1). A value of 5 to 9 for N has proved to be reasonable; it is desirable to have a history for the context vector, but too long a history will not highlight changes in tempo.

Beat times for the audio input are calculated differently. As it is not possible to obtain reliable beat information from an automated beat detection system for live audio and in order not to further introduce errors into our system, the beat times for the audio input have been transcribed by aural and visual examination. It is possible to obtain very precise time estimates with the aid of one-sample resolution audio editing tools. In cases where the players are not in good synchronization with each other, the average onset time is used. Similar to the calculation of the conductor’s implied tempo, using the inter-beat times, and some averaging, an instantaneous “performed tempo” is calculated. This reflects the tempo that has been played by the ensemble. The model uses this tempi information as well as lag information for the coupling of the tempi. Lag information is defined as the difference between the conductor’s beat and the performed beat. The lag is measured in beats rather than absolute time differences to make it independent from absolute tempo.

3 Generalization

Initially, by visually inspecting the difference between the implied and the performed tempi, the lag between the two, particularly during sections of tempo change, can be clearly identified (especially in initial rehearsals). This relationship is modeled using context dependent discretization of the tempo disparity as obtained from the two sources and is explained below.

At constant tempo there is generally no significant disparity between the tempi, although there is a small identifiable constant lag. As long as the disparity is small, the previously performed tempo and the current tempo will be in accordance. We refer to this state as a steady (or stable) state in the coupling. This highly predictable concord between the two parties of the interaction is of less interest to us from a modeling standpoint. The data of interest to us for use in modeling the coupling is available during musical segments in which there is significant difference between the implied and the performed tempi.

Due to the amount of data collected, which is one vector per beat, compression is performed, resulting in a compact representation of the coupling. Compression consists in finding the summary behavior by using a k-means clustering algorithm on the set of context vectors. These vectors can be from a single performance or from a set of multiple performances. Experiments have been done with differing values of k to produce generalizations: k needs to be large enough to distinguish periods of tempo change and to differentiate among different types of tempo changes but too large a value produces no generalization. Ideally, clustering would identify steady states, accelerandos and ritardandos. Clustering would also distinguish between different types of tempo changes; thus two ritardandos that take a similar number of beats to slow down a similar amount would belong to the same cluster; but a ritardando that takes four measures would belong to a different cluster than a more abrupt slowing down. Values of k around 10 seem to show clusters that identify periods of tempo change. The diagram in Figure 1 illustrates information from a live performance. The graphs use the number of beats as the domain. The top portion of the figure indicates the clusters. The middle overlapping line graphs are the implied (solid) and performed (dashed) tempi. The bottom graph shows the lag between the implied and performed beats. There is a period of ritardando starting at beat 4 and continuing until beat 17, then there is a period of accelerando from beat 20 to 35. Note that these periods of tempo change correspond to the clustering at the top of the figure.

Once the clusters are calculated, a representative mean cluster vector is found by taking the average of all the N+1-dimensional context vectors from that cluster. Also for each cluster, the means of the associated lags (positive or negative) between the conductor’s beats and the performed beats are also calculated. These pairs can be interpreted as the generalized learned behavior corresponding to tempo coupling. As a result of the clustering, each mean cluster vector represents a certain lag or lead of the performance with respect to the context. Figure 2 shows a table of mean cluster vectors and their associated lag (+) or lead (-) times. In this table, clusters 7, 8 and 10 can be understood to
represent different phases of a ritardando. On the other hand clusters 1, 2 and 3 refer to an accelerando. Finally, clusters 4, 5, 6 and 9 are seen to represent periods of no significant tempo change. Although we know that certain clusters represent certain kinds of tempo changes (one would indicate a slow ritardando, another an abrupt accelerando, etc.) and certain others represent a steady state, these associations and classifications are not necessary in order for the model to operate.

The system can be made to learn each performance separately, which ensures that learned characteristics are due to the piece, ensemble’s rehearsal level and the conductor. This information, which is called a T set, is stored in a separate file that can be loaded back into memory during the perform modes as explained in the next section. The system can also be set to learn across pieces, performances, ensembles or conductors, but incorporation of any incompatible style or different rehearsal level with the current set will blur the tempo coupling characteristics. For this reason, in cases where clustering is done across performances, it is important to select those that have similar characteristics to be represented in a single T set.

### Figure 2. T set: cluster context vectors and their corresponding lags

<table>
<thead>
<tr>
<th>Cluster Number</th>
<th>Mean Lag (Beats)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.196</td>
</tr>
<tr>
<td>2</td>
<td>-0.143</td>
</tr>
<tr>
<td>3</td>
<td>-0.092</td>
</tr>
<tr>
<td>4</td>
<td>-0.235</td>
</tr>
<tr>
<td>5</td>
<td>-0.169</td>
</tr>
<tr>
<td>6</td>
<td>-0.162</td>
</tr>
<tr>
<td>7</td>
<td>-0.013</td>
</tr>
<tr>
<td>8</td>
<td>0.014</td>
</tr>
<tr>
<td>9</td>
<td>0.014</td>
</tr>
<tr>
<td>10</td>
<td>0.047</td>
</tr>
</tbody>
</table>

### 4 Perform Mode

The user conducts the system in a perform mode. A MIDI file is played according to the detected tempo of the conductor. Position information is taken from the tracker and is used to detect the conductor’s beats, which give tempo. There are two types of perform modes. One is the direct perform mode that enables one to conduct the pre-sequenced musical score using an ideal coupling. In this mode there is no incorporation of the learned behavior into the system except for the lag time incurred in the steady state. The tempo adaptation is implemented by principles similar to those reported by Baird, Blevins and Zahler (1993). In this case the ensemble is assumed to be instantaneously mimicking the conductor and smoothly adapting to the currently implied tempo.

The second mode is the one in which the coupling of tempi is controlled by the learned behavior. In this mode the adaptation of the computer performer is based on the generalized T sets mentioned in the previous section. First of all, a specific T set must be chosen, according to the desired characteristics of performance and conductor. For each (normalized) tempo contour during live performance, the cluster membership for the current context vector is determined. The lag is then retrieved from the T set. The appropriate context vector in the T set is chosen by calculating the distance from the current tempo contour to each of the mean context vectors in the T set (across the N+1 dimensions) and then choosing the mean context vector that has the smallest distance and thus is most similar to the current tempo contour. Once the mean context vector is chosen, the corresponding lag or lead difference (in beats) is used to dictate the computer ensemble’s response.

Although it is hard to quantify the results of the computer performance, a re-analysis identical to the analysis performed on the real ensemble performance and conductor
movements has shown that the system performs in accordance with the input data. The parameters of the system play an important role in the performance. Long history window sizes \( N \) lead to strong specific event memory and make it hard for the system to generalize. On the other hand, too short window sizes prevent the tempo context from being captured in the representation, hence leading to ambiguous responses. The choice of \( k \) in the \( k \)-means clustering algorithm is also important. Although dependent on the other parameters, a small \( k \) leads to less detailed summarization and more compression of the data which makes the system model only crude tempo responses. A large \( k \) will result in little summarization and larger \( T \) set sizes.

5 Discussion and Future Work

A system to analyze and model the tempo coupling between an ensemble and the conductor has been presented. The proposed system has proved to be useful in modeling the stages of coupling level and efficiency between an ensemble and a conductor. The disparity occurring, in reality, between the implied and performed tempi is reflected in the learned behavior of the system. The system can also be useful in the education of beginning conducting students in order to practice with different levels of ensemble experience and responsiveness. A student can use the last rehearsal to form the basis for the next performance.

As for future work, the modeling tool will be used in the investigation of the validity of generalization over many conducting styles by increasing the complexity of the context. Many improvements are possible on the many components of the system. More information can be acquired from the conductor’s movements. On the performance side, a musically more expressive auralization can be considered. Analysis of coupling across conductors with one ensemble or, across ensembles with a single conductor might result in viable generalizations.

Automating the beat detection from the live audio recording would make the system more practical; currently, the analysis involves a tedious manual onset transcription of the performed music. We plan to implement a beat detection front-end for the system to make it more practical to use and to increase the amount of data that can be collected.

In order to be able to make broad generalizations, a significant amount of data need to be collected. This will enable us to draw conclusions about general eminent characteristics of tempo coupling. This will also enable us to infer commonalities between conductor ensemble combinations.

Another modification of the system includes giving more prominence to down beats and important beats, either by only considering those beats or by giving them more weight in calculations.

References


