**Automatic Selection and Concatenation System for Jazz Piano Trio Using Case Data**

Takeshi Hori, Kazuyuki Nakamura, Shigeki Sagayama
Graduate School of Advanced Mathematical Sciences, Meiji University
4-21-1 Nakano-ku, Tokyo, 164-8525 Japan

E-mail: {cs51003, knaka, sagayama}@meiji.ac.jp

**Abstract**

In this paper, we discuss a computational model of an automatic jazz session system that is statistically trainable model using lead sheet and jazz session data, in addition, we provide an implementation as prototype system based on this model. In contrast to most previous jazz session systems that required heuristic rules and the human labeling of training data to estimate musical intention of human players, we suggested a statistically trainable mathematical model of jazz session using stochastic state transition model approximating a musical trajectory model. Based on the model, we developed a jazz session system as a prototype using concatenation of case data from real jazz session recordings to show the validity of our model. This system consists of training phase and concatenating phase. In the training phase, the system learns some parameters to classify the piano, bass, and drums data using non-negative matrix factorization, and calculates the chain probabilities by trigram and co-occurrence probabilities between piano, bass, and drums. In the concatenating phase, the system estimates musical states of bass and drums from piano midi-format input, searches and selects a suitable musical data from case data, and concatenates a musical data matching the key between input piano and bass. As a result of the comparative evaluation experiment using some concatenated midi-format data by above methods, our system was found to generate a jazz piano trio musical data having naturalness and shown validity of our proposing model.

**1 Introduction**

We previously developed an automatic accompaniment system called 'Eurydice' [1] that allows tempo changes and note insertion/deviation/substitution errors in human performance as well as repeats and skips. Although this system can deal with various errors, a score information should be fixed in automatic accompaniment systems like Eurydice since such system need to follow and to match between the score information and player’s performance. Previous studies for an automatic accompaniment system have been proposed from a variety of perspectives such as adding expression [2], support system for children’s practice [3]. On the other hand, to deal with improvised performance and to realize interaction of each player’s performance such as jazz music, there are various researches related ‘jazz session systems’ where improvised performances are allowed. As the next step, we are also working an automated jazz session system that can follow improvised human performances to extend our system.

Jazz session consists of frequently improvised part, where the players improvise on a score by tracking other players’ performances and estimating their intentions based on their previous performances and score information. Therefore, a jazz session system needs to learn relationship between observable musical features and human intentions in order to generate musical performance to cooperate with other musical instruments.

In contrast to most previous session systems based on jazz [4-6] required heuristic rules and the human labeling of training data to estimate musical intention of human players, we suggested a statistically trainable mathematical model of jazz session based on stochastic state transition model using a lead sheet and musical performance data to develop an automated jazz session system [7]. Figure 1 shows our conceptual model.

This paper describes an automatic jazz-piano-trio generating system to show the validity of our session model.
2 Modeling

Our approach is to realize a statistically trainable session system, where imitates relationship between input and output of human jazz sessions which are excluded heuristic rules and data labeling by human. To train these relationship statistically, we need to define a musical performance mathematically. Firstly, since we can feel musicality from connected sounds of a certain span, and music can be represented by a series of short-time feature vectors including all features of music. If a musical information is expressed as a point on a space, the time-series information represented by such vectors can be defined as a trajectory on a space. We defined this space as a ‘musical event space’, where theoretically feature vectors comprise all musical events. Nextly considering extending from individual performances to sessions based on this trajectory model, we can regard musical performance of a session as a set of individual trajectories in a musical event space. Hence, good session is modeled as a set of well-intertwined trajectories in this space. Figure 2 is an idea of our model.

Since we can get the dataset of well-intertwined trajectories from actual performance data, if a session system learns the essence of way of entanglement from good performance data, the system can realize a human interplay.

2.1 Practical problems for realizing

To train statistically based on such trajectory model, there are at least four problems based on data sparsity.

1. Defined space is too high-dimensional to train using a limited data since a musical event space encloses all musical events (such as a number of note, velocity, activities of a player).

2. It is difficult to train relationship of trajectories from a limited data because of continuity.

3. We need to make clear relationship between observed multidimensional features and musical performances.

4. The system needs to generate other instruments’ parts.

We can arrange these problems as based on ‘features’, ‘trajectory’, ‘mapping’, and ‘rendition’. In an ideal case where an unlimited amount of data are available, we would be able to deal with these problems, however, actually we have a limited amount of data, so that to solve these problems, we first developed as prototype system dealt with above problems and next considered whether it can be solved by the following methods in this research.

- **Features**
  - Dimensionality reduction

- **Trajectory**
  - Discretization (interpolation)
  - Mapping between features of each player

- **Musical mapping**
  - Mapping between features and performances
  - Sample-based (substitution using case data)

- **Rendition**
  - Sample-based (substitution using case data)

3 Outline of the prototype system

The objective of the prototype system is to show validity of our proposing mathematical model for jazz session. As configuration in this paper, we adopted a jazz piano trio consisting of a piano, bass, and drums. This system first receives midi-format data consisting of an only piano performance data, and next outputs a midi-format data of piano trio synthesized performance data of bass/drums using case data. Although we modeled a musical performance as trajectory in all musical events space, where the session system needs to train a set of intertwined trajectories, we can approximate a trajectory model with a stochastic state transition model because of discretization, so that we can approximately estimate a corresponding trajectories with input trajectory expressed by discretizing points (Figure 3).

3.1 Features: Style parameter

We selected parameters closely related to the jazz session on the basis of musical knowledge and defined them as ‘style parameters’ to set an effective axis in the session model. These parameters are used to track musical performances and to calculate a degree of similarity between every instruments. We defined 68 parameters that are extractable from the music performance at every unit time (every bar in this paper) as follows:
Fig. 3: Concept of case-data-based automatic session system approximated by a stochastic state transition model

- **Piano-specific features**
  - The number of notes composed of diatonic chords, and the character of notes such as tension notes, avoid notes, and blue notes.
  - The range between the highest and lowest tones.
- **Bass-specific features**
  - The range between the highest and lowest tones.
- **Drums-specific features**
  - Each number of notes of the hi-hat cymbal, snare drum, and crash cymbal.
- **Common features**
  - The number of notes, the number of simultaneous sounds, the average velocities.
  - The ratio of the above features between adjacent time spans.
  - The ratio of sum of the above features throughout the music.
  - The ratio of off-beat notes to all notes in the unit time.

3.2 Trajectory and Musical mapping: NMF, trigram

Although we need to learn co-occurrence of every instruments’ trajectories to search a matching performance between instruments, relationship of trajectories is non-linear since defined musical features axis of the space by style parameters are different from each other. This problem is classified a non-linear identification problem. In recent years, deep neural network (DNN) is often used to learn the non-linear co-occurrence statistically, however to simplify, we estimated a matching performance data by assuming linearity between the same instrument’s space, and searching nearest neighbor with Euclidean distance criterion. On the other hand, since the assumption of linearity is a strong constraint, we considered not only all space but also subspaces. In order to segment whole space into subspaces, we used a clustering method.

To determine a effective clustering method, we previously compared three methods: k-means clustering, Gaussian mixture model (GMM), and non-negative matrix factorization (NMF)[8]. As a result of our previous research, since we could get a experimental results where NMF clustering yielded the highest prediction accuracy, we used NMF clustering for discretization.

### 3.2.1 NMF

NMF is a method for factorizing a non-negative matrix into a pair of non-negative matrices with a lower rank [6]. A non-negative original matrix $X$ is factorized

$$X_{i,j} \approx \sum_k H_{i,k} U_{k,j},$$

where $k$ denotes an index of the basis. $H$ is called basis matrix and $U$ represents an activation. To calculate the criterion for approximation, a distance measure between $X$ and $HU$ is generally selected from the Euclidean distance, generalized Kullback-Leibler divergence, or Itakura-Saito divergence, we used generalized Kullback-Leibler divergence since the distance was achieved the highest prediction accuracy compared other methods in this model by our previous research. In the case of generalized Kullback-Leibler divergence, emplyed the probability density distribution between observed data and approximated matrix $HU$ is continuous log Poisson distribution. The distance based on generalized Kullback-Leibler divergence $D(X||HU)$ is expressed by

$$D(X||HU) = \sum_{i,j} \left[ X_{i,j} \log \frac{X_{i,j}}{\sum_k H_{i,k} U_{k,j}} - \left( X_{i,j} - \sum_k H_{i,k} U_{k,j} \right) \right].$$

On the other hand, the continuous log Poisson distribution $\log Po(X||HU)$ is

$$\log Po(X||HU) = \sum_{i,j} \left[ X_{i,j} \log \sum_k H_{i,k} U_{k,j} - \log X_{i,j} - \sum_k H_{i,k} U_{k,j} \right].$$

Considering the minimization for $HU$, these formulas are the same.

Every parameters $h_{ik}, u_{kj}$ can be calculated by multiplicative update equation as follows based on auxiliary function technique:

$$h_{ik} \leftarrow h_{ik} \frac{\sum_j x_{i,j} u_{kj}}{\sum_j u_{kj}}, \quad u_{kj} \leftarrow u_{kj} \frac{\sum_i x_{i,j} h_{ik}}{\sum_i u_{ik}}.$$ 

An actual activation matrices $U$ was given as follows by using an original matrix $X$ and a generalized inverse matrices of the basis matrices $H^+$ obtained basis
matrices $H$:

$$U_{\text{train}} = H_{\text{train}}^+ X_{\text{train}}$$
$$U_{\text{test}} = H_{\text{train}}^+ X_{\text{test}}.$$ 

Then class numbers $c_k$ were assigned by

$$c_k(x_j) = \arg \max_k u_{kj}.$$ 

Since a high-dimensional musical event space is segmented into subspace by using NMF clustering, time series characteristics of trajectories are approximated by class series. Although there are various methods for tracking time series characteristics of a musical performance which are expressed by stochastic state transition model, for instance trigram and hidden Markov models, we used trigram because the prediction accuracy of trigram is better than that of HMMs in the simply discretized model according to our previous research.

### 3.2.2 Trigram

Trigram is one of an N-gram model (in the case of $N = 3$). In an N-gram, given $n$ states $\{s_1, s_2, \ldots, s_n\}$, the chain probability is given as follows:

$$P(s_1, s_2, \ldots, s_n) = \prod_{i=1}^n P(s_i|s_{i-N+1}, \ldots, s_{i-1}).$$

For a trigram, the number of transitions from $i-2$ to $i$ is expressed by $N(s_{i-2})$ and the chain probability is given as

$$P(s_i|s_{i-1}^{-1}) = \frac{N(s_{i-2})}{N(s_{i-2}^{-1})}.$$ 

In this system, chain probabilities by trigram are used to estimate classes of not piano but bass and drums since we don’t need to estimate a piano performance because of batch processing.

### 3.2.3 Musical mapping

Since the system needs to estimate input-piano/bass/drum classes from input piano data in the case of considering subspace, the system also needs to assume a class number of bass/drums corresponding to the estimated input-piano class at every bars using chain probabilities and co-occurrence probabilities. A class number of each instrument in the measure number $t$ is expressed as

$$c_t^{\text{Piano}}, c_t^{\text{Bass}}, c_t^{\text{Drums}}.$$ 

Similarly, a chain probability based on trigram is

$$p(c_t^{\text{Bass}}|c_{t-1}^{\text{Bass}}, c_{t-2}^{\text{Bass}})$$
$$p(c_t^{\text{Drums}}|c_{t-1}^{\text{Drums}}, c_{t-2}^{\text{Drums}}).$$

Co-occurrence probability between piano and bass/drums is

$$p(c_t^{\text{Drums}}|c_t^{\text{Piano}})$$
$$p(c_t^{\text{Bass}}|c_t^{\text{Piano}}).$$

Consequently, bass/drums class number at times $t$ is estimated from product of these probabilities as

$$c_t^{\text{Bass}} = \arg \max_{c_{\text{Bass}}} p(c_t^{\text{Bass}}|c_{t-1}^{\text{Bass}}, c_{t-2}^{\text{Bass}})$$
$$c_t^{\text{Drums}} = \arg \max_{c_{\text{Drums}}} p(c_t^{\text{Drums}}|c_{t-1}^{\text{Drums}}, c_{t-2}^{\text{Drums}})$$
$$p(c_t^{\text{Drums}}|c_t^{\text{Piano}}).$$

Using above probabilities, The system can estimate the bass and drums classes considered time series of them and co-occurrence information with piano.

### 3.3 Rendition

Although the system uses case data selected as a similar performance data with input-piano data to generate a synthesized music, a mapping between features of instruments is often missing in the case data because of data sparsity. To compensate the problem, as mentioned above, we assumed linearity in a same instrument’s space, and the system first searches the nearest piano-case data with Euclidean distance criterions and substitutes the input based on the linearity, and next extracts bass/drums performance data co-occurring the piano data. To search the best case data, we compared following four criterions and evaluated the validity.

1. **Random choice**
   Randomly chose a bar from all songs

2. **Nearest neighbor**
   Choose the nearest neighbor case in the entire case data in the Euclidean distance sense

3. **NMF**
   Choose the nearest neighbor case with the cluster the input belongs to

4. **Co-occurrence + trigram**
   - Choose a case of the highest joint probability of co-occurrence and trigram.
   - Use co-occurrence across the input and the others (i.e., matching constraint)
   - Use trigram of classes along the individual instrument (i.e., time-series constraint)

The differences are that ‘Nearest neighbor’ searches nearest neighbor data from the whole space, by contrast, ‘NMF’ and ‘Co-occurrence + trigram’ use the subspaces by clustering. Meanwhile, in contrast to ‘NMF’ considers the matching classes between only piano, ‘Co-occurrence + trigram’ estimates the bass/drums classes and searches the matching data under the constraint that all instruments classes are the same.
4 Automatic selection and concatenation system for jazz piano trio: SCSJ

Based on above discussion, we developed an automatic system (SCSJ) that received midi-format data consisting of an only piano performance data and output a midi-format data of piano trio concatenated performance of bass/drums data from case data.

SCSJ comprises two phases (training phase and concatenating phase) and three parts (training part, analyzing part, and output part). Figure 4 shows a process chart of SCSJ.

4.1 Training phase

In a training phase, first, SCSJ estimates the key because modulation often occurs in a jazz session. Nextly, using style parameters based on a estimated key and some observable features of training date, SCSJ obtains basis matrices from NMF algorithm at every instruments, so that SCSJ can estimate a class number at every bars from these basis matrices. Additionally, SCSJ calculated chain probabilities based on trigram and co-occurrence probabilities between piano and the others’ class number. These parameters (basis matrices, chain probabilities, and co-occurrence probabilities) are stored into a database to estimate bass/drums classes from piano midi-format input data. This part for calculating these parameters is named as training part.

4.2 Concatenating phase

A concatenating phase mainly consists of two parts: an analyzing part and an output part. Firstly, midi-format data consisting of only a piano-data inputs into SCSJ. SCSJ first estimates key because of modulation as well as training phase. In an analyzing part, SCSJ estimates a class number of input data using basis matrices. Nextly, SCSJ assumes class numbers of bass/drums corresponding to the estimated input data’s class at every bars using chain probabilities and co-occurrence probabilities.

Moreover, SCSJ searches a bar number having a performance data at every bars for synthesizing from case data having same class with input data and having the most nearest style parameters based on Euclidean distance. In contrast to a bass/drums data is searched from same piano class with input piano in the case of \( t < 2 \), a bass/drums data is searched from same input and case data of piano and estimated bass/drums class in the case of \( t \geq 2 \). In the case between input piano data and selected bass performance’s keys are disharmony, the key of the bass is shifted to be matched key of input data.

Finally, input piano and selected/shifted bass and drums data are concatenated, so that the piano trio data is generated and output as midi-format in an output part.

4.3 Key estimation

Modulation is often carried out in a jazz. For instance, some parts of chord progression of ‘Autumn Leaves’ (key: G\textsubscript{moll}) which is one of a jazz standard music are given as \( \{ C\text{m7} - F\text{7} - B\text{b7} - E\text{b7} - A\text{b5} - D\text{7} - G\text{m} \} \). We should estimate a key to generate a matching performance because of such modulation. The key estimation on this system is described by the rule-based. One of a method for analyzing a modulation is to utilize ‘II\textsubscript{m7} − V\text{7}’ motion commonly used in a jazz. We can find ‘C\text{m7} − F\text{7}’ as ‘II\text{m7} − V\text{7}’ with B\text{b7} key. On the other hand, ‘A\text{b5} − D\text{7}’ is the same of ‘II\text{m7} − V\text{7}’ with G\text{moll} key. As a result, we can see that this chord progression consists of a modulation from B\text{b7} to G\text{moll}.

Figure 5 shows the result of chord analysis for ‘Autumn Leaves’. Chord name written in black is a chord progression of ‘Autumn Leaves’ and red represents a estimated key name.

Therefore, this song comprises a configuration that is repeated several times part modulation of B\text{b7} (relative key) and G\text{moll} (tonic key) via pivot chord (common diatonic chord in multiple key) like E\text{b}\text{m7}. 

Fig. 4: Process chart for automatic selection and concatenation system for jazz piano trio

Fig. 5: Chord progression of Autumn Leaves
Fig. 6: Comparative evaluation by five-grade evaluation

In addition, some cyclic chord (such as ‘I – VI\textsubscript{m7} – II\textsubscript{m7} – V\textsuperscript{7}’), a function of chord (such as tonic, dominant, and subdominant), and secondary dominant (dominant chord corresponding assumed a temporary tonic in a diatonic chord) are utilized as modulation patterns to estimate the key.

Estimated key is utilized not only to extract style parameters like tension note but also to match the key between input piano data and selected bass data since a key of input data is often different from that of case data. A pitch of selected bass data is shifted to be the sound of diatonic scale with estimated key of an input data to match with an input data.

5 Experimental evaluation

As configuration, we used fifteen NMF-clustered classes to generate a music, where the number of classes was to maintain accuracy rate over 80% according to previous research. Chain and conditional probabilities were calculated by parameters based on NMF. On the other hand, We used cross validation upon training and generating. Input piano data was extracted from a song out of 13 songs data (1: ALL OF ME, 2: AUTUMN LEAVES, 3: BLUE MONK, 4: BYE BYE BLACK-BIRD, 5: YOU’D BE SO NICE TO COME HOME TO, 6: HOW INSENSITIVE, 7: MOANIN’, 8: NIGHT AND DAY, 9: ROUND MIDNIGHT, 10: SOFTLY AS IN A MORNING SUNRISE, 11: STELLA BY STARLIGHT, 12: WALTZ FOR DEBBY, 13: THE DAYS OF WINE AND ROSES), the other songs were used for training. Research participant evaluated random segments of approximately 45 seconds omitting intro.

We used a five-grade evaluation by two student with musical experience of more than ten years and two students no musical experience. Figure 6 shows the result of comparative evaluation of four searching methods by ‘random choice’, ‘nearest neighbor’, ‘NMF’, and ‘Co-occurrence + trigram’. The x axis expresses a music number (based on above number), and y axis illustrates mean opinion scores using five-grade evaluation.

We can find that ‘co-occurrence + trigram’ had the most highest evaluation. Meanwhile, BLUE MONK and HOW INSENSITIVE were evaluated lower than others. Since tonic I\textsuperscript{7} is often regarded as dominant in blues, the lower evaluation of BLUE MONK was due to key estimation’s errors. Similarly, HOW INSENSITIVE was due to key estimation’s errors too. Because there is no tonic key, a tonality is floating overall in this song. Moreover, in this research, since we extracted style parameters and analyzed a bar unit, the system could hardly grasp fine features changes by clustering.

6 Conclusions

We modeled a session as a trajectory model, approximated as a stochastic state transition model, and developed the SCSJ based on the mathematical model. Although this system is prototype using case data to simplify, it could show validity our proposing model based on statistically trainable model since co-occurrence and trigram (and NMF clustering) performed better than others on listening evaluation. As future works, we plan to reconstruct using continuous mixture HMMs and DNN. Although we approximated time series characteristics with class series in this model to deal with robustly, using the HMMs, we will deal with clustering and time series characteristics integrally rather than individually. In addition, we want to approximate the trajectory in more detail by LSTM and DBN. To generate bass and drums performances without case data is also an important issue. Furthermore, non-parametric bayesian inference might be effective to exclude parameters more, and to deal with an acoustical input, we need to research about multi-pitch analysis.

References


