A Proposal of an Interactive Music Composition System Using Gibbs Sampler

Akira Shirai and Tadahiro Taniguchi

Ritsumeikan University,
312 RacoonD3 3-1-7 Nojihigashi Kusatsushi Shiga,
525-0058, Japan
{shirai,taniguchi}@em.ci.ritsumei.ac.jp

Abstract. In this paper, we propose a novel method for generating a melody from a generative probabilistic model by using Gibbs sampler. Furthermore, users can modify the composed music by interacting with the generation process. This method enables users to create their favorable melodies. Recently, music composition using computer software is gathering attentions. Many people want to compose their original music. However, musical composition is still too difficult for beginners to obtain their favorable original music. Our system highly supports such people easy to create own music. We also evaluate the probabilistic music composition method by several experiments.

Keywords: automatic music composition, interactive system, probabilistic model.

1 Introduction

Recently, a music composition using a computer named DTM is very popular means of the musical activity. People can make an original music easily. However, a music composition is still too difficult for the people who have less knowledge of music. They can't assemble a music structure because of complexity of itself. Therefore, it is a very important to develop the system that assists such a people to create music. Although there are many methods for supporting beginners, we focus on an automatic music composition system.

Automatic composition techniques have been investigated in the past. A selection of keywords is a very popular means for such system. It generates melodies based on keywords (e.g. sad, happy, jazz, rock) that an user select. In this means, the user chooses not feelings of melodies in it but keywords corresponding to feelings. In short, the user converts feelings into keywords and then the system translates these into melodies. However, the user doesn't get needed melodies when the translation is irreversible because of a variety of interpretations. In order to solve this problem, the system using interactive evolutionary computation (IEC) has been proposed [1]. In this system, a melody is expressed as a gene. The user listens to melodies and evaluates them. Then, the system uses a crossover algorithm to generate next generation melodies. After a number of iterations, melodies change into preferable music for the

J.A. Jacko (Ed.): Human-Computer Interaction, Part I, HCII 2011, LNCS 6761, pp. 490–497, 2011.

user. This approach is highly suitable for beginners because of easy process of composition. However, there is a problem that the user is exhausted in evaluation process by reason of a total time of listening generated melodies. On the other hand, a system is proposed that using dynamic programming named Orpheus [2]. This system is designed with considering composition as an optimal-solution search problem under constraints of the prosody of the Japanese lyrics. Users can compose their original music with lyrics automatically and easily on this. However, it is difficult to search for a solution when global constraints are used. Moreover, in the same conditions of former generations, the system cannot generate various types of music due to take an optimal-solution. To solve these problems, we focus on a music generation from statistical models [3]. An example of this approach is IDyOM [5]. It is a model of melody perception based on n-gram model. Moreover, melody generation from this model using Metropolis-Hastings algorithm, a type of Markov Chain Monte Carlo (MCMC) sampling method, is reported. Although this model shows a good performance on the music segmentation, generated melody is not so creative. We think a lack of the code progressions causes this failure. It is one of the most important aspects of the music structure. Another problem is the generation process that is based on entirely the learnt models. In a word, people can't be involved in the generation process. Melodies generated by such a process are almost good, however these are useless when the user is not interested in. The users need their own original and favorable music.

In this paper, we propose a new interactive system for music composition. Our system generates melodies by using Gibbs sampler, a type of MCMC sampling method, from a probabilistic language model according to an n-gram model that learns existing music. Users can interact with the process of generation under constraints given by several musical structures.

We describe the whole system in Section 2. In Section 3, we show some experimental result using our system and conclude in Section 4.

2 Interactive Music Composition System Using Gibbs Sampler

Our system generates melodies using Gibbs Sampler from bigram model with constraints given by grouping structures of a user's preferred song. bigram model learns from music corpora and then grouping structures are calculated by using exGTTM. Furthermore, users can revise the generated melody by pointing a dissatisfaction part of it. Fig. 1 shows a flowchart of this system. Its details are described in sections below.

2.1 Music Corpora

To get probabilities of bigram model, we define music corpora that have notes and code progression, these are divided into eighth note, from existing music. Notes are represented as $\mathbf{s} = (s_1, s_2, ..., s_N)$ each s has a MIDI note number, rest or duration where N is a dimension that expresses length of melodies. Code progression is $\mathbf{c} = (c_1, c_2, ..., c_N)$ with a code name.

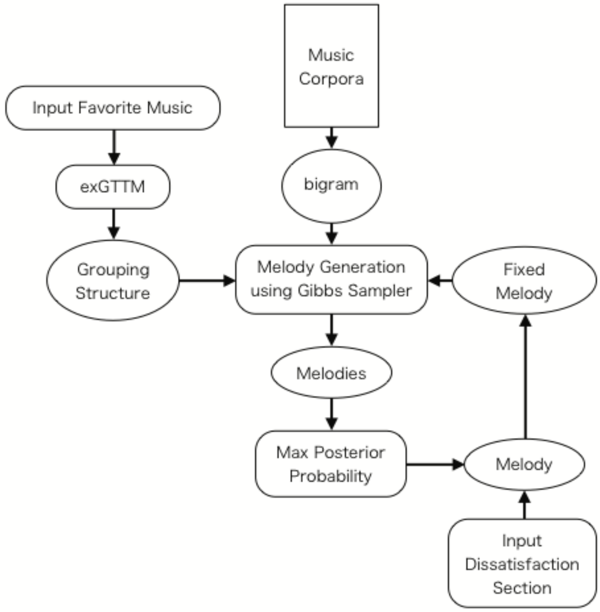

Fig. 1. Flow chart of our system. It generates a melody with favorite music input and music corpora.

2.2 n-Gram Model

We use n-gram model with $n = 2$ (bigram models) as the probabilistic music model β. This model is often used as language models, which approximates the distribution over sentences using the conditional distribution of each word given a context consisting of only the previous $n-1$ words. In this paper, we take notes with codes as words and melodies as sentences. Therefore, a melody's posterior probability is given by

$$p(\mathbf{s} \mid \beta) = \prod_{i=1}^{N} p(s_i \mid s_{i-1}, c_i) . \tag{1}$$

To training this model, bigram probabilities are calculated by counting notes in the corpus

$$p(s_t \mid s_{t-1}, c_t) = \frac{\mathrm{C}(c_t, s_{t-1}, s_t)}{\mathrm{C}(c_t, s_{t-1})} . \tag{2}$$

$\mathrm{C}(\bullet)$ is a function that counts notes. However, it is possible that probability denotes 0 because of inexistent sequence. Therefore, smoothing method is needed for training n-gram models. In this system, instead of (2), probabilities are trained by using

$$\hat{p}(s_t \mid s_{t-1}, c_t) = \sum_{R} w_R \, p_R . \tag{3}$$

$$\sum_{R} w_R = 1. \tag{4}$$

This smoothing method uses weighted summation. w_R denotes weight of each probabilities and p_R denotes the several different probabilities that are marked by R, where $R = \{(\text{uni}, \text{oncode}), (\text{uni}, \text{offcode}), (\text{bi}, \text{oncode}), (\text{bi}, \text{offcode}), (\text{bi}, \text{degree})\}$. For example, $p_{\text{uni,offcord}}$ is calculated by

$$p_{\text{uni,offcord}}(s_t \mid s_{t-1}, c_t) = \frac{\mathrm{C}(s_t) + B}{\sum_t \mathrm{C}(s_t) + B\lambda}. \tag{5}$$

B denotes a baseline probability and λ denotes a number of the kind of notes. $p_{\text{bi,degree}}$ is peculiar. It is calculated by transporting notes to $\mathrm{C}_{\text{major}}$.

$$p_{\text{bi,degree}}(s_t \mid s_{t-1}, c_t) = \frac{\mathrm{C}(\mathrm{d}(s_t, c_t), \mathrm{d}(s_{t-1}, c_t)) + B}{\mathrm{C}(\mathrm{d}(s_{t-1}, c_t)) + B\lambda}. \tag{6}$$

$\mathrm{d}(\bullet,\bullet)$ is a function that transports notes on codes to $\mathrm{C}_{\text{major}}$.

2.3 Generative Theory of Tonal Music

A generative theory of tonal music (GTTM) [6] is a theory of music that is intended to be a formal description of the musical intuitions of a listener who is experienced in a musical idiom. It is composed of four components. *Grouping structure*, one of these components, expresses a hierarchical segmentation of the piece into motives, phrases, and sections. We suppose that feeling of music is connected with *grouping structure*. According to this hypothesis, users can generate a new favorable melody under constraints given by *grouping structure* of a user's preferred melody.

We use *extended GTTM* [7] to make *grouping structure*. This algorithm is GTTM reorganized for computation. Although *grouping structure* is composed of several rules, our system uses these four rules, GPR (*Grouping Preference Rule*) 2a, 2b, 3a and 3d. These are rules of boundaries between groups. Each rules has a different attribute, slur/rest, attack-point, register and length of notes, which are calculated with their respective parameters. Boundaries are represented as vectors; each value is 0 or 1, of N dimensions. Our system sets these vectors as constraints $\mathbf{z}$ of musical structure.

2.4 Probabilistic Model

We define a probabilistic model of generating grouping structure (Fig. 2) that is expressed by

$$p(\mathbf{z} \mid \mathbf{s}, \beta). \tag{7}$$

In this system, we need a posterior probability of melody $\mathbf{s}$. Thus, an equation (7) is transformed by using Bayes' theorem into

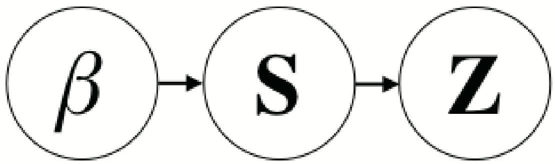

Fig. 2. Graphical model of generation **z**. β is bigram model, **s** is melody and **z** is grouping structure.

$$p(\mathbf{s} \mid \mathbf{z}, \beta) = \frac{p(\mathbf{z} \mid \mathbf{s}) p(\mathbf{s} \mid \beta)}{\sum_{\mathbf{s}'} p(\mathbf{z} \mid \mathbf{s}') p(\mathbf{s}' \mid \beta)}. \tag{8}$$

This can be calculated using equation (1) and conditional probability of **z**

$$p(\mathbf{z} \mid \mathbf{s}) = \frac{1}{(\sqrt{2\pi})^N \sqrt{|\Sigma|}} \exp(-\frac{1}{2}(\mathbf{z} - \mu(\mathbf{s}))^t \Sigma^{-1} (\mathbf{z} - \mu(\mathbf{s}))). \tag{9}$$

$\mu(\mathbf{s})$ denotes grouping structure of melody **s**.

In order to take grouping structure into the probabilistic model, we suppose that it is generated from *N*-dimensional Gaussian distribution (9).

Since there are λ^N possible melodies, computation is infeasible. However, we can get high probability melody on low complexity by using Gibbs sampler.

2.5 Gibbs Sampler

Gibbs Sampler [4] is a simple and widely applicable MCMC algorithm and can be seen as a special case of the Metropolis-Hastings algorithm. Consider the distribution from which we need to sample. Each step of the Gibbs sampler involves replacing the value of one of the variables by a value drawn from the distribution of that variable conditioned on the values of the remaining variables. The distribution of initial states must also be specified in order to complete the algorithm, although samples drawn after many iterations will effectively become independent of this distribution.

To implement this algorithm into our model, **s** is separated into

$$p(s_i \mid \mathbf{s}_{\setminus i}, \mathbf{z}, \beta). \tag{10}$$

According to the algorithm, we replace s_i by a value drawn from the distribution $p(s_i \mid \mathbf{s}_{\setminus i}, \mathbf{z}, \beta)$, where s_i denotes the i^{th} note of melody **s**, and $\mathbf{s}_{\setminus i}$ denotes $s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_N$. Thus, from (1) and (8), we can obtain

$$p(s_i = k \mid \mathbf{s}_{\setminus i}, \mathbf{z}, \beta) \propto p(\mathbf{z} \mid s_i = k, \mathbf{s}_{\setminus i}) p(s_i = k \mid s_{i-1}, c_i) p(s_{i+1} \mid s_i = k, c_{i+1}). \tag{11}$$

Melodies are generated by using Gibbs sampler algorithm with (11).

2.6 Melody Revision

Automatic composition systems can generate original music. However, it is not creation for users but recommendation from the system. Therefore, composition processes in the system must interact with them.

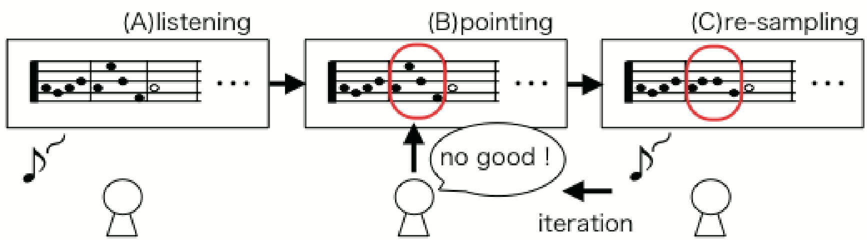

Fig. 3. Revision process of the system. It re-samples part of melody where the user don't satisfy.

In our system, users can revise the generated melody. Revision process (Fig. 3)is executed when the user cannot satisfy with melody. First, the user listens the generated melody and sees the score (Fig. 3 (A)). Next, the user inputs a bar number who feels dissatisfaction (Fig. 3 (B)). Finally, the melody is re-sampled by using Gibbs sampler with cut range of i (Fig. 3 (C)). Revision sampling is affected by $\mathbf{s}_{\backslash i}$ from the property of Gibbs sampler. The user can obtain own original melody after some iteration.

3 Experimental Results

We performed experiments on our interactive music composition system using a 32 verses corpus including 4 Japanese famous musicians. The system generated verse, $N = 64$, $B = 0.0001$, $T = 100$ [1] , code progression was Dm7-G-Em-Am and $w_R = \{0.2, 0.2, 0.2, 0.2, 0.2\}$.

Fig. 4 shows Mean squared errors of GPRs between generated melodies and user's preferred music. All MSE are reduced on the left in case of $\Sigma = 0.05$ compared with the right in case of $\Sigma = 0.5$.

Furthermore, we performed evaluation experiments in same conditions except for $w_R = \{0.3, 0.1, 0.2, 0.2, 0.2\}$. Subjects selected a preferred music for constraints, and then the system generated melodies. After Gibbs iterations, they listened a generated melody and evaluated it. They could revise it until they felt satisfaction.

Table 1 shows a number of revisions for each subject. Most of subjects revised a generated melody only one time. Surprisingly, subject 4 didn't revise a melody (Fig. 5). It shows that our system can compose interesting melody for users in a short time. However, it is possible that subject 4 hadn't had intention of composition.

[1] T : A number of Gibbs iteration.

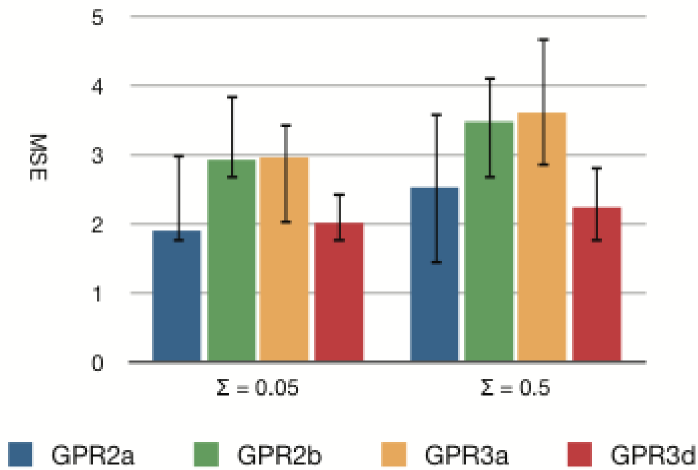

Fig. 4. Mean squared error of GPRs between generated melodies and user's preferred music. Σ is constraint parameter.

Table 1. Results of experiment for subjects. Subject 4 didn't need a revision

Subjects	Artist	Number of revision
Subject 1	Spitz	1
Subject 2	Spitz	1
Subject 3	KinKi Kids	1
Subject 4	**Bump of Chicken**	**0**
Subject 5	Spitz	1

Fig. 5. Generated melody (subject 4). Constraints are given by J-Pop song "Stage of the ground - Bump of Chicken".

4 Conclusion

We have described an interactive music composition system using Gibbs sampler. In addition, we have shown some results of experiments using this system. The experiments showed good results. Generated melodies have less non-harmonic notes. Moreover, the structure of generated melodies is similar to the structure that is given as constraints. The creativity of melodies is another important aspect. It is highly subjective, melodies generated by our system satisfy a user's demand. Most of users that use this system prefer generated melodies. The interaction part also receives a good evaluation. However, this part is claimed with its UI and function. Our system now doesn't have good UI. The user has to input by keyboard on console screen. It is a critical problem for intuitive system that we aim at. The users probably feel troublesome. Thus, a special UI that properly affects melody (when manipulations link with user's intuitions for music, they may not aware of musical structures) is needed.

Music has various aspects (such as pitch, metric, code progression, hierarchical structure and so on) that affect the user's intuition. Therefore, our system must include more and more structural constraints. We develop our system with results described above. The system that associates with the singing voice synthesis system is considerable in the future work.

References

1. Tokui, N.: Music composition with interactive evolutionary computation. In: Third International Conference on Generative Art, Milan, Italy, pp. 215–222 (2000)
2. Fukayama, S., Nakatsuma, K., Sako, S., Yonebayashi, Y., Kim, T.H., Qin, S.W., Nakano, T., Nishimoto, T., Sagayama, S.: Orpheus: Automatic composition system considering prosody of Japanese lyrics. In: Natkin, S., Dupire, J. (eds.) ICEC 2009. LNCS, vol. 5709, pp. 309–310. Springer, Heidelberg (2009)
3. Conklin, D.: Music Generation from Statistical Models. In: Proceedings of the AISB 2003 Symposium on Artificial Intelligence and Creativity in the Arts and Sciences, Aberystwyth, Wales, pp. 30–35 (2003)
4. Geman, S., Geman, D.: Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. IEEE Transactions on Pattern Analysis and Machine Intelliegence 6(1), 721–741 (1984)
5. Wiggins, G.A., Pearce, M.T., Mullensiefen, D.: Computational modeling of music cognition and musical creativity. In: Dean, R. (ed.) Oxford Handbook of Computer Music and Digital Sound Culture. Oxford University Press, Oxford (2009) (in press)
6. Lerdahl, F., Jackendoff, R.: A generative theory of tonal music. MIT Press, Cambridge (1983)
7. Hamanaka, M., Hirata, K., Tojo, S.: Implementing "A Generative Theory of Tonal Music". Journal of New Music Research 35(4), 249–277 (2006)