AUTOMATIC SONG COMPOSITION FROM THE LYRICS EXPLOITING PROSODY OF JAPANESE LANGUAGE

Satoru Fukayama, Kei Nakatsuma, Shinji Sako, Takuya Nishimoto and Shigeki Sagayama
the University of Tokyo, Nagoya Institute of Technology,
{fukayama,nishi,sagayama}@hil.t.u-tokyo.ac.jp
tsuma@alab.t.u-tokyo.ac.jp, s.sako@nitech.ac.jp

ABSTRACT

Automatic composition techniques are important in sense of upgrading musical applications for amateur musicians such as composition support systems. In this paper, we present an algorithm that can automatically generate songs from Japanese lyrics. The algorithm is designed by considering composition as an optimal-solution search problem under constraints given by the prosody of the lyrics. To verify the algorithm, we launched *Orpheus* which composes with the visitor's lyrics on the web-site, and 56,000 songs were produced within a year. Evaluation results on the generated songs are also reported, indicating that *Orpheus* can help users to compose their own original Japanese songs.

1. INTRODUCTION

Recently, there has been wide interest in automatic composition algorithms which can help amateur musicians to compose original tunes. Although considerable amount of research has been done on automatic composition[1][2][3], much less has been done on composing songs from the lyrics, and the question of what information in the lyrics should be exploited for generating songs remains. Syntactic information of the lyrics are used in some researches[4] to compose a song from the lyrics. Musicologists argue that there are considerable correlations between music and prosody. In case of composing songs, prosody plays a more important role. However, no system that uses prosody has yet been attempted.

2. MELODY COMPOSITION ALGORITHM EXPLOITING PROSODY OF JAPANESE LYRICS

Our objective is to develop a method to generate a melody automatically which satisfies the constraints given by the prosody of Japanese lyrics. We define melody composition here as generating a melody given the lyrics, patterns of rhythms (by "rhythm tree" which is described later in this section), and harmony sequence with specifications of tonality and scale.

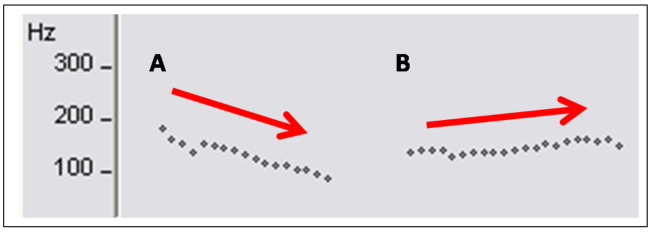

Figure 1. Pitch contuor of "ka'mi" and "kami"': the pitch accent of the Japanese word is lexically contrastive in ka'mi(A) 'god' vs. kami'(B) 'paper'.

The composition algorithm consists of two parts. The former part is for designing the rhythm for the melody by considering the uniformity of the rhythm in the song. The latter is for designing the pitch of each note with probabilistic inference.

We firstly review the properties of Japanese prosody and argue the importance of considering prosodic information when composing Japanese songs. Secondly, we describe a method to generate rhythm of the melody. Then, we discuss how to obtain a proper pitch given the rhythm, harmony sequence, and the lyrics. Finally, we introduce our automatic song composition system *Orpheus*, which is based on our proposed algorithm and used in the evaluations.

2.1 Japanese Prosody and its Role in Composition

Japanese is said to "have a fixed shape consisting of a sharp decline around the accented syllable, a decline that is usually analysed as a drop from a H [1] tone to a L [2] "[5]. Furthermore, "the place of the accent is lexically contrastive, as in ka'mi 'god' vs. kami' 'paper"'[5]. (Fig. 1)

A melody attached to the lyrics cause an effect similar to the accent. Therefore we can assume that the prosody of Japanese lyrics imposes constraits on pitch motions of the melody.

2.2 Composition of Rhythm

In order to design a rhythm on the given lyrics, two problems have to be considered. The first problem is the allocation of lyrics on the melody, and the other problem is how to handle the unity of rhythm in the same song.

[1] H: high
[2] L: low

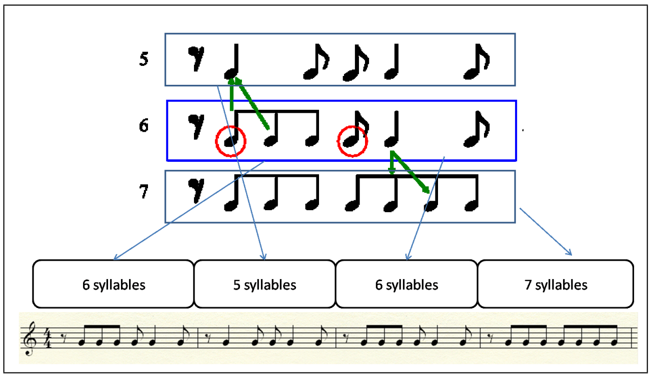

Figure 2. By using the "rhythm tree"(above), rhythm corresponding to the number of syllables are generated with consideration of keeping the unity of rhythm feature in the same song.

2.2.1 *Allocation of Lyrics on Melody*

In order to solve the first problem, we assumed that melody consists of segments and the lyrics should be divided into these segments. For instance, 2 bars can be handled as a segment for a song with a length of 8 bars. This is because the lyrics input by the user are not always "formatted" as poem like the usual lyrics are.

Furthermore, in most classical Japanese songs, one syllable (mora) corresponds to one note in a melody. Thus the number of notes in each segment is determined by allocating the syllables. When we consider the constraits on dividing the lyrics, the following 3 criteria can be assumed:

- A simillar number of syllables in each segments is prefered.
- The border of the segments should not be crossed over within a word.
- Too short lyrics should be iterated prior to allocation.

Under these constraits, we can solve the syllable allocation problem by using dynamic programming.

2.2.2 *Keeping Unity of Rhythm in Melody*

Even though the numbers of notes in each of the segments are decided, there still exist a large degree of freedom in rhythm. One possible way to put constraits on rhythm is to arrange that the generated rhythm belogs to a same "family" of rhythm. It is reasonable to assume that the "rhythm family" does not change in a relatively short song.

Here the second problem, that is, the requirement to keep the unity of rhythm in melody arises. To cope with this problem, we introduce a "rhythm tree" that is one rhythm has a simillar feature when one can be derived by uniting or dividing the note just one on the another. In practice, a tree structured templates of rhythm as shown in Fig. 2 are prepared by hand beforhand. Since the number of syllables in each segment corresponds to the number of notes, this tree structured template determine the rhythm in each segment.(Fig. 2)

2.3 Composition of Pitches

2.3.1 *Composition with Probabilistic Inference*

In this section, we discuss on how to obtain a pitch sequence. Although there are some discussions on the definition of melody, still it is likely to say there are certain tendency in melody. For example, in case of song, pitches of the melody would be constrained by the usual voice range of the singer. The prosody of the lyrics also impose constraints on pitch motions of the melody. As we reviewed at section 2.1, pitch motions of Japanese songs largely follow the up-ward and down-ward motions based on the prosody of the lyrics. Furthermore, chord progression, bass line of the accompaniment part and durations of each notes impose constraints on occurence and transition of pitches on the basis of écuriture of composition, such as harmony and counterpoint. Althogh exploiting these écuritures are not always indispensable to discuss how can we generate a cutting edge contemporary music automatically, still we can assume that these écuritures would secure the quality of generated songs with our algorithm for the purpose of composition support system for amateur musicians.

If a certain melody were obtained, the melody would satisfy these constraits as we disucussed above. Conversely, we can compose a song by finding the melody which optimally meets the condition. Let the pitch sequence as a sequence of MIDI note number be $X_1^N = x_1 x_2 \cdots x_N$, and the sequence of conditions on pitch sequence be $Y_1^N = y_1 y_2 \cdots y_N$, where each y_n involves chord label with annotations of scale and tonality(c_n), duration of the note (d_n), MIDI note number of the accompaniment bass (b_n), and pitch accent information, i.e. $y_n = (c_n, d_n, b_n, a_n)$. Let us also denote $P\left(X_1^N | Y_1^N\right)$ as conditional probability for X_1^N given Y_1^N which represent the tendency of pitch sequences X_1^N under condition Y_1^N. The composition of pitch for melody can be considered as finding an optimal sequence X_1^{N*} given Y_1^N which maximize $P\left(X_1^N | Y_1^N\right)$:

$$X_1^{N*} = \underset{X_1^N}{\operatorname{argmax}} \, P\left(X_1^N | C_1^N\right). \tag{1}$$

By assuming

$$P\left(x_n | X_1^{n-1}, Y_1^N\right) \simeq P\left(x_n | x_{n-1}, Y_1^N\right), \tag{2}$$

equation (1) will be as follows:

$$X_1^{N*} = \underset{X_1^N}{\operatorname{argmax}} \prod_{n=1}^{N} P\left(x_n | x_{n-1}, Y_1^N\right), \tag{3}$$

where $P\left(x_1 | x_0, Y_1^N\right) = P\left(x_1 | Y_1^N\right)$.

Since there are 128^N possible sequence of pitches, it is computationally unfeasible to search the optimal sequence by calculating probabilities for all of the possible sequences. However, we can obtain the optimal pitch sequence in order $O\left(N\right)$ using dynamic programming[6].

2.4 Implementation of the Composition System

Orpheus is an automatic composition system that we implemented using proposed algorithm for melody composition. This system computes melody from the lyrics input with choices of chord progressions, rhythm pattern,

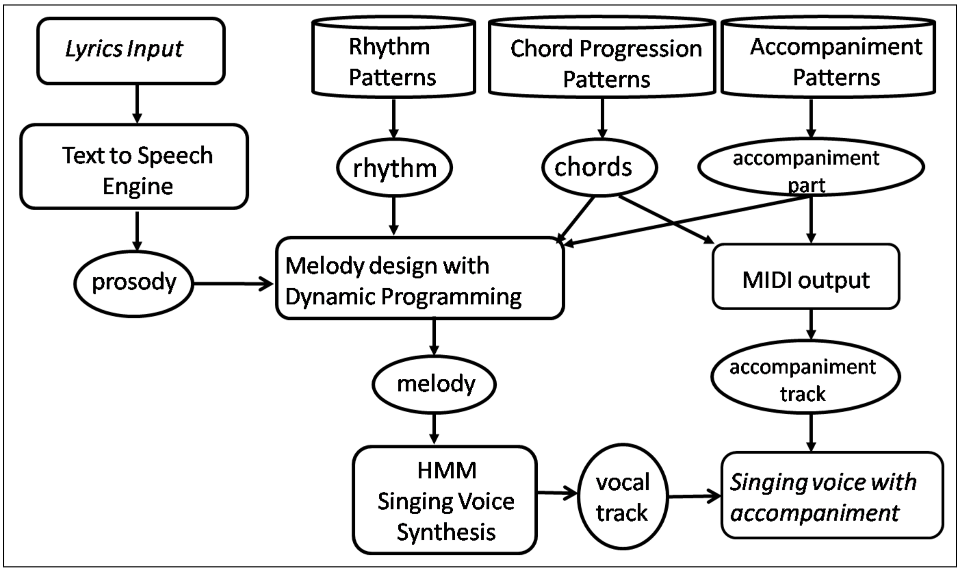

Figure 3. Flow chart of processes: *Orpheus* generates songs with the lyrics input and the choices of patterns.

and accompaniment instruments. Flow chart of the processes is shown in Fig. 3. We used Galatea-Talk[7] text-to-speech engine to analyze the prosody of Japanese lyrics, and HMM singing voice synthesizer[8] to generate the vocal part. We also implemented the system as a web-based application [3] .

3. EVALUATION RESULTS

We did two experiments to evaluate the algorithm. Firstly, we asked a classical music composer to evaluate generated songs in five-grade evaluation. The results on 59 generated songs are shown in Fig. 5 These results indicate that 83.1% of the generated pieces satisfactorily follow classical music theory, and 91.6% of the songs were voted as attractive aside from musical theory. Example of generated song is shown in Fig. 4.

Secondly, we uploaded our system to get comments from a large number of users on the internet. During a year of operation, about 56,000 songs were generated by the users and 1378 people answered the questions about *Orpheus* and the generated songs. Summarization of answers in five-grade evaluation is shown in Fig. 6 Judging from the results, about 70.8% commented that the generated songs are attractive, and 84.9% of the users had fun trying this system.

4. DISCUSSIONS

The evaluations results by a composer indicate that most of the generated songs are able to be called "melody" at least in theory. Songs that are evaluated "very poor" had irregular usages of non-chord tones which rule cannot be described with the relationship between the current note and the previous note.

Evaluation by the users on the internet suggest that our composition system is an enjoyable solution for amateur

[3] http://orpheus.hil.t.u-tokyo.ac.jp

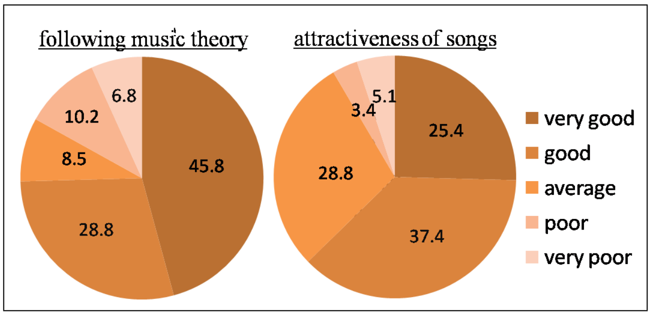

Figure 5. Evaluation results on 59 generated songs by a classical music composer. 83.1% of the generated pieces satisfactorily follow classical music theory, and 91.6% of the songs were voted as attractive aside from musical theory.

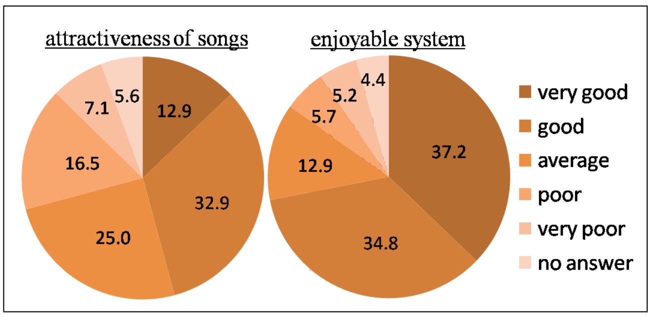

Figure 6. Evaluation results on generated songs and the *Orpheus* by 1378 users. 70.8% commented that the generated songs are attractive, and 84.9% of the users had fun trying this system.

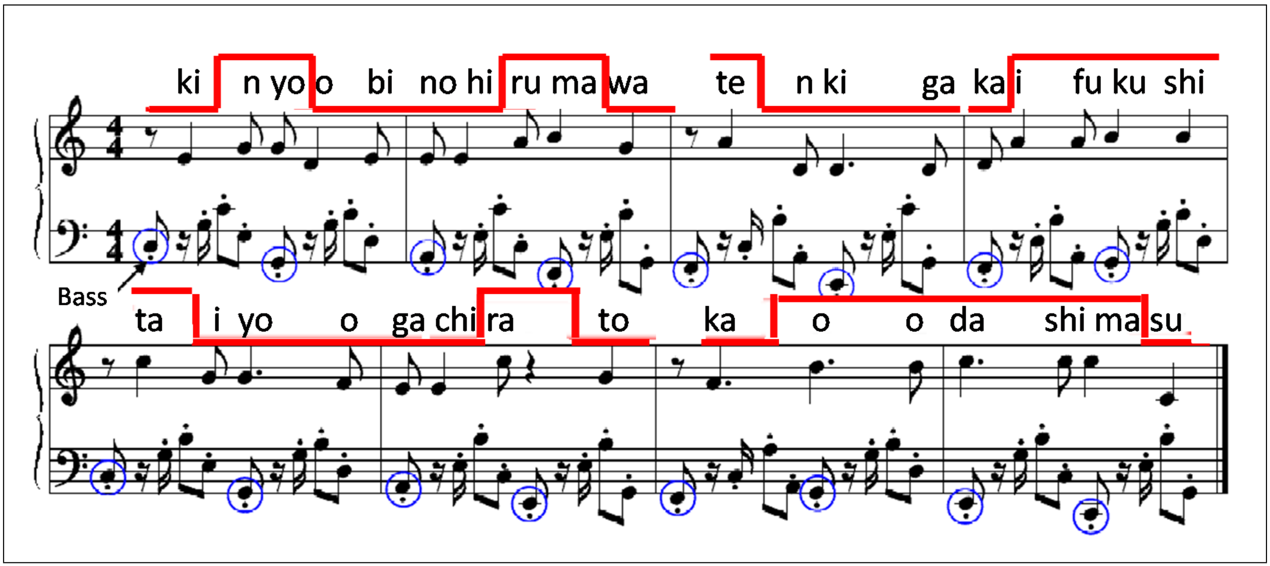

Figure 4. Example of generated song: this song was generated with the lyrics input of weather forecast. The red lines indicates the pitch accents of the Japanese lyrics. The pitch motions of the melody follows the pitch accent of the lyrics.

to compose their original songs. One reason for the result could be the directability on generating songs. Users can generate various melody by typing arbitrary lyrics since the generated song will vary based on the prosody of the lyrics. This may enabled the user not only generating a song automatically but also to generate their original songs with their original lyrics.

5. CONCLUSION

This research attempted to design an algorithm to compose a song automatically from the lyrics using prosody information, which enables users to make their original songs easily. The results indicate that our method and implemented system *Orpheus* is an enjoyable solution for amateur musicians.

However, it should be noted that our algorithm can be applied to lyrics written in "pitch accent" languages only. As a next step, we plan to extend the composition algorithm to handle "stress accent" languages, such as English, by putting constraints on metric structure of the melody.

6. REFERENCES

[1] L. Hiller and L. Isaacson, *Experimental Music*. McGraw-Hill, 1959.

[2] I. Xenakis, *Formalized Music. Revised edition*. Pendragon Press, 1992.

[3] D. Cope, *Computers and Musical Style*. A-R Editions, 1991.

[4] K. H. et al., "Automatic song composer from phrase structure of lyrics," in *Proc. of 57th Information Processing Society of Japan Annual Convention*, pp. 11–12, 1998.

[5] M. E. Beckman and J. B. Pierrehumbert, "Intonational structure in japanese and english," in *Phonology Yearbook 3*, pp. 255–309, 1986.

[6] R. E. Bellman, *Dynamic Programing*. Princeton University Press, 1957.

[7] S. K. et al., "Galatea: Open-source software for developing anthropomorphic spoken dialog agents," in *Life-Like Characters*, pp. 187–212, Springer-Verlag, 2004.

[8] S. S. et al., "A singing voice synthesis system based on hidden marcov model," in *Transactions of Information Processing Society of Japan*, pp. 719–727, 2004.