


# Development of Agents that Create Melodies based on Estimating Gaussian Functions in the Pitch Space of Consonance

Hidefumi Ohmura<sup>1</sup><sup>a</sup>, Takuro Shibayama<sup>2</sup>, Keiji Hirata<sup>3</sup> and Satoshi Tojo<sup>4</sup>

<sup>1</sup>*Department of Information Sciences, Tokyo University of Science, 2641 Yamazaki, Noda-shi, Chiba, Japan*

<sup>2</sup>*Department of Information Systems and Design, Tokyo Denki University,  
Ishizaka, Hatoyama-cho, Hikigun, Saitama, Japan*

<sup>3</sup>*Department of Complex and Intelligent Systems, Future University Hakodate,  
116-2, Kamedanakano-cho, Hakodate-shi, Hokkaido, Japan*

<sup>4</sup>*Graduate School of Information Science, Japan Advanced Institute of Science and Technology,  
1-1 Asahidai, Nomi-shi, Ishikawa, Japan  
hidefumi.ohmura@gmail.com*

**Keywords:** Music, Melody, Lattice Space, GMM, EM Algorithm.

**Abstract:** Music is organized by simple physical structures, such as the relationship between the frequencies of tones. We have focused on the frequency ratio between notes and have proposed lattice spaces, which express the ratios of pitches and pulses. Agents produce melodies using distributions in the lattice spaces. In this study, we upgrade the system to analyze existing music. Therefore, the system can obtain the distribution of the pitch in the pitch lattice space and create melodies. We confirm that the system fits the musical features, such as modes and scales of the existing music as GMM. The probability density function in the pitch lattice space is suggested to be suitable for expressing the primitive musical structure of the pitch. However, there are a few challenges of not adapting a 12-equal temperament and dynamic variation of the mode; in this study, we focus on these challenges.


## 1 INTRODUCTION

Music is essential in various cultures, and people have used music for various purposes (DeNora, 2000). It is often thought that only professional musicians create music; however, this is not true because almost everyone creates music, for example, when they hum and whistle a melody by intuition in the bathroom (Jordania, 2010). Why do people with limited musical education enjoy listening to music and creating melodies? We believe that the reason comes from the gestalt perception of humans. Music is organized by simple physical structures, such as the relationship between the frequencies of tones. Humans can understand musical structures because they can often discern the relationship between frequencies. Ledahl and Jackendoff proposed the theory to analyze music based on musical gestalt perception (Meyer, 1956; Lerdaahl and Jackendoff, 1983).

We focused on the frequency ratios of the fundamental relations between tones, and the development

of agents that create melodies as a system (Ohmura et al., 2018; Ohmura et al., 2019). The frequency ratio refers to the interval between two basic frequencies of tones and note values between pulse frequencies of the sound timing. The agents in the system produce notes based on the probability density function. There are two types of spaces, one for pitch and another for musical values. The agents have a probability density function consisting of one or two normal distributions in every two spaces. This system provides simple melodies like humming and whistling. Moreover, the system creates a structure of the musical theory, such as musical modes and complex rhythms. Therefore, it was suggested that the spaces based on frequency ratios could express musical structures quantitatively.

However, the system was only capable of creating melodies and was unable to analyze existing music. In this study, we make improvements to the system to analyze existing music and express the probability density functions of the spaces based on frequency ratios. First, we provide a system analyzing pitches of existing music. This system can read a Standard MIDI file (SMF) as existing music. The system anal-

<sup>a</sup> <https://orcid.org/0000-0003-4373-0890>

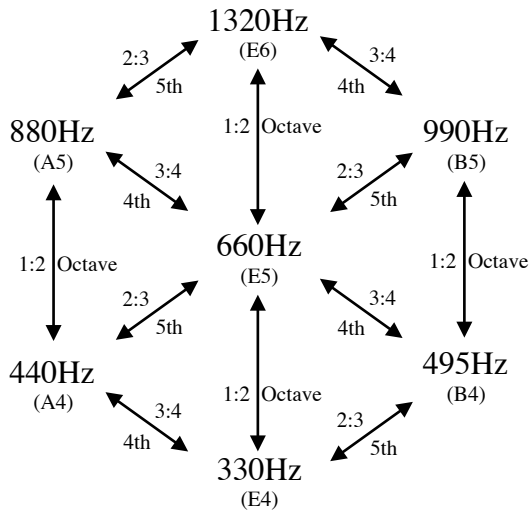


Figure 1: Relationships between pitch (interval).

yses the file and creates a density function for each melody. The system can then output melodies based on each density function. Moreover, users can perform readjustment of parameters of the distributions to control the musical structures of the outputs.

## 2 LATTICE SPACES BASED ON FREQUENCY RATIOS

### 2.1 Interval and Musical Temperament

The pitch of a note is defined by the frequency of air vibration. Real sounds consist of various frequencies, and humans recognize the lowest frequency, which is called the fundamental frequency, as the pitch of the note. The relationship between two notes, which is called the interval, is defined as the frequency ratio. There are four intervals called the perfect consonance, a unison, a perfect fourth, a perfect fifth, and an octave. Humans feel they are the best-matched interval group.

These groups are based on primitive ratios. A frequency ratio of 1:1 between two pitches creates a unison. A frequency ratio of 1:2 between two pitches creates an octave. A frequency ratio of 2:3 between two pitches creates a perfect fifth. A frequency ratio of 3:4 between two pitches creates a perfect fourth.

Figure 1 shows pitches of notes created by perfect consonance based on 660Hz (E5);

The frequency values are created from 660Hz using Pythagorean temperament, which is one of the musical temperaments. The Pythagorean temperament is only based on ratios, 1:2, 2:3, and 3:4, and its temperament provides accurate values of per-

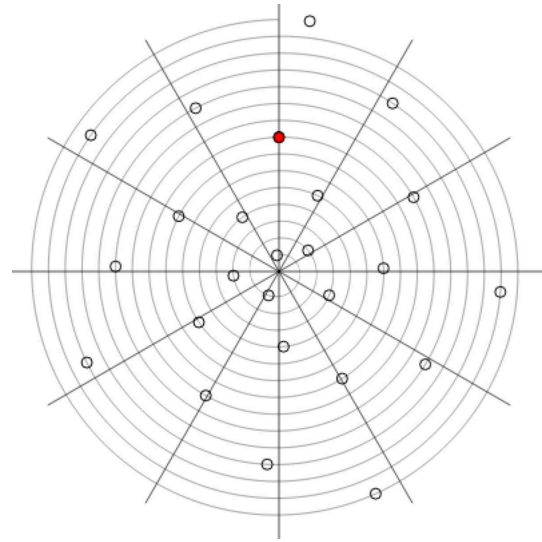


Figure 2: Comparing Pythagorean tuning with 12-equal temperament.

fect consonance. However, Pythagorean temperament does not define 12 notes because of the Pythagorean comma based on  $2^7 \neq (3/2)^{12}$ . The most popular temperament is 12-equal temperament, which divides an octave into twelfths. The 12-equal temperament treats 12 notes equally but does not provide accurate values of the consonance. Figure 2 shows differences between Pythagorean temperament and the 12-equal temperament. The whorl shows the mean values of the pitch, and the outer position is larger than the inner position. The same angle signifies octaves (1:2:4:8...). Twelve lines show the pitch notation of the 12-equal temperament. Circles show positions depending on the perfect fifths  $((3/2)^n, n = 1, 2, 3)$  from the red circle. In this study, because of SMF, we adopt the 12-equal temperament to the system.

### 2.2 Lattice Spaces

There are two spaces in the system. The first space expresses pitch frequencies, called the pitch lattice space, and the other space expresses frequencies of the sound timing of pulses, called the rhythm lattice space. Figure 3 is the expanded space from Figure 1.

Next, we explain the rhythm lattice space. In rhythm, the frequencies of pulses are vital features. When a listener hears two pulses whose relationship is 1:2, they may feel a duple meter. Figure 5 shows the relationship. When the relationship is 1:3, the listener may feel a triple meter. Moreover, when a relationship is 1:5, the listener may feel a quintuple meter. However, one generally feels a quintuple meter as a 2 + 3 meter, for the quintuple meter is relatively challenging to perceive by the human ear. Actual music

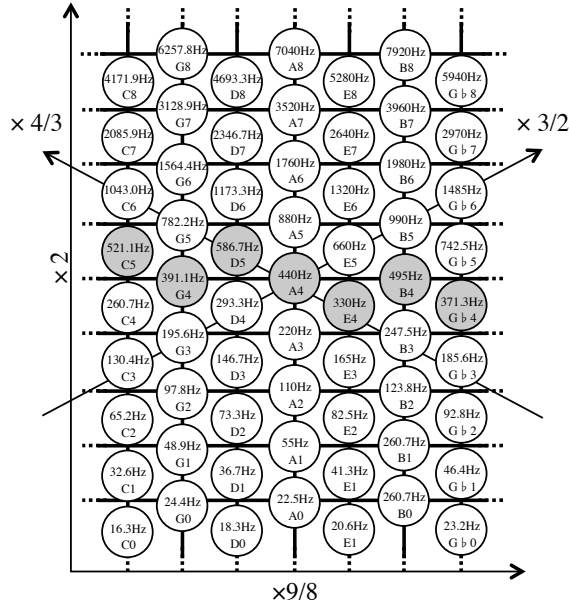


Figure 3: The pitch lattice space based on f 2:3 and 3:4.

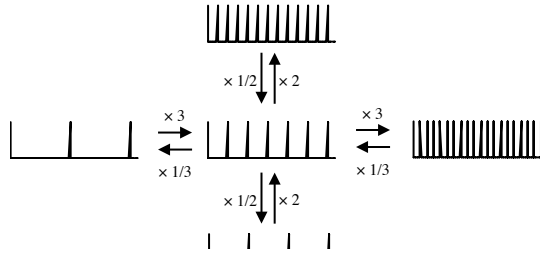


Figure 4: Relations between pulses.

consists of many pulses. A listener feels the strongest or most frequent pulse as the meter of the music, and the less frequent pulses as weak beats and up beats. Monophony, however, lacks beats, such that a listener at times may not feel any meter. For example, some pieces of the Gregorian chant provide some rhythmic interpretations, which is also true in the melodies of humming.

We provide the rhythm lattice space, which also consists of the ratios of 1:2 and 1:3 (see Figure 5). The unit in this lattice space is bpm (beats per minute). In this figure, 72 bpm is the basic frequency of the pulse. The x-axis indicates triple relationships, and the y-axis shows the duple relationships. Each point of intersection is the frequency of a pulse. In this figure, there are symbols of musical notes; a quarter note is 72 bpm.

### 2.3 Outputting Note with GMM

In the system, there are probability density functions in each space. The sound timing and pitches of an

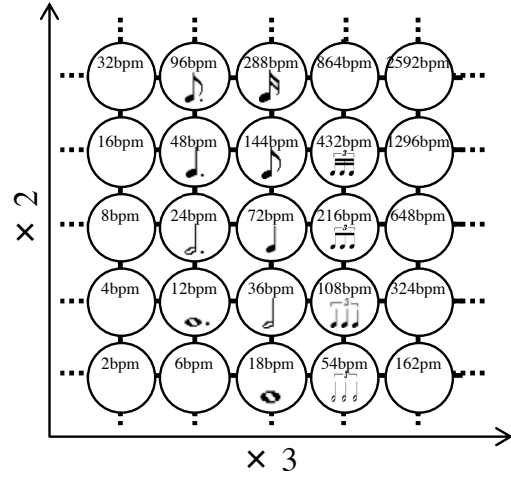


Figure 5: The lattice space for musical values with duple and triple relationships.

output note depend on each function. The probability density functions consist of one or two normal distributions.

A normal distribution is expressed by the following formula.

$$\mathcal{N}(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

$\mu$  is the mean and  $\sigma^2$  is the variance.

The function is extended to two dimensions as follows.

$$\mathcal{N}(\mathbf{x}) = \frac{1}{(2\pi)^2 |\Sigma|^{1/2}} \exp\left(-\frac{(\mathbf{x}-\boldsymbol{\mu})^T \Sigma (\mathbf{x}-\boldsymbol{\mu})}{2}\right)$$

The details of each value are as follows.

$$\mathbf{x} = \begin{pmatrix} x \\ y \end{pmatrix}, \boldsymbol{\mu} = \begin{pmatrix} \mu_x \\ \mu_y \end{pmatrix}, \Sigma = \begin{pmatrix} \sigma_x & Cov \\ Cov & \sigma_y \end{pmatrix}$$

*Cov* means a covariance.  $\rho$  means a coefficient of correlation between values on the x- and y-axis and is calculated from *Cov* as follows.

$$\rho = \frac{Cov}{\sigma_x \cdot \sigma_y}$$

Using  $\sigma_x, \sigma_y, \mu_x, \mu_y, \rho$ , the function of the 2-dimension normal distribution is expressed as follows

$$\mathcal{N}(x, y) = \frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} \times \exp\left(-\frac{1}{2(1-\rho^2)} \left( \frac{(x-\mu_x)^2}{\sigma_x^2} - 2\rho \frac{(x-\mu_x)(y-\mu_y)}{\sigma_x\sigma_y} + \frac{(y-\mu_y)^2}{\sigma_y^2} \right)\right)$$

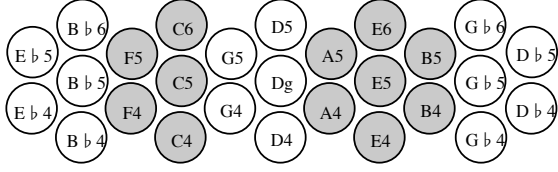


Figure 6: Miyako-bushi scale in the lattice space of pitches (Gray notations are concerned pitches).

If the agent has a normal distribution in the pitch lattice space, it can create musical modes. An agent with a normal distribution can only create a simple musical mode. If the agent needs to create a melody with a complicated mode, it must have a more complex distribution. For example, if agents create a melody of the Miyoko-bushi scale, which is a traditional Japanese mode (see Figure 6), it must have two normal distributions. For these reasons, the agent in the system has two normal distributions in each space.

When more than one normal distribution is used, there is a Gaussian mixture model (GMM), which is expressed as follows.

$$\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \mathbf{w}) = \sum_{k=1}^K w_k \cdot \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \quad (1)$$

At this moment, there are two normal distribution functions ( $K = 2$ ).  $w_k$  shows the weight of each function, and  $w_1 + w_2 = 1$ . Each agent has these parameters for creating melodies. Users can adjust the parameters with sliders of the interface of the system.

Here, we explain the flow execution of the program. When users push the play button, iterative processing occurs as follows

1. Select a pitch from the rhythm lattice space according to the probability density function.
2. Is the timing of the pulse hitting a note?
  - yes:** Select a pitch from the pitch lattice space according to the probability density function and output it.
  - no:** Do nothing
3. Go to 1 as the next step.

### 3 PROPOSED SYSTEM

In this study, we improve the existing system by adding new features. The improved system analyzes existing music and creates GMM in the pitch lattice space, and it also accepts Standard MIDI Files (SMF) as existing music. First, we explain SMF, and then we elaborate on how to fit GMM.

#### 3.1 Standard MIDI File

MIDI (Musical Instrument Digital Interface) is the standard of how to connect and share the information of the musical performance between electronic instruments. Standard MIDI File (SMF) is a file format of MIDI for saving musical data. This system analyzes the pitch data of SMF as data of existing music. There are three formats of SMF; however, the system accepts only format 1 depending on the implementation.

#### 3.2 Fitting GMM

The system considers a Gaussian Mixture model (GMM) consisting of two normal distributions. We adopt the EM algorithm as an approximation function of existing music. The Probability density function consisting of two normal distribution is expressed by formula 1. Therefore, the log-likelihood function is as follows;

$$\begin{aligned} \ln p(\mathbf{X}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \mathbf{w}) &= \ln \prod_{n=1}^N \left( \sum_{k=1}^K w_k \cdot \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right) \\ &= \sum_{n=1}^N \ln \left( \sum_{k=1}^K w_k \cdot \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right) \end{aligned}$$

Let us define  $z_{nk}$  as hidden values, which means that data  $\mathbf{x}_n$  belongs to class  $k$ . The posterior probability  $\gamma(z_{nk})$  is calculated as follows using Bayes' theorem.

$$\gamma(z_{nk}) = \frac{w_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{i=1}^K w_i \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)} \quad (2)$$

The partial differentiation of each parameter provides us with the updated formulas as follows:

$$\begin{aligned} N_k &= \sum_{n=1}^N \gamma(z_{nk}) \\ \boldsymbol{\mu}_k^{new} &= \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) \mathbf{x}_n \\ \boldsymbol{\Sigma}_k^{new} &= \frac{1}{N_k} \sum_{n=1}^N \gamma(z_{nk}) (\mathbf{x}_n - \boldsymbol{\mu}_k^{new})(\mathbf{x}_n - \boldsymbol{\mu}_k^{new})^T \\ w_k^{new} &= \frac{N_k}{N} \end{aligned} \quad (3)$$

Using two computation processes alternately, Formula 2 called the E-step and Formula 3 called the M-step, the system can find optimum values of parameters.

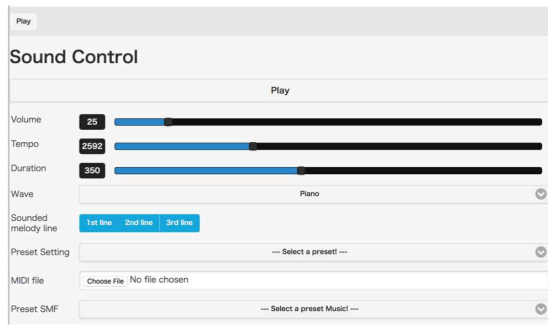


Figure 7: Sound Control Panel.

### 3.3 Implementation

We implemented the proposed agents as a system using HTML and JavaScript for creating music system<sup>1</sup>. We used a web audio API as sounding notes. Moreover, we used `tone.js`<sup>2</sup> for analyzing SMF and `tempura.js`<sup>3</sup> for executing the EM algorithm. We confirmed the operation of the system in Google Chrome.

An agent creates a melody line. The system outputs up to three melody lines because the system has three agents. Users control parameters of the probability density functions for pitch and rhythm.

We prepared some preset data for examples which provide musical modes, such as the Miyako-bushi scale (Figure 6).

The system reads SMF by analyzing the existing music and determines the pitch of each track. The system calculates the optimal parameters of GMM from the pitch data using the EM algorithm.

### 3.4 System Operating Instructions

The operational screen consists of three panels, the sound control panel, the pitch control panel, and the note-value control panel. Herein, we provide a step-by-step explanation of their usage.

#### 3.4.1 Sound Control Panel

At [Sound Control] (Figure 7), users can control play/stop, volume, tempo, duration, waveform, and melody lines of the output. The header of the operation screen also includes a play/stop button. Sliders control the values of the volume, tempo, and duration. The value of the tempo indicates the program cycle time in bpm. The value of the duration is the length of time of each note. By controlling

<sup>1</sup><http://ohmura.sakura.ne.jp/program/pitchMaker/pitchMaker010/>

<sup>2</sup><https://github.com/Tonejs/Midi>

<sup>3</sup><http://mil-tokyo.github.io/tempura/>

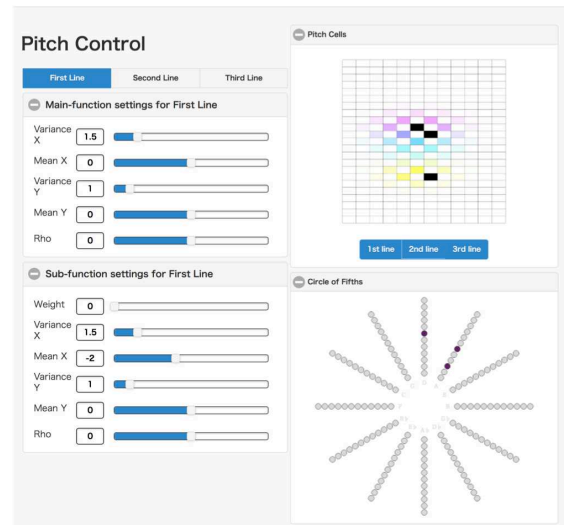


Figure 8: Pitch control panel.

this value, melodies show articulations as staccato and tenuto. With the waveform selector, users can select from “Sin,” “Square,” “SawTooth,” and “Triangle.” Users can select “Bongo” and “Piano” as actual sound source samples. The [sound control] includes a preset selector that provides each setting for the musical mode. Using ‘choose file button’, the system can be read arbitrary SMF. Moreover, the [sound control] includes a selector of preset SMF.

#### 3.4.2 Pitch Control Panel

At [pitch control] (Figure 8), users can control the parameters of each probability density function for the pitch of the melody lines using sliders. Each value of the probability density function is shown in the upper right [pitch cells]. The values of the melody lines are shown in different colors. The first line is cyan, the second line is magenta, and the third line is yellow. A darker color indicates a higher value. Using buttons at the bottom of the [pitch cells], each probability density function is set as visible or invisible. The operations of the melody lines are independent. Using the upper left buttons, the users select an operating melody line. Sliders control the parameters of the primary function in the [Main-function Settings]. The sliders control the parameters of the subfunction in the [subfunction settings]. During system execution, the selected pitches are shown at the bottom right [circle of fifth]. Therefore, users can confirm the output pitch in real-time.

#### 3.4.3 Note Value Control Panel

At [note value control] (Figure 9), users can control the parameters of each probability density function

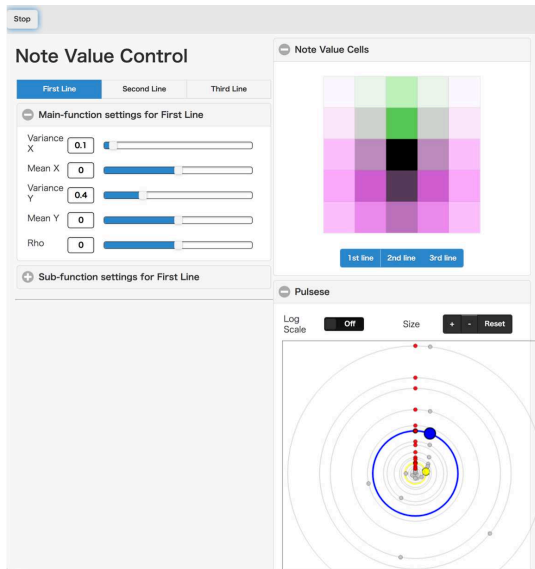


Figure 9: Note value control panel.

for the note values of the melody lines using sliders. Each value of the probability density function is shown in the upper right [note value cells]. As is the case with [pitch control], the values of the melody lines are shown in different colors. The first line is cyan, the second line is magenta, and the third line is yellow. A darker color indicates a higher value. Using the buttons at the bottom in the [note value cells], each probability density function is set as visible or invisible. The operations of the melody lines are independent, as in the case of [pitch control]. During system execution, the selected note values are shown at the bottom right [pulses]. Therefore, users can confirm the output pulses of the note values in real-time. The pulses can be zoomed using buttons and displayed on a log scale using a toggle button.

## 4 DISCUSSION

When the system reads some SMF, it creates a musical scale and mode of the existing SMF. For example, using the SMF preset, Usagi (Japanese nursery song), the system shows Figure 10 in 'Pitch Cells'. As seen from the figure, the system fits the Miyakobushi scale using two normal distribution functions. However, there are some challenges with this system, as discussed below.

For example, using the SMF preset, Debussy Prelude, the system shows Figure 11 in 'Pitch Cells'. As seen from the figure, the areas of distribution are far from each other. The reason is that this SMF is written in G-flat major, which includes Gb, Ab, Bb, B,

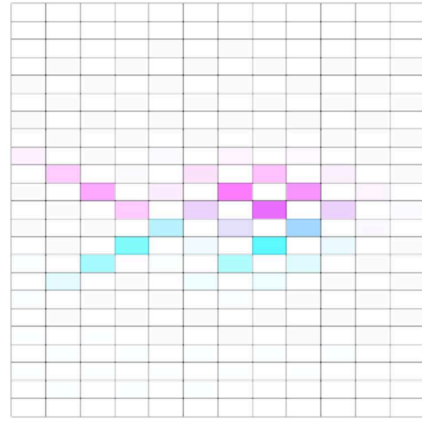


Figure 10: Distribution of Usagi (Japanese nursery song) in the lattice space for pitch.

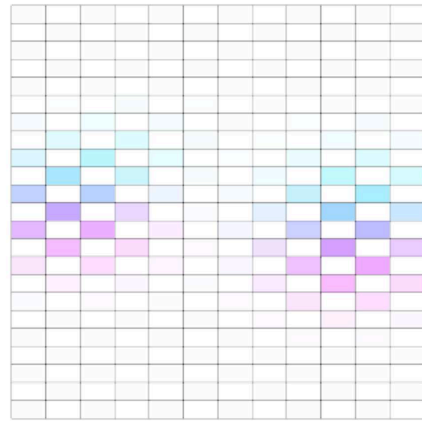


Figure 11: Distribution of Debussy 1-8 in the lattice space for pitch.

Db, Eb and F, in contrast, the center of the pitch lattice space is D. As a solution, the system analyzes the modes and scales of the existing music, then the key of the mode can be set in the center of the pitch lattice space.

Another challenge is that the spread of the pitch lattice space continues infinitely in Pythagorean tuning, and yet the spread of the pitch lattice space loop is over twelve notes. If the system targets a 12-equal temperament, we may need to adopt the von Mises distribution, which considers the direction, rather than the normal distribution. We should consider and update the system so that it treats various temperaments.

Furthermore, this system cannot express the dynamic variation of music because it reads music as a whole and creates one probability density function. For example, the probability density function of the result of Beethoven's Moonlight sonata 1 includes various notes on the x-axis because the music transposes various keys. As a solution, the system needs to

consider dynamic variations.

Additionally, the upgrade function is limited to the pitch of the music. In the future, we will add a function for the music values and rhymes to the system. When the system has this function, it will be able to consider dynamic variations.

The lattice spaces based on frequency ratios we have proposed are inspired by human musical cognition, which differs from musical scores based on creating music; therefore, the system cannot consider macro structures but can create primitive structures. The system might be able to treat not only rhythm and musical value but also musical forms such as reprises and developments.

## 5 CONCLUSIONS

We have focused on the frequency ratio between notes based on pitch and sound timings and have developed an agent that creates music using a web system. We have proposed lattice spaces that express the ratios of pitches and pulses. Agents create melodies based on the GMM of the lattice spaces. In this study, we upgraded the system to analyze existing music. Therefore, the system can get the distribution of pitch in the pitch lattice space and create melodies. We confirm that the system fits musical features, such as modes and scales of the existing music like GMM. It is suggested that the pitch lattice space and GMM are suitable for expressing primitive musical structures of pitch. However, there are some challenges of not adapting a 12-equal temperament and of dynamic variation of the mode. We are going to approach these problems in our future work.

## ACKNOWLEDGEMENTS

This work was supported by JSPS KAKENHI Grant Numbers JP17K12808, JP17K02347, JP16H01744 and JP19K00227.

## REFERENCES

- DeNora, T. (2000). *Music in everyday life*. Cambridge, UK: Cambridge University Press.
- Jordania, J. (2010). Music and emotions: humming in human prehistory. *Proceedings of the International Symposium on Traditional Polyphony (Tbilisi)*, pages 41–49.
- Lerdahl, F. and Jackendoff, R. (1983). *A Generative Theory of Tonal Music*. MIT Press.
- Meyer, L. B. (1956). *Emotion and meaning in music*. University of Chicago Press.
- Ohmura, H., Shibayama, T., Hirata, K., and Tojo, S. (2018). Music generation system based on a human instinctive creativity. In *Proceedings of Computer Simulation of Musical Creativity (CSMC2018)*.
- Ohmura, H., Shibayama, T., Hirata, K., and Tojo, S. (2019). Development of agents for creating melodies and investigation of interaction between the agents. In *ICAART2019: Proceedings of the 11th International Conference on Agents and Artificial Intelligence*, volume 1: HAMT, pages 553–569.