Development of Agents for Creating Melodies and Investigation of Interaction between the Agents

Hidefumi Ohmura[1], Takuro Shibayama[2], Keiji Hirata[3] and Satoshi Tojo[4]
[1] *Department of Information Sciences, Tokyo University of Science, 2641 Yamazaki, Noda-shi, Chiba, Japan*
[2] *Department of Information Systems and Design, Tokyo Denki University,*
Ishizaka, Hatoyama-cho, Hikigun, Saitama, Japan
[3] *Department of Complex and Intelligent Systems, Future University Hakodate,*
116-2, Kamedanakano-cho, Hakodate-shi, Hokkaido, Japan
[4] *Graduate School of Information Science, Japan Advanced Institute of Science and Technology,*
1-1 Asahidai, Nomi-shi, Ishikawa, Japan

Keywords: Music, Melody, Lattice Space.

Abstract: In this study, we attempted to construct computational musical theory by creating musical structure using physical features of sound without relying on the existing musical theory. Subsequently, we developed an agent system to create melodies. The agents can select the next note (a sound timing and a pitch) depending on the lattice spaces consisting of physical relationships (ratios) and probabilities. Further, we improve the agents which are interacting with each other in the system, and the system outputs various sounds such as music. We confirmed that the system could create structures of musical theory, such as mode, scale, and rhythm. The advantages and disadvantages of the lattice spaces are discovered.

1 INTRODUCTION

Most human beings can hum and whistle melodies in an improvisational way in their daily lives. This generation of melodies is considered a beneficial human quality for surviving in society (Jordania, 2010). Interestingly, children can also hum melodies without music education.

We considered how they create melodies and developed an agent system creating melodies (Ohmura et al., 2018). The system can provide computational musical structures such as musical scale and mode in music theory because we adopted lattice spaces depending on the physical relationships of sounds to the system.

In this study, we aim to improve the agent system including the three agents that interact with each other. We herein detail the basic elements of the previous system in creating melodies, and demonstrate the improved features. Subsequently, we discuss the outputs from the system as the music is created by the interactions between agents.

First, we propose a hypothesis on how humans create melodies such as humming and whistling, and demonstrate the physical features of sounds underlying the hypothesis. Next, we show the agent system creating melodies, and how agents interact with each other. Subsequently, we describe the system with agents and the operation of the system. Finally, we discuss music as the output of the system.

2 MUSICAL FEATURES

2.1 Hypothesis of Creating Melodies

It is typical for a musically educated person to easily select the next pitch and next sound timing of appropriateness of a present note. However, the question arises as to why both adults and children without musical education can hum or whistle melodies. We generate the following hypothesis regarding the selection of the next note except in the case of recollecting a melody. “They select purely a note of physical proximity to the present note.” Physical proximity includes two elements. The first element is the relationship between the sound timings of notes. The second element is the relationship between the pitches of notes. We considered how they create the melodies and developed an agent system creating melodies

Ohmura, H., Shibayama, T., Hirata, K. and Tojo, S.
Development of Agents for Creating Melodies and Investigation of Interaction between the Agents.
DOI: 10.5220/0007702303070314
In *Proceedings of the 11th International Conference on Agents and Artificial Intelligence (ICAART 2019)*, pages 307-314
ISBN: 978-989-758-350-6

based on the hypothesis(Ohmura et al., 2018). In this section, we explain their relationships and detail the theories of musical expectation to demonstrate how an agent selects a note.

2.2 Relationships between Pulses (Note Values)

In music, an iteration is an important primitive pattern, and a pulse is the most primitive element in a rhythm.

For example, when a listener hears two pulses (whose relationship is 1:2), he/she may feel a duple meter (see Figure 1). When a relationship is 1:3, a listener may perceive a triple meter.

Next, we consider 2:3 and 3:2 ratios combining duple and triple meters. These relationships provide the listener with a polyrhythm. A listener perceives one basic pulse as an upbeat, and another as a downbeat. In the cases of 3:4 and 4:3, a listener perceives a polyrhythm consisting of triple and quadruple meters.

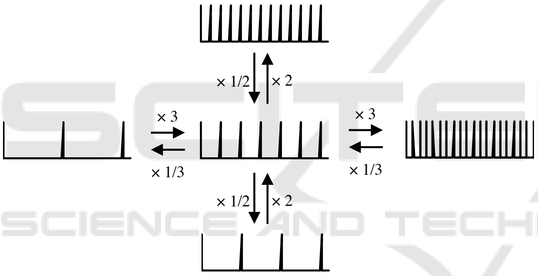


Figure 1: Relationships between pulses (note values).

Actual music consists of many pulses. A listener perceives the strongest or most-frequent pulse as the meter of the music, and less-frequent pulses as weak beats and up beats. Monophony, however, lacks beats, such that a listener at times may not perceive any meter. This is true in the melodies of humming.

2.3 Relationships between Pitches (Intervals)

In music, patterns consisting of pitches are important. These patterns are explained in musical theories of temperament (how to determine the frequency of each note) and mode (which notes to use). The value of a pitch depends on the vibrational frequency of air. Real sound consists of multiple frequencies, and we perceive the lowest frequency as the pitch, also called as the fundamental frequency. As with rhythm, the patterns of two pitches are called musical intervals, and are defined by the relationship between the frequencies.

A relationship of 1:2 creates an interval of a perfect octave, and a relationship of 1:3 creates an interval of a perfect fifteenth, which is two octaves. The relationship of 2:3 and 3:2 are combined duple and triple frequencies (see Figure 2). This pattern is called a perfect fifth. The relationships of 3:4 and 4:3 are combined triple and quadruple (see Figure 2). This pattern is called a perfect fourth, which is a consonant interval following a perfect fifth.

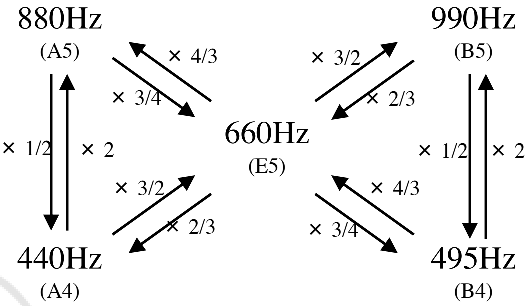


Figure 2: Relationships between pitch (intervals).

The Pythagorean scale (and principle of the Sanfen Sunyi) involve only the relationships of duples and triples. However, depending on temperament, some intervals are imprecise.

In the case of intervals, a quintuple is important. In particular, in a relationship between three pitches, a 3:4:5 creates a consonance code, called a major triad. A relationship consisting of a double, triple, and quintuple creates intonation. In this study, however, we employ only the double and triple for a simple and easy operation.

2.4 Theories of Musical Expectation

We attempted to control the musical expectations based on the theories of musical expectations (Ohmura et al., 2016). Here, we introduce these theories that provide how an agent selects the next note.

Meyer demonstrated that the deviations in expectations arouse emotions when listening to music (Meyer, 1956). This concept is based on Dewey's theory, according to which conflict causes emotions (Dewey, 1894). The deviation from the listeners' expectation when listening to music arises from a partial or complete disregard of rules that were accepted in advance. This indicates an increase in contingency because of augmented uncertainties. These uncertainties present a commonality with complexity in the optimal complexity model (Berlyne, 1971) that illustrates the relationship between complexity and hedonic values (Figure 3). This commonality suggests the existence of a relationship between uncertainty and emotion. Comparing sounds in our everyday life and their relationships reveals interesting viewpoints. We shall survey the points in Figure 3. At Position ①, the complexity is relatively low. A listener can easily predict the features of the sound, and a prediction event is likely to arise in the proposed system. For example, the pure tick-tock beat of a clock not only sounds boring, but also causes displeasure in listeners. Indeed, some clock users cannot sleep with this sound. Position ② has a higher complexity than Position ①, thus eliciting pleasure in the listener. The listener can predict the next sound, and recognize both realizations and deviations from expectation at that position. Listeners may regard sounds as musical, because the sounds comprise rules as well as deviations from the rules at this position. Different levels of complexity exist in each musical genre. Rhythms, children's songs, and folk songs have lower complexities than pop music, for example. Therefore, a wide range of sounds are under this position. Position ③ exhibits an appreciably high complexity. In this case, a listener cannot predict the next sound and can only recognize deviations from expectations. Free jazz and contemporary music are comparatively new genres that can be considered in this context. These styles of music are unpleasant for some listeners. Non-musical sounds such as the noise from a crowd at a sports field may belong to Position ③. According to Berlyne, the relationship between complexity and emotion adapts to the listener's experiences (the dotted line in Figure 3). That is, music at Position ③ elicits the pleasure of highly experienced listeners. They may feel displeasure from the music at Position ②. A listener who has experienced a large amount of music may discover the complexity in music.

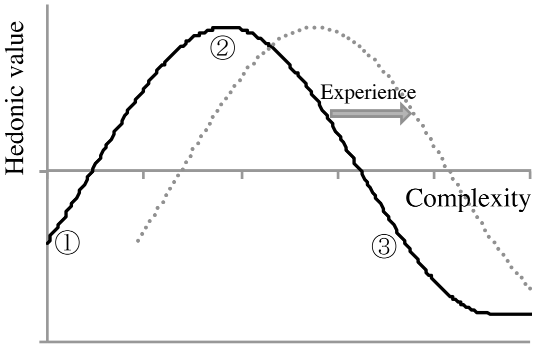


Figure 3: Optimal complexity model (modified from (Berlyne, 1971)). When a listener is musically experienced, his or her function moves to more/less complex.

As we regard complexities as uncertainty, we can calculate the next note with probability density functions. In the next section, we describe how the agent selects the next note with probability density functions.

3 AN AGENT CREATING MELODIES

To create a melody, the proposed agent selects a sound timing, and subsequently selects a pitch. These actions are based on the relationships as shown in 2.2 and 2.3. The selections depend on the probabilities that are built on the theory of musical expectation. In this section, we introduce the theory of musical expectation, and explain an agent-creating melody.

3.1 Lattice Spaces with Duple and Triple Relationships

We provide a lattice space for rhythm that consists of ratios of 1:2 and 1:3 (see Figure 4). The unit in this lattice space is beats per minute (bpm). In this figure, 72 bpm is the basic frequency of the pulse, which is located in the middle of the figure. The x-axis indicates triple relationships, and the y-axis duple relationships. Each point of intersection is the frequency of a pulse. In this figure, symbols of musical notes are depicted; a quarter note is 72 bpm.

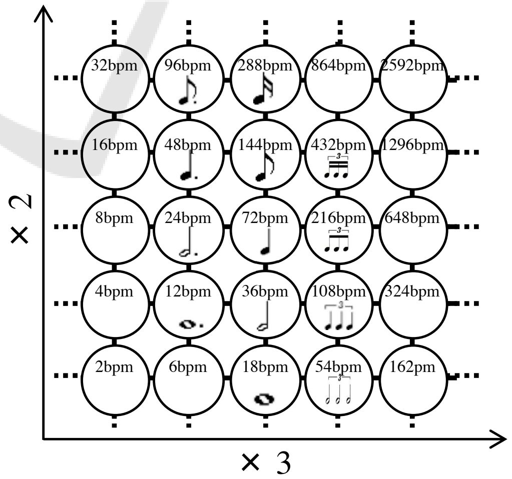


Figure 4: Lattice space for musical values with duple and triple relationships.

Next, we provide a lattice space for intervals that consists of ratios of 1:2 and 1:3 (see Figure 5). The x-axis indicates triple relationships, and the y-axis duple relationships. Each point of intersection is the frequency of a note. In this figure, 440 Hz is a basic frequency that is located in the middle of the figure. Each frequency value is rounded. Although each point of intersection indicates a letter notation with a frequency, the farther the point from the base frequency (in this case 440 Hz), the larger is the error based on the Pythagorean comma.

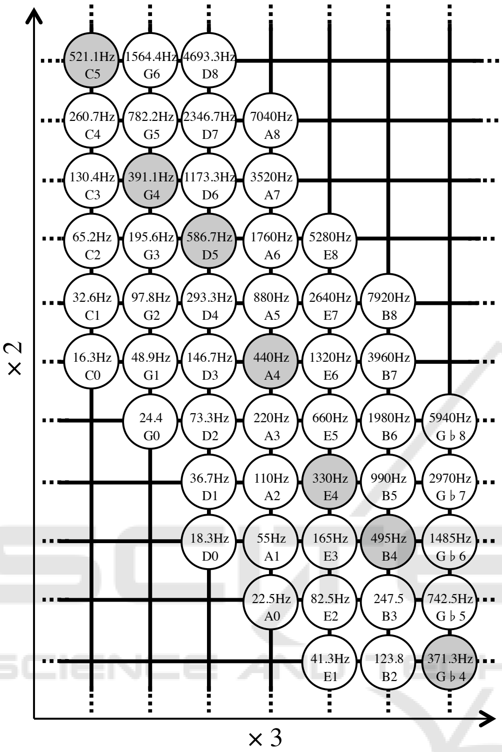


Figure 5: Lattice space for pitches with duple and triple relationships.

In the lattice space of Figure 5, the distance between one frequency and another of close value is long. Thus, the lattice space of Figure 5 is slanted as in Figure 6. This will be useful in setting the parameters.

Unfortunately, in the spaces of x-axis (×9/8) and y-axis (×2), each note does not intersect at right angles. Therefore, we create a space of x-axis (×3/2) and y-axis (×4/3) as shown in Figure 7).

3.2 Probability Density Functions

The proposed agent uses the probabilities at points of intersection using a probability density function. In this study, we employ a two-dimensional Gaussian function for the probability density function. The function is represented as follows:

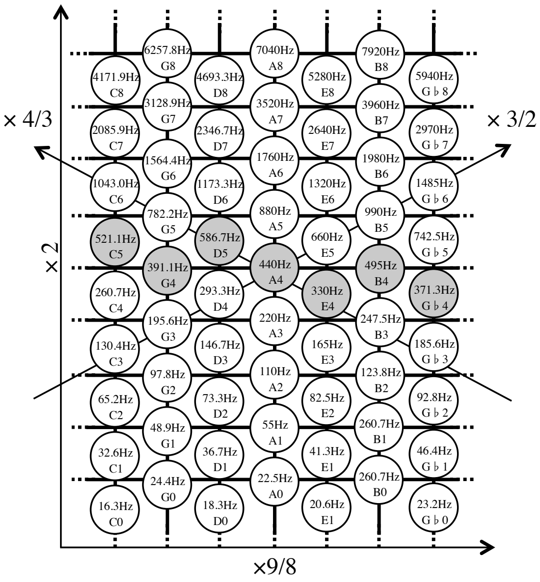


Figure 6: Lattice space of slanted Figure 5.

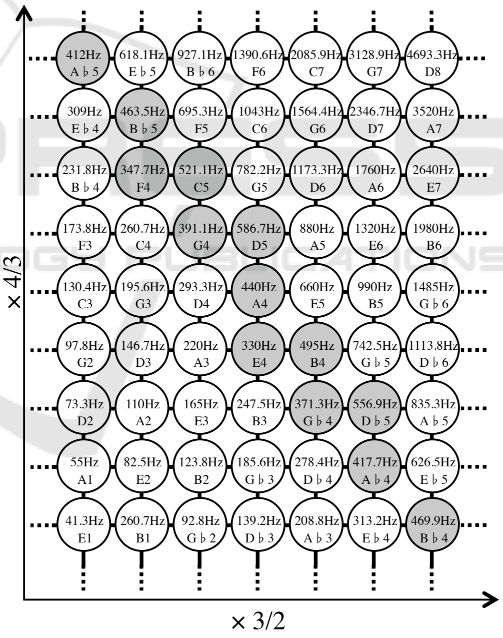


Figure 7: Lattice space of rotated Figure 6.

$$
\begin{aligned}
f(x,y) = {} & \frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} \\
& \times exp\left(-\frac{1}{2(1-\rho^2)}\left(\frac{(x-\mu_x)^2}{\sigma_x^2}\right.\right. \\
& \left.\left. -2\rho\frac{(x-\mu_x)(y-\mu_y)}{\sigma_x\sigma_y}+\frac{(y-\mu_y)^2}{\sigma_y^2}\right)\right)
\end{aligned}
$$

This equation describes a two-dimensional normal distribution with mean μ and variance σ^2 in each axis. ρ means the coefficient of correlation between values on the x- and y-axis. Assigning values to x and y, the probabilities are calculated from μ_x, μ_y, σ_x, σ_y, and ρ.

Adjusting σ, the shape of the function changes as shown in Figure 8.

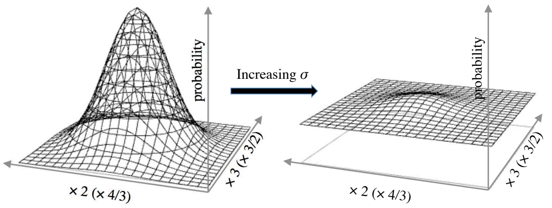


Figure 8: Shapes of the function depening on σ.

In the lattice space of pitch, when we apply a mixture distribution to a probability density function, the model expresses musical modes such as major and minor (Ohmura et al., 2016).

The agent selects a pulse based on the probabilities of the lattice space for the musical values at each step. When the selecting pulse is the timing of sound, the agent selects a pitch based on the probabilities of the lattice space for pitches. The default cycle of steps is 25 ms, which is the pulse of the least frequency in the lattice space for musical values.

3.3 Features of Multiagent System

The agents have functions of the lattice space for musical values and the lattice space of the pitch. When multiple agents exist on each lattice space, we consider two modes: (i) they use a common function, and (ii) they use an independent function. The system provides each mode.

In the independence mode of the system, each agents Gaussian functions are probability density functions. The relationships between the agents are calculated from the parameters of the functions on each lattice space. The relationship rules are determined by calculating the values. The changing parameters based on the rules can provide the interactions between agents. Herein, we define the distances and degrees between the average (μ) as the relationships between agents. We rotate the functions counterclockwise about μ of the primary function of the first melody line. We regard the transformations of the outputs as interactions between agents.

When the agents follow the hypothesis faithfully, selecting purely a physically proximate note to the present note, the system must update the methods of their functions with the present note. The system provides that as the following mode.

4 IMPLEMENTATION

We implemented the proposed agent using HTML and JavaScript for a music generation system[1]. We confirmed the operation of the system in the latest version of Google Chrome. An agent creates a melody line. The system outputs up to three melody lines because the system includes three agents. Users can control various settings including the parameters of the probability density functions for pitches and rhythm.

The system contains the lattice space for pitches such as those shown in Figure 7, and the lattice space for musical values such as those in Figure 4.

These lattice spaces contain probability density functions. The controllable parameters of the functions are the mean (μ_x, μ_y), variance (σ_x, σ_y), and correlation (ρ). Furthermore, users can also control the parameters of the subfunctions. The parameters of a subfunction includes a weight (w), which is the ratio of a subfunction to a primary function. When the weight (w) is 0, the subfunction is ignored. When the independent flag is true, users can control the parameters of three functions on each lattice space. Meanwhile, if it is false, users can control the parameters of a function because the agents use a common function on each lattice space.

When the following flag is true, it implies that the functions of the lattice space of the pitch is to be updated. We did not implement this workings in the lattice space of value because the mean is converged on the high-frequency pulses.

When the rotation flag for each lattice space is true, each μ is rotated any given degree counterclockwise about μ of the primary function of the first melody line.

We explain the flow of execution through the program. When users push a play button, the program executes iterative processing as follows:

1. Select a pulse from the lattice of value notes according to the probability density function.
2. Is the timing of the pulse hitting a note?
   yes: Select a pitch from the lattice of pitches according to the probability density function and output it.
   no: Do nothing.
3. Is it the step of rotation?
   yes: Rotate each μ on any given degree counterclockwise about μ of the primary function of the first melody line.

---

[1] https://sites.google.com/site/hidefumiohmura/home/program/icaart2019

no: Do nothing.

4. Is the following flag true?

   yes: Update μ of the lattice space of the pitch with the position of the present note.

   no: Do nothing.

5. Go to 1 as the next step

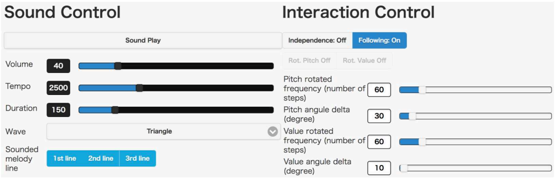
The iterative processing executes at intervals according to the tempo whose initial value is 2592 bpm.

4.1 System Operating Instructions

The operation screen consists of four panels: sound control panel, rotation control, pitch control panel, and note value control panel. Herein, we provide a step-by-step explanation on their usage.

4.1.1 Sound Control Panel

At [sound control] (Figure 9), users can control the play/stop, volume, tempo, duration, waveform, and melody lines of the outputs. The header of the operation screen includes a play/stop button. The values of volume, tempo, and duration are controlled by the sliders. The tempo value indicates the program cycle time in bpm. The duration value is the length of time of each note. By controlling this value, melodies show articulations as staccato and tenuto. With the waveform selector, users can select from "sin," "square," "sawtooth," and "triangle." moreover, users can select "bongo" and "piano" as actual sound source samples. Although each pitch is calculated by a duple and triple, the pitches of the bongo and piano are defined by 12 equal temperaments. Moreover, the [sound control] includes a preset selector that provides each setting for discussion.



Volume 40
Tempo 2500
Duration 150
Wave Triangle
Sounded melody line
Pitch rotated frequency (number of steps) 60
Pitch angule delta (degree) 30
Value rotated frequency (number of steps) 60
Value angule delta (degree) 10

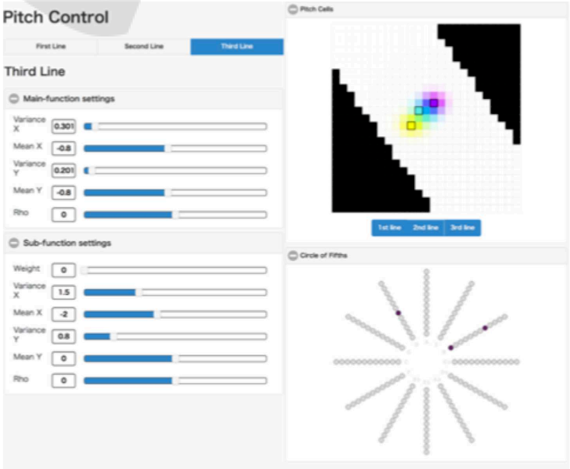
Figure 9: Sound control panel and interaction control panel.

4.1.2 Interaction Control Panel

At the [interaction control panel] (Figure 9), when users change over to the independence mode, following mode, and rotation states with the independence button, following button, Rot.Pitch button, and Rot.Value button. The header of the operation screen also includes the Rot.Pitch button and Rot.Value button. Users can set the interaction parameters between agents. Users can control the value of a rotated circle (number of steps) for a pitch with a slider. Each function of the lattice space for pitch rotates every value. Users can control the value of angular delta (degree) for pitch with a slider. The value is limited from 0 to 360. The direction of positive rotation is counterclockwise. The value of 180 means an opposite side. Similarly, users can control the parameters of the lattice space for musical values with sliders.

4.1.3 Pitch Control Panel

At [pitch control] (Figure 10), users can control the parameters of each probability density function for pitches of 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 colour indicates a higher value. Cells representing less than 20 Hz and greater than 22050 Hz are blacked out because the system cannot output these pitches. The initial values of the means are set to 440 Hz (A). Using buttons at the bottom in 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. The parameters of the primary function are controlled by sliders at the [Main-function Settings]. The parameters of the subfunction are controlled by the sliders at the [subfunction settings]. During system execution, the selected pitches are shown at the bottom right [circle of fifth]. Therefore, users can confirm the output pitches in real time.



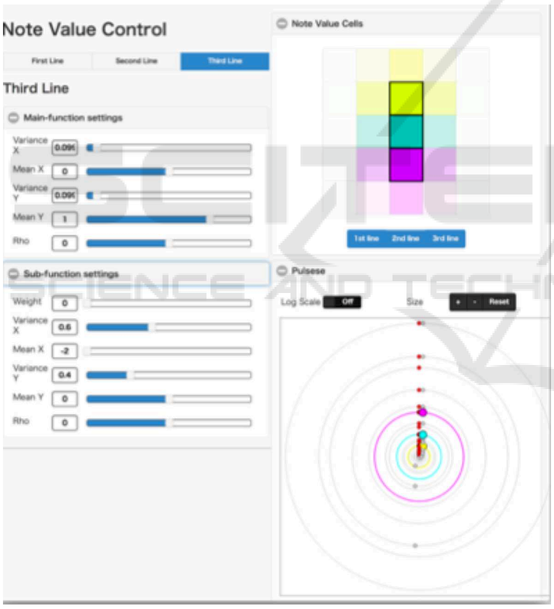
Main-function settings:
Variance X 0.301
Mean X -0.8
Variance Y 0.201
Mean Y -0.8
Rho 0

Sub-function settings:
Weight 0
Variance X 1.5
Mean X -2
Variance Y 0.8
Mean Y 0
Rho 0

Figure 10: Pitch control panel.

4.1.4 Note Value Control Panel

At [note value control] (Figure 11), users can control the parameters of each probability density function 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 colours: the first line is cyan, the second line is magenta, and the third line is yellow. A darker colour 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 with [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.



Third Line

Main-function settings
Variance X 0.09
Mean X 0
Variance Y 0.09
Mean Y 1
Rho 0

Sub-function settings
Weight 0
Variance X 0.6
Mean X -2
Variance Y 0.4
Mean Y 0
Rho 0

Figure 11: Note value control panel.

5 DISCUSSION

We prepared the example settings of the parameters as presets and discuss them herein.

As for the hypothesis, when lower values of σ exist comparatively, the outputs resemble humming melodies. [Humming (a melody line)] provides only one melody line such as a humming melody. Further, [humming (three melody lines)] provides music with three humming melody lines. The variances of the function in the lattice space of the pitches are comparatively low values, and the means are set near A. Depending on the settings, the system outputs A, E, D, B, and G frequently.

The output sounds include melodies of the pentatonic scale, which are used in the folk songs of Scotland, East Asia, and other areas. The rhythms of [humming (a melody line)] resemble duples or quadruples. The rhythms of [humming (three melody lines)] resemble music in six-eight time.

Using ρ, users can obtain the various shapes of the function. For example, when ρ is set as 0.5 in the lattice of the pitches, dispersion occurs from the lower right to upper left. If a user can control ρ freely, the system does not require transformation from Figure 5 to Figure 6 or 7. However, Figure 6 and 7 are better because users can control the important relationships between notes such as a perfect fifth and a perfect fourth. In this system, as shown in Figure 7, it is easy to calculate next note because of the intersection at right angles. However, the controlling parameters in Figure 7 are more difficult than those in Figure 6 because we are accustomed to using the relationships of 1:2.

The previous presets used only the primary function in the lattice space of the pitch. Therefore, the outputs sound like melodies of the Dorian mode. Herein, we discuss the presets using two functions in the lattice space of the pitch. In the [positive mode], the variance in the primary function is a comparatively low value, and the variance in the subfunction is higher than that of the primary function. Moreover, the mean of the subfunction moves to the bottom right from the mean of the primary function. The outputs sound like melodies of the Ionian or Lydian mode. Meanwhile, in the [negative mode], the mean of the subfunction is set opposite to the values of the [positive mode]. The other parameters of the [negative mode] are the same as those of the [positive mode]. The outputs sound like melodies of the Aeolian or Phrygian mode.

The previous two presets are symmetrical. [Rotated sample1] provides the sounds of positive modes and negative modes, alternately using half-turns. In music theory, the transformation is called relative keys. [Rotated sample2] provides some effects of rotated angles. Initially, the subfunctions are set at the lower left of the primary function. The angular variation is 45 degrees In the lattice space of the pitches, a relationship to a right cell means an upper perfect fifth, a relationship to a left cell means a lower perfect fourth, a relationship to an upper cell means an upper perfect fourth, and a relationship to a lower cell means a lower perfect fifth. That is, a direction to the lower right provides positive modes, and a direction to the upper left provides negative modes. Directions to the upper right and lower left provide octaves.

[Okinawa ¡=¿ Miyako-bushi] provides sounds such as Okinawa (Ryukyuan) music and Japanese traditional music alternately. Interestingly, they are symmetric about the mean of the primary function.

[Following a line] provides only one melody line such as a humming melody. [Following three lines] provides music with three humming melody lines. When the system activates for a long time, these outputs sound like transposes increasingly. However, these transposes are slightly exponential because the updates of means depend only on the present note. The updates may need to bypass more sounded notes.

As can be heard from the presets, the system outputs not only simple melodies (such as humming and whistling), but also melodies with musical elements such as scale, mode, rhythm, and metrical structure. These outputs reveal that the musical mode and scale are not discrete but continuous, and that rhythm structures without sequences exist such as musical scores. Moreover, some of them are symmetrical.

6 CONCLUSION

In this study, we demonstrated an agent system to generate three melodies that were based on the relationships between the physical features of notes. Further, we proposed the interactions between agents. We confirmed that the system could create structures of musical theories, such as mode, scale, and rhythm. We discovered the advantages and disadvantages of the lattice spaces. As future work, we will investigate more interaction patterns between agents, and update the lattice spaces.

ACKNOWLEDGEMENTS

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

REFERENCES

Berlyne, D. E. (1971). *Aesthetics and psychobiology*. Appleton Century Crofts.

Dewey, J. (1894). The theory of emotion: I: Emotional attitude. *Psychological Review*, 1(6):553–569.

Jordania, J. (2010). Music and emotions: humming in human prehistory. *Proceedings of the International Symposium on Traditional Polyphony (Tbilisi)*, pages 41–49.

Meyer, L. B. (1956). *Emotion and meaning in music*. University of Chicago Press.

Ohmura, H., Shibayama, T., and Hamano, T. (2016). Generative music system with quantitative controllers based on expectation for pitch and rhythm structure. *Proceeding of The Eighth International Conference on Knowledge and Systems Engineering (KSE2016)*.

Ohmura, H., Shibayama, T., Hirata, K., and Tojo, S. (2018). Music generation system based on a human instinctive creativity. *Proceedings of Computer Simulation of Musical Creativity (CSMC2018)*.