Exploring the application of gate-type quantum computational algorithm for music creation and performance

Satofumi Souma Department of Electrical and Electronic Engineering, Kobe University
1-1 Rokkodai Nada, Kobe 657-8501, Japan
ssouma@harbor.kobe-u.ac.jp

Abstract

We present our recent proposals on developing the music creation system applying the quantum gate circuit, where we consider various principles and models to apply the quantum gate circuits for music creation. In considering how we can connect between the musical note and qubit and associate musical progression with quantum gate operations, we propose two distinct approaches: the wavefunction-based approach and the measurement-based approach. The former approach is intended to translate the quantum wavefunction directly to the musical expression and provide a scheme to perform a new improvisational live performance called quantum live coding. The latter is based on the quantum gate circuits' measurement results, enabling various music creation schemes, including the quantum phase modulation model and the quantum game theoretical model.

Introduction

The role of computers in modern society is increasingly importance not just as a basis for information and communication technology, but also as a means of supporting different social activities relating to human senses, such as art and entertainment. In particular, computers have played an essential role in various algorithmic compositions, recording and performances in music. Additionally, now, with a growing interest in an essentially new computing principle, "quantum computing," thereby the prospect of music production unique to quantum computing is also expected. Several projects and conferences have been held in an attempt to apply quantum mechanical concept to music creation. For example, the proposal of conceptual correspondence between quantum mechanics and musical expression Putz and Svozil (2017), Putz and Svozil (2021), pioneering experimental and theoretical works by the University of Plymouth group Kirke and Miranda (2017), Kirke (2019), Miranda (2020), Miranda and Bask (2021). The application of gate quantum computing to counterpoint composition Weaver (2018), the application of quantum network to beat-making performance Oshiro and Hamido (2020), an interesting discussion between musicians and physicists in a project called *Quantum Music* in Serbia Ser (2018), and the recent symposium *International symposium on Quantum Computing and Musical Creativity* organized by QuTune project isq (2021).

In this chapter, we present our recent proposals on developing a music creation system applying quantum gate circuit, conducted as 2018 IPA (Information-technology Promotion Agency, Japan) unexplored target projects –gate-type quantum computer division–. We investigated the music application of quantum computing in this project by proposing the music creation principle using the quantum gate circuit and developing the quantum gate composition software based on the proposed principle. Our research is composed of two distinct issues. (1) Consideration of the music creation principle using the quantum gate circuit, which is essential to answer the non-trivial question on how we can connect between the musical note and the qubit and associate musical progressions with quantum gate operations. (2) Developing software that enables us to do the quantum gate composition comprehensively.

We propose two different music creation approaches based on quantum gate circuits. (1) An approach that applies the time evolution of the wave function of a qubit, which changes with the quantum gate operation, directly to musical expressions in an analog way. This approach relies on the use of full information on the quantum wavefunction. Therefore, we do not presume the use of an actual quantum computer unless the full information of the wavefunction is made available through quantum tomography, and instead offer our unique simulator-based music creation and performance system. This scheme is referred to as the wavefunction-based approach. As we explain later, this wavefunction-based approach can provide an interesting framework for a new musical performance called quantum live coding. (2) An approach that applies the result of measurement (wave function collapse) for the quantum state associated with quantum gate operation to the musical expressions in digitally. This is a measurement-based approach. Therefore, we can assume the use of real quantum computers. This scheme is referred to as the measurement-based approach hereafter.

Wavefunction-based approach and quantum live coding

Basic principle of music generation based on the wavefunction-based approach

First, we describe the details of the first approach: (1) The analog approach, in which one musical note is represented by one qubit or multiple qubits. More specifically, the musical pitch is represented by one qubit, and optionally other note-related quantities such as the note value can be represented by additional qubits. Keeping in mind that the quantum state $|\Psi\rangle$ of a single qubit, given by the equation

$$|\Psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\phi}\sin\left(\frac{\theta}{2}\right)|1\rangle \tag{1}$$

can be visualized using the Bloch sphere shown in Fig. 1, we propose that the angle ϕ within the x-y plane of the Bloch sphere is made to correspond to the note name within one octave. Here the discrete values in $\pi/6$ increments of ϕ correspond to each note of the piano keyboard (Fig. 1), while the other continuous angles between the above discrete values do not apply to the piano keyboard can be used to express the microtonal pitch.

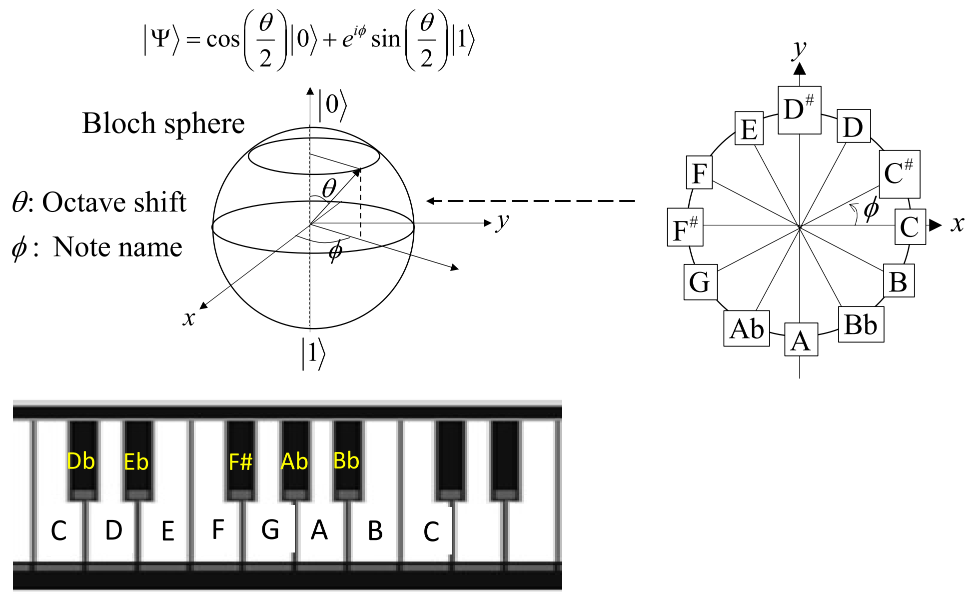

Figure 1. Schematic illustration of the correspondence between the note name and quantum bit state in the wavefunction-based (analogue) approach. The angle ϕ in the Bloch sphere (left top) is made to correspond to the note name as illustrated in the right top panel, where the discrete values in $\pi/6$ increments of ϕ correspond to each note of the piano keyboard (left bottom panel), while the other continuous angles between the above discrete values do not apply to the piano keyboard can be used to express the microtonal pitch. The angle θ is made correspond to the octave shift.

Based on this primitive idea of pitch-qubit correspondence, we propose using a multiple qubit system for music progression, such as in Fig. 2, where we use five qubits in total. In Fig. 2 we assume that two of the qubits are note qubits (qubits used to describe musical note information), while the remaining three qubits are observation (or audience) qubits (qubits that are not directly converted into the musical note). Then, for each qubit, various single-qubit gates (e.g., Hadamard gate, phase rotation gate, etc.) are applied. Two qubits gates such as controlled NOT (CNOT) gate and controlled phase gates are also provided sequentially between any two observation qubits or between one note qubit and one observation qubit. Here we assume that the quantum gate placement is designed such that the two note-qubits are not entangled with each other and can be represented as the direct products of two note-qubits states as

$$\begin{aligned} |\Psi\rangle = &\overbrace{\left(a_{000}\,|0\rangle + b_{000}\,|1\rangle\right)\left(c_{000}\,|0\rangle + d_{000}\,|1\rangle\right)}^{\text{Note qubits}} \quad \overbrace{|000\rangle}^{\text{Observation qubits}} \\ &+ \left(a_{001}\,|0\rangle + b_{001}\,|1\rangle\right)\left(c_{001}\,|0\rangle + d_{001}\,|1\rangle\right)|001\rangle \\ &+ \cdots \\ &+ \left(a_{111}\,|0\rangle + b_{111}\,|1\rangle\right)\left(c_{111}\,|0\rangle + d_{111}\,|1\rangle\right)|111\rangle\,. \end{aligned} \tag{2}$$

Then, two note-qubits are represented as two independent Bloch spheres, and two musical notes can be defined according to the above mentioned pitch-qubit correspondence. This operation (placing quantum gates) corresponds to one of the process of composition or the improvised performance of music.

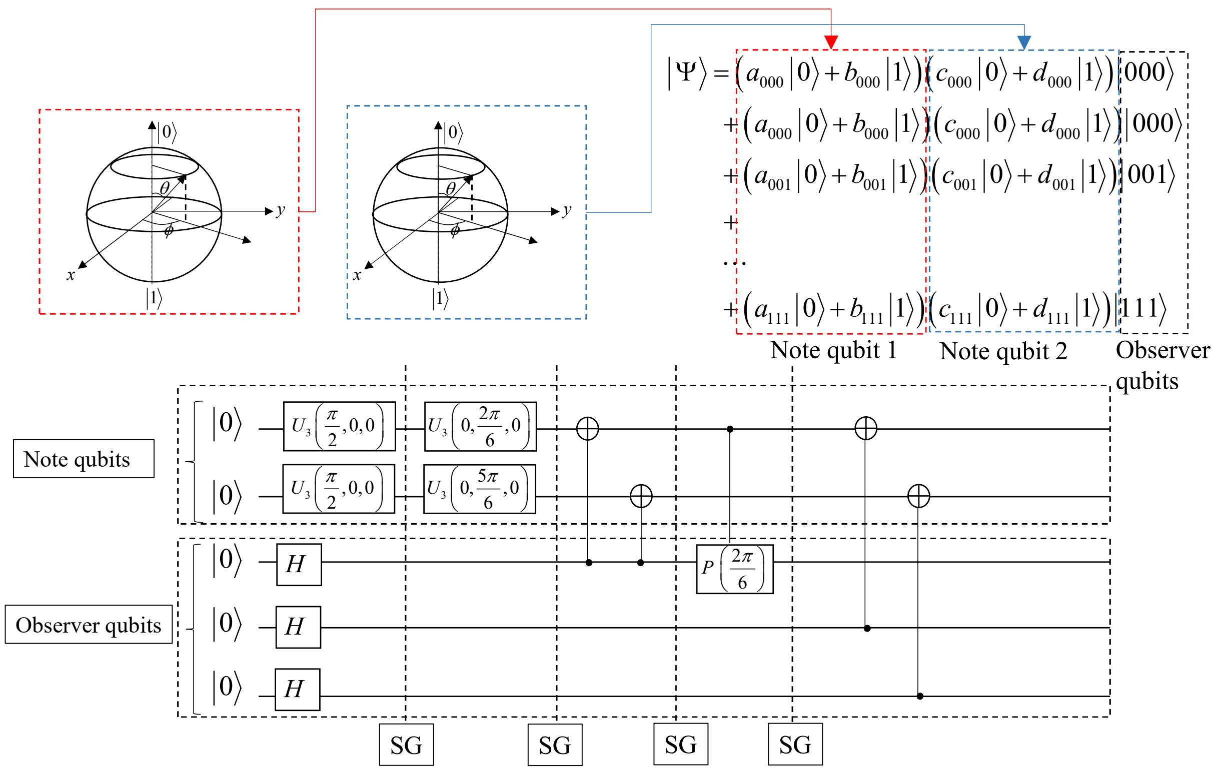

Figure 2. Schematic illustration of the correspondence between the quantum gate operation and music progression in the wavefunction-based approach. This illustration is for the two note-qubits + three observer-qubits case. Sound generation is performed at the positions of vertical dashed bars indicated as SG (sound generation) through the proposed wavefunction – note name correspondence. See the text (example subsection) for the detailed explanation of this circuit.

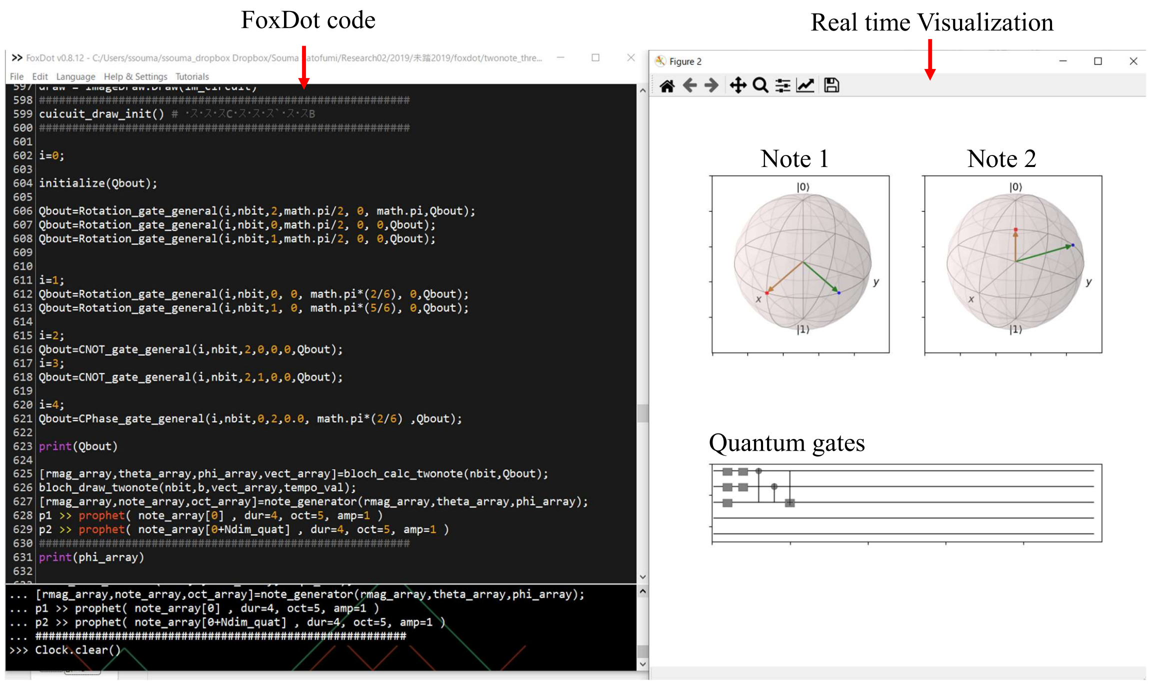

Figure 3. Screen shot of quantum live coding system developed based on FoxDot. Left part is the quantum live coding screen of FoxDot, right top part is the real time visualization of Bloch spheres, and the right bottom part is the real time visualization of quantum gate circuit schematics. Everything are intended to be shown to the audience.

In Fig. 2, the quantum state (including note and observation qubits) changes each time it passes through each gate, and then the resulting change in the note qubit is converted (translated) to the change in the musical note according to the rule above-mentioned based on the Bloch sphere, resulting in the real-time music generation. An interesting feature of the proposed "note qubit – observation qubit system" is as follows. By providing CNOT gates between the note qubit and the observation qubit, quantum entanglement can be brought about between the note qubit and the observation qubit. Then, in general, the quantum state of the note qubit differs depending on the state of the observation qubit (eight different states from $|000\rangle$ to $|111\rangle$ if three qubits are used as observation qubits), so the music we can hear will differ depending on which frame is selected as the state of the observation (audience) qubit. Furthermore, when there are two qubits categorized as note qubits as shown in Fig. 2, these two notes are used to form chords.

In our implementation, these quantum gates are improvisely placed on a live-coding system based on SuperCollider McCartney (2016) + FoxDot Kirkbride (2016). Figure 3 shows the screen shot of an example of quantum live-coding system actually developed based on SuperCollider + FoxDot. The left panel is the FoxDot coding screen to be shown to the audience, where the quantum gates are arranged in real-time (write the codes or put the prepared codes and execute). The Bloch spheres of the note qubits associated with the executed gate is designed to be automatically drawn in the upper right panel in a real-time manner, and the quantum circuit diagram is also drawn automatically in real time in the lower right. The Bloch sphere were drawn using the QuTiP library Johansson, Nation, and Nori (2012), Johansson, Nation, and Nori (2013), and the quantum circuit diagram was drawn using Tkinter library (standard GUI library of Python). In this quantum live-coding environment, various improvisational performances are possible. After the note qubit and the observation qubit form a quantum entanglement, for example, the quantum state of the note qubit can be modified by adding a phase rotation gate to the observation qubit (note–to–observer controlled phase rotation gate in the case of multiple note qubits), where non-local properties in quantum mechanics can play an important role. Furthermore, for a two-note qubit and a three-observation qubit, it is possible to obtain music reproduction with a change by selecting a different state from eight superposition states each time and performing loop playback. This means that, even if the gate arrangement as a score is the same, different music can be generated by selecting the frame at the time of measurement, which means that the audience can actively intervene in the way of enjoying music.

Example of music generation based on the wavefunction-based approach

Below we present one specific simple example of music generation based on FoxDot. Here we consider three qubits system for simplicity, composed of two note-qubits and one obserbation-qubit. Initially these qubits are initizlized to $|000\rangle$, which is given in the following FoxDot code by the one-dimensional array $\mathrm{Qbit} = [1, 0, 0, 0, 0, 0, 0, 0]$ in the basis represented in Eq. (2). Subsequently the following FoxDot code is performed (complete source code is available in GitHub Souma (2022)):

$$\mathrm{Qbit} = \mathrm{Rotation\_gate}(2, \mathrm{pi}/2, 0, \mathrm{pi}, \mathrm{Qbit}) \tag{3}$$

$$\mathrm{Qbit} = \mathrm{Rotation\_gate}(0, \mathrm{pi}/2, 0, 0, \mathrm{Qbit}) \tag{4}$$

$$\mathrm{Qbit} = \mathrm{Rotation\_gate}(1, \mathrm{pi}/2, 0, 0, \mathrm{Qbit}) \tag{5}$$

$$\mathrm{Sound\_generator}(\mathrm{Qbit}) \to \texttt{[CC]} \tag{6}$$

$$\mathrm{Qbit} = \mathrm{Rotation\_gate}(0, 0, \mathrm{pi} * (2/6), 0, \mathrm{Qbit}); \tag{7}$$

$$\mathrm{Qbit} = \mathrm{Rotation\_gate}(1, 0, \mathrm{pi} * (5/6), 0, \mathrm{Qbit}); \tag{8}$$

$$\mathrm{Sound\_generator}(\mathrm{Qbit}) \to \texttt{[DF]} \tag{9}$$

$$\mathrm{Qbit} = \mathrm{CNOT\_gate}(2, 0, \mathrm{Qbit}); \tag{10}$$

$$\mathrm{Qbit} = \mathrm{CNOT\_gate}(2, 1, \mathrm{Qbit}); \tag{11}$$

$$\mathrm{Sound\_generator}(\mathrm{Qbit}) \to \texttt{[DF]}\, |0\rangle \,/\ \ \texttt{[B}^\flat\texttt{G]}\, |1\rangle \tag{12}$$

$$\mathrm{Qbit} = \mathrm{CPhase\_gate}(0, 2, \mathrm{pi} * (2/6), \mathrm{Qbit}) \tag{13}$$

$$\mathrm{Sound\_generator}(\mathrm{Qbit}) \to \texttt{[DF]}\, |0\rangle \,/\ \ \texttt{[CG]}\, |1\rangle\,. \tag{14}$$

This code is illustrated in Fig. 2 (up to the 4th SG in Fig. 2). In this code $\mathrm{Rotation\_gate}(\mathrm{i}, \theta, \phi, \xi, \mathrm{Qbit})$ is the general rotation gate defined by

$$U_3\left(\theta, \varphi, \xi\right) = \begin{pmatrix} \cos(\theta/2) & -e^{i\xi}\sin(\theta/2) \\ e^{i\phi}\sin(\theta/2) & e^{i(\phi+\xi)}\cos(\theta/2) \end{pmatrix}, \tag{15}$$

applied to the ith qubit component of Qbit, and $\mathrm{CNOT\_gate}(\mathrm{i_c}, \mathrm{i_t}, \mathrm{Qbit})$ is the CNOT gate with $\mathrm{i_c}$th and $\mathrm{i_t th}$ being the control and target qubits, respectively. Similarly $\mathrm{CPhase\_gate}(\mathrm{i_c}, \mathrm{i_t}, \phi, \mathrm{Qbit})$ is the controlled phase rotation gate with ϕ being the phase rotation angle within the x-y plane of Bloch sphere. Now let us take a look at the above code line by line. In Eq. (3) the Hadamard gate $H = U_3\left(\pi/2, 0, \pi\right)$ is applied to the observation (third) qubit to obtain $|00\rangle\left(|0\rangle + |1\rangle\right)/\sqrt{2}$. In Eqs. (4, 5) the two note-qubits' phases are rotated to $(\theta, \phi) = (\pi/2, 0)$, giving rise to the wavefunction $\left(|0\rangle + |1\rangle\right)\left(|0\rangle + |1\rangle\right)\left(|0\rangle + |1\rangle\right)/\sqrt{2^3}$ and the notes' name [CC], subsequently in Eqs. (7, 8) they are further rotated by $\phi = 2\pi/6$ and $5\pi/6$, respectively, giving rise to the wavefunction $\left(|0\rangle + e^{i2\pi/6}|1\rangle\right)\left(|0\rangle + e^{i5\pi/6}|1\rangle\right)\left(|0\rangle + |1\rangle\right)/\sqrt{2^3}$ and the notes' name [DF]. We note that in Eqs. (4-8) the wavefunction of the three qubits system can be written in general as $\left(a|0\rangle + b|1\rangle\right)\left(c|0\rangle + d|1\rangle\right)\left(|0\rangle + |1\rangle\right)/\sqrt{2}$ with a, b, c, and d being the complex number coefficients. Next in Eqs. (10) and (11), CNOT gates (controlled by the observer-qubit and targeted to two note-qubits) are applied. Then, in the above wavefunction $|0\rangle$ and $|1\rangle$ in the note-qubits are interchanged only when the observer qubit is $|1\rangle$, and the wavefunction becomes $\left(a|0\rangle + b|1\rangle\right)\left(c|0\rangle + d|1\rangle\right)|0\rangle/\sqrt{2} + \left(b|0\rangle + a|1\rangle\right)\left(d|0\rangle + c|1\rangle\right)|1\rangle/\sqrt{2}$, meaning that the note-qubits and observation-qubit are entangled with each other. Here, the note-qubits' states in the 2nd term can be interpreted as being rotated as $\theta \to \theta - \pi$ and $\phi \to \pi - \phi$ in the Bloch sphere ($\phi \to -\phi$ change when projected to the x-y plane of Bloch sphere) compared with the 1st term. In the note name representation this entangled state is written as $|\mathtt{DF}\rangle|0\rangle + \left|\mathtt{B^\flat G}\right\rangle|1\rangle$. Therefore depending on which state ($|0\rangle/|1\rangle$) the observer choose, the different sounds ([DF]/ [$B^\flat$G]) are produced. Finally in Eqs. (13) the controlled phase rotation gates (the 1st note-qubit is the control qubit and the observer-qubit is the target bit) with the phase $\phi = 2\pi/6$ is applied. This gate operation gives the phase rotation in the 1st *note-qubit* by the phase $\phi = 2\pi/6$ only when the *observer-qubit* is $|1\rangle$ through the phase kickback mechanism, resulting into the quantum state in the note name representation as $|\texttt{DF}\rangle\,|0\rangle + |\texttt{CG}\rangle\,|1\rangle$. The note name modulation by the controlled phase rotation gates introduced here enables us to modulate the note name in a way depending on the states of the observation qubits within the quantum superpositioned states. This is the key issue to generate wider variety of music progression patters forming quantum superpositioned states, especially when there are multiple observation qubits.

Measurement-based approach

In this section, we present the principle and examples of the measurement-based approach, and describes how the measurement result for the quantum state associated with quantum gate operation is applied to musical expressions.

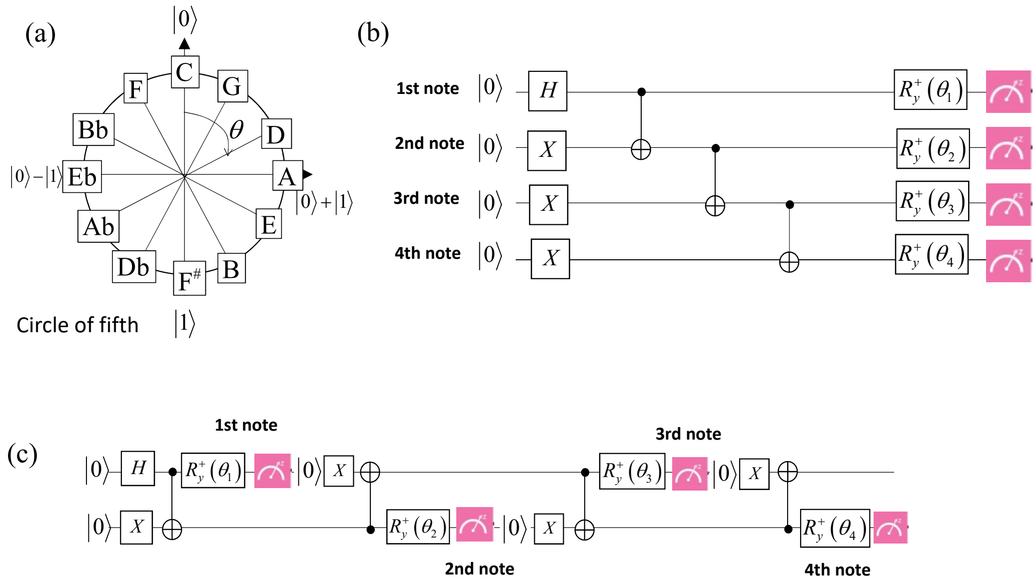

Figure 4. Circle of fifth (top left). An example of quantum gate circuit to generate stochastic music with entabglement (top right). Reduced quantum gate circuit (bottom).

Stochastic note generation model

As a way to express the note name in the measurement-based digital approach, we first propose to make the measurement axis of one qubit corresponds to a user-determined specific direction (angle) in the circle of fifths (Fig. 4(a)), so that the probabilistic realization of ↑ and ↓ along the measurement axis is translated to the probabilistic realization of two different notes that differs by the angle π in the circle of fifth. We propose a method for obtaining stochastic note sequences from the measurement results of the quantum gate circuit shown in Fig. 4(b) as one of the most simple possible application example based on this scheme.

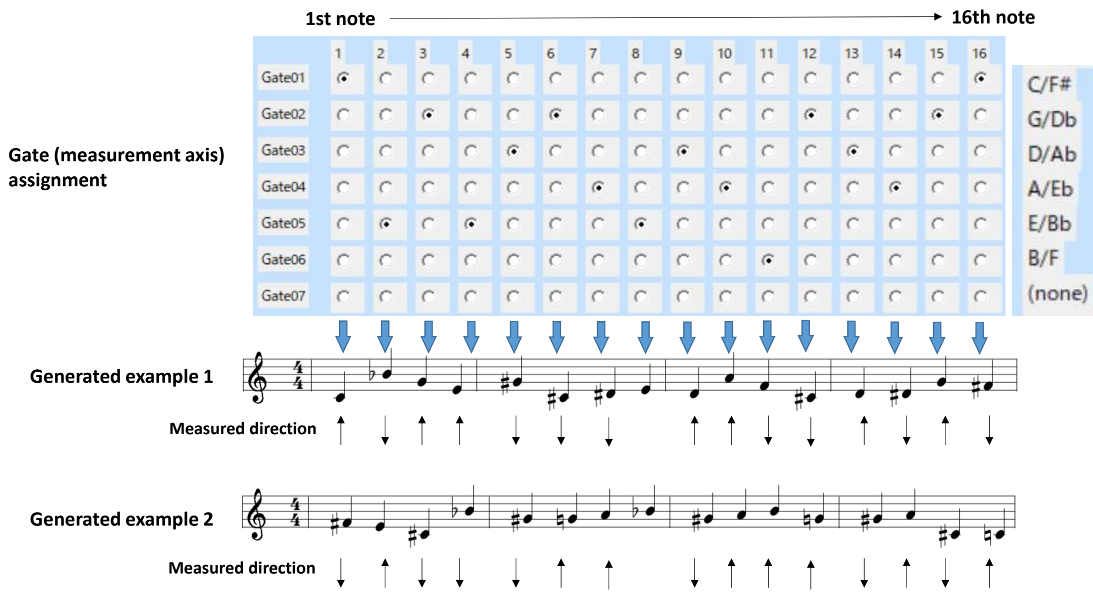

Figure 5. Graphical user interface (GUI) screen to assign the measurement axes required for all the notes (top). Two different examples of the actually generated note sequences (bottom).

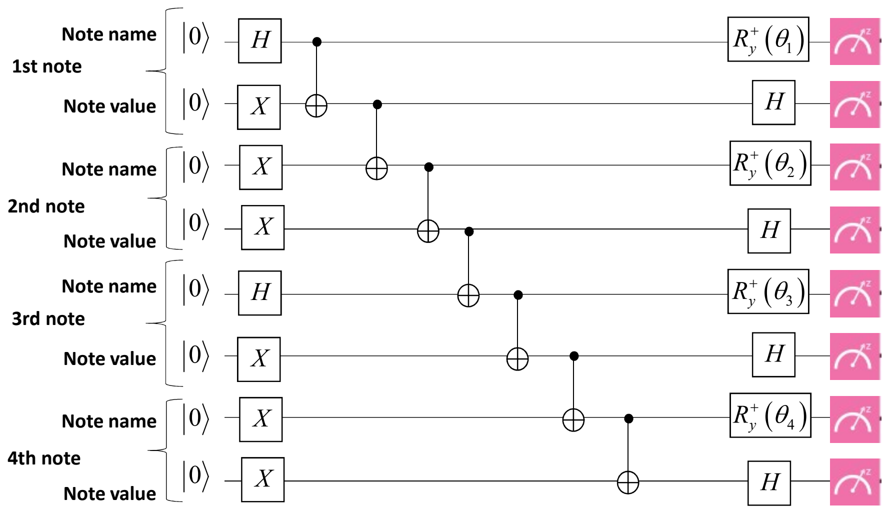

Figure 6. Quantum gate circuit for two qubits – one note correspondance model, where one qubit is for the note name and the other one for the note value (note length).

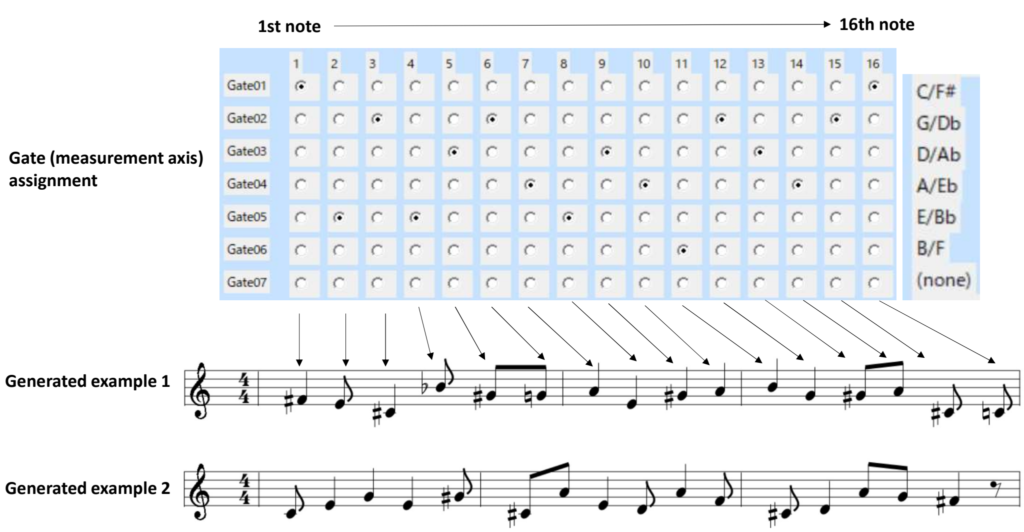

Figure 7. GUI screen to assign the measurement axes in the two qubits – one note correspondence model (top). Two different examples of the actually generated note sequences (bottom).

In Fig. 4(b), the measurement axis of the each ith note is set by the phase θ_i of the phase rotation gate provided before the measurement for each qubit (this is the information given by the user, which corresponds to a part of the composition). In the example shown in Fig. 4(b), quantum entanglement is generated by the CNOT gate between adjacent sounds (adjacent qubits). Here, since the qubits connected by the CNOT gate are only between two consecutive qubits, one can reduce the total number of required qubits to two as shown in Fig. 4(c) regardless of the total number of notes, where the qubits are once measured they are initialized to $|0\rangle$ and used again to describe the next note.

In Fig. 5, we show a part of the graphical user interface (GUI) developed for this purpose, where users must first assign measuring axes information to all the notes. Then the assigned information is sent to Qiskit (an open-source software development kit for quantum computing) qis (2021). Once the quantum computation is done, the measurement results are in turn sent to another software Takt: a text-based music programming language Nishimura (2014a)Nishimura (2014b)Nishimura (1998) to playback the generated music and optionally, generate the MIDI file. The bottom panel of Fig. 5 shows two different examples of the actual generated note sequences. We note that every time the measurement is performed, the different measurement results, and thus different note sequences are generated. The arrow in the bottom panel of Fig. 5 indicates the direction each note was measured along the pre-assigned measurement axis. Here, adjacent notes are entangled in quantum mechanical sense.

As an extension of this method, we next consider the model to take into account not only the note name but also the note value in the quantum gate composition. In this model, we use two qubits for one each note, where one is used for the note name, and the other is used for the note value, as illustrated in Fig. 6. Regarding the qubit representing the note value, a quarter note is assigned when the measured result of the note value bit is $|0\rangle$, while a half note is assigned when the measured result of the note value bit is $|1\rangle$. As shown in Fig. 6, the note value of one note is intended to be entangled with the note name of the next note. Here, the quarter and the half notes are generated in a seemingly random form with some entangled correlation. An actual measurement example is shown in Fig. 7.

Digital note-expression model

We next consider another possible note-expression model within the measurement-based approach, called the digital note-expression model. As shown in Fig. 8(left), one note name is represented by three qubits. The 1st, 2nd, and 3rd digits correspond to the raising or lowering of the note at intervals of V, II, and III degrees, respectively. Here the order of notes is based on the concept of the circle of fifth (right panel), which is a way of organizing the 12 chromatic pitches as a sequence of perfect fifths. This scheme makes it possible to generate music quantum mechanically compatible with the standard theory of tonal music. For example, we can use the same quantum gate algorithm for different keys in accordance with the circle of fifth shown in Fig. 8(right).

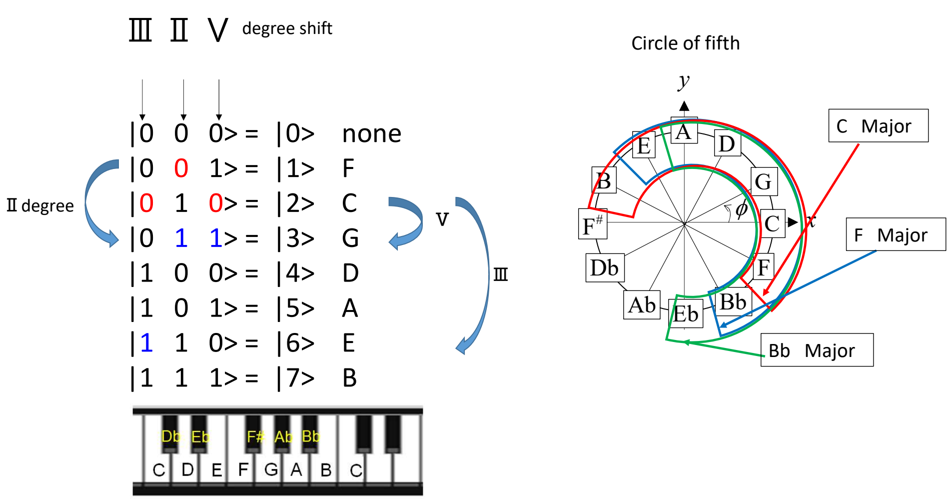

Figure 8. Digital note expression (left) and circle of fifth (right). One of the advantages of using the concept of circle of fifth for the qubits–note name correspondence is that the same algorithm can be used in different keys (for example, C major and F major) as depicted in the right panel.

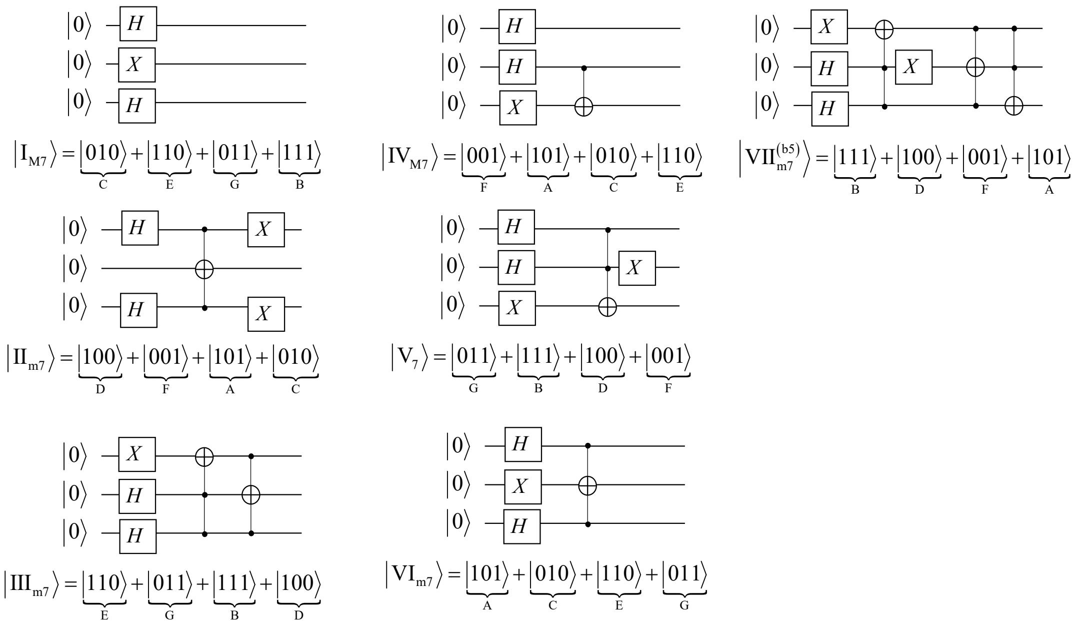

Figure 9. Possible example of the quantum gate circuit that can produce chord tones of various diatonic codes.

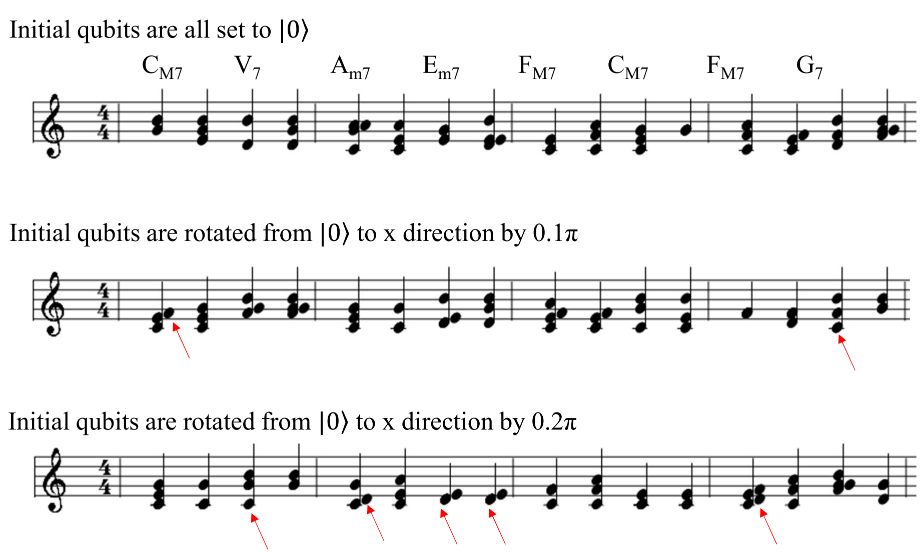

Figure 10. Generation examples of chord tones corresponding to the given chord progression. The top one is the generation example following the given chord progression in the absence of noise, while the bottom two case are in the presence of the noise. Notes indicated by arrow are out of the diatonic chord.

Diatonic chord generation model

Based on this digital note-name expression model, we devised quantum gate circuit models that can create a superpositioned state of each diatonic chord tone (four notes chords), and developed a program to generate probabilistic phrases that match a given chord progression pattern. Fig. 9 shows the quantum gates that produces chord tones of seven major diatonic chords. When a measurement of three qubits is made for the state of passing through these gates, the result is measured in one of the four chord tones if there is no noise. For example, as shown in Fig. 10, after giving (determining) the chord progression: C_{M7}-V_7-A_{m7}-E_{m7}-F_{M7}-C_{M7}-F_{M7}-G_7, the quantum gates are arranged corresponding to the chord progression of each bar, and the measurement is performed sequentially for each bar. If the measurement is performed several times (for example, three times) and recorded, and the recorded results for three measurements are simultaneously played, the progression of the chord can be obtained as in the top pattern of Fig. 10. This top panel is the case when the input state is exactly prepared in $|000\rangle$, but if the input state slightly deviates from $|000\rangle$ (corresponding to noise), the measurement result also slightly deviated from the perfect chord tone, as shown in the middle and bottom pattern of Fig. 10. However, the deviation can give the music a unique tension and depth as a musical tension. Aiming and obtaining such a tension sound (as a gate operation) is an issue for the future research, but since it is desirable that the tension sound also appears at an appropriate frequency, the noise as obtained in Fig. 10 is preferable to some extent.

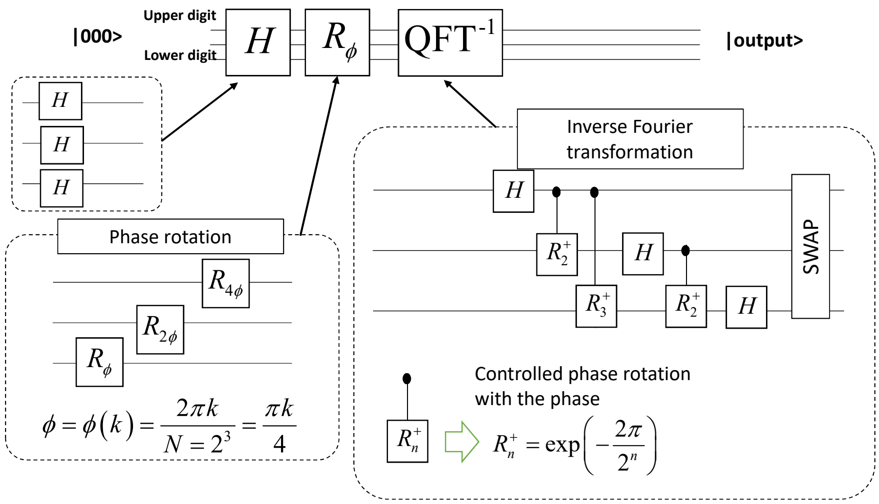

Figure 11. Basic unit of the quantum gate circuit used in the phase-modulated music generation model and its conceptual explanation especially on the role of phase in this circuit.

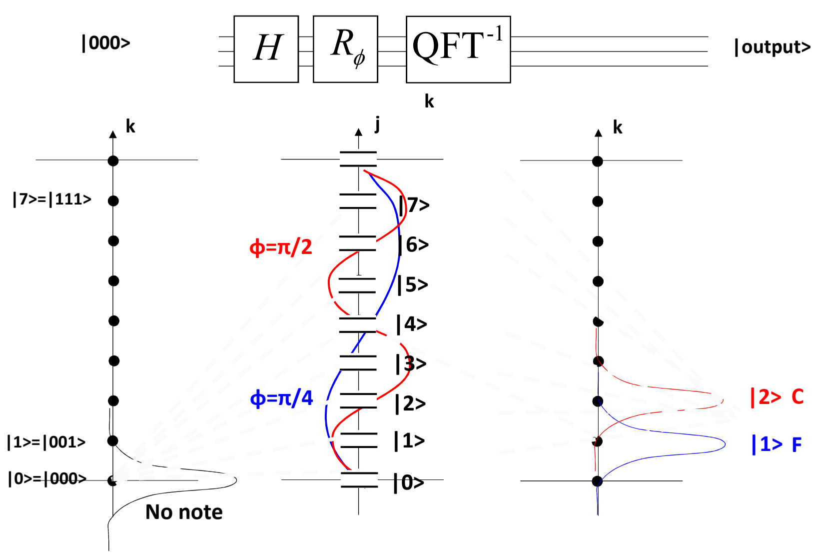

Figure 12. Conceptual explanation of the phase implementation and the note generation principle.

Phase-modulated music generation model

We next propose a phase-modulated music generation model using a digital note-name expression and a phase estimation algorithm, one of the typical quantum algorithms, as one of more complicated examples implemented Qiskit + Takt (including GUI). In this model, the three qubits representing one note (initially initialized to $|000\rangle$) are first passed through the Hadamard gate, and then the phase evolution (phase rotation gate R_ϕ) corresponding to the note name aimed at each digit of the three qubits is performed. Finally, by passing through the inverse quantum Fourier transform, the quantum state corresponding to the desired note-name can be obtained resonantly. The actual quantum gate and the conceptual explanation of the phase's role in this model are shown in Figs. 11 and 12, respectively.

It is possible to determine the note-name sequence in a non-independent manner by using this basic mechanism and changing the phase of the next note by the output of the quantum state of one previous note by the control phase gate as shown in Fig. 13. Here, the basic flow of note names is determined by the phase rotation gate R_ϕ (located immediately after the Hadamard gate) given inside the three qubit system that constitutes each note. However, the existence of a control phase gate from one note to the next plays the role of disturbing this predetermined basic flow of note names (original song). In other words, the original song is the theme, and the song modulated by the control phase gate can be regarded as a variation of the theme. Fig. 14 shows an example in which a certain theme is given and a variation is obtained with a controlled rotation gate. The topmost note sequence in Fig. 14 is the theme (given note sequence), and the other cases below are phase modulated pattern (variations), where the modulation strength q is largest for the bottom pattern. Here we note that different variations can be obtained for each measurement even with the same degree of modulation strength q (see the results for $q = 0.05$ in Fig. 14). Since the deviation from the original theme increases as the modulation is strengthened, it is possible to control how much the given theme changes by changing the modulation strength. Furthermore, since the phase modulation in this scheme is applied from one note to the next note, the deviation from the original theme increases as the music progresses in general as demonstrated in Fig. 14.

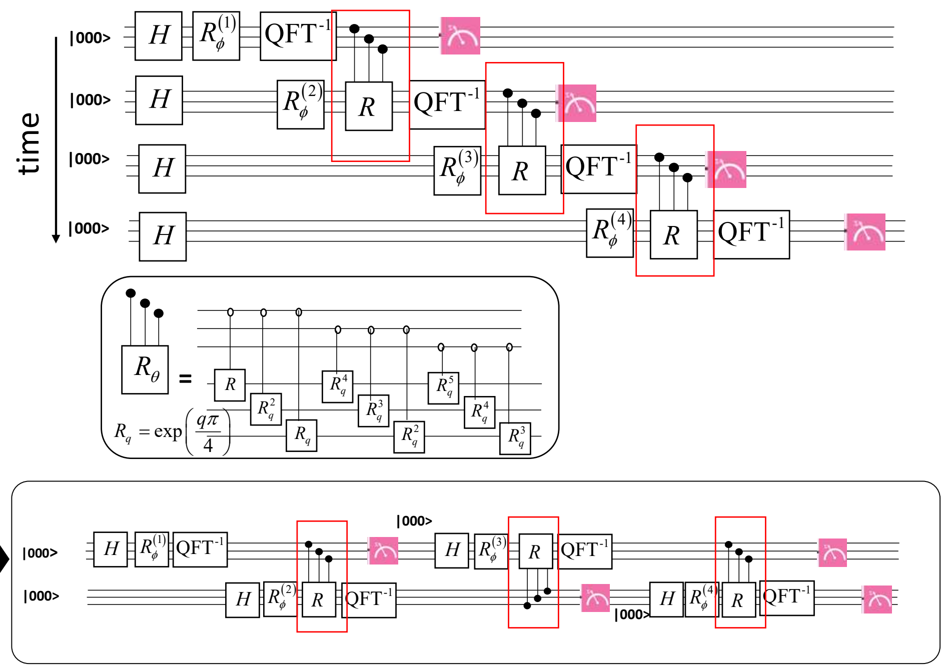

Figure 13. A music generation model using the phase modulation from one note to the next (top). The bottom panel is the reduced (equivalent) circuit model. When the value of the phase in the inter-note controlled phase rotation gate (middle panel) is $q = 1$, one note $|k\rangle$ adds the value k to the next note in the decimal expression. For example, if the original sequence pattern is $|k_1\rangle \rightarrow |k_2\rangle$, the modulation with $q = 1$ results in the sequence pattern $|k_1\rangle \rightarrow |k_2 + k_1\rangle$.

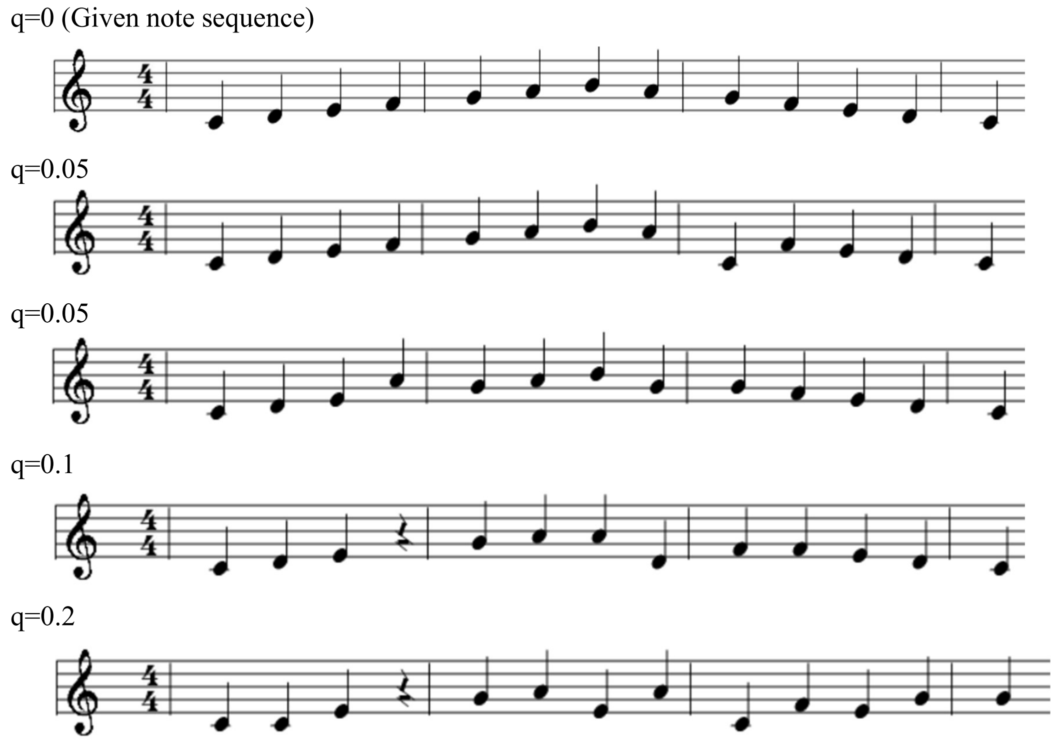

Figure 14. Examples of the generation of the variation for a given theme, where the quantum phase modulation model is employed. The topmost note sequence is the theme (given note sequence), and the other cases below are phase modulated pattern (variations), where the modulation strength q is largest for the bottom pattern.

(a)

Pareto optimal — Nash equilibrium

Prisoner's dilemma model	Bob C $\|0\rangle_B$	Bob D $\|1\rangle_B$
Alice C $\|0\rangle_A$	$(\pi_A(C,C),\pi_B(C,C))$ =(3,3)	$(\pi_A(C,D),\pi_B(C,D))$ =(0,5)
Alice D $\|1\rangle_A$	$(\pi_A(D,C),\pi_B(D,C))$ =(5,0)	$(\pi_A(D,D),\pi_B(D,D))$ =(1,1)

(b)

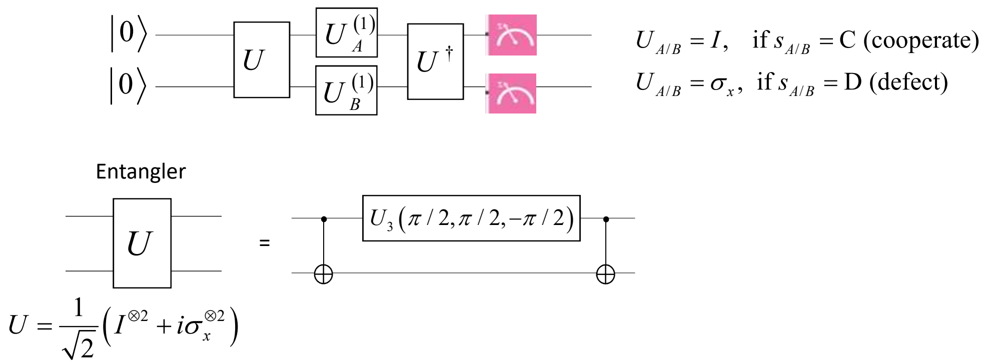

Figure 15. Schematic explanation of the classical and quantum game theory scheme. (a) is the payoff table of the prisoner's dilemma. (b) is the quantum gate circuit model used in the quantum version of the prisoner's dilemma (see the text for detail).

Two players model based on Quantum Game Theory

So far, our proposals have assumed the creation and performance of music by a single player. In this subsection, we consider the possibility of extending the basic principle of quantum mechanical music creation to the case of the multiple players model by employing the concept of quantum game theory. Game theory studies mathematical models of strategic interactions among rational agents. It has a long history of research, which began in 1944 with the book *The Theory of Games and Economic Behavior*, by John von Neumann and Oscar Morgenstern von Neumann and Oscar (1944). One of the classic representative examples analyzed by the game theory is the prisoner's dilemma, a paradox in decision analysis in which two individuals acting in their self-interests do not produce the optimal outcome. We first explain the essence of the classical prisoner's dilemma briefly and later consider its quantum generalization with the reference to Grabbe (2005). Suppose that there are two players: Alice and Bob, and they can choose one of two strategies: cooperate (C) or defect (D). When Alice's move is s_A and Bob's move is s_B with s_A and s_B are C or D, the payoff value to Alice and Bob are given as $\pi_A\left(s_A, s_B\right)$ and $\pi_B\left(s_A, s_B\right)$, respectively. Here the actual values of $\pi_A\left(s_A, s_B\right)$ and $\pi_B\left(s_A, s_B\right)$ are given in the table in Fig. 15. Then the Nash equilibrium is defined by the strategy $\left(s_A^*, s_B^*\right)$ which satisfies the conditions $\pi_A\left(s_A^*, s_B^*\right) \geq \pi_A\left(s_A, s_B\right)$ and $\pi_B\left(s_A^*, s_B^*\right) \geq \pi_B\left(s_A, s_B\right)$ simultaneously. From Fig. 15 we can see that the Nash equilibrium is established when both of Alice's and Bob's moves are defect so that $\left(s_A, s_B\right)=(\mathrm{D}, \mathrm{D})$, while actually the strategy $\left(s_A, s_B\right)=(\mathrm{C}, \mathrm{C})$ is optimal for both players, called the Pareto optimal. Now let us consider this classical prisoner's dilemma from the viewpoint of the quantum formulation. To do this, we consider the quantum gate circuit depicted in Fig. 15(a), which is composed of the entangler U, disentangler $U^{\dagger}$, and the single-qubit gates U_A and U_B for Alice and Bob, respectively. Here the two qubits gate U is given as given by

$$U=\frac{1}{\sqrt{2}}\left(I^{\otimes 2}+i \sigma_x^{\otimes 2}\right), \tag{16}$$

while the single qubit gate $U_{A/B}$ is given as $U_{A/B}=I$ if $s_{A/B}=\mathrm{C}(\text{cooperate})$ and $U_{A/B}=\sigma_x$ if $s_{A/B}=\mathrm{D}(\text{defect})$. Then one can calculate the final output quantum state $\left|\Psi_f\left(s_A, s_B\right)\right\rangle=U\left(U_A \otimes U_B\right) U^{\dagger}$, and the expectation values of the payoffs for Alice and Bob becomes

$$\begin{aligned}
& \bar{\pi}_A=3\left|\left\langle 00 \mid \Psi_f\right\rangle\right|^2+0\left|\left\langle 01 \mid \Psi_f\right\rangle\right|^2+5\left|\left\langle 10 \mid \Psi_f\right\rangle\right|^2+1\left|\left\langle 11 \mid \Psi_f\right\rangle\right|^2 \\
& \bar{\pi}_B=3\left|\left\langle 00 \mid \Psi_f\right\rangle\right|^2+5\left|\left\langle 01 \mid \Psi_f\right\rangle\right|^2+0\left|\left\langle 10 \mid \Psi_f\right\rangle\right|^2+1\left|\left\langle 11 \mid \Psi_f\right\rangle\right|^2
\end{aligned} \tag{17}$$

for any combinations of s_A and s_B. For example, when $(s_A = \sigma_x, s_B = \sigma_x)$ we obtain

$$\begin{aligned} & |\Psi_f\left(s_A=\sigma_x, s_B=\sigma_x\right)\rangle=|11\rangle \\ & \rightarrow \bar{\pi}_A=3|\langle 00 \mid \Psi_f\rangle|^2+0|\langle 01 \mid \Psi_f\rangle|^2+5|\langle 10 \mid \Psi_f\rangle|^2+1|\langle 11 \mid \Psi_f\rangle|^2=1 \\ & \rightarrow \bar{\pi}_B=3|\langle 00 \mid \Psi_f\rangle|^2+5|\langle 01 \mid \Psi_f\rangle|^2+0|\langle 10 \mid \Psi_f\rangle|^2+1|\langle 11 \mid \Psi_f\rangle|^2=1 \end{aligned} \tag{18}$$

in consistent with the values in the table in Fig. 15(a). The payoff values for other values of (s_A, s_B) are also calculated similarly Grabbe (2005). In the case of the quantum version of prisoner's dilemma, the single qubit gate $U_{A/B}$ is generalized to

$$U_{A/B}\left(\theta, \varphi\right)=U_3\left(\theta, \varphi, \pi\right)=\left(\begin{array}{cc} \cos\left(\theta/2\right) & \sin\left(\theta/2\right) \\ e^{i\varphi}\sin\left(\theta/2\right) & -e^{i\varphi}\cos\left(\theta/2\right) \end{array}\right). \tag{19}$$

We note that $U_{A/B}\left(0, \pi\right) = I$ (C) and $U_{A/B}\left(\pi, 0\right) = \sigma_x$ (D) correspond to the classical strategies. However, other angles are neither C nor D and are quantum strategies. Some special cases of the quantum strategies are $U_{A/B}\left(\pi/2, 0\right) = H$ (Hadamard gate) and $U_{A/B}\left(0, 0\right) = \sigma_z$ (Z gate). For example, when Alice's strategy is σ_z (quantum move) and Bob's strategy is σ_x (D), the output state becomes $|\Psi_f\left(s_A = \sigma_z, s_B = \sigma_x\right)\rangle = |10\rangle$, so that Alice's actual move is D and Bob's actual move is C, meaning that Bob's actual move was changed from his original strategy D. Similarly, when Alice's move is σ_x (D) and Bob's move is σ_z (quantum move), the output state becomes $|\Psi_f\left(s_A = \sigma_z, s_B = \sigma_x\right)\rangle = |01\rangle$, so that Alice's actual move is C and Bob's actual move is D, where Alice's actual move was changed from her original strategy D. Therefore, when one's strategy is quantum, the other's actual move can be changed from his/her original strategy. More details of the quantum game theory itself are referred to in the reference Grabbe (2005). With the above understanding of the quantum prisoner's dilemma problem, we next consider how we can apply this concept to the music creation.

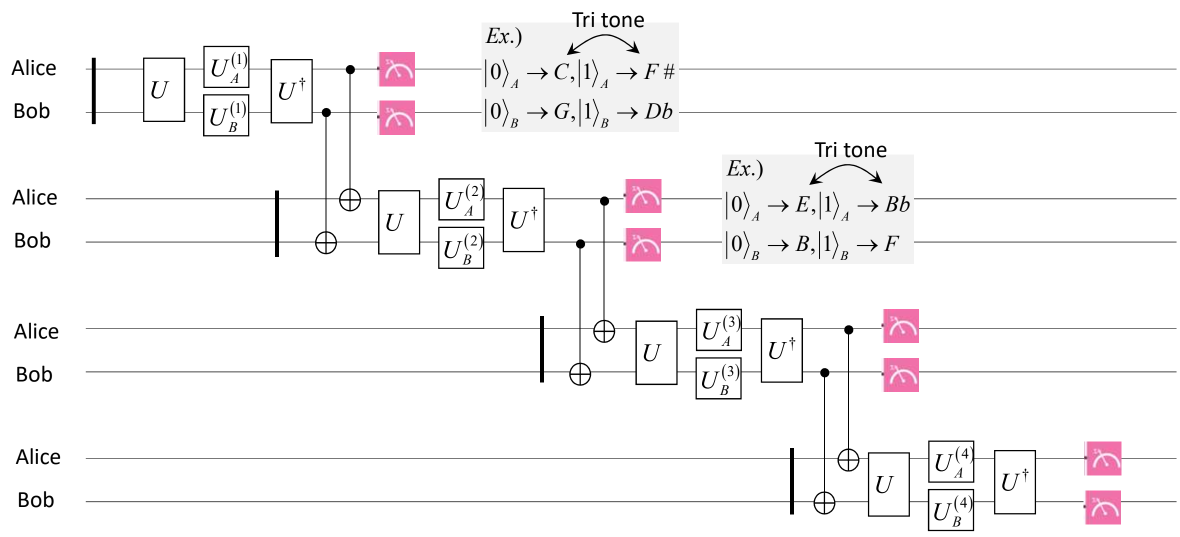

Figure 16. Two players model of music creation based on quantum game theory. Music progresses from the top to the bottom, where two players (Alice and Bob) are intended to make their sound acccording to their own strategies. Correspondence between $|0/1\rangle$ and note name for the each i-th note is pre-assigned as one of the composition process. The quantum gates U_A and U_B (strategies) for the i-th note are composed by the players A and B, respectively.

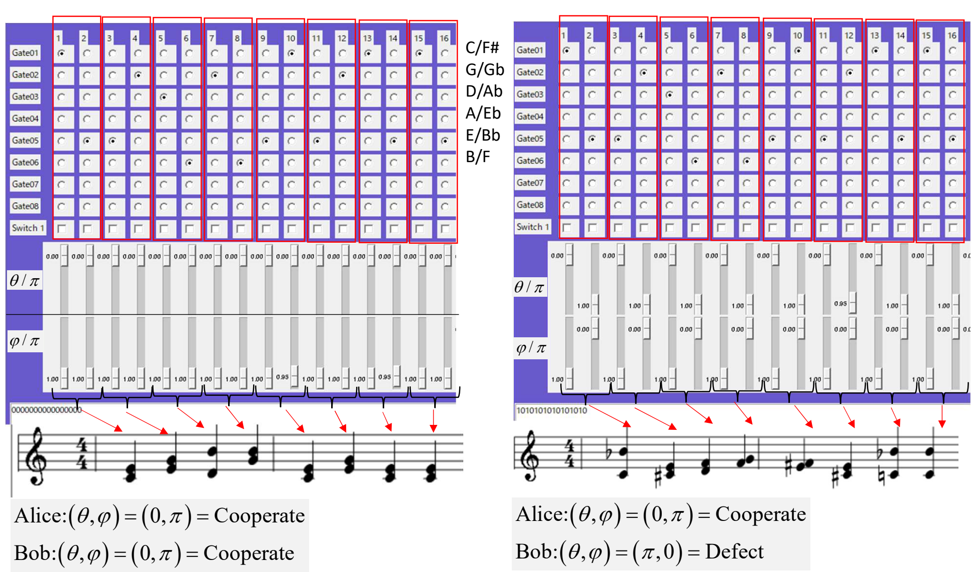

Figure 17. Examples of the composition applying the quantum game theory. In the left panel Alice's and Bob's strategy is both cooperate, while in the right panel Alice's strategy is cooperate and Bob's strategy is defect. These are both classical strategy.

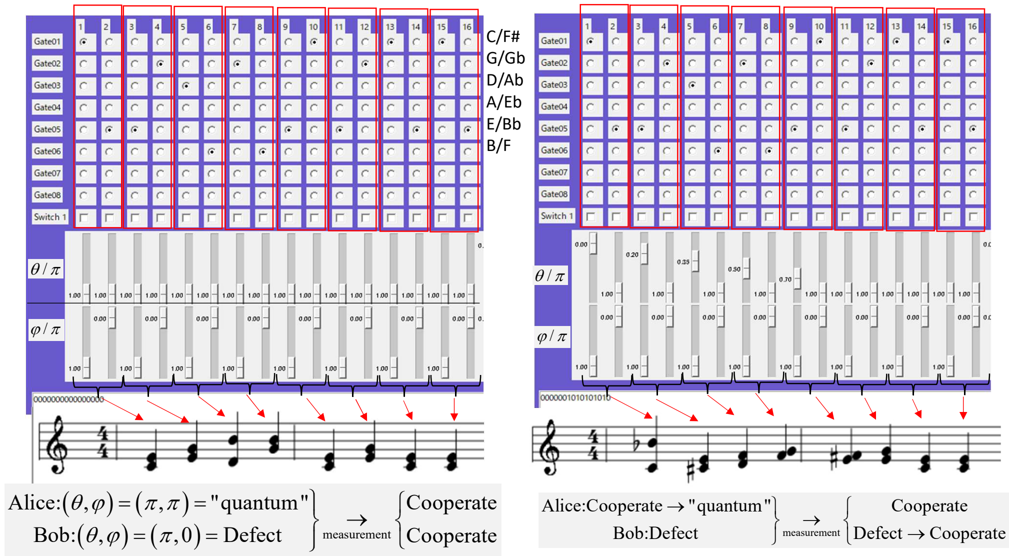

Figure 18. Examples of the composition applying the quantum game theory. In the left panel Alice's strategy is $(\theta, \phi) = (\pi, \pi)$ (one of quantum strategies) and Bob's strategy is defect. After the measurement the Bob's move is turned to "cooperate", while the Alice's move is also cooperate. In the right panel Alice's strategy is gradually changed from cooperate (classical) to $(\theta, \phi) = (\pi, \pi)$ (quantum), while the Bob's strategy is defect. After the measurement the Bob's result (move) is gradually changed from defect to cooperate, while Alice's move is coperate.

In our proposal of the quantum game-based music creation principle, the above-mentioned quantum game unit $U\,(U_A \otimes U_B)\,U^\dagger$ is used to generate one pair of musical notes, which is composed of two notes played by Alice and Bob. Then the measured results (either $|0\rangle$ or $|1\rangle$) of each Alice's and Bob's qubits are made correspond to musical notes following the pre-assigned qubit – note name correspondence. For example, in Fig. 15(a), $|0\rangle$ and $|1\rangle$ of Alice's qubit are made to correspond to C and F$^\sharp$, respectively, which are π difference in a circle of fifth. Similarly Bob's qubit is made correspond to G and D$^\flat$, respectively. The note measurement axis in the circle of fifth (e.g.., C-F$^\sharp$ axis and G-D$^\flat$ axis in the above example) is intended to be pre-assigned manually for each note timing as a part of the composition process, as illustrated in Fig. 16. It is also possible to place optionally CNOT gates connecting between one note slot to the next, as in Fig. 16, by which if the quantum state of one note is $|1\rangle$ the initial state of the next note is changed from $|0\rangle$ to $|1\rangle$. In Figs. 17 and 18, we show an examples of the composition by applying the quamtum game theory. In this example Alice's and Bob's measurements of $|0\rangle$ and $|1\rangle$ states produce the prepared melodies in C major and F$^\sharp$ major scale, respectively, as shown in the GUI interface shown in Figs. 17 and 18. In the left panel of Fig. 17 Alice's and Bob's strategy is both cooperate, making harmony in C major scale. On the other hand, in the right panel Alice's strategy is cooperate and Bob's strategy is defect, resulting into unharmony. We note that these are both classical strategy.

We next consider the quantum strategy. In the left panel of Fig. 18, Alice's strategy is $(\theta, \phi) = (\pi, \pi)$ (one of quantum strategies) and Bob's strategy is defect. After the measurement is performed, Bob's move is turned to "cooperate", while the Alice's move is cooperate, thereby making harmony in C major scale, meaning that one player's quantum strategy can change the other player's original strategy to make harmony. In the right panel, we assume that Alice's strategy is gradually changed from cooperate (classical) to $(\theta, \phi) = (\pi, \pi)$ (quantum), while the Bob's strategy is defect. After the measurement, the Bob's result (move) is gradually changed from defect to cooperate, while Alice's move is coperate. Then the transition from unharmony to harmony can be obtained as shown in Fig. 18.

Conclusion

We presented our recent proposals on developing a music creation system applying the quantum gate circuit, where we considered various principles and models to apply the quantum gate circuits for music creation. In considering how we can connect between the musical note and qubit and associate musical progression with quantum gate operations, we proposed two distinct approaches: the wavefunction-based approach and the measurement-based approach. In the former approach, we intended to translate the quantum wavefunction directly to the musical expression and proposed a new improvisational live performance, called the quantum live coding. In the latter approach, the measurement results of the quantum gate circuits are used in creating music variously, and we proposed various music creation schemes, including the quantum phase modulation model and the quantum game theoretical model.

Acknowledgements

This research was supported by 2018 IPA (Information-technology Promotion Agency, Japan) unexplored target business –gate-type quantum computer division–. The authors would like to thank Enago (www.enago.jp) for the English language review.

References

2018. "*International Interdisciplinary Conference, Quantum music (and beyond). Music and new technologies in the 21st century (Mar 21-22, 2018).*" http://quantummusic.org/2018/03/26/quantum-music-conference/.

2021. "*International symposium on Quantum Computing and Musical Creativity (Nov 19-20, 2021).*" https://iccmr-quantum.github.io/1st_isqcmc/.

2021. "*Qiskit: an open-source software development kit for quantum computing*." https://qiskit.org/.

Grabbe, J. O. 2005. "An Introduction to Quantum Game Theory." *arXiv preprint* :arXiv:quant–ph/0506219.

Johansson, J. R., P. D. Nation, and F. Nori. 2012. "QuTiP: An open-source Python framework for the dynamics of open quantum systems." *Comp. Phys. Comm.* 183:1760–1772.

Johansson, J. R., P. D. Nation, and F. Nori. 2013. "QuTiP 2: A Python framework for the dynamics of open quantum systems." *Comp. Phys. Comm.* 184:1234.

Kirkbride, R. 2016. "FoxDot: Live coding with python and supercollider." In *Proceedings of the International Conference of Live Interfaces, 2016*, pp. 194–198.

Kirke, A. 2019. "Applying Quantum Hardware to non-Scientific Problems: Grover's Algorithm and Rule-based Algorithmic Music Composition." *arXiv:1902.04237* :1902.04237.

Kirke, A., and E. R. Miranda. 2017. *"Experiments in Sound and Music Quantum Computing" in "Guide to Unconventional Computing for Music"*. Springer International Publishing.

McCartney, J. 2016. "Rethinking the computer music language: Supercollider." *Computer Music Journal* 26:61–68.

Miranda, E. R. 2020. "Quantum Computer: Hello, Music!" *arXiv preprint* :arXiv:2006.13849.

Miranda, E. R., and S. T. Bask. 2021. "Quantum Computer Music: Foundations and Initial Experiments." *arXiv preprint* :arXiv:2110.12408.

von Neumann, J., and M. Oscar. 1944. *The Theory of Games and Economic Behavior*. New York: Wiley.

Nishimura, S. 1998. "PMML: A music description language supporting algorithmic representation of musical expression." In *Proc. of the 1998 International Computer Music Conference, pp. 171-174, 1998*, pp. 171–174.

Nishimura, S. 2014a. "*Takt: a text-based music programming language*." http://takt.sourceforge.net/.

Nishimura, S. 2014b. "Takt: A read-eval-play-loop interpreter for a structural/procedural score language." In *Proc. of the 2014 International Computer Music Conference, 2014*, pp. 1736–1741.

Oshiro, S., and O. C. Hamido. 2020. "A Quantum-Classical Network for Beat-Making Performance." *Journal of Network Music and Arts* 2:1.

Putz, V., and K. Svozil. 2017. "Quantum music." *Soft Computing* 21:1467.

Putz, V., and K. Svozil. 2021. "Quantum music, quantum arts and their perception." *arXiv preprint* :arXiv:2108.05207.

Souma, S. 2022. https://github.com/ssouma3/qumusic.

Weaver, J. L. 2018. "*Jamming with a Quantum Computer*." https://medium.com/rigetti/jamming-with-a-quantum-computer-bed05550a0e8.