Flexible Representation and Manipulation of Audio Signals on Quantum Computers

Fei Yan, Yiming Guo, Abdullah M. Iliyasu, and Huamin Yang

Abstract—By analyzing the numerical representation of amplitude values in audio signals and integrating the time component, a representation for audio signals on quantum computers, FRQA, is proposed. The FRQA representation is a normalized state that facilitates basic audio signal operations targeting these parameters. The preparation and retrieval for FRQA are discussed and, based on the FRQA state, we realize the circuits to accomplish basic audio signal operations such as signal addition, signal inversion, signal delay, and signal reversal. These operations can be employed as the major components to build advanced operations for particular applications in the quantum computing domain.

Index Terms—Quantum computing, quantum audio, audio representation, audio manipulation.

✦

1 INTRODUCTION

QUANTUM computation and quantum information, an area of study focused on accomplishing information processing tasks on quantum mechanical systems, has become a rapidly growing field since Feynman proposed the quantum computation model in 1982 [1]. Afterwards, further developments by a number of researchers, notably, Deutsch's quantum parallelism assertion [2], Shor's integer factoring algorithm [3], and Grover's database searching algorithm [4], marked a leap forward in terms of the anticipated computing capability and secure communication for quantum computing devices. Not surprisingly, the appeal of this novel computational paradigm has continued to increase especially from scientists and engineers who work in areas related to this.

An image is a common artifact that depicts visual perception, which makes its realization and handling on the quantum computing framework an alluring pursuit. Attempts to undertake image processing based on quantum mechanics predate 2003, while those efforts had an inclination towards quantum optics and physics [5], [6]. Recent developments in research on quantum image processing (these with more computer science flair) developed with the proposals of quantum image representations. Since 2003, a number of quantum image representations have been proposed for encoding and transforming images on the quantum computing framework [7].

Research findings on quantum computation and quantum information have provided necessary theoretical basis for improving the possibility and reliability of quantum image processing. Nevertheless, there has not been sufficient effort to exploit another kind of information-carrying medium and its related manipulations, i.e. the quantum audio processing, to further extend the potential capabilities of quantum information processing. This is largely attributed to the lack of representations to encode quantum audio signals.

Classical audio processing may arise in either digital or analog domain, however, most modern audio systems prefer to digital representations because the techniques of digital signal processing are much more powerful and efficient [8]. In order to process and transmit digital audio on the quantum computing framework, it is necessary to explore a flexible and efficient quantum representation for storing and representing digital signals. In digital signal processing domains, theoretical analyses and derivations are typically performed on discrete-time signal models created by the abstract process of sampling. Inspired by similar methods, in [9], a quantum representation of digital audio (QRDA), which uses two entangled qubit sequences to store the audio amplitude and time information, was proposed. While QRDA extends audio processing into the quantum computing domain, its numerical representation of amplitude values only offers an encoding method for unipolar (viz. non-negative) numbers which lead to operations that are susceptible to errors.

In order to develop quantum algorithms in the field of audio signal processing, a set of basic signal operations must be developed. Considering amplitude transformations required to process audio signals on quantum mechanical computers, we analyze the numerical representation of amplitude values and describe a flexible representation of quantum audio (FRQA). The FRQA encodes the amplitude information in two's complement notation and integrates the time component to a normalized quantum state, which is more flexible and better suited to be the fundamental protocol for quantum audio processing. Based on the FRQA state, audio signal operations including addition, inversion, delay, and reversal are proposed in this study. These operations are implemented by using the quantum circuit model, where algorithms are compiled into a sequence of common gates acting on one or more qubits, such as NOT, Hadamard, CNOT, and Toffoli gates [10].

- F. Yan, Y. Guo, and H. Yang are with School of Computer Science and Technology, Changchun University of Science and Technology, No. 7089, Weixing Road, Changchun 130022, China. E-mail: yanfei@cust.edu.cn

- A. M. Iliyasu is with Electrical Engineering Department, College of Engineering, Prince Sattam Bin Abdulaziz University, Al-Kharj 11942, Kingdom of Saudi Arabia.

The rest of the paper is organized as follows. We start with the proposal, preparation, and retrieval of the FRQA state in Section 2. In Section 3, various definitions for signal operations including addition, inversion, delay, and reversal, along with their quantum circuit constructions are introduced. Finally, concluding remarks and possible future works are discussed in Section 4.

2 A FLEXIBLE REPRESENTATION OF QUANTUM AUDIO AND ITS PREPARATIONS

2.1 FRQA

In electrical engineering, analog audio signals are often considered as a voltage that is varying over time. An Analog-to-Digital Converter (ADC) discretely takes samples from the analog signals at a given frequency (i.e. sampling rate) and according to the given binary sequence length (a.k.a. resolution), each sampled value is converted into a number based on its voltage level. In this fashion, digital audio is produced and represented as a sequence of numbers that express instantaneous amplitudes of the audio signal being sampled.

As a quantum representation, QRDA utilizes two entangled qubit sequences to store the audio amplitude and time information. Therein, the amplitude information is considered as a string of non-negative integers in the form:

$$a_t \in \{0, 1, \ldots, 2^q - 1\}, \tag{1}$$

where q is the length of binary sequence used to store each number, a_t ($t = 0, 1, \ldots, L\text{-}1$) denotes the tth amplitude value in an L-sized QRDA audio.

This extension facilitates the audio representation and manipulation in quantum computing domain. Nonetheless, for reasons we adduce below, the unipolar encoding strategy used in QRDA may not be applicable for the accurate computation and processing operations in quantum audio processing:

(1) In QRDA, a_t in Eq. (1) can only represent non-negative numbers. Hence, some arithmetic operations pertaining to amplitude values are prone to errors. For instance, there are two amplitude values a_m and a_n ($a_m < a_n$), it is virtually impossible to implement the operation $a_m - a_n$ and depict the result.

(2) QRDA is an unipolar representation, thus it cannot display or determine the midrange of the waveform in the processing operations. Accordingly, the single values from two waveforms are added together is by no means to give a correct mixing function, e.g., addition of the opposite amplitude values in two waveforms will accumulate to higher amplitude instead of counteracting or offsetting each other.

In this study, a flexible representation of quantum audio, i.e. FRQA, is proposed. FRQA encodes the amplitude values in quantum audio in a bipolar (both non-negative and negative) manner, i.e. $s_t \in \{-2^{q-1}, \ldots, -1, 0, 1, \ldots, 2^{q-1} - 1\}$. Compared with the only known effort to perform audio signal processing on quantum computers (i.e. the QRDA), our proposal is more intuitive and graphic with the sign indicating the variation of the waveform, making it conceptually more like the real waveform of digital audio signals. More importantly, FRQA facilitates the realization of basic processing operations in an effective manner, e.g., (1) FRQA allows two sample values to be accurately added (or mixed in audio parlance) and it enables easy handling of the overflow and warp-around situations by using quantum circuits; (2) FRQA permits signal subtraction to be executed by simple logical addition operation with the inverted inputs, which offers a considerable saving in hardware complexity (since only carry logic is necessary and no borrow mechanism need be supported).

As conceived, the formalism of logic arithmetics provide FRQA with the adequate tools needed for effective quantum audio processing. Moreover, in digital audio processing, the most widespread method to represent bipolar numbers is based on using two's complement notation. Hence, the extension to encode the amplitude value in quantum audio using the two's complement arithmetic is both analogous and astute. Equation (2) describes the stipulation in the form:

$$S_t = S_t^0 S_t^1 \ldots S_t^{q-1}, S_t^i \in \{0, 1\}, i = 0, 1, \ldots, q - 1, \tag{2}$$

where $t = 0, 1, \ldots, 2^l - 1$ denotes the time information of a 2^l-sized quantum audio signal, $S_t = S_t^0 S_t^1 \ldots S_t^{q-1}$ is the binary sequence encoding the two's complement notation of the amplitude value. Two cases of the binary sequence S_t are listed as follows:

(1) If the amplitude value is non-negative, then $S_t^0 = 0$ and S_t is simply represented as a binary sequence of the value itself.

(2) If the amplitude value is negative, then $S_t^0 = 1$ and S_t is represented by the two's complement mode of its absolute value.

Therefore, the quantum representative expression of an 2^l-sized FRQA audio can be written as below:

$$|A\rangle = \frac{1}{2^{l/2}} \sum_{t=0}^{2^l-1} |S_t\rangle \otimes |t\rangle, \tag{3}$$

where $|S_t\rangle = |S_t^0 S_t^1 \ldots S_t^{q-1}\rangle$ is the two's complement representation of each amplitude value and $|t\rangle = |t_0 t_1 \ldots t_{l-1}\rangle$, $t_i \in \{0, 1\}$, is the corresponding time information. The state $|A\rangle$ is normalized, i.e. $\||A\rangle\| = 1$. It is trivial that, as formalised in Eq. (3), the FRQA needs $(q + l)$ qubits to represent a quantum audio with 2^l samples.

For an L-sized (L cannot be represented by 2^l) FRQA audio, employing l qubits to represent the time information will produce $2^l - L$ audio redundancies [9]. In this study, the amplitude values of all the redundancies are set as $|0\rangle$, so the FRQA state in Eq. (3) can be rewritten into the form:

$$|A'\rangle = \frac{1}{2^{l/2}} \left(\sum_{t=0}^{L-1} |S_t\rangle \otimes |t\rangle + \sum_{t=L}^{2^l-1} |0\rangle^{\otimes q} \otimes |t\rangle \right), \tag{4}$$

where

$$l = \begin{cases} \lceil \log_2 L \rceil, & L > 1 \\ 1, & L = 1 \end{cases}. \tag{5}$$

Figure 1 shows a segment of an audio signal and its representative expression using the FRQA representation. In this example, amplitude values are sampled between -2
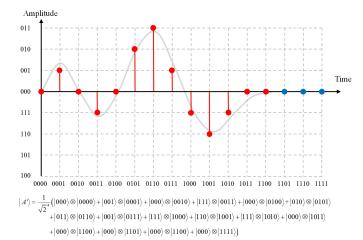
$$|A'\rangle = \frac{1}{\sqrt{2}^4}\big(|000\rangle \otimes |0000\rangle + |001\rangle \otimes |0001\rangle + |000\rangle \otimes |0010\rangle + |111\rangle \otimes |0011\rangle + |000\rangle \otimes |0100\rangle + |010\rangle \otimes |0101\rangle$$
$$+ |011\rangle \otimes |0110\rangle + |001\rangle \otimes |0111\rangle + |111\rangle \otimes |1000\rangle + |110\rangle \otimes |1001\rangle + |111\rangle \otimes |1010\rangle + |000\rangle \otimes |1011\rangle$$
$$+ |000\rangle \otimes |1100\rangle + |000\rangle \otimes |1101\rangle + |000\rangle \otimes |1100\rangle + |000\rangle \otimes |1111\rangle\big)$$

Fig. 1. A segment of an FRQA audio signal and its representation (blue points are the redundancies).

and 3, for which 3 qubits are needed to store the amplitude information in FRQA audio. The length of the audio is 13, then $l = \lceil \log_2 13 \rceil = 4$. Therefore, in total 7 qubits are required to represent this quantum audio signal.

2.2 FRQA preparation

In quantum computation, in order to prepare a desired quantum audio signal, it is necessary to transform a quantum computer from its initialized state to the specified state. The quantum mechanical build up of quantum computers dictate that all transformations are unitary in nature. This specifies the use of unitary matrices and simple basic gates to manipulate quantum data. In this subsection, we will present a description of the procedure required to prepare an FRQA state.

As stated earlier, in digital audio, the value of each sampled point is stored on an assigned q-length binary sequence, i.e. $B_t^0 B_t^1 \ldots B_t^{q-1}$, $B_t^i \in \{0,1\}$, which indicates the sequence is able to hold the amplitude values from 0 to $2^q - 1$. In order to quantize the amplitude value (B_t) and convert it into the two's complement system ($|S_t\rangle$), we need to perform the value-setting operation Ω_t which could be achieved by the following two sub-operations:

(1) *Quantization*: Using the classical binary representation of B_t as the reference, the initialized states in the quantum computer could be transformed into their desired quantum states. The layout of how to accomplish this has been widely discussed in the literature, notably [11], [12].

(2) *Conversion*: In order to convert the ordinary quantum state to its two's complement system, the CNOT gate operation is executed on the Most-Significant-Bit (MSB) of the qubit sequence. In this manner, the output will be the desired quantum amplitude state, i.e. $|S_t\rangle$.

To illustrate these steps, consider the conversion of a 3-qubit resolution to the amplitude information in quantum audio as presented in TABLE 1. A 3-qubit resolution holds the values ranging from 0 to 7. These numbers are binarized

TABLE 1
Conversion of a 3-qubit resolution to the amplitude information in quantum audio.

Resolution	Binary sequence	Two's complement	Amplitude
7	111	$\lvert 011 \rangle$	3
6	110	$\lvert 010 \rangle$	2
5	101	$\lvert 001 \rangle$	1
4	100	$\lvert 000 \rangle$	0
3	011	$\lvert 111 \rangle$	-1
2	010	$\lvert 110 \rangle$	-2
1	001	$\lvert 101 \rangle$	-3
0	000	$\lvert 100 \rangle$	-4

and quantized in two's complement notation for further computation in quantum audio processing.

Therefore, the value-setting operation Ω_t to perform the above procedure can be executed using Eqs. (6) and (7):

$$\Omega_t = \overset{q-1}{\underset{i=0}{\otimes}} \Omega_t^i, \tag{6}$$

$$\Omega_t^i (|0\rangle) = \begin{cases} \overline{|0 \oplus B_t^i\rangle}, & i = 0 \\ |0 \oplus B_t^i\rangle, & i \neq 0 \end{cases}, \tag{7}$$

where $\oplus$ is the XOR operation. It is apparent to clarify that the operation Ω_t^i works by means of an additional CNOT gate to negate the MSB only when $B_t^i = 1$, otherwise, it remains unchanged. Following this, the quantum transformation of Ω_t to set amplitude value for each sample is shown in Eq. (8):

$$\Omega_t |0\rangle^{\otimes q} = \overset{q-1}{\underset{i=0}{\otimes}} \left(\Omega_t^i |0\rangle \right)$$
$$= \overline{|0 \oplus B_t^0\rangle} \otimes \left(\overset{q-1}{\underset{i=1}{\otimes}} |0 \oplus B_t^i\rangle \right) \tag{8}$$
$$= \overset{q-1}{\underset{i=0}{\otimes}} |S_t^i\rangle = |S_t\rangle.$$

In this manner, a unitary transform that encodes the amplitude information by means of two's complement arithmetic is available during the preparation procedure. There are two steps to achieve the preparation procedure that are discussed as below:

Step 1: The initial state $|0\rangle^{\otimes q+l}$ is transformed into the intermediate state $|H\rangle$ using the 2-D identity matrix I and 2-D Hadamard matrix H presented in Eq. (9):

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \tag{9}$$

Since the tensor product of q Identity matrices and l Hadamard matrices are denoted by $I^{\otimes q}$ and $H^{\otimes l}$, respectively, applying the transform $\mathcal{H} = I^{\otimes q} \otimes H^{\otimes l}$ on $|0\rangle^{\otimes q+l}$ produces the intermediate state $|H\rangle$ in the form:

$$|H\rangle = \mathcal{H}\left(|0\rangle^{\otimes q+l}\right) = \frac{1}{\sqrt{2}^l} \sum_{t=0}^{2^l-1} |0\rangle^{\otimes q}|t\rangle. \tag{10}$$

So far, the time information of the FRQA model has been initialized. At this moment, the intermediate state $|H\rangle$ can be regarded as the superposition of all the samples of an empty digital audio, i.e. with all amplitude values set as $|0\rangle$.

Step 2: The value-setting operation Ω_t is used to generate the amplitude information for each sample. Since Ω_t can only handle one sample at a time, considering a 2^l-sized quantum audio, 2^l sub-operations are needed to execute this transformation. While R_t is considered as an l-controlled Ω_t operation to integrate the amplitude values into each instant of time. For a given sample k, R_t is defined in Eq. (11):

$$R_k = \left(I \otimes \sum_{t=0, t \neq k}^{2^l - 1} |t\rangle\langle t| \right) + \Omega_k \otimes |k\rangle\langle k|. \quad (11)$$

Applying R_k on the the intermediate state $|H\rangle$ gives us:

$$
\begin{aligned}
R_k \left(|H\rangle \right) &= R_k \left(\frac{1}{\sqrt{2^l}} \sum_{t=0}^{2^l - 1} |0\rangle^{\otimes q} |t\rangle \right) \\
&= \frac{1}{\sqrt{2^l}} R_k \left(\sum_{t=0, t \neq k}^{2^l - 1} |0\rangle^{\otimes q} |t\rangle + |0\rangle^{\otimes q} |k\rangle \right) \\
&= \frac{1}{\sqrt{2^l}} \left(\sum_{t=0, t \neq k}^{2^l - 1} |0\rangle^{\otimes q} |t\rangle + \Omega_k |0\rangle^{\otimes q} |k\rangle \right) \\
&= \frac{1}{\sqrt{2^l}} \left(\sum_{t=0, t \neq k}^{2^l - 1} |0\rangle^{\otimes q} |t\rangle + |S_k\rangle |k\rangle \right).
\end{aligned}
\quad (12)
$$

From Eq. (12), it is clear that for all the sub-operations R_t, we have:

$$\mathcal{R}|H\rangle = \left(\prod_{t=0}^{2^l} R_t \right) |H\rangle = \frac{1}{2^{l/2}} \sum_{t=0}^{2^l - 1} |S_t\rangle \otimes |t\rangle = |A\rangle. \quad (13)$$

After the two steps described above, the initialized state $|0\rangle^{\otimes q+l}$ is transformed into the desired FRQA state by applying the unitary transform $\mathcal{R}\mathcal{H}$. Subsequently, we will continue to discuss the complexity of the preparation procedure.

Complexity theory on quantum computation has been studied to discover and analyze the transformations from the basic gates, viz. the complexity of quantum algorithms is usually computed in terms of quantum gates [10]. Indeed, the circuit complexity depends largely on the strategy employed for circuit decomposition and the basic operation unit that is designated to be [13]. In this study, we will confine our discussion to the CNOT gate for the complexity evaluation as it is a relatively "expensive" elementary gate that can be easily utilized to simulate the other complicated gates. The decomposition networks for some of the complicated quantum gates used in this paper is presented in Fig. 2, where the l-controlled NOT gate can be decomposed into $2(l-1)$ Toffoli gates and 1 CNOT gate [as indicated in (c)] and the Toffoli gate can be approximately simulated by 6 CNOT gates [14].

Based on the foregoing, it is clear that the implementation of transform $\mathcal{H}$ in Step 1 requires l Hadamard gates to execute. In addition, transform $\mathcal{R}$ in Step 2 can be divided into 2^l sub-operations (i.e. R_t) to store the amplitude information for each sample. Therefore, with the assistance of enough ancillary qubits, each sub-operation R_t can be directly implemented using $2(l-1)$ Toffoli gates and no more

(a) (b)

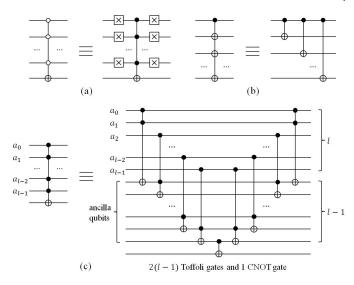(c) $\quad 2(l-1)$ Toffoli gates and 1 CNOT gate

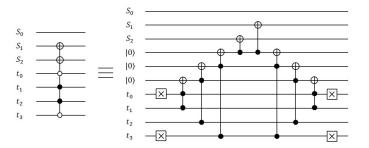Fig. 2. Decomposition of some complicated quantum gates.

Fig. 3. Network of implementing the R_6 operation when preparing the sample at $|0110\rangle$ of FRQA state in Fig. 1.

than q CNOT gates. Hence, the complexity of preparing a 2^l-sized FRQA state is:

$$2^l \times [2(l-1) \times 6 + q] = (12l + q - 12) \cdot 2^l, \quad (14)$$

which indicates the efficiency of the preparation process. In order to further illustrate the complexity of constructing an FRQA quantum audio, we select the initialization of the sample at $|0110\rangle$ in Fig. 1 as an example. The network for implementing the operation, i.e. R_6 [when $k = 6$ in Eq. (11)], is depicted in Fig. 3.

2.3 FRQA retrieval

The essential requirements for representing a quantum audio are the simplicity and efficiency in the storage and retrieval of the audio information. The storage of an FRQA audio is achieved by the preparation process as discussed in the preceding subsection. In this subsection, the rudiments required for the FRQA signal retrieval will be formalised.

Quantum measurement is the unique way of recovering classical information from a quantum state. To retrieve the amplitude information of each sample, quantum measurements Γ and M as defined in Eqs. (15) and (16) are used:

$$\Gamma = \sum_{t=0}^{2^l - 1} I^{\otimes q} \otimes |t\rangle\langle t|, \quad (15)$$

$$M = \sum_{m=0}^{2^q-1} m|m\rangle\langle m|. \qquad (16)$$

Firstly, executing the quantum measurement operation Γ on the time sequence extracts corresponding information of sample t as $|P_t\rangle$, which is shown in Eq. (17):

$$|P_t\rangle = |S_t\rangle|t\rangle, \qquad (17)$$

and then, the projective measurement M is used to recover the amplitude value from the quantum state:

$$\langle S_t|M|S_t\rangle = \langle S_t|\left(\sum_{m=0}^{2^q-1} m|m\rangle\langle m|\right)|S_t\rangle$$
$$= \sum_{m=0}^{2^q-1} m\langle S_t||m\rangle\langle m||S_t\rangle = S_t. \qquad (18)$$

Through these measurement operations, the amplitude value of sample k will be recovered. This means that all the samples in a quantum audio signal can be recovered in the same way, so the original digital audio can be retrieved from the FRQA state (Note that it is the theoretical discussion about the quantum information retrieval. We will not touch at all on the consequence of decoherence or any errors appeared in the process.)

3 OPERATIONS ON QUANTUM AUDIO SIGNALS

Basic signal operations are the foundations of digital audio processing. The ability to extend similar operations to quantum audio processing is essential to validating the utility of this emerging sub-field of quantum information processing. In this section, we will present a few basic audio signal operations including signal addition, signal inversion, signal delay, and signal reversal based on the FRQA representation.

3.1 Signal addition

Signal addition is among the fundamental operations in audio signal processing. This operation involves the addition of amplitudes of two or more signals at each instant of time. By means of this operation, a series of audio signal processing like echo or reverb addition and active noise reduction can be implemented. Representing the amplitude values in two's complement system in FRQA facilitates determining the results of the arithmetic operations and depicting the waveform with respect to the midrange. Hence, we can define the signal addition operation based on this arithmetic advantage as below.

Assuming that $|A_x\rangle$ and $|A_y\rangle$ are two 2^l audio segments as presented in Eqs. (19)-(20), the signal addition operation produces the output of $|A_z\rangle$ as shown in Eq. (21):

$$|A_x\rangle = \frac{1}{2^{l/2}} \sum_{x=0}^{2^l-1} |S_x\rangle \otimes |t_x\rangle,$$

where

$$|t_x\rangle = |t_x^0 t_x^1 \dots t_x^{l-1}\rangle, t_x^i \in \{0,1\},$$
$$|S_x\rangle = |S_x^0 S_x^1 \dots S_x^{q-1}\rangle, S_x^i \in \{0,1\}, \qquad (19)$$
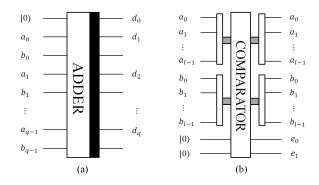

Fig. 4. (a) The quantum ADDER module and (b) the quantum comparator module.

and

$$|A_y\rangle = \frac{1}{2^{l/2}} \sum_{y=0}^{2^l-1} |S_y\rangle \otimes |t_y\rangle,$$

where

$$|t_y\rangle = |t_y^0 t_y^1 \dots t_y^{l-1}\rangle, t_y^i \in \{0,1\},$$
$$|S_y\rangle = |S_y^0 S_y^1 \dots S_y^{q-1}\rangle, S_y^i \in \{0,1\}, \qquad (20)$$

so

$$|A_z\rangle = \frac{1}{2^{l/2}} \sum_{z=0}^{2^l-1} |S_z\rangle \otimes |t_z\rangle,$$

where

$$|t_z\rangle = |t_z^0 t_z^1 \dots t_z^{l-1}\rangle, t_z^i \in \{0,1\},$$
$$|S_z\rangle = |S_z^0 S_z^1 \dots S_z^q\rangle = |S_x + S_y\rangle, S_z^i \in \{0,1\},$$
$$t_z = t_x = t_y. \qquad (21)$$

Quantum algorithms are tools for constructing advanced operations using circuit elements. In order to construct the quantum circuit to execute the signal addition operation, we start by introducing two commonly-used quantum modules that are presented in Fig. 4.

The ADDER module [13] is designed to realize the operation $|a\rangle|b\rangle \to |a\rangle|a+b\rangle$, where $|a\rangle$ and $|b\rangle$ are two quantum sequences both with q qubits, and $|a + b\rangle = |d_0 d_1 \dots d_q\rangle$ is a quantum sequence with $(q + 1)$ qubits that represents the result of addition. By using the ADDER module and necessary control conditions, we can realize the addition of amplitude values in two segments of quantum audio at any given instant of time.

The other module presented in Fig. 4 is the quantum comparator module [15] that is used to compare two l-qubit binary sequences ($|a\rangle$ and $|b\rangle$) and stores the result in the ancilla qubits $|e_0 e_1\rangle$. A useful output of $|e_0 e_1\rangle$ in this study is when $|e_0 e_1\rangle = |00\rangle$, then $|a\rangle = |b\rangle$. The comparator module guarantees that the two operands in the addition are the amplitude values of the two audio segments at the same instant of time.

Hence, the operation of FRQA-based signal addition operation $\mathcal{U}_A$ can be written in the form:

$$\mathcal{U}_A : |S_x\rangle|S_y\rangle \to |S_x\rangle|S_x + S_y\rangle. \qquad (22)$$

The quantum circuit to realize the operation $\mathcal{U}_A$ is depicted in Fig. 5, where the ADDER operation is applied
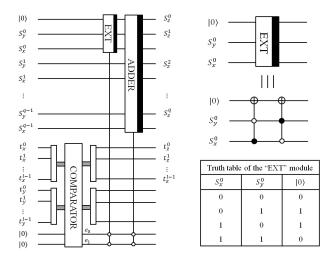
Fig. 5. Circuit construction to execute the quantum audio signal addition operation.

Fig. 6. An example of quantum audio signal addition operation.

on the amplitude content and the comparator operation is applied on the time content. Additionally, in order to avoid the overflow during two's complement addition, we design a sign extension module (denoted as "EXT" in Fig. 5) which helps to increase the length of the binary sequence while preserving the sign information in the ADDER module. To conclude our discussion of the signal addition operation, an example for the case when $l = 2$ and $q = 2$ is presented in Fig. 6.

As seen in Fig. 5, the circuit network for implementing the signal addition operation is divided into 3 parts: a comparator module, a 2-controlled "EXT" module, and a 2-controlled ADDER module. Implementations of the ADDER and comparator module have been extensively discussed in [15] and [16]. Additionally, the computational complexities of the two modules are $28q - 12$ and $24l^2 + 6l$, respectively, where q and l denote the length of the inputs as shown in Fig. 4. Consequently, to effectively quantify the complexity of the operation $\mathcal{U}_A$ depicted in Fig. 5, we only need to determine the complexity of the 2-controlled "EXT" module and 2-controlled ADDER module.

The 2-controlled "EXT" module is a pair of 4-controlled NOT gates, so complexity of this part is $2 \times [2 \times (4-1) \times 6 + 1] = 74$. The ADDER module consists of $4q - 2$ Toffoli gates and $4q$ CNOT gates [16], so the 2-controlled unit consists of $4q - 2$ 4-controlled NOT gates and $4q$ 3-controlled NOT gates, thereby the complexity of this controlled module is $248q - 74$. Based on these analyses, the complexity of the entire signal addition operation is $24l^2 + 6l + 248q$.

3.2 Signal inversion

Inverting a signal is common in signal processing algorithms. A meaningful usage of inversion is to transform the signal subtraction to signal addition by means of inverted inputs (audio cancellation is realized as so). In audio signal processing, signal inversion is realized by inverting all amplitude values of an audio signal. When amplitude values are represented by bipolar values, the operation is essentially an alteration of positive and negative signs. In this subsection, the FRQA-based signal inversion operation is formalised.
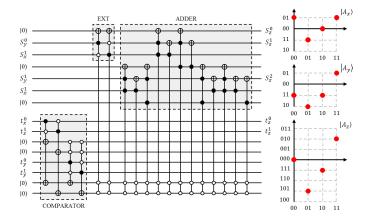
Assuming that $|A\rangle$ is an FRQA audio in the form presented in Eq. (3), the signal inversion operation $\mathcal{U}_I$ applied on $|A\rangle$ will produce the output of $|A_I\rangle$ in the form shown in Eq. (23):

$$|A_I\rangle = \frac{1}{2^{l/2}} \sum_{t=0}^{2^l-1} |S_I\rangle \otimes |t\rangle,$$

where

$$|t\rangle = |t_0 t_1 \ldots t_{l-1}\rangle, t_i \in \{0,1\},$$
$$|S_I\rangle = |-S_t^0 S_t^1 \ldots S_t^{q-1}\rangle = |-S_t\rangle, S_t^i \in \{0,1\}. \quad (23)$$

As in two's complement arithmetic, an effective way to negate a number is inverting all the qubits and adding "1". This procedure in quantum audio processing is further explained using the following steps:

Step 1: Invert all the qubits in $|S_t\rangle$ by performing operation $\mathcal{U}_I^1$:

$$\mathcal{U}_I^1(|A\rangle) = \frac{1}{2^{l/2}} \mathcal{U}_I^1 (\sum_{t=0}^{2^l-1} |S_t\rangle) \otimes |t\rangle$$
$$= \frac{1}{2^{l/2}} \sum_{t=0}^{2^l-1} |\overline{S_t}\rangle \otimes |t\rangle,$$

where

$$|S_t\rangle = |S_t^0 S_t^1 \ldots S_t^{q-1}\rangle, S_t^i \in \{0,1\},$$
$$|\overline{S_t}\rangle = |\overline{S_t^0 S_t^1 \ldots S_t^{q-1}}\rangle, \overline{S_t^i} \in \{0,1\}. \quad (24)$$

Step 2: Add "1" to the inverted result $|\overline{S_t}\rangle$ and neglect the overflow by applying operation $\mathcal{U}_I^2$:

$$\mathcal{U}_I^2 : |\overline{S_t}\rangle \to = |S_I\rangle,$$

where

$$|\overline{S_t}\rangle = |\overline{S_t^0 S_t^1 \ldots S_t^{q-1}}\rangle,$$
$$|(\overline{S_t^0 S_t^1 \ldots S_t^{q-1}} + 1) \bmod 2^q\rangle = |S_I\rangle. \quad (25)$$

It should be noted that similar to what is obtained in (classical) digital audio signal processing, a slight skewing of the data range is unavoidable as we cannot find the inverted counterpart of $|-2^{q-1}\rangle$ in the sampled amplitude values s_t. To overcome this, we restrict our discussion within the amplitude value range from $|-2^{q-1}+1\rangle$ to $|2^{q-1}-1\rangle$. The quantum circuit for inverting an FRQA audio
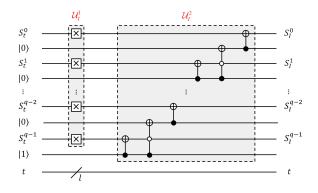
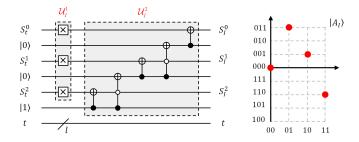Fig. 7. Circuit construction to execute the quantum audio signal inversion operation.

Fig. 8. An example of the quantum audio signal inversion operation ($|A_I\rangle$ is the inverted version of $|A_z\rangle$ in Fig. 6).
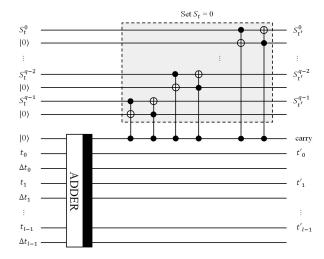

Fig. 9. General circuit for quantum audio signal delay operation.

Fig. 10. (a) The input quantum audio signal and (b) the delayed signal.

signal is shown in Fig. 7, wherein, the operation $\mathcal{U}_I^1$ in Step 1 can be directly implemented by using q NOT gates and in the operation $\mathcal{U}_I^2$ in Step 2, $q - 1$ Toffoli gates and q CNOT gates are required, thus the complexity of this operation is $7q - 6$.

A simple example to illustrate how a quantum audio signal is inverted would suffice. The circuit network required to execute this procedure is presented in Fig. 8, where the FRQA audio $|A_z\rangle$ in Fig. 6 is used as the input audio signal.

3.3 Signal delay

Signal delay is the operation that records an input signal and then plays it back after a period of time. This operation is a common audio effect that is often used to create the sound of a repeating and decaying echo. In this subsection, the FRQA-based signal delay operation is formalised.

Assuming $A(t)$ is the original audio signal, then audio signal $B(t)$ is regarded as a delayed version of audio $A(t)$ if:

$$A(t) = B(t'), t' = t + \triangle t, \tag{26}$$

where t denotes the time information of amplitude values, $\triangle t$ is a fixed interval which specifies the desired delay of the system. Using the foregoing, the signal delay operation $\mathcal{U}_D$ can be described using Eqs. (27)-(28):

$$\mathcal{U}_D : A(t) \to B(t'), t' = t + \triangle t, \tag{27}$$

$$B(t') = \begin{cases} 0, & 0 \le t' \le \triangle t - 1 \\ A(t), & \triangle t \le t' \le 2^l - 1 \end{cases}. \tag{28}$$

Both the time information t and t' of the original and delayed audio signal need to be in the range $[0, 2^l - 1]$, where

apparently $\triangle t \le t' = t + \triangle t \le 2^l + \triangle t - 1$ is beyond the interval. Therefore, we separate the time information into two parts, i.e. $[\triangle t, 2^l - 1]$ and $[2^l, 2^l + \triangle t - 1]$, for discussion.

As shown in Eq. (28), when $t' \in [\triangle t, 2^l - 1]$, $B(t') = A(t)$, in which case, the time information t' can be obtained directly by means of the ADDER module on time content. However, when $t' \in [2^l, 2^l + \triangle t - 1]$, we need to employ the carry qubit of the ADDER module to achieve the interval shift $[2^l, 2^l + \triangle t - 1] - 2^l = [0, \triangle t - 1]$ and set the amplitudes $S_t = 0$ as the delay period.

The general circuit for the quantum audio signal delay operation is presented in Fig. 9. In addition, Fig. 10 presents a simple example of the FRQA-based signal delay, where (a) is the input audio signal ($l = 3$ and $q = 3$) and (b) is the signal that is delayed by 2 time units. The quantum circuit for this operation can be constructed as in Fig. 11.

The implementation of signal delay consists of one ADDER module (on time content) and $2q$ Toffoli gates, so the complexity of this operation is $28l + 12q - 12$.

3.4 Signal reversal

Signal reversal is the process of reversing a selected audio signal such that the end of the signal is heard first and the beginning last. This operation can be used to create interesting sound effects or make small portions of inappropriate language unintelligible. Based on the FRQA state, we can define the signal reversal operation on quantum computers and construct the quantum circuit to accomplish it.
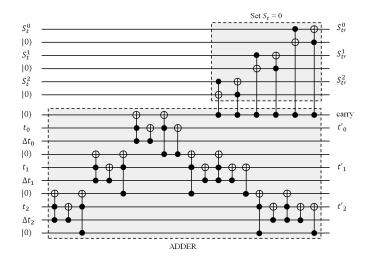
Fig. 11. Circuit construction to execute the quantum audio signal delay example in Fig. 10.

Fig. 12. (a) General circuit for quantum audio signal reversal and (b) a simple example of this operation.

Fig. 13. (a) Circuit construction to execute the restricted quantum audio signal reversal and (b) a simple example of this operation.

Assuming that $|A\rangle$ is an FRQA audio signal in the form presented in Eq. (3), the signal reversal operation $\mathcal{U}_R$ applied on $|A\rangle$ produces an output of the form:

$$|A_R\rangle = \frac{1}{2^{l/2}} \sum_{t=0}^{2^l-1} |S_t\rangle \otimes |\bar{t}\rangle,$$

where

$$|\bar{t}\rangle = |\overline{t_0 t_1 \dots t_{l-1}}\rangle, \bar{t_i} \in \{0, 1\}. \tag{29}$$

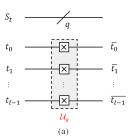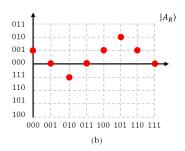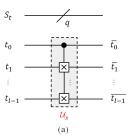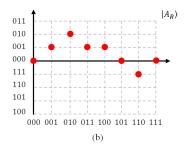Hence, the signal reversal operation $\mathcal{U}_R$ can be defined as below:

$$
\begin{aligned}
\mathcal{U}_R(|A\rangle) &= \frac{1}{2^{l/2}} \sum_{t=0}^{2^l-1} |S_t\rangle \otimes \mathcal{U}_R(|t\rangle) \\
&= \frac{1}{2^{l/2}} \sum_{t=0}^{2^l-1} |S_t\rangle \otimes |\bar{t}\rangle.
\end{aligned}
\tag{30}
$$

The general circuit to execute the signal reversal operation is presented in Fig. 12(a). Although it is glaring that this operation can be directly implemented by using l NOT gates, in order to further enunciate its realization, we use the quantum audio signal in Fig. 10(a) as input signal whose output is presented in Fig. 12(b). As seen, the result shows that all the time points are played in a reversed order.

In addition, some control condition constraints on the time content will allow us to confine the execution of the reversal operation $\mathcal{U}_R$ to a desired period. For example, in Fig. 13(a), the time wire t_0 is employed as the control wire to confine the reversal operation to the last half of the audio signal. As seen from the result in Fig. 13(b) that only the last 4 time points in Fig. 10(a) are reversed and the others are retained as in the input signal.

It should be mentioned that the complexity of the extended operation increases with the number of control qubits. In the worst case, the implementation of this operation requires an $(l - 1)$-controlled NOT gate, thereby the complexity is $12l - 23$.

4 CONCLUSIONS

The characteristics of the physical support used to embody information decide the power of a certain computational paradigm. Quantum computation has emerged as a powerful tool for efficient and secure information processing. As with any effort to process any type of information, the requirements for its encoding must be formalised. The study presented in this work explored the use of quantum mechanical properties that are at the core of quantum computers in order to encode, store, and manipulate audio signals. Inspired by the rudiments employed to capture digital audio signals, the first known effort to represent quantum audio signals (the QRDA) was broached. While QRDA extended audio processing into the quantum computing domain, its numerical representation of amplitude values only offers an encoding method for unipolar (i.e. non-negative) numbers so that its operations are susceptible to errors. The FRQA proposed in this study offers an accurate computation and extensive signal operations alternative to the QRDA. The FRQA protocol encodes amplitude values in two's complement notation so that the arithmetic advantages of doing so can be utilized to facilitate the construction of quantum circuits for amplitude transformations. By exploiting this, various operations that allow for designing advanced audio processing applications can be proposed.

Based on this new FRQA representation, basic manipulations targeting the amplitude and time components of the audio signals such as the quantum audio signal addition, inversion, reversal, and delay were formalised and insightful examples depicting their utility were also presented and discussed. The quantum circuits required to execute these basic operations on quantum audio signals were also presented and the computational requirements for their realization were analyzed.

In future work, the results from this study can be extended towards the following directions. First, considering the importance of basic operations in quantum audio processing, the operations presented in the study need to improved and new ones added to ensure the execution of more advanced algorithmic tasks. Secondly, the possibility of using quantum-based signal processing operations, such as quantum Fourier transformations [14] and wavelet transformations [17] which have proven well-studied in other quantum information processing applications to facilitate quantum audio manipulation needs to explored. Thirdly, the "silent" quantum movies proposed in [18] provided impetus for the need of sound or audio on the quantum computing paradigm. A useful integration of movie strip (technically a collection of quantum images) with audio signal information will complete the quantum movie representation and facilitate more interesting applications. Finally, when the basic quantum audio signal operations are perfected, they could serve as the building block for the realization of more sophisticated algorithms used for particular applications such as watermarking, steganography, and encryption. These improvements will facilitate multimedia security during quantum information transmission.

ACKNOWLEDGMENTS

REFERENCES

[1] Feynman R. Simulating physics with computers. International Journal of Theoretical Physics, 1982, 21: 467-488

[2] Deutsch D. Quantum theory, the church-turing principle and the universal quantum computer. In: Proceedings of the Royal Society of London A, 1985, 97-117

[3] Shor P. Algorithms for quantum computation: discrete logarithms and factoring. In: Proceedings of the 35th Annual Symposium on Foundations of Computer Science, 1994, 124-134

[4] Grover L. A fast quantum mechanical algorithm for database search. In: Proceedings of the 28th Annual ACM Symposium on Theory of Computing, 1996, 212-219

[5] Vlasov A. Quantum computations and images recognition. arXiv:quant-ph/9703010, 1997

[6] Lugiato L, Gatti A, Brambilla E. Quantum imaging. Journal of Optics B: Quantum and Semiclassical Optics, 2002, 4: 176-183

[7] Yan F, Iliyasu A, Venegas-Andraca S. A survey of quantum image representations. Quantum Information Processing, 2016, 15: 1-35

[8] Zölzer U. Digital audio signal processing. John Wiley & Sons Ltd, United Kingdom, 2008

[9] Wang J. QRDA: quantum representation of digital audio. International Journal of Theoretical Physics, 2016, 55: 1622-1641

[10] Barenco A, Bennett C, Cleve R, et al. Elementary gates for quantum computation. Physical Review A, 1995, 52: 3457-3467

[11] Venegas-Andraca S, Bose S. Storing, processing, and retrieving an image using quantum mechanics. In: Proceedings of SPIE Conference of Quantum Information and Computation, 2003, 134-147

[12] Le P, Dong F, Hirota K. A flexible representation of quantum images for polynomial preparation, image compression, and processing operations. Quantum Information Processing, 2011, 10: 63-84

[13] Vlatko V, Adriano B, Artur E. Quantum networks for elementary arithmetic operations. Physical Review A, 1996, 54: 147-153

[14] Nielsen M, Chuang I. Quantum computation and quantum information. Cambridge University Press, United Kingdom, 2000

[15] Wang J, Jiang N, Wang L. Quantum image translation. Quantum Information Processing, 2015, 14: 1589-1604

[16] Jiang N, Wu W, Wang L. The quantum realization of arnold and fibonacci image scrambling. Quantum Information Processing, 2014, 13: 1223-1236

[17] Fijany A, Williams C. Quantum wavelet transforms fast algorithms and complete circuits. Lecture Notes in Computer Science, 1999, 1509: 10-33

[18] Iliyasu A, Le P, Dong F, et al. A framework for representing and producing movies on quantum computers. International Journal of Quantum Information, 2011, 9: 1459-1497