

## Reversible Arithmetic Coding for Quantum Data Compression

Isaac L. Chuang, *Member, IEEE*, and  
Dharmendra S. Modha, *Member, IEEE*

**Abstract**—We study the problem of compressing a block of symbols (a block quantum state) emitted by a memoryless quantum Bernoulli source. We present a simple-to-implement quantum algorithm for projecting, with high probability, the block quantum state onto the *typical subspace* spanned by the leading eigenstates of its density matrix. We propose a fixed-rate quantum Shannon–Fano code to compress the projected block quantum state using a per-symbol code rate that is slightly higher than the von Neumann entropy limit. Finally, we propose quantum arithmetic codes to efficiently implement quantum Shannon–Fano codes. Our arithmetic encoder and decoder have a cubic circuit and a cubic computational complexity in the block size. Both the encoder and decoder are quantum-mechanical inverses of each other, and constitute an elegant example of reversible quantum computation.

**Index Terms**—Arithmetic coding, noiseless coding, quantum communication, quantum computation, quantum information theory, quantum measurement, reversible computation, Schumacher coding.

### I. INTRODUCTION

Modern information theory makes fundamental assumptions concerning the physical representation and processing of information. Following the lead of classical mechanics, modern information theory assumes that an information bit can exist in either one of two states, say, 0 or 1. However, classical physics is known to fail spectacularly under many circumstances, for example, when the objects being described are very small or have very large energies. This regime of physics is described by the laws of quantum mechanics. Conventional information theory fails to properly describe how information can be represented and transformed in such physical systems, and must be replaced by an appropriate quantum analog: quantum information theory. In contrast to the classical information bit, a quantum information bit can exist in a superposition of two orthogonal quantum states.

Quantum information can, in principle, provide significant advantages for certain problems. For example, quantum algorithms for calculating discrete logarithms (see Shor [1]) and searching unsorted databases (see Grover [2]) have been discovered which are faster than their classical counterparts. Quantum bits, in contrast to classical bits, cannot be copied perfectly, and this is useful in such tasks as quantum cryptography (see Bennett, Brassard, and Ekert [3]). Furthermore, Fuchs [4] has shown that, rather unexpectedly, there exist certain quantum communication channels for which the optimal classical information transmission rate is achieved only using nonorthogonal quantum states as the symbols. Finally, quite surprisingly, quantum error correction codes have been developed (see Calderbank *et al.* [5] and references therein). Such codes might provide the key technology needed to prevent decoherence of quantum states, and, hence, a way to realize large-scale quantum computing devices. For excellent reviews of the field, see, for example, Bennett and Shor [6], Rieffel and Polak [7], and Steane [8].

The problem of compression is central to storage and transmission of quantum data. In this correspondence, we investigate quantum al-

gorithms for compressing a sequence of symbols emitted by a memoryless quantum Bernoulli source. The basis for compression of classical data is Shannon's noiseless coding theorem [9]: If the per-symbol code rate is slightly larger than the *Shannon entropy*, then there exists a block code (with sufficiently large block size) such that the compressed message can be recovered with *probability* close to unity. A number of algorithms such as Huffman coding, Shannon–Fano coding, enumerative coding, arithmetic coding, and Lempel–Ziv coding for achieving the Shannon entropy limit are known in the classical case; see, for example, Cover and Thomas [10]. In comparison, the field of quantum data compression is still nascent. The quantum analog to Shannon's theorem is Schumacher's theorem [11]: If the per-symbol code rate is slightly larger than the von Neumann entropy, then there exists a block code (with sufficiently large block size) such that the compressed message can be recovered with *average fidelity* close to unity. The similarity of the two theorems makes it possible to use, to a limited extent, classical algorithms for performing quantum data compression. However, classical compression codes cannot immediately be translated into quantum versions; for example, in order to preserve the coherent quantum state, all operations performed on the data must be reversible and must not entangle the state with any temporary variables. Furthermore, it is essential that the original state must be entirely obliterated in producing the encoded state, because quantum states cannot be cloned (see Wootters and Zurek [12] and Dieks [13]). Cleve and DiVincenzo [14] have proposed a block coding algorithm, which is, in fact, a generalization of the classical enumerative coding of Cover [15] and Schalkwijk [16]. Recently, Braunstein *et al.* [17] have studied quantum extensions of Huffman coding.

The statistics underlying a quantum memoryless Bernoulli source is completely captured by its density matrix. The fundamental idea behind quantum data compression is to analyze the eigenstructure of the joint density matrix associated with a block quantum state emitted by the quantum memoryless Bernoulli source. As our *first* contribution, in Section III, we present a quantum-mechanical algorithm for projecting the block quantum state onto the subspace spanned by the most important eigenstates of the joint density matrix, that is, the eigenstates corresponding to the largest eigenvalues. Our algorithm computes, in parallel, an indicator function that is 0 if the eigenstate is typical and 1 otherwise. By making a measurement on the quantum bit associated with the indicator function, with very high probability, we project the block quantum state onto the *typical subspace* spanned by the leading eigenstates. Our theoretical results represent a strengthening of Schumacher's pioneering result in that they hold for fixed block sizes and they deliver a rate of convergence.

The projection onto the typical subspace wipes out the trailing eigenstates, and, hence, the projected quantum state lies in the low-dimensional typical subspace. Consequently, each leading eigenstate can be represented using roughly the logarithm of the dimension of the typical subspace. The central problem of quantum data compression is to efficiently compute such low-dimensional representations. As our *second* contribution, in Section IV, we propose a quantum version of the classical Shannon–Fano code to represent and, hence, compress the projected block quantum state using a per-symbol code rate that is slightly higher than the von Neumann entropy limit. Conceptually, we achieve data compression by dimensionality reduction.

As our *third* contribution, in Section IV, we propose quantum arithmetic codes to efficiently implement quantum Shannon–Fano codes. Our arithmetic encoder and decoder use a certain finite-precision arithmetic process that is inspired by classical arithmetic coding. For original papers on classical arithmetic coding see Pasco [18] and Rissanen [19] and, for reviews, see Bell, Cleary, and Witten

Manuscript received July 12, 1999; revised December 5, 1999.

The authors are with IBM Almaden Research Center, San Jose, CA 95120-6099 USA (e-mail: {ichuang; dmodha}@almaden.ibm.com).

Communicated by A. M. Barg, Associate Editor for Coding Theory.

Publisher Item Identifier S 0018-9448(00)03138-2.

[20], Langdon [21], and Rissanen and Langdon [22]. The novelty of quantum arithmetic coding is to implement finite-precision arithmetic processes in a quantum-mechanically reversible fashion. Our arithmetic encoder and decoder have a cubic circuit and a cubic computational complexity in the block size. The proposed encoder and decoder are quantum-mechanical inverses of each other, and constitute a very satisfying example of reversible quantum computation. For fundamental references on reversible computation see Bennett [23], [24] and Toffoli [25]. For reversible computation in relation to quantum data compression see Cleve and DiVincenzo [14].

## II. PRELIMINARIES

We begin by reviewing the definitions of quantum sources and quantum states relevant to the present coding problem. We also present precise quantum counterparts for the classical notions of fidelity and entropy, and describe how encoding and decoding is done using quantum computation. In addition to establishing the notation used in the remainder of the correspondence, this preliminary discussion will highlight the special quantum aspects of our coding task.

### A. Memoryless Bernoulli Sources

A classical memoryless Bernoulli source emits a sequence of independent and identically distributed (i.i.d.) symbols each of which is 0 with probability  $p$  or 1 with probability  $1 - p$ , where  $0 \leq p \leq 1$ . The problem of classical noiseless data compression is to transmit sequences of samples emitted by such a source using a minimal number of bits. Shannon [9] established that on average each symbol can be transmitted in (slightly larger than)

$$H(p) = -p \log p - (1 - p) \log (1 - p)$$

bits with high probability of correct reception, where  $H(p)$  is known as the *Shannon entropy*. In this correspondence, all logarithms will be base 2, as is usual in information theory.

A pure two-dimensional quantum state is known as a *quantum bit* or *qubit*. It is mathematically represented by a unit norm vector in a two-dimensional complex vector space (called a Hilbert space) written as  $\mathcal{H}_2$ . A qubit may be thought of as a column vector, and is usually written using Dirac's *ket* notation; for example,  $|\phi\rangle$  denotes a qubit. The conjugate transpose of  $|\phi\rangle$ , namely,  $\langle\phi|$ , is written in Dirac's *bra* notation as  $\langle\phi|$ . The inner product between an ordered pair of qubits  $(\phi, \varphi)$  is written in Dirac's *bra-ket* notation as

$$\langle\phi|\varphi\rangle.$$

We write the *fidelity* between a pair of qubits  $(\phi, \varphi)$  as

$$\langle\phi|\varphi\rangle \langle\varphi|\phi\rangle = |\langle\phi|\varphi\rangle|^2.$$

Let  $|\phi_0\rangle$  and  $|\phi_1\rangle$  denote two arbitrary qubits. A quantum memoryless Bernoulli source emits a sequence of i.i.d. symbols each of which is  $|\phi_0\rangle$  with probability  $p$  or  $|\phi_1\rangle$  with probability  $1 - p$ , where  $0 \leq p \leq 1$ . The per-symbol distribution of this source is described by the *density matrix*

$$\rho = p|\phi_0\rangle\langle\phi_0| + (1 - p)|\phi_1\rangle\langle\phi_1|$$

where  $|\phi\rangle\langle\phi|$  denotes the  $2 \times 2$  matrix given by the outer product between the vector  $|\phi\rangle$  and its conjugate transpose  $\langle\phi|$ .

The problem of (pure-state) quantum noiseless data compression is to transmit such sequences of symbols with high fidelity, using a min-

imal number of quantum bits. According to Schumacher's theorem [11], on average each symbol can be transmitted in (slightly larger than)

$$S(\rho) = -\text{Tr}(\rho \log \rho)$$

quantum bits with high probability of correct reception, where  $S(\rho)$  is known as the *von Neumann entropy*. A surprising contrast between the classical and the quantum cases is that

$$S(\rho) \leq H(p)$$

where the equality is achieved if and only if the quantum states  $|\phi_0\rangle$  and  $|\phi_1\rangle$  are orthogonal. Intuitively, this holds since two nonorthogonal qubits cannot be distinguished with certainty by measurement.

We will let  $P$  and  $E$  denote the probability and the expectation, respectively, with respect to the quantum memoryless source.

### B. Blocks of Symbols

We shall focus on compressing a block of  $n$  symbols emitted by the quantum source. Let

$$|\psi_{[1, n]}\rangle \equiv |\psi_1\rangle \otimes |\psi_2\rangle \otimes \cdots \otimes |\psi_n\rangle \quad (1)$$

be a sequence of symbols emitted by the quantum memoryless source, where  $\otimes$  denotes the tensor product, and  $|\psi_i\rangle$  represents the  $i$ th sample from the source, a *random* state which is either  $|\phi_0\rangle$  with probability  $p$  or  $|\phi_1\rangle$  with probability  $1 - p$ .

This notation is slightly unconventional. Usually, a quantum state written using the ket notation  $|\cdot\rangle$  is definite or pure: it has a von Neumann entropy of zero. However, as we have written above, we shall find it convenient to use similar notation to denote *mixed* or random quantum states (random mixtures of pure states); such states will be written with their label in **bold**, in analogy to the often used classical notation for random variables. Usually, in the physics literature, a mixed state is denoted *implicitly* by the corresponding density matrix that captures all the statistical information present in the state. However, in an information-theoretic context, we are interested in the *explicit* "message" contained in a mixed state, and this notion is conveniently captured by dealing directly with the underlying mixed state. By writing the mixed state using the ket notation, we can think of it as a random probabilistic linear combination of the eigenstates of the density matrix, see, (3). Such an explicit representation clarifies the meaning of the eigenvalues of the density matrix as the "expected projections" of the mixed state along the respective eigenstates as in (4), and makes the ideas in Sections III and IV more accessible to a classical information theorist.

Let  $X$  denote a binary string of length  $n$ . We will let  $X_{[i, j]}$  denote bits  $i$  through  $j$  of  $X$ . For brevity, as an alternative to  $X_{[1, k]}$ , we may sometimes write  $X_{[k]}$  to represent the first  $k$  significant bits of  $X$ . Similar notation, when used with qubits, should be clear by analogy. For example, we will write  $|\boldsymbol{\psi}_{[1, n]}\rangle$  as  $|\boldsymbol{\psi}_{[n]}\rangle$ .

Throughout this correspondence, for every  $a > 0$ , we write

$$0_a = \underbrace{00 \cdots 0}_a \quad \text{and} \quad 1_a = \underbrace{11 \cdots 1}_a.$$

When it is clear how many zeros or ones are necessary, sometimes the subscript will be suppressed.

### C. Eigenstructure of Block Quantum Systems

Although  $|\phi_0\rangle$  and  $|\phi_1\rangle$  may be nonorthogonal, there always exists a basis for  $\mathcal{H}_2$  in which the same per-symbol distribution from the quantum Bernoulli source can be obtained from a different source that

emits one of two *orthogonal* states  $|0\rangle$  and  $|1\rangle$  with probabilities  $\lambda_0$  and  $\lambda_1$  (instead of  $|\phi_0\rangle$  and  $|\phi_1\rangle$  with probabilities  $p$  and  $1-p$ ). The special basis  $\{|0\rangle, |1\rangle\}$  is defined as follows. The density matrix  $\rho$  which characterizes the per-symbol distribution of the original quantum Bernoulli source is self-adjoint, positive-definite, and has unit trace. Hence, its eigenvalues, say,  $\lambda_0 \geq \lambda_1$ , are real and nonnegative, and sum to one. If  $\lambda_0 \neq \lambda_1$ , choose the states  $|0\rangle$  and  $|1\rangle$  to be eigenvectors of  $\rho$  corresponding to  $\lambda_0$  and  $\lambda_1$ , respectively. By construction, these states are orthonormal, and thus form a basis for  $\mathcal{H}_2$ . If  $\lambda_0 = \lambda_1$ , then select  $\{|0\rangle, |1\rangle\}$  to be any orthonormal basis of  $\mathcal{H}_2$ . In this basis, the original symbols  $|\phi_0\rangle$  and  $|\phi_1\rangle$  are given as

$$\begin{aligned} |\phi_0\rangle &= \langle 0|\phi_0\rangle |0\rangle + \langle 1|\phi_0\rangle |1\rangle \\ |\phi_1\rangle &= \langle 0|\phi_1\rangle |0\rangle + \langle 1|\phi_1\rangle |1\rangle \end{aligned}$$

and the density matrix can be written as

$$\rho = \lambda_0 |0\rangle\langle 0| + \lambda_1 |1\rangle\langle 1|$$

where

$$\lambda_0 = 1 - \lambda_1 = p|\langle 0|\phi_0\rangle|^2 + (1-p)|\langle 0|\phi_1\rangle|^2.$$

Furthermore, we can now write

$$S(\rho) = H(\lambda_0).$$

For  $1 \leq i \leq n$ , we can write the mixed quantum state  $|\psi_i\rangle$  as

$$|\psi_i\rangle = \langle 0|\psi_i\rangle |0\rangle + \langle 1|\psi_i\rangle |1\rangle$$

where  $\langle 0|\psi_i\rangle$  and  $\langle 1|\psi_i\rangle$  are random quantities such that

$$E|\langle 0|\psi_i\rangle|^2 = \lambda_0 \quad \text{and} \quad E|\langle 1|\psi_i\rangle|^2 = \lambda_1. \quad (2)$$

The sequence of symbols  $|\psi_{[n]}\rangle$  in (1) is a mixed quantum state in the Hilbert space  $\mathcal{H}_2^{\otimes n} = \otimes_{i=1}^n \mathcal{H}_2$ . Using properties of the tensor product, we can write

$$\begin{aligned} |\psi_{[n]}\rangle &= \otimes_{i=1}^n |\psi_i\rangle \\ &= \otimes_{i=1}^n \sum_{\chi_i=0}^1 \langle \chi_i|\psi_i\rangle |\chi_i\rangle \\ &= \sum_{\chi \in \{0,1\}^n} \langle \chi|\psi_{[n]}\rangle |\chi\rangle, \end{aligned} \quad (3)$$

where

$$|\chi\rangle = \otimes_{i=1}^n |\chi_i\rangle \quad \text{and} \quad \langle \chi|\psi_{[n]}\rangle = \prod_{i=1}^n \langle \chi_i|\psi_i\rangle.$$

Now, we have that

$$\begin{aligned} E \left| \langle \chi|\psi_{[n]}\rangle \right|^2 &\stackrel{\text{a)}}{=} \prod_{i=1}^n E |\langle \chi_i|\psi_i\rangle|^2 \\ &\stackrel{\text{b)}}{=} \prod_{i=1}^n \lambda_0^{(1-\chi_i)} \lambda_1^{\chi_i} \\ &\equiv \Lambda(\chi; \lambda_0, \lambda_1) \end{aligned} \quad (4)$$

where a) follows from independence and b) follows from (2). We may think of the  $2^n$  quantum states  $|\chi\rangle$ ,  $\chi \in \{0,1\}^n$ , as the *eigenstates* of the tensor product density matrix  $\rho^{\otimes n} = \otimes_{i=1}^n \rho$ , and the numbers

$\Lambda(\chi; \lambda_0, \lambda_1)$  as the corresponding *eigenvalues*. Note that the eigenstates  $|\chi\rangle$ ,  $\chi \in \{0,1\}^n$  constitute an orthonormal basis for the Hilbert space  $\mathcal{H}_2^{\otimes n}$ .

It follows from (3) that we can write the message  $|\psi_{[n]}\rangle$  to be encoded as a linear superposition of the  $2^n$  eigenstates:  $|\chi\rangle$ ,  $\chi \in \{0,1\}^n$ . The “randomness” of the message is completely contained in the coefficients  $\langle \chi|\psi_{[n]}\rangle$ , and the eigenstates are not a function of the particular message to be transmitted. Physically, the randomness is embedded entirely in the complex amplitude associated with each path or eigenstate.

#### D. Computation: Encoding and Decoding

The encoding and decoding of classical information is specified by a mapping between bit-strings. Similarly, for quantum information, one specifies a mapping between quantum states; however, additional reversibility constraints must be satisfied. For example, a reversible transformation conserves energy. Since quantum states are mathematically represented by vectors with unit norm, reversible transformations must preserve norm. It also turns out that with the appropriate description of the system, the most general transformation preserves orthogonality between states. If we think of the quantum states in  $\mathcal{H}_2^n$  as  $2^n$ -dimensional column vectors, then most general transformations are described by  $2^n \times 2^n$  unitary matrices acting on the Hilbert space of the quantum states. Recall that a unitary matrix is one whose conjugate transpose is its inverse.

This model of computation subsumes classical computation, because mappings between bit-strings can be described as permutation matrices acting on the basis elements of the Hilbert space. Of course, unitary transforms are always invertible or reversible; nonetheless, all irreversible (classical) computation can be made reversible with only a polynomial amount of overhead; see, Bennett [23]. Conversely, however, not all unitary transforms represent reversible classical computation. In other words, not all unitary transforms can be described by permutation matrices. A unitary transform can be completely specified by its action on all the basis elements of a Hilbert space. Transformations which are not permutations take basis elements to superpositions of basis elements; these are at the heart of the speedup of quantum computation and quantum error correction, but will not be of much concern for our problem.

Quantum algorithms are generally very difficult to construct, but choosing the eigenstates  $|\chi\rangle$ ,  $\chi \in \{0,1\}^n$ , as the basis vastly simplifies the descriptions of our encoding and decoding transforms. In this special basis, we need only employ unitary transformations which are permutations of basis elements to achieve our goal (why this is true is not obvious, but will be demonstrated later). However, these transforms shall be applied to input states which are generally in non-classical superpositions of basis elements. As suggested by Deutsch [26], it is convenient to think of what happens as being “quantum parallelism”—for an input  $|\phi\rangle = a|0\rangle + b|1\rangle$ , a computation  $U$  produces  $U|\phi\rangle = aU|0\rangle + bU|1\rangle$ , by linearity. Thus in this example, we can think of *two* “classical” computations happening in parallel, one with input  $|0\rangle$  and the other with  $|1\rangle$ , with the two computational paths being weighted with complex amplitudes  $a$  and  $b$ , respectively. Similar observations hold for arbitrarily large states. As long as  $U$  is simply a permutation (as it will be in our case), these different paths never interfere and a coherent quantum state is maintained.

We shall symbolically describe encoding and decoding unitary transforms for quantum information using algorithms which at first glance look very classical, but in reality, are specially constructed to be quantum. Three characteristics make our algorithms quantum-mechanical. First, they are reversible; this is required as previously explained. Second, they completely erase their inputs; this is a necessity because quantum states cannot be cloned [12], [13], and thus there is no sense to a sender sending a faithfully encoded quantum

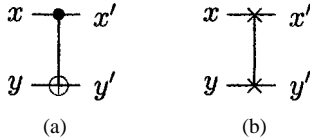


Fig. 1. Two quantum gates which are used in later quantum circuits. (a) The controlled-NOT gate, which produces  $x' = x$  and  $y' = x \oplus y$  ( $\oplus$  denoting (bitwise) addition modulo 2). (b) The swap gate, for which  $x' = y$  and  $y' = x$ . Note that time goes from left to right.

state elsewhere without erasing her own knowledge of that state in the process. Third, our algorithms produce no information other than the encoded (or decoded) state, which would allow differentiation between computational paths. Producing such *entanglement* would ruin the superposition which is being encoded, because any potential for obtaining “which path” information implies the existence of a physical measurement which would (at least partially) collapse the superposition state. Fundamentally, this nondisturbance requirement is deeply related to the no-cloning theorem, and it is a subtle, but very important point which we shall return to with further discussion later.

Another model we shall employ for clarity of exposition is that of quantum circuits, which succinctly capture the same information as the algorithms, and often effectively convey additional structural information about the procedure. A wide body of knowledge about quantum circuits exists (see Barenco *et al.* [27] and Barenco [28]) but we shall draw from it only the subset which is convenient for describing reversible classical circuits, including the controlled-NOT and swap gates, as shown in Fig. 1.

### E. Representing Real Numbers as Eigenstates

One final piece of notation will be useful for expressing our coding procedure. Suppose we are given a fractional number  $\zeta$ ,  $0 \leq \zeta \leq 1$ . Let

$$\zeta = 0.\zeta_1\zeta_2\zeta_3 \dots$$

denote a binary representation of the number. Then, we can associate this number to a pure quantum state

$$|\zeta\rangle = |\zeta_1\rangle \otimes |\zeta_2\rangle \otimes |\zeta_3\rangle \otimes \dots$$

in the infinite-dimensional Hilbert space  $\mathcal{H}_2^{\otimes \infty}$ . This allows us to represent a fractional real number as a quantum state.

### III. TYPICAL SUBSPACE

The basic idea of quantum data compression is that the eigenstates associated with smaller eigenvalues can be discarded without incurring significant loss of average fidelity. We will attain this goal by employing a measurement of a certain quantum observable associated with the given message, as described below.

#### A. Measurement

Let  $w(\chi)$ ,  $\chi \in \{0, 1\}^n$ , denote the Hamming weight of the string  $\chi$ , that is, the number of ones in the string. It follows from (4) that we can write

$$\Lambda(\chi; \lambda_0, \lambda_1) = \lambda_0^{n-w(\chi)} \lambda_1^{w(\chi)}. \quad (5)$$

Now, since  $\lambda_0 \geq \lambda_1$ , it follows from (5) that the smaller the Hamming weight of an eigenstate the larger the eigenvalue associated with the eigenstate. Let  $\tau \geq 0$  denote a *truncation threshold*; we will determine

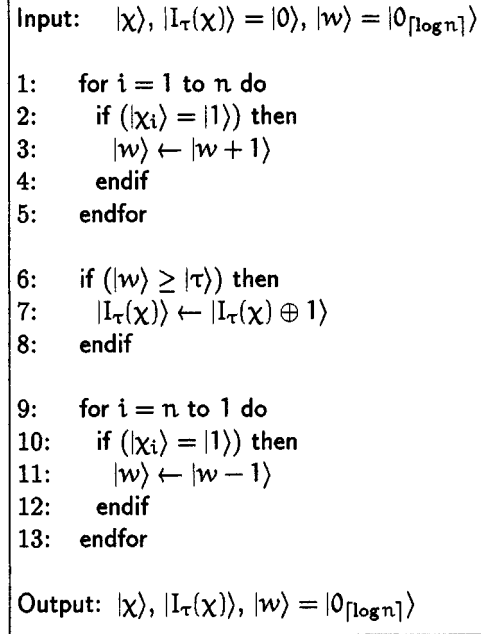


Fig. 2. A symbolic algorithm or “psuedocode” for computing (6). “ $\leftarrow$ ” denotes an assignment operation; when describing a pre-existing state or comparison operation, we use “ $=$ .” A temporary quantum register  $|w\rangle$  of length  $\lceil \log n \rceil$  is used; this register is *initialized* and *finalized* to  $|0_{\lceil \log n \rceil}\rangle$ . The precise value of  $\tau$  should satisfy (8), and will be specified later in (16).

an appropriate value for  $\tau$  in the sequel. Let  $\mathcal{G}_\tau$  and  $\mathcal{B}_\tau$  denote the sets of “good” and “bad” eigenstates such that

$$\begin{aligned} \mathcal{G}_\tau &= \{\chi | w(\chi) < \tau\} \\ \mathcal{B}_\tau &= \{\chi | w(\chi) \geq \tau\}. \end{aligned}$$

With appropriate values of  $\tau$ , the subspace spanned by the good eigenstates, namely,

$$\text{span} \{|\chi\rangle | \chi \in \mathcal{G}_\tau\}$$

will become the typical subspace that contains most of the information present in an average quantum message.

For every eigenstate  $|\chi\rangle$ ,  $\chi \in \{0, 1\}^n$ , let  $I_\tau(\chi)$  denote the good–bad indicator function such that

$$I_\tau(\chi) = \begin{cases} 0, & \text{if } \chi \in \mathcal{G}_\tau \\ 1, & \text{if } \chi \in \mathcal{B}_\tau. \end{cases}$$

We now compute the following transformation:

$$|\chi, 0\rangle \rightarrow |\chi, I_\tau(\chi)\rangle. \quad (6)$$

We exhibit a quantum algorithm for computing (6) in Fig. 2, which is implemented by the quantum circuit in Fig. 3. This algorithm makes use of subroutines previously described in the literature [14] for conditional addition and subtraction, and comparison. Using (3), the action of the algorithm on the quantum message can be written as

$$|\psi_{[n]}, 0\rangle \rightarrow \sum_{\chi \in \{0, 1\}^n} \langle \chi | \psi_{[n]} \rangle |\chi, I_\tau(\chi)\rangle \equiv |\hat{\psi}_{[n]}, I_\tau\rangle \quad (7)$$

where  $|\hat{\psi}_{[n]}, I_\tau\rangle$  is an output state in which  $I_\tau$  is now a function of  $\hat{\psi}_{[n]}$  and thus, in general, is *entangled* with it. Let

$$\left\{ |I_\tau\rangle \stackrel{m}{=} |0\rangle \right\} \quad \text{or} \quad \left\{ |I_\tau\rangle \stackrel{m}{=} |1\rangle \right\}$$

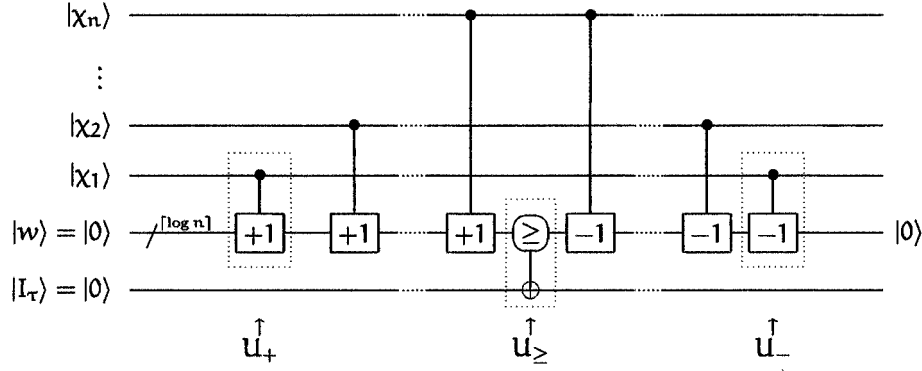


Fig. 3. A quantum circuit implementing the algorithm in Fig. 2. The gates labeled as  $U_+$  and  $U_-$  implement lines 2–4 and lines 10–12 of the algorithm, respectively. These gates are quantum-mechanical inverses of each other. The gate  $U_{\geq}$  implements lines 6–8 of the algorithm. As in Barenco [28], we generally use rounded symbols to denote the control qubits, and boxed symbols to indicate the targets, with the exception of “ $\oplus$ ” which always sits on a target. The  $/\lceil \log n \rceil$  notation indicates a wire bundle with  $\lceil \log n \rceil$  qubits.

denote the two possible events corresponding to measuring  $|I_{\tau}\rangle$  to be  $|0\rangle$  or  $|1\rangle$ , respectively. We now determine the truncation threshold  $\tau$  to ensure that the probability of the event  $\{|I_{\tau}\rangle \stackrel{m}{=} |0\rangle\}$  is close to 1.

**Theorem III.1:** Assume that  $1/2 < \lambda_0 < 1$ . For a fixed  $n \geq 1$  and a fixed  $\delta > 0$ , if we set

$$\tau \geq \left\lceil n \left( \lambda_1 + \frac{\delta}{\log \lambda_0 / \lambda_1} \right) \right\rceil \quad (8)$$

then

$$\begin{aligned} P \left\{ |I_{\tau}\rangle \stackrel{m}{=} |0\rangle \right\} &= 1 - P \left\{ |I_{\tau}\rangle \stackrel{m}{=} |1\rangle \right\} \\ &\geq 1 - 2^{-(2n\delta^2)/(\log \lambda_0 / \lambda_1)^2}. \end{aligned} \quad (9)$$

*Proof:* We adapt the technique in Hoeffding [29]

$$\begin{aligned} &P \left\{ |I_{\tau}\rangle \stackrel{m}{=} |1\rangle \right\} \\ &= \sum_{\chi \in \mathcal{B}_{\tau}} E |\langle \chi | \psi_{[n]} \rangle|^2 \\ &\stackrel{a)}{=} \sum_{\chi \in \mathcal{B}_{\tau}} \Lambda(\chi; \lambda_0, \lambda_1) \\ &\stackrel{b)}{\leq} \sum_{\{\chi | -\log \Lambda(\chi; \lambda_0, \lambda_1) \geq n(S(\rho) + \delta)\}} \Lambda(\chi; \lambda_0, \lambda_1) \\ &\stackrel{c)}{\leq} \min_{\gamma > 0} \left[ \sum_{\{\chi | -\log \Lambda(\chi; \lambda_0, \lambda_1) \geq n(S(\rho) + \delta)\}} \cdot 2^{\gamma(-\log \Lambda(\chi; \lambda_0, \lambda_1) - n(S(\rho) + \delta))} \Lambda(\chi; \lambda_0, \lambda_1) \right] \\ &\leq \min_{\gamma > 0} \left[ 2^{-\gamma n \delta} \sum_{\chi \in \{0, 1\}^n} \cdot 2^{\gamma(-\log \Lambda(\chi; \lambda_0, \lambda_1) - nS(\rho))} \Lambda(\chi; \lambda_0, \lambda_1) \right] \\ &= \min_{\gamma > 0} \left[ 2^{-\gamma n \delta} \prod_{i=1}^n \left( \lambda_0 2^{\gamma(-\log \lambda_0 - S(\rho))} + \lambda_1 2^{\gamma(-\log \lambda_1 - S(\rho))} \right) \right] \\ &\stackrel{d)}{\leq} \min_{\gamma > 0} \left[ 2^{-\gamma n \delta} \prod_{i=1}^n \left( 2^{(1/8)\gamma^2 (\log(\lambda_0/\lambda_1))^2} \right) \right] \\ &\stackrel{e)}{=} 2^{-2n\delta^2/(\log(\lambda_0/\lambda_1))^2} \end{aligned}$$

where a) follows from (4); b) follows since  $\lambda_1^{\rho} \lambda_0^{n-\rho}$ ,  $0 \leq \rho \leq n$  is a decreasing function of  $\rho$ , and, hence, by using (8)

$$\begin{aligned} \lambda_1^{\tau} \lambda_0^{n-\tau} &\leq \lambda_1^{n\lambda_1 + n\delta/(\log(\lambda_0/\lambda_1))} \lambda_0^{n-n\lambda_1 - n\delta/(\log(\lambda_0/\lambda_1))} \\ &= \lambda_1^{n\lambda_1} \lambda_0^{n\lambda_0} \left( \frac{\lambda_1}{\lambda_0} \right)^{n\delta/(\log(\lambda_0/\lambda_1))} \\ &= 2^{-n(S(\rho) + \delta)}. \end{aligned}$$

c) holds for all  $\gamma > 0$ , since

$$2^{\gamma(-\log \Lambda(\chi; \lambda_0, \lambda_1) - n(S(\rho) + \delta))} \geq 1.$$

d) follows from [29, eq. (4.16)], if  $1/2 < \lambda_0 < 1$ ; and e) follows by selecting the minimizing value

$$\gamma = \frac{4\delta}{(\log(\lambda_0/\lambda_1))^2}. \quad \square$$

### B. Projection

Observe that  $|\hat{\psi}_{[n]}\rangle$  and  $|I_{\tau}\rangle$  in (7) are, in general, entangled. Hence, a measurement on the last qubit will irreversibly affect the first  $n$  qubits. Precisely, using von Neumann's postulate [30, Ch. VI], the effect on  $|\hat{\psi}_{[n]}\rangle$  of measuring  $|I_{\tau}\rangle$  is the following:

$$|\hat{\psi}_{[n]}\rangle = \begin{cases} \frac{1}{\sqrt{\sum_{\chi \in \mathcal{G}_{\tau}} |\langle \chi | \psi_{[n]} \rangle|^2}} \sum_{\chi \in \mathcal{G}_{\tau}} \langle \chi | \psi_{[n]} \rangle |\chi\rangle, & \text{if } \{|I_{\tau}\rangle \stackrel{m}{=} |0\rangle\} \\ \frac{1}{\sqrt{\sum_{\chi \in \mathcal{B}_{\tau}} |\langle \chi | \psi_{[n]} \rangle|^2}} \sum_{\chi \in \mathcal{B}_{\tau}} \langle \chi | \psi_{[n]} \rangle |\chi\rangle, & \text{if } \{|I_{\tau}\rangle \stackrel{m}{=} |1\rangle\}. \end{cases} \quad (10)$$

In words, if the event  $\{|I_{\tau}\rangle \stackrel{m}{=} |0\rangle\}$  occurs, then  $|\hat{\psi}_{[n]}\rangle$  will collapse to the renormalized projection of the message  $|\psi_{[n]}\rangle$  onto the subspace spanned by the good eigenstates, otherwise  $|\hat{\psi}_{[n]}\rangle$  will collapse to the renormalized projection of the message  $|\psi_{[n]}\rangle$  onto the subspace spanned by the bad eigenstates. It follows from Theorem III.1 that the event  $\{|I_{\tau}\rangle \stackrel{m}{=} |0\rangle\}$  occurs with very high probability. When this event occurs, we now show that the collapsed state  $|\hat{\psi}_{[n]}\rangle$  is not much different from the original message  $|\psi_{[n]}\rangle$ , that is, the *average fidelity* between the two is close to the maximum possible value of 1. Recall that the average fidelity is the probability that the message  $|\hat{\psi}_{[n]}\rangle$  passes a test for being the same as the original message  $|\psi_{[n]}\rangle$ , when the test is conducted by someone who knows the original message (see Schumacher [11]).

*Corollary III.1:* Suppose that all hypotheses of Theorem III.1 hold, then

$$E \left[ \left| \langle \psi_{[n]} | \hat{\psi}_{[n]} \rangle \right|^2 \left| \{ |I_\tau\rangle^{\underline{m}} | 0 \rangle \} \right. \right] \geq 1 - 2^{-(2n\delta^2 / (\log \lambda_0 / \lambda_1)^2)}. \quad (11)$$

*Proof:*

$$\begin{aligned} & E \left[ \left| \langle \psi_{[n]} | \hat{\psi}_{[n]} \rangle \right|^2 \left| \{ |I_\tau\rangle^{\underline{m}} | 0 \rangle \} \right. \right] \\ &= E \left[ \left| \sum_{\chi \in \{0,1\}^n} \sum_{\xi \in \{0,1\}^n} \langle \chi | \psi_{[n]} \rangle^\dagger \right. \right. \\ &\quad \left. \left. \cdot \langle \xi | \hat{\psi}_{[n]} \rangle \langle \chi | \xi \rangle \right|^2 \left| \{ |I_\tau\rangle^{\underline{m}} | 0 \rangle \} \right. \right] \\ &\stackrel{a)}{=} E \left[ \left| \sum_{\chi \in \{0,1\}^n} \langle \chi | \psi_{[n]} \rangle^\dagger \langle \chi | \hat{\psi}_{[n]} \rangle \right|^2 \left| \{ |I_\tau\rangle^{\underline{m}} | 0 \rangle \} \right. \right] \\ &\stackrel{b)}{=} E \left[ \frac{\sum_{\chi \in \mathcal{G}_\tau} \left| \langle \chi | \psi_{[n]} \rangle \right|^2}{\sqrt{\sum_{\chi \in \mathcal{G}_\tau} \left| \langle \chi | \psi_{[n]} \rangle \right|^2}} \right]^2 \\ &= E \sum_{\chi \in \mathcal{G}_\tau} \left| \langle \chi | \psi_{[n]} \rangle \right|^2 \\ &\stackrel{c)}{=} \sum_{\chi \in \mathcal{G}_\tau} \Lambda(\chi; \lambda_0, \lambda_1) \\ &\stackrel{d)}{=} 1 - \sum_{\chi \in \mathcal{B}_\tau} \Lambda(\chi; \lambda_0, \lambda_1) \\ &\stackrel{e)}{\geq} 1 - 2^{-(2n\delta^2 / (\log \lambda_0 / \lambda_1)^2)} \end{aligned}$$

where a) follows by using the orthonormality of the eigenstates; b) follows from (10); c) follows from (10); d) follows by applying the binomial theorem to  $1 = (\lambda_0 + \lambda_1)^n$ ; and e) follows from Theorem III.1.  $\square$

Together, our Theorem III.1 and Corollary III.1 represent a strengthening of Schumacher's pioneering result in that they hold for fixed block sizes and they deliver a rate of convergence.

#### IV. QUANTUM SHANNON-FANO CODING

##### A. Motivation

We now propose the following scheme for transmitting the quantum message  $\psi_{[n]}$ :

```

compute (7)
measure  $|I_\tau\rangle$ 
if  $\left( \{ |I_\tau\rangle^{\underline{m}} | 0 \rangle \} \right)$  then
    transmit  $|\hat{\psi}_{[n]}\rangle$ 
else
    do nothing.
```

It follows from Theorem III.1 and from Corollary III.1 that the above scheme has high average fidelity with high probability, and only an exponentially small probability of failing to transmit any information. From now on, we assume that the event  $\{ |I_\tau\rangle^{\underline{m}} | 0 \rangle \}$  has occurred, and focus on transmitting  $|\hat{\psi}_{[n]}\rangle$ . It follows from (10) that  $|\hat{\psi}_{[n]}\rangle$  lies in the typical subspace spanned by the good eigenstates. We shall select the truncation threshold in Theorem IV.1 such that the typical subspace

has dimension at most  $2^{n(S(\rho)+\delta)+1}$  which is much less than the original dimension of  $2^n$ . Hence, by appropriately "relabeling" the leading eigenstates, we should be able to represent, and, hence, compress the  $n$  qubit message  $|\hat{\psi}_{[n]}\rangle$  to  $n(S(\rho) + \delta) + 1$  qubits. The main problem, which we now tackle, is how to compute such a *dimensionality-reducing* or *relabeling* transformation efficiently.

##### B. Truncating the Eigenvalues

The eigenvalues  $\lambda_0$  and  $\lambda_1$  are real numbers, and, when represented as fractional binary numbers, may require an infinite precision to represent. Since, in practice, we can only store and manipulate a finite number of bits, from now on, we approximate the eigenvalues using fractional numbers with  $q$  significant bits after the binary point. In particular, we let  $\lambda_0^\diamond$  denote the fractional number obtained by truncating all but the  $q$  most significant bits of  $\lambda_0$ . And we let

$$\lambda_1^\diamond = \lambda_1 + (\lambda_0 - \lambda_0^\diamond).$$

Since,  $\lambda_0 + \lambda_1 = 1$ , it follows that  $\lambda_1^\diamond$  has at most  $q$  nonzero significant bits, and the remaining bits must be zeros. Furthermore, we have that

$$\lambda_0^\diamond + \lambda_1^\diamond = 1.$$

In the remainder of the correspondence, instead of the original eigenvalues, we will use  $\lambda_0^\diamond$  and  $\lambda_1^\diamond$ . To be sure, such an approximation will slightly increase the per-symbol rate needed for compression by

$$D(\lambda_0 | \lambda_0^\diamond) = \lambda_0 \log(\lambda_0 / \lambda_0^\diamond) + \lambda_1 \log(\lambda_1 / \lambda_1^\diamond).$$

The quantity  $D(\cdot | \cdot)$  is known as the *relative entropy* or as the *Kullback-Leibler distance*. This increase in the per-symbol rate can be made as small as desired by selecting a large enough  $q$ . However, we will subsequently demonstrate that the amount of quantum hardware required to implement our encoders and decoders will increase quadratically in  $q$ .

##### C. Dimensionality Reduction

We now introduce a quantum "encoder" transformation that transforms each eigenstate  $|\chi\rangle$ ,  $\chi \in \{0, 1\}^n$ , as follows:

$$|\chi, 0_{nq}\rangle \rightarrow |0_n, C(\chi)\rangle \quad (12)$$

where, for  $a > 0$ ,  $0_a$  represents a string of  $a$  zeros, and

$$C(\chi) = \sum_{\xi \in \{0,1\}^n, \xi \prec \chi} \Lambda(\xi; \lambda_0^\diamond, \lambda_1^\diamond) \quad (13)$$

where  $\Lambda(\xi; \lambda_0^\diamond, \lambda_1^\diamond)$  is obtained from (4) and  $\prec$  denotes some total order on the strings in  $\{0, 1\}^n$ . We will specify a computationally simple-to-implement lexicographical order in Section V-A. Observe that for every eigenstate  $|\chi\rangle$ ,  $C(\chi)$  is a number in the real interval  $[0, 1)$ . Hence, given  $C(\chi)$ , we write  $|C(\chi)\rangle$  using the terminology of Section V-E. Intuitively,  $C(\chi)$  is the sum of the eigenvalues of all eigenstates of length  $n$  that are less than or equal to the  $\chi$  in the total order  $\prec$ . Since  $C(\chi)$  is a monotonically increasing function of the eigenstates arranged in lexicographical order, it is uniquely decodable. In other words, the transformation

$$|0_n, C(\chi)\rangle \rightarrow |\chi, 0_{nq}\rangle \quad (14)$$

exists for every eigenstate  $|\chi\rangle$ ,  $\chi \in \{0, 1\}^n$ . Hence, (12) is reversible, and can be implemented as an unitary transformation.

Each eigenvalue in the sum (13) is a product of  $n$  numbers each of which has a precision of  $q$  bits. Hence, each eigenvalue can be written as a fractional binary number with at most  $nq$  nonzero significant bits. Finally, this implies that, for each eigenstate, the number  $C(\chi)$  has precision no more than  $nq$  bits. In other words, the encoder is a unitary transformation from  $\mathcal{H}_2^{\otimes n}$  to a  $2^n$ -dimensional subspace of  $\mathcal{H}_2^{\otimes nq}$ .

However, since generally  $q > 1$ , this hardly constitutes data compression. We now achieve compression by truncating a large number of nonsignificant bits of  $C(\chi)$ .

For a given *truncation parameter*  $k \geq 0$  and a given eigenstate  $|\chi\rangle$ ,  $\chi \in \{0, 1\}^n$ , we define the truncated encoder transform as

$$|\chi, 0_{nq}\rangle \rightarrow |0_n, C(\chi)_{[k]}\mathbf{1}_{nq-k}\rangle \quad (15)$$

where  $C(\chi)_{[k]}$  denotes the truncation of  $C(\chi)$  to the  $k$  most significant qubits. Observe that only  $k$  qubits on the right-hand side depend upon the eigenstates, and, hence, only these bits need be transmitted. Consequently, the encoder in (15) maps messages of a fixed length  $n$  to codewords of fixed length  $k$ . In other words, the encoder is a unitary transformation from  $\mathcal{H}_2^{\otimes n}$  to a subspace of  $\mathcal{H}_2^{\otimes k}$ .

The decodability of the untruncated map in (12) is immediate from the fact that  $C(\chi)$  is a monotonically increasing function of the eigenstates arranged in the lexicographical order. In contrast, the decodability of the truncated map in (15) is a delicate matter. If  $k < n$ , then the truncated map cannot hope to correctly decode all the eigenstates. However, fortunately, we only need to correctly decode the *good* eigenstates. We now establish that if the threshold parameter  $\tau$  and truncation parameter  $k$  are carefully selected, then inverse of (15) exists for all the good eigenstates.

*Theorem IV.1:* Suppose all hypotheses of Theorem III.1 hold. Set

$$\tau = \left\lceil n \left( \lambda_1 + \frac{\delta}{\log \lambda_0 / \lambda_1} \right) \right\rceil \quad (16)$$

$$k \geq nS(\rho) + nD(\lambda_0 || \lambda_0^\diamond) + n\delta + \log(\lambda_0 / \lambda_1). \quad (17)$$

Then, there exists a decoder such that, for every  $\chi$  in  $\mathcal{G}_\tau$

$$|0_n, C(\chi)_{[k]}\mathbf{1}_{nq-k}\rangle \rightarrow |\chi, 0_{nq}\rangle. \quad (18)$$

*Proof:* Given the encoding  $|C(\chi)_{[k]}\mathbf{1}_{nq-k}\rangle$  of the eigenstate  $|\chi\rangle$ , we define the corresponding *decoded* or *reconstructed* eigenstate as  $|\tilde{\chi}\rangle$ ,  $\tilde{\chi} \in \{0, 1\}^n$ , that satisfies the following two inequalities:

$$C(\tilde{\chi}) \leq C(\chi)_{[k]} + \sum_{i=k+1}^{nq} 2^{-i} \quad (19)$$

$$C(\chi)_{[k]} + \sum_{i=k+1}^{nq} 2^{-i} < C(\tilde{\chi}) + \Lambda(\tilde{\chi}; \lambda_0^\diamond, \lambda_1^\diamond). \quad (20)$$

In general, owing to truncation, the decoded eigenstate  $\tilde{\chi}$  need not equal the original eigenstate  $\chi$ . We now show that for values of  $\tau$  as in (16), for values of  $k$  as in (17), and for all good eigenstates, the inequalities (19) and (20) are satisfied if and only if  $\tilde{\chi} = \chi$ . This will establish the theorem.

Suppose that  $\tilde{\chi} = \chi$ . In this case, the first inequality (19) is trivial, and holds for all  $\chi$  in  $\{0, 1\}^n$ . Now, observe that the second inequality (20) holds if

$$\Lambda(\tilde{\chi}; \lambda_0^\diamond, \lambda_1^\diamond) = \Lambda(\chi; \lambda_0^\diamond, \lambda_1^\diamond) \geq 2^{-k} > \sum_{i=k+1}^{nq} 2^{-i}.$$

It follows from (4) that the second inequality (20) holds if

$$(\lambda_1^\diamond)^{w(\chi)} (\lambda_0^\diamond)^{n-w(\chi)} \geq 2^{-k}.$$

We would like the above inequality to hold for all good eigenstates. Since  $(\lambda_1^\diamond)^\theta (\lambda_0^\diamond)^{n-\theta}$ ,  $0 \leq \theta \leq n$ , is a decreasing function of  $\theta$ , it is sufficient that the above inequality holds for the good eigenstates corresponding to the smallest good eigenvalue. If we select  $\tau$  as in (16), then the smallest eigenvalue is larger than

$$(\lambda_1^\diamond)^\tau (\lambda_0^\diamond)^{n-\tau}.$$

Hence, we require that

$$(\lambda_1^\diamond)^\tau (\lambda_0^\diamond)^{n-\tau} \geq 2^{-k}.$$

Equivalently, we require that

$$(\lambda_1)^\tau (\lambda_0)^{n-\tau} \left( \frac{\lambda_1^\diamond}{\lambda_1} \right)^\tau \left( \frac{\lambda_0^\diamond}{\lambda_0} \right)^{n-\tau} \geq 2^{-k}.$$

It follows from simple algebraic manipulations that the above holds if

$$k \geq nS(\rho) + nD(\lambda_0 || \lambda_0^\diamond) + n\delta + \log(\lambda_0 / \lambda_1).$$

This is exactly the requirement in (17).

We now establish the converse, that is, if  $\chi \neq \tilde{\chi}$ , then either (19) or (20) does not hold. There are two cases: either  $\chi \prec \tilde{\chi}$  or  $\tilde{\chi} \prec \chi$ . In the former case (19) cannot hold, and in the latter case (20) cannot hold.  $\square$

Observe that the desired encoder transform in (15) annihilates the quantum state  $\chi$ . This is necessary since both  $|\chi\rangle$  and  $|C(\chi)_{[k]}\rangle$  contain the same information, and since quantum states cannot be cloned, it is impossible to faithfully transmit weighted superpositions of different  $|\chi\rangle$  without the sender obliterating her knowledge about it in the process of transforming the state into a weighted superposition of  $|C(\chi)_{[k]}\rangle$ .

Observe that the untruncated map in (12) and the truncated map in (15) map one eigenstate to one encoded state. Hence, in the terminology of Section V-D, they can be thought of as unitary transforms that are permutations of the basis states.

#### D. Quantum Parallelism

So far, we have specified the desired encoder (15) and the corresponding decoder in (18) in terms of the eigenstates alone. For the sake of completeness, by using linearity of the encoder and the decoder, we now describe their action on the quantum message of interest:

$$\begin{aligned} |\hat{\psi}_{[n]}, 0_{nq}\rangle &= \sum_{\chi \in \mathcal{G}_\tau} \langle \chi | \hat{\psi}_{[n]} \rangle |\chi, 0_{nq}\rangle \\ &\xrightarrow{\text{encode}} \sum_{\chi \in \mathcal{G}_\tau} \langle \chi | \hat{\psi}_{[n]} \rangle |0_n, C(\chi)_{[k]}\mathbf{1}_{nq-k}\rangle \\ &\xrightarrow{\text{transmit}} \sum_{\chi \in \mathcal{G}_\tau} \langle \chi | \hat{\psi}_{[n]} \rangle |C(\chi)_{[k]}\rangle \\ &\xrightarrow{\text{prepare}} \sum_{\chi \in \mathcal{G}_\tau} \langle \chi | \hat{\psi}_{[n]} \rangle |0_n, C(\chi)_{[k]}\mathbf{1}_{nq-k}\rangle \\ &\xrightarrow{\text{decode}} \sum_{\chi \in \mathcal{G}_\tau} \langle \chi | \hat{\psi}_{[n]} \rangle |\chi, 0_{nq}\rangle \\ &= |\hat{\psi}_{[n]}, 0_{nq}\rangle \end{aligned}$$

where we have implicitly used the fact that a measurement on the qubit  $|I_\tau\rangle$  has been made, and the event  $\{|I_\tau\rangle^m = |0\rangle\}$  has occurred.

#### V. REVERSIBLE ARITHMETIC CODING

We now propose quantum algorithms and associated quantum circuits to efficiently realize the encoder in (15) and the corresponding decoder in (18).

##### A. Arithmetic Recursions

First, we consider the computation of the function  $C(\chi)$  in (13). A straightforward algorithm for computing  $C(\chi)$  by explicitly performing the summation would require an exponential amount of complexity in the block size  $n$ . One of the main contributions of classical arithmetic coding is to observe that if we select the total order  $\prec$  in (13) to be the following lexicographical order, then the function  $C(\chi)$  can

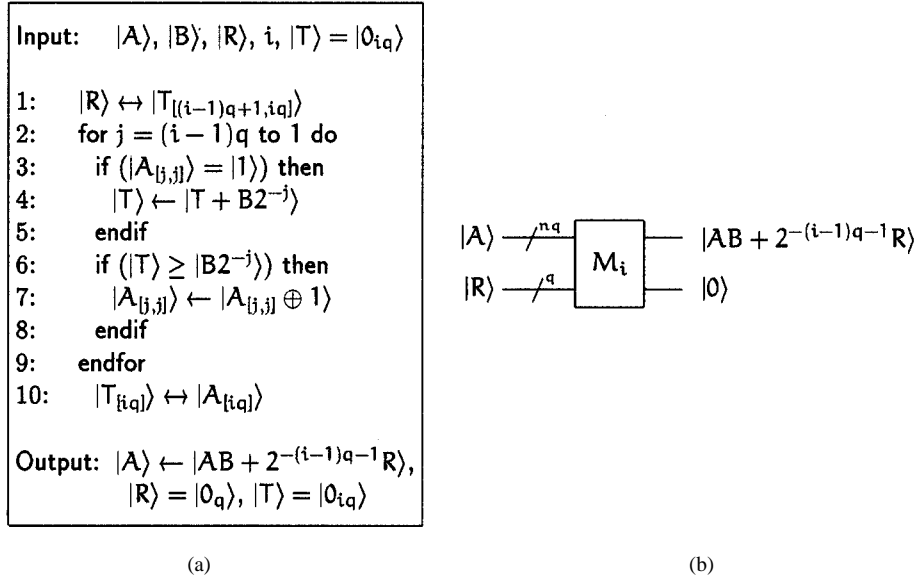


Fig. 4. (a) Quantum algorithm **multiply** ( $|A\rangle, |B\rangle, |R\rangle, i$ ). Note that  $i$  is a classical parameter. The algorithm works as desired whenever  $R < B$  and  $B \neq 0$ . (b) Schematic circuit symbol for the algorithm.

be efficiently computed. If  $\xi \equiv \xi_1 \xi_2 \cdots \xi_n$  and  $\chi \equiv \chi_1 \chi_2 \cdots \chi_n$  are in  $\{0, 1\}^n$ , then we say that

$$\xi \prec \chi \quad \text{if and only if} \quad \sum_{i=1}^n \xi_i 2^{i-1} < \sum_{i=1}^n \chi_i 2^{i-1}. \quad (21)$$

Under this definition of the total order  $\prec$ , we can write the function  $C(\chi)$  recursively as follows (see, [22, eq. (1)]):

```

C(χ) = 0
for i = 1 to n do
  if (χi = 0)
    C(χ) = C(χ) × λ0◊
  else
    C(χ) = C(χ) × λ1◊ + λ0◊
  endif
endfor.
    
```

*Remark V.1:* Instead of the lexicographical order in (21), we can also use the following *dual* order. If  $\xi \equiv \xi_1 \xi_2 \cdots \xi_n$  and  $\chi \equiv \chi_1 \chi_2 \cdots \chi_n$  are in  $\{0, 1\}^n$ , then we say that

$$\xi \overset{\text{dual}}{\prec} \chi \quad \text{if and only if} \quad \sum_{i=1}^n \xi_i 2^{-i+1} < \sum_{i=1}^n \chi_i 2^{-i+1}. \quad (22)$$

Under this dual definition, we can also write the function  $C(\chi)$  recursively, see, [22, eq. (2)]. Although both the recursions are amenable to a quantum implementation, the recursion corresponding to the total order in (21) turns out to slightly simpler and, hence, is used in this correspondence.

### B. Quantum Algorithms for Division and Multiplication

Important parts of the encoding and decoding algorithms are multiplication and division, respectively, and in order to build the quantum coders, we must first construct quantum algorithms for such arithmetic. Suitable addition and subtraction circuits have already been described

in the literature [31], [14], but appropriate multiplication and division algorithms have not been. These are described below.

We present in Fig. 4 an algorithm **multiply** ( $|A\rangle, |B\rangle, |R\rangle, i$ ) that takes the following inputs: a) a fixed index  $i, i = 1, 2, \dots, n$ , b)  $nq$  qubit register  $|A\rangle$  such that all but the first  $(i-1)q$  qubits are zeros, c)  $q$  qubit register  $|B\rangle$ , and d)  $q$  qubit register  $|R\rangle$ . The algorithm also requires an  $nq$  qubit temporary register  $|T\rangle$  that is initialized and finalized to  $|0_{nq}\rangle$ . The algorithm computes

$$|A, R\rangle \rightarrow |AB + 2^{-(i-1)q-1}R, 0_q\rangle$$

where multiplications and additions are to be interpreted by treating  $A, B$ , and  $R$  as fractional binary numbers. A quantum circuit which implements the algorithm is shown in Fig. 5.

We term the conjugate inverse of this algorithm as **divide** ( $|A\rangle, |B\rangle, |R\rangle, i$ ). Given an  $nq$  qubit register  $|A\rangle$  such that all but the first  $iq$  qubits are zeros, a  $q$  qubit register  $|B\rangle$ , and a  $q$  qubit register  $|R\rangle$  that is initialized to  $|0_q\rangle$ , the circuit **divide** ( $|A\rangle, |B\rangle, |R\rangle, i$ ) uses an  $nq$  qubit temporary register  $|T\rangle$  that is initialized and finalized to  $|0_{nq}\rangle$  and divides  $A$  by  $B$  up to the first  $(i-1)q$  bits, and stores the quotient also in  $A$ , and keeps the  $q$  qubit remainder in  $R$ .

For the quantum multiplication (or division) to be reversible we need that the multiplier (or divisor)  $|B\rangle$  be nonzero, that is,  $\langle B|0_q\rangle = 0$ . In the sequel,  $|B\rangle$  is never zero.

Observe that the algorithm for multiplication works as desired when the remainder is zero, that is,  $|R\rangle = |0_q\rangle$ . It was pointed out by a reviewer that if  $R \geq B$  along a certain coherent path of computation, the algorithm does not carry out the desired computation. The reasoning behind this claim is as follows. Along a certain path, suppose that  $R \geq B$  and that  $A_{[(i-1)q, (i-1)q]} = 0$ , then Step 4 in Fig. 4 will not be executed. However, since  $T = 2^{-(i-1)q}R \geq 2^{-(i-1)q}B$ , the inequality in Step 6 will still test positive, and, hence, the algorithm will try to erroneously erase  $A_{[(i-1)q, (i-1)q]}$ . However, in this correspondence, the remainder fed to the multiplication algorithm is either zero or arises as an output from a corresponding division algorithm that guarantees that along all coherent paths of computation  $R < B$ . We now establish this fact. Note that the division algorithm is the conjugate inverse of the multiplication algorithm, and is obtained by running the algorithm in Fig. 4 in reverse. The division algorithm

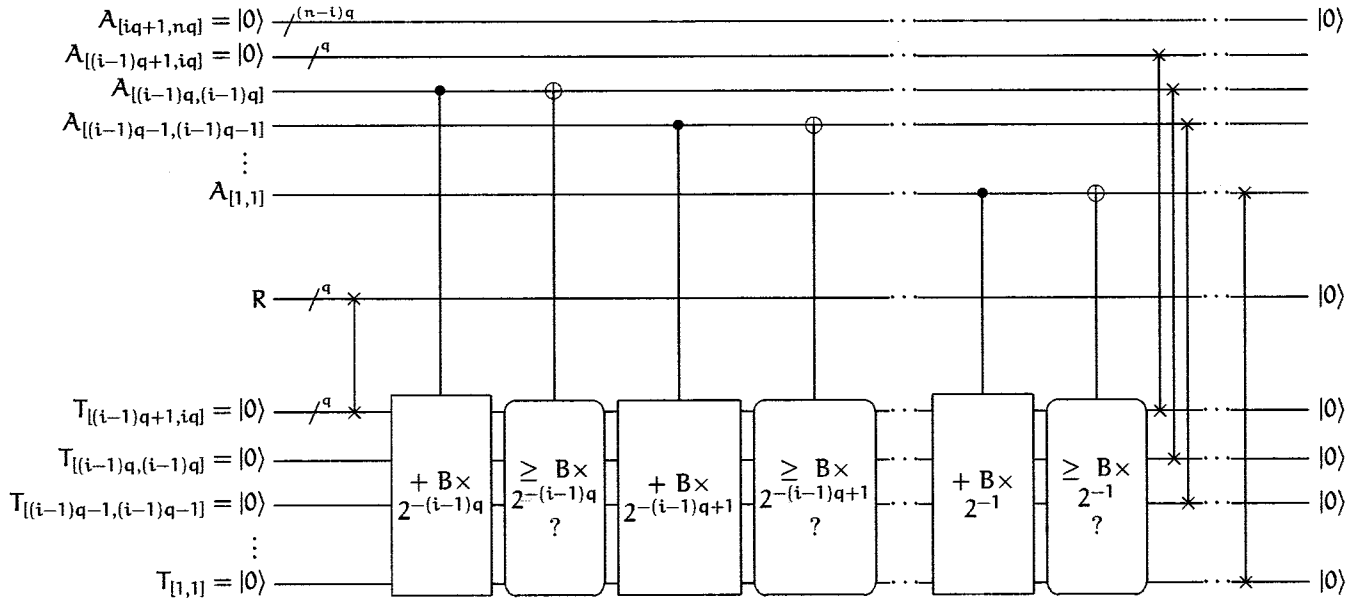
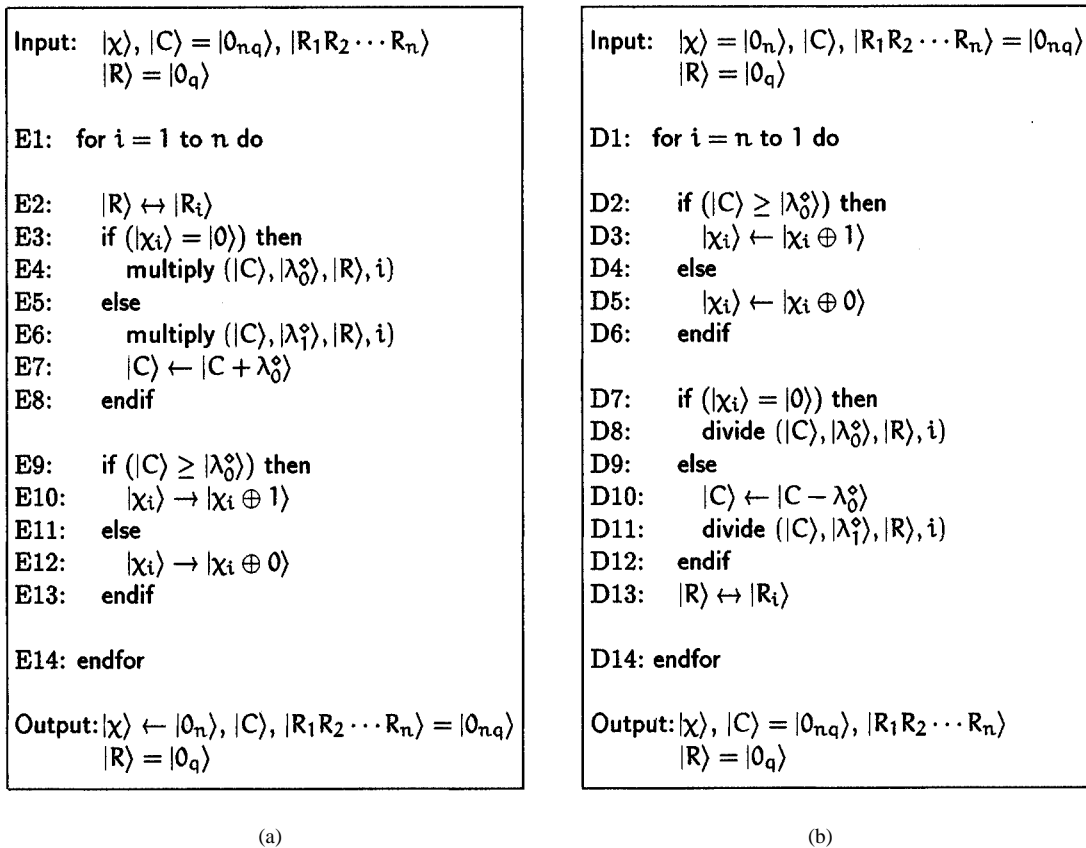


Fig. 5. Quantum circuit implementing the quantum multiplication algorithm of Fig. 4.



(a)

(b)

Fig. 6. Algorithms: (a) "E" and (b) "D."

sequentially erases bits of  $T$ . Now, suppose that we are at the very last step of the division algorithm corresponding to  $j = (i-1)q$ . After this step is completed, we are guaranteed that  $T < B2^{-(i-1)q}$  and that  $R = T2^{(i-1)q} < B$ .

### C. Building Blocks

We now use the ideas from arithmetic recursions, and the above circuits for multiplication and division to construct building blocks for the desired encoder in (15).

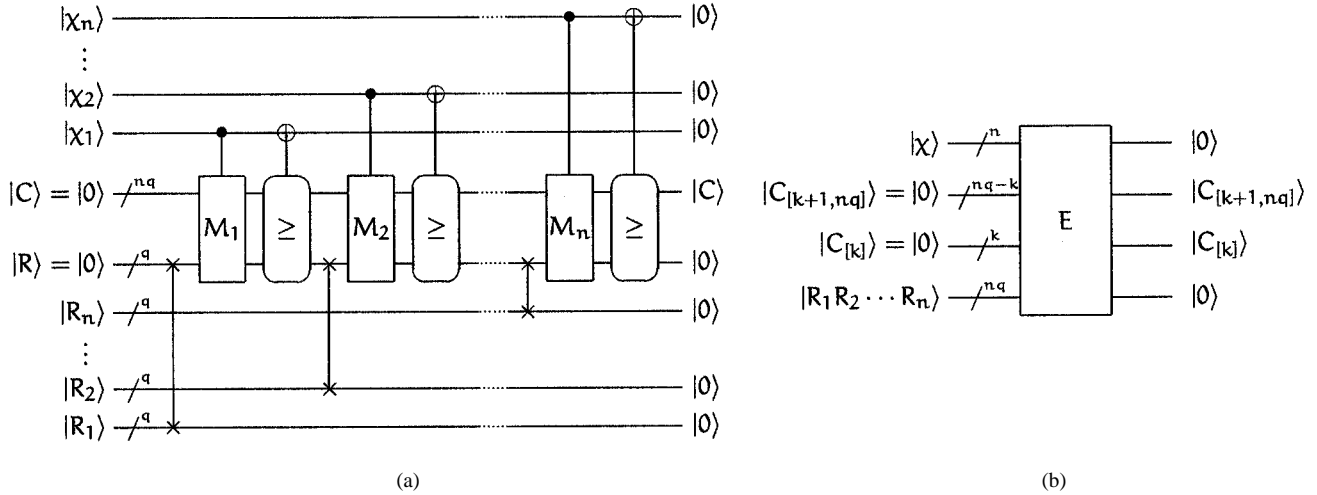


Fig. 7. (a) Quantum circuit implementing the block encoder algorithm  $E$  in Fig. 6 and (b) schematic symbol for the circuit.

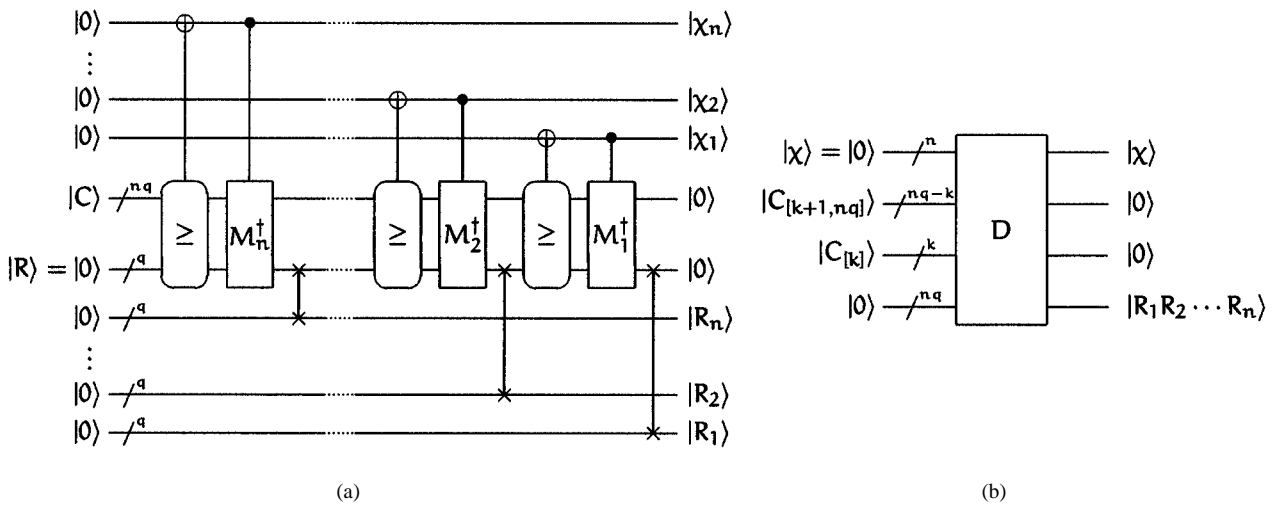


Fig. 8. (a) Quantum circuit implementing the block decoder algorithm  $D$  in Fig. 6 and (b) schematic symbol for the circuit.

In Fig. 6, we present two recursive algorithms “ $E$ ” and “ $D$ .” Formally and literally, these algorithms are inverses of each other: lines E2–E8 are literal inverses of lines D7–D13, lines E9–E13 are literal inverses of lines D2–D6, and, finally, the **for** loop in the algorithm  $E$  processes the message symbols in the original order from 1 to  $n$  while the **for** loop in the algorithm  $D$  emits the message symbols in the inverse order from  $n$  to 1. We exhibit quantum circuits for implementing the algorithms  $E$  and  $D$  in Figs. 7 and 8, respectively. Observe that these circuits are also quantum-mechanical inverses of each other.

We intend to use the algorithms  $E$  and  $D$  with two different sets of inputs. We now explain the functionality of these algorithms on the first set of inputs.

*Lemma V.1:* Let  $|\chi\rangle$ ,  $\chi \in \{0, 1\}^n$ , denote any eigenstate. The algorithms  $D$  and  $E$ , respectively, compute the following maps:

$$D_1: |0_n, C(\chi), 0_{nq}\rangle \rightarrow |\chi, 0_{nq}, 0_{nq}\rangle \quad (23)$$

$$E_1: |\chi, 0_{nq}, 0_{nq}\rangle \rightarrow |0_n, C(\chi), 0_{nq}\rangle. \quad (24)$$

*Proof:* With the inputs as above,  $E_1$  is a quantum version of the arithmetic recursion presented in Section V-A. The desired assertion for  $D_1$  follows by observing that it is a literal inverse of  $E_1$ . In both

of these cases, the quantum register  $|R_1 R_2 \cdots R_n\rangle$  always remains in the same initial state  $|0_{nq}\rangle$ .  $\square$

Lemma V.1 furnishes a way of implementing (12) and its inverse (14). Recall, however, that to achieve compression we are interested in implementing (15). The obvious strategy of first implementing (12) and simply transmitting the  $k$  most significant qubits of  $|C(\chi)\rangle$  does not work, since these  $k$  qubits are entangled with the  $nq - k$  least significant qubits of  $|C(\chi)\rangle$ . Hence, a measurement on these  $nq - k$  least significant qubits will irreversibly change the  $k$  most significant qubits. To avoid such an accident, we must *erase* the  $nq - k$  qubits. This is the central difficulty that we must overcome. We now explain the functionality of the algorithms  $E$  and  $D$  on the second set of inputs.

*Lemma V.2:* Suppose that all hypotheses of Theorem IV.1 hold. Furthermore, suppose that a measurement on the qubit  $|I_\tau\rangle$  has been made, and the event  $\{|I_\tau\rangle \stackrel{m}{=} |0\rangle\}$  has occurred. Let  $|\chi\rangle$ ,  $\chi \in \mathcal{G}_\tau$ , denote any good eigenstate. The algorithms  $D$  and  $E$ , respectively, compute the following maps:

$$D_2: |0_n, C(\chi)_{[k]} 1_{nq-k}, 0_{nq}\rangle \rightarrow |\chi, 0_{nq}, R_1 R_2 \cdots R_n\rangle \quad (25)$$

$$E_2: |\chi, 0_{nq}, R_1 R_2 \cdots R_n\rangle \rightarrow |0_n, C(\chi)_{[k]} 1_{nq-k}, 0_{nq}\rangle. \quad (26)$$

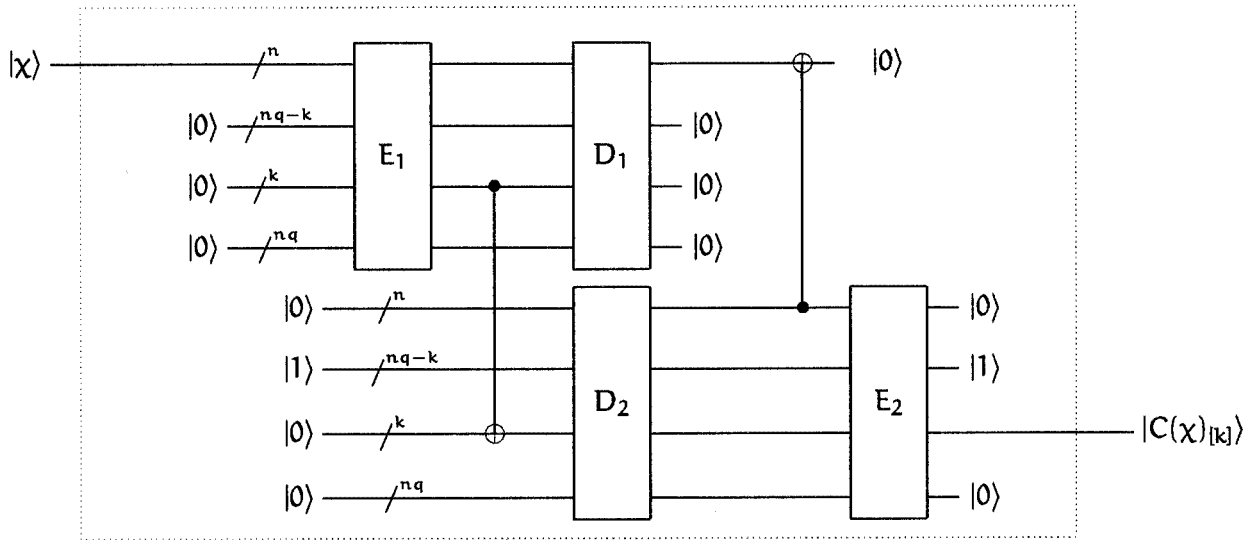


Fig. 9. Quantum circuit implementing the Shannon–Fano encoder in (15). The corresponding decoder is obtained by running the same circuit in reverse.

*Proof:* We establish the assertion for  $D_2$  in detail. The desired assertion for  $E_2$  follows by observing that it is a literal inverse of  $D_2$ . Fix a good eigenstate  $|\chi\rangle = |\chi_1\chi_2\cdots\chi_n\rangle$ . Theorem IV.1 showed that  $|\chi\rangle$  can be decoded correctly; the gist of what follows is that *not only can  $|\chi\rangle$  be decoded correctly—in fact, it can be decoded correctly in a sequential or recursive fashion.* For an index  $i, i = 1, 2, \dots, n$ , recall that  $|\chi_{[i]}\rangle \equiv |\chi_1\chi_2\cdots\chi_i\rangle$  and let  $|C^{(i)}(\chi)\rangle$  denote the content of the register  $|C\rangle$  before the  $i$ th iteration of the “for” loop in D1–D14 begins. Also, note that we are dealing with one physical path for a fixed eigenstate  $|\chi\rangle$ , and, hence, it suffices to treat the contents of the quantum register  $|C^{(i)}(\chi)\rangle$  as real numbers. Since all operations are implemented using reversible quantum gates, the decoder also functions as desired for linear superpositions.

Step 1: We first show that lines D7–D13 behave as desired. For any  $i = n, n-1, \dots, 2$  and for real number  $C^{(i)}(\chi)$ , if

$$C(\chi_{[i]}) \leq C^{(i)}(\chi) < C(\chi_{[i]}) + \Lambda(\chi_{[i]}; \lambda_0^\circ, \lambda_1^\circ) \quad (27)$$

then, after the computation in lines D7–D13, we have

$$C(\chi_{[i-1]}) \leq C^{(i-1)}(\chi) < C(\chi_{[i-1]}) + \Lambda(\chi_{[i-1]}; \lambda_0^\circ, \lambda_1^\circ).$$

However, we have from (19) and (20) that, if we set

$$C^{(n)}(\chi) = C(\chi)_{[k]} + \sum_{i=k+1}^{nq} 2^{-i}$$

then the inequality (27) holds for  $i = n$ . Hence, by induction, the inequality (27) holds for all  $i = n, n-1, \dots, 1$ .

Step 2: We now show that lines D2–D6 behave as desired. For any  $i = n, n-1, \dots, 1$  and for real number  $C^{(i)}(\chi)$ , if the inequality (27) holds, then

$$C^{(i)}(\chi) \geq \lambda_0^\circ \quad \text{if and only if} \quad C(\chi_{[i]}) \geq \lambda_0^\circ. \quad (28)$$

The “if” part of (28) follows trivially from (27). To see the “only if” part, observe that if  $C^{(i)}(\chi) \geq \lambda_0^\circ$ , then, by (27)

$$C(\chi_{[i]}) + \Lambda(\chi_{[i]}; \lambda_0^\circ, \lambda_1^\circ) > \lambda_0^\circ.$$

Hence

$$C(\chi_{[i]}) > \lambda_0^\circ - \Lambda(\chi_{[i]}; \lambda_0^\circ, \lambda_1^\circ).$$

Hence, again by (27),

$$C^{(i)}(\chi) \geq C(\chi_{[i]}) > \lambda_0^\circ - \Lambda(\chi_{[i]}; \lambda_0^\circ, \lambda_1^\circ).$$

By definition (13), the only allowed values of  $C(\chi_{[i]})$  that satisfy the above inequality are  $C(\chi_{[i]}) \geq \lambda_0^\circ$  as desired. The essence of this step is to show that instead of the true quantity  $C(\chi_{[i]})$  a slightly perturbed quantity  $C^{(i)}(\chi)$  is sufficient for making the correct decision in the “if” loop in D2–D6.  $\square$

Observe that (25) is almost the desired decoder (18) except the “remainder”  $|R_1R_2\cdots R_n\rangle$  that is left over. Once again, the decoded state  $|\chi\rangle$  is entangled with this remainder, and, hence, the remainder must be erased. Similarly, (26) is almost the desired encoder (15) except that it requires the above left over remainder as an input.

#### D. Putting the Puzzle Together

It follows from the above discussion that the algorithms described by Lemmas V.1 and V.2 do not, in themselves, yield either the desired encoder (15) or the decoder (18). We now present an algorithm, in Fig. 9, that uses all the four pieces in these lemmas to construct the desired encoder. The desired decoder is obtained by literally running the encoder in reverse.

We now briefly explain our construction. The circuit is started by applying the transformation  $E_1$  in Lemma V.1

$$E_1: |\chi, 0_{nq}, 0_{nq}\rangle \rightarrow |0_n, C(\chi), 0_{nq}\rangle.$$

After the  $k$  most significant qubits of  $|C(\chi)\rangle$  are copied (of course, they are not truly copied in the classical sense, since qubits cannot be cloned; they are entangled with an auxiliary set of qubits prepared in the  $|0\rangle$  state), the output of  $E_1$  is acted upon by the transformation  $D_1$  in Lemma V.1

$$D_1: |0_n, C(\chi), 0_{nq}\rangle \rightarrow |\chi, 0_{nq}, 0_{nq}\rangle.$$

This has the effect of annihilating all the  $nq$  qubits of  $|C(\chi)\rangle$ . However, it recreates the input quantum state  $|\chi\rangle$  which must also be erased. Now, by employing the  $k$  copied qubits  $|C(\chi)_{[k]}\rangle$ , we can apply  $D_2$  in Lemma V.2

$$D_2: |0_n, C(\chi)_{[k]}^{1_{nq-k}}, 0_{nq}\rangle \rightarrow |\chi, 0_{nq}, R_1 R_2 \cdots R_n\rangle.$$

The quantum state  $|\chi\rangle$  produced at the output of  $D_2$  is used to erase the same quantum state produced at the output of  $D_1$ . Now, by applying  $E_2$  in Lemma V.2 to the output produced by  $D_2$ , we have the desired output

$$E_2: |\chi, 0_{nq}, R_1 R_2 \cdots R_n\rangle \rightarrow |0_n, C(\chi)_{[k]}^{1_{nq-k}}, 0_{nq}\rangle.$$

In the end, we are guaranteed that no quantum register inside the “dotted rectangle” in Fig. 9 is entangled with the final output  $|C(\chi)_{[k]}\rangle$ . Hence, the output can now be freely transmitted. Observe that the cascade of  $E_1$  and  $D_1$  is the identity map, and, similarly, the cascade of  $D_2$  and  $E_2$  is also the identity map.

This four-gate construction was inspired by Bennett’s *pebbling* procedure for reversible classical computation [23], but applied for a different purpose for which it serves surprisingly well.

#### E. A Complexity Analysis

We now analyze the hardware complexity of implementing the  $E_1$  block in Fig. 9. The  $E_1$  block can be implemented using the circuit presented in Fig. 7. The “ $\geq$ ” operator compares a  $nq$  qubit register  $C$  to a  $q$  bit constant. Using the TEST-GREATER-THAN circuits [14], such comparisons can be implemented quantum-mechanically in  $O(nq)$  elementary quantum gates. We have used a “swap” or  $\leftrightarrow$  operator in circuits for **multiply and divide**. A quantum-mechanical operator that swaps two quantum registers of length  $q$  can be implemented using  $O(q)$  quantum Fredkin gates [32]–[34]. For the index  $i$ ,  $1 \leq i \leq n$ , the overall circuit for  $M_i$  can be implemented in  $O(i^2 q^2)$  elementary quantum gates. In conclusion, the overall circuit for the  $E_1$  block can be implemented using  $O(n^3 q^2)$  elementary quantum gates. The blocks  $D_1$ ,  $E_2$ , and  $D_2$  have the same complexity as the block  $E_1$ . Hence, the overall encoder in Fig. 9 can also be implemented using  $O(n^3 q^2)$  elementary quantum gates. Also, using similar reasoning, it follows that the overall encoder in Fig. 9 has a  $O(n^3 q^2)$  computational complexity.

## VI. CONCLUSION

We have constructed a quantum algorithm for block compression of quantum information which is an analog of classical arithmetic coding. In contrast to the classical case, the quantum algorithm must take extra care to leave behind no residual traces of its past history. The algorithm thus begins by projecting the state into the typical subspace, then a sequence of encoding and decoding using finite precision arithmetic is done in a manner so as to obliterate all possible imprecisions.

Unlike the classical algorithms for arithmetic coding, the multiplication steps used in our algorithm require a linearly increasing precision in the block size  $n$ . In the classical case, it is known how to implement these multiplications using precision that is independent of  $n$  [19], [22]. Although we believe that a similar result holds for the quantum multiplier, since one has *a priori* knowledge that the multiplication will not change an increasingly large number of bits in  $C(\chi)$  as  $i$  increases in the encoder, such quantum extensions are currently an open problem.

We also believe it is straightforward to perform this algorithm in parallel, so as to reduce the number of time-steps necessary for its circuit implementation. Multiplication and addition are known to be in  $NC(1)$ , and believed to also be in the quantum counterpart to this class,

so that it should be possible to obtain an  $O(n)$  running time implementation of our algorithm.

Quantum circuits such as the one we presented may also find use as reversible classical circuits, which potentially require much less power for their execution when using technologies such as reversible CMOS or charge recovery logic [35].

Future work may also extend the explicit examples given here from quantum memoryless Bernoulli sources to more complex source distributions. Finally, in this correspondence, we have considered block arithmetic codes. Classically, arithmetic codes have also been used in an online or a sequential fashion. Such a step would be a natural generalization from our results, but feasibility of truly online quantum codes is currently an open problem. We suspect that such extensions are ruled out in principle, if the receiver can use timing to obtain “which path” information about the transmitted state.

## REFERENCES

- [1] P. W. Shor, “Algorithms for quantum computation: Discrete logarithms and factoring,” in *Proc. 35th Annu. Symp. Foundations of Computer Science, Santa Fe, NM*. Los Alamitos, CA: IEEE Comp. Soc. Press, 1994, pp. 124–134.
- [2] L. K. Grover, “A fast quantum mechanical algorithm for database search,” in *Proc. 28th Annu. ACM Symp. Theory of Computation*, Philadelphia, PA, 1996, pp. 212–219.
- [3] C. H. Bennett, G. Brassard, and A. K. Ekert, “Quantum cryptography,” *Sci. Amer.*, vol. 267, pp. 50–57, Oct. 1992.
- [4] C. Fuchs, “Nonorthogonal quantum states maximize classical information capacity,” *Phys. Rev. Lett.*, vol. 79, pp. 1162–1165, 1997.
- [5] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane, “Quantum error correction via codes over GF(4),” *IEEE Trans. Inform. Theory*, vol. 44, pp. 1369–1387, July 1998.
- [6] C. H. Bennett and P. W. Shor, “Quantum information theory,” *IEEE Trans. Inform. Theory (Commemorative Issue, 1948–1998)*, vol. 44, pp. 2724–2742, Oct. 1998.
- [7] E. Rieffel and W. Polak. (1998) An introduction to quantum computing for nonphysicists. [Online] <http://xxx.lanl.gov/abs/quant-ph/9809016>.
- [8] A. Steane, “Quantum computing,” *Reps. Progr. Phys.*, vol. 61, pp. 117–173, 1998.
- [9] C. E. Shannon, “A mathematical theory of communication,” *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, 1948.
- [10] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.
- [11] B. Schumacher, “Quantum coding,” *Phys. Rev. A*, vol. 51, pp. 2738–2747, 1995.
- [12] W. K. Wootters and W. H. Zurek, “A single quantum cannot be cloned,” *Nature*, vol. 299, pp. 802–803, 1982.
- [13] D. Dieks, “Communication by EPR devices,” *Phys. Lett. A*, vol. 92, no. 6, pp. 271–272, 1982.
- [14] R. Cleve and D. P. DiVincenzo, “Schumacher’s quantum data compression as a quantum computation,” *Phys. Rev. A*, vol. 54, pp. 2636–2650, Oct. 1996.
- [15] T. M. Cover, “Enumerative source encoding,” *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 73–77, Jan. 1973.
- [16] J. P. M. Schalkwijk, “An algorithm for source coding,” *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 395–399, May 1972.
- [17] S. L. Braunstein, C. A. Fuchs, D. Gottesman, and H.-K. Lo, “A quantum analog of Huffman coding,” in *Proc. 1998 IEEE Int. Symp. Information Theory*. Cambridge, MA, Aug. 16–21, 1998, [Online] <http://xxx.lanl.gov/abs/quant-ph/9805080>, p. 353.
- [18] R. C. Pasco, “Source coding algorithms for fast data compression,” Ph.D. dissertation, Dep. Elec. Eng., Stanford Univ., Stanford, CA, 1976.
- [19] J. Rissanen, “Generalized Kraft inequality and arithmetic coding,” *IBM J. Res. Devel.*, vol. 20, no. 3, pp. 198–203, 1976.
- [20] T. C. Bell, J. G. Cleary, and I. H. Witten, *Text Compression*. Englewood Cliffs, NJ: Prentice-Hall, 1990.
- [21] G. G. Langdon Jr., “An introduction to arithmetic coding,” *IBM J. Res. Devel.*, vol. 28, no. 2, pp. 135–149, 1984.
- [22] J. Rissanen and G. G. Langdon Jr., “Arithmetic coding,” *IBM J. Res. Devel.*, vol. 23, no. 2, pp. 149–162, 1979.
- [23] C. H. Bennett, “Logical reversibility of computation,” *IBM J. Res. Devel.*, vol. 17, pp. 525–532, 1973.
- [24] —, “Time/space trade-offs for reversible computation,” *SIAM J. Comput.*, vol. 18, pp. 766–776, 1989.

- [25] T. Toffoli, "Reversible computing," in *Automata, Languages, and Programming*, W. de Bakker and J. van Leeuwen, Eds. New York: Springer, 1980, pp. 632–644.
- [26] D. Deutsch, "Quantum theory, the Church–Turing principle and the universal quantum computer," in *Proc. Roy. Soc. London A*, vol. 400, 1985, pp. 97–117.
- [27] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. W. Shor, T. Sleator, J. Smolin, and H. Weinfurter, "Elementary gates for quantum computation," *Phys. Rev. A*, vol. 52, no. 5, pp. 3457–3467, 1995.
- [28] A. Barenco, "A universal two-bit gate for quantum computation," in *Proc. Roy. Soc. London A*, vol. 449, 1995, pp. 679–683.
- [29] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *Amer. Statist. Assoc. J.*, vol. 58, pp. 13–30, 1963.
- [30] J. von Neumann, *Mathematical Foundations of Quantum Mechanics*. Princeton, NJ, USA: Princeton Univ. Press, 1955.
- [31] D. Beckman, A. N. Chari, S. Devabhaktuni, and J. Preskill, "Efficient networks for quantum factoring," *Phys. Rev. A*, vol. 54, no. 2, pp. 1034–1063, 1996.
- [32] E. Fredkin and T. Toffoli, "Conservative logic," *Int. J. Theor. Phys.*, vol. 21, no. 3/4, pp. 219–253, 1982.
- [33] H. F. Chau and F. Wilczek, "Simple realization of the Fredkin gate using a series of two-body operators," *Phys. Rev. Lett.*, vol. 75, no. 4, pp. 748–750, 1995.
- [34] I. L. Chuang and Y. Yamamoto, "Simple quantum computer," *Phys. Rev. A*, vol. 52, pp. 3489–3496, 1995.
- [35] S. Younis and T. Knight, "Non dissipative rail drivers for adiabatic circuits," in *Proc. 16th Conf. Advanced Research in VLSI 1995*. Los Alamitos, CA: IEEE Comput. Soc. Press, 1995, pp. 404–414.

## On the AEP of Word-Valued Sources

Mikihiko Nishiara, *Student Member, IEEE*, and  
Hiroyoshi Morita, *Member, IEEE*

**Abstract**—We consider a new class of information sources called word-valued sources in order to investigate coding algorithms based upon string parsing. A word-valued source is defined as a pair of an independent and identically distributed (i.i.d.) source with a countable alphabet and a function that maps each symbol into a finite sequence over a finite alphabet. A word-valued source is a nonstationary process and has countable states. If the function of a word-valued source is prefix-free, the entropy rate is characterized with a simple expression and the AEP (Asymptotic Equipartition Property) holds.

**Index Terms**—AEP, entropy rate, information source class, lossless data compression, string parsing.

### I. INTRODUCTION

A word-valued source is defined as a pair of an independent and identically distributed (i.i.d.) source with a countable alphabet and a function that maps each symbol into a word over a finite alphabet, where a *word* is defined as a finite sequence. Since the output sequence of words emitted out of a word-valued source is sequentially observed symbol by symbol, any two consecutive words cannot be uniquely separated from each other without additional conditions on the function, say the prefix-free property, such that no words are prefixes of others. In gen-

eral, a word-valued source is a nonstationary process with countable states.

A compressed sequence emitted from a fixed-to-variable length encoder for a stationary source can be modeled as a word-valued source, in the sense that the original source can be approximated by sufficiently large blockwise-independent source. Similarly, a compressed sequence of a variable-to-fixed length encoder with prefix-free parsing, or more generally, of a variable-to-variable length encoder with prefix-free parsing can be thought of as a word-valued source as well.

Moreover, a word-valued source is interpreted as a source statistics estimator used in a variety of source-coding algorithms based upon a string parsing approach [1], [2]. A string parsing algorithm consists of the following three procedures.

- 1) Divide an individual source sequence into words.
- 2) Estimate an i.i.d. source fitted to the occurrence of the words. As the probability distribution, the relative frequencies are usually used.
- 3) Encode the sequence of the words under the estimated static source model by means of a source-coding technique like arithmetic coding.

Since the set of words produced by parsing the source sequence is countable in general, the i.i.d. source, which governs the occurrence of the words, has a countable alphabet. Moreover, the relation between this countable alphabet and the words is a function. Hence, we can consider the pair of the i.i.d. source and this function as a statistical model to estimate the probability of the individual source sequence.

One of the coding algorithms based upon string parsing is LZ78, whose parsing method is called incremental parsing. Actually, LZ78 does not apply the static coding described above but an adaptive coding. That is, the three procedures are to be done simultaneously as described in [3].

Recently, Kieffer and Yang proposed an idea for constructing a kind of parsing rule for individual source sequences [4]. Its purpose is to find a context-free grammar that generates only the given sequence. The production rule for the start symbol is considered as a parsing for the individual sequence, since each nonterminal symbol represents a word. Thus this grammatical approach is also considered as a word-valued source if we note that the description length for production rules characterizes the probability of the represented words.

In this correspondence, we investigate the stochastic behavior of word-valued sources and show that if the function of a word-valued source is prefix-free, the entropy rate of the source is characterized by a simple expression and the source has the AEP (Asymptotic Equipartition Property).

In Section II, we define a word-valued source and a recurrent source. We then show that a word-valued source is equivalent to a countable-state source. In Section III, we state two theorems on the entropy rate and the AEP of a word-valued source. In Section IV, we prove the theorems. In Section V, we discuss the probability distribution of compressed sequences and the class of word-valued sources.

### II. WORD-VALUED SOURCES

Let  $\mathbf{X} = X_1 X_2 \cdots$  be an i.i.d. source with a countable alphabet  $\mathcal{X}$ . The distribution of  $\mathbf{X}$  will be denoted by  $P$ . Let  $\mathcal{Y}$  be a finite alphabet and  $\mathcal{Y}^*$  the set of all finite sequences over  $\mathcal{Y}$ , including the null sequence  $\lambda$  of length 0. Let us consider a mapping  $\varphi: \mathcal{X} \rightarrow \mathcal{Y}^*$ . Concatenating the sequences  $\varphi(X_1), \varphi(X_2), \cdots$  for the source symbols  $X_1, X_2, \cdots$  produces a sequence  $\varphi(X_1)\varphi(X_2)\cdots$ , which is denoted by  $\varphi(\mathbf{X})$ . A source  $\varphi(\mathbf{X})$  induced by a pair of  $\mathbf{X}$  and  $\varphi$  is called a

Manuscript received October 21, 1998; revised April 11, 1999.  
The authors are with the Graduate School of Information Systems, University of Electro-Communications, 1-5-1 Chofugaoka, Chofu, Tokyo 182-8585, Japan.  
Communicated by N. Merhav, Associate Editor for Source Coding.  
Publisher Item Identifier S 0018-9448(00)03100-X.