Multicasting in Quantum Switching Networks

Manish Kumar Shukla and A. Yavuz Oruç

Abstract—In this paper, we present a quantum multicasting network, called a *generalized quantum connector* (n-GQC), which can be used to multicast quantum information from n inputs to n outputs. This network is recursively constructed using $n/2$-GQCs and consists of $O(n \log^2 n)$ quantum gates. The key component of the n-GQC is another network, called an n-quantum concentrator (n-QC). This concentrator is also an $n \times n$ quantum network, and can route arbitrary quantum states on any m of its inputs to its top m outputs, for any $m, 1 \leq m \leq n$. Its quantum gate complexity is $O(n \log n)$. The quantum gate-level depths of n-QC and n-GQC are $O(\log^2 n)$ and $O(\log^3 n)$, respectively. Both n-QC and n-GQC are based on the classical self-routing concentrators and generalized connection networks given by Lee and Oruç [1]. While these networks work for multicasting classical packets, they cannot be used to multicast quantum packets as they employ balancer switches with both forward and backward propagation of packets. We introduce a quantum balancer switch that works using a forward propagation of packets only, thereby facilitating the n-QC and n-GQC designs presented in the paper.

Index Terms—Generalized quantum connector, quantum concentrator, quantum switching, quantum information, switching networks, quantum multicasting.

1 Introduction

Quantum information science is an emerging area of research that seeks to use special properties of quantum systems such as quantum parallelism and entanglement to develop efficient solutions for classically intractable problems. Research in this area has been spurred by some key algorithms that have shown that quantum systems could be used to solve some important exponentially complex problems with speedups that are impossible in classical computing. Examples include Shor's polynomial time algorithm [2] for finding the prime factors of a composite number and Grover's search algorithm [3] that can find an element in an unstructured database containing n elements in $O(\sqrt{n})$ time.

Building quantum systems would require means to transport quantum information from one place to another. Several architectures are being considered to build *quantum wires* over which quantum data can be transmitted. Primary examples are quantum swapping and teleportation-based architectures for building quantum wires as described in [4]. For n quantum sources that wish to communicate with one another by sharing quantum information, $O(n^2)$ quantum wires are needed. This complexity can be greatly reduced using advanced switching architectures. The relationship between quantum circuits and permutation maps was identified in [5] and it was shown that any permutation map can be realized by a quantum circuit consisting of six layers of controlled-not gates. However, this result requires a different quantum circuit for each permutation map to be realized. It was also shown in [5] that the classical components of a qubit can be replicated using controlled-not gates. This copying was used in [6] to implement multicasting of qubits using directed graph representations of multicast maps. However, this approach also requires that each multicast assignment be realized by a separate quantum circuit.

Shukla et. al. [7], [8] presented a self-routing $O(n \log n)$ quantum baseline network that can be used to permute quantum packets from n inputs to n outputs. They showed that this network can also be used to resolve internal blocking when transmitting classical packets by creating a superposition of packets whenever they contend for the same wire in the network. However, this network can realize only $n^{n/2}$ permutations between n inputs and n outputs out of n total possible permutations. Recently, Cheng and Wang [9] proposed a quantum merge sorting-based switching network that can realize all n permutations using $O(n \log^2 n)$ quantum gates. More recently, Sue [10] described nonblocking quantum switch with $O(n^2)$ quantum gates. This design relies on quantum circuit representations of unicast and multicast maps given in [5] and [6]. The networks in [7], [8], [9], [10] are limited to satisfying one-to-one or unicast assignments between inputs and outputs. In this paper, we focus on multicasting of quantum information on an n-input and n-output network.

Multicasting or generalized connection networks have been extensively studied in the classical information domain. A survey of these networks is given in [11]. These networks can be widely classified into three classes. The first class contains multicast networks based on the three-stage Clos network. Networks in this class require complex routing algorithms and are not self-routing in general. Consequently, these networks cannot easily be implemented using quantum circuits. The second class consists of networks in which multicasting is decomposed into two stages. The first stage, called a generalizer, generates all the required copies of the input packets and the second stage routes these copies to their desired outputs [12], [13], [14].

- *The authors are with the Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742. E-mail: manishk@umd.edu, yavuz@eng.umd.edu.*

Manuscript received 2 Mar. 2009; revised 21 Sept. 2009; accepted 21 Sept. 2009; published online 12 Feb. 2010.
Recommended for acceptance by S. Shukla.
For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number TC-2009-03-0094.
Digital Object Identifier no. 10.1109/TC.2010.52.

The third class of multicasting networks, introduced by Nassimi and Sahni [15], is based on a recursive decomposition of generalized connectors into smaller ones. Nassimi and Sahni's generalized connector, however, requires a parallel computer model connected in cube or perfect shuffle topology for computing routes. Lee and Oruç used a similar approach in their $O(n \log^2 n)$ generalized connector in which routes are determined locally at internal nodes by using local packet headers [1], [11]. Thus, their network is self-routing.

As in the case of quantum Baseline or sorting networks, we begin with a meaningful interpretation of multicasting in a quantum network. In a classical multicasting network, packets at inputs are routed to outputs under the assumption that a packet at any given input may be routed to one or more outputs. One restriction that is often applied is that such multicast assignments may not direct more than one packet to a given output. Furthermore, assignments in a classical multicast network can only be issued one assignment at a time even though they can be overlapped by pipelining. In contrast, in a quantum multicast network, packets themselves are made out of quantum bits and as such they represent a superposition of possible assignment patterns of packets that may be routed through such a network all at once due to the principle of quantum parallelism. It is this quantum parallelism aspect of multicasting that will be explored in this paper.

We extend Lee and Oruç's approach to the domain of quantum information processing to design an $n \times n$ generalized quantum connector that will also be called n-GQC. Since it is impossible to copy arbitrary quantum states due to the no-cloning theorem, multicasting of quantum states is also impossible. However, a certain set of orthogonal states can be copied by Wootters and Zurek's copying machine. For example, a controlled-not gate transforms the two-qubit state $(\alpha|0\rangle + \beta|1\rangle) \otimes |0\rangle$ to $\alpha|00\rangle + \beta|11\rangle$, which can be interpreted as follows: a source qubit that is $|0\rangle$ with probability α^2 and $|1\rangle$ with probability β^2 is copied such that the source and target qubits are in state $|00\rangle$ with probability α^2 and $|11\rangle$ with probability β^2. In essence, the copying of a single qubit in state $|0\rangle$ or $|1\rangle$ into a multitude of qubits as described in [6] is a consequence of such copying. We extend this controlled-not gate-based single-qubit copy operation to a quantum gate that copies quantum information from a set of qubits to another set of qubits, each of which is initialized to the blank state $|0\rangle$. This copying operation amounts to the copying of classical packets contained in a multiqubit quantum state as a probabilistic superposition. Therefore, the n-GQC essentially multicasts *superposed* classical packets that will be referred to as *quantum packets* in this paper.

Such switching of classical packets via quantum switching networks may prove useful for building quantum communication networks when quantum systems become feasible and information is transmitted entirely in quantum domain from source to destination. We would not need to convert inherently quantum data to classical domain, switch it, and then convert it back to quantum. This would lead to the realization of extremely fast quantum communication networks in the future.

In the course of designing a quantum generalized connector (n-GQC), we introduce another quantum network, called an $n \times n$ quantum concentrator (n-QC). This network maps quantum states on any m of its inputs to its top m outputs, for any $m, 1 \leq m \leq n$. The n-QC is a self-routing multistage network that uses $O(n \log n)$ quantum gates. It plays a key role in the decomposition of a generalized connector into smaller ones. As there is no multicasting involved in a concentrator, it can concentrate both arbitrary quantum states and superposed classical packets.

This quantum concentrator is derived by modifying the classical concentrator described by Lee and Oruç. The main bottleneck in transforming this classical concentrator to the quantum domain is the balancer network used in their design that distributes packets from n input ports onto two $n/2$-input networks. Their balancer network is based on a binary tree on which bits are propagated in both forward and backward directions. Here, we propose a reversible balancer, called an n-quantum balancer, on which data are propagated only in the forward direction. This quantum balancer then facilitates the design of a quantum concentrator.

The rest of the paper is organized as follows: In Section 2, we give a brief introduction to core concepts in quantum information science and quantum switching. In Section 3, we define some basic terms and introduce the notation that is used in the rest of the paper. In Sections 4 and 5, we present the designs of the n-QC and the n-GQC, respectively. In Section 6, we study the behavior of n-GQC when more than one input packet is addressed to the same output, i.e., when output contention occurs. In Section 7, we compute the complexities of these networks in terms of number of gates used and routing time or gate-level depth. Finally, Section 8 contains the concluding remarks.

2 Preliminaries

We give a short introduction to core concepts in quantum information science and introduce the quantum gates that are used to construct the n-GQC.

2.1 Qubits and Quantum Gates

The indivisible unit of classical information is the bit that can take either one of the two values: 0 or 1. The corresponding unit of quantum information is the quantum bit or *qubit* that can simultaneously be both 0 and 1. In general, a qubit's state is a unit vector in two-dimensional complex Hilbert space and can be expressed as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, where $\alpha, \beta \in \mathbb{C}$ and $|\alpha|^2 + |\beta|^2 = 1$. Vectors $|0\rangle$ and $|1\rangle$ are called computational bases. On measurement in which the qubit's state is projected onto the computational bases, the qubit is observed to be found either in state $|0\rangle$ or in state $|1\rangle$ with probabilities $|\alpha|^2$ and $|\beta|^2$, respectively.

The state of an n-qubit system is a vector in a 2^n-dimensional complex Hilbert space that is a tensor product of the two-dimensional spaces associated with individual qubits. An n-qubit state can be expressed as

$$|\bar{\psi}\rangle = \alpha_0|00\cdots 0\rangle + \alpha_1|00\cdots 1\rangle + \cdots + \alpha_{2^n-1}|11\cdots 1\rangle, \quad (1)$$

where $\alpha_i \in \mathbb{C}$ and $\sum_{i=0}^{2^n-1} |\alpha_i|^2 = 1$, and the n-bit vectors $|00\cdots 0\rangle, \ldots, |11\cdots 1\rangle$ constitute the bases of the space.

As in a single qubit system, if all the qubits in the n-qubit system are measured, the n-bit string $|\bar{i}\rangle$ is observed with probability $|\alpha_i|^2$, where $i = 0, 1, \ldots, 2^n - 1$. The notation $|\bar{\psi}\rangle$

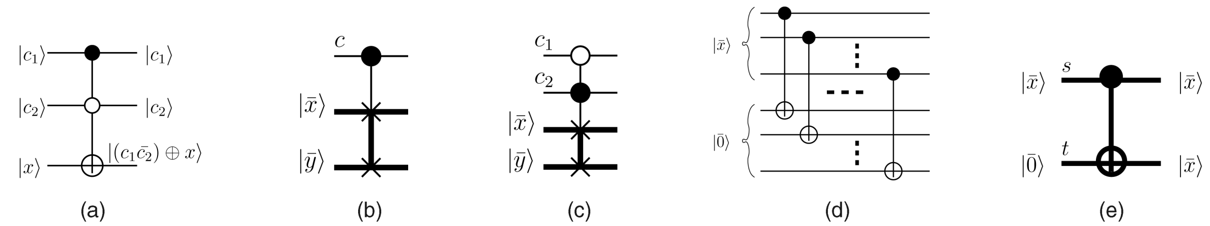


Fig. 1. Quantum gates: (a) a controlled-controlled not gate, (b) switch gate or controlled-swap gate, (c) switch gate with multiple control qubits, (d) controlled-not-gates-based quantum copier, and (e) compact representation of the quantum copier.

will be used to denote a multiqubit vector or bit string in this paper. A single qubit state will be denoted by $|\psi\rangle$.

A quantum gate is a linear and unitary transformation from one quantum state, called an input state, to another quantum state, called an output state. An example for a one-qubit gate is the Hadamard gate, which transforms the basis vectors $|0\rangle$ and $|1\rangle$ as follows:

$$|0\rangle \xrightarrow{\mathbf{H}} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \quad |1\rangle \xrightarrow{\mathbf{H}} \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle). \tag{2}$$

Due to linearity, this gate transforms a general state $\alpha|0\rangle + \beta|1\rangle$ to $\frac{\alpha+\beta}{\sqrt{2}}|0\rangle + \frac{\alpha-\beta}{\sqrt{2}}|1\rangle$. Unitarity implies that quantum gates are *reversible*, i.e., it is possible to identify the input state uniquely from a given output state as seen below:

$$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \xrightarrow{\mathbf{H}} |0\rangle, \quad \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \xrightarrow{\mathbf{H}} |1\rangle, \tag{3}$$

where, in this case, the Hadamard gate is its own inverse.

Another type of quantum gate that is extensively used for manipulating qubits is a controlled quantum gate, e.g., a controlled-Hadamard or a controlled-NOT gate. One such gate, called controlled-controlled-not (CC-NOT) gate that has two control qubits (c_1 and c_2), is shown in Fig. 1a. This gate performs the following operation:

$$|c_1, c_2, x\rangle \overset{\mathbf{CC-NOT}}{\longrightarrow} |c_1, c_2, (c_1.\bar{c}_2) \oplus x\rangle, \tag{4}$$

i.e., it inverts x when $c_1 = 1$ (indicated by solid circle) and $c_2 = 0$ (indicated by open circle). Therefore, it maps basis vector $|100\rangle$ to $|101\rangle$ and $|101\rangle$ to $|100\rangle$. Rest of the basis vectors are passed unchanged. We follow this notation of solid and open circles to indicate the functioning of control qubits in the rest of the paper.

2.2 Quantum Switch Gate and Copier

The basic building block of quantum switching networks is a multiqubit gate, called a controlled-swap gate or a *switch gate*, as shown in Fig. 1b [8], [16], [17]. It swaps two sets of qubits when a control qubit c is $|1\rangle$; otherwise, it passes them unchanged. Therefore, it can be used as a 2×2 switch for routing quantum information. The transformation performed by this gate can be expressed as

$$\begin{aligned} |0\rangle_c|\bar{x}\rangle|\bar{y}\rangle &\overset{\mathbf{SWG}}{\longrightarrow} |0\rangle_c|\bar{x}\rangle|\bar{y}\rangle, \\ |1\rangle_c|\bar{x}\rangle|\bar{y}\rangle &\overset{\mathbf{SWG}}{\longrightarrow} |1\rangle_c|\bar{y}\rangle|\bar{x}\rangle, \end{aligned} \tag{5}$$

where $|\bar{x}\rangle$ and $|\bar{y}\rangle$ are equal length binary strings. A switch gate can have multiple control inputs. For example, the switch gate shown in Fig. 1c swaps its input packets when control qubits c_1 and c_2 are $|0\rangle$ and $|1\rangle$, respectively; otherwise, it directly passes them on its outputs.

While quantum switch gates are essential to perform permutation maps, we need quantum copiers to replicate the classical components of multiqubit quantum states. This is done by using a collection of controlled-not gates as shown in Fig. 1d by which a set of *source qubits* is copied to a set of *target qubits* initialized to state $|0\rangle$. We represent these gates collectively as one controlled-not gate using bold lines, as shown in Fig. 1e. The copying operation performed by this gate is

$$|\bar{x}\rangle_s|\bar{0}\rangle_t \overset{\mathbf{COP}}{\longrightarrow} |\bar{x}\rangle_s|\bar{x}\rangle_t. \tag{6}$$

As an example, two source qubits in state

$$1/\sqrt{2}|00\rangle_s + 1/2|10\rangle_s + 1/2|11\rangle_s \tag{7}$$

are transformed as

$$\begin{aligned} &\left(\frac{1}{\sqrt{2}}|00\rangle_s + \frac{1}{2}|10\rangle_s + \frac{1}{2}|11\rangle_s\right)|00\rangle_t \\ &\quad\overset{\mathbf{COP}}{\longrightarrow} \frac{1}{\sqrt{2}}|00\rangle_s|00\rangle_t + \frac{1}{2}|10\rangle_s|10\rangle_t + \frac{1}{2}|11\rangle_s|11\rangle_t, \end{aligned} \tag{8}$$

which shows that a copy of each component of a quantum state is created. This copier, which is essentially Wootters and Zurek's quantum copying machine [18], [19], is exactly what we need when multicasting superposed classical data using a quantum switching network.

3 Quantum Packets and Quantum Assignments

A quantum packet consists of two sets of qubits, called the *address field* and *data* qubits, respectively, and an extra qubit, called a routing qubit that indicates the presence (when set to $|1\rangle$) or absence (when set to $|0\rangle$) of a quantum packet on the corresponding address field and data qubits. The address field contains routing information that is used to route the data qubits.

A *quantum packet*, consisting of k classical packets in which packet i has routing bit r_i, address field $\bar{a}_i$, and data $\bar{d}_i$, for $i = 0, 1, \ldots, k-1$, is represented as

$$\sum_{i=0}^{k-1} \alpha_i |r_i, \bar{a}_i, \bar{d}_i\rangle, \tag{9}$$

where each $|r_i, \bar{a}_i, \bar{d}_i\rangle$ denotes a classical packet with probability coefficient of $|\alpha_i|^2$ and it is to be multicast to the set of outputs specified by its address field $\bar{a}_i$ when r_i is

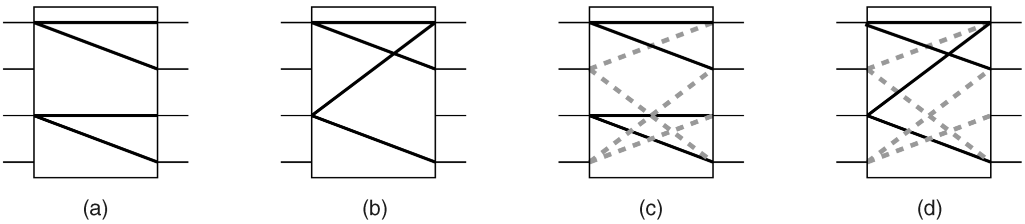


Fig. 2. *Assignment patterns:* (a) noncontending assignment pattern; (b) contending assignment pattern; *Quantum assignments* (solid and dashed lines show two assignment patterns): (c) noncontending quantum assignment; (d) contending quantum assignment.

1. The set of outputs specified by the address field $\bar{a}_i$ is called the *fan-out set* of packet i and is represented as F_i. The size of the fan-out set is called the *fan-out* of the packet i. When r_i is 0, packet $|r_i, \bar{a}_i, \bar{d}_i\rangle$ is considered to be empty, i.e., even though the packet exists, a quantum switching network routing this packet will ignore its address and data bits.

We give an example to make the definition of a quantum packet more clear. Consider a 4×4 network in which an input has two packets A and B, which are to be routed with probabilities $3/4$ and $1/4$, respectively. Suppose that the fan-out sets of A and B are $\{1, 3\}$ and $\{1\}$, respectively. We use a 4-bit address field, $o_0o_1o_2o_3$, where o_i is set to 1 when the fan-out set of a packet contains output i. The quantum packet on this input is a superposition of two classical packets, expressed as

$$\frac{\sqrt{3}}{2}|1, 0101, A\rangle + \frac{1}{2}|1, 0010, B\rangle. \tag{10}$$

In general, at least n bits are needed to specify the destinations of an input in an $n \times n$ multicast network, as noted in [1]. This is because an input can have up to 2^n destination patterns.

There is more than one possible interpretation of this quantum packet representation. We mention two such interpretations. One is that if A and B denote the same packet, then this packet will likely be routed to outputs 1 and 3 with probability $3/4$ and to output 2 with probability $1/4$. The second interpretation is that the input source is likely to generate one of the two different packets. One of these two packets is generated and routed to outputs 1 and 3 with probability $3/4$ and the other is generated and routed to output 2 with probability $1/4$. Nonetheless, in both interpretations, either a packet appears at both outputs 1 and 3 with probability $3/4$ or at output 2 only with probability $1/4$. Therefore, for consistency of our statements, we shall assume the second interpretation.

An *assignment pattern* over an $n \times n$ quantum switching network is a sequence of classical packets, each of which belongs to a quantum packet on a distinct input of the network from top to bottom. We say that an assignment pattern is *noncontending* when no two classical packets with routing bits of 1 in the pattern are addressed to the same output. A *quantum assignment* on an $n \times n$ quantum switching network is a superposition of a set of assignment patterns. A quantum assignment is called *noncontending* if and only if all of its assignment patterns are noncontending; and called *contending* otherwise. These definitions are illustrated in Fig. 2. An assignment pattern is said to be *unicast* when each classical packet is addressed to at most one output; otherwise, it is said to be *multicast*. A noncontending unicast assignment pattern in which every input is paired with some output is said to be a *permutation* assignment pattern.

A quantum assignment consisting of a superposition of a set of M assignment patterns on an $n \times n$ quantum switching network is called a *quantum M-assignment* and can be expressed as

$$\sum_{k=0}^{M-1} \beta_k |(r_{k,0}, \bar{a}_{k,0}, \bar{d}_{k,0}) \cdots (r_{k,n-1}, \bar{a}_{k,n-1}, \bar{d}_{k,n-1})\rangle, \tag{11}$$

where assignment pattern $|\bar{P}_k\rangle = |(r_{k,0}, \bar{a}_{k,0}, \bar{d}_{k,0}), \ldots, (r_{k,n-1}, \bar{a}_{k,n-1}, \bar{d}_{k,n-1})\rangle$ is a sequence of classical packets in which $(r_{k,i},\ \bar{a}_{k,i}, \bar{d}_{k,i})$ is the classical packet on input i. The probability with which the kth assignment pattern is realized is $|\beta_k|^2$, where $\sum_{k=0}^{M-1} |\beta_k|^2 = 1$.

When all the inputs have quantum packets of the form given in (9), the expression for the corresponding quantum assignment can be obtained by taking the tensor product of all the input quantum packets.

We illustrate this by considering the example of a 4×4 network in which input 0 issues the quantum packet $\frac{\sqrt{3}}{2}|1, 0101, A\rangle + \frac{1}{2}|1, 0100, B\rangle$, input 1 has no packet, input 2 issues the quantum packet $|1, 0010, D\rangle$, and input 3 issues the quantum packet $\frac{1}{\sqrt{2}}|1, 1001, E\rangle + \frac{1}{\sqrt{2}}|1, 1000, F\rangle$. Then, the quantum assignment is

$$\left(\frac{\sqrt{3}}{2}|1, 0101, A\rangle + \frac{1}{2}|1, 0100, B\rangle\right) \otimes |0, 0000, C\rangle \otimes |1, 0010, D\rangle \otimes \left(\frac{1}{\sqrt{2}}|1, 1001, E\rangle + \frac{1}{\sqrt{2}}|1, 1000, F\rangle\right), \tag{12}$$

which can be written as a superposition of four multicast assignment patterns

$$\begin{aligned} &\frac{\sqrt{3}}{2\sqrt{2}}|(1, 0101, A), (0, 0000, C), (1, 0010, D), (1, 1001, E)\rangle \\ &+ \frac{\sqrt{3}}{2\sqrt{2}}|(1, 0101, A), (0, 0000, C), (1, 0010, D), (1, 1000, F)\rangle \\ &+ \frac{1}{2\sqrt{2}}|(1, 0100, B), (0, 0000, C), (1, 0010, D), (1, 1001, E)\rangle \\ &+ \frac{1}{2\sqrt{2}}|(1, 0100, B), (0, 0000, C), (1, 0010, D), (1, 1000, F)\rangle. \end{aligned} \tag{13}$$

It is seen that the first assignment pattern in the above quantum assignment is *contending* because packets A and E

in this pattern are addressed to output 3. The rest of the three assignment patterns are *noncontending*.

Quantum assignments are realized by a generalized quantum connector, which is defined and constructed in Section 5. This network is constructed using a quantum concentrator, which is described in Section 4. The basic building block of our quantum concentrator design is another network, which is called a quantum odd-even splitter and is described in Section 4.2. This network is, in turn, constructed using what is called a quantum balancer that is described in Section 4.1.

4 Quantum Concentrator

To construct a generalized quantum connector, a quantum concentrator will be used. A quantum concentrator is a quantum switching network that transforms an input assignment pattern in which only the routing bits matter and the address fields are ignored. The packets with routing bits of 1 are routed consecutively to the top outputs of the network in some nonspecifiable order. The remaining packets are routed to the remaining outputs. An $n \times n$ quantum concentrator will be denoted by n-QC.

We represent the transformation applied by an n-QC to an input assignment pattern $|\bar{P}\rangle$ as

$$|\bar{P}\rangle|\bar{0}\rangle_{aux} \stackrel{n\text{-QC}}{\longrightarrow} |QC(\bar{P})\rangle|\Phi(\bar{P})\rangle_{aux}, \quad (14)$$

where $|\Phi(\bar{P})\rangle_{aux}$ denote output auxiliary qubits and $|QC(\bar{P})\rangle$ denotes the output assignment pattern for input assignment pattern $|\bar{P}\rangle$. Due to the linearity of quantum networks, a quantum assignment $\sum_{k=0}^{M-1} \beta_k|\bar{P}_k\rangle$ is transformed by an n-QC as

$$\begin{aligned}&\left(\sum_{k=0}^{M-1} \beta_k|\bar{P}_k\rangle\right)|\bar{0}\rangle_{aux}\\ &\stackrel{n\text{-QC}}{\longrightarrow} \sum_{k=0}^{M-1} \beta_k|QC(\bar{P}_k)\rangle|\Phi(\bar{P}_k)\rangle_{aux}.\end{aligned} \quad (15)$$

Thus, all the assignment patterns in the quantum assignment are individually concentrated in parallel.

When a quantum concentrator maps the packets in two or more input assignment patterns with the same number of coinciding routing bits of 1, it must apply the same permutation to all the input assignment patterns. For example, if a 4-QC concentrates input pattern $|(1,A)(0,X)(1,B)(0,Y)\rangle$ to pattern $|(1,B)(1,A)(0,X)(0,Y)\rangle$, then it must concentrate another pattern $|(1,A')(0,X')(1,B')(0,Y')\rangle$ to pattern $|(1,B')(1,A')(0,X')(0,Y')\rangle$.

For an n-QC to be reversible, auxiliary qubits are needed since two or more input assignment patterns may be mapped to the same output assignment pattern. The following theorem specifies the number of auxiliary output qubits needed to make an n-QC reversible:

Theorem 1. *The minimum number of auxiliary qubits needed in order to realize an* n-*QC is* $\lceil \log_2 \binom{n}{\lceil \frac{n}{2} \rceil} \rceil$.

Proof. For a fixed output assignment pattern in which top m packets have routing bits of 1, there are $\binom{n}{m}$ possible input assignment patterns that would be concentrated to the given output pattern. In order to ensure reversibility, the output state of the auxiliary qubits should be different for each of these input patterns. Therefore, we require at least $\lceil \log_2 \binom{n}{m} \rceil$ auxiliary qubits, where $0 \le m \le n-1$. Since $\lceil \log_2 \binom{n}{m} \rceil$ is maximum for $m = \lceil \frac{n}{2} \rceil$, we need at least $\lceil \log_2 \binom{n}{\lceil \frac{n}{2} \rceil} \rceil$ auxiliary qubits.

To show sufficiency, assume that the total number of input qubits for the n-QC is $n_i = n(1+d)$, where d is the number of address and data qubits combined in each packet, and we have n_{aux} auxiliary qubits, where n_{aux} satisfies the condition given in the theorem. Including the auxiliary qubits, the n-QC is a $(n_i + n_{aux})$-qubit quantum circuit. We can concentrate the 2^{n_i} bit strings $|\bar{P}\rangle|\bar{0}\rangle_{aux}$ that correspond to all the assignment patterns by breaking the ties using auxiliary qubits. Having mapped bit strings $|\bar{P}\rangle|\bar{0}\rangle_{aux}$ in a one-to-one fashion, the remaining $2^{(n_i+n_{aux})} - 2^{n_i}$ input bit strings in which auxiliary qubits are not all in state $|0\rangle$ can be mapped to the same number of remaining output bit strings arbitrarily in a one-to-one fashion. This establishes the reversibility of n-QC and the statement follows. □

Observe that $\lceil \log_2 \binom{n}{\lceil \frac{n}{2} \rceil} \rceil$ is $O(n)$. Therefore, an n-QC requires at least $O(n)$ auxiliary qubits. The construction of n-QC given in this paper uses $O(n \log n)$ auxiliary qubits. Also, since the n-QC operation is defined as a permutation of the packets in an assignment pattern, it is easy to verify that the n-QC routes arbitrary quantum states on any m of the inputs to its top m outputs [20]. This can be accomplished by setting the routing qubits on the inputs that need to be concentrated to $|1\rangle$.

In rest of the section, we present an n-QC (n being a power of 2) using the classical n-concentrator given by Lee and Oruç [1], [11] as a starting point. We first design a recursive quantum balancer network, which is constructed using controlled-not gates and does not require backward data propagation. This quantum balancer is used to obtain a quantum odd-even splitter, which divides the packets with routing bits of 1 in its input assignment pattern equally between its odd- and even-numbered outputs. The packets on odd and even sets of outputs are then recursively concentrated by two $n/2$-QCs and are finally merged using a shuffle stage to obtain an n-QC.

4.1 Quantum Balancer

An *n-quantum balancer*, denoted by n-QB, is an n-qubit quantum gate, which transforms a computational basis vector or bit string $|b_0 b_1 \cdots b_{n-1}\rangle$ in which m bits are 1 to an output bit string with $\lfloor m/2 \rfloor$ of these bits converted to 0 and $\lceil m/2 \rceil$ left unchanged, where $1 \le m \le n$. The bits b_i which are 0 can be converted to either 0 or 1 so that the overall mapping is reversible, i.e., there is a one-to-one mapping between the 2^n input and output bit strings. We show by induction that it is possible to construct an n-QB.

For $n = 1$, an identity gate works as a 1-QB. For $n = 2$, the following transform, which is a controlled-not gate, works as a 2-QB: $|00\rangle \to |00\rangle$, $|01\rangle \to |01\rangle$, $|10\rangle \to |11\rangle$, $|11\rangle \to |10\rangle$. For $n > 2$, suppose that we have a reversible k-QB for every $k < n$. Choose nonzero n_1 and n_2 such that $n_1 + n_2 = n$. The first n_1 qubits are balanced using an n_1-QB

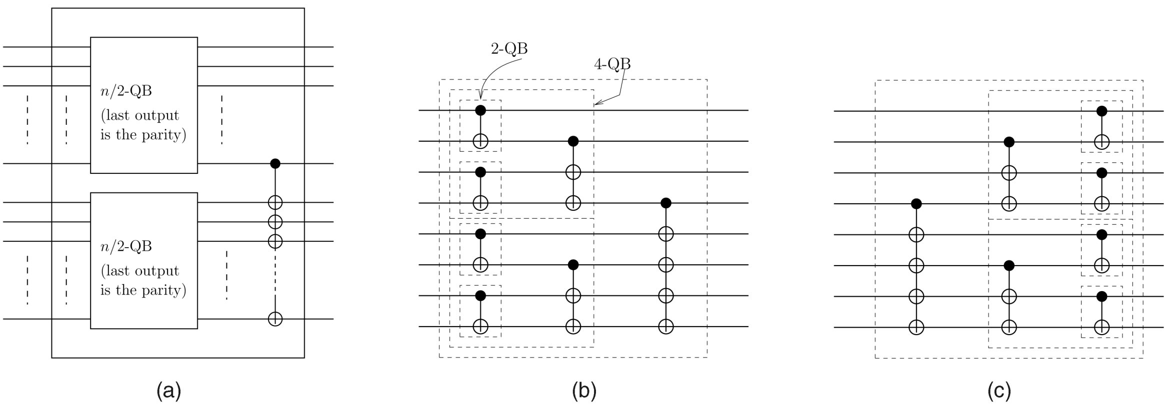


Fig. 3. Quantum balancer: (a) an n-QB in which the last output bit is 1 if the inputs have odd parity and is 0 if the input bits have even parity; (b) 8-QB; and (c) inverse 8-QB.

and the rest of the qubits are balanced using an n_2-QB. Suppose that there are m_1 ones in string $b_0 \cdots b_{n_1-1}$ and m_2 ones in $b_{n_1} \cdots b_{n-1}$. We complement every output of the n_2-QB if m_1 is odd; otherwise, we leave them unchanged. To see that we now have a n-QB: if m_1 is even, $\lfloor m_1/2 \rfloor + \lfloor m_2/2 \rfloor = \lfloor (m_1+m_2)/2 \rfloor$ of the ones are converted to zeros. If m_1 is odd, $\lfloor m_1/2 \rfloor + \lceil m_2/2 \rceil = \lfloor (m_1+m_2)/2 \rfloor$ of the ones are converted to zeros. Next, we verify that this network is reversible. Since n_1-QB is reversible, we can uniquely determine $b_0 \cdots b_{n_1-1}$ from the top n_1 outputs from which we can obtain m_1. We complement the bottom n_2 output bits if m_1 is odd; otherwise, we leave them unchanged. Hence, we can uniquely determine $b_{n_1} \cdots b_{n-1}$ using the resulting bits, since n_2-QB is reversible. Therefore, we can construct an n-QB for any $n \geq 1$ using the procedure that has been described.

For n being a power of 2, i.e., $n = 2^p$, we can recursively construct an $O(\log n)$ depth n-QB by choosing $n_1 = n_2 = n/2$. We present a quantum circuit realization of the n-QB using this procedure in Fig. 3. We place an additional requirement on the n-QB that its last or bottommost output bit is always equal to the parity of the input string $|b_1 b_2 \cdots b_n\rangle$, while satisfying the functionality of an n-QB given in the above definition. This network consists of two similar $n/2$-QB gates followed by an $(n/2+1)$-input controlled-not gate, which complements the output bits of the bottom $n/2$-QB when the parity output of the top $n/2$-QB is 1; otherwise, it does not affect any of the outputs. Next, we show that this network is an n-QB.

Theorem 2. *The recursive network shown in Fig. 3a is an n-QB (for n being a power of 2, i.e.,* $n = 2^p$*) in which the last output is the parity of the input bits.*

Proof. For $p = 1$, input sequences 00, 01, 10, and 11 are mapped to 00, 01, 11, and 10, respectively, by a controlled-not gate, therefore, it is a 2-QB and it also satisfies the parity requirement. For $p > 1$, since the $(n/2+1)$-input controlled-not gate complements all the outputs of the lower $n/2$-QB when the last output of upper $n/2$-QB is 1, it is sufficient to show that the bottommost or last output bit of n-QB represents the parity of its input bits. Suppose that 2^p-QB satisfies this condition. Assuming that m_1 and m_2 are the number of input bits that are 1 in the upper and lower halves of a 2^{p+1}-QB, respectively, we have the following four cases for the 2^{p+1}-network:

1. *Both* m_1 *and* m_2 *are even*: In this case, the (2^p+1)-input controlled-not gate is not active since the parity output of the top 2^p-QB is 0. The output parity bit is 0 since the parity output of bottom 2^p-QB is 0.
2. m_1 *even and* m_2 *odd*: Again, the controlled-not gate is not active and the output parity bit is 1 since the parity output of lower 2^p-QB is 1.
3. m_1 *odd and* m_2 *even*: In this case, the controlled-not gate is active since the parity output of top 2^p-QB is 1. It complements all the output bits of the bottom 2^p-QB. Thus, the output parity bit becomes 1.
4. *Both* m_1 *and* m_2 *odd*: In this case also, the controlled-not gate complements all the output bits of the bottom 2^p-QB. Thus, the output parity bit becomes 0.

Therefore, the parity requirement is satisfied for the 2^{p+1}-QB, and this completes the proof. □

It can be easily verified that the mirror image of the n-QB, as shown in Fig. 3c, restores the output qubits of the quantum balancer to their initial state. This network is needed for restoring some of the auxiliary qubits used in the quantum odd-even splitter to their initial state as explained in the next section.

4.2 Quantum Odd-Even Splitter

An *n-quantum odd-even splitter* is an $n \times n$ switching network, which permutes an input assignment pattern in which m packets have routing bits of 1, such that $\lceil m/2 \rceil$ of these packets appear on the even outputs and the remaining appear on the odd outputs, where $1 \leq m \leq n-1$. A set of auxiliary qubits initialized to $|\bar{0}\rangle$ is used to ensure reversibility.

Our construction of an 8-input odd-even splitter is shown in Fig. 4. The n-quantum odd-even splitter network consists of $n/2$ splitter switches, SW_0, $SW_1, \ldots, SW_{n/2-1}$ driven by control qubits $c_0, c_1, \ldots, c_{n/2-1}$, respectively. For a given assignment pattern, switch SW_i is said to be *balanced* if it has packets on both of its inputs or no packets at all, i.e., $r_{2i} = r_{2i+1}$. The address and data fields of the packets in the assignment pattern are collectively represented as $\bar{p}_i$ in the

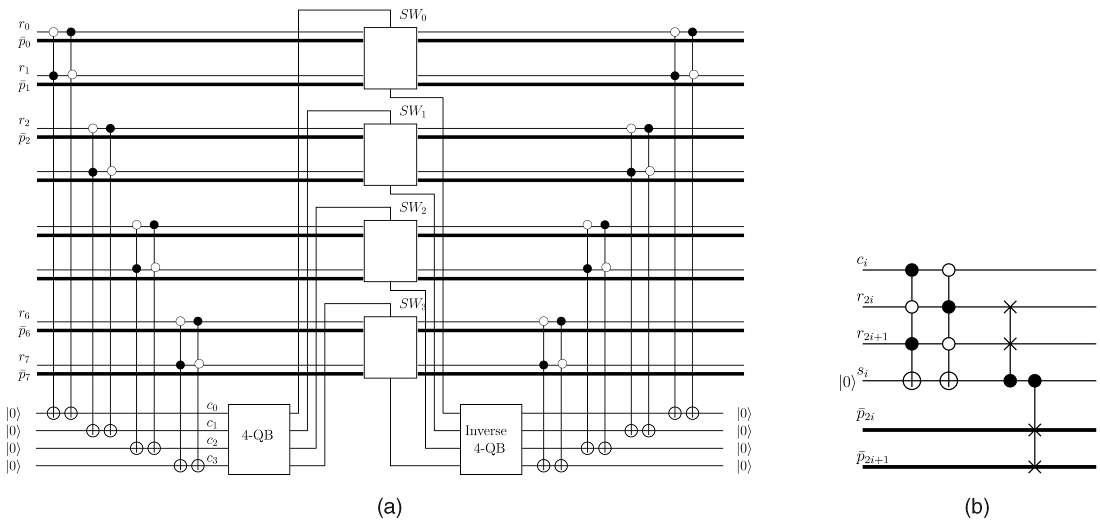


Fig. 4. Quantum odd-even splitter: (a) 8-input quantum odd-even splitter, (b) splitter switch (SW_i).

figure. Using the $n/2$-QB on qubits $c_0, c_1, \ldots, c_{n/2-1}$, half of the packets at the unbalanced switches are routed to the even-numbered outputs and the other half are routed to the odd-numbered outputs of quantum odd-even splitter. The packets on the balanced switches are always equally distributed between the odd-and even-numbered outputs, irrespective of the switch settings. Therefore, the quantum odd-even splitter equally distributes the input packets between the odd and even outputs. A brief description of the quantum circuit follows.

Using two controlled-not gates, qubit c_i is set in state $|1\rangle$ if switch SW_i is unbalanced; otherwise, it is set in state $|0\rangle$. The $n/2$-QB balances the control qubits c_i, $i = 0, \ldots, n/2-1$, which control the splitter switches. The quantum circuit for a splitter switch is shown in Fig. 4b. This switch uses one extra qubit s_i, called *switching qubit*, which was not shown in Fig. 4a. By setting this qubit appropriately, and using switch gates as shown in the figure, the input packet on an unbalanced switch is routed to the upper (or even) output when the control qubit c_i is $|1\rangle$; otherwise, it is routed to the lower (or odd) output. A balanced splitter switch may be set either way without affecting the splitting property of the odd-even splitter.

The quantum gates and the inverse $n/2$-QB shown on the right side of switches SW_i in Fig. 4a are used for restoring qubits c_i to $|0\rangle$. Even though the routing bits might have been switched by the splitter switches along with the input packets, they would still maintain their balanced or unbalanced status. Therefore, these qubits can be used to restore the control qubits c_i to their original state $|0\rangle$, as shown in the figure, so that decoherence on qubits c_i does not have any effect on the performance of the network. Only the switching qubits s_i have not been restored to their initial state.

4.3 Construction of n-QC

We can recursively realize an n-QC by using an n-quantum odd-even splitter and two $n/2$-QCs as shown in Fig. 5a. Using a Banyan connection pattern, the even outputs of the odd-even splitter are connected to the upper $n/2$-QC and the odd outputs are connected to the lower $n/2$-QC. The outputs of $n/2$-QCs are connected alternately to the final n outputs using a shuffle connection pattern.

Theorem 3. *The network shown in Fig. 5a is an n-QC.*

Proof. For $n = 2$, a splitter switch in which the control input is set to $|1\rangle$ works as a 2-QC. For $n = 2^p$, where $p > 1$,

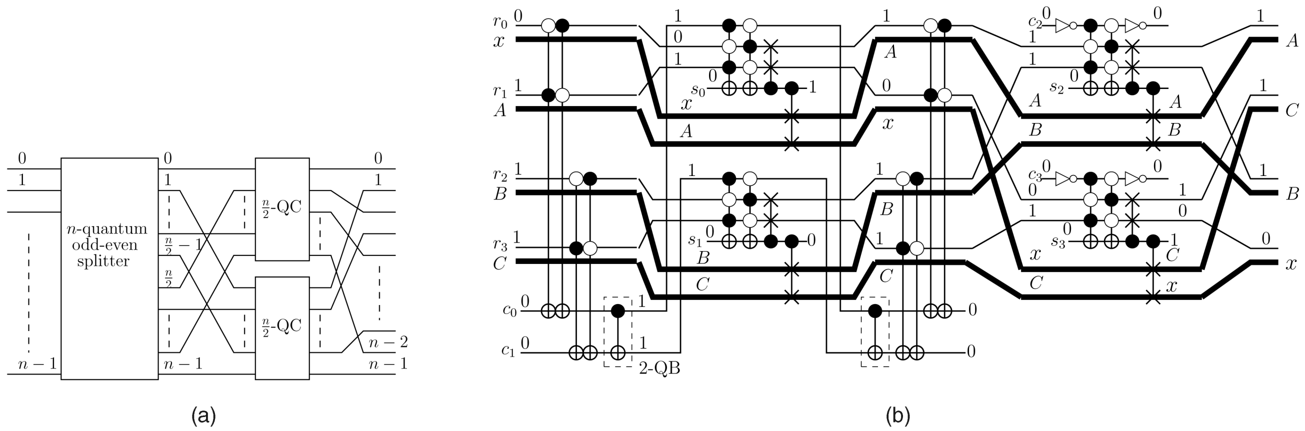


Fig. 5. n-quantum concentrator (n-QC): (a) recursive construction of n-QC, (b) 4-QC quantum circuit.

consider an input assignment pattern in which m packets have routing bits of 1, where $0 \leq m \leq n$. The odd-even splitter transforms it into an assignment pattern in which $\lceil m/2 \rceil$ of these packets are at the inputs of the upper $n/2$-QC and $\lfloor m/2 \rfloor$ are at the inputs of the lower $n/2$-QC. The $n/2$-QCs concentrate their input assignment patterns and these packets are concentrated at their top $\lceil m/2 \rceil$ and $\lfloor m/2 \rfloor$ outputs, respectively. These packets are routed to the top m outputs of the n-QC using the shuffle connection. Therefore, by induction, the packets having routing bits of 1 are concentrated to the top m outputs and the packets with routing bits of 0 are sent to the bottom $n - m$ outputs.

We now show that when an n-QC maps the packets in two or more input assignment patterns with the same number of coinciding routing bits of 1, it applies same permutation to all of them. Since the sequence of routing bits is fixed for all assignment patterns, the settings of the splitter switches in the odd-even splitter are fixed, therefore, the sequences of routing bits at the inputs of the upper and lower $n/2$-QCs are also fixed. This inductively implies that the setting of every splitter switch in the network is fixed. Consequently, same permutation is applied to all assignment patterns having the same sequence of routing bits.

Next, we show that if two different assignment patterns $|\bar{P}_1\rangle$ and $|\bar{P}_2\rangle$ are mapped to the same output pattern $|QC(\bar{P}_1)\rangle = |QC(\bar{P}_2)\rangle$, then $|\phi(\bar{P}_1)\rangle_{aux} \neq |\phi(\bar{P}_2)\rangle_{aux}$. Since the control qubits are restored to their initial state, only the switching qubits constitute as auxiliary qubits. We need to show that the output state of at least one of the switching qubits is different for these two input assignment patterns. Clearly, $|\bar{P}_1\rangle$ and $|\bar{P}_2\rangle$ must have the same number of packets with routing bits of 1. Also, since any two assignment patterns with coinciding routing bits of 1 are concentrated using the same permutation, there must be at least one input at which the routing bits in $|\bar{P}_1\rangle$ and $|\bar{P}_2\rangle$ are different. Therefore, this input would be connected to one of the top m outputs for one pattern and one of the bottom $n - m$ outputs for the other pattern. Hence, the overall $n \times n$ permutations applied to concentrate $|\bar{P}_1\rangle$ and $|\bar{P}_2\rangle$ must be different. The topology of the n-QC is easily seen to provide a unique path between each input and each output. Therefore, for the two permutations to be different, at least one of the 2×2 splitter switches must be set in through state for one of the assignment patterns and in cross state for the other. Consequently, the output states of the two assignments are different when the switching qubits s_i are taken into account and we have $|\phi(\bar{P}_1)\rangle_{aux} \neq |\phi(\bar{P}_2)\rangle_{aux}$. □

To illustrate the concentration operation done by n-QC, we give an example for $n = 4$. An expanded quantum circuit for the 4-QC is shown in Fig. 5a. The input quantum assignment pattern has three classical packets A, B and C, on inputs 1, 2, and 3, respectively. Input 0 has no packet, i.e., $r_0 = 0$. The address and data fields of this input are denoted by x in the figure. Eight auxiliary qubits labeled $c_0, \ldots, c_3$ and $s_0, \ldots, s_3$ are used. All of them are initialized to state $|0\rangle$. For clarity, we have not used the *ket* notation in the figure. The transformation done by the 4-QC in this example is expressed as

$$|(0,x)(1,A),(1,B),(1,C)\rangle|\bar{0}\rangle_{aux} \stackrel{\text{4-QC}}{\longrightarrow} |(1,A)(1,C),(1,B),(0,x)\rangle\big|0_{c_0}0_{c_1}0_{c_2}0_{c_3}1_{s_0}0_{s_1}0_{s_2}1_{s_3}\big\rangle_{aux.}$$

It is seen that the input packets are concentrated on the top three outputs of the 4-QC. Control qubits c_i are restored to $|0\rangle$, whereas some of the switching qubits s_i are not restored.

5 Generalized Quantum Connector

A classical switching network is called a *generalized connector* if it realizes any noncontending multicast assignment pattern between its inputs and outputs. In this section, we extend this definition to the quantum domain for multicasting quantum packets. We go back to the notation introduced in Section 3 to represent quantum assignment patterns in which an input classical packet is represented by the 3-tuple $(r, \bar{a}, \bar{p})$, where $\bar{a}$ and $\bar{p}$ are m_a and m_d bit binary strings, respectively.

An *n-generalized quantum connector or n-GQC* is an $n \times n$ quantum switching network, which transforms any noncontending assignment pattern $|\bar{P}\rangle = |(r_0, \bar{a}_0, \bar{d}_0), \ldots, (r_{n-1}, \bar{a}_{n-1}, \bar{d}_{n-1})\rangle$ such that, for $i = 0, \ldots, n-1$, if $r_i = 1$, then the data $\bar{d}_i$ of the packet on input i are copied onto the data fields of all the outputs in its fan-out set. The routing bits on these outputs are set to 1. The routing bit of an output to which no input packet is addressed is set to 0. Each output consists of routing and data qubits only and does not contain address qubits. We represent this transformation as follows:

$$\begin{aligned} |\bar{P}\rangle|\bar{0}\rangle_{\text{aux}} &\stackrel{n\text{-GQC}}{\longrightarrow} |GQC(\bar{P})\rangle|\Psi(\bar{P})\rangle \\ &= \big|(r'_0, \bar{d}'_0), \ldots, (r'_{n-1}, \bar{d}'_{n-1})\big\rangle|\Psi(\bar{P})\rangle. \end{aligned} \tag{16}$$

The auxiliary qubits on the left-hand side are needed for two reasons: to ensure reversibility and to create copies of the input packets. The auxiliary and address qubits are transformed to a state $|\Psi(\bar{P})\rangle$ such that, for any two different input assignment patterns $|\bar{P}_1\rangle$ and $|\bar{P}_2\rangle$ for which $|GQC(\bar{P}_1)\rangle = |GQC(\bar{P}_2)\rangle$, we have $|\Psi(\bar{P}_1)\rangle \neq |\Psi(\bar{P}_2)\rangle$. Again, due to the linearity of quantum networks, an n-GQC simultaneously realizes all the assignment patterns in a noncontending quantum assignment, i.e., a quantum assignment $\sum_{k=0}^{M-1} \beta_k|\bar{P}_k\rangle$, in which every assignment pattern is noncontending, that is transformed as follows:

$$\begin{aligned} &\left(\sum_{k=0}^{M-1} \beta_k|\bar{P}_k\rangle\right)|\bar{0}\rangle_{aux} \\ &\stackrel{n\text{-GQC}}{\longrightarrow} \sum_{k=0}^{M-1} \beta_k|GQC(\bar{P}_k)\rangle|\Psi(\bar{P}_k)\rangle. \end{aligned} \tag{17}$$

The address bits in our quantum packet representation denote the outputs of the n-GQC and they are not needed once the packets have reached their desired n-GQC outputs. When the n-GQC is used as a part of a larger network and the packets are to be routed further, extra address bits are needed. In our representation, these address bits can be included in the data parts of the packets and they are not discarded by the n-GQC.

Before giving our n-GQC construction, we describe the addressing schemes that we use to represent address fields in quantum packets.

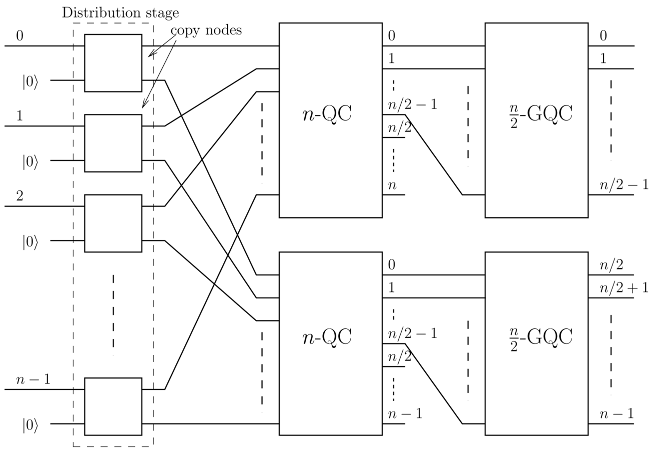


Fig. 6. Generalized quantum connector.

5.1 Addressing Schemes for the n-GQC

As mentioned in Section 3, since there are 2^n possible fan-out sets for each input in an $n \times n$ network, at least n bits are needed per input to address these patterns. The most straightforward way to code these fan-out sets is to allocate n bits $o_0, o_1, \ldots, o_{n-1}$ for each input in which o_j is set to 1 when that input is paired with output j. In this paper, we use a $(2n-2)$-bit addressing scheme that is more suitable for quantum circuit realization of a multistage n-GQC. Both of these addressing schemes were introduced in [1].

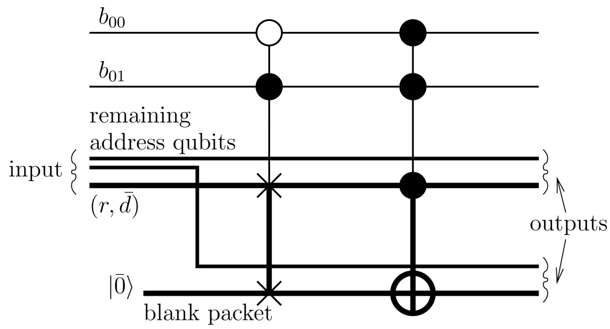
In the $(2n-2)$-bit addressing scheme, where n is a power of 2, each input uses a binary address of the form $b_{00}b_{01},\ b_{10}b_{11}b_{12}b_{13}, \ldots, b_{p-1,0}b_{p-1,1}\cdots b_{p-1,n-1}$ to specify the outputs it is paired with, where $p = \log_2 n$. The first two bits specify whether the packet at an input is routed to the upper half or lower half or both upper and lower halves of outputs or not routed at all. The next group of four bits is then used to resolve the location of the same packet within each half of the upper and lower halves of outputs, and this is inductively extended. More specifically, for $k = 0, \ldots, p-1$, the outputs are divided into 2^{k+1} sets of size 2^{p-k-1} of the form $i2^{p-k-1} \leq j \leq (i+1)2^{p-k-1} - 1$, where $i = 0, \ldots, 2^{k+1} - 1$. For a given input, an address bit b_{ki} is set to 1 when that input is paired with at least one output in set $i2^{p-k-1} \leq j \leq (i+1)2^{p-k-1} - 1$.

We use the $(2n-2)$-bit addressing scheme in this paper since it leads to a simpler quantum circuit implementation. These bits collectively form the address field $\bar{a}_i$ in the representation of a quantum packet given in (9).

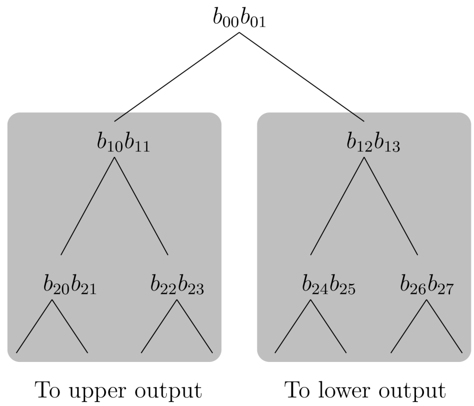
5.2 Construction of n-GQC

In this section, we present a multistage quantum network realization of n-GQC, where $n = 2^p$. It is a recursive network consisting of a distribution stage followed by two n-QCs and two $n/2$-GQCs, as shown in Fig. 6. The distribution stage consists of n copy nodes, labeled 0 to $n-1$ from top to bottom. Given an input assignment pattern for the distribution stage, a copy node does one of the following:

- It creates copies of its input packet on both of its outputs if the fan-out set of the packet contains at least one output from both top and bottom $n/2$ outputs of the n-GQC. For the $(2n-2)$-bit addressing scheme, this happens when $b_{00}b_{01} = 11$.
- It routes the input packet on its upper output if that input packet has at least one output from the top $n/2$ outputs but no output from the bottom $n/2$ outputs of the n-GQC in its fan-out set. No packet is sent to the bottom half of outputs in this case. This happens when $b_{00}b_{01} = 10$.
- It routes the input packet on its lower output if that input packet has at least one output from the bottom $n/2$ outputs but no output from the top $n/2$ outputs of the n-GQC in its fan-out set. No packet is sent to the top half of outputs in this case. This happens when $b_{00}b_{01} = 01$.
- It divides the remaining address bits between the two outputs using the scheme shown in Fig. 7b. Consequently, the size of the address fields in the packets received by the $n/2$-GQCs in Fig. 6 is $n-2$. The same addressing scheme is recursively followed in the subsequent stages and a copy node always uses the first two address bits of its input packet to determine its settings, irrespective of its stage.



(a)



(b)

Fig. 7. Copy node: (a) quantum circuit for a copy node, (b) division of the address field at a copy node for the $(2n-2)$-bit addressing scheme.

The address bits $b_{00}b_{01}$ are never set to 00 for an input packet with routing bit of 1. Consequently, the routing and data bits of an input packet having $b_{00}b_{01} = 00$ can be passed to either of the outputs without affecting the n-GQC functionality.

The quantum circuit implementation of a copy node is shown in Fig. 7a. When b_{00} and b_{01} are 00, routing and address bits are passed to the upper output by convention. When b_{00} and b_{01} are 10, the routing and data bits are passed to the upper output. When they are 01, the routing and

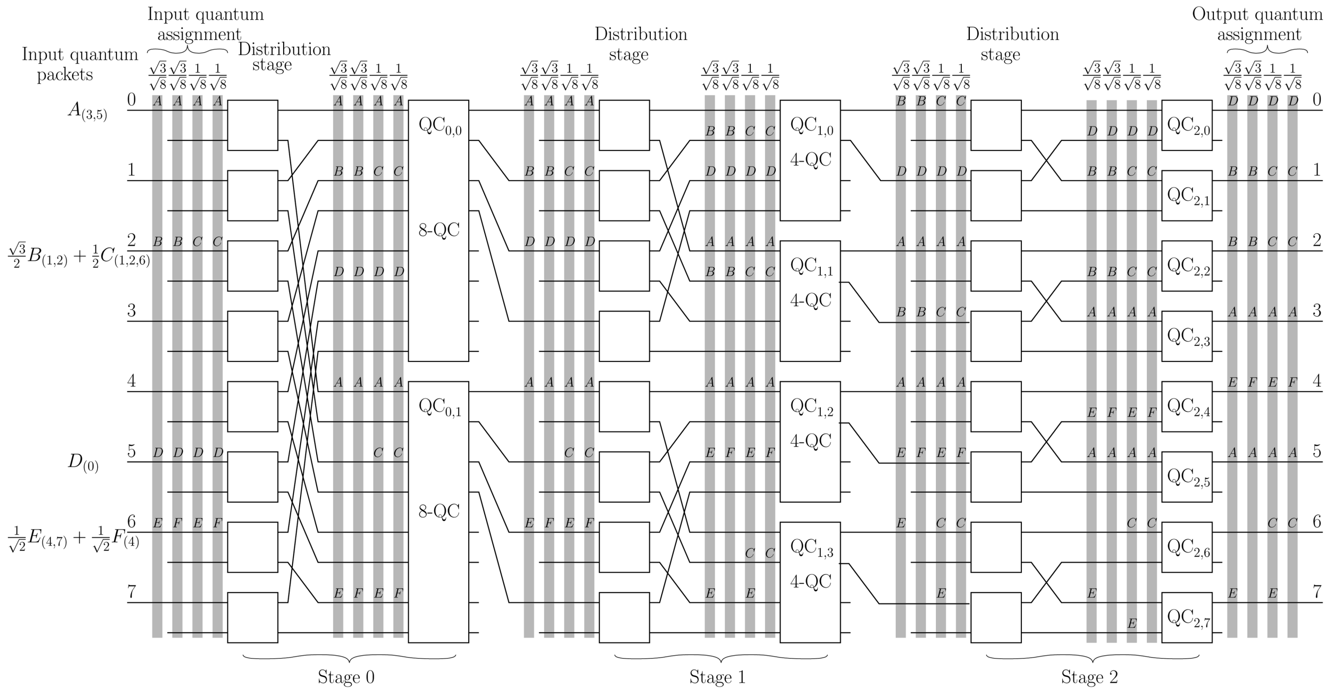


Fig. 8. 8-GQC realizing a noncontending quantum assignment.

data bits of the input packet are sent to the lower output by swapping them with a blank quantum packet initialized in state $|\bar{0}\rangle$, using the switch gate as shown. When b_{00} and b_{01} are 11, copies of the routing and data bits are created on both outputs of the copy node by using the multiqubit quantum copier gate. The remaining address bits are divided between the two outputs as described before.

The routing and data qubits on an output to which no packet is sent are set to $|0\rangle$. The top $n/2$ outputs of the two n-QCs are connected to two $n/2$-GQCs in the next stage, as shown in Fig. 6. The bottom $n/2$ outputs of the n-QCs are dropped. We now have the following theorem:

Theorem 4. *The network shown in Fig. 6 is an n-GQC.*

Proof. For a noncontending assignment pattern of size n, a maximum of $n/2$ packets are assigned to be routed to the top as well as to the bottom $n/2$ outputs of the n-GQC. Therefore, the output of the distribution stage is an assignment pattern of size $2n$, which is a concatenation of two n-assignment patterns at the inputs of the two n-QCs, each pattern having a maximum of $n/2$ packets with routing bits of 1. These packets are concentrated on the top $n/2$ outputs of the n-QCs. The bottom $n/2$ outputs of the n-QCs receive no packets. These outputs are in state $|\bar{0}\rangle$ and dropped. Therefore, all the auxiliary qubits used in the distribution stage are restored to their initial state. The two $n/2$-GQCs receive noncontending assignment patterns of size $n/2$ each, which are inductively realized. For $n = 2$, a copy node works as a 2-GQC. Consequently, the shown quantum network is an n-GQC by induction, where n is a power of 2. □

An expanded version of n-GQC is shown in Fig. 8 for $n = 8$. For simplicity, we have not expanded the quantum concentrators in the figure. We denote the 2^{p-k}-QCs in the kth concentrator stage of the n-GQC as $\mathrm{QC}_{k,0}, \mathrm{QC}_{k,1} \ldots, \mathrm{QC}_{k,2^{k+1}-1}$ from top to bottom, where $p = \log_2 n$ and $0 \leq k \leq p-1$. The set of outputs of n-GQC which can be reached from $\mathrm{QC}_{k,j}$ is represented as $\mathrm{O}_{k,j} = \{j2^{p-k-1}, \ldots, (j+1)2^{p-k-1} - 1\}$, where $0 \leq j \leq 2^{k+1} - 1$.

Fig. 8 also illustrates how a quantum multicast assignment is realized by the 8-GQC. The input quantum packets to the network are shown in the figure. We use subscripts to show the output addresses in a quantum packet. For example, input 2 has quantum packet $\frac{\sqrt{3}}{2}B_{(1,2)} + \frac{1}{2}C_{(1,2,6)}$, where packet B having fan-out set $\{1,2\}$ is to be routed with probability $3/4$ or packet C having fan-out set $\{1,2,6\}$ is to be routed with probability $1/4$. Inputs 1, 3, 4, and 7 do not have any packets. The corresponding quantum assignment is a superposition of four assignment patterns with coefficients $\sqrt{3}/\sqrt{8}$, $\sqrt{3}/\sqrt{8}$, $1/\sqrt{8}$, and $1/\sqrt{8}$, respectively, which are shown in the figure by gray vertical columns on the input side with coefficients on top. This quantum assignment is noncontending since all the four assignment patterns are noncontending. The figure illustrates how the four assignment patterns are realized by the 8-GQC, by showing the output quantum state at each stage of the network. On measurement, one of the four patterns shown at the output will be observed with probabilities $3/8$, $3/8$, $1/8$, and $1/8$, respectively. Therefore, packets D, A and A reach outputs 0, 3, and 5 with probability 1. Packet B is observed on outputs 1 and 2 with probability $3/4$. With probability $1/4$, packet C is observed on these two outputs. The probability of observing a packet on output 6 is $1/4$ and on output 7 is $1/2$.

We now count the number of auxiliary qubits used in the n-GQC. Every copy node requires $m_d + 1$ auxiliary qubits as a blank packet, where m_d is the number of bits in the data part of a packet. Consequently, the total number of auxiliary qubits used in the distribution stages of the network is

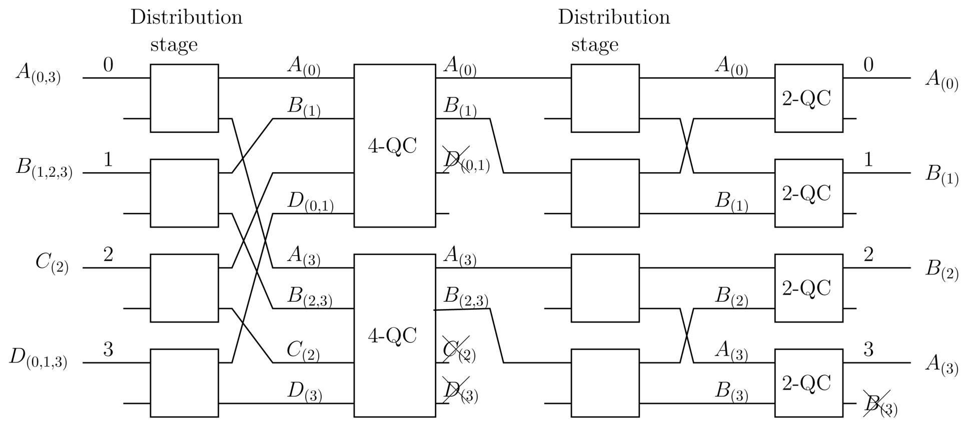


Fig. 9. Contending assignment through a 4-GQC.

$(m_d+1)n\log_2 n$. Also, since an n-QC uses $\frac{n}{2}\log_2 n$ auxiliary qubits, the total number of auxiliary qubits used in the concentration stages of the n-GQC is

$$\sum_{k=0}^{p-1} 2^{k+1}\frac{n/2^k}{2}\log_2(n/2^k) = \frac{n}{2}(\log_2 n + 1)\log_2 n. \qquad (18)$$

Therefore, the n-GQC uses $O(n\log^2 n)$ auxiliary qubits.

It is always possible to restore all of the auxiliary qubits used in a switching network to their initial state once the desired switching operation has been completed, i.e., the data fields of the input packets have reached their desired destinations. This can be accomplished by the inverse of the switching network, which uses routing, address, and auxiliary qubits. However, it is desirable that the auxiliary qubits are restored as early as possible so that they can be reused, and also their decoherence does not affect the switching operation [16], [17]. We adhere to this policy by restoring all of the auxiliary qubits used in the n-QC except the switching qubits. For noncontending assignment patterns, the switching qubits used in the quantum concentrators are the only auxiliary qubits that are not restored in an n-GQC. These can always be restored afterward using the inverse of n-GQC as described before.

So far, we have seen that an n-GQC realizes a noncontending assignment pattern without any blocking. In the next section, we describe how blocking occurs when a contending assignment pattern is routed through the n-GQC design given in this section.

6 Behavior of n-GQC for Contending Assignments

A substantial amount of recent research on multicast switching has been focused on developing scheduling algorithms, which aim to maximize the throughput of an input-queued multicast switch [21], [22], [23], [24]. A number of such algorithms require that the packets in a contending assignment are routed using a *fan-out splitting* policy, where a multicast packet can be sent to a subset of the outputs in its fan-out set. The rest of the fan-out set is realized in subsequent attempts. It is usually desired that the multicast switch has the capability to do such a fan-out splitting internally. It is also desired that the multicast switch be *work-conserving*, which means that if an output is in the fan-out set of at least one of the packets in a contending assignment, then it should not happen that this output does not receive any packet. In this section, we show that, due to internal blocking in case of contending assignment patterns, the n-GQC works in fan-out splitting fashion. However, it is not work-conserving.

Consider a contending assignment pattern to an n-GQC that has m classical packets, out of which m_u are addressed to only upper $n/2$ outputs, m_l are addressed to only lower $n/2$ outputs, and m_b are addressed both upper and lower $n/2$ outputs. Then, in the recursive construction of n-GQC, the upper n-QC receives m_u+m_b packets and lower n-QC receives m_l+m_b packets. Since a contending assignment can have more than $n/2$ packets addressed to the top $n/2$ outputs, m_u+m_b can be more than $n/2$. In this case, $m_u+m_b-n/2$ packets are blocked or dropped since only $n/2$ outputs of the n-QC are connected to the next stage. Similarly, if $m_l+m_b>n/2$, then $m_l+m_b-n/2$ packets are dropped at the lower n-QC. It is easily seen that for such blocking to occur at either of the n-QCs, it is necessary but not sufficient that $m>n/2$. This is because when $m_l=m_u=0$, m_b has to be greater than $n/2$ for blocking. Also, for a contending assignment, such a blocking will certainly occur at one of the quantum concentrators in the n-GQC. Thus, we see that the n-GQC realizes a subset of the set of output addresses for every packet in a contending multicast assignment pattern. This subset can also be empty, which means that some of the packets may be blocked entirely. Therefore, the n-GQC works in a fan-out splitting fashion.

This is illustrated by an example of a 4-GQC, as shown in Fig. 9. The contending assignment pattern has four packets A, B, C, and D with the fan-out sets shown in the figure. Packets C and D are blocked at the 4-QCs. The fan-out set of A is fully realized, but the fan-out set of B is partially realized, since a copy of B is blocked at the bottom 2-QC in the final stage.

Because of blocked packets on the unused outputs of the intermediate quantum concentrators, the n-GQC is not robust against decoherence on these outputs when realizing a contending quantum assignment. Such a decoherence would collapse the quantum assignment to only those assignment patterns that contain the observed packets on these outputs. This problem does not arise when realizing a noncontending quantum assignment. This is because the unused outputs of the intermediate concentrators are always empty for every assignment pattern in the assignment, as explained before in Section 5.2. Next, we define the work-conserving property for an n-GQC.

An assignment pattern $|\bar{P}'\rangle = |(r'_0, \bar{a}'_0, \bar{d}'_0), \ldots, (r'_{n-1}, \bar{a}'_{n-1}, \bar{d}'_{n-1})\rangle$ is called a *subpattern* of another assignment pattern $|\bar{P}\rangle = |(r_0, \bar{a}_0, \bar{d}_0), \ldots, (r_{n-1}, \bar{a}_{n-1}, \bar{d}_{n-1})\rangle$ if, for every i, $0 \leq i \leq n-1$, $r'_i = 1$ implies that $r_i = 1$, $\bar{d}'_i = \bar{d}_i$, and $F'_i \subseteq F_i$. A noncontending subpattern $|\bar{P}'\rangle$ of $|\bar{P}\rangle$ is said to be *maximal* if the fan-out sets $\cup_{i=0}^{n-1}\{F'_i : r'_i = 1\}$ and $\cup_{i=0}^{n-1}\{F_i : r_i = 1\}$ are equal.

An n-GQC is called *work-conserving* if it realizes a maximal noncontending subpattern of every contending multicast assignment pattern. It is seen that a contending multicast assignment pattern can have several maximal noncontending subpatterns and a work-conserving n-GQC realizes one of these subpatterns. We have the following for the n-GQC design given in this paper:

Theorem 5. *The n-GQC shown in Fig. 6 is not work-conserving.*

Proof. This is seen by considering a contending multicast assignment pattern having n classical packets in which the fan-out set of every packet on the top $n-1$ inputs is $\{1, \ldots, n-1\}$ and the fan-out set of the packet on the nth input is $\{0\}$. Copies of all the packets on top $n-1$ inputs and the packet on the nth input are routed to the upper n-QC. The packet with fan-out $\{0\}$ is concentrated to the nth output of the n-QC, and therefore, is dropped. Thus, no packet is routed to output 0, which means that the n-GQC is not work-conserving. □

As the example in Fig. 9 shows, there are some contending assignment patterns which do get routed in a work-conserving fashion by the n-GQC. For a given contending assignment, this happens when no fan-out loss occurs at any of the quantum concentrators in the n-GQC. In other words, at any quantum concentrator, the fan-out set union of the output packets on top half of outputs is equal to fan-out set union of input packets, where the unions are restricted to the outputs reachable by the quantum concentrator. Also, if we were able to design an n-QC which concentrates in such a way that it always maximizes the union of the fan-out sets of its top $n/2$ output packets when receiving more than $n/2$ input packets, then the n-GQC constructed using such n-QCs will be work-conserving. Designing such an n-QC is a potential approach toward the realization of a work-conserving n-GQC.

Finally, we observe that the results given in this section for contending assignments also hold for the classical version of generalized connector given in [1], i.e., this network works in a fan-out splitting manner and is not work-conserving. This is due to the fact that the classical generalized connector is functionally similar to the n-GQC and realizes a classical noncontending assignment in the same fashion as n-GQC without any blocking. In the next section, we compute the complexities of the n-QC and n-GQC.

7 Complexity Analysis

In this section, we compute the complexities of the n-QC and the n-GQC in terms of the total number of quantum gates and the gate-level depth. Representing these complexities for the n-QC as $C_{qc}(n)$ and $D_{qc}(n)$, respectively, we have

$$C_{qc}(n) = 2C_{qc}(n/2) + C_{split}(n), \tag{19}$$

$$D_{qc}(n) = D_{qc}(n/2) + D_{split}(n), \tag{20}$$

where $C_{split}(n)$ and $D_{split}(n)$ are the corresponding complexities for an n-quantum odd-even splitter. Since an n-QB has $n-1$ controlled-not gates, and each splitter switch has a constant number of controlled-not gates, $C_{split}(n)$ is $O(n)$. Also, the depth of the n-quantum odd-even splitter is mainly determined by the depth of n-QB, which is equal to $\log_2 n$. Consequently, $D_{split}(n)$ is $O(\log n)$. Thus, we have $C_{qc}(n) = O(n \log n)$ and $D_{qc}(n) = O(\log^2 n)$.

For the n-GQC, the complexities in terms of number of quantum gates $C_{gqc}(n)$, and gate-level depth $D_{gqc}(n)$, are given by

$$C_{gqc}(n) = 2C_{gqc}(n/2) + 2C_{qc}(n) + C_{dist}(n), \tag{21}$$

$$D_{gqc}(n) = D_{gqc}(n/2) + D_{qc}(n) + D_{dist}(n), \tag{22}$$

where $C_{dist}(n)$ and $D_{dist}(n)$ are the corresponding costs for the distribution stage. It is easy to verify that $C_{dist}(n) = O(n)$ because each copy node uses a constant number of gates, and $D_{dist}(n) = O(1)$. Thus, we have $C_{gqc}(n) = O(n \log^2 n)$ and $D_{gqc}(n) = O(\log^3 n)$.

8 Conclusion and Future Work

We have presented an $n \times n$ multistage quantum switching network called quantum generalized connector (n-GQC) that can realize quantum multicast assignments. The quantum packets at each input consist of a number of classical multicast packets in a probabilistic quantum superposition. We showed that quantum assignments can be expressed as superpositions of multicast assignment patterns, where each assignment pattern is a sequence of classical packets across the inputs of the network. All the assignment patterns are simultaneously realized by the n-GQC due to quantum parallelism. All the packets in a noncontending assignment pattern are routed to their desired outputs. However, the n-GQC is not work-conserving when the input assignment pattern is contending and it realizes a subpattern of such an assignment pattern due to internal blocking.

The main motivation behind the design of n-GQC is that it has inherent quantum parallelism that provides high throughput by definition, while the packets are en route to their destinations. To give an example, consider unicast assignments, which are a subset of multicast traffic. Unicast switches with nearly 100 percent throughput have been reported in the literature [25], [26], but such switches require complex scheduling algorithms. To avoid using such algorithms, one can potentially employ a multilayer switch to route a fixed number of packets from the head of each input queue and an n-GQC would be a natural replacement for such a switch in the quantum domain.

When routing a noncontending quantum assignment in which different assignment patterns contain packets addressed to same outputs, the n-GQC realizes all the assignment patterns in parallel and creates a superposition of all of these packets at those outputs. Although a computational basis measurement collapses the output state to one of the patterns at the outputs, a better measurement scheme can be used to possibly increase the throughput. An investigation of such measurement techniques will be deferred to another place.

In this paper, we have mainly focused on multicasting classical data using the n-GQC. However, the n-GQC can potentially be extended to multicast arbitrary quantum states. Even though the no-cloning theorem makes such an operation impossible, approximate quantum multicasting is possible due to the recent advances in quantum copying. By using advanced quantum copiers such as Bužek and Hillery's universal quantum copying machine [19], [27] in the distribution stages of n-GQC, this network can be used for approximate multicasting of arbitrary quantum states. However, as in the case of quantum copying, there is going to be a trade-off between the amount of entanglement in the output copies and the quality of multicasting. It would be interesting to study how well the n-GQC can multicast arbitrary quantum states and how the parameters of copy nodes affect the above-mentioned trade-off.

The n-GQC can be used to unicast arbitrary quantum states between its n inputs and outputs provided that there is no output contention among the input quantum states. This is due the fact that no copying would be needed at any of the distribution stages. In the course of designing the n-GQC, we have also designed an n-quantum concentrator that can route arbitrary quantum states on any m of its inputs to the top m outputs. This network can potentially be used in other applications as well, apart from its critical role in the n-GQC.

REFERENCES

[1] C.Y. Lee and A.Y. Oruç, "Design of Efficient and Easily Routable Generalized Connectors," *IEEE Trans. Comm.*, vol. 43, nos. 2-4, pp. 646-650, Feb.-Apr. 1995.

[2] P.W. Shor, "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer," *SIAM Rev.*, vol. 41, no. 2, pp. 303-332, June 1999.

[3] L.K. Grover, "A Fast Quantum Mechanical Algorithm for Database Search," *Proc. 28th Ann. ACM Symp. Theory of Computing*, pp. 212-219, May 1996.

[4] M. Oskin, F.T. Chong, I.L. Chuang, and J. Kubiatowicz, "Building Quantum Wires: The Long and the Short of It," *Proc. 30th Ann. Int'l Symp. Computer Architecture*, pp. 374-385, 2003.

[5] C. Moore and M. Nilsson, "Parallel Quantum Computation and Quantum Codes," *SIAM J. Computing*, vol. 31, no. 3, pp. 799-815, 2001.

[6] I.M. Tsai and S.Y. Kuo, "Digital Switching in the Quantum Domain," *IEEE Trans. Nanotechnology*, vol. 1, no. 3, pp. 154-164, Sept. 2002.

[7] M.K. Shukla, R. Ratan, and A.Y. Oruç, "A Quantum Self-Routing Packet Switch," *Proc. 38th Ann. Conf. Information Sciences and Systems (CISS '04)*, pp. 484-489, Mar. 2004.

[8] M.K. Shukla, R. Ratan, and A.Y. Oruç, "The Quantum Baseline Network," *Proc. 39th Ann. Conf. Information Sciences and Systems (CISS '05)*, Mar. 2005.

[9] S.T. Cheng and C.Y. Wang, "Quantum Switching and Quantum Merge Sorting," *IEEE Trans. Circuits and Systems*, vol. 53, no. 2, pp. 316-325, Feb. 2006.

[10] C.C. Sue, "An Enhanced Universal n x n Fully Nonblocking Quantum Switch," *IEEE Trans. Computers*, vol. 58, no. 2, pp. 238-250, Feb. 2009.

[11] C.Y. Lee and A.Y. Oruç, "Design of Efficient and Easily Routable Generalized Connectors," Technical Report UMIACS-TR-92-22,CS-TR-2846, Inst. for Advanced Computer Studies, Univ. of Maryland, College Park, 1992.

[12] J.P. Ofman, "A Universal Automaton," *Trans. Moscow Math. Soc.*, vol. 14, pp. 200-215, 1965.

[13] C.D. Thompson, "Generalized Connection Networks for Parallel Processor Intercommunication," *IEEE Trans. Computers*, vol. 27, no. 12, pp. 1119-1125, Dec. 1978.

[14] Y. Yang and G.M. Masson, "Nonblocking Broadcast Switching Networks," *IEEE Trans. Computers*, vol. 40, no. 9, pp. 1005-1015, Sept. 1991.

[15] D. Nassimi and S. Sahni, "Parallel Permutation and Sorting Algorithms and a New Generalized Connection Network," *J. ACM*, vol. 29, pp. 642-667, July 1982.

[16] M.A. Nielsen and I.L. Chuang, *Quantum Computation and Quantum Information.* Cambridge Univ. Press, Sept. 2000.

[17] J. Preskill, "Physics 229: Advanced Mathematical Methods of Physics—Quantum Computation and Information," http://www.theory.caltech.edu/people/preskill/ph229, 1998.

[18] W. Wootters and W. Zurek, "A Single Quantum Cannot be Cloned," *Nature*, vol. 299, pp. 802-803, 1982.

[19] V. Bužek and M. Hillery, "Quantum Copying: Beyond the No-Cloning Theorem," *Physical Rev. A*, vol. 54, no. 3, pp. 1844-1852, Sept. 1996.

[20] R. Ratan, M.K. Shukla, and A.Y. Oruç, "Quantum Switching Networks with Classical Routing," *Proc. 41st Ann. Conf. Information Sciences and Systems (CISS '07)*, pp. 789-793, Mar. 2007.

[21] N. McKeown and B. Prabhakar, "Scheduling Multicast Cells in an Input Queued Switch," *Proc. IEEE INFOCOM*, vol. 1, pp. 261-278, Mar. 1996.

[22] B. Prabhakar, N. McKeown, and R. Ahuja, "Multicast Scheduling for Input-Queued Switches," *IEEE J. Selected Areas Comm.*, vol. 15, no. 5, pp. 855-866, June 1997.

[23] M.A. Marsan, A. Bianco, P. Giaccone, E. Leonardi, and F. Neri, "Multicast Traffic in Input-Queued Switches: Optimal Scheduling and Maximum Throughput," *IEEE/ACM Trans. Networking*, vol. 11, no. 3, pp. 465-477, June 2003.

[24] D. Pan and Y. Yang, "FIFO-Based Multicast Scheduling Algorithm for Virtual Output Queued Packet Switches," *IEEE Trans. Computers*, vol. 54, no. 10, pp. 1283-1297, Oct. 2005.

[25] N. Mckeown, A. Mekkittikul, V. Anantharam, and J. Walrand, "Achieving 100 Percent Throughput in an Input-Queued Switch," *IEEE Trans. Comm.*, vol. 47, no. 8, pp. 1260-1267, Aug. 1999.

[26] D. Shah, P. Giaccone, and B. Prabhakar, "Efficient Randomized Algorithms for Input-Queued Switch Scheduling," *IEEE Micro*, vol. 22, no. 1, pp. 10-18, Jan./Feb. 2002.

[27] V. Bužek, S.L. Braunstein, M. Hillery, and D. Bruss, "Quantum Copying: A Network," *Physical Rev. A*, vol. 56, no. 5, pp. 3446-3452, Nov. 1997.

Manish Kumar Shukla received the BTech degree in electrical engineering from the Indian Institute of Technology, Kanpur, in 1999, and the MS degree in electrical and computer engineering from the University of Maryland, College Park, in 2003, where he is currently working toward the PhD degree. From 1999 to 2001, he was a member of the Technical Staff at the Centre for Development of Telematics in India. From 2001 to 2008, he worked as a research and teaching assistant in the Department of Electrical Engineering, University of Maryland, College Park. His research interests include parallel and distributed systems, interconnection networks, and quantum information science.

A. Yavuz Oruç received the BSc degree in electrical engineering from the Middle East Technical University in 1976, the MSc degree in electronics from the University of Wales in 1978, and the PhD degree in electrical engineering from Syracuse University in 1983. He has done research in parallel processing and interconnection network theory. His more recent research has focused on quantum packet switching, an emerging field of research in quantum information processing. He was the director of the Computer Systems Architecture Program at the US National Science Foundation from 2000 to 2002, and more recently, served as a senior advisor to the President of the Scientific Research and Technological Council of Turkey. He has been a full professor at the University of Maryland, College Park, since 1995. He previously held faculty positions at RPI and Bilkent University. He was an associate editor for the *IEEE Transactions on Parallel and Distributed Systems* between 2003 and 2007.