Quantum Switching Networks with Classical Routing

Rahul Ratan, Manish Kumar Shukla, A. Yavuz Oruç
Department of Electrical and Computer Engineering
University of Maryland, College Park, MD 20742
Email: [rahulr, manishk, yavuz]@umd.edu

***Abstract*— Flexible distribution of data in the form of quantum bits (qubits) amongst spatially separated entities is an essential component of envisioned scalable quantum computing architectures. Since qubits cannot be copied, this operation of moving qubits can be relatively costly in terms of resources. Moreover, implementation of quantum gates requires precise and extensive classical control and computation too. Accordingly, we consider the problem of dynamically permuting groups of qubits, i.e., qubit packets using reconfigurable quantum switches in which routing information is calculated classically as a possible way to reduce this cost. We design a 2×2 switch based on the controlled-swap quantum gate and show that if switch settings are determined using efficient classical algorithms, then quantum switches can be mapped onto classical non-blocking interconnection switch topologies with low cost by using this switch. Specific quantum switch designs for the optimal Beneš network and the planar Spanke-Beneš network are given.**

I. Introduction

Quantum computers can solve many interesting problems with substantial advantages over known algorithms with traditional computation. the most well known results include Shor's algorithm for factoring a product of large primes in polynomial time [1] and Grover's algorithm for a $O(\sqrt{N})$ search on an unstructured database [2]. Other algorithms include adiabatic solution of optimization problems [3], Pell's equation [4] and Gauss Sums [5]. As a result there has been a tremendous interest in investigating issues related to quantum computing. Any realistic scenario for the future of quantum computing involves spatially distributed quantum devices which interact with each other. This spatial aspect introduces as a critical requirement the need for quantum wires which transport qubits. Transferring qubits and hence building quantum wires is not trivial as, in general, qubits can not be copied, which is a consequence of the quantum no-cloning theorem. Each qubit must be physically transported from the source to the destination, a restriction causing great constraints on quantum data distribution. In [6], [7] a proposal is given for building quantum wires using solid-state technologies: short wires which transfer information by swapping qubits and long wires which use quantum teleportation. The complex nature of such quantum wires means that the $O(N^2)$ wires needed to fully connect N spatially distributed quantum devices can incur a huge cost in implementing quantum systems.

Quantum non-blocking switches can reduce this cost by allowing full connectivity between N devices with only $O(N)$ quantum wires. The idea is that each device inputs qubits along its own quantum wire to the switch which then switches them to the required destination. Tsai and Kuo [8] gave a method to permute individual qubits (not qubit packets) using *fixed* or *non-reconfigurable* quantum circuits. They decompose a given permutation into disjoint cycles and these cycles into transpositions or swaps to generate a quantum circuit which realizes only that particular permutation. Thus, a new circuit needs to be designed for each permutation using this method. Reconfigurable or controllable quantum switches can avoid this problem. In addition to reducing wire count, reconfigurable quantum switches can form fundamental parts of the quantum data distribution system in envisioned architectures for scalable quantum computing. For example, in the Quantum Logic Array (QLA) microarchitecture proposed in [9] the high-level quantum computer structure consists of logical qubits connected with a programmable communication network where integrated switch islands are used to redirect quantum data from nearby logical qubits.

The authors of this paper were the first to propose reconfigurable quantum switches to permute qubit packets [10], [11]. Our aim in these designs was not only to permute qubit packets but also to explicitly utilize quantum superposition to reduce the problem of blocking in packet switches. As a result we used blocking switch topologies (which do not route all permutations) and these switches cannot realize all permutations. Cheng and Wang have proposed a reconfigurable Batcher Sorter based fully non-blocking quantum switch made up of $\Theta(N \log^2 N)$ 2×2 sorters [12] with routing done in the quantum domain. This switch permutes single qubits only, not qubit packets. Each sorter potentially requires $\Theta(\log N)$ elementary quantum gates to implement which leads to a total gate count of $\Theta(N \log^3 N)$. This is a high cost compared to many classical fully non-blocking switching networks which use $\Theta(N \log N)$ gates. Serial routing algorithms for such networks, like the "looping" algorithm have $O(N \log N)$ complexity and hence are only suitable for offline routing [13]. Faster parallel algorithms for online routing take $O(\log^2 N)$ time but also need N fully connected processors [14], [15]. For permuting qubits dynamically, keeping the route calculation in the classical domain while implementing the switch using quantum circuits based on such topologies gives us the best of both worlds, i.e., efficient routing and a low cost qubit switching network. This is a reasonable choice as virtually all proposed scalable implementations of quantum computing namely trapped ions[16] and solid-state silicon based systems [17] require significant and precise classical control to implement quantum gates and qubit manipulations. The ready availability of classical computation resources implies that classical route computation should fit nicely in this framework.

In this paper we show that to dynamically permute qubits in any arbitrary permutation we can use any classical non-blocking switching network topology with the internal switches replaced by reconfigurable quantum switches. Note that unlike in our previous work [10], [11] we are not explicitly creating superpositions in the quantum switches to reduce blocking as we are using non-blocking topologies which can route all the $N!$ permutations between N inputs and N outputs. The switch settings can be determined classically using any well known efficient routing algorithm for that topology. By doing so we can take advantage of the vast number of "classical" non-blocking switching fabrics which have already been investigated and found to have many, e.g., the Beneš network which uses $O(N \log N)$ switches and can be routed in $O(N \log N)$ serial time [13]. Specifically, we describe the operation of a 2×2 quantum switch implemented using a controlled-swap gate. The arrangement of such switches to form a non-blocking network is illustrated via the examples of the $N \times N$ Beneš network and the $N \times N$ Spanke-Beneš network.

The remainder of the paper is organized as follows. In Section II we introduce some basic concepts related to quantum computing needed to understand the subsequent material. Section III shows the working of our controlled-swap gate based 2×2 quantum switch. In Section IV we give the design and operation of $N \times N$ Beneš and Spanke networks constructed using such switches. Section V concludes the paper.

II. Preliminaries

We give a brief overview of the basic terminology and building blocks of quantum computation. The purpose of this section is to introduce the terms and notation used in the subsequent parts of the paper. An in-depth treatment of the same can be found in [18].

A. Qubits and Quantum States

A quantum bit or qubit is the quantum analogue of the classical bit. A qubit, ψ, is described by the equation $|\psi\rangle = a_0|0\rangle + a_1|1\rangle$ where the probability amplitudes a_0 and a_1 are complex numbers whose modulus squared sums to one, i.e., $|a_0|^2 + |a_1|^2 = 1$. Here the "classical" states $|0\rangle$ and $|1\rangle$ form the computational basis states. In the case when both a_0 and a_1 are non-zero, qubit $|\psi\rangle$ is said to be in a superposition of the basis states. Some examples for valid qubit states are $|0\rangle$, $|1\rangle$, $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|0\rangle$ and $\frac{-i}{2}|0\rangle + \frac{\sqrt{3}}{2}|0\rangle$. Larger quantum systems can be composed from multiple qubits, e.g., $|01\rangle$ or $\frac{1}{2}|01\rangle + \frac{1}{\sqrt{2}}|10\rangle - \frac{1}{2}|11\rangle$. Thus, in general, an n-qubit system can be represented as $\sum_{x=0}^{2^n-1} a_x|\mathbf{x}\rangle$ where $\mathbf{x}$ are n-bit binary strings representing the basis states and a_x are the corresponding complex probability amplitudes.

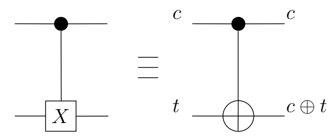


Fig. 1. The CNOT gate.

The composition of qubits into a multi-qubit representation is done using the tensor product operator $\otimes$, $|\mathbf{x}\rangle \otimes |\mathbf{y}\rangle = \sum_x a_x|\mathbf{x}\rangle \otimes \sum_y a_y|\mathbf{y}\rangle = \sum_{x,y} a_x a_y|\mathbf{x} \otimes \mathbf{y}\rangle$ where $\mathbf{x} \otimes \mathbf{y}$ is the string formed by concatenating $\mathbf{x}$ and $\mathbf{y}$.

In addition to superposition quantum systems exhibit the unique property of entanglement which has no classical analogue. Entanglement occurs when a multi-qubit state can not be expressed as a composition (tensor product) of smaller qubit states, e.g., there exist no single qubit states $|\psi_A\rangle$ and $|\psi_B\rangle$ such that the two qubit state $|\psi\rangle = \frac{1}{\sqrt{2}}|01\rangle + \frac{1}{\sqrt{2}}|10\rangle$ can be expressed as the composite state $|\psi_A\rangle \otimes |\psi_B\rangle$. Thus, the state of the larger quantum system can not be expressed as a product of its parts.

Although a quantum system can exist in a superposition of orthogonal states, only one of those states can be observed or measured. After measurement, the quantum system is no longer in a superposition, the quantum state collapses into the measured state and the probability amplitude of all the other states goes to zero. The probability of measuring a particular state is given by the square of the modulus of its probability amplitude, e.g., when $|\psi\rangle = \frac{1}{\sqrt{2}}|10\rangle + \frac{1}{\sqrt{2}}|01\rangle$ is measured, the outcome is either 10 or 01 with equal probability, 00 or 11 is never measured. Also, if a subset of qubits are measured then the rest of the qubits are left in a state consistent with that measurement.

B. Quantum Gates

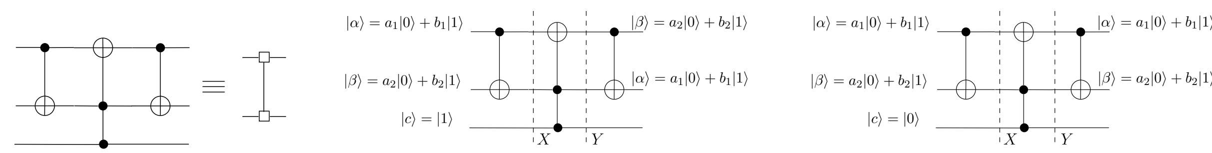
Quantum gates are used to manipulate qubits just as classical bits are manipulated by gates such as NOT, XOR and AND. Single qubit gates include the X gate (similar to NOT), the Z gate (phase flip) and the H or Hadamard gate. The mappings performed by these gates are:

$$a|0\rangle + b|1\rangle \xrightarrow{X} b|0\rangle + a|1\rangle \tag{1}$$
$$a|0\rangle + b|1\rangle \xrightarrow{Z} a|0\rangle - b|1\rangle \tag{2}$$
$$a|0\rangle + b|1\rangle \xrightarrow{H} \frac{a+b}{\sqrt{2}}|0\rangle + \frac{a-b}{\sqrt{2}}|1\rangle \tag{3}$$

Note that H gate does the conversions $|0\rangle \rightarrow \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ and $|1\rangle \rightarrow \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$. Two qubit gates include the Controlled-NOT (CNOT) gate shown in figure 1. This gate inverts the target qubit, t, for states in which the source or control qubit, c is one. This mapping can be written as:

$$|c\rangle|t\rangle \xrightarrow{\mathbf{C-NOT}} |c\rangle|c \oplus t\rangle \tag{4}$$

The CNOT gate is the analogue of the classical XOR gate. Such gates can be extended to quantum gates with multiple control qubits. In quantum circuits drawn using such gates the



(a) Schematic for the switch gate.

(b) Cross state, $|c\rangle = |1\rangle$.

(c) Through state, $|c\rangle = |0\rangle$.

Fig. 2. Quantum Switch Gate.

horizontal lines represent qubits which evolve in time from left to right.

III. 2×2 Quantum Switch

The basic building block of the proposed reconfigurable quantum switch fabrics is the controlled-2×2 quantum swap gate (also known as the Fredkin gate). This gate is shown in figure 2. A circuit for this gate using two Controlled-NOT (CNOT) and one Toffoli (Controlled-Controlled NOT) gate is given in figure 2(a) along with the equivalent representation we will use later in the paper.

We give a detailed explanation of the operation of this gate below.

Let the input to the switch in figure 2 be

$$|\alpha\rangle \otimes |\beta\rangle \tag{5}$$
$$= (a_1|0\rangle + b_1|1\rangle) \otimes (a_2|0\rangle + b_2|1\rangle) \tag{6}$$
$$= a_1a_2|00\rangle + a_1b_2|01\rangle + b_1a_2|10\rangle + b_1b_2|11\rangle \tag{7}$$

In the first Controlled-NOT (CNOT) gate, the second qubit is flipped if the first qubit is $|1\rangle$. Thus, the qubit state at X is:

$$a_1a_2|00\rangle + a_1b_2|01\rangle + b_1b_2|10\rangle + b_1a_2|11\rangle \tag{8}$$

Note that the last two coefficients have been swapped and the resulting state can no longer be expressed as a tensor product of two independent qubit states, i.e., the two qubits are now entangled. Now, if the control qubit, c, is equal to $|1\rangle$, the second gate (Toffoli gate) fires and the resulting state at Y in figure 2(b) is

$$a_1a_2|00\rangle + a_1b_2|11\rangle + b_1b_2|10\rangle + b_1a_2|01\rangle \tag{9}$$

In this state the two qubits are still entangled. After passing through the final CNOT gate, the output state is given by

$$a_1a_2|00\rangle + b_1a_2|01\rangle + a_1b_2|10\rangle + b_1b_2|11\rangle \tag{10}$$
$$= (a_2|0\rangle + b_2|1\rangle) \otimes (a_1|0\rangle + b_1|1\rangle) \tag{11}$$
$$= |\beta\rangle \otimes |\alpha\rangle \tag{12}$$

Thus, we see that by setting the control qubit we can swap the input qubits, or equivalently put the switch in a "cross" state. When the control qubit, c, is equal to $|0\rangle$, then the second gate does not fire and the state at the output (as seen in figure 2(c)) is

$$a_1a_2|00\rangle + a_1b_2|01\rangle + b_1a_2|10\rangle + b_1b_2|11\rangle \tag{13}$$
$$= (a_1|0\rangle + b_1|1\rangle) \otimes (a_2|0\rangle + b_2|1\rangle) \tag{14}$$
$$= |\alpha\rangle \otimes |\beta\rangle \tag{15}$$

which is the same as the input state, thus we see that the switch is in a "through" or "pass" state when c is $|0\rangle$. Therefore, we have configurable 2×2 quantum switch which can be used to pass or swap two qubits depending on the setting of the control qubit.

Note that switching qubits involves only the exchange of the qubit co-efficients a_1, b_1 and a_2, b_2. Unlike conventional or "classical" switching with normal bits, there is no physical movement of qubits between the terminals in the switching operation, we merely exchange the information between the two qubits to be switched. Also, one qubit c can be used to control many such switches in parallel, hence, we can switch groups or packets of qubits. Though the swap gate can be composed from CNOT gates as shown above, it is often available natively for a given technology, leading to savings in the gate count for quantum switches.

IV. $N \times N$ Quantum Switches with Classical Routing

We describe the structure and routing for the Beneš and Spanke-Beneš switches in this section.

A. Quantum Beneš Switch

A $N \times N$ Beneš network ($N = 2^n$) is defined recursively as shown in figure 3. It consists of $2\log N - 1$ stages of 2×2 switches, with each stage having $N/2$ such switches. This can be seen more clearly in the fully expanded 8×8 Beneš network shown in figure 4(a). The smaller 4×4 and 2×2 Beneš networks are enclosed by dotted boxes in figure 4(a). The corresponding Beneš switching fabric constructed using quantum switches is shown in 4(b). The quantum switch is drawn in the traditional quantum circuit representation without any crossovers or rearrangeable lines which represent qubits. It can be easily verified that the connection pattern is same in both the cases. The Beneš network is known to be rearrangeably non-blocking, i.e., for any permutation, say

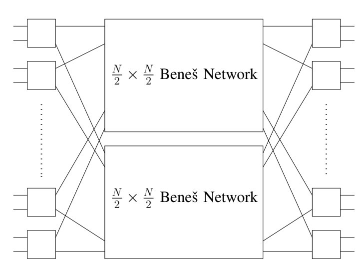
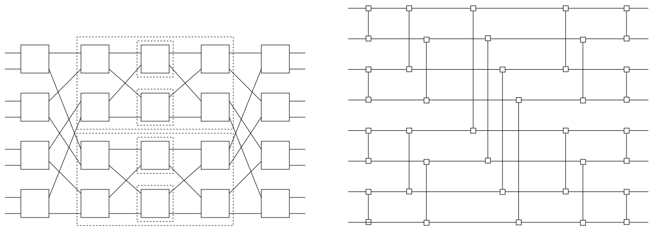


Fig. 3. The $N \times N$ Beneš Network.



(a) The classical 8×8 Beneš network (b) The quantum 8×8 Beneš network

Fig. 4. 8×8 Classical and Quantum Beneš Networks.

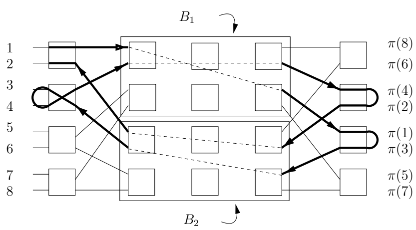
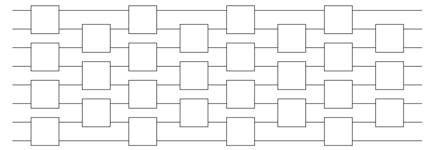


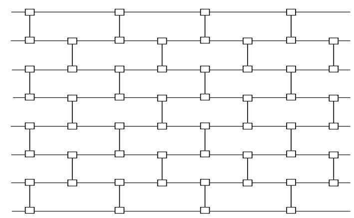
Fig. 5. Looping algorithm on a Beneš network

π, there exists a setting of the 2×2 switches such that π can be realized by the network. Many serial as well as parallel routing algorithms are known for this network, the simplest of which is the so-called "looping algorithm" [13]. The main idea behind this algorithm is shown in figure 5. Assume that we have to realize the permutation map π, i.e., input i is to be routed to output $\pi(i)$. Any two inputs or any two outputs connected to the same 2×2 switch are called buddies. Let the upper $N/2 \times N/2$ Beneš network be B_1 and the lower $N/2 \times N/2$ Beneš network be B_2. In figure 5 assume that we start the routing from input 1. The switch connected to input 1 is arbitrarily set to through state so that 1 gets routed to B_1. Follow the output of B_1 going to the switch connected to $\pi(1)$ to determine the state of this switch (through in this case). Loop to the buddy output for $\pi(1)$: $\pi(3)$ and follow the route to B_2 and continue the loop on the input side till we reach input 1 again. The loop in this case is $1 \rightarrow \pi(1) \rightarrow \pi(3) \rightarrow 3 \rightarrow 4 \rightarrow \pi(4) \rightarrow \pi(2) \rightarrow 2$ and the corresponding switch settings in order are through-through-cross-through. Now we can pick any other unrouted input and follow the same process again till all switches in the first and last stages are set. We see that the dotted lines in B_1 and B_2 form 2 new permutations. The same process can be repeated on them recursively till all the switch settings for the whole network are found. The complexity of this algorithm is clearly $O(N \log N)$.

Once the switch settings are known the corresponding control qubits for the quantum switches can be set to either $|1\rangle$ or $|0\rangle$ depending on whether the switch is in "cross" or "through" state respectively. In scalable implementations of quantum computers [17], [16] often it is easier to form gates by making only adjacent or nearly adjacent qubits interact with each other. Although the Beneš network has asymptotically optimal number of switches ($O(N \log N)$), the internal paths



(a) Classical Spanke-Beneš Network.



(b) Quantum Spanke-Beneš Network.

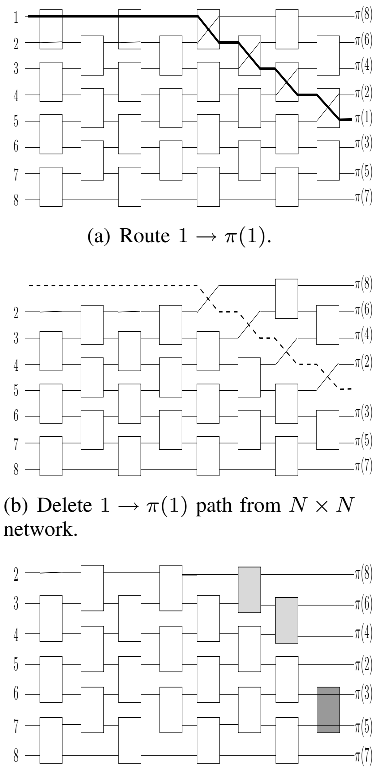
Fig. 6. 8×8 Spanke-Beneš Network.

cross each other which implies that non-adjacent qubits need to interact with each other leading to greater overhead associated with transporting these qubits. Thus, planar switching networks which do not have any crossing of internal lines are suitable candidates for making switches in such technologies. One such switch, the Spanke-Beneš switch, is described below.

B. Quantum Spanke-Beneš Switch

An $N \times N$ Spanke-Beneš network is a planar network containing N stages of 2×2 switches as seen in figure 6(a). This network was originally proposed for optical switches with the aim of minimizing crosstalk due to crossing of waveguides or optical fibers which form the "optical wires" in the switch [19]. Note that any switch only connects adjacent input and output lines and hence, the network is planar and has no crossings of wires. Also note that the corresponding quantum switching network shown in figure 6(b) has exactly the same structure due to the planar nature of this network. This network is also non-blocking, i.e., it can route all $N!$ permutations but it has a switch count of $N(N-1)/2$ which was shown to be the minimum possible for a planar permutation network in [19]. Spanke-Beneš networks are a special case of the more general cellular arrays described first by Kautz et al [20]. The routing cost on the Spanke-Beneš network is $O(N)$. The routing algorithm is based on a iterative procedure of routing inputs in sequence from input 1 to input N. After an input is routed the cells or switches set along its path are removed to get a $N-1 \times N-1$ network which is then routed using the same process. The algorithm is shown step-by-step in figure 7. Briefly, the steps (from [19]) are:

(1) Let the permutation map be given by π where input i is mapped to output $\pi(i)$ $i = 1, \cdots, N$.
(2) Route the $1 \rightarrow \pi(1)$, signal by continuing straight across the planar network (i.e., set switches in the through state) until the last possible stage. With the network oriented as in figure 6(a), this is the $N - \lceil \pi(1)/2 \rceil$ stage.



(a) Route $1 \to \pi(1)$.

(b) Delete $1 \to \pi(1)$ path from $N \times N$ network.

(c) Condense to $N-1 \times N-1$ network.

Fig. 7. Routing on 8×8 Spanke-Beneš Network [19].

(3) Cut downward through the rest of the network (i.e., set switches in the cross state) starting with stage $N - \lceil \pi(1)/2 \rceil$ and continuing through stage $N-1$, to reach the switch of $\pi(1)$.
(4) Set the switch in stage N as required to achieve desired output. Figure 7(a) depicts the $N \times N$ network with signal 1 routed.
(5) Delete the completed path and associated switches from the network (figure 7(b)). (6) Move the separated upper right corner triangle down and left to reconstruct the planar topology. Set any switches remaining in stage N below output $\pi(1)$, to the through state (figure 7(c)).
(7) What remains is an $N-1 \times N-1$ planar network. This can be routed by recursively applying steps 1-6 and reducing switch dimensions by 1 in each iteration.
By initializing the control qubits for the switches in cross state to $|1\rangle$ and those in through state to $|0\rangle$ we can realize any permutation map π.

V. Conclusion

Movement or communication of qubits amongst spatially distributed entities is a fundamental feature of scalable quantum computing architectures. Since qubits cannot be copied, this operation of moving sets of qubits can be relatively costly in terms of resources. Moreover, implementation of quantum gates requires precise and extensive classical control and computation too. Hence, we have proposed reconfigurable qubit switches based on well known switching network topologies which have quantum switching planes and in which routing information is calculated classically as a possible way to reduce this cost. A specific design for the asymptotically optimal Beneš network was given. In many quantum computing technologies it is easier to make closer or adjacent qubits interact with each other, hence we also gave the design of a quantum Spanke-Beneš network which by virtue of its planar topology is easier to implement in such instances.

References

[1] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Review*, vol. 41, no. 2, pp. 303–332, Jun 1999.

[2] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, May 1996, pp. 212–219.

[3] A. M. Childs, E. Farhi, and J. Preskill, "Robustness of adiabatic quantum computation," *Physical Review A*, vol. 65, no. 1, Dec 2001.

[4] S. Hallgren, "Polynomial-time quantum algorithms for pell's equation and the principal ideal problem," in *STOC '02: Proceedings of the thiry-fourth annual ACM Symposium on Theory of Computing*. New York, NY, USA: ACM Press, 2002, pp. 653–658.

[5] W. van Dam and G. Seroussi. (2002) Efficient quantum algorithms for estimating gauss sums. [Online]. Available: http://www.citebase.org/abstract?id=oai:arXiv.org:quant-ph/0207131

[6] M. Oskin, F. T. Chong, I. L. Chuang, and J. Kubiatowicz, "Building quantum wires: the long and the short of it," in *Proceedings of 30th Annual International Symposium on Computer Architecture, 2003.*, June 2003, pp. 374–385.

[7] D. Copsey, M. Oskin, F. Impens, T. Metodiev, A. Cross, F. T. Chong, I. L. Chuang, and J. Kubiatowicz, "Toward a scalable, silicon-based quantum computing architecture," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 9, no. 6, pp. 1552–1569, Nov-Dec 2003.

[8] I. M. Tsai and S.-Y. Kuo, "Digital switching in the quantum domain," *IEEE Transactions on Nanotechnology*, vol. 1, no. 3, pp. 154–164, 2002.

[9] T. S. Metodi, D. D. Thaker, A. W. Cross, F. T. Chong, and I. L. Chuang, "A quantum logic array microarchitecture: scalable quantum data movement and computation," in *Proceedings of 38th Annual IEEE/ACM International Symposium on Microarchitecture, 2005. MICRO-38.*, 2005, pp. 12 pp.–.

[10] M. K. Shukla, R. Ratan, and A. Y. Oruç, "A quantum self-routing packet switch," in *Proceedings of the 38th Annual Conference on Information Sciences and Systems CISS'04*, Princeton, NJ, USA, March 2004, pp. 484–489.

[11] ——, "The quantum baseline network," in *Proceedings of the 39th Annual Conference on Information Sciences and Systems CISS'05*, Johns Hopkins University, Baltimore, MD, March 2005.

[12] S. T. Cheng and C. Y. Wang, "Quantum switching and quantum merge sorting," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 53, no. 2, pp. 316–325, Feb 2006.

[13] V. E. Beneš, *Mathematical Theory of Connecting Networks and Telephone Traffic*. Academic Press, 1965.

[14] D. Nassimi and S. Sahni, "Parallel algorithms to set up the benes permutation network," *IEEE Transactions on Computers*, vol. C-31, no. 2, pp. 148–154, 1982.

[15] C.-Y. Lee and A. Y. Oruc, "A fast parallel algorithm for routing unicast assignments in benes networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 06, no. 3, pp. 329–334, 1995.

[16] D. Kielpinski, C. Monroe, and D. J. Wineland, "Architecture for a large-scale ion-trap quantum computer," *Nature*, vol. 417, no. 6890, pp. 709–711, 2002.

[17] B. E. Kane, "A silicon-based nuclear spin quantum computer," *Nature*, vol. 393, no. 6681, pp. 133–137, 1998.

[18] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*. Cambridge University Press, September 2000.

[19] R. A. Spanke and V. E. Beneš, "N-stage planar optical permutation network," *Applied Optics*, vol. 26, no. 7, pp. 1226–1229, Apr 1987.

[20] W. H. Kautz, K. N. Levitt, and A. Waksman, "Cellular interconnection arrays," *IEEE Transactions on Computers*, vol. C-17, no. 5, pp. 443–451, 1968.