Implementing a Toffoli Gate in an N-Qubit Linear Nearest Neighbor Array

Saman Daraeizadeh*
Faculty: Dr. Preethika Kumar

Department of Electrical Engineering and Computer Science, College of Engineering

Abstract. Most proposals for quantum computers are based on linear nearest neighbor (LNN) arrangements where qubits only interact with neighbors. A Toffoli gate is a three-qubit operation, which is used in many quantum applications such as error correction, and algorithms like Shor's factorization. Typically, to implement a Toffoli gate in an LNN architecture, additional operations called swap gates are required to bring the qubits adjacent to each other. We propose a new method to implement a Toffoli gate in LNN arrays without using swap gates. As such, compared to other circuits, the quantum cost of our circuit is much lower.

1. Introduction

In quantum computing, a qubit (or quantum bit) is represented by the state $\alpha|0\rangle+\beta|1\rangle$ where $|\alpha|^2 + |\beta|^2 = 1$. A Toffoli gate is a three-qubit gate, which has two controls and one target qubit. When the two controls are each in the $|1\rangle$ state, the state of the target is flipped. Currently, to implement a Toffoli gate in a three-qubit system 11 elementary gates are required (single qubit gates and controlled-NOT gates) [1]. As the number of qubits increases, the total number of elementary gates increases. This is because most physical arrangements of qubits are along an LNN array where a qubit only interacts with its neighbors. As such, to interact two non-NN qubits, additional gate operations are required to bring them adjacent to each other. Here, we show methods for implementing a three-qubit Toffoli gate efficiently in an N-qubit system.

2. Toffoli gate in a three qubit system

Fig. 1 shows a Toffoli gate for a three-qubit system with qubit 3 as the target. Here, we implement the gate by sandwiching a $C^2(-I)$ gate between two controlled-Hadamard ($C^1(H)$) gates. A $C^2(-I)$ gate has two controls and 1 target [2]. A π phase is picked up by the target only when the two controls are each in the $|1\rangle$ state. Overall, a $C^2(-I)$ gate is equivalent to a controlled-Z ($C^1(Z)$) gate, which is a symmetric gate (either qubit can be the target or the control). The $C^1(Z)$ and $C^1(H)$ gates implement a Z and an Hadamard (H) gate, respectively, on the target qubit when the control qubit is in the $|1\rangle$ state. The Z and H gates are defined as:

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad ; \quad H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \qquad (1)$$

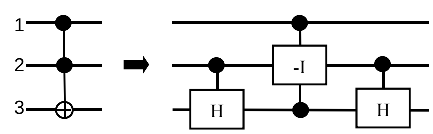

Fig.1. Implementation of a Toffoli gate with qubit 3 as the target, using 3 elementary gates [2].

3. Toffoli gate in a four qubit system

Fig. 2 shows our method for implementing a Toffoli gate in a four qubit system where two of participants are not adjacent. Hereafter, we will refer to the control and target qubits as participants. The Toffoli gate is realized by sandwiching a $C^2(Z)$ gate between two H gates applied on the target qubit. For realizing the $C^2(Z)$ gate, two Toffoli gates between qubits 1, 2 and 3 with qubit 2 as the target and two $C^1(Z)$ gates between qubits 2 and 4, are implemented . The two $C^1(Z)$ gates between qubits 2 and 4 are replaced by two $C^2(-I)$ gates (Fig. 2).

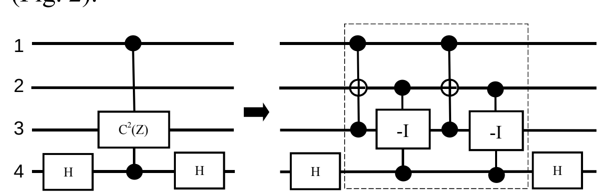

Fig. 2. Implementation of a Toffoli gate in a four qubit system using qubit 2 as the dummy.

4. Toffoli gate in an *N* qubit system

Fig. 3 shows a Toffoli gate with three participants in an *N* qubit system, where qubits *p, q* and *r* are the participants. Here and throughout, *p* is the 1st qubit, *q* is in the middle, and *r* is the *N*th qubit. As before, to perform a Toffoli gate, a $C^2(Z)$ gate is realized on these three participants which is sandwiched between two H gates applied on the target qubit. In our method, we divide qubits into two major groups of "*m*" and "(*N*–*m*)+1" qubits. If *N* is odd, we choose "*m*=(*N*+1)/2", and if *N* is even, we choose "*m*=(*N*+2)/2". In either case, if *m*=*q*, we change *m*=*q*+1. The dummy qubit is selected on *m*'s position (a dummy qubit is one that is used as a target that is finally restored to its initial state). Depending on whether participant *q* is in group "*m*" (case 1) or in group "(*N*–*m*)+1" (case 2), each $C^2(Z)$ gate is decomposed into four gates. In case 1, the $C^2(Z)$ gate is replaced by two Toffoli gates between *p*, *q* and *m*, with *m* as the target, and two $C^1(Z)$ gates between *m* and *r*. In case 2, the $C^2(Z)$ gate is substituted by two Toffoli gates between *r*, *q* and *m* (*m* as the target), and two $C^1(Z)$ gates between *m* and *p*.

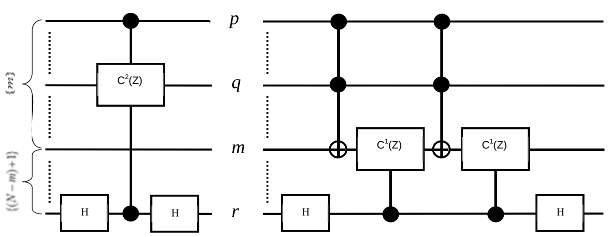

Fig. 3. Implementation of a Toffoli gate in an *N* qubit system (case 1) where participant *q* is in group "*m*".

In Fig. 3, the two Toffoli implemented between qubits *p*, *q* and *m* (group "m") need not be between NN qubits. If the Toffoli gates are not of the form of Fig. 1, further decomposition into sub-groups may be needed. This process of decomposition is carried out until all Toffoli gate operations are of the form of Fig. 1. Likewise, in each group/subgroup, if the $C^1(Z)$ gates are such that the separation between the control and target qubits is more than one qubit, each gate will have to be broken down into subgroups. In Ref. [2], the author shows how to implement a $C^1(Z)$ gate between non-NN qubits. For $P \geq 7$, where *P* is the number of qubits, the total gate count for the $C^1(Z)$ gate was shown to be [2]:

$$G_P = (20 + (P-7) \times 6)\,;\; P \geq 7 \qquad (2)$$

Note that, here $P = (N–m)+1$.

Following the procedure of decomposing the *N* qubit system into groups and subgroups, the total gate count for the Toffoli gate was calculated for different *N*. Fig. 4 shows the plot (blue line). From the plot, a second order polynomial trend line was calculated (black line), which gave an equation for the gate count, G_N (total number of elementary gates) as a function of N:

$$G_N = 0.2519N^2 + 6.6723N - 36.973 \qquad (3)$$

As a measure of the closeness of this equation to the actual gate count, for N = 40, Eq. (3) gives G_N = 633, while the exact value of G_N using our method is 626.

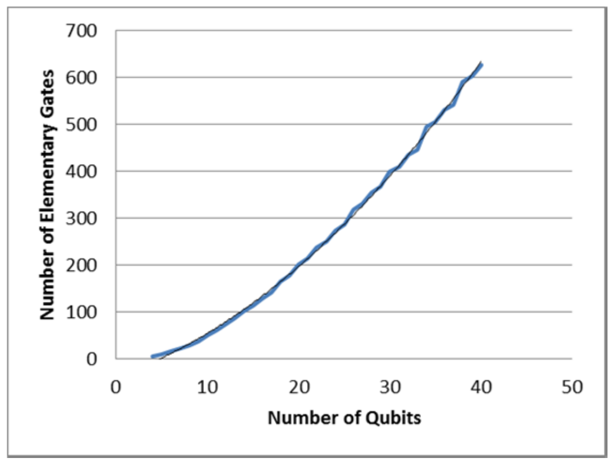

Fig. 4. Number of qubits versus number of elementary gates.

Observing Eq. (2), G_N is a quadratic function of N. The gate count can be further reduced by making it a linear function of N. This can be done by combining our method with swap operations. It was found that for $N \geq 7$:

$$G_N = (22 + (N-7) \times 6)\,;\; N \geq 7 \qquad (4)$$

5. Conclusions

In this paper, we introduced a new method to implement a Toffoli gate in an LNN array efficiently, by reducing the number of elementary gates. A quadratic as well a linear variation of the total gate count with the number of qubits has been presented.

[1] Mehdi Saeedi, Robert Wille, Rolf Drechsler, Quantum Information Processing June 2011, Volume 10, Issue 3, pp 355-377
[2] Preethika Kumar, Quantum Information Processing April 2013, Volume 12, Issue 4, pp 1737-1757