Efficient Implementation of Controlled Operations for Multivalued Quantum Logic

David Rosenbaum[1], Marek Perkowski[2]
[1]Portland State University, Department of Computer Science
[2]Portland State University, Department of Electrical Engineering
Email: drosenba@cs.pdx.edu, mperkows@ee.pdx.edu

Abstract

This paper presents a new quantum array that can be used to control a single-qudit hermitian operator for an odd radix $r > 2$ by n controls using $\Theta\left(n^{\log_2 r+2}\right)$ single-qudit controlled gates with one control and no ancilla qudits. This quantum array is more practical than existing quantum arrays of the same complexity because it does not require the use of small roots of the operation that is being implemented. Another quantum array is also presented that implements a single-qudit operator with n controls for any radix $r > 2$ using $\lceil \log_{r-1} n \rceil$ ancilla qudits and $\Theta\left(n^{\log_{r-1} 2+1}\right)$ single-qudit gates with one control.

1 Introduction

Controlled single-qudit gates are a fundamental concept in multivalued quantum computing. Because controlled single-qudit gates with many controls are not directly available for synthesis, implementing these gates efficiently using multiple single-qudit gates with only one control is an important problem for multivalued quantum computing. Many applications exist for single-qudit gates with multiple controls including implementing unitary arithmetic operations [8, 5], synthesizing multivalued quantum circuits [6, 4] and oracles for Grover's algorithm in multivalued quantum logic [3]. The problem of controlling a single-qubit gate by many controls was solved by Barenco et al. [1] for binary quantum logic using $\Theta\left(n^2\right)$ single qubit gates with one control and no ancilla qubits. Muthukrishnan and Stroud [7] developed a quantum array that can be used to control multivalued single-qudit operations in a radix $r > 2$ by $n \geq 2$ controls using $\Theta(n)$ single-qudit gates with one control and $\left\lceil \frac{n-1}{r-2} \right\rceil$ ancilla qudits. The quantum array by Barenco et al. [1] was extended by Brennen, Bullock and O'Leary [2] for multivalued quantum computing using $\Theta\left(n^{\log_2 r+2}\right)$ single-qudit gates with one control without using any ancilla qudits where r is the radix and n is the number of controls. However, this quantum array requires taking small roots of the operation that is being controlled which is not practical since these roots correspond to rotations by small angles on the Bloch sphere. A new quantum array for implementing hermitian operations in an odd radix $r > 2$ with n controls will be shown that uses $\Theta\left(n^{\log_2 r+2}\right)$ single-qudit gates with one control and no ancilla qudits but does not require taking small roots as is the case for existing quantum arrays that do not use ancilla qudits. Another quantum array will be shown that requires only $\Theta\left(n^{\log_{r-1} 2+1}\right)$ single-qudit gates with one control and can be used to control any single-qudit unitary operator in an arbitrary radix r but requires an additional $\lceil \log_{r-1} n \rceil$ ancilla qudits. These ancilla qudits can be reused later because their states are restored to $|0\rangle$ by the quantum array. Note that because the bases for the logarithms in these expressions are $r-1$, this second quantum array requires fewer gates and ancilla qudits for higher radices. In fact, it can be shown that as the radix increases, the number of gates required by this quantum array approaches n.

2 Introduction to Multivalued Quantum Logic

Before presenting the new quantum arrays introduced in this paper, a brief introduction to multivalued quantum logic will be provided. The multivalued analog of a qubit in an arbitrary radix r is called a qudit. In ternary logic, $r = 3$ and qudits are referred to as qutrits. The basis states of a qudit of radix r are $|0\rangle, \ldots, |r-1\rangle$ and the state of single qudit is of the form $\sum_{i=0}^{r-1} \alpha_i |i\rangle$ where each $\alpha_i, i = 0, \ldots, r-1$ is a complex number and $\sum_{i=0}^{r-1} |\alpha_i|^2 = 1$ assuming that the qudit is not entangled with any other qudits. Several different types of operations are used in this paper. The "+1" operation maps any basis state $|i\rangle$ to $|(i+1) \bmod r\rangle$. The "−1" operation is similar except that it maps any basis state $|i\rangle$ to $|(i-1) \bmod r\rangle$. Another type of operation is the transposition operation which is denoted by (jk) where $0 \leq j < k \leq r-1$. Applying the transposition (jk) to any basis state $|i\rangle$ results in $|k\rangle$ if $i = j$, $|j\rangle$ if $i = k$ and $|i\rangle$ if $i \neq j$ and $i \neq k$. An important property of transpositions that will be exploited later is that every transposition is its own inverse and all transpositions are therefore hermitian. In this paper, controlled operations in quantum arrays are always controlled by $|r-1\rangle$ on each control unless otherwise indicated.

3 Controlling Single-Qudit Operations Without Using Ancilla Qudits

This section will show how to control a single-qudit operation U by $n \geq 2$ controls without using any ancilla qudits or small roots of operations. For this quantum array, the single-qudit operator that is being controlled must be hermitian and the radix $r > 2$ must be an odd number. The reasons for these requirements will be discussed later. The general form of the quantum array is shown in figure 1. This quantum array is applied recursively to construct the controlled U operation with $n-1$ controls. The "+1" operations in figure 1 that are controlled by $|r-1\rangle$ on each of the qudits $|x_i\rangle, i = 1, \ldots, n-1$ are implemented using a special quantum array that was developed by Brennen, Bullock and O'Leary [2] and was based on a quantum array for binary quantum logic by Barenco et al. [1]. This quantum array is shown in figure 2 and applies a "+1" operation controlled by $|r-1\rangle$ on each of the qudits $|x_i\rangle, i = 1, \ldots, n$ to the qubit $|y\rangle$ where $n > 2$. The quantum array in figure 2 does not control by the qudit $|x_{n+1}\rangle$ and it will have the same value after the quantum array is applied. Since the quantum array in figure 2 cannot be used to implement "+1" operations with two controls it is necessary to use a special trick in this case. From group theory, $+1 = (0\, r-1) \ldots (01)$. This allows "+1" operations with two controls to be implemented using $r-1$ transposition operations. Because every transposition is its own inverse, every transposition is a hermitian operator. Therefore, the quantum array in figure 1 can be used to implement each transposition. The quantum array in figure 1 will now be explained. The basic idea is to apply a U operation to the qudit $|y\rangle$ if each of the qudits $|x_i\rangle, i = 1, \ldots, n-1$ is equal to $|r-1\rangle$. This gives the correct behavior unless $|x_n\rangle \neq |r-1\rangle$. However, in this case the control for exactly one of the U gates that are controlled by $|x_n\rangle$ will be equal to $|r-1\rangle$ so a second U gate will be applied to the qudit $|y\rangle$. This will reverse the effect of the first U operation so the net effect will be applying the identity operation.

3.1 A Simple Example

An example will now be shown to illustrate how the quantum array in figure 1 works. For this example, the values $r = 3$ and $n = 3$ will be used. The controlled operation will be U. Note that r is odd and U must be a hermitian operation. Applying the quantum array in figure 1 for $r = 3$ and $n = 3$ results in the quantum array in figure 3. The "+1" operations with 2 controls cannot be implemented using the quantum array in figure 2. Therefore, since $+1 = (02)(01)$, the "+1" operations with 2 controls are implemented using (02) and (01) operations with 2 controls. Note that (02) and (01) operations with 2 controls can be implemented using the quantum array in figure 1 because (02) and (01) are hermitian operations whereas the "+1" operator is not hermitian.

3.2 Correctness of the Quantum Array

The following theorem shows that the quantum array in figure 1 applies a U operation to the qudit $|y\rangle$ that is controlled by $|r-1\rangle$ on the qudits $|x_i\rangle, i = 1, \ldots, n$.

Theorem 1. *If $r > 2$ is an odd radix, $n \geq 2$ is the number of controls and U is a single-qudit hermitian operation then the quantum array in figure 1 applies a U operation to $|y\rangle$ that is controlled by $|r-1\rangle$ on each of the qudits $|x_i\rangle, i = 1, \ldots, n$.*

Proof. To prove this theorem, it will first be proven by induction on the number of controls j that the quantum array in figure 1 works correctly for all input basis states. For the basis case, $j = 2$. Consider the quantum array in figure 1 for $j = 2$ controls. Assume that $|x_1\rangle \neq |r-1\rangle$. Then the first U operation with one control will not be applied since its control will not be $|r-1\rangle$. The "+1" operations with one control also will be not applied for the same reason. Hence, in this case $r-1$ U operations controlled by $|r-1\rangle$ on the qudit $|x_2\rangle$ will be applied to the qudit $|y\rangle$. Because r is odd $r-1$ is even. Therefore, since U is hermitian, $U = U^\dagger = U^{-1}$ so applying these $r-1$ U operations to the qudit $|y\rangle$ will be equivalent to applying the identity operation. Thus, the quantum array is correct if $|x_1\rangle \neq |r-1\rangle$. Now assume that $|x_1\rangle = |r-1\rangle$. Then all of the operations controlled by $|r-1\rangle$ on the qudit $|x_1\rangle$ will be applied. If $|x_2\rangle \neq |r-1\rangle$ then exactly one of the U operations controlled by $|x_2\rangle$ will have its control equal to $|r-1\rangle$ so the entire quantum array will apply exactly two U operations to the qudit $|y\rangle$ and will thus be equivalent to applying the identity operation. If $|x_2\rangle = |r-1\rangle$ then none of the U operations controlled by $|x_2\rangle$ will have its control equal to $|r-1\rangle$ so the entire quantum array will apply exactly one U operation to the qudit $|y\rangle$. This proves the basis case. For the inductive case, assume that the quantum array is correct for j controls. Consider figure 1 for $j+1$ controls. By the inductive hypothesis the U operation with j controls works correctly. By applying the same reasoning as in the basis case it can be shown that the quantum array is correct for $j+1$ controls. Thus, the quantum array works correctly for all basis states by the principle of mathematical induction. Because unitary matrices are linear operators the quantum array in figure 1 works correctly for all quantum states. □

3.3 Complexity of the Quantum Array

The number of single-qudit gates with one control required by the quantum array in figure 1 will now be calculated. First, the number of single-qudit gates with one control required to implement a "+1" operation with n controls will be found. Let $f_{r,n}$ denote the number of single-qudit gates with one control required to implement a "+1" operation with n controls using the quantum array in figure 2 when $n > 2$. When $n = 2$ the quantum array in figure 2 cannot be used and the "+1" operation is decomposed to transpositions as previously described. The quantum array in figure 1 is then used to implement each transposition with 2 controls. Then $f_{r,n}$ is defined by the recurrence

$$f_{r,2} = 2r(r-1) \tag{1}$$

$$f_{r,n} = r\left(f_{r,\lceil\frac{n+1}{2}\rceil} + f_{r,\lfloor\frac{n+1}{2}\rfloor}\right) \tag{2}$$

Let $g_{r,n}$ be the number of single-qudit gates with one control required to control a hermitian single-qudit operator U by n controls using the quantum array in figure 1. Then $g_{r,n}$ is defined by the recurrence

$$g_{r,2} = 2r \tag{3}$$

$$g_{r,n} = g_{r,n-1} + rf_{r,n-1} + r - 1 \tag{4}$$

It can be shown that

$$g_{r,n} \in \Theta\left(n^{\log_2 r+2}\right) \tag{5}$$

Thus, the quantum array in figure 1 requires $\Theta\left(n^{\log_2 r+2}\right)$ single-qudit gates with one control. The notation $g_{r,n} \in \Theta\left(n^{\log_2 r+2}\right)$ means that $g_{r,n}$ is asymptotically bounded below by some positive constant multiple of $n^{\log_2 r+2}$ and is is asymptotically bounded above by some positive constant multiple of $n^{\log_2 r+2}$. Although this is the same as for the quantum array by Brennen, Bullock and O'Leary [2], this quantum array does not require taking small roots of the operation that is being controlled. This makes it much more practical to implement on an actual quantum computer since it does not require rotations by small angles on the Bloch sphere.

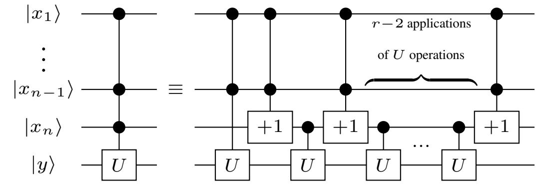

Figure 1: The quantum array for implementing single-qudit controlled operations without using ancilla qudits

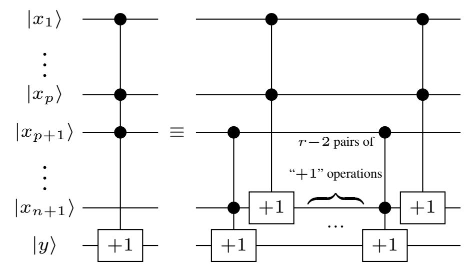

Figure 2: The quantum array for implementing "+1" controlled operations without using ancilla qudits where $p = \lceil\frac{n+1}{2}\rceil$

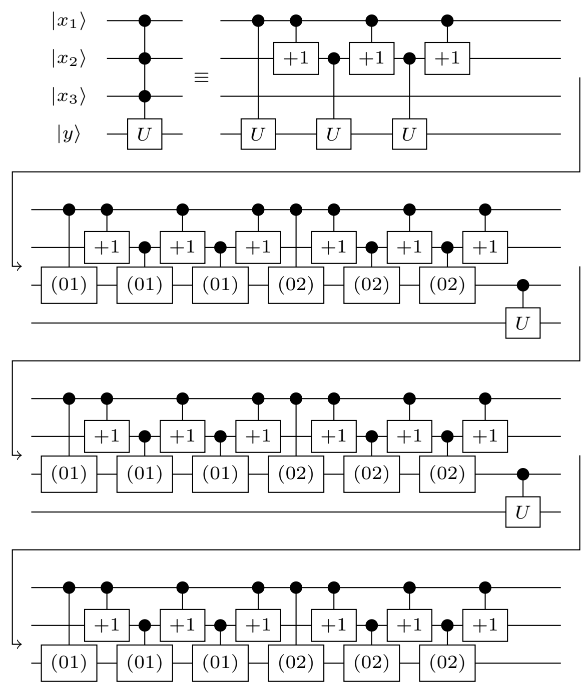

Figure 3: A quantum array for implementing a U operation with 3 controls in ternary quantum logic without using ancilla qutrits

4 Controlling Single-Qudit Operations Using $\lceil \log_{r-1} n \rceil$ Ancilla Qudits

This section will demonstrate how implement a U single-qudit operation with $n \geq 2$ controls using a logarithmic number of ancilla qudits. Although it is desirable to use as few ancilla qudits as possible, the number of single-qudit gates with one control can be dramatically reduced by adding a logarithmic number of ancilla qudits. Additionally, this quantum array does not require the radix $r > 2$ to be odd and the controlled single-qudit operation U may be an arbitrary unitary operation and does not need to be hermitian. Figure 4 shows the general form of the quantum array where $m = \lceil \log_{r-1} n \rceil$ is the number of ancilla qudits. Each of the ancilla qudits $|c_i\rangle, i = 1, \ldots, m$ must be initialized to the state $|0\rangle$ before the quantum array is applied. These ancilla qudits can be reused later for other tasks as the quantum array restores their states to $|0\rangle$. In order to apply this quantum array directly, it is necessary for the number of controls n to be a power of $r-1$ so that $n = (r-1)^m$. However, it will be shown later that this quantum array can also be applied when n is not a power of $r-1$. The quantum array in figure 4 is applied recursively to implement each of the $r-1$ "+1" and "−1" operations with $(r-1)^{m-1}$ controls. The idea is that the control for the U operation will only be equal to $|r-1\rangle$ if all of the "+1" operations with $(r-1)^{m-1}$ controls have been applied. This will only happen if each of the qudits $|x_i\rangle, i = 1, \ldots, n$ is equal to $|r-1\rangle$. Therefore, the U operation will only be applied to the qudit $|y\rangle$ when each of the qudits $|x_i\rangle, i = 1, \ldots, n$ is equal to $|r-1\rangle$. By applying this reasoning recursively, it can be seen that the quantum array works correctly. Now if n is not a power of $r-1$ then $n' = (r-1)^{\lceil \log_{r-1} n \rceil}$ is a power of $r-1$ and $n' > n$. Hence, in this case the quantum array in figure 4 can be used to implement a controlled U operation with n' controls using $\lceil \log_{r-1} n' \rceil = \lceil \log_{r-1} n \rceil$ ancilla qudits. This results in an extra $n - n'$ controls which can be removed by treating them as set to $|r-1\rangle$. Any of the "+1" and "−1" gates with one control that was controlled by a control that is now constant is replaced by a new gate with the same operation but no controls.

4.1 A Simple Example

An example will now be presented to show how the quantum array in figure 4 works. For this example, $r = 3$ and $n = 4$ will be used. The controlled operation will be U. Note that although r is odd in this case, this is not required and r may be even. The controlled operation U can be any single-qudit unitary operation and does not need to be hermitian. The number of required ancilla qutrits is $m = 2$ since $\lceil \log_2 4 \rceil = 2$. Applying the quantum array in figure 4 results in the quantum array in figure 5. As mentioned earlier, it is necessary to initialize the ancilla qutrits $|c_1\rangle$ and $|c_2\rangle$ to $|0\rangle$ before the quantum array is applied. Observe that some of the "+1" and "−1" operations in figure 5 are not required. However, these have been left in the quantum array in order to show the pattern.

4.2 Correctness of The Quantum Array

The following theorem shows that the quantum array in figure 4 is correct.

Theorem 2. *If $r > 2$ is an arbitrary radix, $n = (r-1)^m$ is the desired number of controls, $m \geq 1$ is the number of ancilla qudits and U is an arbitrary single-qudit unitary operation then the quantum array in figure 4 applies a U operation to $|y\rangle$ that is controlled by $|r-1\rangle$ on each of the qudits $|x_i\rangle, i = 1, \ldots, n$.*

Proof. Let $n_j = (r-1)^j$. Then $n = n_m$. Assume that the state that the quantum array is figure 4 is applied to is a basis state. It will be proven that the quantum array is correct by induction on j. For the basis case $j = 1$ and $n_1 = r-1$. Consider the U operation in figure 4. Because the ancilla qudits are initialized to $|0\rangle$, the control for the U operation will be equal to $|r-1\rangle$ if and only if each of the preceding "+1" operations is applied. This will happen if and only if each of the qudits $|x_i\rangle, i = 1, \ldots, r-1$ is equal to $|r-1\rangle$ which implies that the U operation will be applied to the qudit $|y\rangle$ if and only if each of the qudits $|x_i\rangle, i = 1, \ldots, r-1$ is equal to $|r-1\rangle$. This proves the basis case. For the inductive case assume that the quantum array in figure 4 is correct for n_j controls. Consider the quantum array in figure 4 for n_{j+1} controls. Observe that because in this case each of the "+1" and "−1" operations has n_j controls, each of the "+1" and "−1" operations is implemented correctly by the inductive hypothesis. By applying the same reasoning for the control of the U operation as in the basis case, the U operation is applied if and only if each of the qudits $|x_i\rangle, i = 1, \ldots, n_{j+1}$ is equal to $|r-1\rangle$. This proves that the quantum array works correctly for all basis states by the principle of mathematical induction. Because unitary matrices are linear operators it works correctly for all quantum states. □

4.3 Complexity of The Quantum Array

The number of single-qudit gates with one control required by the quantum array in figure 4 will now be calculated. Let $h_{r,m}$ be the number of single-qudit gates required to implement a U operation with $n = (r-1)^m$ controls where m is the number of ancilla qudits used. The reason the recurrence uses the number of ancilla qudits instead of the number of controls is because this simplifies the analysis. Note that this assumes that n is a power of $r-1$. This result will be generalized later in this paper. The recurrence $h_{r,m}$ is defined by

$$h_{r,1} = 2r - 1 \tag{6}$$

$$h_{r,m} = 2(r-1)h_{r,m-1} + 1 \tag{7}$$

It can be shown that

$$h_{r,m} = \frac{2(r-1)n^{\log_{r-1} 2+1} - 1}{2r-3} \tag{8}$$

This implies that

$$h_{r,m} \in \Theta\left(n^{\log_{r-1} 2+1}\right) \tag{9}$$

Thus, $\Theta\left(n^{\log_{r-1} 2+1}\right)$ single-qudit gates with one control are required to implement a U operation with n controls using the quantum array in figure 4. It is important to note that equation (8) only holds when n is a power of $r-1$. However, equation (9) holds even when n is not a power of $r-1$ because taking n' to be the smallest power of $r-1$ greater than n increases n by a factor that is less than $r-1$. The complexity of the quantum array in figure 4 has several desirable properties. In addition to using significantly fewer single-qudit gates with one control than the quantum array in figure 1, the quantum array in figure 4 becomes more efficient as the radix r becomes larger in contrast to the quantum array in figure 4 which requires more single-qudit gates with one control when the radix is larger. This is true both in terms of the number of single-qudit gates with one control that are required as well as the number of required ancilla qudits. This can be seen from equation (8) because $\log_{r-1} 2$ will be smaller for larger values of r. In fact, in can be shown that the number of single-qudit gates that are required approaches n as the radix r increases. Since $m = \lceil \log_{r-1} n \rceil$ is the number of ancilla qudits, a larger value for r will cause m to be smaller as well.

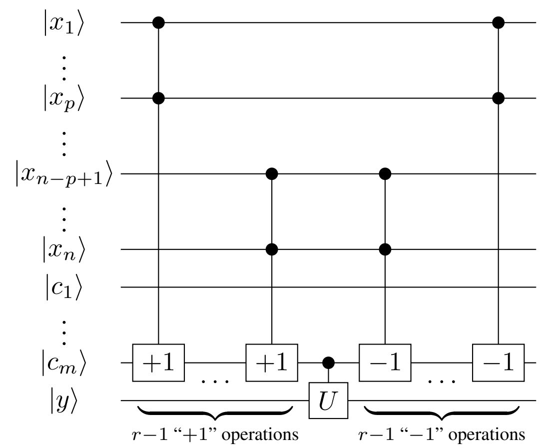

Figure 4: The quantum array for implementing single-qudit controlled operations using $\lceil \log_{r-1} n \rceil$ ancilla qudits where $p = (r-1)^{m-1}$

5 Comparison of Implementations of Controlled Operations for Multivalued Quantum Logic

The different approaches to implementing controlled operations presented in this paper will now be compared. The quantum array developed by Muthukrishnan and Stroud [7] uses only $\Theta(n)$ single-qudit gates with one control. This is better than the other quantum arrays, however the quantum array created by Muthukrishnan and Stroud also requires $\left\lceil \frac{n-1}{r-2} \right\rceil$ ancilla qudits which is significantly more than any of the other quantum arrays. The quantum array invented by Brennen, Bullock and O'Leary [2] does not use any ancilla qudits but requires $\Theta\left(n^{\log_2 r+2}\right)$ single-qudit gates with one control. The quantum array shown in figure 1 requires the same number of single-qudit gates with one control and does not need any ancilla qudits but requires the radix $r > 2$ to be odd and the controlled operation U to be hermitian. However, this quantum array has the advantage of not requiring small roots of the operation being controlled as is the case with the quantum array by Brennen, Bullock and O'Leary [2]. The quantum array shown in figure 4 requires $\Theta\left(n^{\log_{r-1} 2+1}\right)$ single-qudit gates with one control and $\lceil \log_{r-1} n \rceil$ ancilla qudits and can be used to control any single qudit hermitian operator in an arbitrary radix $r > 2$. It is important to note that as the radix r becomes large, this quantum array uses only n single-qudit gates with one control which is the same as the quantum array by Muthukrishnan and Stroud [7]. These comparisons are summarized in table 1.

6 Conclusion

This paper presented two new quantum arrays for implementing controlled operations in multivalued quantum logic using single-qudit gates with one control. The first quantum array was shown in figure 1 and is capable of controlling a single-qudit hermitian operator by n qudits in an odd radix $r > 2$ using $\Theta\left(n^{\log_2 r+2}\right)$ single-qudit

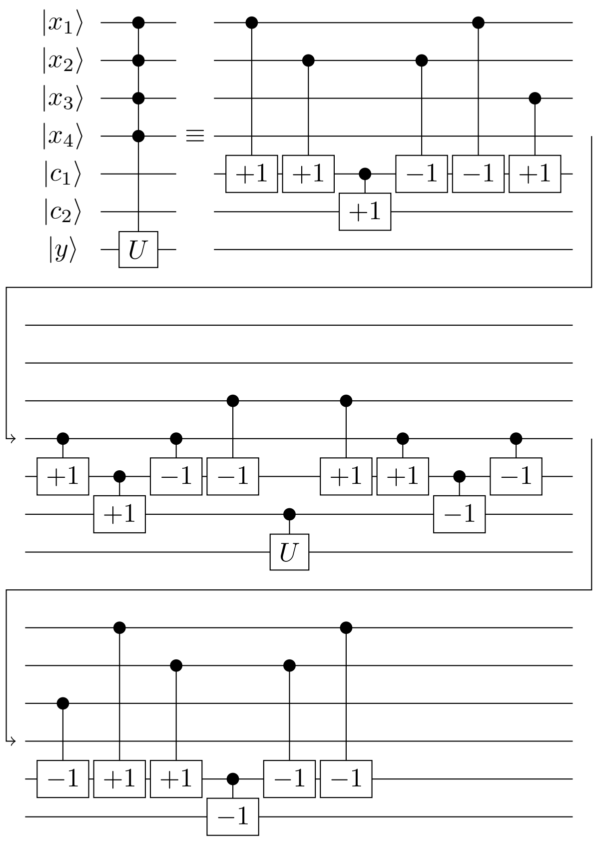

Figure 5: A quantum array for implementing a U operation with 4 controls in ternary quantum logic using 2 ancilla qutrits

gates with one control and no ancilla qudits. A quantum array developed by Brennen, Bullock and O'Leary [2] is also capable of implementing controlled single-qudit operations using the same number of single-qudit gates with one control without using ancilla qudits. However, it is not practical to implement because it requires taking small roots of the operation that is being controlled whereas the quantum array in figure 1 does not. The second quantum array shown in this paper was given in figure 4 and can be used for controlling an arbitrary single-qudit unitary operation by n qudits in an arbitrary radix $r > 2$. It uses $\Theta\left(n^{\log_{r-1} 2+1}\right)$ single-qudit gates with one control and requires $\lceil \log_{r-1} n \rceil$ ancilla qudits. This drastically reduces the number of required single-qudit gates with one control without using a linear number of ancilla qudits as was previously required [7]. Because controlled operations are fundamental in multivalued quantum computing, these quantum arrays are useful as they provide different tradeoffs between the required number of single-qudit gates with one control

Quantum Array	Gates	Ancillas
Muthukrishnan	$\Theta(n)$	$\left\lceil \frac{n-1}{r-2} \right\rceil$
Brennen	$\Theta\left(n^{\log_2 r+2}\right)$	0
Figure 1	$\Theta\left(n^{\log_2 r+2}\right)$	0
Figure 4	$\Theta\left(n^{\log_{r-1} 2+1}\right)$	$\lceil \log_{r-1} n \rceil$

Table 1: Comparison of quantum arrays for implementing controlled operations where r is the radix and n is the number of controls

and the number of ancilla qudits.

References

[1] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter. Elementary gates for quantum computation. *Physical Review A*, 52:3457–3467, 1995.

[2] G. K. Brennen, S. S. Bullock, and D. P. O'Leary. Efficient circuits for exact-universal computation with qudits. *Quantum Information and Computation*, 6:436–454, 2006.

[3] Y. Fan. Applications of multi-valued quantum algorithms. *Available online at http://arxiv.org/pdf/0809.0932*, 2007.

[4] F. S. Khan and M. M. Perkowski. Synthesis of ternary quantum logic circuits by decomposition. In *Proceedings of the 7th International Symposium on Representations and Methodology of Future Computing Technologies*, 2005.

[5] M. H. A. Khan and M. A. Perkowski. Quantum ternary parallel adder/subtractor with partially-look-ahead carry. *Journal of Systems Architecture*, 53:453–464, 2007.

[6] D. M. Miller, D. Maslov, and G. W. Dueck. Synthesis of quantum multiple-valued circuits. *Multiple Valued Logic and Soft Computing*, 12:431–450, 2006.

[7] A. Muthukrishnan and C. R. Stroud. Multivalued gates for quantum computation. *Physical Review A*, 62:052309, 2000.

[8] M. A. Thorton, D. W. Matula, and L. Spenner. Quantum logic implementation of unary arithmetic operations. In *Proceedings of the 38th International Symposium on Multiple Valued Logic*, pages 202–207, 2008.