Elementary Quantum Gate Realizations for Multiple-Control Toffoli Gates

D. Michael Miller[1], Robert Wille[2] and Z. Sasanian[1]
[1]Department of Computer Science, University of Victoria, Canada
[2]Group for Computer Architecture, University of Bremen, Germany
mmiller,sasanian@uvic.ca, rwille@informatik.uni-bremen.de

***Abstract*—A new method for determining elementary quantum gate realizations for multiple-control Toffoli (MCT) gates is presented. The realization for each MCT gate is formed as a composition of realizations of smaller MCT gates. A marking algorithm which is more effective than the traditional moving rule is used to optimize the final circuit. The main improvement is that the resulting circuits make significantly better use of ancillary lines than has been achieved in earlier approaches. Initial results are also presented for circuits with nearest-neighbour communication. These results show that the overall approach is not as effective for that problem indicating that research on direct synthesis of nearest-neighbour quantum circuits should be considered. While, the results presented are for the NCV quantum gate library (*i.e.* for quantum circuits composed of NOT gates, controlled-NOT gates, and controlled-V/V^+ gates), the approach can be applied to other libraries of elementary quantum gates.**

I. Introduction

Many reversible circuit synthesis methods have been presented in the literature. A good review can be found in [1]. These methods generally produce a circuit composed as a cascade of basic reversible gates. After, or sometimes during, synthesis the reversible gates are mapped to elementary quantum gates implemented in the target technology. The interest in this paper is how to realize the commonly used *multiple-control Toffoli* (MCT) gates using the *NCV quantum gate library* (*i.e.* using NOT gates, controlled-NOT gates, and controlled-V/V^+ gates only). In particular, we consider how to make best use of the available ancillary lines. We also consider how the method can be applied to the situation where the target technology allows only nearest-neighbour connections, *i.e.* the target and control lines must be adjacent for every quantum gate in the circuit. Our initial results, indicate that the approach is not very effective for the nearest-neighbour case. The results presented indicate the problem is in using MCT gates as an intermediary step.

Although the paper concentrates on MCT gates, the proposed methods can be applied to other reversible gates, *e.g.* Fredkin [2] gates, by transforming them to Toffoli gate realizations. The approach can also be targeted to other quantum gate libraries.

Barenco *et al*. [3] provided the first comprehensive study of the realization of MCT gates in terms of elementary quantum gates. The decomposition methods presented in [4] were developed from key ideas presented in that work and provide significantly less costly realizations than had been commonly used in the literature [5], [6]. The methods presented here further improve the use of ancillary lines.

Nearest-neighbour circuits have been considered in [7]–[11]. In that work, the nearest neighbour is applied for reversible gates, *e.g.* MCT gates. In this paper, we examine the nearest-neighbour issue at the level of elementary quantum gates and demonstrate the significant complexity of the problem. The techniques presented are related to the work in [12].

All circuits presented in this paper have been verified using the QMDD circuit equivalence checker described in [13]. The catalog of NCV circuit realizations for MCT gates and the program that generates that catalog (in Python) are available from the first author.

The remainder of this paper is structured as follows. The next section introduces the notation and preliminaries needed in this paper. Section III presents our decomposition method which generates NCV quantum circuit for MCT gates. The decomposition method is extended in Section IV to incorporate the nearest-neighbour condition. Results are presented in those two sections. Finally, the paper is concluded and future work is suggested in Section V.

II. Background

Definition 1. *A multiple-output Boolean function is* **reversible** *if it maps each input assignment to a unique output assignment,* i.e. *it is a bijection. To satisfy this requirement, the function must have the same number of inputs and outputs and must be completely-specified,* i.e. *have no don't-care conditions. A function that is not reversible is termed* **irreversible**.

A reversible function can be realized by a circuit comprised of a cascade of reversible gates with no fan-out or feedback [14]. Many reversible gates have been proposed. Here, we consider multiple-control Toffoli gates which are defined as follows:

Definition 2. *A* **multiple-control Toffoli** *(MCT) gate with* **target line** x_j *and* **control lines** $\{x_{i_1}, x_{i_2} \cdots x_{i_k}\}$, *maps* $(x_1 \ldots x_j \ldots x_n)$ *to* $(x_1 \ldots (x_{i_1} x_{i_2} \cdots x_{i_k}) \oplus x_j \ldots x_n)$. *Note that all controls must be 1 for the target to be inverted.*

An MCT gate with no control line always inverts the target line and is thus the well-known **NOT** *gate. An MCT gate with a single control line is called a* **controlled-NOT** *(CNOT) gate (also called a Feynman gate). The case of two control lines is the original* **Toffoli** *gate [15].*

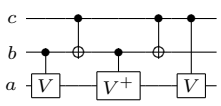

Fig. 1. NCV realization of Toffoli gate $T(c, b; a)$.

We use $M(C; t)$ to denote an MCT gate with C being the set of controls and t being the target. $T(a, b; t)$ will denote a Toffoli gate with controls a and b and target t, while $CN(a; t)$ will denote a CNOT gate with control a and target t. For drawing these gates, we follow the normal convention of using a $\oplus$ to indicate the target line and a $\bullet$ to indicate a control connection. Note that we do not consider the use of negative controls in this paper.

Definition 3. *A line which is not the target or a control of an MCT gate but is used in implementing the MCT gate as a cascade of simpler gates is termed an* **ancillary line**.

Many quantum gates have been defined and studied in the literature [14]. In this paper, we concentrate on the following gates (termed the **NCV quantum gate library**):

- NOT and controlled-NOT (CNOT);
- The 2-line controlled-V gate which changes the target line using the transformation defined by the matrix $\mathbf{V} = \frac{1+i}{2}\begin{pmatrix} 1 & -i \\ -i & 1 \end{pmatrix}$ if the single control line has the value 1;
- The 2-line controlled-V^+ gate which changes the target line using the transformation $\mathbf{V}^+ = \mathbf{V}^{-1} = \frac{1-i}{2}\begin{pmatrix} 1 & i \\ i & 1 \end{pmatrix}$ if the single control line has the value 1.

Since V and V^+ gates are always used with a single control line, we for simplicity omit the controlled qualifier.

Gates V and V^+ are referred to as ***square-root-of-*NOT** gates since $\mathbf{V}^2 = (\mathbf{V}^+)^2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$. Note that in this work V and V^+ are fixed gate types. In some work, [3] in particular, V varies depending on the context.

For V and V^+ gates, a box containing the appropriate symbol is placed on the target line and the control line is indicated as for MCT gates.

A Toffoli gate can be realized with 5 NCV gates, as shown in Fig. 1 [3].

Definition 4. *The cost of an NCV circuit is the number of gates in the circuit,* i.e. *NCV gates are assumed to have unit cost.*

For example, the circuit in Fig. 1 has cost 5.

The following properties and definitions are useful for simplifying circuits.

Property 1. *MCT gates, including NOT, CNOT and Toffoli gates, are self-inverse and two identical such gates which are adjacent (or can be moved to be adjacent) yield the identity mapping. V and V^+ gates with the same target and the same control are the inverse of each other and hence if adjacent (or can be moved to be adjacent) yield the identity mapping.*

Property 2. *Given a cascade of reversible gates $G_1 G_2 \ldots G_k$ realizing the reversible function F, the cascade $G_k^{-1} \ldots G_2^{-1} G_1^{-1}$ realizes the function F^{-1}, where G_i^{-1} is the inverse gate for G_i.*

Definition 5. *Since an MCT gate is self-inverse applying Property 2 to a realization of the gate yields an alternate realization for the same gate. We term this the* **reverse** *realization.*

Property 3. *In a circuit realizing a reversible function, the V and V^+ gates can be interchanged with no effect on the functionality.*

For example, the realization of a Toffoli gate shown in Fig. 1 can be used in four distinct ways: as given, reversed, and in both those cases with the V and V^+ gates interchanged. We note further that the rightmost gate can be moved to any position in the circuit. That is particular to this example. In general, certain gates can be moved within a circuit, a property we use to advantage below.

Definition 6. *A single control gate is a* **nearest-neighbour gate** *if its control and target are on adjacent circuit lines.*

This definition applies to CNOT, V and V^+ gates. NOT gates are not an issue since they involve only a single line.

Definition 7. *An NCV circuit is termed a* **nearest-neighbour circuit** *if every gate in the circuit is nearest-neighbour.*

The term nearest-neighbour has been applied directly to MCT and other reversible gates [7]–[11]. In that case, the idea is that the controls and target(s) of the gates occupy a consecutive set of lines with no intervening lines. However, we will show below that applying that constraint to MCT or other reversible gate does not, in general, lead to the least costly NCV realization.

Swap operations are usually required to implement a nearest-neighbour circuit. In this paper, we do not assume a swap gate is available. Rather, we note that two lines can be interchanged by a sequence of three CNOT gates as given in the following property.

Property 4. *Two circuit lines denoted a and b are swapped by the gate sequence $CN(a; b), CN(b; a), CN(a; b)$. The sequence $CN(b; a), CN(a; b), CN(b; a)$ can also be used to swap lines a and b.*

Note that two sequences are always available and one must be careful to choose the one leading to the best simplification of the circuit.

III. NCV Realizations of MCT Gates

This section introduces the proposed improvements on the decomposition of MCT gates. First, the basic concept already applied in previous work is briefly reviewed. Afterwards, our decomposition procedure is presented. At the end of this section, results obtained by these methods are given.

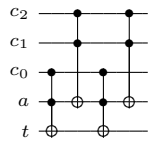

Fig. 2. A decomposition of $M(c_0, c_1, c_2; t)$ with one ancillary line a.

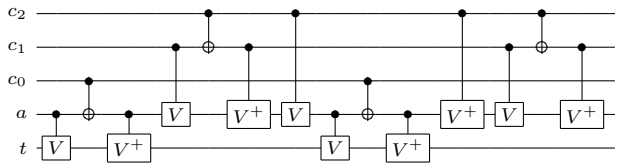

Fig. 3. NCV circuit for $M(\{c_0, c_1, c_2\}; t)$ with ancillary line a.

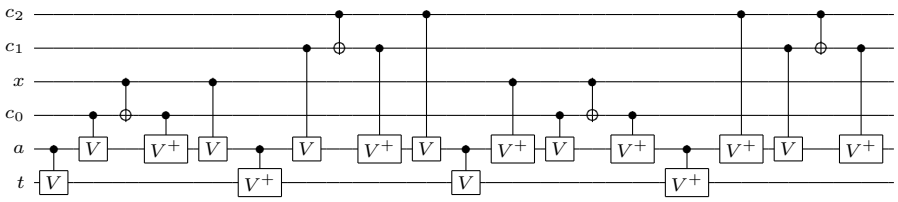

Fig. 4. $M(c_2, c_1, x, c_0; t)$ with one ancillary a.

A. Basic Concept

Consider realizing the MCT gate $M(c_0, c_1, c_2; t)$ given one ancillary line a. A well-known decomposition from [3] is shown in Fig. 2. This can be mapped to NCV gates as follows:

1) Expand the leftmost gate, $T(c_0, a; t)$ using the Toffoli gate realization in Fig. 1.
2) Expand the next gate, $T(c_2, c_1, a)$ using the Toffoli gate realization in Fig. 1.
3) Expand the next gate, $T(c_0, a; t)$ using the Toffoli gate realization in Fig. 1 reversed with V and V^+ interchanged.
4) Expand the last gate, $T(c_2, c_1, a)$ using the Toffoli gate realization in Fig. 1 reversed with V and V^+ interchanged.

The resulting circuit has 20 gates. But, a pair of gates from the first and third Toffoli gates cancels and two pairs of gates from the second and fourth Toffoli gates cancel. Thus, the 14 gate circuit as shown in Fig. 3 results.

The circuit just presented suggests a way to realize an MCT gate with more control lines as an NCV circuit. For example, consider the circuit in Fig. 3. We can insert a fourth control line (labeled x) by changing the two $CN(c_0; a)$ gates into Toffoli gates incorporating the new control. By expanding these gates in the manner outlined above, we obtain the circuit in Fig. 4 which has a cost of 20.

B. Decomposition Procedure

The idea of replacing the two CNOT gates in Fig. 3 as shown above can be extended to more controls and combined with the general form of the decomposition illustrated in Fig. 2. This leads to a new decomposition structure given by the following equation where $C = C_0 \cup C_1$ and $C_0 \cap C_1 = \phi$:

$$\begin{aligned} M(C; t) \quad = \quad & V(a_0; t) M_0(C_0; a_0) V^+(a_0; t) M_1(C_1; a_0) \\ & V(a_0; t) M_2(C_0; a_0) V^+(a_0; t) M_3(C_1; a_0) \end{aligned} \quad (1)$$

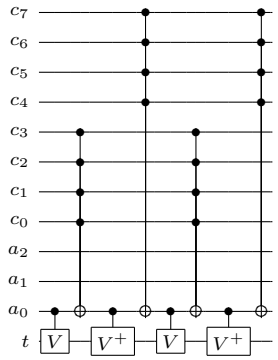

Fig. 5. Illustration of the decomposition in (1).

An example of this decomposition for 8 controls and 3 ancillary lines is illustrated in Fig. 5.

Fig. 1 is the optimal realization for a Toffoli gate. We also believe Fig. 3 is the optimal NCV realization for a 3-control MCT gate with 1 ancillary line. In particular, an extensive exhaustive search has not found a simpler circuit. These two circuits form the basis for our decomposition method which builds a catalog of NCV realizations of MCT gates for successively higher numbers of controls.

To determine a circuit with p controls and q ancillary lines $1 \leq q \leq p - 2$, our method proceeds as follows:

1) The lines are ordered as shown in Fig. 5, *i.e.* target line followed by the ancillary lines followed by the control lines.
2) The set of controls C ($|C| = p$) is partitioned as $C_0 C_1$ in all possible ways where $|C_0| >= 2$, $|C_1| >= 2$, and the order of the controls from C is preserved across C_0 and C_1. Note that there are $|C| - 3$ such partitionings. It is not necessary to try all permutations since reordering of the controls of an MCT gate has no effect on its operation.

3) For each partitioning of C, Equation (1) is applied. Gate M_0 is replaced by the optimized circuit from the catalog. The number of ancillary lines available is $q+|C_1|$. M_2 is replaced with the same circuit reversed with the V and V^+ gates interchanged. Likewise, gate M_1 is replaced by the optimized circuit from the catalog. The number of ancillary lines available is $(q-1)+|C_0|$. M_3 is replaced with the same circuit reversed with the V and V^+ gates interchanged.
4) The NCV circuit for each partitioning is simplified using the line labeling procedure described in the next subsection.
5) As the possible partitionings are tried, record is kept of which one leads to the circuit with the fewest NCV gates. That circuit becomes the catalog entry for p controls and q ancillary lines.

A critical factor is how the ancillary lines are assigned when replacing each M_i gate with a catalog circuit. For M_0 and M_2, t is used as the first ancillary, followed by the ancillary lines except a_0 in order followed by controls from C_1 as needed. For M_1 and M_3, the ancillary lines except a_0 are used in order followed by controls from C_0 as needed. Note that the target could be used as an ancillary for the M_1, M_3 pair, but this has been found to block simplifications involving the M_0, M_2 pair. Experimentation has shown this approach leads to the best circuits, but so far we have no proof that it is an optimal approach.

C. Line Labeling Procedure

Step 4 of the method proposed above makes use of a line labeling procedure. This procedure, presented in [16], [17], marks all the line segments between gates in a circuit in such a way that if two segments on a line have the same label, they represent the same function. A stack is used for each circuit line to keep a record of consecutive gates that together realize the identity.

To begin, all input lines are labeled 0. Then for each gate G in the circuit from the inputs towards the outputs the following steps are performed:

1) If there is a gate H_1 or a pair of gates H_1H_2 (H_2 above H_1) at the top of the stack S_t (t is the target line of G) which combined with G realizes the identity, then the target line t at the output side of G is assigned the label on t at the input side of H_1. Otherwise, set the increment flag.
2) Any set of two or three gates (containing V, V^+, CNOT) at the top of stack S_t that realizes the identity function is pulled from the stack and step (a) is repeated.
3) If the increment flag is not set, G is pushed onto stack S_t and the labeling procedure continues for the next gate.
4) If there is any occurrence of G or its inverse G^{-1} with the same labeling at its input side, the output side of G is marked accordingly. Otherwise, the output side of the target of the G is assigned the maximum label used on that line thus far plus one.

For more details and examples we refer to [16], [17].

TABLE I
NUMBER OF NCV GATES REQUIRED FOR MCT GATES FOR UP TO 15 CONTROL LINES.

		Number of Ancillary Lines					
		1	2	3	4	5	6
Number of Control Lines	3	14					
	4	20					
	5	32					
	6	44					
	7	64	56				
	8	76	68				
	9	96	88	80			
	10	108	100	92			
	11	132	120	112	104		
	12	156	132	124	116		
	13	180	156	148	136	128	
	14	204	180	172	148	140	
	15	228	204	198	172	160	152

D. Results

The procedures described above have been implemented using Python. Table I shows the results for up to 15 controls. The table is restricted to 15 controls for space reasons, the method applies to any number of controls. Note that in each row, allowing further ancillary lines does not reduce the size of the circuit. For example, we achieve the smallest circuit for 15 controls using only 6 ancillary lines – previously, 13 have been required.

To put these results in context, consider Table II showing the results presented in [4]. The procedures presented above yield circuits with significantly lower gate counts and require fewer ancillary lines. Table II also shows the gate counts for 1 ancillary line up to 10 controls as presented in [5] and commonly used in benchmark suites [6]. Once again, it is clear that the results presented here are significantly better. Note that the cost of 13 for 3 controls and 1 ancillary from [6] requires gates that realize the fourth root of NOT and not V and V^+, *i.e.* this particular result is not comparable here.

IV. NEAREST-NEIGHBOUR NCV CIRCUITS

Any NCV circuit can be made nearest-neighbour by inserting appropriate swap operations. The point of interest is how to do that in a minimal way.

A. A Nearest-neighbour NCV Toffoli Gate Realization

Consider the NCV realization of the Toffoli gate shown in Fig. 1. The four gates on the left are nearest-neighbour, but the rightmost gate is not since there is an unused line between the control and target lines. This circuit can be made nearest-neighbour by inserting two CNOT swap sequences as shown in Fig. 6 [12]. In the resulting cascade, the fourth gate and the fifth gate cancel giving the circuit shown in Fig. 7 which has cost 9. The SAT-based exhaustive synthesis procedure described in [12] has been used to verify that this is a minimal cost circuit. The right swap sequence can be omitted if it is not necessary to restore the line order.

This circuit represents eight distinct realizations. As before, the circuit can be reversed and the V and V^+

TABLE II
NUMBER OF NCV GATES REQUIRED FOR MCT GATES AS PRESENTED IN [4].

Number of Control Lines	Number of Ancillary Lines														
	[5] 1	[6] 1	1	2	3	4	5	6	7	8	9	10	11	12	13
3	15	13	14												
4	37	29	28	26											
5	54	52	48	40	38										
6	80	80	68	60	52	50									
7	100	100	92	80	72	64	62								
8	128	128	116	104	92	84	76	74							
9	152	152	140	128	116	104	96	88	86						
10	176	176	164	152	140	128	116	108	100	98					
11	—	200	188	176	164	152	140	128	120	112	110				
12	—	224	212	200	188	176	164	152	140	132	124	122			
13	—	248	236	224	212	200	188	176	164	152	144	136	134		
14	—	272	260	248	236	224	212	200	188	176	164	156	148	146	
15	—	296	284	272	260	248	236	224	212	200	188	176	168	160	158

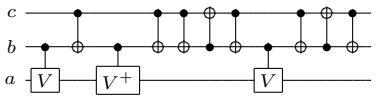

Fig. 6. Nearest-neighbour NCV realization of Toffoli gate $T(c, b; a)$.

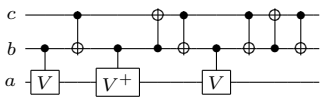

Fig. 7. Reduced Nearest-neighbour NCV realization of Toffoli gate $T(c, b; a)$.

gates can be interchanged. In addition, the rightmost three gates $CN(c; b), CN(b; c), CN(c; b)$ can be replaced by $CN(b; c), CN(c; b), CN(b; c)$.

B. Nearest-neighbour MCT Gate Realizations

Having shown how to transform a Toffoli gate to a nearest-neighbour NCV realization, we now consider how to apply the same approach to finding nearest-neighbour NCV realizations for MCT gates.

Consider the circuit in Fig. 3 (showing an NCV realization of a Toffoli gate with three controls). By adding swap sequences, this circuit can be converted to the nearest-neighbour realization shown in Fig. 8. Note that each swap sequence requires three CNOT gates. However, by choosing the swaps appropriately, pairs of gates cancel so that each swap results in only two CNOTs as shown in Fig. 8 where the gates implementing swaps are highlighted. The cost of the non-nearest-neighbour circuit is 14 while the cost of the nearest neighbour circuit is 26.

Applying the same methods to the circuit in Fig. 4 (*i.e.* a Toffoli gate with four controls), which has 20 gates, yields a circuit with 48 gates. In this case, far more than half the gates are required to make the circuit nearest-neighbour. The indication is that this will continue and get even worse as the number of control lines increases.

To further appreciate the complexity of the problem, consider the circuit in Fig. 9 which is the circuit from Fig. 4 with the lines reordered to reduce the number of nearest-neighbour violations. Applying the above methods to this circuit, with careful choice of CNOT swap sequences, yields the circuit in Fig. 10 which has 35 gates. Each of the three highlighted pairs can be replaced by a $V(a; t)$ gate giving a final gate count of 32.

V. CONCLUSIONS AND FUTURE WORK

The major contribution of this paper is a new approach to finding NCV realizations for MCT gates. As shown, the new decomposition approach leads to smaller circuits compared to the ones that have been used in the past. A major factor is that the circuits produced by the methods described here require considerably fewer ancillary lines. We emphasize that while the applied decomposition procedure is systematic and produces what appear to be very good circuits, we as yet have no proof that the circuits are optimal.

Our work to date on nearest-neighbour circuits shows two important things. First is that positioning of the target and the ancillary line (or ancillary lines) is important. It is not sufficient to require the target and controls of an MCT gate be clustered with no intervening lines. Second, extending our method for non-nearest-neighbour NCV realizations to the nearest neighbour case seems to generate quite expensive circuits. This needs to be further investigated. In particular, it should be considered whether the problem is in fact inherently complex or whether the problem lies in the nature of the decomposition given by (1). At the moment, we expect the latter to be true, since (1) was not developed with nearest-neighbour communication in mind. We anticipate that better circuits will be found by direct NCV circuit synthesis rather than through MCT gate based decomposition.

Finally, the work presented here has concerned NCV circuits but can be applied to other quantum gates. Our current work is considering whether higher-order root-of-NOT gates, *e.g.* fourth root, eighth root, *etc.*, will lead to smaller circuits. The applicability of that work will depend on the practicality of realizing the higher roots.

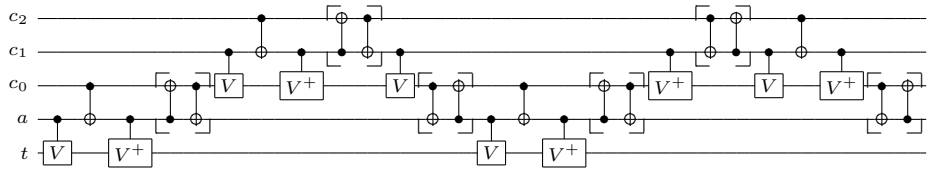

Fig. 8. Nearest-neighbour NCV circuit derived from Fig. 3.

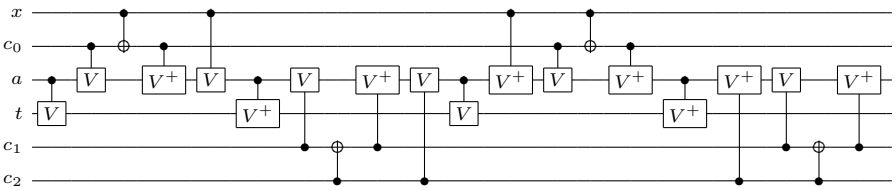

Fig. 9. Circuit from Fig. 4 with a different line ordering.

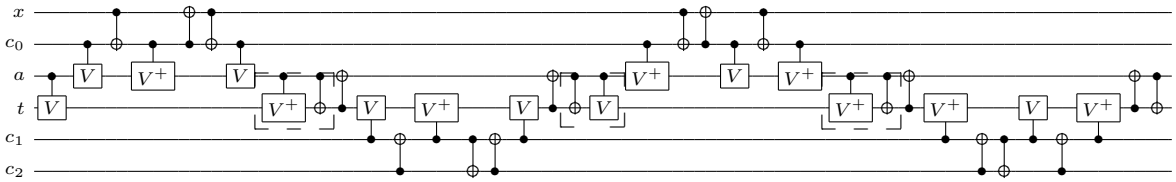

Fig. 10. NCV nearest-neighbour realization of an MCT gate with 4 controls and 1 ancillary line. The highlighted gate pairs can each be replaced by a V gate or a $V+$ gate, respectively.

VI. Acknowledgments

This work was supported in part by a Discovery Grant from the Natural Sciences and Engineering Research Council of Canada and by the German Academic Exchange Service (DAAD).

The authors thank referee 2 for a very detailed and constructive review.

References

[1] R. Wille and R. Drechsler, *Progress in Applications of Boolean Functions (Synthesis Lectures on Digital Circuits and Systems)*. Morgan and Claypool, 2010, ch. Synthesis of Boolean Functions in Reversible Logic.

[2] E. Fredkin and T. Toffoli, "Conservative logic," *Int'l J. of Theoretical Physics*, vol. 21, pp. 219–253, 1982.

[3] A. Barenco, C. Bennett, R. Cleve, D. DiVinchenzo, M. Margolus, P. Shor, T. Sleator, J. Smolin, and H. Weinfurter, "Elementary gates for quantum computation," *Physical Review A*, vol. 52, no. 5, pp. 3457–3467, Nov. 1995.

[4] D. M. Miller and Z. Sasanian, "Improving the ncv realization of multiple-control toffoli gates," in *Proc. 9th Int'l Workshop on Boolean Problems*, 2010, pp. 37–44.

[5] D. Maslov, C. Young, D. M. Miller, and G. W. Dueck, "Quantum circuit simplification using templates," in *Proc. Design, Automation and Test in Europe*, 2005, pp. 1208–1213.

[6] R. Wille, D. Große, L. Teuber, G. W. Dueck, and R. Drechsler, "RevLib: An online resource for reversible functions and reversible circuits," in *Int'l Symp. on Multi-Valued Logic*, 2008, pp. 220–225, RevLib is available at www.revlib.org.

[7] A. Chakrabarti and S. Sur-Kolay, "Neareat neighbour based synthesis of quantum boolean circuits," *Engineering Letters*, vol. 15, no. 2, 2007.

[8] M. H. A. Khan, "Cost reduction in nearest neighbour based synthesis of quantum boolean circuits," *Engineering Letters*, vol. 16, no. 1, 2008.

[9] A. Chakrabarti and S. Sur-Kolay, "Rules for synthesizing quantum boolean circuits using minimized nearest-neighbor templates," in *15th International Conference on Advanced Computing and Communications*, 2007, pp. 183–188.

[10] Y. Hirata, M. Nakanishi, S. Yamashita, and Y. Nakashima, "An efficient method to convert arbitrary quantum circuits to ones on a linear nearest neighbor architecture," pp. 26–33, 2009.

[11] S. Yamashita and I. L. Markov, "Fast equivalence-checking for quantum circuits," in *Proceedings of the 2010 IEEE/ACM International Symposium on Nanoscale Architectures*, ser. Nanoarch '10. Piscataway, NJ, USA: IEEE Press, 2010, pp. 23–28. [Online]. Available: http://portal.acm.org/citation.cfm?id=1835957.1835965

[12] M. Saeedi, R. Wille, and R. Drechsler, "Synthesis of quantum circuits for linear nearest neighbor architectures," *Quantum Information Processing*, pp. 1–23, 2010, 10.1007/s11128-010-0201-2. [Online]. Available: http://dx.doi.org/10.1007/s11128-010-0201-2

[13] R. Wille, D. Große, D. M. Miller, and R. Drechsler, "Equivalence checking of reversible circuits," in *Proc. Int'l Symp. on Multiple-valued Logic (CD)*, 2009, pp. 324–330.

[14] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information*. Cambridge Univ. Press, 2000.

[15] T. Toffoli, "Reversible computing, Tech Memo LCS/TM-151, MIT Lab for Comp. Sci," 1980.

[16] D. M. Miller and Z. Sasanian, "Lowering the quantum gate cost of reversible circuits," in *Proc. Midwest Symp. on Circuits and Systems*, 2010, pp. 260–263,.

[17] Z. Sasanian and D. M. Miller, "Mapping a multiple-control toffoli gate cascade to an elementary quantum gate circuit," in *Proc. Workshop on Reversible Computation*, 2010, pp. 83–90.