A Pre-Optimization Technique to Generate Initial Reversible Circuits with Low Quantum Cost

Nurul Ain Binti Adnan, Kouhei Kushida and Shigeru Yamashita
Graduate School of Information Science and Engineering
Ritsumeikan University
1-1-1 Noji Higashi, Kusatsu, Shiga 525-8577, Japan
Email: {nu_ain, kino, ger}@ngc.is.ritsumei.ac.jp

***Abstract*—In order to generate an initial reversible/quantum circuit to realize a given Boolean function, one of the major approaches is to find a small Exclusive-or Sum-Of-Products (ESOP) expression for the function; each product term in the ESOP expression naturally corresponds to a single Mixed Polarity Multiple-Control Toffoli (MPMCT) gate. In this paper, we propose a technique to perform *pre-optimization* before generating an ESOP expression. Our approach is unique in a sense that instead of finding a small ESOP expression for a given function, we try to *change* it by adding MPMCT gates so that the modified function has a smaller ESOP expression. We expect that our approach can generate a better initial circuit compared to using ESOP minimization techniques only. Indeed our preliminary experiments for small functions confirm this expectation. Thus, we expect that our approach would be a good pre-optimization technique that can be used with most existing reversible circuit synthesis techniques.**

I. Introduction

To demonstrate the ability of quantum computing in the near future, an efficient quantum algorithm should be implemented efficiently. In general, a quantum algorithm includes a part to calculate Boolean functions corresponding to a problem instance. Thus, an efficient design technique for realization of a Boolean function is very crucial even for quantum circuits, as pointed out in the literature (e.g., [7]). Therefore, the design methodology of reversible circuits has been studied very extensively in the reversible computation as well as quantum computation research communities [5].

There are many ways to design a reversible circuit to calculate a Boolean function; one of the most popular ways is to generate an initial circuit consisting of Mixed Polarity Multiple-Control Toffoli (MPMCT) gates [1], [2] based on an small Exclusive-or Sum-Of-Products (ESOP) expression [3], and then decompose a large gate (i.e., with the large number of inputs) into elementary gates. After getting an initial circuit, there have been various *post-optimization* methods, e.g., library-based, transformation-based and template-based optimization method [5].

Our proposed method in this paper can be easily explained by using ESOP expressions. Thus we consider (ESOP)-based initial circuit design methods in the following. Note that there are other approaches that are not based on ESOP expressions (e.g., BDD-based and cycle-based methods [5]). When we design initial circuits based on ESOP expressions, an important task is to find a *small* (ESOP) expression for a given Boolean function because we can generate a reversible circuit for a Boolean function by concatenating a Mixed Polarity Multiple-Control Toffoli (MPMCT) gate corresponding to each product term in the ESOP expression (as we will explain below). For example, the previous work [1] proposes a method to find a small reversible circuit from a given specification by essentially finding a small ESOP expression.

In this paper, we propose a different approach; instead of finding a small ESOP expression for a given specification directly, we try to change the given specification by adding MPMCT gates so that the modified specification has a smaller quantum circuit implementation. This approach can be considered as *pre-optimization* unlike existing methods. We expect that our approach may generate a better initial circuit than the previous methods; indeed our preliminary experiments confirm this expectation.

II. Preliminaries and Previous Work

A. Quantum Cost

In the research community of the quantum computation, it is usually assumed that elementary gates in the quantum computation cannot have more than two qubit interactions. Therefore, in most researches on the quantum circuit design, the cost of a quantum gate with many inputs is defined as the number of basic (i.e., less than three inputs) gates to realize the gate. That is often called as a *quantum cost.*

In this paper, we take the quantum cost from the previous work [6] as shown in Table I. The table shows the cost of an MPMCT gate with m control bits when $m-3$ auxiliary bits are available. Even if we have more efficient ways to realize an MPMCT gate, our framework does not change; we simply reduce the cost values accordingly.

B. Realizing Boolean Functions with MPMCT Gates

A **minterm** of a Boolean function is the combination of all the input variables (negative or positive) when the Boolean function becomes one. In the following, an $MPMCT_n$ gate means an MPMCT gate that has n control bits.

To realize an n-input Boolean function with k minterms by a reversible circuit, one possible way is to put k $MPMCT_n$ gates such that (1) each $MPMCT_n$ gate corresponds to each minterm of the function, and (2) the polarity of each control bit for an MPMCT gate corresponds to each variable's polarity in the corresponding minterm. In other words, if x_i or $\overline{x_i}$ appears in a minterm, the corresponding control bit is positive

978-1-4799-5341-7/16/$31.00 ©2016 IEEE

TABLE I. QUANTUM COST OF MPMCT GATES WITH m CONTROL BITS $(0 \leq m \leq 15)$

Number of Control Bits m	Number of Negative Controls															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	1															
1	1	2														
2	5	5	6													
3	14	14	16	18												
4	20	20	20	22	24											
5	32	32	32	34	36	38										
6	44	44	44	44	46	48	50									
7	56	56	56	56	58	60	62	64								
8	68	68	68	68	68	70	72	74	76							
9	80	80	80	80	80	82	84	86	88	90						
10	92	92	92	92	92	92	94	96	98	100	102					
11	104	104	104	104	104	104	106	108	110	112	114	116				
12	116	116	116	116	116	116	116	118	120	122	124	126	128			
13	128	128	128	128	128	128	128	130	132	134	136	138	140	142		
14	140	140	140	140	140	140	140	140	142	144	146	148	150	152	154	
15	152	152	152	152	152	152	152	152	154	156	158	160	162	164	166	168

TABLE II. A TRUTH TABLE FOR A 4-INPUT BOOLEAN FUNCTION WITH 4 MINTERMS

x_1	x_2	x_3	x_4	$f(x)$
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

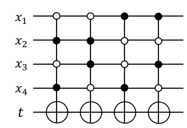

Fig. 1. The Quantum Circuit for Table II

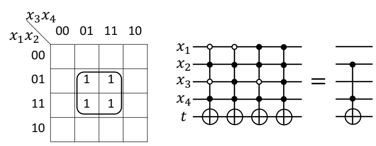

Fig. 2. Optimization for a Group with 2^2 Cells

or negative, respectively. In this construction, the target bit of all the $MPMCT_n$ gates is the same as the qubit where we want to realize the function.

For instance, Table II shows a 4-input Boolean function with 4 minterms, and the circuit in Fig. 1 realizes the function: $x_2 \cdot x_4 \cdot \overline{x_1} \cdot \overline{x_3} \oplus x_2 \cdot x_3 \cdot \overline{x_1} \cdot \overline{x_4} \oplus x_1 \cdot x_4 \cdot \overline{x_2} \cdot \overline{x_3} \oplus x_1 \cdot x_3 \cdot \overline{x_2} \cdot \overline{x_4}$. For example, the left most gate in Fig. 1 corresponds to $x_2 \cdot x_4 \cdot \overline{x_1} \cdot \overline{x_3}$; the control bits for x_2 and x_4 are in the positive polarities denoted by black circles, and x_1 and x_3 are in the negative polarities denoted by white circles.

C. Initial Circuit Generation by Minimizing ESOP Expression

To understand our proposal in this paper, it is beneficial to understand how we can design a reversible circuit by minimizing ESOP expressions. So, in the following, we review a design method proposed by Arabzadeh et. al. [1]; the method can find a circuit with a low quantum cost based on some rules by using a Karnaugh map (Kmap, hereafter) to minimize an ESOP expression.

In the following, we refer to a blank cell or a cell having the 0 value as **0-value cell** in a Kmap. Also, a cell having the 1 value is called **1-value cell**. For a minterm with n input variables, it can be expressed as one $MPMCT_n$ gate. Also one 1-value cell corresponds to one minterm in a Boolean function. Therefore, in a Kmap for an n-input function, one 1-value cell corresponds to one $MPMCT_n$ gate.

In the classical logic design, we can optimize a two-level AND/OR/NOT expression of a function by manipulating the Kmap for the function with some rules. Arabzadeh et. al. proposed a similar optimization method for quantum circuits based on Kmaps [1]; we refer to the method as the Arabzadeh method in the following. In the Arabzadeh method, they apply the so-called CTRs (Common-Target Rules) to quantum circuits; the rules are stated as follows.

(a) All 1-value cells, should at least belong to one *group* of cells.
(b) Each group has a size of 2^p $(p \geq 0)$.
(c) Each group should have a maximum size.
(d) Each 1-value cell should belong to an odd number of groups.
(e) Each 0-value cell should belong to an even number of groups.
(f) The number of groups should be minimum (as much as possible).

A *group* is a collection of cells that are next to each other, and denoted by a rectangular inside a Kmap. The above rules indicates that we can also cover 0-value cells when we consider the optimization of a quantum circuit; this is in contrast to the case of a two-level AND/OR/NOT expression where we focus only on 1-value cells.

A group in a Kmap is used to optimize a part of an expression of a certain Boolean function. For instance, 2^p $(p \geq 0)$ cells correspond to 2^p $MPMCT_n$; if they can be in one group of 2^p $(p \geq 0)$ cells, the 2^p $MPMCT_n$ can be optimized to a single $MPMCT_{n-p}$ gate. Thus, we can reduce the quantum cost drastically.

Figure 2 shows how this rule is applied to optimize a circuit. The four cells in a Kmap on the leftmost in Fig. 2 belong to a group that can be described as $\overline{x_1}x_2\overline{x_3}x_4 \oplus \overline{x_1}x_2x_3x_4 \oplus x_1x_2\overline{x_3}x_4 \oplus x_1x_2x_3x_4$. This corresponds to the circuit (in the middle of the same figure) that has four $MPMCT_4$ gates. After the optimization by using the above rules, we can optimize the function to be x_2x_4, and thus we can get the optimized circuit on the rightmost in the same figure. By looking up the quantum cost listed in the literature [6], the quantum cost is reduced from 80 to only 5 in this example. Figure 3 shows typical three groups of cells that should be transformed into a single gate, and Fig. 4 shows the equivalence of two quantum circuits by considering the three transformations.

In short, the quantum cost can be reduced by eliminating more control bits of MPMCT gates in a circuit. When the size of a group (of adjacent cells) is 2^p, the number of gates after optimization becomes $\frac{1}{2^p}$, and the number of control bits become $\frac{n-p}{n}$. Thus, when the size of a group is larger, we can reduce the quantum cost by using the above optimization strategy.

For the Boolean function shown in Table II, if we make

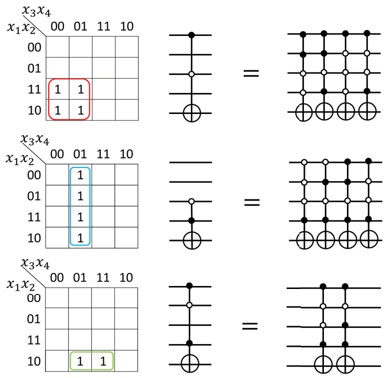

Fig. 3. Transforming 3 Groups to Quantum Gates

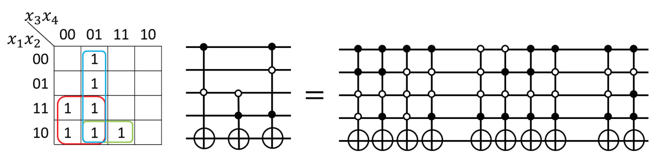

Fig. 4. Realization of Fig. 3 to a Quantum Circuit

groups using a Kmap as shown in Fig. 5, we are able to realize the quantum circuit as shown in Fig. 6 as a result. In the above example, the quantum cost is reduced from 80 to 20. Thus, in most cases, we can reduce the quantum cost by using the Arabzadeh method. In the next section, we will propose another way to reduce the quantum cost; our method can reduce the cost of the same example to only 9.

III. Reducing Cost by Inserting MPMCT Gates

A. *Our Idea with a Motivational Example*

When we utilize the Arabzadeh method, the location of a 1-value cell always stays at the initial point as a given specification. It seems to be natural to do so, but the initial locations of 1-value cells may cause bad results for some cases. For example, for a Kmap as shown in Fig. 7, the method can only generate the circuit as shown in Fig. 8; we can find a

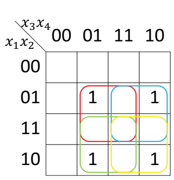

Fig. 5. Group Patterns for Kmap for Table II

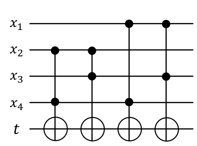

Fig. 6. The Quantum Circuit for Fig. 5

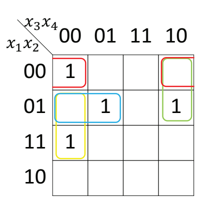

Fig. 7. A Bad Case for Arabzadeh Method

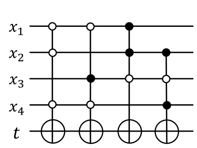

Fig. 8. The Quantum Circuit for Fig. 7

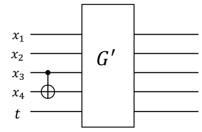

Fig. 9. Insertion of a CNOT Gate before G_1.

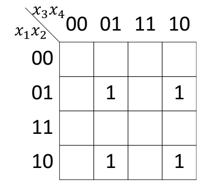

Fig. 10. The Kmap before the Insertion of the First Gate

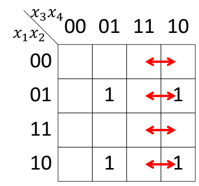

Fig. 11. The Movement of Cells by the Insertion of the First CNOT Gate

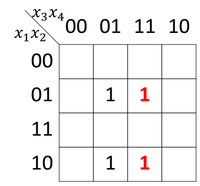

Fig. 12. The Kmap after the Insertion of the Fist CNOT Gate

much better circuit by using our pre-optimization technique as follows.

Our proposal in this paper involves moving 1-value cells so that the resulting Kmap can be implemented by a reversible circuit with a much lower quantum cost. In order to do so, we insert MPMCT gates to the initial quantum circuit. Note that inserting MPMCT gates changes the functionality of the quantum circuit. Therefore we also add the corresponding MPMCT gates at the end in order to cancel the effect so that we can have the desired functionality as we will see in the following.

Let us explain what we mean by "Inserting an MPMCT gate" in the following example. Let G be a quantum circuit realizing the Boolean function as shown in Table II. In our proposed method, we insert an MPMCT gate whose positive control bit is x_3 and the target bit is x_4 (i.e., CNOT gate) before G' as shown in Fig. 9. Our idea is as follows: if we implement a circuit G' whose Kmap is as shown in Fig. 12, the entire circuit (i.e., G' with the inserted CNOT) realizes the original function (of G) whose Kmap is as shown in Fig. 10. The reason is as follows: the inserted CNOT (the control bit is x_3 and the target bit is x_4) inverts the value of x_4 when $x_3 = 1$. This means that the gate changes the input state $(x_1, x_2, x_3, x_4) = (0110)$ to (0111), for example. Also, the gate moves (1010) to (1011). More precisely, the gate swap four pairs of cells in a Kmap as shown in Fig. 11. Therefore, we can conclude that by inserting the gate, it is enough for G' to realize the function as shown in Fig. 12 to realize the desired function as shown in Fig. 10.

Similarly, if we insert the second CNOT gate as shown in Fig. 13, the sub-circuit G'' in Fig. 13 should realize the

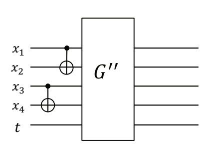

Fig. 13. The Insertion of the Second CNOT Gate

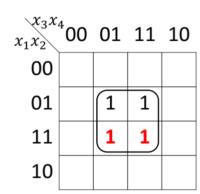

Fig. 14. The Kmap after the Insertion of Two CNOT Gates

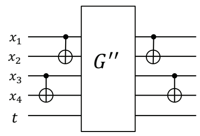

Fig. 15. The Insertion of Two CNOT Gates to Restore the Original Function

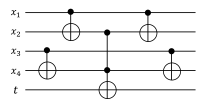

Fig. 16. The Final Quantum Circuit by Our Method: Cost 9

function whose Kmap is as shown in Fig. 14. This is because the second gate move the minterm (1001) to (1101), and (1011) to (1111). Thus G'' should implement the function as Fig. 14, which can be done by just a single Toffoli gate.

Note that the resultant states of qubits after the circuit in Fig. 13 are not exactly the same as the ones of the desired circuit as shown in Fig. 6, because we changed the functionality of x_2 and x_4 by inserting the two MPMCT gates. Therefore, we then insert the same MPMCT gates after G'' at the end of the circuit as shown in Fig. 15.

In this example, we can get the circuit as shown in Fig. 16 whose quantum cost is 9. It should be noted that we get a circuit with a quantum cost of 20 for this example if we simply use the Arabzadeh method only.

B. How to Insert MPMCT Gates for a Good Pre-Optimization

The above-mentioned example shows that there is a case where inserting some MPMCT gates improves the quantum cost. As we explained, inserting an MPMCT gate corresponds to the movement of 1-value cells in a Kmap, i.e., changing the specification of a given function. Thus, such a change of the specification can be considered as *pre-optimization* before generating a quantum circuit.

For simplicity, we consider an n-input function having k $(= 2^p)$ 1-value cells here. If k is small, we can try all the possible movements of 1-value cells as follows.

- We select one cell from all the k 1-value cells, and move all the other 1-value cells so that all k 1-value cells form a group of adjacent cells. By this movement, the function is changed so that it can be realized by only one $MPMCT_{(n-p)}$ gate.
- The number of the above possible movements is $k! \times {}_nC_p$. Thus, if this number is not large, we can try all the possible movements to find the best movement corresponding to the best insertion of MPMCT gates with the lowest quantum cost.

Obviously, we can have various heuristics for large functions instead of using the above naive exhaustive search.

C. Preliminary Experimental Results

To check the validity of the above method, we applied the method to all the 4-input functions with four minterms, i.e., we tried to find the best insertion with the lowest quantum cost for each of ${}_{16}C_4$ = 1820 functions.

To evaluate the effectiveness of our pre-optimization technique, we also tried to find a good quantum circuit for each function by finding a small ESOP expression. Note that this method should be essentially similar to designing reversible circuits by minimizing ESOP expressions, e.g., Arabzadeh method. To get small ESOP expressions, we used EXORCISM4 [4] which is considered to be a state-of-the-art ESOP minimizer.

TABLE III. Preliminary Experimental Results

Quantum Cost	EXORCISM4 [4]	Proposed Pre-Optimization
$0-20$	292	1096
$21-40$	968	724
$41-60$	560	0
$61-80$	0	0
Average Cost	31.6	20.2

Table III shows the comparison between our method and the method based on the ESOP minimization by EXORCISM-4 [4]. The table shows the number of functions that are generated by the two methods, divided into three groups in terms of the quantum costs. As we can see from this table, our proposed method succeeds to design most of the given quantum circuits with lower quantum costs. We confirmed that our proposed method achieves lower quantum cost for about 97.6% (1776 Boolean functions) of all the functions compared to the EXORCISM4 program. In average, our method can reduce the cost for 36% of the circuits.

IV. Conclusion

This paper has proposed a pre-optimization technique to change a given function so that the quantum cost of the final circuit may become smaller. We tried only 4-input functions with four minterms in our preliminary experiments; our obvious future work is to try larger functions. To do so, we need to develop an efficient heuristic to find a good movement of cells for larger cases. Note that our pre-optimization technique may be used with most of reversible circuit design methods; we need to study how we can utilize our technique with other existing design methods.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number 24106009 and 15H01677.

References

[1] Mona Arabzadeh, Mehdi Saeedi, and Morteza Saheb Zamani. Rule-based optimization of reversible circuits. In *Design Automation Conference (ASP-DAC), 2010 15th Asia and South Pacific*, pages 849–854. IEEE, 2010.

[2] Kamalika Datta, Gaurav Rathi, Indranil Sengupta, and Hafizur Rahaman. An improved reversible circuit synthesis approach using clustering of esop cubes. *J. Emerg. Technol. Comput. Syst.*, 11(2):15:1–15:16, November 2014.

[3] K Fazel, M Thornton, and JE Rice. Esop-based toffoli gate cascade generation. In *IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, pages 206–209. Citeseer, 2007.

[4] Alan Mishchenko. Two-level exclusive sum-of-product minimization. http://web.cecs.pdx.edu/~alanmi/research/min/minEsop.htm.

[5] Mehdi Saeedi and Igor L. Markov. Synthesis and optimization of reversible circuits—a survey. *ACM Comput. Surv.*, 45(2):21:1–21:34, March 2013.

[6] Zahra Sasanian and D. Michael Miller. Reversible and quantum circuit optimization: A functional approach. In *Reversible Computation 4th International Workshop, RC 2012, Copenhagen, Denmark, July 2-3, 2012. Revised Papers,*, pages 112–124, 2013.

[7] Yamashita Shigeru, Minato Shin-ichi, and Miller D.Michael. DDMF: An Efficient Decision Diagram Structure for Design Verification of Quantum Circuits under a Practical Restriction. *IEICE Transactions on Fundamentals of Electronics,Communications and Computer Sciences*, pages 3793–3802, 2008.