Controlled and Uncontrolled SWAP Gates in Reversible Logic Synthesis

Md Asif Nashiry[1(✉)], Mozammel H.A. Khan[2], and Jacqueline E. Rice[1]

[1] Department of Mathematics and Computer Science, University of Lethbridge, Lethbridge, AB T1K 3M4, Canada
{asif.nashiry,j.rice}@uleth.ca

[2] Department of Computer Science and Engineering, East West University, Aftabnagar, Dhaka 1212, Bangladesh
mhakhan@ewubd.edu

Abstract. This paper presents a quantum-level realization and synthesis approach using SWAP and Fredkin (SF) gates. Our quantum realization of negative-controlled Fredkin gate requires five 2-qubit elementary quantum gates, the same as that required for realizing a positive-controlled Fredkin gate. We also propose and evaluate the performance of a synthesis approach using SF gates for realizing conservative reversible functions. Our result shows that circuit realization for conservative function using SF gates is more efficient than Toffoli gates. We achieve up to 87% improvement in gate count and quantum cost for (4×4) conservative reversible functions.

Keywords: Reversible logic · SWAP gate · Fredkin gate · Toffoli gates · Mixed polarity gates · Quantum gates · Logic synthesis · Conservative functions

1 Introduction

A logic gate is a reversible gate if the output function of the gate is bijective [1]. The two most widely used reversible logic gate families are NOT-CNOT-Toffoli (NCT) and SWAP-Fredkin (SF). A SWAP gate is a (2×2) reversible logic gate which interchanges the input bits at the output. Fredkin and Toffoli proposed a reversible controlled swap gate (also called Fredkin gate) in [2]. This gate is a positive-controlled gate i.e. it swaps the two target inputs when the control input is 1. The authors showed that it is a universal gate and thus any reversible circuit can be synthesized using only Fredkin gates. One example of this is in [4], where Bruce et al. proposed a design for a full-adder using five positive-controlled Fredkin gates.

Smolin and DiVincenzo presented an implementation of the positive-controlled Fredkin gate using five 2-qubit elementary quantum gates in [3]. We propose a realization of the negative-controlled Fredkin gate, that like the positive-controlled Fredkin gate, requires five 2-qubit elementary quantum gates.

I. Phillips and H. Rahaman (Eds.): RC 2017, LNCS 10301, pp. 141–147, 2017.
DOI: 10.1007/978-3-319-59936-6_11

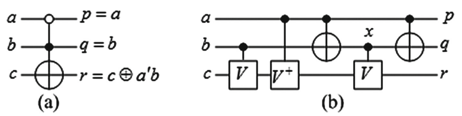

Fig. 1. (a) Toffoli gate with top negative control and bottom positive control and (b) its realization.

Table 1. Behaviour of the circuit shown in Fig. 1(b).

Control		Target	Output
a	b	c	r
0	0	c	c
0	1	c	c'
1	0	c	c
1	1	c	c

We also propose a transformation based synthesis algorithm using SF gates for realizing conservative reversible functions. A conservative reversible function has the same number of 1s in both the input and output vectors of the function.

2 Realization of Negative-Controlled Fredkin Gate

To realize our proposed negative-controlled Fredkin gate we use a Toffoli gate with top negative control and bottom positive control as an intermediate gate. Realization of a Toffoli gate with top positive control and bottom negative control is presented in [5]. This realization requires five 2-qubit elementary quantum gates. We follow this technique and present a realization of a Toffoli gate with top negative control and bottom positive control in Fig. 1(b), also requiring five 2-qubit elementary quantum gates. The symbol and realization of a negative-controlled Fredkin gate are shown in Fig. 2. The two target inputs are only swapped at the target outputs when the control input $a = 0$ (Table 1). If the

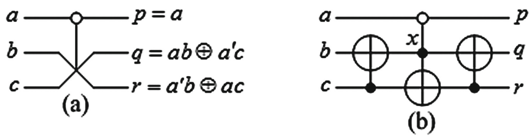

Fig. 2. (a) Negative-controlled Fredkin gate and (b) its realization with two Feynman gates and one Toffoli gate with top negative control and bottom positive control.

Toffoli gate shown in Fig. 2(b) is decomposed using the realization illustrated in Fig. 1(b), and the last two Feyman gates are rearranged using the equivalence shown in Fig. 3, the result is the circuit in Fig. 4. The operation of the two gates in a dashed box can be expressed using 4×4 unitary matrices. As these two gates are in cascade, their final operation will be another 4×4 unitary matrix. Therefore, the two gates in practice work as one 2-qubit quantum gate. Thus, the realization of the negative-controlled Fredkin gate requires five 2-qubit elementary quantum gates. A similar argument is used in [3].

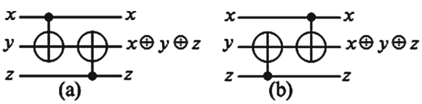

Fig. 3. Circuits of (a) and (b) are equivalent.

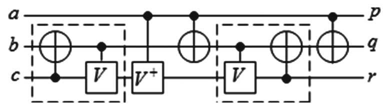

Fig. 4. Realization of a negative-controlled Fredkin gate.

3 SF Based Synthesis Approach

The basic working principle of the transformation based synthesis algorithm is to apply reversible operations to a reversible function in order to generate an identity function. The first such algorithm was proposed by Miller et al. [6]. The authors applied gates from the NCT gate library. In the basic algorithm the reversible logic operations are applied to the output of the function's truth table. The following is the basis of transformation based logic synthesis approach.

Table 2. Truth table of a (3×3) reversible function.

	Input			Output			
	a_i	b_i	c_i	a_o	b_o	c_o	
(0)	0	0	0	0	0	0	(0)
(1)	0	0	1	1	0	0	(4)
(2)	0	1	0	0	0	1	(1)
(3)	0	1	1	0	1	1	(3)
(4)	1	0	0	0	1	0	(2)
(5)	1	0	1	1	0	1	(5)
(6)	1	1	0	1	1	0	(6)
(7)	1	1	1	1	1	1	(7)

Step 0: If $f(0) = 0$, no transformation is required; go to step 1. If $f(0) \neq 0$, apply a (1×1) Toffoli gate (NOT gate) in order to achieve $f(0) = 0$.

Step 1: For $1 \leqslant i < 2^m - 1$: If $f(i) = i$, no transformation is required and proceed to next i. If $f(i) \neq i$, apply the smallest $(k \times k)$ Toffoli gate, $k = 2$ to n in order to make $f^i(i) = i$.

The choice of gate during each step of transformation is crucial in order to maintain convergence. The gate chosen in each step of transformation must not change the order of bits of the previous steps. Consider the (3×3) reversible function $f = \sum(0, 4, 1, 3, 2, 5, 6, 7)$ in Table 2. The circuit which is generated by following the basic transformation algorithm is presented in Fig. 5.

The basic premise of SF-based transformation synthesis is the same as that presented in [6]; however instead of using logic gates from the NCT gate family

Table 3. Transformation stages of the function in Table 2 using SF based transformation.

Output			Step 0			Step 1			Step 2			Step 3			Step 4		
			(i)			(ii)			(iii)			(iv)			(v)		
a	b	c	a^0	b^0	c^0	a^1	b^1	c^1	a^2	b^2	c^2	a^3	b^3	c^3	a^4	b^4	c^4
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0	1	0	0	1	0	0	1	0	0	1
0	0	1	0	0	1	1	0	0	0	1	0	0	1	0	0	1	0
0	1	1	0	1	1	1	1	0	1	1	0	0	1	1	0	1	1
0	1	0	0	1	0	0	1	0	1	0	0	1	0	0	1	0	0
1	0	1	1	0	1	1	0	1	0	1	1	1	1	0	1	0	1
1	1	0	1	1	0	0	1	1	1	0	1	1	0	1	1	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
			—			S(a,c)			S(a,b)			F(b;a,c)			F(a;b,c)		

we use only SWAP and Fredkin gates. We use the same function from Table 2 to demonstrate the SF-based transformation synthesis. We also use the simple one direction transformation for this example. Table 3 shows the transformation stages. The resulting circuit realization of the function from Table 2 is displayed in Fig. 6.

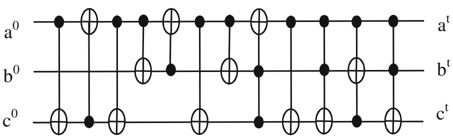

Fig. 5. Basic transformation synthesis for the function in Table 2.

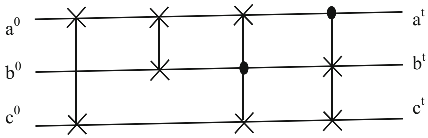

Fig. 6. SF based synthesis for the function in Table 2.

4 Comparison of NCT and SF Based Synthesis Approaches

It is important to observe that the function in Table 2 is a conservative function and Figs. 5 and 6 show two circuit designs for this function. In Fig. 6, we have a gate count of 4 as compared to a gate count of 12 for the circuit in Fig. 5. The quantum cost of the implementation in Fig. 6 is $(2 \times 3) + (2 \times 5) = 16$, where the quantum cost for the circuit realization in Fig. 5 is 28. The percentages of decrease in gate count and quantum cost are 67% and 43% respectively, which is a very significant improvement.

In order to compare the SF based transformation approach with NCT based transformation from a wider perspective, we have generated all possible (3×3) conservative reversible functions. We have realized all 36-(3×3) conservative functions using both algorithms. The highest percentage of reduction in gate count is 67% for more than half of the (3×3) conservative reversible functions. The ability of changing two bits at a time gives SF gates an advantage over the NCT gate family for realizing conservative reversible circuits.

SF based synthesis also performs better than NCT based synthesis when comparing quantum cost. Among the 36 functions, we have achieved lower QC for almost 70% of the functions. For the remaining functions, the QC is the same for both approaches. There is not a single instance where the NCT based synthesis performs better than our proposed approach. The highest percentage of decrease in quantum cost is 70% and the average percentage of reduction of quantum cost is 29%.

As mentioned above, the proposed transformation algorithm using the SF gate family follows the greedy approach. We have designed our algorithm in this way in order to offer a fair comparison, since the basic transformation based synthesis algorithm which is proposed in [6] also follows the greedy approach. At every step of transformation, the algorithm selects a gate which costs less in terms of quantum cost. For example, if we observe column (ii) of Table 3, we need to transform 100 into 010. There are two choices for this mapping. We could use either a SWAP gate S(a,b) or a negative controlled Fredkin gate, $F^{'}(a, b; c)$. The proposed SF gate based transformation selects a SWAP gate, S(a,b) because a SWAP gate has lower quantum cost than a Fredkin gate. However, if we use a $F^{'}(a, b; c)$ at this stage, we get a circuit which is presented in Fig. 7. The use of $F^{'}(a, b; c)$ gate reduces the quantum cost from 16 to 13 as we compared with the circuit in Fig. 6. Moreover, one less gate is needed in this circuit realization. The circuit in Fig. 8 is even more simplified design for the reversible function from Table 2. Figure 8 shows that the gate count is 2 and the quantum cost is 10. Now if we compare the gate count and quantum cost of Fig. 8 with that of the NCT gate based basic transformation synthesis (Fig. 5), the gate count has been reduced from 12 to 2, a 6 times reduction. The quantum cost has been reduced from 28 to 10, which is an improvement of almost a factor of 3. We have also generated all possible 414720 conservative (4×4) reversible function. However unlike the case of (3×3) functions, there are some circuit realizations where the gate count and quantum cost increase when using SF

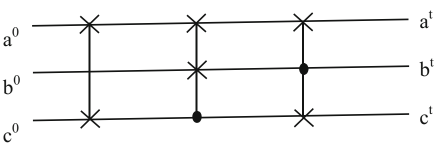

Fig. 7. Another circuit realization for the function from Table 2.

Fig. 8. More efficient circuit realization for the function from Table 2.

gate based transformation synthesis. Among all the (4×4) conservative reversible functions, the quantum cost increases for 27213 (6.5%) functions and the gate count increases for 2 functions. The highest percentage of reduction in gate count by using our proposed synthesis algorithm is 87% and the reduction in gate count, on average, is 61%. We achieve the highest percentage of reduction of quantum cost is 87%. The average percentage of decrease of quantum cost over all 414720 functions is 35%.

5 Conclusion

The contribution of this work is twofold. First, we present a unique realization of a negative-controlled Fredkin gate using five 2-qubit elementary quantum gates. Secondly, we propose a transformation based synthesis algorithm using SF gates for the realization of conservative reversible functions. After applying our approach to all possible (3×3) and (4×4) conservative functions we see that the synthesis of conservative reversible functions using SF gates is more efficient than using NCT gates. For (3×3) functions we show reductions in GC and QC of 67% and 70% respectively, while for (4×4) functions we achieve even higher reductions of 87% in both GC and QC. We also show that the percentage of reduction in GC and QC can be further improved by choosing gates intelligently instead of by following a greedy approach. Finally, this paper shows the usefulness of a negative control Fredkin gate in circuit realization.

The outcome of this work indicates that the synthesis process in reversible logic could be more efficient if we knew the class of a reversible function in advance. Therefore, classifying reversible functions and using the benefits of SF-gates in circuit realization for different classes of functions will be an important area of further research.

Acknowledgment. The second author was involved in this work at the University of Lethbridge, Lethbridge, AB, Canada while on sabbatical from East West University, Dhaka, Bangladesh. The first and third author carried out this work with the support of NSERC.

References

1. Shende, V.V., Prasad, A.K., Markov, I.L., Hayes, J.P.: Synthesis of reversible logic circuits. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. **22**(6), 710–722 (2003)
2. Fredkin, E., Toffoli, T.: Conservative logic. In: Andrew, A. (ed.) Collision-Based Computing, pp. 47–81. Springer, London (2002)
3. Smolin, J.A., DiVincenzo, D.P.: Five two-bit quantum gates are sufficient to implement the quantum Fredkin gate. Phys. Rev. A **53**(4), 2855 (1996)

4. Bruce, J.W., Thornton, M.A., Shivakumaraiah, L., Kokate, P.S., Li, X.: Efficient adder circuits based on a conservative reversible logic gate. In: Proceedings of IEEE Computer Society Annual Symposium on VLSI 2002, pp. 83–88 (2002)
5. Maslov, D., Miller, D.M.: Comparison of the cost metrics for reversible and quantum logic synthesis. arXiv preprint quant-ph/0511008 (2005)
6. Miller, D.M., Maslov, D., Dueck, G.W.: A transformation based algorithm for reversible logic synthesis, in Proceedings on Design Automation Conference, pp. 318–323. IEEE (2003)