Post Synthesis-Optimization of Reversible Circuit using Template Matching

Chandan Bandyopadhyay[1], Robert Wille[2], Rolf Drechsler[1], Hafizur Rahaman[3]
[1]Institute of Computer Science, University of Bremen & Cyber-Physical Systems, DFKI GmbH, 28358 Bremen, Germany
[2]Indian Institute of Engineering Science and Technology Shibpur, India-711103
[3]Institute for Integrated Circuits, Johannes Kepler University Linz, A-4040 Linz, Austria
Email: chandan@uni-bremen.de, robert.wille@jku.at, drechsle@uni-bremen.de, rahaman_h@it.iiests.ac.in

Abstract— **In this work, we propose an optimization technique for reversible circuits. The involved steps in this optimization are divided in two stages: First, we define templates and, second, we execute an optimization algorithm, where templates are applied on a specific order over the input circuit and exhaustively search through the circuit for possible replacements. The proposed optimization approach has been tested over a wide spectrum of benchmarks and improvement over state-of-the-art design approaches is registered.**

Keywords — Quantum Cost (QC), Gate Count (GC), Templates

I. INTRODUCTION

In the last couple of years, semiconductor industry has witnessed some tremendous advancement in fabrication technology. Besides, the packing density of transistors in IC also has increased exponentially whereas miniaturization of ICs is in continuous trend. This miniaturization of ICs has resulted several issues like high thermal noise, poor yield and also affected in the reliability of the ICs. So, the necessity of designing energy efficient circuits is observed and one of the possible future solutions to this problem can be inform of reversible circuit.

Unlike non-reversible(classical) computation which only allows the computation from input to output, reversible computation perform bidirectional traversal – input to output and output to input. As per the observation made by Landauer [2] and Bennett[3], such special ability to perform bidirectional computation enables information-lossless property in circuits which may leads to a power-efficient design. This relation between logical reversibility and power consumption is recently verified in [4]. Since couple of years, this field has observed extensive researches and new findings are being reported on daily basis. The most promising area where reversible computation has seen its application is quantum computation [5] where all the operations inherently reversible. Apart from quantum computation, the applications of reversible computation is also seen in on-chip interconnects [6] and adiabatic computation [7].

At same time, developing efficient algorithms for reversible circuit design bear high importance. But as the designs of reversible circuits are very different from conventional logic circuits, traditional circuit synthesis schemes cannot be applied for synthesis of such special type of circuit. Although different synthesis techniques are already existing, developing efficient synthesis algorithms have high significance. Based on the level of scalability and type of algorithm used, synthesis schemes can broadly be separated in the following two classes.

Optimal solution based: This class contains synthesis algorithms [8-10] that have low scalability level but generate optimal solutions for small variable functions (up-to 6 input variable) and when the algorithms of this class are scaled then high time and space complexities [11] are observed.

Sub-optimal solution based: This class of algorithms have quite opposite properties than optimal methods. Solutions based on sub-optimal techniques have higher scalability level but cannot ensure optimality in solutions. Approaches like Binary Decision Diagrams (BDD) [12] based technique or Exclusive-Sum-of-Products (ESOP) [13], [24] based solution are the examples of this class.

Apart from these two classes, there also exist various techniques like Reed-Muller expansion based synthesis schemes [14], [25], Group theory based synthesis schemes [15], heuristic algorithm based synthesis process [16] etc. But such synthesis techniques, in most of the time generate sub-optimal solutions and have a limited level of scalability such as upto 30 variable functions.

So, optimizing such already synthesized circuits generated from sub-optimal synthesis algorithms bear much significance and here the necessity of post-synthesis optimization is observed. Not only the cost optimization [17-18] is a priority, but also line optimization [19], gate count optimization, etc. found much interest.

In our work, we too have proposed a similar optimization algorithm which focuses on cost reduction in synthesized circuit. We have proposed several templates which replace the high cost sub-structures with equivalent low-cost solutions in circuits. We also have undertaken a wide scale testing for our optimization algorithm and it is found very effective.

Now, here we summarize the rest of the content. Section II introduces the preliminaries on reversible circuits. Details on the designed templates and the resulting optimization technique are stated in Section III. Experimental results and a comparative study over exist methodologies are given in Section IV. Finally, the work is concluded in Section V.

978-1-7281-9369-4/20/$31.00 © 2020 IEEE

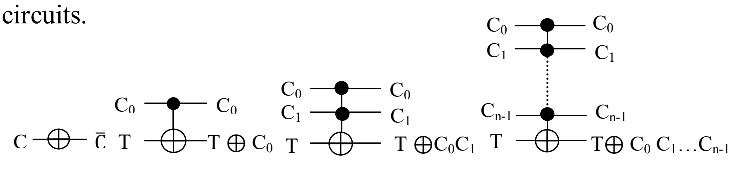
Authorized licensed use limited to: STAATS U UNIBIBL BREMEN. Downloaded on October 05,2020 at 08:11:15 UTC from IEEE Xplore. Restrictions apply.

II. BACKGROUND

To familiarize the readers with reversible circuit and related terminologies, here we are introducing the basics of reversible family.

Definition 1: *A circuit is called reversible if it has the equal number of input-output lines and performs one to one correspondence between input and output vector pairs.*

Reversible circuits are designed with reversible gates and NOT, CNOT and Toffoli [20] (see in Fig. 1) are the most common reversible gates used for designing reversible circuits.



(a): NOT **(b):** CNOT **(c):** 2-control Toffoli **(d):** *n* - control Toffoli gate

Fig.1: The NCT library

In order to evalute the performance of such reversible circuits, two cost parameters namely Gate Count (GC) and Quantum Cost (QC) [21] are considered as important cost metric. Apart from these two parameters, T-count and T-depth too are also used to judge circuit performances. Here, we define the commonly used cost metrics.

Definition 2: *A quantum circuit is a collection of quantum gates over a set of circuit lines, where each of the gates performs its unique quantum operation. The number of quantum operations required to realize a functionality using such elementary quantum gates is known as quantum cost.*

For a circuit, it is the cumulative value of all individual gate's costs present in the design and can be represented as $QC = \sum_{i=1}^{k}$ quantum cost (g_i), where g_k represents the k^{th} reversible gate in the circuit.

Definition 3: *The Gate Count (GC) metric of a circuit is the count value of number of gates present in the design.*

III. PROPOSED TECHNIQUE

Here, we are introducing a template based post-synthesis optimization technique for cost efficient realization of reversible circuits. Not only, this approach is very effective over the post-synthesis optimization but also can be employed over logic minimization as well. The entire optimization process involves two stages. In the first stage, the required templates are formulated and then in second stage, an optimization algorithm runs which takes a circuit as input and exhaustively searches through it for possible replacements of sub-circuits with the templates and removes the extraneous gates from the design.

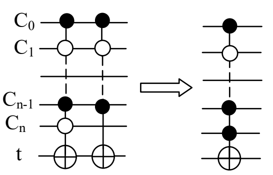
Here we are stating all the stages of our algorithm in detail.

A. *Formation of templates:*

In this phase of optimization, we design the templates and form a template library. Five templates are defined here and depending on the nature of these templates we have categorized them into three classes - *circuit optimization*, *circuit expansion* and *control node sharing.* Now, we show the designs and the formation patterns of all the templates.

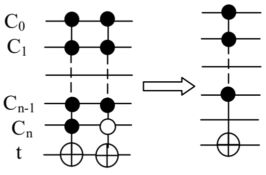
Template1: *If there exist two consecutive Toffoli gates TOF $(C_1;T)$ and TOF $(C_2;T)$, where $C_2 \subset C_1$ and $C_1\setminus C_2$ has a single element c_i , then TOF $(C_1;T)$ TOF $(C_2;T)$ = TOF $(C_2 \cup \overline{c_i};T)$.*

The template is presented in form of an example in Fig.2. Though it is not necessary that all the controls in gates to be consecutive, but for an easy of understanding, we have assumed that they are consecutive in examples.



(a): A pair of consecutive gate **(b):** Transformed design

Fig. 2: Template1 (T-1)



(a): Example circuit **(b):** Transformed design

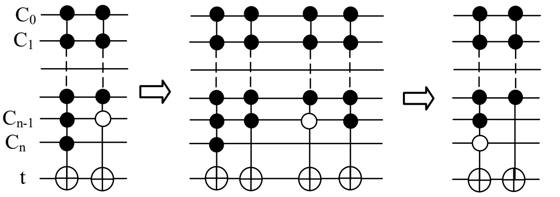
Fig. 3: Template2 (T-2)

Template2: *If there exist two consecutive Toffoli gates TOF $(C_1;T)$ and TOF $(C_2;T)$, (where $C_1= C_2$ and $C_2\setminus C1$ is Φ) in such a way that the polarity of a control node at $D_{1_k} = D_{2_{\bar{k}}}$, then TOF $(C_1;T)$ TOF $(C_2;T)$ = TOF $(C_1\setminus \overline{c_k}; T)$.*

The structure for template2 is depicted in Fig. 3.

Template3: *Let two consecutive Toffoli gates $TOF_1(C;T)$ and $TOF_2(C;T)$ operate over n control lines and have a single target line T. If these two gates contain m controls and (m-1) controls respectively where [m≤n] all the (m-2) controls in $TOF_1(C;T)$ are same as the (m-2) controls of $TOF_2(C;T)$ and the polarity of $(m\text{-}1)^{th}$ control in $TOF_1(C;T)$ is inverse to the polarity of $(m\text{-}1)^{th}$ control in $TOF_2(C;T)$, then addition of a pair of gates constructed from $TOF_1(C;T)$ or $TOF_2(C;T)$ with (m-2) controls into the existing netlist produces a reduced cost netlist.*

The design of template3 is shown in Fig.4, where initially the adjacent gates are expanded and then they are reduced to a low cost-based solution.



(a): Initial design **(b):** Expanded circuit **(c):** Final design

Fig. 4: Template3 (T-3)

Template4: *Let two consecutive m-control Toffoli gates $TOF_1(C;T)$ and $TOF_2(C;T)$ are operating over n control lines and have a single target line at T, where m≤n. Now, if the hamming distance between the controls of the gates be k, then the Toffoli pair can be substituted with k number of p-control Toffoli gate where p<m.*

978-1-7281-9369-4/20/$31.00 © 2020 IEEE

Authorized licensed use limited to: STAATS U UNIBIBL BREMEN. Downloaded on October 05,2020 at 08:11:15 UTC from IEEE Xplore. Restrictions apply.

For easy understanding of the above stated rule, in Fig. 5 and 6, two circuits with different hamming distances and their respective transformed designs are shown.

Template 4 (T-4):

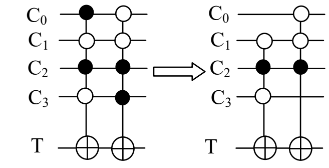


Fig. 5(a): Two gates having hamming distance of 2 **Fig. 5(b):** Transformed circuit

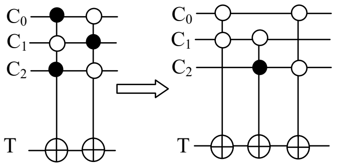
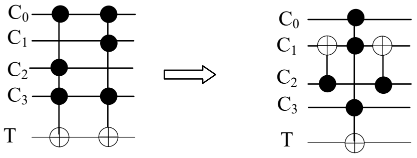


Fig. 6(a): Two gates maintaining hamming distance of 3 **Fig. 6(b):** Transformed circuit

Template5: *Let two consecutive Toffoli gates $TOF_1(C;T)$ and $TOF_2(C;T)$ are operating over n control lines and have a single target line T. If the control nodes in both the gates are located in such a position that they have at least one common control, then the gate pair can be replaced with a set of new gates by sharing the functional expression between their control lines.*

If the above stated condition is satisfied, then both the gates can be substituted with the structure as shown in Fig. 7.



(a): Cascade of two gates **(b):** Paired structure

Fig. 7: Template5 (T-1)

B. *Formulation of the optimization Algorithm:*

The design of templates is now complete, and now a template library (named as T_{lb}) is formed by taking all the five templates in a group. These formed templates find their structure matches in sub-circuits and replace them with lower cost designs. But all the templates cannot be executed arbitrarily over an input circuit, rather it need some sequential execution to achieve higher level of optimization. Steps involved in this template matching technique are summarized in Algorithm 1.

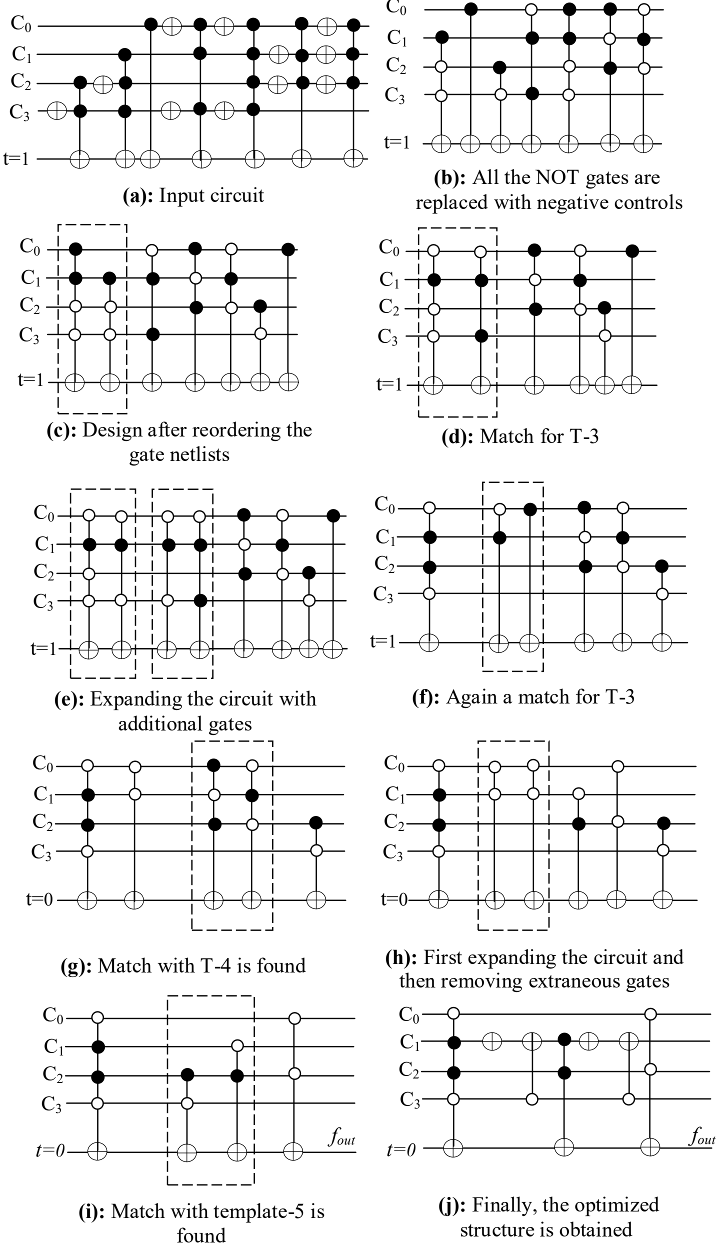
Algorithm1: Template Matching Process

```
Input: Un-optimized circuit: $C_{in}$ , $T_{lb}$
Output: Redundancy free design: $C_{optimized}$

begin
    $C_{upgrade}$ =Replace_NOT($C_{in}$);
    $C_{ordered}$ =Shuffle($C_{upgrade}$);
    Flag=1;
    while(Flag==1)
        do
            Match_Template($T_i$, $C_{ordered}$).// where i∈ {1,2,3}.
            $C_{updated}$ = Replace_with_Template($T_i$, $C_{ordered}$);
        while(no further matches are found in $C_{updated}$)
        $C_{improved}$ =Invoke($T_5$ , $C_{updated}$);
        S=Circuit_Expansion($T_4$ , $C_{improved}$)
        if (S==False)
            Flag=0;
        end if
    end while
    $C_{final}$ = $C_{improved}$
end
```

Let's take an example to understand the optimization algorithm.

Example 5.1: Consider the circuit of Fig. 8(a) representing the function f_{out} = $1\oplus c_2\bar{c}_3\oplus c_1\bar{c}_2\bar{c}_3\oplus\bar{c}_0c_1\bar{c}_2\oplus c_0\bar{c}_1c_2$ $c_0\oplus\bar{c}_0c_1c_3\oplus$ $c_0c_1\bar{c}_2\bar{c}_3$. This input circuit is an un-optimized design which incurs a quantum cost of 87 from 18 gates. In Fig. 8 we have shown all the involved optimization steps and also show how gradually the cost of the circuit is finally reduced to 46.



(a): Input circuit

(b): All the NOT gates are replaced with negative controls

(c): Design after reordering the gate netlists

(d): Match for T-3

(e): Expanding the circuit with additional gates

(f): Again a match for T-3

(g): Match with T-4 is found

(h): First expanding the circuit and then removing extraneous gates

(i): Match with template-5 is found

(j): Finally, the optimized structure is obtained

Fig.8: Showing optimization steps over an example circuit

IV. Experimental Evaluation

The stated optimization technique has been tested over a wide range of benchmarks and computed results are summarized in table1. The obtained data have been compared with related optimization techniques and post-synthesis works [18], [22].

978-1-7281-9369-4/20/$31.00 © 2020 IEEE

Authorized licensed use limited to: STAATS U UNIBIBL BREMEN. Downloaded on October 05,2020 at 08:11:15 UTC from IEEE Xplore. Restrictions apply.

In the comparison, we have evaluated two cost parameters – Quantum Cost (QC) and Gate Count (GC). In our result table, we have observed steady cost improvement and gate count reduction from our approach.

V. Conclusion

This work has presented a template-based optimization scheme for post-synthesis minimization of reversible circuit. The proposed optimization technique has successfully tested over various synthesized netlists and improvements in cost metrics are seen. Not only the developed scheme is applicable for post-synthesis optimizations but it can be helpful in logic minimization as well.

Further reductions in cost metrics by appending ancillary line and incorporating structural changes in the design is under investigation which will be addressed in future.

References

[1] G. E. Moore, "Cramming more components onto integrated circuits," *In Electronics*, *38*, January 1965.

[2] R. Landauer, "Irreversibility and heat generation in the computing process," *In IBM Journal of Research and Development*, 5:183–191, July 1961.

[3] C. H. Bennett, "Logical reversibility of computation," *In IBM Journal of Logical Research and Development*, 6:525–532, November 1973

[4] A. Berut, A. Arakelyan, A. Petrosyan, S. Ciliberto, R. Dillenschneider, and E. Lutz, "Experimental verification of Landauer's principle linking information and thermodynamics," *Nature*, 483:187–189, 2012.

[5] L. Biswal, R. Das, C. Bandyopadhyay, H. Rahaman, A. Chattopadhyay "A template-based technique for efficient Clifford +T -based quantum circuit implementation," Microelectronics Journal**.** 2018 Nov 1;81:58-68.

[6] R. Wille, O. Keszocze, S. Hillmich, M. Walter, and A. G. Ortiz, "Synthesis of approximate coders for on-chip interconnects using reversible logic," *In Design, Automation and Test in Europe*, 2016.

[7] A. Zulehner, M. Frank, and R. Wille, "Design Automation for Adiabatic Circuits," In Asia and South Pacific Design Automation Conference (ASP-DAC), 2019

[8] D. Maslov, G. W. Dueck, and D. M. Miller, "Techniques for the synthesis of reversible Toffoli networks", *ACM Trans. On Design Automation of Electronic Systems*, 12(4), 2007.

[9] O. Golubitsky, S. M. Falconer, and Dmitri Maslov," Synthesis of the optimal 4-bit reversible circuits*," In Proceedings of the 47th Design Automation Conference*, pages 653-656. ACM, 2010.

[10] D. Große, R. Wille, G. W. Dueck, and Rolf Drechsler, "Exact multiple-control Toffoli network synthesis with SAT techniques," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp.703-715, 2009.

[11] A. Chattopadhyay, C. Chandak, and K. Chakraborty, "Complexity analysis of reversible logic synthesis," *arXiv preprint arXiv:1402.0491,* 2014.

[12] R. Wille and R. Drechsler,"BDD-based synthesis of reversible logic for large functions", *In Design Automation Conf 2009*, pp. 270–275.

[13] K. Fazel, M.Thornton, and J. E. Rice,"ESOP based Toffoli gate cascade generation", *In PACRIM,* pp. 206–209, 2007.

[14] P. Gupta, A. Agrawal, and N.K. Jha, " An algorithm for synthesis of reversible logic circuits", *IEEE Trans. on CAD*, 25(11):2317–2330, 2006.

[15] G. Yang, X. Song, W. N. N. Hung, F. Xie, and M. A. Perkowski, "Group theory based synthesis of binary reversible circuits," *In Proceedings of TAMC*, pages 365–374, 2006.

[16] M. Li, Y. Zheng, M. S Hsiao, and Chao Huang, "Reversible logic synthesis through ant colony optimization," *In Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2010, pages 307-310. IEEE, 2010.

[17] M. Soeken, R. Wille, G. W. Dueck, and R. Drechsler, "Window optimization of reversible and quantum circuits," *In Symposium on Design and Diagnostics of Electronic Circuits and Systems*, pages 341–345, 2010.

[18] K.Datta, I. Sengupta and H. Rahaman, "A Post-Synthesis Optimization Technique for Reversible Circuits Exploiting Negative Control Lines," *IEEE Transactions on Computers*, 2016

[19] A. Zulehner and R. Wille, "Exploiting Coding Techniques for Logic Synthesis of Reversible Circuits," In Asia and South Pacific Design Automation Conference (ASP-DAC), 670-675, 2018.

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

[21] D. Maslov, " Reversible logic synthesis benchmark page,". In *http://www.cs.uvic.ca/dmaslov/*, 2002.

[22] K. Datta, G. Rathi, R. Wille, I. Sengupta, H. Rahaman, R. Drechsler, " Exploiting Negative Control Lines in the Optimization of Reversible Circuits," *Reversible Computing* 2013: 209-220

[23] R. Wille, D. Grosse, L. Teuver, G. W. Dueck, and R. Drechslere, "Revlib: an online resources for reversible functions and reversible circuits," *In International Symposium on Multi-Value Logic,* 2008.

[24] C. Bandyopadhyay, D. Roy, D. K. Kole, K. Datta, H. Rahaman, "ESOP-based Synthesis of Reversible Circuit Using Improved Cube list", *IEEE 4th International Symposium on Electronic System Design (ISED), 201*3, pp.26-30**,** Singapore

[25] C. Bandyopadhyay, and H. Rahaman "Synthesis of ESOP-based Reversible Logic using Positive Polarity Reed-Muller Form", IEEE (ETCC)-2014, pp. 363-376, Calcutta,

Table1: Comparison with related optimization techniques

Benchmarks Specifications		**Optimization using [22]**		**Optimization using [18]**		**Proposed optimization**	
Names	Inputs/ Outputs	Quantum Cost	Gate count	Quantum Cost	Gate count	Quantum Cost	Gate count
decod_217	5/16	1745	79	613	21	432	18
c7552_205	5/16	1745	79	623	23	417	16
sqr6_259	6/12	1034	66	876	53	539	72
sqn_258	7/3	2041	50	1887	51	2113	58
inc_237	7/9	2104	72	1745	63	1553	68
rd73_312	7/3	214	65	200	53	203	63
sqn_258	7/3	2041	50	1887	51	1685	69
5xp1_194	7/10	1327	65	1155	59	987	61
dc2_222	8/7	1789	55	1688	53	1542	61
life_238	9/1	5740	57	5744	57	5893	61
max46_240	9/1	4498	51	4538	52	3995	47
9symml_91	9/1	12747	58	13026	62	3276	126
clip_206	9/5	6535	111	6119	109	5419	97
apex4	9/19	237748	5039	158095	3469	79452	1408
alu2_199	10/6	4776	87	4611	87	4219	97
apla_203	10/12	3438	74	3024	64	2987	69
add6_196	12/7	6005	179	5534	167	4745	158
tial_265	14/8	47145	516	47556	522	41110	487
f51m_233	14/8	33333	358	32882	355	29004	325
misex3_242	14/14	115637	1199	99119	1043	83123	978
misex3c_243	14/14	111258	1188	96064	1049	79880	746
in0_235	15/11	18999	245	16985	218	14413	198
cordic_218	23/2	348566	1567	348532	1567	253019	253019

978-1-7281-9369-4/20/$31.00 © 2020 IEEE

Authorized licensed use limited to: STAATS U UNIBIBL BREMEN. Downloaded on October 05,2020 at 08:11:15 UTC from IEEE Xplore. Restrictions apply.