An Efficient Post-Synthesis of Reversible Circuits[⋆]

Raphael Bernardino[1][0000−0001−9730−901X], Franklin de Lima
Marquezino[2][0000−0001−9712−1930], Edinelço Dalcumune[3][0000−0002−3999−6664],
and Luis Antonio Kowada[1][0000−0002−7975−0060]

[1] Universidade Federal Fluminese, Niterói, Brazil
`raphaell@id.uff.br, luis@ic.uff.br`
[2] Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil
`franklin@cos.ufrj.br`
[3] Universidade Federal dos Vales do Jequitinhonha e Mucuri, Teófilo Otoni, Brazil
`edcomune@ufvjm.edu.br`

1 Introduction

Reversible computing is one of the most promising solutions to the huge energy consumption and heat dissipation of current computers. The theory of reversible circuits is also fundamental for quantum computing, since the postulates of quantum mechanics establish that the evolution of closed quantum systems is unitary and thus reversible. Recent studies [2] have shown that spent energy per bit erased can be reduced to 17 meV, as Landauer [4] has predicted, instead of 5 keV of the current CMOS technology. We are considerably far from the ideal scenario, and still facing many barriers imposed by physics.

The process of transforming a given reversible function into a reversible circuit is known as circuit synthesis. The generated circuit may still be computationally expensive, so an optimization process known as post-synthesis is important to achieve efficient circuits. Reversible computing is free from the aforesaid problems since there is no information loss. Therefore, the synthesis of reversible circuits represents a significant advance in classical computing. Bennett [1] has proved that by using only a few universal reversible gates one can implement any Boolean function and reduce the energy consumption on calculations.

In this work, we present an efficient post-synthesis algorithm to reduce the gate count of reversible circuits. Our method identifies each maximal sub-circuit with up to three different lines, uses different heuristic rules to cancel or merge similar gates, then finds the permutations associated with each of them, and finally replaces them with minimum sub-circuits. We tested our method with well-known benchmark functions and successfully optimized most circuits. The average number of gates has been reduced by more than 10% with our method.

This work is organized as follows. The proposed algorithm, which is the main contribution of this paper, is introduced in Sec. 2. Our results, conclusions and final remarks are described in Sec. 3.

---

2   Our proposed algorithm

The complete algorithm comprises two phases: preprocessing and optimization phase. The preprocessing phase needs to be run only once to generate all minimum circuits. On the other hand, the optimization phase needs to be run for every new input circuit.

In the preprocessing phase we generate the minimum circuit for each set of three bits using mixed Toffoli gates. Each reversible circuit is associated to a permutation P_n. The minimum circuits are stored into a look-up table and used later to optimize general circuits.

In the optimization phase, the circuit is reorganized trying to choose the best neighbors, *i.e.*, the Hamming distance is less than or equal to three—as we only have minimum circuits for $n \leq 3$ bits. Then, use a look-up strategy to find the best circuit replacement based on the original permutation. For a more efficient approach, we split the circuit into sub-circuits. We expect to reduce its cost by analyzing small subsets of the problem.

3   Results and Conclusion

We evaluate our algorithm using well-known comparisons and benchmark functions. The results are presented in Table 1. The first and the second columns show, respectively, the name of each benchmark function and its number of variables considered in the literature. The columns marked with "**our**" on the right of each author show the best results obtained by applying the proposed algorithm on the author final circuit. All specifications for benchmark functions and their names were taken from the *RevLib site*[4] and from the *Reversible Benchmarks*[5].

The proposed algorithm finds a better circuit in 82% of the cases. In terms of gate cost, we further improve the literature in 8 cases. On average, the algorithm reduced gates by a factor of 9.26, which represents the best result so far. On average, the proposed algorithm improves cases without mixed controls by 17.24%. As expected, our results prove the efficiency of such cases. On average, it is optimized by 6.17%. On large circuits, as expected, we get better results as we SWAP and move gates to optimize the overall circuit.

Current computers have a huge carbon footprint and a very negative impact on the environment. Reversible computing is one of the most promising solutions to the problem of high energy consumption and heat dissipation of classical computation. Moreover, the synthesis of reversible circuits is fundamental for the development of quantum computing.

Our main contribution was the introduction of an efficient post-synthesis algorithm to reduce the gate count of reversible circuits. Our hybrid method combines templates with heuristic rules for circuit optimization. We tested our method with a series of well-known comparisons and benchmark functions. Our proposed algorithm finds significant reductions in the gate counts for most input

---

[4] Available at http://www.revlib.org.
[5] Available at https://reversiblebenchmarks.github.io/.

benchmark	n	DKRFM [3]	(our)	Z-16 [7]	(our)	MMD-05 [5]	(our)	MMD-07 [6]	(our)
3_17	3	6	5	4	4	6	4	6	4
4_49	4	14	12	-	-	16	13	12	10
4b15g_2	4	17	16	12	12	-	-	-	-
4b15g_4	4	19	18	12	12	-	-	-	-
4b15g_5	4	18	17	14	14	-	-	-	-
ham3	3	6	6	-	-	5	5	5	5
hwb4	4	18	17	-	-	17	15	11	11
hwb5	5	43	43	-	-	55	54	24	21
hwb6	6	103	100	-	-	126	122	42	40
hwb7	7	282	273	603	558	289	281	236	229
mod5adder	6	17	15	-	-	-	-	15	9
mod5mils	5	3	3	-	-	5	3	-	-
nth_prime4_inc	4	13	13	-	-	-	-	12*	10
nth_prime5_inc	5	38	37	-	-	-	-	25*	23
nth_prime6_inc	6	79	76	-	-	-	-	55*	50
nth_prime7_inc	7	231	224	427	396	-	-	-	-

* Result available at https://reversiblebenchmarks.github.io/

Table 1. Our results of post-synthesis are listed in columns marked with "**our**". The main current results, using gate count metric, are in the other columns.

circuits. On average, our algorithm reduces the gate count by a factor of 9.26, which represents an advance in the state of the art.

References

1. C. H. Bennett. Logical reversibility of computation. *IBM Journal of Research and Development*, 17(6):525–532, Nov 1973.
2. Antoine Bérut, Artak Arakelyan, Artyom Petrosyan, Sergio Ciliberto, Raoul Dillenschneider, and Eric Lutz. Experimental verification of landauer's principle linking information and thermodynamics. *Nature*, 483(7388):187–189, 2012.
3. E. Dalcumune, L. A. B. Kowada, A. C. Ribeiro, C. M. H. Figueiredo, and F. L. Marquezino. A reversible circuit synthesis algorithm with progressive increase of controls in generalized toffoli gates. *JUCS - Journal of Universal Computer Science*, 27(6):544–563, 2021.
4. R. Landauer. Irreversibility and heat generation in the computing process. *IBM Journal of Research and Development*, 5(3):183–191, July 1961.
5. D. Maslov, G. W. Dueck, and D. M. Miller. Toffoli network synthesis with templates. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 24(6):807–817, 2005.
6. D. Maslov, G. W. Dueck, and D. M. Miller. Techniques for the synthesis of reversible Toffoli networks. *ACM Trans. Des. Autom. Electron. Syst.*, 12(4), September 2007.
7. D. V. Zakablukov. Application of permutation group theory in reversible logic synthesis. In Simon Devitt and Ivan Lanese, editors, *Reversible Computation*, pages 223–238, Cham, 2016. Springer International Publishing.