An Efficient Reduction of Common Control Lines for Reversible Circuit Optimization

Arighna Deb* Robert Wille*† Rolf Drechsler*† Debesh K. Das‡
*Institute of Computer Science, University of Bremen, Bremen, Germany
†Cyber Physical Systems, DFKI GmbH, Bremen, Germany
‡Computer Sci. & Engg., Jadavpur University, Kolkata, India
{deb, rwille, drechsle}@informatik.uni-bremen.de
debeshd@hotmail.com

***Abstract*—New prospects in several emerging technologies such as quantum computation and certain aspects of low-power design motivated an intensive consideration of the design of reversible circuits. Since most of the existing synthesis approaches usually generate circuits of high costs, post-synthesis optimization is frequently applied. Here, the reduction of control line connections is a major focus as they are a main reason for high quantum costs in the respective reversible circuits. Previous approaches aimed for exploiting so-called *common control lines* for this purpose. However, while these solutions indeed lead to substantial improvements in the costs, they inherit some drawbacks and restrictions.**

In this work, we propose an alternative approach for the reduction of common control lines in reversible circuits, which (1) is based in the concepts of previously proposed solutions, but (2) combines them in a new fashion. This enables us to achieve the same or even better improvements, while – at the same time – overcome their drawbacks. Experimental evaluations confirm these benefits, i.e. significant improvements compared to the previous methods can often be achieved without the need to deal with their drawbacks.

Keywords*—*Reversible circuits, synthesis, optimization, quantum costs, common control lines

I. Introduction

In the recent years, new prospects in several emerging technologies came up which motivated a detailed consideration of *reversible logic*. Here functionality is specified and, eventually, realized in a bijective fashion, i.e. a unique input/output mapping is enforced. In particular, applications in the domain of quantum computation and certain aspects of low-power design may profit from the corresponding functional descriptions. More precisely, quantum computation [1] offers a new way of computing which allows for speed-up in solving many important problems such as database search [2] or factorization [3]. This is because, instead of bits, qubits are used to represent information during a quantum computation. Besides the conventional values (i.e. logical 1 and logical 0), qubits can assume a superposition of state 0 and state 1. As a result, a set of qubits can represent multiple states at the same time enabling enormous computational speed-ups. Reversible circuits play a significant role in this area, since each quantum gate operation is inherently reversible.

Low-power design may, in the future, profit from reversible logic because of observations by Landauer [4]. He proved that, during a computation, each information loss causes a certain amount of power dissipation. Now, most of the conventional, i.e. irreversible, circuits available thus far frequently lose information (e.g. when performing the basic AND operation, two input bits are mapped to a output bit, i.e. one bit of information is lost). These losses lead to a frequent power dissipation which, with increasing miniaturization, might become crucial in the near future. Entirely relying on reversible computation never leads to an information loss and, hence, may avoid the corresponding power dissipation. Moreover, Bennett [5] showed that any circuit which aims for a (theoretical) power dissipation of zero, indeed has to be reversible. Recently, this has been experimentally confirmed in [6]. Reversible logic has also found applications in the design of low power encoders [7].

Motivated by the reasons cited above, researchers started working towards the development of (automatic) design methods for such circuits. In particular, the synthesis of reversible circuits has become an intensely studied topic in the last decade. Many synthesis approaches exploiting e.g. permutations [8], [9], truth-tables [10]–[12], decision diagrams [13], or exclusive-sum-of-products (ESOP) [14]–[16] have been proposed. In most of the cases (except for very small circuits), these synthesis approaches do not generate optimal results with respect to the circuit cost. Therefore, post-synthesis optimization is frequently applied to further reduce the costs of a circuit.

The reduction of control line connections is thereby a major focus. This is caused by the fact that gates with many control lines are a main reason for high quantum costs in the respective reversible circuits – reducing their amount allows for substantial reductions. Hence, researchers particularly tried to make use of so-called *common control lines*, i.e. equal or similar control line connections in neighboring gates. They offer the possibility of sharing which allows for a significant reduction of control lines. Accordingly, several approaches exploiting common control lines have recently been proposed. For example, in [17] common control lines are shared by adding additional signal lines to the circuit. The optimization approach from [18] exploits so-called multiple target lines, i.e. basically merges reversible gates with identical control lines to a single one. The respective ideas are reviewed in somewhat more detail later in Section III.

However, although these optimization approaches indeed lead to substantial improvements in the costs, they come with some drawbacks and restrictions: Either an addition of circuit lines or a restriction to a rather small set of common control lines connections to be optimized. Hence, they do not fully unleash the potential of exploiting common control connections in reversible circuits.

In this work, we propose an alternative scheme for the optimization of reversible circuits. We propose a post-synthesis optimization technique which exploits the characteristics of reversible circuits – in particular common control connections – based on the concepts presented in [17], [18]. But we combine these techniques in a new fashion which allows for achieving the same or even better improvements without the need to deal with their drawbacks.

Experimental evaluations confirm that, applying the proposed approach, significant improvements compared to the methods in [17] and [18] can often be observed. At the same time, we overcome their drawbacks.

The remainder of this paper is structured as follows. Section II provides the background on reversible logic. A brief review of the approaches proposed in [17] and [18] is given in Section III which eventually leads to a sketch of the general idea of our solution. Afterwards, the detailed algorithm is described in Section IV. Section V summarizes the results obtained by our experimental evaluation and, finally, the paper is concluded in Section VI.

II. Background

To keep the paper self-contained, this section provides some background on reversible circuits. For a more detailed introduction, readers are referred to [1].

Definition 1. *A multiple-output Boolean function* $\mathrm{B^n} \rightarrow \mathrm{B^n}$ *is said to be* reversible *if and only if it maps each input pattern to a unique output pattern.*

A reversible function can be realized by a reversible circuit, $G = g_1, g_2, \ldots g_k$, where each g_i is a reversible gate. Many reversible gates have been proposed in the past [1]. Among them, the *Toffoli gate* [19] is widely used and also considered in this paper.

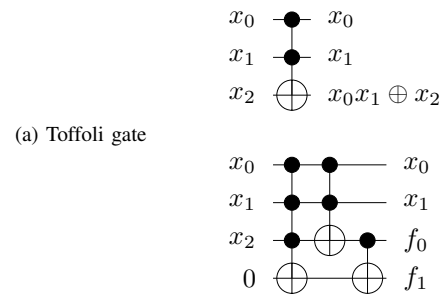
Definition 2. *Given a set of Boolean variables* $X = \{x_0, x_1, \ldots, x_{n-1}\}$*, a multiple-control Toffoli gate* $g(C;t)$ *is a tuple of a possibly empty set* $C \subset X$ *of control lines and a single target line* $t \in X \setminus C$*. The Toffoli gate inverts the value on the target line* t *if and only if all the control lines are set to* 1 *or if* $C = \emptyset$*.*

In case of $|C| = 0$ and $|C| = 1$, the Toffoli gate $g(C;t)$ is called *NOT gate* and *CNOT gate*, respectively.

Example 1. *Fig. 1a shows a Toffoli gate with two control connections (denoted by* $\bullet$*) and a single target connection (denoted by* $\oplus$*). A circuit composed of several Toffoli gates is shown in Fig. 1b.*

The costs of a reversible circuit are commonly measured using the quantum cost metric as originally introduced in [20] and refined in later work. The quantum cost of a circuit is the sum of the quantum costs of the individual reversible gates.

For reversible circuits, the quantum costs depend on the number of control connections i.e. the control size of the each gate. For example, a Toffoli gate with no or one control connection has quantum costs of one, whereas, a Toffoli gate with two control connections has costs of five. The detailed quantum costs for remaining reversible gates can be found in [20], [21].



(a) Toffoli gate

(b) Toffoli circuit

Fig. 1: Toffoli gate and Toffoli circuit

In many cases, the objective of synthesizing a reversible circuit involves the realization of an irreversible Boolean function. This requires the irreversible function to be embedded into a reversible one [22] using additional constant inputs and garbage outputs defined as follows:

Definition 3. *A constant input to a reversible circuit is one that has a fix input value (either* 0 *or* 1*).*

Definition 4. *A garbage output from a reversible circuit is one which is don't-care for all possible input conditions.*

III. Motivation and General Idea

The optimization of reversible circuits has significantly been considered in the recent past. A major focus has been put on the reduction of control line connections as they constitute as main cause for high quantum costs in the respective reversible circuits. A frequently applied approach was thereby the reduction of *common control lines*, i.e. equal or similar control line connections in neighboring gates. They frequently occur in reversible circuits and, hence, lead to redundant computations of the respective factors[1]. Optimization approaches addressing this objective in the past allowed for substantial reductions in common control line connections and, hence, significantly reduced costs. At the same time, they often inherit drawbacks and restrictions which needed to be traded-off and considered, respectively.

In this section, two prominent optimization approaches presented recently (namely in [17] and [18]) are briefly reviewed and their drawbacks and restrictions are discussed. Afterwards, the general idea of a new solution is proposed which combines the core ideas of these previously proposed approaches in a clever fashion. By this, the discussed drawbacks and restrictions are addressed and, in fact, reversible circuits with even smaller costs than those obtained before are generated.

[1]The terminology "factor" has frequently been used here since the consideration of an equal or similar control set C for gates $g_i(C_i, t_i), g_{i+1}(C_{i+1}, t_{i+1}), \ldots$ with $C_i \cap C_{i+1} \cap \cdots = C$ is essentially considering a factor of the respective AND function for these control lines.

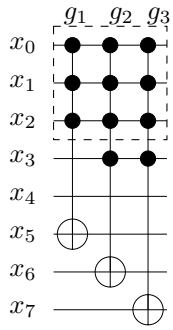


Fig. 2: Given circuit

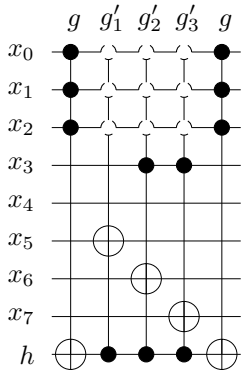


Fig. 3: Exploiting additional circuit lines

A. Exploiting Additional Circuit Lines

In [17], the authors proposed one alternative to reduce common control lines in reversible circuits. The main idea was to extend the respective reversible circuits by additional circuit lines which are used to buffer the respective value of a factor (obtained by the common control lines) to be re-used at a later time. More precisely, they introduced one or more so-called *helper lines* whose (1) inputs are set to a constant value 0 and (2) whose output is used as a garbage output. This provided a storing entity and allowed for significantly reducing redundant control connections. The idea is clarified in the following example.

Example 2. *Consider the reversible circuit depicted in Fig. 2. The gates in this cascade have a common control factor $F = \{x_0, x_1, x_2\}$ (highlighted by a dashed rectangle). Hence, the cost of this circuit can be reduced as shown in Fig. 3 by adding an additional line h (at the bottom of the circuit) as well as the gates $g(F, h)$ before and after the cascade. This leads to additional quantum cost of $2 \times 14 = 28$. However, the factored gates can reuse the result of F leading to a reduction of two control lines per gate. The removed control lines are shown as dashed white circles in Fig. 3. In total this reduces the quantum cost from 54 to 39.*

B. Exploiting Multiple Target Lines

A complementary approach to reduce common control lines has been proposed in [18]. Here, the observation is exploited that many reversible circuits are composed of exactly the same control line connections and differ only in their target lines[2]. That is, cascades such as $g_i(C_i, t_i), g_{i+1}(C_{i+1}, t_{i+1}), \ldots$ with $C_i = C_j$, $t_i \notin C_j$, and $t_{i+1} \notin C_i$ frequently occur. Instead of considering each of those gates (and its costs) solely, an alternative structure has been proposed, in which only one of those gates (e.g. $g_i(C_i, t_i)$) is applied first. Afterwards, the next gate (with the different target line) is realized by placing two further gates of the form $g(\{t_i\}, t_{i+1})$ before and after g_i. By this, the switching activity of target line t_i (triggered by the common control lines C_i) is reused: If the value of the target line t_i switches, then also the target line t_{i+1} switches. Otherwise, all values remain unchanged. In a similar fashion, this is conducted for all further gates. This exactly realizes the functionality of $g_i(C_i, t_i), g_j(C_{i+1}, t_{i+1}), \ldots$. Again, the following example clarifies the idea.

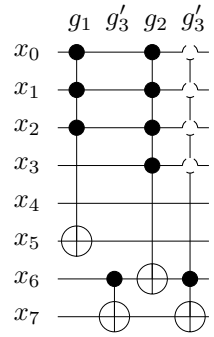


Fig. 4: Exploiting multiple target lines

Example 3. *Reconsider the reversible circuit depicted in Fig. 2. The gates g_2 and g_3 in this cascade have exactly same control line connections $\{x_0, x_1, x_2, x_3\}$ with different target lines. Due to that, the switching activity of the target line of the gate g_2 is reused leading to the complete removal of all the control connections from gate g_3. The removed control connections are shown by dashed white circles in Fig. 4. Hence, the cost of this circuit can be reduced as shown in Fig. 4 from 54 to 36 by replacing gate g_3 with the CNOT gates g'_3 before and after gate g_2 in the cascade.*

C. Discussion and General Idea

The optimization approaches reviewed above complementary address the objective of reducing the amount of common control connections and, by this, the resulting quantum costs of many reversible circuits. In fact, the experimental evaluations summarized in [17] and [18] unveiled reductions in the quantum costs of up to 70% and 66%, respectively. However, the proposed solutions come with some drawbacks and restrictions and do not fully unleash the potential of exploiting common control connections. More precisely:

- The solution presented in [17] and reviewed in Section III-A requires additional circuit lines. This is a serious drawback since the number of circuit lines directly correspond to the number of qubits in respective quantum circuit applications. As qubits are usually

[2] Accordingly, such structures are considered as a single gate with *multiple target lines* in [18].

considered a very limited resource, the designer must trade-off whether possible reductions in the quantum costs pay off against the increase in the number of qubits[3].

- The solution presented in [18] and reviewed in Section III-B is restricted to a rather small set of common control lines connections to be optimized. In fact, this solution can only be applied if an exact match between common control lines can be detected. While this is particularly suited for circuits obtained by synthesis schemes based on so-called ESOP descriptions (as applied e.g. in [14]–[16]), its effect is significantly smaller on other circuits. In fact, the full potential of common control lines is not used and further possible reductions remain unexploited.

In this work, we propose an alternative scheme for the optimization of reversible circuits. We re-use the core ideas and of the previously proposed solutions, but combine them in a new fashion which allows for (1) avoiding the drawbacks and shortcomings discussed above and (2) leads to even more significant improvements.

The general idea is as follows: As in [17], we consider cascades of reversible gates with common (not necessarily equal) control connections; but avoid buffering the respective factors by using an additional line. Instead, we keep the first gate of the cascade unchanged and, as in [18], exploit the switching activity of its target line in order to realize the functionality of the remaining gates (by adding cheaper gates before and after it). But, in contrast to [18], we allow this also for subsets of common control lines. Then, further control lines of an original gate (not represented by the switching activity of the target line) are simply added to the newly introduced gates. More precisely, we consider cascades of the form $G_d = g_i(C_i; t_i), g_{i+1}(C_{i+1}; t_{i+1}), \ldots, g_{i+k}(C_{i+k}, t_{i+k})$ with a common control factor $F = (C_i \cap C_{i+1} \cap \cdots \cap C_{i+k})$ and no overlapping target lines, i.e. $\{t_i\} \cap \{t_{i+1}\} \cap \cdots \cap \{t_{i+k}\} = \emptyset$. Those cascades are substituted with an alternative cascade where

- the first gate of the cascade $g_i(C_i; t_i)$ with control connections C_i and target connection t_i remains unchanged
  and,
- all remaining gates $g_j(C_j; t_j) \in \{g_{i+1}, \ldots, g_{i+k}\}$ are substituted by Toffoli gates of the form $g(\{t_i\} \cup (C_j \setminus F); t_j)$ which are placed before and after the first gate g_i.

[3] A more detailed study on the trade-off between the number of circuit lines and the quantum costs has been conducted in [23].

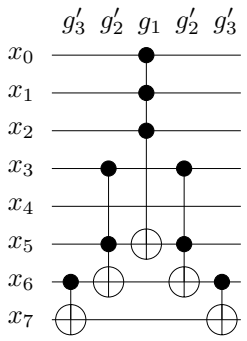


Fig. 5: Exploiting target line activity

Example 4. *Reconsider the reversible circuit shown in Fig. 2. The first gate g_1 and second gate g_2 in the circuit share the control connections $\{x_0, x_1, x_2\}$. Now, we keep the gate g_1 and substitute gate g_2 by adding $g'_2(C'; x_6)$ before and after gate g_1, where, $C' = \{(C_1 \cap C_2) \cup x_5\}$ and C_1 and C_2 are the control sets of gate g_1 and g_2, respectively. In a similar fashion, we reduce the control connections of the last gate g_3. This leads to a circuit shown in Fig. 5 with a reduction in quantum cost from 54 to 26.*

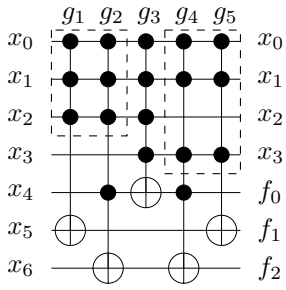
Using the above idea, an algorithm for optimizing the reversible circuits can be formulated. The next section describes the detailed steps of the algorithm.

IV. Algorithm

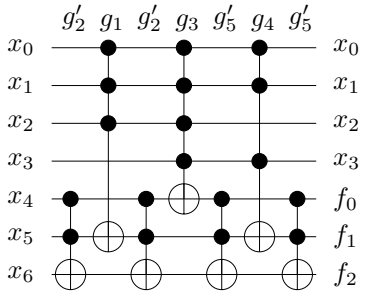
In this section, we now propose an algorithm to reduce the quantum cost of a given reversible circuit based on the idea presented in the previous section. More precisely, we show how to determine the common set of controls from multiple-control Toffoli gates in the circuit. Afterwards, a suitable example illustrates its application.

Algorithm [Common Control Reduction] Consider a reversible circuit G.

1) Traverse all the gates of G from left to right.
2) Extract a sub-circuit G_d consisting of gates $G_d = g_i, g_{i+1}, g_{i+2}, \ldots, g_{i+k}$ such that $(\bigcap_{g_j(C_j;t_j) \in G_d} t_j) = \emptyset$. If no further sub-circuit is left to be considered, terminate.
3) Check whether there exists a common set of control connections, i.e. whether $(\bigcap_{g_j(C_j;t_j) \in G_d} C_j) \neq \emptyset$ holds. If this is *not* the case, continue with another cascade in Step 2.
4) Compute the set of common control lines $F = (\bigcap_{g_j(C_j;t_j) \in G_d} C_j)$.
5) Traverse all gates $g_j(C_j; t_j) \in g_{i+1}, g_{i+2}, \ldots, g_{i+k}$ and
   a) remove each g_j as well as
   b) add a new gate $g(\{t_i\} \cup (C_j \setminus F); t_j)$ before and after gate g_i.
6) Calculate the quantum cost (QC) of the resulting new circuit G'_d.
7) If $(QC(G'_d) < QC(G_d))$, then replace G_d with G'_d. Otherwise, keep the original cascade G_d.
8) Continue in step 2.



(a) Reversible circuit



(b) Optimized circuit

Fig. 6: Exploitation in existing circuit

Example 5. *Consider the circuit shown in Fig. 6a. Applying the steps introduced above, the possible sub-circuits are determined (marked by the dashed rectangle) and then, the functionally equivalent circuits are substituted leading to the circuit shown in Fig. 6b. In this case, the costs of the circuit are reduced from* 88 *to* 68*.*

V. Experimental Evaluation

This section provides a summary of the experimental results obtained by the proposed optimization technique. The method described above has been implemented in C++ and evaluated using circuits from the *RevLib* benchmark library [24]. All experiments have been conducted on an Intel Core i5-2600 with 8 GB of memory.

During the experimental evaluations, circuits optimized by the proposed approach have been compared to the results

- available at the *RevLib* benchmark library [24],
- generated by the approach proposed in [17] which exploits helper lines in order to buffer the value of common control lines as reviewed in Section III-A[4], and
- generated by the approach proposed in [18] which exploits multiple target lines as reviewed in Section III-B.

The results are summarized in Table I. The first two columns give the name of the circuit and the number of circuit lines (#Lines), respectively. In the following columns, the quantum costs for the previously proposed approaches (i.e. from *RevLib*, from [17] with 1 helper line and two helper lines, as well as from [18]) are reported. Finally, the last columns give the quantum costs obtained by the proposed approach as well as the percentage improvement in quantum costs with respect to the previously proposed solutions. Note that, for some circuits, no results were available in [18]; this is denoted by "–" in Table I. All results have been obtained in negligible run-time (i.e. just a few seconds).

The results confirm the benefits of the proposed optimization approach: For *all* considered circuits, substantial improvements with respect to the costs of the original circuit (taken from *RevLib*) are obtained. Moreover, even with respect to the circuits taken from [17], [18] – which have already been optimized – very impressive *further* reductions are observed. In the best cases (i.e. *table3*), the improvements in quantum costs can further be reduced by just over 70%, 60%, and 20% compared to the solutions from [17] with 1 helper line, [17] with two helper lines, and [18], respectively. In contrast, in some cases (e.g. *misex1, C17, cu, dc2, root, sqr6,x2*), the quantum cost of the optimized circuit produced by the proposed method is larger than that of [17]. But considering the significant potential for the other circuits, the proposed approach indeed turns out to be beneficial.

Besides that, the solution proposed here completely overcomes the drawbacks of previously proposed solution. This particularly holds with respect to the method from [17] which relies on adding further lines to the circuit. While this generally allows for an easier cost reduction (as e.g. also observed in [23]), it is a serious drawback for quantum computation. Caused by the fact that the number of circuit lines corresponds to the number of qubits, circuit lines are usually assumed a highly limited resource. The solution proposed here allows for the significant reductions reported above without adding any single circuit line. Compared to [18], the potential of reducing the amount of common control lines is much better exploited. While the approach in [18] is restricted to exact matches between common control lines, our approach also optimizes subsets of common control lines. This enables the further reductions as reported in Table I.

VI. Conclusion

In this work, we proposed an alternative approach for the reduction of common control lines in reversible circuits and, hence, an improvement for quantum cost reduction. The general idea is thereby similar to the methods previously applied in [17], [18]. We combine the methods proposed earlier in a new fashion which enables us to achieve the same or even better improvements. At the same time, drawbacks of the approaches from [17], [18], i.e. the need for additional circuit lines as well as the restricted applicability, respectively, are avoided. Experimental results confirmed these benefits.

Future work focuses on transferring the findings of this work to other gate libraries, e.g. gates additionally composed of negative control lines or based on the library introduced in [25]. Besides that, the consideration of the common control line exploitation as introduced here directly for synthesis methods such as [10], [11], [14] is left for future work.

[4]Here, two comparisons (with one and with two helper lines, respectively) have been conducted.

TABLE I: Experimental evaluation

		Quantum costs obtained by previous approaches				Quantum costs obtained by the proposed approach				
		RevLib [24]	Sect. III-A [17]		Sect. III-B [18]		Improvements w.r.t.			
Circuit	#Lines		(w/ 1 helper line)	(w/ 2 helper lines)			RevLib	[17] w/ 1	[17] w/ 2	[18]
table3_ 209	28	97537	76317	53496	26903	20467	79%	73%	62%	24%
C7552_ 119	21	1458	1036	813	–	339	77%	67%	58%	–
apex4_ 103	28	222208	183087	157792	69846	63272	72%	65%	60%	9%
in0_ 162	26	23016	15673	11777	9410	7022	69%	55%	40%	25%
ex1010_ 155	20	165244	137510	120110	61646	50562	69%	63%	58%	18%
decod_ 137	21	1458	1036	813	510	448	69%	57%	45%	12%
misex3_ 180	28	139047	125578	113218	56627	48559	65%	61%	57%	14%
misex3c_ 181	28	132292	119233	108156	56456	47546	64%	60%	56%	16%
sao2_ 199	14	4858	3963	2378	2274	1876	61%	53%	21%	18%
inc_ 170	16	1823	1271	947	900	730	60%	43%	23%	19%
apla_ 107	22	3384	2452	1900	1620	1430	58%	42%	25%	12%
pm1_192	14	324	216	155	–	137	58%	37%	12%	–
cm42a_125	14	324	216	155	177	139	57%	36%	10%	21%
dist_144	13	5762	4000	3179	3051	2523	56%	37%	21%	17%
misex1_178	15	859	578	427	436	428	50%	26%	0%	2%
dc2_222	15	1720	1188	923	–	930	46%	22%	-1%	–
dk17_224	21	1601	1227	700	–	872	46%	29%	-25%	–
f2_ 158	8	209	164	117	125	115	45%	30%	2%	8%
root_197	13	2645	1795	1344	–	1474	44%	18%	-10%	–
cm163a_213	29	739	650	522	561	421	43%	35%	19%	25%
wim_220	11	199	145	120	–	114	43%	21%	5%	–
mlp4_184	16	3446	2577	2292	–	2152	38%	16%	6%	–
sqr6_204	18	957	690	596	–	607	37%	12%	-2%	–
cu_219	25	1140	864	426	950	804	29%	7%	-89%	15%
x2_223	17	575	415	370	509	429	25%	-3%	-16%	16%
C17_204	7	90	65	50	78	68	24%	-5%	-36%	13%
radd_250	13	587	443	408	–	541	8%	-22%	-33%	–
pcler8_190	21	313	202	159	–	305	3%	-51%	-92%	–

All results have been obtained in negligible run-time (i.e. just a few seconds).

References

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

[2] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Theory of computing*, 1996, pp. 212–219.

[3] P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," *Foundations of Computer Science*, pp. 124–134, 1994.

[4] R. Landauer, "Irreversibility and heat generation in the computing process," *IBM J. Res. Dev.*, vol. 5, p. 183, 1961.

[5] C. H. Bennett, "Logical reversibility of computation," *IBM J. Res. Dev*, vol. 17, no. 6, pp. 525–532, 1973.

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

[7] R. Wille, R. Drechsler, C. Osewold, and A. G. Ortiz, "Automatic design of low-power encoders using reversible circuit synthesis," in *Design, Automation and Test in Europe*, 2012, pp. 1036–1041.

[8] V. V. Shende, A. K. Prasad, I. L. Markov, and J. P. Hayes, "Synthesis of reversible logic circuits," *IEEE Trans. on CAD*, vol. 22, no. 6, pp. 710–722, 2003.

[9] M. Saeedi, M. S. Zamani, M. Sedighi, and Z. Sasanian, "Synthesis of reversible circuit using cycle-based approach," *J. Emerg. Technol. Comput. Syst.*, vol. 6, no. 4, 2010.

[10] D. M. Miller, D. Maslov, and G. W. Dueck, "A transformation based algorithm for reversible logic synthesis," in *Design Automation Conf.*, 2003, pp. 318–323.

[11] D. Maslov and G. W. Dueck, "Reversible cascades with minimal garbage," *IEEE Trans. on CAD*, vol. 23, no. 11, pp. 1497–1509, 2004.

[12] D. Große, R. Wille, G. W. Dueck, and R. Drechsler, "Exact multiple control Toffoli network synthesis with SAT techniques," *IEEE Trans. on CAD*, vol. 28, no. 5, pp. 703–715, 2009.

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

[14] K. Fazel, M. Thornton, and J. Rice, "ESOP-based Toffoli gate cascade generation," in *Pacific Rim Conference on Communications, Computers and Signal Processing*, 2007, pp. 206–209.

[15] Y. Sanaee and G. W. Dueck, "ESOP-based Toffoli network generation with transformations," in *Int'l Symp. on Multi-Valued Logic*, 2010, pp. 276–281.

[16] R. Drechsler, A. Finder, and R. Wille, "Improving ESOP-based synthesis of reversible logic using evolutionary algorithms," in *Applications of Evolutionary Computation*, 2011, pp. 151–161.

[17] D. M. Miller, R. Wille, and R. Drechsler, "Reducing reversible circuit cost by adding lines," in *Int'l Symp. on Multi-Valued Logic*, 2010, pp. 217–222.

[18] R. Wille, M. Soeken, C. Otterstedt, and R. Drechsler, "Improving the mapping of reversible circuits to quantum circuits using multiple target lines," in *ASP Design Automation Conf.*, 2013, pp. 85–92.

[19] T. Toffoli, "Reversible computing," in *Automata, Languages and Programming*, W. de Bakker and J. van Leeuwen, Eds. Springer, 1980, p. 632, technical Memo MIT/LCS/TM-151, MIT Lab. for Comput. Sci.

[20] A. Barenco, C. H. Bennett, R. Cleve, D. DiVinchenzo, N. Margolus, P. Shor, T. Sleator, J. Smolin, and H. Weinfurter, "Elementary gates for quantum computation," *The American Physical Society*, vol. 52, pp. 3457–3467, 1995.

[21] D. M. Miller, R. Wille, and Z. Sasanian, "Elementary quantum gate realizations for multiple-control Toffolli gates," in *Int'l Symp. on Multi-Valued Logic*, 2011, pp. 288–293.

[22] R. Wille, O. Keszöcze, and R. Drechsler, "Determining the minimal number of lines for large reversible circuits," in *Design, Automation and Test in Europe*, 2011, pp. 1204–1207.

[23] R. Wille, M. Soeken, D. M. Miller, and R. Drechsler, "Trading off circuit lines and gate costs in the synthesis of reversible logic," *Integration*, vol. 47, no. 2, pp. 284–294, 2014.

[24] 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 http://www.revlib.org.

[25] Z. Sasanian, R. Wille, and D. M. Miller, "Realizing reversible circuits using a new class of quantum gates," in *Design Automation Conf.*, 2012, pp. 36–41.