Circuits for Reversible Computing
A 4-Pages Tutorial

Claudio Moraga

Faculty of Computer Science, TU Dortmund University, 44221 Dortmund, Germany
Department of Informatics, Technical University "Federico Santa María", Valparaíso, Chile

Abstract Reversible circuits are characterized to be information preserving. They realize bijections on given domains. Their relevant features are their low power consumption and their proximity to circuits for quantum computing. Circuits for quantum computing follow the laws of Quantum Mechanics and therefore, they are reversible. Many research efforts are dedicated to the development of efficient synthesis algorithms for reversible circuits.

INTRODUCTION

The earliest contributions to the development of reversible computing circuits may be traced back to E. Fredkin [14] and T. Toffoli [36] who introduced the first controlled, functionally complete, reversible gates with three inputs and three outputs. The Fredkin gate has one control input and two target inputs. If the control input has the value 1, the gate becomes active and exchanges (the positions of) the targets. If the control input has the value 0, the gate becomes inhibited and behaves as an identity. The Toffoli gate has two control inputs and one target input. If *both* control inputs have the value 1 then the gate becomes active and the target input will be complemented. If *any* of the control inputs has the value 0, the gate becomes inhibited and behaves as an identity preserving the target.

If a Toffoli gate is constrained to work with $c_1 = c_2$, this originates a "controlled NOT" (CNOT) reversible gate, which behaves as a Toffoli gate under the constrain, but has obtained an own symbol, with obviously just one control signal. (See Fig.1). It should also be recalled that a simple inverter satisfies the reversibility conditions: it realizes a bijection in {0, 1}. The EXOR symbol is used to represent inverters –(controlled or not)– in the community of reversible/quantum circuits. The set {⊕, CNOT, Toffoli} is functionally complete and has become the standard basis for the realization of reversible circuits. Also the Fredkin gate has been given a representation as a subcircuit on this basis. Figure 1 shows the symbols of all these gates.

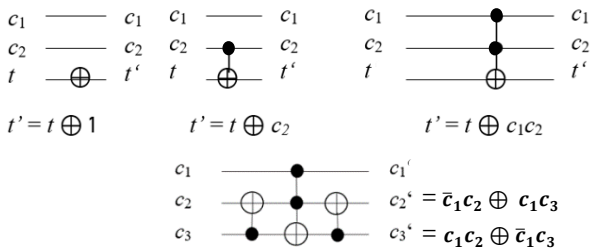


Fig. 1: Symbols and functionality of the NOT, CNOT, Toffoli gates and of the Fredkin sub-circuit.

The realization of reversible circuits as fanout-free and feedback-free cascades of reversible gates was stimulated by R. Landauer's Theorem [20] stating that erasing or deleting information in a circuit would produce heat dissipation. See [8]. Moreover, C. Bennet [7] showed that a computer could work with low power dissipation if *all* its circuits would be reversible.

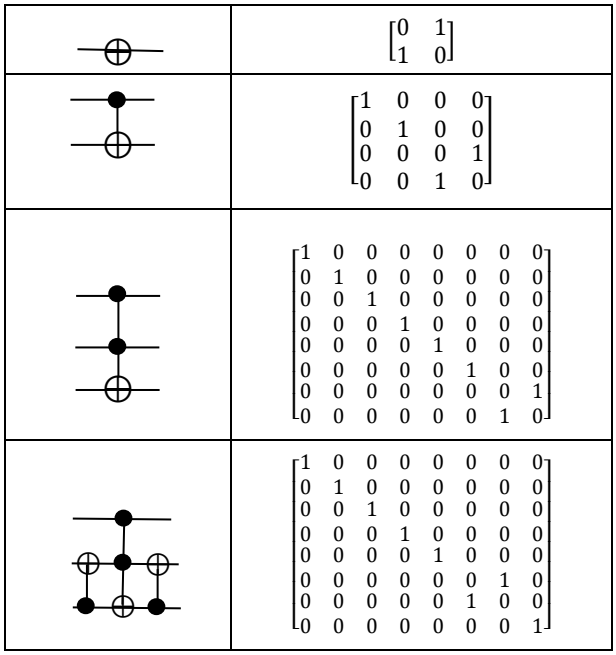
The Toffoli gate has intensively been used ever since and has received several "extensions", like multi-controlled Toffoli gates and their decomposition as V-shaped cascades of elementary Toffoli gates and ancillary bits [6], a quantum realization model [6], the Peres gate [30], mixed polarity controlled Toffoli and Peres gates [25], [26], double gates [28], disjunctive controlled Toffoli and Peres gates [27], and Clifford-T realizations [2], [3] as well as mappings to the IBM QX quantum computers [1].

FORMALISMS

Reversible gates may be formally specified with matrices and in the case of circuits, an appropriate composition of the gates-matrices –(see below)– provides the transfer function of the circuit.

Fig. 2 shows the matrix specification of the basic reversible gates.

For the specification of circuits, the so-called Swap gate, as a model of twisting neighbor lines, is very useful. Its symbol has been borrowed from quantum computing. This is shown in Fig. 3. Notice that a Fredkin gate has the functionality of a controlled Swap.



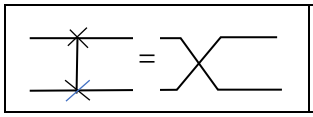
$$\begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}$$

$$\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 1 & 0\end{bmatrix}$$

$$\begin{bmatrix}1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1\\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0\end{bmatrix}$$

$$\begin{bmatrix}1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1\end{bmatrix}$$

Fig. 2: Matrix specification of the basic reversible gates and of the Fredkin sub-circuit



$$\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

Fig.3: Symbol, equivalence, and specification of a Swap gate

For the realization of circuits, basic gates may be used in a way different to their basic specifications, e.g, "up-side-down". In this case their transfer function may be calculated with the help of Swap gates.

Example 1: Let NC denote an up-side-down CNOT and let **S** denote a Swap gate. Then:

$$\text{NC} = \mathbf{S}\cdot\text{CNOT}\cdot\mathbf{S} =$$

$$= \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\cdot\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 1 & 0\end{bmatrix}\cdot\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix} =$$

$$= \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 1 & 0\\ 0 & 1 & 0 & 0\end{bmatrix} \qquad (1)$$

It is fairly obvious that for the realization of circuits, a transfer matrix as in (1) should be calculated only once and be saved in a library of e.g. "generalized gates".

A usual method to calculate the transfer function of a circuit consists of calculating the transfer of bitslices and then calculating their cascade. A quite frequent situation comprises bit-slices with basic gates embedded in a network of several variables.

Example 2: Using the notation Gate_name(control; target), consider the case of a gate in a bitslice of three variables x_1, x_2, x_3. Which is the transfer matrix of CNOT(x_3; x_1)?

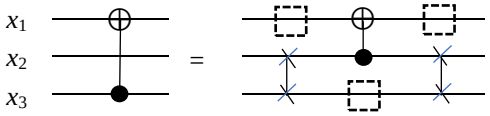


Fig. 4: Equivalent sub-circuit for CNOT(x_3; x_1)

Fig. 4 shows an equivalent sub-circuit to make the calculation. The dotted boxes on the "empty" line segments denote 2x2 identities **I**. The transfer matrix of each bitslice is obtained as the Kronecker product of the matrices of the corresponding gates. Therefore,

$$\text{CNOT}(x_3; x_1) = (\mathbf{I}\otimes\mathbf{S})\cdot(\text{CNOT}(x_2; x_1)\otimes\mathbf{I})\cdot(\mathbf{I}\otimes\mathbf{S}) =$$

$$= \begin{bmatrix}\mathbf{S} & 0\\ 0 & \mathbf{S}\end{bmatrix}\cdot\begin{bmatrix}\mathbf{I} & 0 & 0 & 0\\ 0 & 0 & 0 & \mathbf{I}\\ 0 & 0 & \mathbf{I} & 0\\ 0 & \mathbf{I} & 0 & 0\end{bmatrix}\cdot\begin{bmatrix}\mathbf{S} & 0\\ 0 & \mathbf{S}\end{bmatrix}, \qquad (2)$$

where the 0s represent 0-matrices of the dimension of **S** and **I**, respectively.

A direct calculation of (2) gives:

$$\text{CNOT}(x_3; x_1) = \begin{bmatrix}1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1\\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0\end{bmatrix} \qquad (3)$$

DESIGN OF REVERSIBLE CIRCUITS

Reversible circuits comprise cascades of fanout-free reversible gates and do not have feedbacks. The design of optimal (irreversible) binary digital circuits is NP complete [37]. Due to the additional constraints posed by reversibility, the design of optimal reversible circuits with constrained ancillary lines is NP-hard [31]. Therefore, heuristic strategies have been developed for the design of reversible circuits [9], [32]. In the case of reversible circuits with a reduced number of variables, design alternatives based on ESOPs [10], [13], as well as on Reed-Muller polynomial expressions with mixed polarity have been used [25]. Also design of reversible circuits based on Decision Diagrams [35], and using SAT-solvers [15] has been considered. It has also been shown that pairs of reversible gates may exhibit a symbiotic relationship contributing to improve prevailing synthesis algorithms [28]. Controlling gates with 0-valued control signals (identified with white dots) has added an effective flexibility to design algorithms [25], [26], [33].

Furthermore, Toffoli and Peres gates with disjunctive control have been developed [25], [27]. The possibly best known general synthesis algorithm, known as "MMD", the initials of their authors, was introduced in 2003 as a greedy synthesis algorithm [22], and has received important extensions [24] and improvement as e.g. "bottom-up" MMD [4] and inputs reordering [18], [34]. It should be recalled that early this century, the "standard" desktop computer capabilities were in the order of a Pentium 3, running close to 1 GHz and with about 250 MB of RAM. On the other hand, we presently have laptops running over 2 GHz and having over 10 GB of RAM. This increasing computing power of the last decades has opened new improvements for design alternatives as designing reversible circuits with evolutionary algorithms [21], [16], distributed design on clusters [17], and introducing bounded search in MMD [24]. Furthermore, the design of 3-valued reversible circuits has gained interest, [5], [11], [23], considering the aspects of higher information density. Notice that in the binary case, there are $2^3! = 40{,}320$ reversible functions on *three* variables, whereas there are $3^2! = 9\cdot(2^3!) = 362{,}880$ ternary reversible functions on just *two* variables.

An important development of the recent decade is the accessibility of (small) quantum computers via internet [19], where reversible and quantum computing circuits "meet" in the context of Clifford+T based circuits [12], [1], [2], [3], [29] and face both new constrains and challenging perspectives.

Example 3:

A particular property of reversible circuits, not found in classical digital circuits, is that of *borrowing and returning* bits. This is illustrated in the possibly simplest case: that of realizing a 3-controlled Toffoli gate using standard Toffoli gates and an auxiliary bit of *unknown binary value*, which is returned after using it. This is shown in Fig. 5.

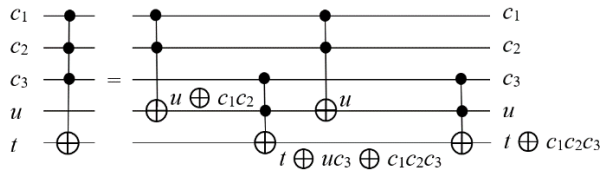


Fig. 5: Realization of a Toffoli gate with 3 controls using standard Toffoli gates and an auxiliary bit of unknown value.

Another important property is the simple realization of irreversible circuits embedded in a reversible one with auxiliary bits known as ancillae.

Example 4: Reversible realization of a multiplier of two 2-bit factors following the "school method". (Fig. 6).

$x_1 \quad x_0 \cdot y_1 \quad y_0$ $x_1y_0 \quad x_0y_0$ $x_1y_1 \quad x_0y_1$ $p_3 \quad p_2 \quad p_1 \quad p_0$	$p_0 = x_0y_0$ $p_1 = x_1y_0 \oplus x_0y_1$ $c_1 = x_0y_0x_1y_1$ $p_2 = x_1y_1 \oplus c_1$ $c_2 = c_1x_1y_1 = c_1$ $p_3 = c_2 = c_1 = (x_0y_0)(x_1y_1)$

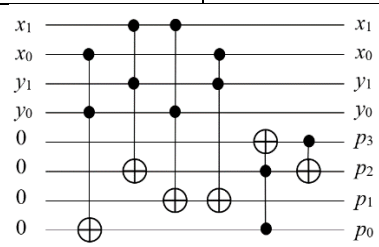


Fig. 6: Reversible realization of an irreversible multiplier

The MMD synthesis algorithm, in its original version [22] was meant to work with the set {⊕, CNOT, Toffoli}, it processed the inputs top-down in lexicographic order and comprised a sequence of elementary permutations, each one transforming an input to the corresponding output (or the other way around), but preserving all former transformations. In [22] it was stated that the permutation expressed as $\left(\frac{0}{7}\frac{1}{1}\frac{2}{4}\frac{3}{3}\frac{4}{0}\frac{5}{2}\frac{6}{6}\frac{7}{5}\right)$ in the Cauchy notation, would be the most difficult to be realized as a 3×3 reversible circuit. To show the effectiveness of some of the improvements that have been contributed to MMD [22], we will design a reversible circuit for the mentioned permutation using MMD in the "backward mode" with the following additional features:

i) Input-output pairs split in two blocks. An upper block where changes are needed and a lower block where the inputs should simply be preserved. Within the blocks, the decreasing Hamming-weight of the inputs gives the order.

ii) Gates with mixed polarity. (Control signals that at the algorithmic level are effective when their value is 0, are identified by white dots.)

iii) Allow Fredkin gates (as controlled Swaps).

The synthesis steps are shown in Table 1. The obtained circuit with just 4 gates is shown in Fig. 7.

Table 1. Backward synthesis with "extended" MMD

Input		Output		Op.1	Op.2	Op.3	Op.4
xyz		*x'y'z'*					
111	7	101	5	**111**	**111**	**111**	**111**
101	5	010	2	010	100	**101**	**101**
100	4	000	0	000	000	000	**100**
010	2	100	4	100	010	010	**010**
000	0	111	7	101	101	100	**000**
011	3	011	3	011	011	011	**011**
110	6	110	6	110	110	110	**110**
001	1	001	1	001	001	001	**001**
Op.1: $y \oplus xz$; Op.2: if z=0, swap(x,y) Op.3: $z \oplus x(y \oplus 1)$; Op.4: $x \oplus (y \oplus 1)(z \oplus 1)$							

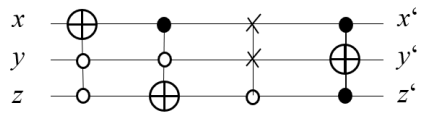


Fig. 7: Reversible circuit for the permutation (7 1 4 3 0 2 6 5)

CLOSING REMARK

Basic aspects of reversible circuits have been presented and a comprehensive list of references is provided to allow an in-depth covering of further properties and particular features of these promising circuits.

REFERENCES

[1] A.A.A. Almeida, G.W. Dueck, A.C. Rodrigues da Silva, "Efficient Realization of Toffoli and NCV Circuits for IBM QX Architectures," M.K. Thomsen and M. Soeken (Eds.), RC 2019, LNCS 11497, 131-145, Springer, 2019.

[2] M. Amy , D. Maslov, M. Mosca, and M. Roetteler, "A meet-in-the-middle algorithm for fast synthesis of depth-optimal quantum circuits," *IEEE Trans. Computer-Aided Design of Integrated Circuits and* Systems, 32, 818-830, 2013. Also arXiv:1206.0758, 2012.

[3] N. Abdessaied, M. Amy, M. Soeken, and R. Drechsler, "Technology mapping of reversible circuits to Clifford+T quantum circuits," in *Proc. 2016 46th Int. Symp. on Multiple-Valued Logic*, 150–155, IEEE Press, 2016.

[4] C. Barbieri, *Algoritmos para a Síntese de Circuitos Reversíveis Ternários: Análise Comparativa.* Ph.D. Thesis, Universidade Estadual Paulista "Júlio De Mesquita Filho", Sao Paulo, Brazil. (In Portuguese), 2018.

[5] C. Barbieri and C. Moraga, "Cycles-based and Transformation-based Synthesis of Ternary Reversible Circuits. Aspects of Complexity." *Journal of Applied Logics*, 8 (5), 1295-1309, 2021.

[6] A. Barenco, C.H. Bennett, R. Cleve, D.P. Di Vincenzo, N. Margolus, P. Shor, T. Sleator, J.A. Smolin, and H. Weinfurter, "Elementary gates for quantum computation," *Phys. Rev.* A 52, 3457–3467, 1995.

[7] C. Bennett, "Logical reversibility of computation," *IBM J. of Research and Development*, 17, 525-532, 1973.

[8] A. Bérut et al. "Experimental verification of Landauer's principle linking information and thermodynamics". *Nature* 483, 187-189. doi: 10.1038/nature10872, 2012.

[9] C.S. Cheng, and A.K. Singh, "Heuristic Synthesis of Reversible Logic – A Comparative Study," *Theoretical and Applied Electrical Engineering*, 12, (3), 210-225, 2014.

[10] R. Drechsler, A. Finder, and R. Wille. "Improving ESOP-based synthesis of reversible logic using evolutionary algorithms." In *Proceedings of EvoApplications (2)*. LNCS, 6625, 151–161, Springer, 2011.

[11] A. De Vos, and Y. van Rentergem, "Multiple-valued Reversible Logic Circuits," *J. Multiple-Valued Logic and Soft Computing*, vol. 15 (5-6), 489-506, 2009.

[12] J.M. Farinholt, "An ideal characterization of the Clifford operators,".lanl.gov/abs/1307.5087, 2014.

[13] K. Fazel, M.A. Thornton, and J. E. Rice. "ESOP-based Toffoli gate cascade generation." In *Proc. IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*. 206–209, 2007.

[14] E. Fredkin and T. Toffoli. "Conservative Logic". I*nternational Journal of Theoretical Physics* 21 (3-4) 219-253, 1982.

[15] M. Grosse, R. Wille, G.W. Dueck, and R. Drechsler. "Exact multiple-control Toffoli network synthesis with SAT techniques." *IEEE Trans. CAD* 28 (5), 703-715, 2009.

[16] F.Z. Hadjam, and C. Moraga, "RIMEP2. Evolutionary Design of Reversible Digital Circuits," *ACM Jr. of Emerging Technologies in Computing Systems*, 11(3), 27:1-27:23, 2014.

[17] F.Z Hadjam, and C. Moraga, "Distributed Evolutionary Design of Reversible Circuits." In B. Steinbach, (Ed.), *Further Improvements in the Boolean Domain*, 372-383. Cambridge Scholars Publishing, 2018.

[18] M.M. Hawash. "Methods for Efficient Synthesis of Large Reversible Binary and Ternary Quantum Circuits and Applications of Linear Nearest Neighbor Model." Ph.D. Dissertation Portland State University, Portland OR., 2013.

[19] IBM-Q. https://www.research.ibm.com/ibm-q/.

[20] R. Landauer, "Irreversibility and heat generation in the computing process," *IBM J. Res. Dev*., 5, 183-191, 1961.

[21] M. Lukac, M. A. Perkowski, H. Goi, M. Pivtoraiko, Ch. H. Yu, K. C., Hyunkoo Jeech, B.-G. Kim, and Y. D. Kim, "Evolutionary Approach to Quantum and Reversible Circuits Synthesis." *Artif. Intel. Rev*. 20, 3, 361–417, 2003.

[22] D.M. Miller, D. Maslov, and G.W. Dueck, "A transformation-based algorithm for reversible logic synthesis," In *Proc DAC 2003*, ACM Press, 2003.

[23] D.M. Miller, G.W. Dueck, and D. Maslov, "A synthesis method for MVL reversible logic," *Proc. 34th Int. Symp. on Multiple-Valued Logic*, 74-80. IEEE-CS-Press, 2004.

[24] D.M. Miller, and G.W. Dueck, "Search-Based Trans-formation Synthesis for 3-Valued Reversible Circuits," I. Lanese, and M. Rawski (Eds.), *Reversible Computation*, LNCS 12227, 218-236, Springer, 2020.

[25] C. Moraga, "Hybrid GF(2)-Boolean Expressions for Quantum Computing Circuits," A. De Vos and R. Wille (Eds.), *RC 2011*, LNCS 7165, 54-63, Springer, 2012.

[26] C. Moraga, Mixed polarity reversible Peres gates. *Electronics Letters.* 50 (14), 987-989, 2014.

[27] C. Moraga, "OR-Toffoli and OR-Peres reversible gates". In *Reversible Computation* (S.Yamashita, T.Yokoyama, Eds.), LNCS 12805, 266-273, Springer, 2021.

[28] C. Moraga and F.Z. Hadjam, "On double gates for reversible computing circuits." *Proc. Int. Workshop on Boolean Problems*. 131-138. Freiberg, 2012.

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

[30] Peres A., "Reversible Logic and Quantum Computers." *Physic Review*, 32, (6),3266–3276, 1985.

[31] I. Rahul, B. Loff, and I.C. Oliveira, "NP-hardness of circuit minimization for multi-output functions." in *Proc. 35th Computational Complexity Conference (CCC 2020)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.

[32] M. Saeedi, and I.L. Markov, "Synthesis and optimization of reversible circuits – a survey." *Comput. Surveys*, arXiv 1110.2574v1.

[33] M. Soeken, and M.K. Thomsen, "White Dots *do* Matter: Rewriting Reversible Logic Circuits," G.W. Dueck and D.M. Miller (Eds.), *RC 2013*, LNCS 7948, 196-208, Springer, 2013.

[34] M. Soeken, G.W. Dueck, M.D. Miller, "A fast symbolic transformation-based algorithm for reversible logic synthesis," Devitt S., Lanese I., (Eds.). *Reversible Computation*, LNCS 9720, 307-321, Springer, 2016.

[35] S. Stojković, M. Stanković, and C. Moraga, "Complexity reduction of Toffoli networks based on FDD," *Facta Universitatis.* Series Electr. Eng., 28 (2), 251-262, 2015.

[36] T. Toffoli, "Reversible Computing," J.W. Baker and J. van Leeuwen (Eds.), ALP 1980, LNCS 84, 6732-644, Springer, 1980.

[37] I. Wegener, *The CompIexity of Boolean Functions.* John Wiley and Sons, New York, 1987.