UDC 519.6

Toffoli gate implementation method based on Margolus gate on four or more qubits

Andrii Tereshchenko[1], Valeriy Zadiraka[2]

[1] *V. M. Glushkov Institute of Cybernetics of the National Academy of Sciences, Kyiv, Ukraine*
[2] *V. M. Glushkov Institute of Cybernetics of the National Academy of Sciences, Kyiv, Ukraine*

Abstract

This paper considers the method of building the Toffoli Gate based on the Margolus Gate on four or more qubits. In the first part of the considered method, the modification of the Margolus gate on four or more qubits is suggested. In the second part of the method, the modification of the Margolus gate is transformed into the modification of the Toffoli gate implementation using step-by-step phase rotation compensations. The phase rotation compensation for an N-qubit quantum circuit can be performed with N successive steps, where at each step the gates with phase rotation $\pm\pi/2^{s-1}$ are added, where s is the step number, starting from one. The compensation phase requires 2 two-qubit gates and 2N-1 one-qubit gates.

Keywords: The Toffoli Gate, the Margolus Gate, Quantum Gate, Universal Gate, Reversible Gate, Quantum Circuit

Introduction

The emergence of various new computer systems is associated with the solution of applied problems in various fields. It is expected that the advent of quantum computers will allow solving problems that cannot be solved even with the use of all available computing systems and years of continuous computing. The publication of Shor's algorithm for solving the factorization problem of large numbers [1] was one of the biggest impacts on accelerating the development of quantum computers. The architecture of the computer, which includes two-level quantum mechanical systems (qubits), defined quantum arithmetic in the form of elementary operations performed by gates, the sequential and parallel execution of which forms a quantum circuit. Unfortunately, a quantum computer does not support classical operations such as multiplication [2–4], addition, subtraction, etc. It is theoretically proven that any classical algorithm can be transferred to the quantum model of computation. There is a need to transfer the arithmetic operations [5–7] to the quantum model of computation for solving problems with trans-computational complexity. It is very important to have universal methods and tools for implementing algorithms of high complexity. One such universal tool is the Toffoli gate, which allows the implementation of mathematical models of algorithms in "hardware", which significantly speeds up calculations.

1. The Toffoli Gate

In addition to the sequential and parallel computational models, the Toffoli gate has also found its uses in the quantum computational model. It is proven that any classical reversible logic operations can be translated for the quantum computational model, but it cannot be claimed that in the quantum model, these implementations will be optimal, even if they are optimal for the sequential or parallel computational models.

Figure 1 shows a graphic representation of the Toffoli gate. The states of the A and B qubits are not changed, which is important for reversible calculations [8–10] for quantum computing. The qubits A and B are called control qubits, and C is called the controlled qubit.

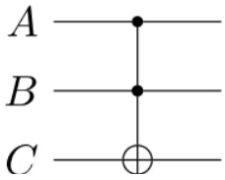


Figure 1: Graphic representation of the Toffoli gate

Figure 2 presents the truth table of the Toffoli gate based on XOR and AND elements.

A	B	C	$C \oplus (A \cdot B)$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

Figure 2: Truth table

The truth table displays the dependence of the result in the right column on the input parameters. The table shows that the consequence C is changed only if A=B=1.

Figure 3 shows a matrix representation of the Toffoli gate operation (qubit numbering starts from the top/left).

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

Figure 3: CCNOT Permutation table

The Toffoli gate, elements NOT and XOR form a universal set for classical calculations, although the NOT and XOR elements can be implemented based on the Toffoli gate and the input states $|0\rangle$ and $|1\rangle$ as shown in Figure 4.

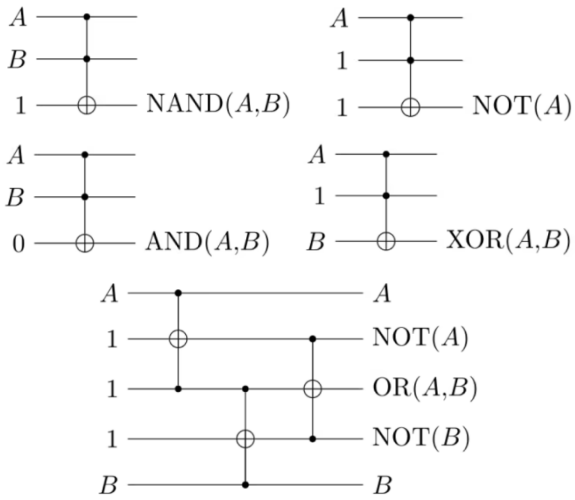


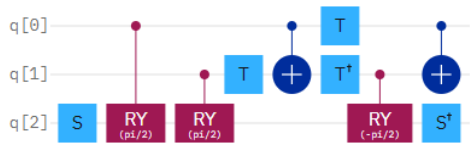
Figure 4: Implementation of NAND, NOT, AND, OR, XOR elements based on the Toffoli gate

The usage of the Toffoli gate facilitates the transfer of classical computation to the quantum computational model. Since the Toffoli gate is universal, it is very convenient to estimate the complexity of calculations by the number of gates. In practice, other universal gates such as Fredkin and Peres are used, but the Toffoli gate is the most common and convenient to use, so the main attention is paid to the analysis of the Toffoli gate.

1.1. Implementation of the Toffoli gate in the quantum computational model

The physical implementation of the Toffoli gate has a long and rich history. There are many implementations of the Toffoli gate using different basis gates. DiVincenzo and Smolin showed that five two-qubit gates are necessary and sufficient to implement the Toffoli gate [11–13].

The authors studied the implementation scheme, which is given in Figure 5. It can be seen from the table that the initial states of the qubits $|0\rangle$ and $|1\rangle$ after calculating the scheme will also be $|0\rangle$ and $|1\rangle$. If other states appear during the implementation of the scheme, for example, such as $|i\rangle$, $|-i\rangle$, $|-1\rangle$ and others, then this means that the result of the scheme calculation has a phase rotation for such an initial state. In Figure 5 gates S, S', T, T' are used to compensate for the phase rotation of the circuit calculation result.



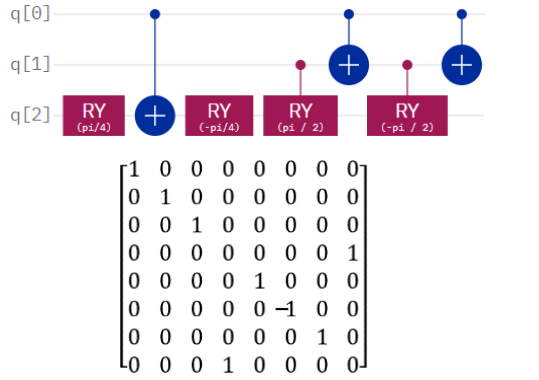
$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figure 5: The Toffoli gate implementation scheme based on the CRY, CX, S, S', T, T' gates and the representation of the Toffoli gate in matrix form

The scheme in Figure 5 is not convenient for constructing a higher-order Toffoli gate. This is because the two-qubit gates CRY and CX combine different pairs of qubits (0-1 (CX), 0-2 (CRY), 1-2 (CRY)).

1.2. Implementation based on the Margolus gate

Let's replace the gate 0-2 (CRY) in Figure 5 with the gates RY($\pi/4$), CX, RY($-\pi/4$) and we obtain the Margolus modification [4] in Figure 6.



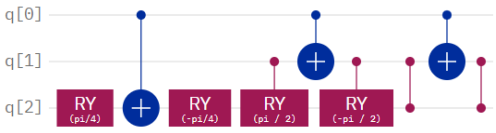
$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figure 6: The implementation scheme of the Margolus gate based on pairs of gates RY, CRY, CX and the representation of the Margolus gate in matrix form

The Margolus gate differs from the Toffoli gate implementation in that the result of the calculation scheme of the initial combination $|101\rangle$ has a phase rotation by π (or $-\pi$). The phase rotation by π in the matrix form corresponds to the value -1 in Figure 6.

Lemma 1. The compensation of the phase rotation by π of the modification of the Margolus gate implementation based on three pairs of gates CY, CRY, CX for the initial state $|101\rangle$ can be performed using a pair of gates CZ, gates S and S', a pair of gates T (phase rotation by $\pi/4$), and one gate T' (phase rotation by $-\pi/4$).

Proof. The compensation of the phase rotation by π for the initial state $|101\rangle$ (Figure 6) can be done in three steps. In the first step, let's add the CZ gates, as shown in Figure 7.

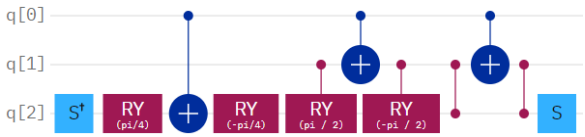


$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Figure 7: The implementation of the Margolus gate with the transfer of the phase rotation from the state $|101\rangle$ to $|011\rangle$ and the representation in matrix form

It makes it possible to transfer the phase rotation of the result from the initial state $|101\rangle$ to the state $|011\rangle$.

In the second step, after adding gates S and S', the phase rotation is "distributed" to the states $|011\rangle$ and $|111\rangle$ with a phase rotation by $3\pi/2$, which corresponds to the value of $-i$ in the matrix representation in Figure 8.



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

Figure 8: The implementation scheme of the Margolus gate with the transfer of the phase rotation of the result to the states $|011\rangle$, $|111\rangle$ and representation in matrix form

In the third step, the final adjustment takes place due to the addition of a pair of gates T and one gate T'. We get the modification in Figure 9, because of which there are no phase rotations for all initial states.

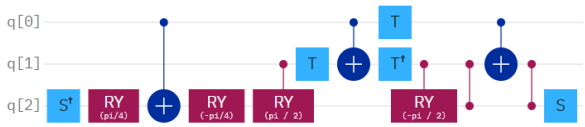


Figure 9: Implementation scheme of the Toffoli gate based on the Margolus gate

The lemma is proved.

2. Implementation of the Toffoli gate on four qubits

Let's replace the CX gates in Figure 6 with CCX gates (Toffoli gates) and we obtain a modification of the implementation of the Margolus gate on four qubits, for which there is a phase rotation for the initial state $|1011\rangle$.

Lemma 2. The compensation of the phase rotation by π of the modification of the Margolus gate based on three pairs of gates CY, CRY, CX for the initial state $|1011\rangle$ can be performed with the help of a pair of gates CZ, gates S and S', pairs of gates CX, gates T and T', pairs of gates P($\pi/8$) and gate P($-\pi/8$).

In Figure 10 sequential addition of gates for phase rotation compensation is highlighted by rectangles of different colors.

Comparing Figure 10 and Figure 11, you can see that one gate T is replaced with a group of gates, which are highlighted by a rectangle in the upper left corner of Figure 11.

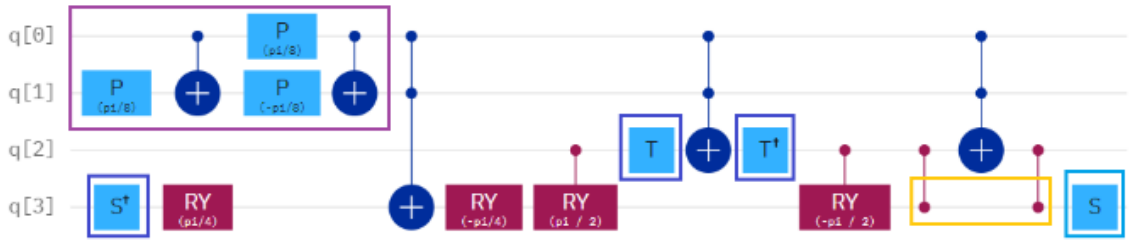


Figure 10: Implementation scheme of the Toffoli gate based on the Margolus gate on four qubits

Phase rotation compensation already consists of four steps. In the first step, CZ gates are added with phase rotation by π, in the second step, S and S' gates are added with phase rotation by $\pi/2$ and $-\pi/2$, in the third step, T and T' gates are added with phase rotation by $\pi/4$ and $-\pi/4$, in the fourth step P gates are added with phase rotation by $\pi/8$ and $-\pi/8$.

Thus, the general method can be described.

3. A method of constructing an N-qubit Toffoli gate based on the Margolus gate

The method consists of two parts.

3.1. Constructing the Margolus gate on four or more qubits

To obtain the N-qubit scheme of the Margolus gate, it needs to replace the CX gates in Figure 6 with C...CX gates (N–1-qubit Toffoli gate schemes). For this modification, there is a phase rotation only for the initial state $|101 \ldots 1\rangle$, where three dots denote N–4 elements. The following method is then used.

3.2. Phase rotation compensation method for the initial state $|101 \ldots 1\rangle$

At each step, a pair of gates with the same value of $\pi/2^{s-1}$ (where s is the number of the step, starting from one), but opposite in sign of rotation, is added phases. Although a pair of CZ gates is added in the first step, we can assume that the rotations on π and $-\pi$ are identical for the CZ gate in this sense. At the last step, instead of a pair, three gates are added, where two gates have a positive phase value, and one has a negative phase value. In general, phase rotation compensation requires 2N+1 additional gates: CZ, S, S', T, ..., P($\pi/2^{N-1}$), P($-\pi/2^{N-1}$), where CZ are two-qubit gates and the rest are single-qubit gates.

Remark. To implement the Toffoli gate on four qubits, an additional pair of CX gates is required (Figure 10), and to implement it on five qubits in addition to the CX pair, a CCX pair is required (Figure 11), etc. If the implementation of the CCX scheme (Toffoli gate) includes CX gates for the same pair of qubits, then this part of the scheme can be optimized and does not require an additional pair of CX gates. Similarly, if CCX gates are present in the implementation of the CCCX scheme (Toffoli gate on 4 qubits), then in this part of the scheme the optimization can be carried out in such a way that it will not require an additional pair of CCX gates. However, within this work, such optimization is not considered.

We will use the method to implement a Toffoli gate on five qubits.

4. Implementation of the Toffoli gate on 5 qubits

To obtain a 5-qubit implementation scheme of the Margolus gate, we will replace the CX gates in Figure 6 with CCCX gates. The phase rotation compensation will consist of five steps, where three gates with a phase rotation of $\pi/16$ and $-\pi/16$ will be added in the last step, as shown in Figure 11.

In Figure 11, the green rectangle shows the last two steps of phase rotation compensation. Note that in the green rectangle, there are pairs of gates CX and CCX.

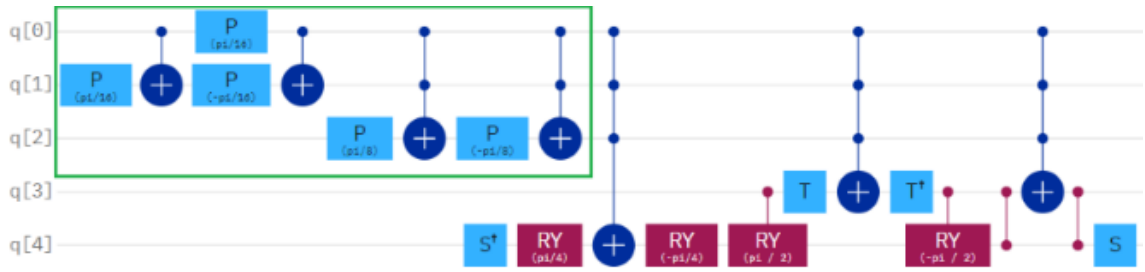


Figure 11: Implementation schemes of the Toffoli gate on five qubits

Conclusions

This work considers the method of constructing the Toffoli gate based on the Margolus gate with four or more qubits. It is proposed to build the Toffoli gate based on the modification of the Margolus gate, which uses three pairs of gates RY, CRY, CX. The gate of higher orders are constructed by replacing CX gates with CCX gates for a four-qubit circuit, CCCX for a five-qubit circuit, etc. The method of phase rotation compensation of the calculation result of the initial state of the form $|101 \ldots 1\rangle$, where three dots denote N–4 elements (N>3). It is shown that the sequential addition of 2N+1 gates (two of which are two-qubit gates and the rest gates are one-qubit gates) compensates the phase rotation for N-qubit gate implementation scheme, and transforms the modification of the Margolus gate to the modification of the Toffoli gate. The addition of 2N+1 gates can be performed with N successive steps, where at each step gates with phase rotation $\pm\pi/2^{s-1}$ are added, where s is the step number, starting from one. The results of the work were tested using IBM quantum simulator [14].

References

[1] P. W. Shor, “Algorithms for quantum computation: discrete log and factoring”, Proc. 35th Ann. Symp. on the Foundations of Computer Science (IEEE Computer Society Press, Los Alamitos, CA, 1994), p. 124.

[2] Draper, T. G. (2000). Addition on a quantum computer. arXiv preprint quant-ph/0008033.

[3] Larasati, H.T.; Awaludin, A.M.; Ji, J.; Kim, H. Quantum Circuit Design of Toom 3-Way Multiplication. Appl. Sci. 2021, 11, 3752. https://doi.org/10.3390/app11093752

[4] V. Vedral, A. Barenco, A. Ekert, Quantum networks for elementary arithmetic operations, www.arxiv.org quant-ph/9511018 v1 (1995).

[5] V. K. Zadiraka, A. M. Tereshchenko, Calculating the Sum of Multidigit Values in a Parallel Computational Model. Cybernetics and Systems Analysis. 2022. № 58. P. 473–480.

[6] A. Tereshchenko, V. Zadiraka, “Algorithm for calculation the carry and borrow signs in multi-digit operations in the parallel computational model”, International Journal of Computing, 22(1), 21-28. (2023).

[7] Tereshchenko A., Zadiraka V. Generating Big Numbers for Testing Multi-Digit Arithmetic Algorithms. Cybernetics and Computer Technologies, 2021, ISSUE 2, P. 39-56.

[8] D. Deutsch, A. Barenco, and A. Ekert, “Universality in quantum computation”, submitted to Proc. R. Soc. Lond. (1995).

[9] T. Toffoli “Reversible Computing”, in Automata, Languages and Programming, eds.J. W. de Bakker and J. van Leeuwen (Springer, New York, 1980), p. 632; Technical Memo MIT/LCS/TM-151, MIT Lab. for Comp. Sci. (unpublished).

[10] A. Peres, “Reversible logic and quantum computers”, Phys. Rev. A 32, 3266 (1985).

[11] D. P. DiVincenzo and J. A. Smolin, Results on two-bit gate design for quantum computers, Proc. of the Workshop on Physics and Computation (1994).

[12] Barenco, Adriano; Bennett, Charles H.; Cleve, Richard; DiVincenzo, David P.; Margolus, Norman; Shor, Peter; Sleator, Tycho; Smolin, John A.; Weinfurter, Harald (1995-11-01). "Elementary gates for quantum computation". Physical Review A. 52 (5). American Physical Society (APS): 3457–3467. arXiv:quant-ph/9503016.

[13] D. P. DiVincenzo, Quantum gates and circuits. Proc. R. Soc. Lond. A, 454:261276, (1998)

[14] https://quantum.ibm.com/composer