Journal of Theoretical and Applied Information Technology

15th October 2023. Vol.101. No 19
© 2023 Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

MODIFICATION OF QUANTUM ALGORITHMS FOR QUDITS WITH AN EVEN NUMBER OF STATES IN ORDER TO OPTIMIZE AND REPLACE OBSOLETE ALGORITHMS

LARISSA CHERCKESOVA[1], ELENA REVYAKINA[2], OLGA SAFARYAN[3], KIRILL LYASHENKO[4]

[1,2,3,4]Department of Cyber Security of Information Systems, Don State Technical University, Rostov-on-Don, Russia

E-mail: elena.a.revyakina@gmail.com

ABSTRACT

In this article, a modification of Shor's algorithm for multiple 2n number systems of quantum computer was implemented. Work was carried out to study the number systems of quantum computers, the features of their use in quantum programming, as well as the simplification of computations by Shor's algorithm itself. The purpose of Shor's algorithm is to factorize any number in less time. The modification of Shor's algorithm developed in the course of the study makes it possible to simplify the calculations of the algorithm, to reduce the volume of circuits (schemes), at least to two digits of a number, which will allow getting rid of unnecessary calculations. This article is devoted to the operation of the algorithm in different number systems. Because working with real quantum computers is available to the narrow circle of researchers, the application of the modification in the emulator can affect the computational speed; in this case, there may be cases when the modification can work more accurately. The author's modification reduces the number of required qubits to 2, practically without reducing the performance of the algorithm and its execution time. Consequently, the cost of the quantum circuit itself also decreases several times (by 5–6 times, since not 12, but 2 qubits are required).

Keywords: *Digital Security, Information Security, Qubit, Factorization, Quantum Programming.*

1. INTRODUCTION

The basis of all computer security is modern cryptography, which includes various areas of mathematical knowledge, such as information theory, coding theory, number theory, probability theory and computational complexity theory. To date, cryptographic methods of information protection have become an integral part of a secure computer system, without which no area of human activity and society as a whole can do.

The fundamental concept of cryptography is encryption. Messages available over the communication channel are encrypted with confidentiality of information and decrypted by trusted users using complex mathematical algorithms.

Throughout the existence of man, scientists have come up with different approaches to encryption, from simple substitution and substitution ciphers to more complex algorithms used on the Internet, such as the RSA public key encryption algorithm. Modern cryptographic algorithms are based on the fact that some problems do not have algorithms that would solve it in less than exponential time from the size of the input. However, the emergence of stable quantum computers will make it possible to solve such problems in a fairly short time [1–5].

Classical methods of information protection approaches become obsolete over time, and in order to maintain the security of computer systems, one has to look for new or alternative approaches to cryptographic methods of information protection. New approaches are quantum and post-quantum cryptography, the rapid development of which is observed in world science. The phenomena of quantum physics, which give rise to the existence and development of a new approach to cryptography, called quantum, it is possible to design and implement a communication system that allows you to detect channel eavesdropping with high probability [6]. Fixing such eavesdropping is ensured by the fact that any attempt to measure interrelated parameters in a quantum communication channel introduces disturbances into the system,

while the signals that were transmitted inside the channel are destroyed or changed, which gives a signal to the participants in the communication about an interception attempt, and also provides an opportunity to fix the interceptor's activity [3, 7–9].

To date, the development of quantum computer cryptographic systems is one of the most promising and significant areas in the field of IT and science, forcing humanity to invent and create new more reliable and tamper-proof algorithms that will replace mathematical cryptography algorithms. In particular, post-quantum cryptography offers more secure information transfer systems based on hash functions. In recent years, it has been determined that algorithms already exist that have been able to survive quantum superiority by being cryptanalytic for classical methods of cryptanalysis [10–12].

But this is at this point in time, the question is how long can such algorithms be relevant and how long can such solutions survive quantum superiority? No one is able to answer this question with certainty and accuracy. Nevertheless, the improvement or complete processing of such algorithms may be a necessary transition bridge from mathematical cryptography algorithms, being a necessary measure to ensure digital security in a digital race in which the loser may incur both financial and reputational losses and problems of a state nature threatening the sovereignty of the state [13-23].

The creation of new algorithms that are more reliable and protected from hacking is necessary right now [24-28]. They are needed as the replacement for existing mathematical cryptographic algorithms, as well as to ensure the digital security in the digital race, in which the loser can bear both financial and reputational losses and the problems of state nature that threaten to the sovereignty of the state.

The improvement is possible in the reducing the size of the quantum algorithm circuit to 2 qudits, when in the standard version, Shor's algorithm uses 8 qudits. Using the special "parallel" version of Shor's algorithm, which requires no longer 8, but 7 qudits to decompose the number 15 into prime factors, it is possible to achieve the overall calculation accuracy of approximately 90%. The improvement is aimed at preserving such result using less number of schemes, thereby increasing the efficiency of calculations by 75% in the equal period of time, if we count from the standard execution of the Shor's algorithm.

According to the literature [29, 30], the standard algorithm, as disadvantage, has the increased size of scheme. The problem is that the number of qubits used for decomposition of numbers into the prime factors will be larger. The calculations will be less productive and efficient.

Therefore, in this study, a new modified cryptography algorithm has been developed that reduces the size of the quantum algorithm circuit to 2 qudits, which potentially leads to a 25%-75% increase in computational efficiency due to the reduction of the circuit and the use of fewer qubits for the same expansions of the same numbers, in comparison with the original Shor algorithm.

In this article, emulation of the Shor's algorithm for quants with even number of states was implemented. The purpose of this work was to study mathematical component and to develop the Shor's algorithm, based on qudits with even number of states.

2. MATHEMATICAL RATIONALE

Shor's algorithm is a polynomial-time quantum computer algorithm for integer factorization. This algorithm runs in polynomial time, the elapsed time is polynomial in log N of the size of the integer given as input. This algorithm uses a set of quantum gates called "Quantum Gates" of order $O((\log N)^2 (\log \log N) (\log \log \log N)$ using fast multiplicationto solve the integer factorization problem using quantum computer. This method factorization is the most efficient in comparison with classical factorization algorithms. Efficiency is due to the quantum Fourier transform and modular exponentiation using repeated squares.

It is known that the smallest, as well as the largest unit of information used in a quantum computer, is a qubit, which, like a classical bit, has two states: 1 and 0. Difference lies in the fact that a qubit is in superposition until the moment of its measurement and after measurement is already taking on the form of a classical bit. What state we get depending on the qubit amplitude. The classical notation for amplitudes in Dirac notation is as follows:

$$A|0> + B|1> \quad (1)$$

where A and B are amplitudes expressed as a complex number. The state of the qubit amplitudes is also commonly written as a vector. Since getting 0 or 1 during measurement is random, you need to know the probability of getting one of the states. The probability of obtaining a state depends entirely on the amplitude, and in the case of an amplitude of 0, the probability will be calculated as follows:

$$|A|^2 \quad (2)$$

In a quantum computer, the amplitude is changed using quantum gates, or gates, which are described by unitary square matrices over the space of complex numbers. In other words, a matrix is unitary if and only if there is a matrix inverse to it. The result of the transformation will be a vector obtained by multiplying a qubit vector by a matrix, of course, the dimension of the gate matrix must correspond to the dimension of the vector. For example, the Hadamard H-gate, which is described by the matrix:

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix} \quad (3)$$

and the qubit is described by the expression:

$$q = 1 \vee 0 > \begin{pmatrix}1\\ 0\end{pmatrix} \quad (4)$$

Together they will produce the following result:

$$Hq = H \vee 0 \geq \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}\begin{pmatrix}1\\ 0\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\ 1\end{pmatrix} = \frac{1}{\sqrt{2}}\left|0 > \frac{+1}{\sqrt{2}}\right| 1 > \quad (5)$$

The Hadamard gate is one of the most useful quantum gates. This gate is sometimes defined as the square root of the NOTgate. This is due to the fact that this gate transforms a|0›-part of the qubit into (|0›+|1›)/ √ 2–"half way" between |0› and |1› states in the geometric interpretation of the qubit on the Bloch sphere (Figure 1).

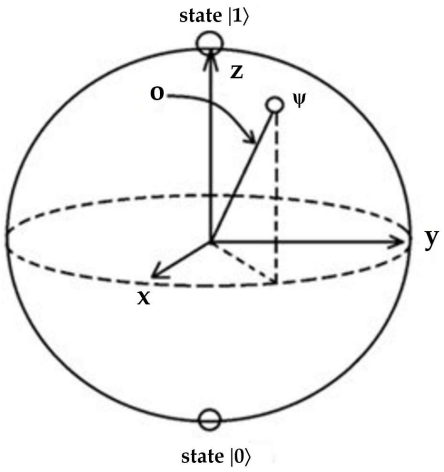

Figure 1: Bloch Sphere

To transfer the translation of one-qubit gates to qudits – (an extended version of qubits) with an even number of states, it is enough to use the product of the tensor product of the corresponding one-qubit gate itself to the appropriate size:

$$H^{\otimes n} \quad (6)$$

The productivity of quantum computers is based on the use of quantum phenomena: superposition, reversibility, and parallelism. According to the superposition principle, the state of quantum system is described by linear combination of wave functions (ψ-functions):

$$\psi = \sum_i C_i \psi_i \quad (7)$$

Shor's quantum algorithm runs in polynomial time (the elapsed time is polynomial in log N, where N is the size of the integer given in the input). It takes quantum gates of order of complexity $O((\log N)^2 (\log \log N) (\log \log \log N)$. The idea of its efficiency is given by the following rough estimate: the problem of factorization of an integer $M\sim$ 2800 cannot be solved in a reasonable time on a classical computer, while the application of the quantum algorithm at a clock frequency of 1 MHz would require a couple of days [29].The algorithm uses the reduction of the factorization problem to finding the period of the function like:

$$f(x) = a^x (mod\ M) \quad (8)$$

where a is chosen randomly. It can be shown that in most cases the period r is even and the number $ar/2\pm1$ has a common factor with N, which is found using the classical Euclidean algorithm. Shor's algorithm includes the detailed description of the efficient execution of the U_f operation. Finding the period $f(x)$ uses quantum modification of the fast Fourier transformation (whose role in the simpler Simon problem was played by the Hadamard transform H_n).

3. SHOR'S ALGORITHM

Shor's algorithm is a huge leap in the development of modern cryptography [30]. Peter Shor's algorithm was developed in 1994, and 7 years

later, in 2001, its efficiency and performance were demonstrated to IBM specialists. The result of the work of a group of IBM enthusiasts was the factorization of the number 15 into factors of 3 and 5, using quantum computer with 7 qubits [31]. Its notability and importance lie in the fact that the Shor's algorithm is potential threat to modern public key cryptosystems [32]. At thismoment, the most popular is the RSA public key algorithm, which can be cracked byfinding the *M* factors with which the encryption took place. With a sufficiently large number *M*, it will not be possible to break the algorithm by any of the most well-known modern classical algorithms. The only ones capable of doing this are the modern Pollard-Strassen algorithm and the algorithm of Shanks (quadratic form method), which require too much time [33].

As mentioned earlier, Shor's algorithm is based on finding the period of the function a^x *(mod N)*. On the Figure 2 the diagram (scheme) applicable to finding the period of a function is demonstrated.

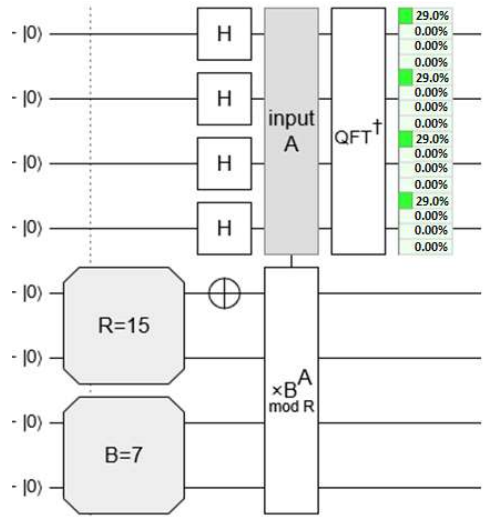

Figure 2: Implementation of Shor's Algorithm on the Quantum Emulator

The presented scheme is intended for the decomposition of the number 15 into the prime factors. You can see that the circuit is divided into two conditional parts, the first is for input, the second for function output. That is, the first 4 qubits are allocated for the input data, and the second 4 qubits are allocated for the output of the result [34-37].

The first step in the algorithm is to apply *N* Hadamard gates to the upper case, after which we get an equiprobable superposition of all the upper case boolean states:

$$x,0> \quad \frac{1}{\sqrt{N}}\sum_{x=0}^{N-1} \tag{9}$$

After that, the unitary transformation is applied, which translates the state of the lowercase (lower register) to function (1), the upper case remains unchanged:

$$x,0> \underset{\rightarrow}{U_f} \frac{1}{\sqrt{N}}\sum_{x=0}^{N-1} x,f(x)> \quad \frac{1}{\sqrt{N}}\sum_{x=0}^{N-1} \tag{10}$$

The last step uses the quantum Fourier transformation:

$$\sum_{k=0}^{N-1} e^{\frac{2\pi ikx}{N}} \vee k,f(x)> \quad \frac{1}{N}\sum_{x=0}^{N-1} \tag{11}$$

The result is *k*, t^x mod *M*> with probability of:

$$\left|\frac{1}{N}\sum_{x:t^x\equiv t^k modM} e^{\frac{2\pi ikx}{N}}\right|^2 \tag{12}$$

On the Figure 3 the section of code responsible for the calculations performed by the Fourier gate is demonstrated.

```
def generate_QFT(n):
    N = 2**n
    W = n_root_of_1(n)
    matrix = [[W**(i*j) for j in range(N)] for i in range(N)]
    return np.matrix(matrix) * 1/np.sqrt(N)
```

Figure 3: Calculation of the Fourier Gate

Let's consider the implementation of Shor's factorization algorithm with following parameter values:

p = 3, q = 5, N = pq = 15,
Φ(N) = (5 − 1)(3 − 1) = 8,
a = 7, f(x)=7^x mod 15.

We represent the function *f(x)* : $\{0,1\}^4 \rightarrow \{0,1\}^4$ in the following form: f(x)=7^x mod 15.Since 4 is the order of 7 in $Z_{(15)}$ and 7^4 = 1(15),

$$(7^2)^{x1} \cdot (7^1)^{x0} \; mod \; 15 = 4^{x1} \cdot 7^{x0} \; mod \; 15 \tag{13}$$

Journal of Theoretical and Applied Information Technology

15th October 2023. Vol.101. No 19
© 2023 Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Schematic implementation of Shor's algorithm for selected values [38] on quantum emulator is demonstrated in the Figure 4.

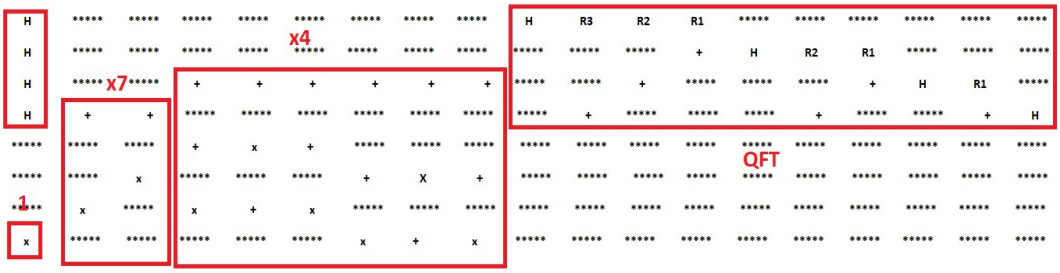

Figure 4: Implementation of the Shor Algorithm in the Emulator for $f(x) = 7^x \bmod 15$

Thus, blocks H_n and QFT in the Figure 1 to us are known.

Now it is important to understand the construction of the blocks labeled 1, ×7 and ×4. Block 1 is the *X* operator on the lowest qubit of the |y> register.

The result of the operation of this block in |y> is the value 1, which, in accordance with equation (1), will need to be multiplied by 7, if the bit x_0 = 1, and multiplied by 4, if the bit x_1 = 1. In the Figure 4 the multiplication of |y> containing 1, by 7 is demonstrated, there is dependence x_0, which is the setting of the remaining bits of the number 7 in the register by the CNOT(x_0) operators.

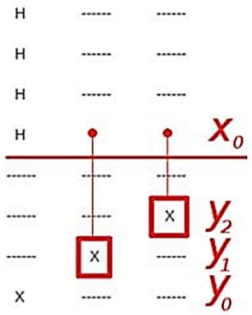

Figure 5: Multiply |y>, Containing 1, by 7

On Figure 6 shows the multiplication of the register |y> by 4 depending on bit x_1.

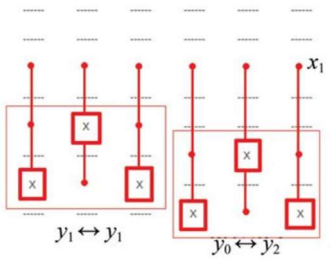

Figure 6: Shor's Algorithm. Operation ×4

Multiplying a number by 4 is a cyclic shift by 2, which is implemented a little more complicated than multiplying |y> by 7. It does this by replacing the places of the bits – (y_0 y_2) and (y_1 y_3). In this case, all replacements are carried out by three CNOT operators, controlled also by bit x_1. After these operations and the quantum Fourier transform in the |x> register, the value becomes 0 × 0100 = 4:

$y/2^n = 4/2^4 = 1/4$

The fraction 1/4 itself is suitable as a candidate for the role of *k* and *r* due to the small denominator. Substituting the value 4 into the *f(x)* function, we get $f(4) = 7^4(15) = 1$.

After we have found the period r, it is necessary to find the values of p and q.

$$GCD\ (7^{4/2} + 1,\ 15) = GCD\ (50,\ 15) = 5 \qquad (14)$$

$$GCD\ (7^{4/2} - 1,\ 15) = GCD\ (48,\ 15) = 3 \qquad (15)$$

The answer is multipliers 3 and 5.

Factorization of a number by the Shor's algorithm consists of 2 stages: finding the factorization period and calculating the factors of the number. The first step is to find the factorization period, which can be done using the following function:

$$(7^2)^{x_1} * (7^1)^{x_0}(15) = 7^{x_0}(15) \qquad (16)$$

where 7 is the order of the number.

On the Figure 7 you can see the result of executing the circuit from Figure 2.

© 2023 Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

```
12
[0.24999999999999983, 6.472967989715876e-33, 1.3099158326042607e-32, 5.933552368868572e-32]
[0.24999999999999983, 1.5901248778818532e-31, 1.1449936989803049e-31, 3.54924136365606e-31]
[0.24999999999999983, 3.944652644918579e-31, 1.0367191190674882e-31, 4.388907585280776e-31]
[0.24999999999999983, 7.5431538223153235e-31, 9.254931637115248e-31, 9.457647833236815e-31]
(5, 3)
```

Figure 7: The Result of Finding the Factorization Period for the Original Shor's Algorithm

The first line of the output prints the resulting number when measured, and the subsequent lines show the probability of obtaining the state when measured.

After obtaining the period of the function, you can get the decomposition of the number by finding the greatest common divisor:

$$p = gcd(a^{r/2} + 1, N) \quad (17)$$

$$q = gcd(a^{r/2} - 1, N) \quad (18)$$

Note that not every r can lead to the desired result. So, for r = 8, or for r = 0, whichis quite a possible result, we get p = 15 and q = 1.

In this research work, it was experimentally found that the desired results of the factorization algorithm can be obtained with such good values *y*, thanks to which, as a result of finding the factorization period, favorable r is obtained [29–41].

The main idea of the implemented Shor's algorithm, in the framework of the study,is to reduce the size of the quantum algorithm circuit (scheme) by changing the basic representation of values in the form of binary notation of number in the number of different number system. It makes it possible to reduce the size of the circuit (scheme) to 2 qudits. In the implemented one, the number of state vectors is reduced to two, which significantly reduces the "width" of the scheme. In this case, only the first vector of states is measured.

Below is an implementation of Shor's algorithm on vectors consisting of 16 possible states, so you can decompose a number using a scheme consisting of only two vectors. On the Figure 8 shows the result of the modification. The result, as a whole, differs little from the usual Shor's algorithm only that after measuring the first state vector the number 4 was obtained.

```
4
[0.24999999999999983, 6.472967989715876e-33, 1.3099158326042607e-32, 5.933552368868572e-32]
[0.24999999999999983, 1.5901248778818532e-31, 1.1449936989803049e-31, 3.54924136365606e-31]
[0.24999999999999983, 3.944652644918579e-31, 1.0367191190674882e-31, 4.388907585280776e-31]
[0.24999999999999983, 7.5431538223153235e-31, 9.254931637115248e-31, 9.457647833236815e-31]
(5, 3)
```

Figure. 8: The result of Finding the Factorization Period of the Shor's Algorithm Modification

Before calculating the factorization period, it is necessary to check for the existence of the factorization period. On Figure 9 the check for the existence of the factorization period for 8 qubits is demonstrated.

```
>>> 7**8//15
384320
>>> 7**8 - 15*_
1
```

Figure 9: Checking for the Existence of the Factorization Period of the Classical Shor's Algorithm

Let us check the modification of the Shor's algorithm. Figure 10 demonstrates the existence of the modification of the factorization period.

```
>>> 7**12//15
922752480
>>> 7**12 - 15*_
1
```

Figure 10: Checking for the Existence of a Modification of the Shor's Algorithm

After the factorization period has been calculated, the factorization of the number takes place. Figure 11 shows the result of factoring the number 15.

```
>>> gcd(7**6 + 1,15)
5
>>> gcd(7**6 - 1,15)
3
```

Figure 11: Finding the Multipliers

The parallel encryption scheme is new implementation of so-called Shor's algorithm, the first quantum method in the history of science to factorize numbers, invented by programmer Peter Shor specifically for quantum computers in 1994 [42].

Using a special "parallel" version of Shor's algorithm that requires 7 qubits instead of 8 qubits to factorize the number 15 into prime factors. It was invented by Russian physicist Alexei Kitaev, who today works at the California Institute of Technology and is on the advisory board of the Russian Quantum Center. It is worth noting the development of the group of scientists at the Massachusetts University, who modified the Shor's algorithm, and experimentally proved the correctness of the algorithm by factoring the number 15 by 5 and 3 with a total calculation accuracy of 99%. Such accuracy indicates that this modification of Shor's algorithm decomposes numbers [43–45].

4. RESULTS AND DISCUSSION

4.1 Implementation of Shor's Algorithm by A. Kitaev's Method

The difference between the classic implementation of Shor's algorithm and Kitaev's algorithm is that Kitaev's algorithm uses different method of qubit binding.

For example, the scheme:

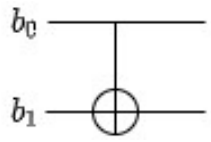

Figure 12: Example of Linked Qubits

can be converted as:

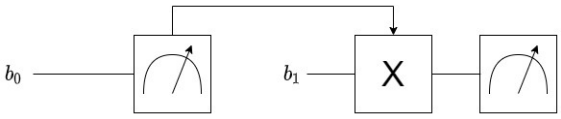

Figure 13: Related Qubits in Kitaev's Algorithm

In this case, gate *X* will be applied to qubit b_1 in the event that after the measurement on qubit b_0, one is obtained. This circuit must be executed sequentially and, in contrast to the usual CNOT gate, the property of quantum connectivity disappears. However, using this approach, the number of used qubits can be reduced. For example, for the factorization algorithm for the number 15 onthe basis 7, the scheme will look like in Figure 14.

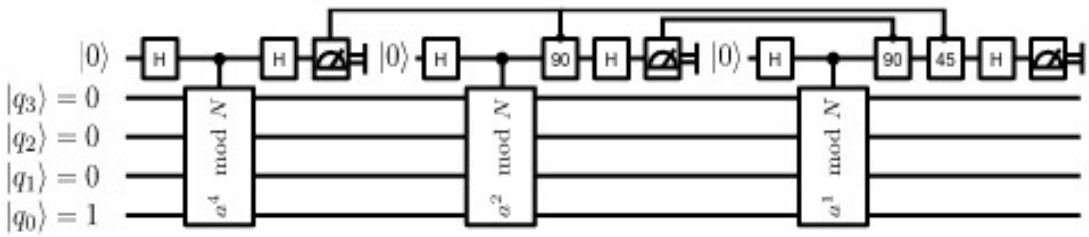

Figure 14: Shor's Algorithm Implemented by the Kitaev's Method

It can be noted that the number of qubits in this scheme has decreased to 7, instead of 8 in the classical implementation of the Shor's algorithm, while the state of the qubits remains the same, as well as the overall accuracy of calculations. At the same time, this scheme is performed sequentially, unlike the usual scheme. One of the authors of this algorithm, Isaac Chuang, stated that to factorize the number 15 — the smallest odd composite number that is not representable as power of prime number (limitation of Shor's algorithm) — traditionally requires 8–12 qubits, while their quantum computer requires only 5 to 7 qubits.

Experimentally, five 40Ca+ ions were was used in the superposition state and enclosed in the quadrupole ion trap or in the Pol's trap. Quantum computer uses laser pulses as logic switches, where 4 atoms are used to perform the operation, and one atom is used for extraction and interpretation of the data.

According to the results of the experiments, the probability of error in the calculating of the period was less than one percent.

However, the researchers themselves pointed out in their work that, in order to get such level of probability actually, the experiment should be repeated 8 (eight) times. Scientists estimated the probability of obtaining the reliable period from the first time at about 50%. It is worth noting that few years ago, American physicists from the University of Santa Barbara were able to implement Shor's quantum algorithm on the system with three qubits. The

algorithm gave the correct answer about in the 48 percent of the time (cases).

4.2 Shor's Algorithm on Qudits with Even Number of States

It was previously indicated that the Shor's algorithm is divided schematically into two conditional parts, equal in the number of used qubits. The first part is intended for the inputs of the exponential function, the other is used to output the result of the function (8).Translation of calculations into even-degree qudits for the Shor's algorithm will allow to reduce the number of used quanta in the quantum processor to two, while not changing the complexity of the calculations. At the same time, single gates used in computing systems on qubits are easily transferred to new states using the tensor product of the gate itself. In this case, the scheme of the Shor's algorithm will look like this (Figure 15):

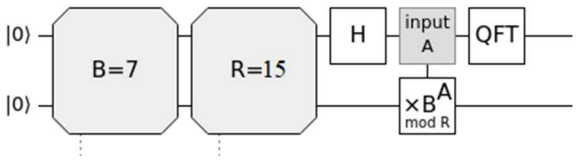

Figure 15: Shor's Algorithm in Qudits of Even Degree

This implementation is able to reduce the number of quanta used to calculate the Shor's algorithm in quantum processors, with reduced quanta noise. The operation of this algorithm is demonstrated in the created emulator.

4.3 System Requirements to Run the Algorithm

The modification of the Shor's algorithm developed during the study can be used on any modern digital computer using the Python3.8+ interpreter. The program should be launched in the emulator, or in the terminal of the computer, or in the development environment.

RAM:8Gb.

GPU Memory:2Gb.

Operating system:Windows 10 ×32, ×64;Linux: Ubuntu18+, Debian.

4.4 Evaluation of the Work of Algorithms:

On Figure 16 the memory consumption required to implement the Shor's algorithm emulator is demonstrated.

Line #	Mem usage	Incriment	Occurences
52	49.781 MiB	49.781 MiB	1
53			
54	51.160 MiB	1.379 MiB	1
55	51.160 MiB	0.000 MiB	1
56	51.160 MiB	0.000 MiB	1
57	51.160 MiB	0.000 MiB	13
58	51.160 MiB	0.000 MiB	13
59	51.160 MiB	0.000 MiB	13
60	51.160 MiB	0.000 MiB	1

Figure 16: Memory Consumption of the Classic Shor's Algorithm

As a result of the measurement, it can be seen that the entire algorithm occupies 49.781MiB of memory. Figure 17 shows the memory consumption for the implemented emulation of a modified version of Shor's algorithm.

Line #	Mem usage	Incriment	Occurences
52	49.816 MiB	49.816 MiB	1
53			
54	51.203 MiB	1.387 MiB	1
55	51.203 MiB	0.000 MiB	1
56	51.203 MiB	0.000 MiB	1
57	51.203 MiB	0.000 MiB	1
58	51.203 MiB	0.000 MiB	1
59	51.203 MiB	0.000 MiB	1
60	51.203 MiB	0.000 MiB	1

Figure 17: Memory Consumption of the Shor's Algorithm on Qudits

As a result of the measurement, it can be seen that the entire algorithm occupies 49.816MiB of memory. Figure 18 shows graphs of the time it takes for the algorithms to perform factorization. On them, the indicators of time for modification are visualized in orange color, and the indicators for the classical algorithm are displayed in blue color.

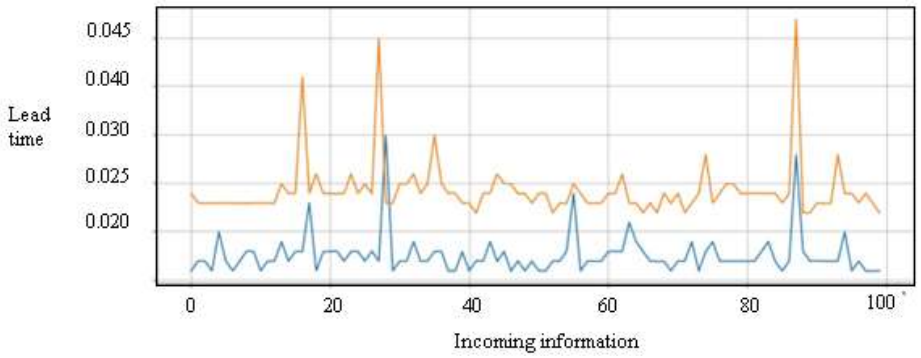

Figure 18: Graph of Indicators of the Time of Execution of the Algorithms

Table 1. Comparison of the Speed of Calculating the Factorization Period

Measured value / The algorithm	Classic Shor's algorithm	Shor's algorithm in Kitaev's implementation	Modification for multiples of 2m qubits
Average execution time	0.022440991401672362	0.0432834607672362	0.028712222576141356
Number of quants used	8	7	2
RAM consumption	49.781MiB	49.77MiB	49.816MiB

The novelty of the research work done is to modify the Shor's algorithm in such a way that this method will reduce the number of quantum processor quanta used by changing the basic representation of values in the form of binary notation of number into a number of another number system. It makes possible to reduce the size of the circuit (scheme) from eight to two, while not greatly changing the complexity of calculations.

Experimentally, it was found that the desired results of the factorization algorithm can be obtained with such good values *y*, thanks to which, as a result of finding the factorization period, favorable values *r* [9] are obtained. As a result of measurements, it can be seen that the entire algorithm occupies 49.816 MiB of memory, being an acceptable value.

5. CONCLUSION

In this article, a new modified cryptography algorithm has been developed that reduces the size of the quantum algorithm circuit to 2 qudits, which potentially leads to a 25%-75% increase in computational efficiency due to the reduction of the circuit and the use of fewer qubits for the same expansions of the same numbers, in comparison with the original Shor algorithm. The problem of integer factorization is extremely important since quite a few systems use the RSA public key. In the course of research, the classical Shor algorithm was implemented for an even number of qubits.

The modification of Shor's algorithm developed during the research can be used on any modern computer using the Python3.8+ interpreter. According to the graph of indicators of the time it takes to perform factorization, the dynamics of the execution time of the modification of the algorithm is clearly visible, in comparison with the original. On the graph, the time indicators for modification are visualized in orange color, and the indicators for the classical algorithm are visualized in blue color. Containing a smaller volume of quantum circuits (schemes), at least one vector of states is required for input data, while the algorithm is not inferior in the speed of finding the factorization period, and the accuracy of calculations does not exceed the value of the value by the classical Shor's algorithm, as well as the scheme proposed by Isaac Chuang. In this case, the classic Shor's algorithm uses $2M$ qubits, where $M = \log_2 N$. The modification developed in the course of the research for multiple of two qubits is oriented at demonstrating the operation of the algorithm in other number systems and showing the decent result. Further research should be aimed at studying the application of the new modified Shor's algorithm.

REFERENCES:

[1] K. Head-Marsden, J. Flick, C. J. Ciccarino, and P. Narang, "Quantum information and algorithms for correlated quantum matter", *Chemical Reviews*, Vol. 121, No. 5, 2020, pp. 3061-3120. https://doi.org/10.1021/acs.chemrev.0c00620

[2] B. Wang, F. Hu, H. Yao, and C. Wang, "Prime factorization algorithm based on parameter optimization of using model", *Scientific Reports*, Vol. 10, 2020, Art. No. 7106. https://doi.org/10.1038/s41598-020-62802-5

[3] B. Yan, H. Jiang, M. Gao, Q. Duan, H. Wang, and Ma, Z., "Adiabatic quantum algorithm for factorization with growing minimum energy gap", *Quantum Engineering*, Vol. 3, 2021, Art. No. e59. https://doi.org/10.1002/que2.59

[4] D.G. Feng, and W.F. Lian, "Security problems in the cyberspace and countermeasures", *Bulletin of Chinese Academy of Sciences*, Vol. 36, 2021, pp. 1239-1245.

[5] S. Gidney, and M. Ekera, "How to factor the 2048-bit RSA integers in 8 hours using 20 million noisy qubits", *Quantum*, Vol. 5, 2021, Art. No. 433. https://doi.org/10.22331/q-2021-04-15-433

[6] Z.G. Wang, S.J. Wei, and G.L. Long, "A quantum circuit design of AES requiring fewer quantum qubits and gate operations", *Frontiers of Physics*, Vol. 17, 2021, Art. No. 41501. https://doi.org/10.1007/s11467-021-1141-2

[7] K. Luo, W. Huang, Z. Tao, L. Zhang, Y. Zhou, J. Chu, W. Liu, B. Wang, J. Cui, S. Liu, F. Yan, M.H. Yung, Y. Chen, T. Yan, and D. Yu, "Experimental realization of two qutrits gate with tunable coupling in superconducting circuits", *Physical Review Letters*, Vol. 130, 2023, Art. No. 030603. https://doi.org/10.1103/PhysRevLett.130.030603

[8] F. Boudot, P. Gaudry, A. Guillevic, N. Heninger, E. Thomé, and P. Zimmermann, "Comparing the difficulty of factorization and discrete logarithm: A 240-digit experiment", in D. Micciancio, and T. Ristenpart (Eds.), *Advances in cryptology – CRYPTO 2020. CRYPTO 2020. Lecture notes in computer science*, Vol. 12171, pp. 62–91. Springer, Cham, 2020. https://doi.org/10.1007/978-3-030-56880-1_3

[9] K. Kim, and B. Dandurand, "Scalable branching on dual decomposition of stochastic mixed-integer programming problems", *Mathematical Programming Computation,* Vol. 14, No. 1, 2022, pp. 1–41. https://doi.org/10.1007/s12532-021-00212-y

[10] NIST Announces First Four Quantum-Resistant Cryptographic Algorithms. 2022. [Online]. Available: https://www.nist.gov/news-events/news/2022/07/nist-announces-first-four-quantum-resistant-cryptographic-algorithms

[11] J. Barzen, and F. Leymann, "Continued fractions and probability estimations in Shor's algorithm: A detailed and self-contained treatise", *AppliedMath*, Vol. 2, 2022, pp. 393-432. https://doi.org/10.3390/appliedmath2030023

[12] A. Lyapin, A. Beskopylny, and B. Meskhi, "Structural monitoring of underground structures in multi-layer media by dynamic methods", *Sensors*, Vol. 20, No. 18, 2020, Art. No. 5241. https://doi.org/10.3390/s20185241

[13] F. Dengguo, and L. Yifeng, "Challenges and countermeasures of cyberspace security", *Chinese Academy of Sciences*, Vol. 36, No. 10, 2021, pp. 1239-1245. https://doi.org/10.16418/j.issn.1000-3045.20210813004

[14] S. Ghosh, M. Zaman, and S. Sampalli, "Quantum-safe asymmetric cryptosystems: Current solutions and future directions against quantum attacks", in *Holistic approach to quantum cryptography in cyber security*, pp. 99-120. CRC Press, Boca Raton, 2023.

[15] R. Kuang, "Generalized uncertainty principles for quantum cryptography", 2023, arXiv:2302.01026. https://doi.org/10.48550/arXiv.2302.01026

[16] G. Dai, K. He, C. Zhao, Y. He, J. Liu, and W. Chen, "Quasi-Shor algorithms for global benchmarking of universal quantum processors", *Applied Sciences*, Vol. 13, No. 1, 2023, Art. No. 139. https://doi.org/10.3390/app13010139

[17] M.R. Habibi, S. Golestan, A. Soltanmanesh, J. M. Guerrero, and J.C. Vasquez, "Power and energy applications based on quantum computing: The possible potentials of grover's algorithm", *Electronics*, Vol. 11, No. 18, 2022, Art. No. 2919. https://doi.org/10.3390/electronics11182919

[18] C.-W. Yang, H.-W. Wang, j. Lin, and C.-W. Tsai, "Semi-quantum identification without information leakage", *Mathematics*, Vol. 11, No. 2, 2023, Art. No. 452. https://doi.org/10.3390/math11020452

[19] X. Song, G. Chen, and A.A. Abd El-Latif, "Quantum color image encryption scheme based on geometric transformation and intensity channel diffusion", *Mathematics*, Vol. 10, No. 17, 2022, Art. No. 3038. https://doi.org/10.3390/math10173038

[20] M.Á. González de la Torre, L. Hernández Encinas, and A. Queiruga-Dios, "Analysis of the FO transformation in the lattice-based post-quantum algorithms", *Mathematics*, Vol. 10, No. 16, 2022, Art. No. 2967. https://doi.org/10.3390/math10162967

[21] L.V. Cherckesova, O.A. Safaryan, N.G. Lyashenko, and D.A. Korochentsev, "Developing a new collision-resistant hashing algorithm", *Mathematics*, Vol. 10, No. 15, 2022,

Art. No. 2769. https://doi.org/10.3390/math10152769

[22] A. Manzano, D. Musso, Á. Leitao, A. Gómez, C. Vázquez, G. Ordóñez, and M.R. Nogueiras, "A modular framework for generic quantum algorithms", *Mathematics*, Vol. 10, No. 5, 2022, Art. No. 785. https://doi.org/10.3390/math10050785

[23] A. Beskopylny, A. Lyapin, H. Anysz, B. Meskhi, A. Veremeenko, and A. Mozgovoy, "Artificial neural networks in classification of steel grades based on non-destructive tests", *Materials*, Vol. 13, No. 11, 2020, Art. No. 2445. https://doi.org/10.3390/ma13112445

[24] L.V. Cherckesova, O.A. Safaryan, A.N. Beskopylny, and E. Revyakina, "Development of quantum protocol modification CSLOE–2022, increasing the cryptographic strength of classical quantum protocol BB84", *Electronics*, Vol. 11, No. 23, 2022, Art. No. 3954. https://doi.org/10.3390/electronics11233954

[25] O.A. Safaryan, K.S. Lemeshko, A.N. Beskopylny, L.V. Cherckesova, and D.A. Korochentsev, "Mathematical analysis of parametric characteristics of the consensus algorithms operation with the choice of the most priority one for implementation in the financial sphere", *Electronics*, Vol. 10, 2021, Art. No. 2659. https://doi.org/10.3390/electronics10212659

[26] A. Mironenko, P. Razumov, V. Kravchenko, I. Pilipenko, L. Cherckesova, and O. Safaryan, "Modification of R. Merkle's post-quantum algorithm", in A. Beskopylny, and M. Shamtsyan, (Eds.), *XIV International Scientific Conference "INTERAGROMASH 2021". Lecture notes in networks and systems*, Vol. 246, pp. 184-191. Springer, Cham, 2022. https://doi.org/10.1007/978-3-030-81619-3_20

[27] N. Lyashenko, K. Rysyatova, L. Chemerigina, P. Razumov, V. Kravchenko, L. Cherckesova, and O. Safaryan, "Post-quantum encryption scheme with supersingular isogenies" in A. Beskopylny, and M. Shamtsyan, (Eds.), *XIV International Scientific Conference "INTERAGROMASH 2021". Lecture notes in networks and systems*, Vol. 246, pp. 164-172. Springer, Cham, 2022. https://doi.org/10.1007/978-3-030-81619-3_18

[28] F.M. Reza, *An introduction to information theory*. Dover, New York, NY, 1994.

[29] Y. Cao, J. Romero, J. P. Olson, M. Degroote, P. D. Johnson, M. Kieferová, I.D. Kivlichan, T. Menke, B. Peropadre, N.P.D. Sawaya, S. Sim, L. Veis, and A. Aspuru-Guzik, "Quantum chemistry in the age of quantum computing", *Chemical Reviews*, Vol. 119, No. 19, 2019, pp. 10856-10915. https://doi.org/10.1021/acs.chemrev.8b00803

[30] S. Holevo, *Quantum systems, channels, information: A mathematical introduction*. Walter de Gruyter GmbH & Co KG, Berlin; Boston, MA, 2019.

[31] Yu.L. Sagalovich, *Introduction to algebraic codes*. IPPI RAN, Moscow, 2014, 310 p.

[32] R.R. Hamming, *Art of doing science and engineering: Learning to learn*. CRC Press, London, 1997. https://doi.org/10.1201/9781482283198

[33] J. Preskill, *Lecture Notes for "Physics 219/Computer Science. 2019. Quantum Computation" (Formerly Physics 229)*. [Online]. Available: http://theory.caltech.edu/~preskill/ph229/

[34] B. Valiron, "A functional programming language for quantum computation with classical control". Master's thesis, University of Ottawa, Ottawa, 2004. http://dx.doi.org/10.20381/ruor-18372

[35] M.V. Lezhinsky, and A.V. Mokryakov, "Encryption algorithms resistant to hacking in the condition of quantum superiority", in *Collection of scientific papers of the Department of Applied Mathematics and Programming based on the results of the permanent seminar "System Theory"*, pp. 141-147. Kosygin Russian State University (Technology. Design. Art), Moscow, 2021.

[36] J. Zhang, G. Pagano, P. Hess, A. Kyprianidis, P. Becker, H. Kaplan, A.V. Gorshkov, Z.-X. Gong, and C. Monroe, "Observation of a many-body dynamical phase transition with a 53-qubit quantum simulator", *Nature*, Vol. 551, 2017, pp. 601–604. https://doi.org/10.1038/nature24654

[37] National Academies of Sciences, Engineering, and Medicine, *Quantum computing: Progress and prospects*. The National Academies Press, Washington, DC, 2018. https://doi.org/10.17226/25196

[38] T. Monz, D. Nigg, E. A. Martinez, M. F. Brandl, P. Schindler, R. Rines, S.X. Wang, I.L. Chuang, and R. Blatt, "Realization of a scalable Shor algorithm", *Science*, Vol. 351, No. 6277, 2016, pp. 1068-1070. https://doi.org/10.1126/science.aad9480

[39] F. Blake, and R.C. Mullin, *An introduction to algebraic and combinatorial coding theory*. Academic Press, 2014.

© 2023 Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[40] D. Petz, *Quantum information theory and quantum statistics*. Springer Science & Business Media, Berlin; Heidelberg, 2007.

[41] J. Gabriel, B.K. Alese, A.O. Adetunmbi, O.S. Adewale, and O.A. Sarumi, "Post-quantum crystography system for secure electronic voting", *Open Computer Science*, Vol. 9, No. 1, 2019, pp. 292-298. https://doi.org/10.1515/comp-2019-0018

[42] Pljonkin, K. Rumyantsev, and P. Kumar Singh, "Synchronization in quantum key distribution systems", *Cryptography*, Vol. 1, No. 3, 2017, Art. No. 18. https://doi.org/10.3390/cryptography1030018

[43] K. Rumyantsev, and E. Rudinsky, "Parameters of the two-stage synchronization algorithm for the quantum key distribution system", in *Proceedings of the 10th International conference on security of information and networks*, pp. 140-147. Association for Computing Machinery, New York, NY, 2017. https://doi.org/10.1145/3136825.3136888

[44] Nechvatal, E. Barker, L. Bassham, W. Burr, M. Dworkin, J. Foti, and E. Roback., "Report on the development of the Advanced Encryption Standard (AES)", *Journal of research of the National Institute of Standards and Technology*, Vol. 106, No. 3, 2001, pp. 511-577. https://doi.org/10.6028/jres.106.023

[45] K. Babenko, E. A. Ischukova., E. A. Maro, I. D. Sidorov, and P. P. Kravchenko, "Development of cryptographic methods and means of information security", *Izvestiya SFedU. Technical Science*, Vol. 4, No. 129, 2012, pp. 40-50.