Quantum algorithms for typical hard problems: a perspective of cryptanalysis

Jingwen Suo[1] · Licheng Wang[1] · Sijia Yang[1] · Wenjie Zheng[1] · Jiankang Zhang[2]

Received: 27 January 2020 / Accepted: 7 April 2020

Abstract
In typical well-known cryptosystem, the hardness of classical problems plays a fundamental role in ensuring its security. While, with the booming of quantum computation, some classical hard problems tend to be vulnerable when confronted with the already-known quantum attacks, as a result, it is necessary to develop the post-quantum cryptosystem to resist the quantum attacks. With the purpose to bridge the two disciplines, it is significant to summarize known quantum algorithms and their threats toward these cryptographic intractable problems from a perspective of cryptanalysis. In this paper, we discussed the designing methodology, algorithm framework and latest progress of the mathematic hard problems on which the typical cryptosystems depend, including integer factorization problem, discrete logarithmic problem and its variants, lattice problem, dihedral hidden subgroup problems and extrapolated dihedral coset problem. It illustrated the reason why some cryptosystems such as RSA and ECC are not resistant to quantum attacks, yet some of them like lattice cryptosystems remain intact facing quantum attacks.

Keywords Quantum algorithms · Cryptanalysis · Lattice problem · Dihedral hidden subgroup problem

✉ Licheng Wang
wanglc2012@126.com

✉ Jiankang Zhang
jz09v@ecs.soton.ac.uk

[1] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

[2] Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK

1 Introduction

The public-key cryptosystems, including RSA, ElGamal, ECC and the related variants, play an ingredient role in securing the confidential communication over the Internet during the past decades. The fundamental principle of designing a secure public-key cryptosystem is to lay its security on the difficulty of certain mathematical problems. For instances, RSA [1] builds up its security on the hardness of integer factorization problem (IFP), and the security of ElGamal and ECC [2] is based on the difficulty of solving the discrete logarithm problem (DLP) and the DLP over elliptic curves (ECDLP), respectively. Even with the system parameters well optimized, the *classical* algorithms ever known, such as those toward IFP, DLP and ECDLP, are no longer efficient to our problem, as the resource required would grow in a sub-exponential manner over the scale of the problem.

Quantum computing is an interdisciplinary subject between quantum mechanics and computer science. Shor's algorithm [3] and Grover's quantum search algorithm [4] are the two most widely used quantum algorithms at present. Shor's algorithm is applied to solve large integer factorization problem and discrete logarithm problem. Grover's quantum search algorithm is adopted to search a number of specific targets in a disordered database. Both of them are of great significance in the perspective of cryptanalysis. Shor's quantum algorithm manifests a serious threat toward the security of RSA, ElGamal and ECC, since both the IFP problem and the DLP problem (including the ECDLP problem) can be solved efficiently with Shor's quantum algorithms. Grover's quantum algorithm is also used to speed up the task of collision finding. Therefore, to secure confidential communication in the so-called post-quantum era, some new public-key cryptosystems, which aim at resisting known quantum algorithmic attacks, appear on the stage of the modern cryptography. NIST launched the competition on post-quantum cryptography in 2016, and 26 outstanding designs have been selected for the second round evaluation so far.

As one of the most well-developed branches of post-quantum cryptography, lattice cryptography enjoys a high implementation efficiency and strong security reductions. In particular, Regev built the connection between the hardness of lattice problems and the hardness of the dihedral subgroup problem in 2002 [5]. However, at present, our confidence toward lattice cryptography is based merely on a heuristic reduction from the hardness of certain lattice problem to the hardness of certain quantum difficult problem, while the reverse reduction required by the logic framework of provable security is still open. Recently, Wen et al. made the first breakthrough toward building such kind of reverse reduction. Therefore, from the perspective of cryptanalysis, it is interesting to made a survey on quantum algorithms for classical hard problems, including lattice problems, IFP, DLP, ECDLP, as well as other related variants.

The rest of the paper is organized as follows: In Sect. 2, we reviewed quantum Fourier transform, for understanding quantum algorithms mentioned in this survey. The background for basis of qubit, quantum gates and quantum circuits is not involved in this work, since we believe it can be found in other textbooks on quantum computations, such as [6,7]. In Sect. 3, we summarized the quantum algorithms for period findings, which helps to understand why some symmetric cipher, such as RC6, tends to be insecure in the post-quantum era. Quantum algorithms for factoring integers, including Shor's algorithm based on quantum circuit model and Jiang's algorithm based on quantum adiabatic model, are summarized in Sect. 4. This is a key to understand why public-key cryptosystems based on difficulty of IFP are no longer secure. In Sect. 5, we explored the quantum algorithms for the DLP problem and their variants over elliptic curves, matrices of group rings, etc. This tell us why ElGamal, ECC as well the related variants are secure when large-scale quantum computers are available. Then, in Sects. 6 and 7, we introduced quantum algorithms for the hidden subgroup problem and the hidden shift problems, respectively, as two common frameworks of designs quantum algorithm. In Sect. 8, we presented quantum algorithms for the dihedral subgroup problem and its relation with lattice problems, in order to understand the potential of lattice cryptography in resisting known quantum attacks. Finally, we conclude the paper in Sect. 9.

2 Quantum Fourier transform

The quantum Fourier transform (QFT), with exponential speedup compared to the classical fast Fourier transform, has played an important role in quantum computation as a vital part of many quantum algorithms [8]. The QFT over $\mathbb{Z}_N$, the group of integers modulo N under addition, is a unitary operator $F_{\mathbb{Z}_N}$ that effects on a basis state as follows:

$$|x\rangle \longmapsto \frac{1}{\sqrt{N}} \sum_{y \in \mathbb{Z}_N} \omega_N^{xy} |y\rangle, \quad \forall x \in \mathbb{Z}_N$$

where $\omega_N := e^{2\pi i/N}$ denotes a primitive Nth root of the unity. Its matrix representation is

$$F_{\mathbb{Z}_N} = \frac{1}{\sqrt{N}} \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \omega_N & \omega_N^2 & \dots & \omega_N^{N-1} \\ 1 & \omega_N^2 & \omega_N^4 & \dots & \omega_N^{2N-2} \\ \vdots & \vdots & \ddots & & \vdots \\ 1 & \omega_N^{N-1} & \omega_N^{2N-2} & \dots & \omega_N^{(N-1)(N-1)} \end{pmatrix}$$

More succinctly, it is denoted by

$$F_{\mathbb{Z}_N} = \frac{1}{\sqrt{N}} \sum_{x,y \in \mathbb{Z}_N} \omega_N^{xy} |y\rangle\langle x|.$$

Further, we can derive QFT over any finite abelian group G. We know that any finite abelian group G can be expressed as a direct product of cyclic subgroups of prime power orders $G \cong \mathbb{Z}_{p_1} \times \cdots \times \mathbb{Z}_{p_r}$. Thus, in this case the QFT over G is the quantum operator $F_G = F_{p_1} \otimes \cdots \otimes F_{p_r}$.

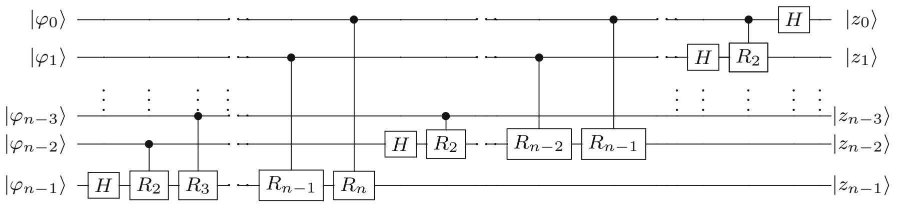

Fig. 1 The circuit of quantum Fourier transform [10]. $|\varphi_0\rangle, \ldots, |\varphi_{n-1}\rangle$ are input bits, and $|z_0\rangle, \ldots, |z_{n-1}\rangle$ are output bits. R_n is two-bit quantum controlled rotation

Without loss of generality, assuming $n = \lceil \log N \rceil$, then the circuit of QFT over $\mathbb{Z}_N$, as depicted in Fig. 1, can be implemented *exactly* by using $\frac{n(n-1)}{2}$ of controlled rotation gates, plus with n Hadamard gates, leading to the gate complexity $O(n^2)$. Recently, Su et al. [9] suggested that QFT over n-qubits can be approximate with $O(n \log n)$ T-gates.

3 Quantum algorithms for finding periods

A function over the domain $\mathcal{D}$ is called periodic if there is a unique and smallest $r > 0$ (called period) so that $f(x) = f(x + r)$ holds for every $x \in \mathcal{D}$. Say, the sine and cosine functions, respectively, have periods 2π and π over $\mathbb{R}$. Although this definition does not require r to be integer and $\mathcal{D}$ to be discrete, for the problems discussed in this survey, we only consider the settings of r being a positive integer and $\mathcal{D}$ being a discrete ring, say $\mathbb{Z}$ or $\mathbb{Z}_n$ (for some $n \in \mathbb{N}$). Intuitively, without any other heuristic information on f, say regarding f as a black box, any classical algorithm for determining whether f has a period r needs to evaluate f on, in the worse case, all elements in $\mathcal{D}$, leading to the time complexity $\mathcal{O}(|\mathcal{D}|)$ and space complexity $\mathcal{O}(1)$.

However, quantum computers can work *exponentially faster* than any classical computers toward the period finding problem. The first breakthrough on this issue can be traced back to the landmark work due to Simon [11]. Simon's algorithm is not only the first algorithm that represents a substantial advance in relativized time complexity vs. classical computing, but also a turning point in the development of quantum computation technology considering that it contains the key ingredients of the relevant algorithms that follow, including the notably Shor's quantum algorithm for integer factoring problem [7]. Very recently, Dong [12] proposed indistinguishable attack and key-recover attack toward one of the well-known cipher structure—the extended Feistel structures, including the typical block ciphers such as CAST256 and RC6.

Simon's algorithm is proposed to deal with the following problem [13]: Given a Boolean function $f : \{0, 1\}^n \to \{0, 1\}^n$ that satisfies the so-called Simon commitment condition

$$x \oplus y \in \{0, s\} \Leftrightarrow f(x) = f(y),$$

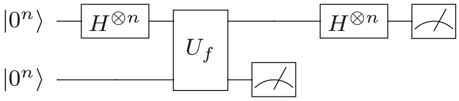

Fig. 2 The circuit of Simon's algorithm [13]. $H^{\otimes n}$ is n-bit Hadamard gate

the objective is to find $s \in \{0, 1\}^n$. Considering that $\oplus$ is the addition over binary field, the Simon's commitment condition is equivalent to that f has period r over $\mathbb{Z}_2^n$. Now, suppose that a quantum circuit U_f for implementing $|x\rangle|0\rangle \rightarrow |x\rangle|f(x)\rangle$ is at hand, then the Simon's algorithm is depicted in Fig. 2, and a modified version due to Mosca consists of the following eight steps [13]:

- *Step 1* Initializing two registers with $2n$ qubit states

$$|\varphi_0\rangle = |00\ldots0\rangle|00\ldots0\rangle$$

  and set $i = 1$.
- *Step 2* Apply n Hadamard gates to the first n-qubit register

$$|\varphi_1\rangle = \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}|x\rangle|0\rangle$$

- *Step 3* Apply U_f to the two registers

$$|\varphi_2\rangle = \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}|x\rangle|f(x)\rangle$$

- *Step 4* Measure and then discard the second register to force the first register collapsing to

$$|\varphi_3\rangle = \frac{1}{\sqrt{2}}(|x_1\rangle + |x_2\rangle)$$

  for some $x_1 \in \{0, 1\}^n$ and $x_2 = x_1 \oplus s$.
- *Step 5* Apply n Hadamard gates to the first register again

$$|\varphi_4\rangle = \frac{1}{\sqrt{2^{n+1}}}\sum_{y=0}^{2^n-1}((-1)^{x_1\cdot y} + (-1)^{x_2\cdot y})|y\rangle.$$

  It can be further simplified to

$$|\varphi_4\rangle = \frac{1}{2^{(n-1)/2}}\sum_{y\cdot s=0}(-1)^{x\cdot y}|y\rangle.$$

- *Step 6* Measure the first register to obtain a string $y_i \in \{0, 1\}^n$, and y_i can be viewed as a n dimension vector $\mathbf{y_i}$.
- *Step 7* If $i = n$, go to the next step; otherwise, let $i \leftarrow i + 1$ and go to Step 2.
- *Step 8* Let $M = [y_1, \ldots, y_n]$. Then, M is invertible with high probability. Now, we can solve the system $M \cdot s = 0$ to get s, say by using the well-known classical Gaussian elimination algorithm.

To summary, the above description of Simon's algorithm requires $O(n)$ quantum operators over $2n$ qubits, plus a classical post-processing with time complexity $O(n^3)$.

4 Quantum algorithms for factoring integer

The integer factorization problem (IFP) is given an integer N, output prime numbers p, q, where $N = pq$. It is an important problem in number theory and has attracted significant attention due to its importance in data encryption [14]. For example, the IFP is used as the basic hardness assumption for RSA cryptosystem. Up to now, the most effective classical algorithm for solving IFP is the general number field sieve [15], while the number of operations required still grows sub-exponentially with the bit length of the integer to be factorized. Quantum computing can effectively reduce the complexity of solving certain problems, and it has attracted much attention in recent years [6]. Some tested quantum computing platforms are already available, such as cloud quantum computers from IBM [16,17] based on nuclear magnetic resonance (NMR) [18] and D-Wave's quantum annealing system.

Researchers are currently focusing on two main research directions to solve the IFP via quantum computing: Shor's quantum factoring algorithm and quantum adiabatic computing (QAC).

4.1 Shor's integer factorization algorithm

It is a challenge to implement Shor's algorithm [19], since it is founded on the quantum circuit model. Vandersypen et al. [20] used a molecule with seven spin-1/2 nuclei to factor 15, yet the experiments cannot be applied to a larger number. Martín-López et al. [21] re-utilized qubits to factor 21 with Shor's algorithm by adopting an iterative protocol. Geller et al. [22] employed Fermat numbers and eight qubits to factor 51 and 85, which are the largest numbers to be factored by Shor's algorithm so far. According to Gidney [23], there should be $2k + 1$ qubits to factor k-bit integers.

From the perspective of universal quantum computation, there is still a long way to go before it could be practical.

Shor's algorithm [3] transforms the problem of factoring a given number N into an equivalent problem: Given a random positive integer a , where $a < N$, $\gcd(a, N) = 1$, find the order r of a, i.e., $a^r \equiv 1 \pmod N$. Then, p and q can be find by Euclidean algorithm. Suppose $t = \lceil \log N \rceil$ and a quantum circuit U_f for implementing $|x\rangle|0\rangle \rightarrow |x\rangle|a^x \bmod N\rangle$ is at hand, then the Shor's algorithm consists of the following seven steps:

- *Step 1* Initialize two t-qubit registers as follows:

$$|\varphi_0\rangle = |0\rangle|0\rangle$$

- *Step 2* Apply a Hadamard gate to the first register

$$|\varphi_1\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle|0\rangle$$

- *Step 3* Apply U_f in the second register

$$|\varphi_2\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle|a^x \bmod N\rangle$$

- *Step 4* Measure the second register and the first register collapsing to

$$|\varphi_3\rangle = \frac{1}{\sqrt{N}} \sum_{n=0}^{r-1} \sum_{m=0}^{N/r-1} |mr+n\rangle$$

- *Step 5* Perform quantum Fourier transform on the first register

$$|\varphi_4\rangle = \sqrt{\frac{r}{N}} \sum_{n=0}^{N/r-1} \left(\frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{-2\pi i j(mr+n)/N} |j\rangle \right).$$

When $j = \frac{kN}{r}, k = 0, 1, \ldots r-1$, it can be further simplified to

$$|\varphi_4\rangle = \frac{1}{\sqrt{r}} \left(\sum_{k=0}^{r-1} e^{-2\pi i \frac{k}{r} n} |\frac{kN}{r}\rangle \right)$$

- *Step 6* Measure the first register; we can observe the value $\lfloor \frac{kN}{r} \rceil$ with a probability no less than $\frac{4}{\pi^2 r}$.
- *Step 7* Finally, the period r can be derived by using the classical continued fraction expansion (CFE) method in polynomial time [10].

During the past decades, there are many attempts to implement Shor's algorithm over different quantum prototype computers. The number of qubits and quantum gate complexities needed for these implementations is summarized in Table 1.

4.2 Factorization by using quantum adiabatic computing

Another promising method for integer factorization is QAC [29–31], which was put forward by Burges [32,33] at first. QAC is being used on the IFP mainly in two ways:

Table 1 Different implementations of Shor's algorithms, comparison of the number of qubits and quantum gate needed

Authors	Year	Gates	Total qubits
Shor [3]	1994	$\Theta(n^3 \log n)$	$\Theta(n)$
Beckman et al. [24]	1996	$\Theta(n^3)$	$5n+1$
Veldral et al. [25]	1996	$\Theta(n^3)$	$4n+3$
Beauregard [26]	2003	$\Theta(n^3 \log \frac{n}{\xi} \log \frac{1}{\xi})$	$2n+3$
Takahashi et al. [27]	2006	$\Theta(n^3 \log \frac{n}{\xi} \log \frac{1}{\xi})$	$2n+2$
Haner et al. [28]	2016	$\Theta(n^3 \log n)$	$2n+2$
Gidney [23]	2017	$\Theta(n^3 \log n)$	$2n+1$

(1) NMR [18,34,35] and (2) quantum annealing leveraging the D-Wave system [36]. D-Wave's quantum computing system is playing a more important role than ever [33]. Although it is the strength of the NMR on long coherence time, high-accuracy quantum control, as well as NMR can be effective implementation on Grover's algorithm [37] using QAC [31,38]. D-Wave's superconducting quantum computer is standing out in terms of the number of qubits. Wang et al. [39] suggested that quantum annealing could potentially be applied to cryptanalysis, representing them to combinational optimization problems to be mapped to the D-Wave machine's theoretical model. Li et al. [40] applied both theoretical reductions and Hamiltonian transformations to successfully factor 291311, while Jiang et al. [41] recently proposed a generalized quadratic unconstrained binary optimization (QUBO) model, which is used to represent the multiplication table and the model is able to factor 376298 with 94 qubits. Wang et al. [33] optimize the problem Hamiltonian to reduce the number of qubits involved in the final Hamiltonian while maintaining the QUBO coefficients in a reasonable range, enabling the improved algorithm to factorize larger integers with fewer qubits. This algorithm using D-Wave's hybrid quantum/classical simulator `qbsolv` confirmed that performance was improved; it can factorize 1,005,973 with only 89 qubits, a new record for quantum factorized integers.

A quantum system remains in its instantaneous eigenstate if the system Hamiltonian varies slowly enough and if there is a gap between this eigenvalue and the rest of the Hamiltonian's spectrum [35]. It has been proved to be equivalent to the conventional circuit model. A quantum computer algorithm can be viewed as a specification of a Hamiltonian $H(t)$ and an initial state $|\psi(0)\rangle$.

The time-dependent Hamiltonian of the quantum system is

$$H(t) = \left(1 - \frac{t}{T}\right) H_B + \frac{t}{T} H_P$$

where H_B is the initial Hamiltonian

$$H_B = -\sum \sigma_x^{(i)}$$

Table 2 Multiplication table for factoring 143 [41]

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
p					1	p_2	p_1	1
q					1	q_2	q_1	1
					1	p_2	p_1	1
				q_1	p_2q_1	p_1q_1	q_1	
				p_2q_2	p_1q_2	q_2		
		1	p_2	p_1	1			
Carries		c_4	c_3	c_2	c_1			
$p \times q = 143$	1	0	0	0	1	1	1	1

Carries are unknown intermediate variables

and H_P is the final Hamiltonian

$$H_P = \sum h_i \sigma_z^{(i)} + \sum J_{ij} \sigma_z^{(i)} \sigma_z^{(j)}$$

The time-dependent Hamiltonian $H(t)$ of the physical system evolves according to Schrodinger equation

$$i \frac{\mathrm{d}}{\mathrm{d}t} |\psi(t)\rangle = H(t) |\psi(t)\rangle$$

Wang et al. improve the algorithm of Jiang et al. [41]. To compute the number of carry variables every column block needs, they use the target value, maximum carry and prodf function values for the column blocks to reduce the number of carry variables needed. They replace p_1 and p_2 with q_1 or $1-q_1$ and q_2 or $1-q_2$, respectively, thus further decreasing the number of qubits needed for the final QUBO model. Take $N = 143$ as an example; the algorithm steps are as follows:

- *Step 1* Divide the multiplication table into k column blocks as Table 2 and get the equations for each block

$$\begin{aligned}
&(p_2 + p_1q_1 + q_2) \times 2 + (p_1 + q_1) \\
&\quad = (11)_2 + (c_2 \times 4 + c_1 \times 2) \times 2 \\
&(q_1 + p_2q_2 + p_1 + c_2) \times 2 + (1 + p_2q_1 + p_1q_2 + 1 + c_1) \\
&\quad = (01)_2 + (c_4 \times 4 + c_3 \times 2) \times 2 \\
&(1 + c_4) \times 2 + (q_2 + p_2 + c_3) \\
&\quad = (100)_2
\end{aligned}$$

Further simplified, then we can get

$$\begin{aligned}
&2p_2 + 2p_1q_1 + 2q_2 - 8c_2 - 4c_1 + p_1 + q_1 - 3 = 0 \\
&\quad 2q_1 + 2p_2q_2 + 2p_1 + 2c_2 - 8c_4 - 4c_3 + p_2q_1 + p_1q_2 + c_1 + 1 = 0 \\
&\quad q_2 + p_2 + c_3 + 2c_4 - 2 = 0
\end{aligned}$$

- *Step 2* Construct the cost function

$$f(p_1, p_2, q_1, q_2, c_1, c_2, c_3, c_4) = (N - pq)^2 = A^2 + B^2 + C^2$$

with

$$\begin{aligned} A &= 2p_2 + 2p_1q_1 + 2q_2 - 8c_2 - 4c_1 + p_1 + q_1 - 3, \\ B &= 2q_1 + 2p_2q_2 + 2p_1 + 2c_2 - 8c_4 - 4c_3 + p_2q_1 + p_1q_2 + c_1 + 1, \\ C &= q_2 + p_2 + c_3 + 2c_4 - 2. \end{aligned}$$

- *Step 3* Transform the k-bit ($k \geq 3$) coupling terms into quadratic term according to the following equations:

$$\begin{aligned} x_1x_2x_3 &= \min_{x_4}(x_4x_3 + 2(x_1x_2 - 2x_1x_4 - 2x_2x_4 + 3x_4)) \\ -x_1x_2x_3 &= \min_{x_4}(-x_4x_3 + 2(x_1x_2 - 2x_1x_4 - 2x_2x_4 + 3x_4)) \end{aligned}$$

- *Step 4* Replace p_1q_1, p_1q_2, p_2q_2 and p_2q_1 with t_1, t_2, t_3 and t_4, respectively. Further, rename the variables $p_1, p_2, q_1, q_2, c_1, \ldots, c_4, t_1, \ldots, t_4$ as $v_1, \ldots, v_12$, and replace $v_i = \frac{1-s_i}{2}$ $(i = 1, \ldots, 12)$ to make the variables lie in the domain $\{-1, 1\}$. Now, the above cost function f can be rewritten as

$$f(p_1, p_2, q_1, q_2, c_1, c_2, c_3, c_4) = 2f'(s_1, \ldots, s_{12})$$

where f' is given in Fig. 3.
- *Step 5* Now, we can review the above cost function as an Ising Hamiltonian with local fields, and the values of h_i and J_{ij} can be derived accordingly (See Fig. 3 for details).
- *Step 6* Solve the Ising Hamiltonian system by calling `qbsolv`, the Python library provided by D-Wave systems, and map the results back to the prime factorization of N.

To summary, the qubits needed for different implementations of quantum factorization based on QAC are shown in Table 3.

Table 3 Different implementations of quantum factorization based on QAC

Authors	Year	The largest integer can be factorized	Qubits
Li et al. [40]	2018	291,311	–
Jang et al. [41]	2018	376,298	94
Wang et al. [33]	2019	1,005,973	89

Comparison of the largest integer can be factorized

$$f'(s_1, s_2, \cdots, s_{12}) = \tfrac{261}{2}s_1 + \tfrac{215}{2}s_2 + \tfrac{261}{2}s_3 + \tfrac{215}{2}s_4 - 41s_5 - 82s_6 + 3s_7 + 6s_8 - 137s_9 - 81s_{10} - 107s_{11} - 81s_{12}$$
$$+2s_1s_2 + 79s_1s_3 + \tfrac{95}{2}s_1s_4 - 2s_1s_5 - 4s_1s_6 - 8s_1s_7 - 16s_1s_8 - 148s_1s_9 - 84s_1s_{10}$$
$$+\tfrac{95}{2}s_2s_3 + 71s_2s_4 - 8s_2s_5 + -16s_2s_6 + s_2s_7 + 2s_2s_8 + 6s_2s_9 + 6s_2s_{10} - 124s_2s_{11} - 84s_2s_{12}$$
$$+2s_3s_4 - 2s_3s_5 - 4s_3s_6 - 8s_3s_7 - 16s_3s_8 - 148s_3s_9 - 84s_3s_{12}$$
$$-8s_4s_5 - 16s_4s_6 + s_4s_7 + 2s_4s_8 + 6s_4s_9 - 84s_4s_{10} - 124s_4s_{11} + 6s_4s_{12}$$
$$+34s_5s_6 - 4s_5s_7 - 8s_5s_8 - 8s_5s_9 + s_5s_{10} + 2s_5s_{11} + s_5s_{12}$$
$$-8s_6s_7 - 16s_6s_8 - 16s_6s_9 + 2s_6s_{10} + 4s_6s_{11} + 2s_6s_{12}$$
$$+34s_7s_8 - 4s_7s_{10} - 8s_7s_{11} - 4s_7s_{12}$$
$$-8s_8s_{10}16s_8s_{11}8s_8s_{12}$$
$$+s_9s_{11}$$
$$+794$$

$$h^T = (\sigma_z^{(1)}, \cdots, \sigma_z^{(12)}) = (130.5, 107.5, 130.5, 107.5, -41, -82, 3, 6, -137, -81, -107, -81)$$

$$J = \begin{array}{c|cccccccccccc}
 & \sigma_z^{(1)} & \sigma_z^{(2)} & \sigma_z^{(3)} & \sigma_z^{(4)} & \sigma_z^{(5)} & \sigma_z^{(6)} & \sigma_z^{(7)} & \sigma_z^{(8)} & \sigma_z^{(9)} & \sigma_z^{(10)} & \sigma_z^{(11)} & \sigma_z^{(12)} \\
\sigma_z^{(1)} & 0 & 2 & 79 & 47.5 & -2 & -4 & -8 & -16 & -148 & -84 & 0 & 0 \\
\sigma_z^{(2)} & & 0 & 47.5 & 71 & -8 & -16 & 1 & 2 & 6 & 6 & -124 & -84 \\
\sigma_z^{(3)} & & & 0 & 2 & -2 & -4 & -8 & -16 & -148 & 0 & 0 & -84 \\
\sigma_z^{(4)} & & & & 0 & -8 & -16 & 1 & 2 & 6 & -84 & -124 & 6 \\
\sigma_z^{(5)} & & & & & 0 & 34 & -4 & -8 & -8 & 1 & 2 & 1 \\
\sigma_z^{(6)} & & & & & & 0 & -8 & -16 & -16 & 2 & 4 & 2 \\
\sigma_z^{(7)} & & & & & & & 0 & 34 & 0 & -4 & -8 & -4 \\
\sigma_z^{(8)} & & & & & & & & 0 & 0 & -8 & -16 & -8 \\
\sigma_z^{(9)} & & & & & & & & & 0 & 0 & 1 & 0 \\
\sigma_z^{(10)} & & & & & & & & & & 0 & 0 & 0 \\
\sigma_z^{(11)} & & & & & & & & & & & 0 & 0 \\
\sigma_z^{(12)} & & & & & & & & & & & & 0
\end{array}$$

Fig. 3 Ising Hamiltonian system for factoring 143

5 Quantum algorithms for discrete logarithmic problems

Let $G = \langle g \rangle$ be a cyclic group of order p and g be a generator of G. The discrete logarithmic problem (DLP) over G is to find an integer r such that $g^r = y$ for given $y \in G$. Diffie–Hellman key exchange protocol, ElGamal encryption and most elliptic curve cryptosystems are based on the difficulty of computing discrete logarithms. At present, the best known classical algorithm for solving DLP is the so-called index-calculate method (ICM) that requires sub-exponential classical operations.

In 1994, Shor [3] put forward a polynomial time quantum algorithm to solve the discrete logarithmic problem in group. Proos et al. [42] further extended Shor's quantum DLP algorithm to elliptic curves [42–44]. In 2012, Myasnikov et al. proposed a quantum algorithm for the DLP over matrices of finite group rings [45]. Childs [46] described an effective quantum algorithm for computing discrete logarithms over semigroups. Recently, further generalized Shor's quantum DLP algorithms are proposed for different algebraic structures [47–50].

With the identity $g^r = y$, if we define a binary function $f : \mathbb{Z}_{p-1} \times \mathbb{Z}_{p-1} \to \mathbb{Z}_p$ as follows:

$$(a, b) \mapsto g^a y^b \bmod p,$$

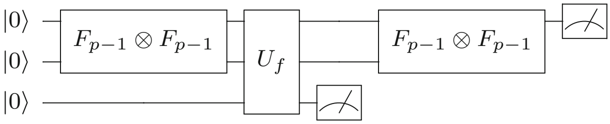

Fig. 4 The circuit of discrete logarithmic problems [51]. F_{p-1} is the Fourier transform over Z_{p-1}

then f takes $(r, -1)$ as its period, considering that

$$f(a+r, b-1) = f(a, b) \quad (\forall a, b \in \mathbb{Z}_{p-1}).$$

Therefore, the aforementioned idea for finding period can be used to solve the discrete logarithms problems.

Now, suppose a quantum circuit U_f for implementing

$$|a\rangle|b\rangle|0\rangle \rightarrow |a\rangle|b\rangle|g^a y^b \text{ mod } p\rangle$$

is at hand, then Shor's discrete logarithm algorithm is depicted in Fig. 4. A modified version of Shor's quantum DLP algorithm, due to Wang [51], consists of the following six steps:

- *Step 1* Initialize three quantum registers

$$|\varphi_0\rangle = |0\rangle|0\rangle|0\rangle$$

- *Step 2* Apply the $F_{p-1} \otimes F_{p-1}$ on the first two registers and get the superposition

$$|\varphi_1\rangle = \frac{1}{p-1}\sum_{a=0}^{p-2}\sum_{b=0}^{p-2}|a\rangle|b\rangle|0\rangle$$

- *Step 3* Perform U_f in the third register

$$|\varphi_2\rangle = \frac{1}{p-1}\sum_{a=0}^{p-2}\sum_{b=0}^{p-2}|a\rangle|b\rangle|g^a y^b \text{ mod } p\rangle$$

- *Step 4* Measure and then discard the third register, leading that the first two registers collapse to

$$|\varphi_3\rangle = \frac{1}{\sqrt{p-1}}\sum_{\lambda=0}^{p-2}|a_0 + \lambda r\rangle|b_0 - \lambda\rangle.$$

- *Step 5* Apply QFT to the first two registers, and we get

$$|\varphi_4\rangle = \frac{1}{\sqrt{p-1}}\sum_{u=0}^{p-1} e^{\frac{2\pi i(a_0+b_0 r)u}{p-1}}|u\rangle|ru\rangle$$

Table 4 The complexity of quantum DLP algorithms

Authors	Year	Time complexity	Space complexity
Shor et al. [3]	1994	$\mathcal{O}(n^3)$	$\mathcal{O}(n)$
Proos et al. [42]	2003	$\mathcal{O}(n^2)$	–
Ekera et al. [49]	2019	–	$\mathcal{O}(\frac{n}{2})$

The comparison of time and space complexities is in table

- *Step 6* Measure the first two registers to get $|u_0\rangle|ru_0\rangle$, and then, derive r by $r = ru_0u_0^{-1} \bmod (p-1)$ (assuming that $\gcd(u_0, p-1) = 1$ with high probability).

The complexities of Shor's quantum DLP algorithm and some related algorithms are collected in Table 4.

6 Quantum algorithms for abelian hidden subgroup problems

Let H be the subgroup of group G, S be any set and $f : G \rightarrow S$ a function that distinguishes cosets of H, i.e., $\forall g_1, g_2 \in G$, $f(g_1) = f(g_2) \Leftrightarrow g_1H = g_2H$. The hidden subgroup problem (HSP) is to find the subgroup H using f. To solve this problem classically, $\Omega(|G|)$ queries on f are required, while it is solvable on a quantum computer using merely $O(\log |G|)$ f-queries.

In 1995, Kitaev [52] gave a polynomial quantum algorithms to solve the abelian stabilizer problem (ASP) and prove that the integer factorization and discrete logarithm problems can be solved as special cases. In 1995, Dan and Lipton [53] first built the relationship between quantum algorithm and HSP and designed a quantum algorithm to solve the hidden linear function. In 1997, Brassard and Hoyer [54] extended the Simon's problem to HSP. In 1998, Jozsa [55] gave a unified description of Deustch–Jozsa's algorithm, Simon's algorithm and Shor's algorithm in the form of HSP. Subsequently, Mosca [56,57] and Jozsa [58] introduced the more general abelian HSP and gave quantum Fourier transform to solve it. Abelian HSP mainly focuses on finite abelian groups, and related algorithms can be seen in [57,59].

The algorithm of general finite abelian HSP rewrote by Damgard [60] consists of the following six steps:

- *Step 1* Prepare the initial state

$$|\varphi_0\rangle = |0\dots0\rangle|0\dots0\rangle$$

- *Step 2* Apply QFT to the first register, and we get

$$|\varphi_1\rangle = \frac{1}{\sqrt{|G|}}\sum_{g\in G}|g\rangle|0\rangle$$

- *Step 3* Apply U_f to get

$$|\varphi_2\rangle = \frac{1}{\sqrt{|G|}} \sum_{g \in G} |g\rangle |f(g)\rangle$$

- *Step 4* Measure the second register, and suppose the value is $f(g_0)$,

$$|\varphi_3\rangle = \frac{1}{\sqrt{H}} \sum_{h \in H} |g_0 + h\rangle |f(g_0)\rangle$$

- *Step 5* Apply QFT to the first register, and we get

$$|\varphi_4\rangle = \frac{1}{\sqrt{|H||G|}} \sum_{g \in G} \left(\chi_g(g_0) \sum_{h \in H} \chi_g(h) \right) |g\rangle$$

  where $\sum_{h \in H} \chi_g(h) = 0$ if and only if $g \notin H^{\perp}$.
  Othogonal subgroup $H^{\perp}$ defined as $H^{\perp} = \{g \in G | \chi_g(h) = 1, \quad \forall h \in H\}$. $|\varphi_4\rangle$ can be further simplied to

$$|\varphi_4\rangle = \frac{1}{\sqrt{|H||G|}} \sum_{g \in H^{\perp}} |H||g\rangle = \sqrt{\frac{|H|}{|G|}} \sum_{g \in H^{\perp}} |g\rangle = |H^{\perp}\rangle$$

- *Step 6* Measure the first register, and we can obtain $H^{\perp}$; then, H can be obtained by $H = (H^{\perp})^{\perp}$.

7 Quantum algorithms for hidden shift problems

Given a finite group G, a finite set R and two maps $f, g : G \to R$, the hidden shift problem is to find some $s \in G$ such that $f(x) = g(x + s)$ for all $x \in G$. At least $\sqrt{N}$ queries are necessary for hidden shift problem by reduction from Grover's problem. However, on quantum computer, only $\mathcal{O}(1)$ queries can solve certain special cases of hidden shift problems. The hidden shift problem was first introduced and studied by van Dam et al. [61,62] in 2003. The shifted Legendre symbol algorithm [63,64] is classified as this special case, and no classical algorithm in $\mathcal{O}(\text{polylog}N)$ time has been found to solve these problems. In addition, the shifted Legendre symbol problem's quantum algorithm will destroy the specific cryptographic pseudorandom generator and it has the ability to make quantum queries to the generator [62]. There has a connection between the hidden shift problem and the hidden subgroup problem, hidden subgroup problem over dihedral group is equivalent to the hidden shift problem over $\mathbb{Z}_N$, and graph isomorphism can be cast as a hidden shift problem over S_n [10–12]. The study of the hidden shift problem can give an arguably more natural view to tackle the graph isomorphism problem [12]. Based on the "pretty good measurement," Childs et al. [65] proposed a quantum algorithm for the generalized hidden shift problem:

$f \in \mathbb{Z}_M \times \mathbb{Z}_N$ satisfying $f(b, x) = f(b+1, x+s)$ for $b \in \mathbb{Z}_M$ and $x, s \in \mathbb{Z}_N$. In 2010, Roetteler [66] gave an efficient quantum algorithm for solving the hidden shift problem for several classes of the so-called bent functions. Gavinsky et al. [67] gave an efficient quantum algorithm for solving the hidden shift problem for the average case Boolean functions in 2001. Ozols et al. [68] gave another quantum algorithm for the Boolean hidden shift problem based on a quantum analogue of the rejection sampling.

The quantum algorithm for a generalized hidden shift problem by Childs is as follows:

- *Step 1* Initialize the three registers

$$|\varphi_0\rangle = |0\rangle|0\rangle|0\rangle$$

- *Step 2* Apply Hadamard gates on the first two registers and get the superposition

$$|\varphi_1\rangle = \frac{1}{\sqrt{MN}} \sum_{b\in\mathbb{Z}_M} \sum_{x\in\mathbb{Z}_N} |b\rangle|x\rangle|0\rangle$$

- *Step 3* Apply U_f on the last two registers

$$|\varphi_2\rangle = \frac{1}{\sqrt{MN}} \sum_{b\in\mathbb{Z}_M} \sum_{x\in\mathbb{Z}_N} |b\rangle|x\rangle|f(b, x)\rangle$$

- *Step 4* Measure the third register and discard it, the second register will collapsed, and then,

$$|\varphi_3\rangle = \frac{1}{\sqrt{M}} \sum_{b=0}^{M-1} |b\rangle|x + bs\rangle.$$

  The results are equal to the mixed state described by the density matrix

$$\rho_s := \frac{1}{N} \sum_{x\in\mathbb{Z}_N} |\phi_{x,s}\rangle\langle\phi_{x,s}|.$$

Now, we need to discuss how to derive s according to three different cases.

- When M is very large, s can be identified by the period finding method mentioned in Sect. 3.
- Otherwise, Childs et al. [65] use $k > 1$ states and PGM to obtain s as follows:
  - Apply QFT on the second register over $\mathbb{Z}_N$ to get

$$\widetilde{\rho}_s^{\otimes k} = \frac{1}{(MN)^k} \sum_{x\in\mathbb{Z}_N^k} \sum_{b,c\in\mathbb{Z}_M^k} \omega^{(b\cdot x - c\cdot x)s} |b, x\rangle\langle c, x|$$

$$= \frac{1}{(MN)^k} \sum_{x\in\mathbb{Z}_N^k} \sum_{\omega,\nu\in\mathbb{Z}_N} \omega^{(\omega-\nu)s} \sqrt{\eta_\omega^x \eta_\nu^x} |S_\omega^x, x\rangle\langle S_\nu^x, x|$$

where

$$|S_\omega^x\rangle := \frac{1}{\sqrt{\eta_\omega^x}} \sum_{b\in S_\omega^x} |b\rangle$$
$$\eta_\omega^x := |S_\omega^x|$$

- Then, the hidden shift s can be identified using the pretty good measurement with at least a constant probability [65].

8 Quantum algorithms for dihedral hidden subgroup problems and lattice problems

The dihedral group is a symmetric group generated by the reflection and rotation. It contains $2N$ elements:

$$D_N = \langle s, r | s^2 = r^N = 1, srs = r^{-1}\rangle$$

where s can be viewed as a reflection about some fixed axis, and r is a rotation by an angle $\frac{2\pi}{N}$. Moreover, D_N is isomorphic to a semidirect product of the two cyclic groups $\mathbb{Z}_2$ and $\mathbb{Z}_N$ of order 2 and N, respectively,

$$D_N = \mathbb{Z}_2 \rtimes_\phi \mathbb{Z}_N$$

with multiplication defined by

$$(a_1, b_1)(a_2, b_2) = (a_1 + a_2, b_1 + \phi(a_1)(b_2)).$$

The homomorphism $\phi : \mathbb{Z}_2 \to \mathrm{Aut}(\mathbb{Z}_N)$ is specified by

$$\phi(0)(b) = b, \quad \text{and} \quad \phi(1)(b) = -b.$$

An element $(a, b) \in D_N$ is a *rotation* if $a = 0$, and a *reflection* if $a = 1$ [69]. Now, let us consider a hidden subgroup $H = \{(0, 0), (1, d)\}$ for some unknown $d \in \mathbb{Z}_N$. That is, H is the subgroup group generated by an unknown reflection $r = (1, d)$. The dihedral hidden subgroup problem (dHSP) with respect to H is to find d. This problem is also formulated quantum as the so-called dihedral coset problem (DCP): Given many superpositions $\{\frac{1}{\sqrt{2}}(|0, x_i\rangle + |1, x_i + d\rangle) : x_i \in \mathbb{Z}_N\}_{i\leq\ell}$ (i.e., the coset of H), the objective is to find d.

In 1998, Ettingcr and Hoyer [69] were the first to study dihedral hidden subgroup problems (dHSP). They divided the dihedral group into two subgroups of rotation and reflection and searched for the hidden subgroups of each subgroup, respectively. They

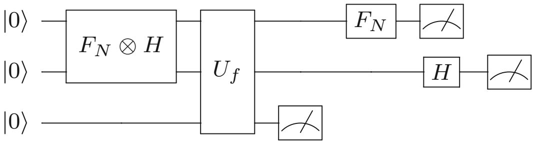

Fig. 5 Determining the parity of d [51]. F_N is Fourier transform, and H is Hadamard gate

claimed that it is enough to solve dHSP if the generator of the hidden subgroup of the reflected subgroup is known. In 2002, Regev [5] found the connection between the unique shortest vector problem (uSVP) and dHSP and pointed out that if dHSP can be effectively solved, the unique shortest vector problem of lattice can also be effectively solved. Kuperberg [76] first proposed a sub-exponential quantum algorithm of dHSP. In 2004, Regev [5] abstracted Kuperberg's sieve method as a pipeline, reducing the quantum space complexity of the original algorithm to polynomial level, but the time complexity is still sub-exponential. In 2011, Kuperberg [76] improved the original algorithm and Regev's polynomial space algorithm and proposed another sub-exponential quantum algorithm of dHSP. The time complexity of the improved algorithm is slightly reduced, but in the worst case, the algorithm is Regev's algorithm. In 2016, Roetteler [71] first raised a quantum algorithm that can solve a special type of dHSP problem in polynomial time and space complexities: When $N = 2^m - 1$, more than $\mathcal{O}(2^{m^2})$ instances are easy to solve among dHSP problem on the dihedral group D_N , that is, the total number of easily solved instances increases exponentially with m.

In recent years, significant progress has been made in lattice-based cryptography among the post-quantum public-key cryptography. Many lattice-based public-key encryption schemes have been proposed in light of the fact that some lattice problems [72–74] such as unique shortest vector problem (uSVP) are the foundation of the trapdoor one-way function. However, uSVP can be reduced to a kind of non-abelian hidden subgroup problem [5]: the dihedral hidden subgroup problem. Therefore, the study on quantum algorithm for the dihedral hidden subgroup problem has great significance for the security of lattice-based cryptography.

8.1 Kuperberg's quantum algorithm for dHSP

In 2003, Kuperberg [76] reduced the dHSP to finding the slope d when $N = 2^n$ and $H = \langle(1, d)\rangle$. Suppose the black box $f : D_N \to R$ for hidden H is given and U_f (i.e., the quantum circuit for implement f) is at hand, where R is the range of f. That is, f on each coset of H is constant. Now, a quantum algorithm for determining the parity of d, due to Kuperberg, is depicted in Fig. 5 [51] and described as the following eight steps:

- *Step 1* Initialize the register

$$|\varphi_0\rangle = |0\rangle|0\rangle|0\rangle$$

- *Step 2* Prepare the initial quantum state

$$|\varphi_1\rangle = \sum_{x=0}^{2^n-1}\sum_{y=0}^{1}|x\rangle|y\rangle|f(x,y)\rangle$$

- *Step 3* Measure the third register

$$\begin{aligned}|\varphi_2\rangle &= \frac{1}{\sqrt{2}}(|x_0\rangle|y_0\rangle + |x_0 + (-1)^{y_0}d\rangle|y_0+1\rangle)\\ &= \begin{cases}\frac{1}{\sqrt{2}}(|x_0\rangle|0\rangle + |x_0+d\rangle|1\rangle) & y_0 = 0\\ \frac{1}{\sqrt{2}}(|x_0-d\rangle|0\rangle + |(x_0-d)+d\rangle|1\rangle) & y_0 = 1\end{cases}\end{aligned}$$

because x_0, y_0 are any value $|\varphi_2\rangle$ which can be generalized to

$$|\varphi_2\rangle = \frac{1}{\sqrt{2}}(|x\rangle|0\rangle + |x+d\rangle|1\rangle)$$

- *Step 4* Apply QFT to the first register

$$\begin{aligned}|\varphi_3\rangle &= \frac{1}{\sqrt{2^{n+1}}}\sum_{z=0}^{2^n-1}(e^{\frac{2\pi izx}{2^n}}|z\rangle|0\rangle + e^{\frac{2\pi iz(x+d)}{2^n}}|z\rangle|1\rangle)\\ &= \frac{1}{\sqrt{2^{n+1}}}\sum_{0}^{2^n-1}e^{\frac{2\pi izx}{2^n}}|z\rangle\otimes(|0\rangle + e^{\frac{2\pi izd}{2^n}}|1\rangle)\end{aligned}$$

- *Step 5* Measure the first register, ignore the global phase $e^{\frac{2\pi iz_0x}{2^n}}$ and the first register, and the second register collapses to

$$|\varphi_{z_0}\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + e^{\frac{2\pi iz_0d}{2^n}}|1\rangle\right)$$

- *Step 6* Apply Kuperberg's sieve, continue the combines operation, and gain the target quantum state

$$\begin{aligned}|\varphi_{2^{n-1}}\rangle &= \left(|0\rangle + e^{\frac{2\pi i2^{n-1}d}{2^n}}|1\rangle\right)\\ &= \left(|0\rangle + e^{\pi id}|1\rangle\right) = \left(|0\rangle + (-1)^d|1\rangle\right)\end{aligned}$$

- *Step 7* Apply Hadamard gate

$$H|\varphi_{2^{n-1}}\rangle = \left(\frac{1+(-1)^d}{2}\right)|0\rangle + \left(\frac{1-(-1)^d}{2}\right)|1\rangle$$

Table 5 The complexity of DHSP

Authors	Year	Time complexity	Space complexity		Query complexity
			Quantum	Classical	
Kuperberg et al. [76]	2003	$2^{\mathcal{O}(\sqrt{\log N})}$	$2^{\mathcal{O}(\sqrt{\log N})}$	–	$2^{\mathcal{O}(\sqrt{\log N})}$
Regev et al. [5]	2004	$2^{\mathcal{O}(\sqrt{\log(\log N)\log N})}$	$2^{\mathcal{O}(\sqrt{N\log N})}$	–	$2^{\mathcal{O}(\sqrt{\log(\log N)\log N})}$
Kuperberg et al. [70]	2011	$2^{\mathcal{O}(\sqrt{\log N})}$	$2^{\mathcal{O}(\sqrt{\log N})}$	$2^{\mathcal{O}(\sqrt{\log N})}$	$2^{\mathcal{O}(\sqrt{\log N})}$

Time, space and query complexity are in the table

- *Step 8* Measure the second register. If 0 is observed, d is even; otherwise, d is odd.

After the parity (i.e., the least significant bit) of d is found, Kuperberg suggested to use the sieving idea to find all bits of d iteratively. The following steps are Kuperberg's sieve idea, reformulated by Regev [5].

- When d is even. Then, consider the black box $f' : D_{N/2} \to R$ given by $f(a,b) := f(a,2b)$. Note that this function hides the subgroup $H' = \langle(1,d')\rangle$ of $D_{N/2}$ with $d' = d/2$.
- When d is odd. Then, consider the black box $f'' : D_{N/2} \to R$ given by $f(a,b) := f(a,2b+1)$. Note that this function hides the subgroup $H'' = \langle(1,d'')\rangle$ of $D_{N/2}$ with $d'' = (d-1)/2$.

We can now obtain the second least significant bit of d (i.e., the parity of d' or d'') by calling the above algorithm with either f' or f'' [5]. By continuing this process iteratively, we can find all the bits of d (Table 5).

8.2 LWE and EDCP

Given $m \geq n$ samples of the form $(a,b) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$, with $a \leftarrow \mathbb{Z}_q^n$ and $b = \langle a, s\rangle + e$, where $e \leftarrow D_{\mathbb{Z},\alpha q}$ and $s \in_R \mathbb{Z}_q^n$, the learning with errors (LWE) problem is to find the secret vector s. The hardness of the learning with errors (LWE) problem is one of the most fruitful resources of modern cryptography. In particular, it is one of the most prominent candidates for secure post-quantum cryptography. Understanding its quantum complexity is therefore an important goal.

In 2005, Regev [75] first proposed the LWE problem and proved that the LWE problem is difficult under proper assumptions. Since then, this problem has proved to be as difficult as the worst-case lattice problem, which has become the basis of a large number of encryption applications in recent years.

Regev [5] showed that uSVP and, therefore, also BDD and LWE are no harder to solve than the DCP problem. The best known algorithm for DCP, due to Kuperberg [76], runs in time$2^{O(\log l + \log N/\log l)}$which does not improve upon classical methods for solving LWE. Regev showed that DCP can be solved given efficient algorithms for the subset-sum problem (which is classically defined), however in a regime of parameters that appear harder to solve than LWE itself. In 2013, Li et al. [77] present a quantum algorithm to generate the input of the two-point problem which hides the solution of LWE; then they give a new reduction from two-point problem to

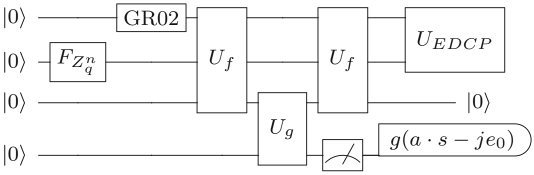

Fig. 6 From LWE to EDCP [80]. GR02 is the algorithm in [81], $F_{Z_q^n}$ is Fourier transform over Z_q^n, and U_g is an unitary transformation. The output state is U_{EDCP}

dihedral coset problem. Their reduction implicates that any algorithm solved DCP in sub-exponential time would lead a quantum algorithm for LWE. In 2016, Eldar and Shor [78] proposed a quantum algorithm to solve the bounded-distance-decoding (BDD) problem on lattice and claimed that some parameters of the algorithm could be improved to attack the cryptographic system based on the LWE problem. Although the algorithm found has problem later [78], the technique of "smoothing" analysis of lattices by using systematic normal form (SysNF) provided a new idea for the direct solution of lattices. Subsequently, they systematically explained how to use SysNF technology to effectively carry out discrete Fourier transform [79] (DFT) in the any distribution that is sufficiently "smooth" of any lattice, which provided a new possible approach for analyzing lattice point structure based on DFT eigenvector and solving SVP equilateral lattice problem.

In 2018, Brakerski et al. [80] show the equivalence between LWE and the extrapolated dihedral coset problem (EDCP) by building quantum reductions between them. The EDCP problem over D_N is specified as follows: Given ℓ many registers in a normalized state corresponding to

$$\sum_{j\in\mathbb{Z}} e^{-\pi\frac{|j|^2}{r^2}} |j, (x_i + j\cdot s) \bmod N\rangle$$

where $x_i \in \mathbb{Z}_N^n$ $(i = 1, \ldots, \ell)$, and $s \in \mathbb{Z}_N^n$ is fixed, the objective of EDCP is to find the secret value s.

(1) Quantum reduction from LWE to EDCP.

An instance of LWE problem over the lattice $\mathcal{L}(A)$, $A \in \mathbb{Z}_q^{m\times n}$, can be reduced to an instance of EDCP problem over the dihedral group D_N, $N = 2^n$, according to the following quantum steps (Fig. 6):

- *Step 1* Initialize the four registers with required qubits

$$|\varphi_1\rangle = |0\rangle|0\rangle|0\rangle|0\rangle$$

- *Step 2* Perform QFT on the second register (normalization omitted)

$$|\varphi_2\rangle = \sum_{s\in\mathbb{Z}_q^n} |0\rangle|s\rangle|0\rangle|0\rangle$$

- *Step 3* Apply GR02 algorithm [81] in the first register, which is a quantum process to create a superposition state according to given probability distribution

$$|\varphi_3\rangle = \sum_{s\in\mathbb{Z}_q^n}(\sum_{j\in\mathbb{Z}}\rho_r(j)|j\rangle)|s\rangle|0\rangle|0\rangle$$

- *Step 4* Suppose that the quantum circuit U_f

$$\mathrm{U}_f|j\rangle|s\rangle|0\rangle \to |j\rangle|s\rangle|As - jb \bmod q\rangle$$

  is at hand. Apply U_f on the first three registers

$$\begin{aligned}|\varphi_4\rangle &= \sum_{s\in\mathbb{Z}_q^n, j\in\mathbb{Z}}\rho_r(j)|j\rangle|s\rangle|As - j\cdot As_0 - je_0\rangle|0\rangle\\ &= \sum_{s\in\mathbb{Z}_q^n, j\in\mathbb{Z}}\rho_r(j)|j\rangle|s + js_0\rangle|As - je_0\rangle|0\rangle\end{aligned}$$

- *Step 5* Further, suppose that the quantum circuit U_g

$$\mathrm{U}_g|x\rangle|0\rangle \to |x\rangle|x/z - w \bmod \bar{q}\rangle$$

  is at hand, where $\bar{q} = q/z = c$. Apply U_g on the last two registers, and we get

$$|\varphi_5\rangle = \sum_{s\in\mathbb{Z}_q^n, j\in\mathbb{Z}}\rho_r(j)|j\rangle|s + j\cdot s_0\rangle|As - je_0\rangle|g(As - je_0)\rangle$$

- *Step 6* Measure the fourth register and discard it

$$|\varphi_6\rangle = \sum_{j\in\mathbb{Z}}\rho_r(j)|j\rangle|s + j\cdot s_0\rangle|As - je_0\rangle$$

- *Step 7* Apply U_f to the first three registers, the third register gives 0 and discard it, and the state is of the form

$$|\varphi_7\rangle = \sum_{j\in\mathbb{Z}}\rho_r(j)|j\rangle|s + j\cdot s_0\rangle$$

- *Step 8* Repeat the above procedure ℓ times, and we obtain ℓ many EDCP states with probability $(1 - \frac{1}{k})^{m\ell}$

$$|\varphi_{\mathrm{EDCP}}\rangle = \{\sum_{j\in\mathbb{Z}}\rho_r(j)|j\rangle|s + j\cdot s_0\rangle\}_{k\leq\ell}$$

  where $x_{k\in\mathbb{Z}_q^n}$.

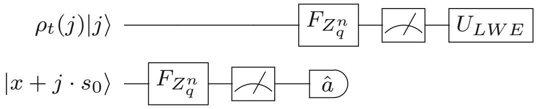

Fig. 7 From EDCP to LWE [80]. $F_{Z_q^n}$ is Fourier transform over Z_q^n, and U_{EDCP} is the output state

(2) Quantum reduction from EDCP to LWE.

The reverse quantum reduction from EDCP to LWE is given below (Fig. 7).

- *Step 1* Prepare the input state

$$|\varphi_1\rangle = \sum_{j \in \mathbb{Z}} \rho_r(j)|j\rangle|x + j \cdot s_0 \bmod q\rangle$$

- *Step 2* Apply QFT on the second register

$$|\varphi_2\rangle = \sum_{a \in \mathbb{Z}_q^n} \sum_{j \in \mathbb{Z}} \omega_q^{\langle (x+j \cdot s_0), a\rangle} \cdot \rho_r(j)|j\rangle|a\rangle$$

where $\omega_q = e^{\frac{2\pi i}{q}}$

- *Step 3* Measure the second register and obtained a_k, omitting global phase $\omega_q^{\langle x, \widehat{a}\rangle}$

$$|\varphi_3\rangle = \sum_{j \in \mathbb{Z}} \omega_q^{j \cdot \langle \widehat{a}, s_0\rangle} \cdot \rho_r(j)|j, \widehat{a}\rangle$$

- *Step 4* Apply QFT on the first register

$$|\varphi_4\rangle = \sum_{b \in \mathbb{Z}_q} \sum_{j \in \mathbb{Z}_q} \omega_q^{j \cdot (\langle \widehat{a}, s_0\rangle + b)} \cdot \rho_r(j)|b\rangle$$

Using Poisson summation formula to reorganize $|\varphi_4\rangle$, then

$$|\varphi_4\rangle = \sum_{e \in \mathbb{Z}} \rho_{\frac{1}{2}}\left(\frac{e}{q}\right)| - \widehat{a}, s_0\rangle + e \bmod q\rangle$$

- *Step 5* Measure the first register, and we can obtain an LWE sample

$$|\varphi_{\text{LWE}}\rangle = (-\widehat{a}, \langle -\widehat{a}, s_0\rangle + e_k)$$

9 Conclusion

With the rapid development of quantum computing, it broke through the defense line of the classic cryptosystems, which makes the post-quantum cryptography become the frontier of research. In order to search the novel cryptography which is resistant to quantum attack, it is of great necessity to conduct a systematical analysis of the quantum algorithms that could solve the typical hard problems. In this paper, we start from the typical hard problems: integer factorization problem, discrete logarithmic problem and dihedral hidden subgroup problems in the public-key cryptosystem (respectively, RSA, ElGamal, ECC); then, we analyze the latest development of quantum algorithms; besides, the limitation of typical cryptosystem (RSA, ElGamal, ECC) and its vulnerability to quantum attacks, as well as the explanation to the resistance of lattice cryptography to quantum attacks are all elaborated. For future research, analyzing the isogeny, multivariable and seeking for the quantum algorithms for problems such as hash collision should be of guiding significance.

Acknowledgements This work was supported by the National Key Research and Development Program of China (2018YFE0126000), the National Natural Science Foundation of China (NSFC) (No. 61972050), the Shandong provincial Key R&D Program of China (Grant No. 2018CXGC0701) and the 111 Project (No. B08004).

References

1. Rivest, R.L., Shamir, A., Adleman, L.: A method for obtaining digital signatures and public-key cryptosystems. Commun. ACM. **21**(2), 120–126 (1978)
2. Miller, V.S.: Use of elliptic curves in cryptography. In: Advances in Cryptology-CRYPTO'85, Santa Barbara, California, USA, pp. 18–22 (1985)
3. Shor, P.W.: Algorithms for quantum computation: Discrete logarithms and factoring. In: Proceedings 35th Annual Symposium on Foundations of Computer Science, pp. 124–134 (1994)
4. Grover, L.K.: A fast quantum mechanical algorithm for database search. arXiv:quant-ph/9605043 (1996)
5. Regev, O.: Quantum computation and lattice problems. SIAM J. Comput. **33**(3), 738–760 (2004)
6. Loceff, M.: A course in quantum computing (for the community college). Foothill College.https://scholar.google.com/scholar?cluster=18303662284423939245&hl=zh-CN&as_sdt=2005&sciodt=0,5 (2015)
7. Nielsen, M.A., Chuang, I.: Quantum Computation and Quantum Information. Cambridge University Press, England (2012)
8. Zhou, S., Loke, T., Izaac, J.A., Wang, J.B.: Quantum fourier transform in computational basis. Quantum Inf. Process. **16**(3), 82 (2017)
9. Nam, Y., Su, Y., Maslov, D.: Approximate quantum fourier transform with O(nlogn) T-gates. arXiv:1803.04933 (2018)
10. Childs, A.M., Van Dam, W.: Quantum algorithms for algebraic problems. Rev. Mod. Phys. **82**(1), 1 (2010)
11. Simon, D.R.: On the power of quantum computation. SIAM J. Comput. **26**(5), 1474–1483 (1997)
12. Dong, X., Wang, X.: Quantum key-recovery attack on feistel structures. Sci. China Inf. Sci. **61**(10), 102501 (2018)
13. Mosca, M.: Quantum algorithms. arXiv:0808.0369v1 (2009)
14. Wagstaff, S.S.: The joy of factoring, vol. 68. American Mathematical Society, Providence (2013)

15. Lenstra, A.K., Lenstra Jr., H.W., Manasse, M.S., Pollard, J.M.: The number field sieve. In: Proceedings of the Twenty-Second Annual ACM Symposium on Theory of Computing, pp. 564–572 (1990)
16. Wei, S.J., Xin, T., Long, G.L.: Erratum to: Efficient universal quantum channel simulation in IBM's cloud quantum computer. Sci. China Phys. Mech. Astron. **62**(1), 70311 (2019)
17. Huang, H.L., Zhao, Y.W., Li, T., Li, F.G., Du, Y.T., Fu, X.Q., Zhang, S., Wang, X., Bao, W.S.: Homomorphic encryption experiments on IBMs cloud quantum computing platform. Front. Phys. **12**(1), 120305 (2017)
18. Xu, N., Zhu, J., Lu, D., Zhou, X., Peng, X., Du, J.: Quantum factorization of 143 on a dipolar-coupling nuclear magnetic resonance system. Phys. Rev. Lett. **108**(13), 130501 (2012)
19. Shor, P.W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Rev. **41**(2), 303–332 (1999)
20. Vandersypen, L.M., Steffen, M., Breyta, G., Yannoni, C.S., Sherwood, M.H., Chuang, I.L.: Experimental realization of Shor's quantum factoring algorithm using nuclear magnetic resonance. Nature. **414**(6866), 883–887 (2001)
21. Martin-Lopez, E., Laing, A., Lawson, T., Alvarez, R., Zhou, X.Q., O'brien, J.L.: Experimental realization of Shor's quantum factoring algorithm using qubit recycling. Nat. Photonics **6**(11), 773 (2012)
22. Geller, M.R., Zhou, Z.: Factoring 51 and 85 with 8 qubits. Sci. Rep. **3**(10), 3023 (2013)
23. Gidney, C.: Factoring with n+2 clean qubits and n-1 dirty qubits. arXiv:1706.07884 (2017)
24. Beckman, D., Chari, A.N., Devabhaktuni, S., Preskill, J.: Efficient networks for quantum factoring. Phys. Rev. A **54**(2), 1034–1063 (1996)
25. Vedral, V., Barenco, A., Ekert, A.: Quantum networks for elementary arithmetic operations. Phys. Rev. A **54**(1), 147–153 (1996)
26. Beauregard, S.: Circuit for Shor's algorithm using 2n+3 qubits. arXiv:quant-ph/0205095 (2002)
27. Takahashi, Y., Kunihiro, N.: A quantum circuit for Shor's factoring algorithm using 2n+2 qubits. Quantum Inf. Comput. **6**(2), 184–192 (2006)
28. Häner, T., Roetteler, M., Svore, K. M.: Factoring using 2n+2 qubits with Toffoli based modular multiplication. arXiv:1611.07995 (2016)
29. Albash, T., Lidar, D.A.: Adiabatic quantum computing. Rev. Mod. Phys. **90**(1), 015002 (2016)
30. Farhi, E., Goldstone, J., Gutmann, S., Lapan, J., Lundgren, A., Preda, D.: A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem. Science **292**(5516), 472–476 (2001)
31. Wang, T., Zhang, Z., Xiang, L., Gong, Z., Wu, J., Yin, Y.: Simulating a topological transition in a superconducting phase qubit by fast adiabatic trajectories. Sci. China Phys. Mech. Astron. **61**(4), 047411 (2018)
32. Burges, C.J.: Factoring as optimization. Microsoft Research MSR-TR-200 (2002)
33. Peng, W., Wang, B., Hu, F., Wang, Y., Fang, X., Chen, X., Wang, C.: Factoring larger integers with fewer qubits via quantum annealing with optimized parameters. Sci. China Phys. Mech. Astron. **62**(6), 60311 (2019)
34. Pal, S., Moitra, S., Anjusha, V.S., Kumar, A., Mahesh, T.S.: Hybrid scheme for factorisation: factoring 551 using a 3-qubit NMR quantum adiabatic processor. Pramana **92**(2), 26 (2019)
35. Peng, X., Liao, Z., Xu, N., Qin, G., Zhou, X., Suter, D., Du, J.: Quantum adiabatic algorithm for factorization and its experimental implementation. Phys. Rev. Lett. **101**(22), 220405 (2008)
36. Dridi, R., Alghassi, H.: Prime factorization using quantum annealing and computational algebraic geometry. Sci. Rep. **7**, 43048 (2017)
37. Hen, I.: Realizable quantum adiabatic search. EPL (Europhys. Lett.) **118**(3), 30003 (2017)
38. Li, H., Liu, Y., Long, G.: Experimental realization of single-shot nonadiabatic holonomic gates in nuclear spins. Sci. China Phys. Mech. Astron. **60**(8), 80311 (2017)
39. Wang, C., Zhang, H.: Impact of commercial quantum computer on cryptography. Inf. Secur. Commun. Priv. **2**, 31 (2012)
40. Li, Z., Dattani, N.S., Chen, X., Liu, X., Wang, H., Tanburn, R., Du, J.: High-fidelity adiabatic quantum computation using the intrinsic Hamiltonian of a spin system: application to the experimental factorization of 291311. arXiv:1706.08061 (2017)
41. Jiang, S., Britt, K.A., McCaskey, A.J., Humble, T.S., Kais, S.: Quantum annealing for prime factorization. Sci. Rep. **8**, 17667 (2018)
42. Proos, J., Zalka, C.: Shor's discrete logarithm quantum algorithm for elliptic curves. Quantum Inf. Comput. **3**(4), 317–344 (2003)
43. Brassard, G. (ed.): Advances in Cryptology-CRYPTO'89: Proceedings, vol. 435. Springer, Berlin (1995)

44. Maslov, D., Mathew, J., Cheung, D., Pradhan, D.K.: An $O(m^2)$-depth quantum algorithm for the elliptic curve discrete logarithm problem over $\mathrm{GF}(2^m)^a$. Quantum Inf. Comput. **9**(7), 610–621 (2009)
45. Myasnikov, A.D., Ushakov, A.: Quantum algorithm for discrete logarithm problem for matrices over finite group rings. Groups Complex. Cryptol. **6**(1), 31–36 (2014)
46. Childs, A.M., Ivanyos, G.: Quantum computation of discrete logarithms in semigroups. J. Math. Cryptol. **8**(4), 405–416 (2014)
47. Banin, M., Tsaban, B.: A reduction of semigroup DLP to classic DLP. Des. Codes Cryptogr. **81**(1), 75–82 (2016)
48. Ekera, M.: On post-processing in the quantum algorithm for computing short discrete logarithms. IACR Cryptology ePrint Archive, p. 1122 (2017)
49. Ekera, M.: Revisiting shor's quantum algorithm for computing general discrete logarithms. arXiv:1905.09084 (2019)
50. Moldovyan, A.A., Moldovyan, N.A.: Post-quantum signature algorithms based on the hidden discrete logarithm problem. Comput. Sci. J. Mold. **26**(3), 301–313 (2018)
51. Wang, F.: The hidden subgroup problem. arXiv:1008.0010 (2010)
52. Kitaev, A.Y.: Quantum measurements and the Abelian stabilizer problem. arXiv:quant-ph/9511026 (1995)
53. Boneh, D., Lipton, R.J.: Quantum cryptanalysis of hidden linear functions. In: Annual International Cryptology Conference, pp. 424–437 (1995)
54. Brassard, G., Hoyer, P.: An exact quantum polynomial-time algorithm for Simon's problem. In: Proceedings of the Fifth Israeli Symposium on Theory of Computing and Systems, pp. 12–23 (1997)
55. Jozsa, R.: Quantum algorithms and the Fourier transform. Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci. **454**(1969), 323–337 (1998)
56. Mosca, M., Ekert, A.: The hidden subgroup problem and eigenvalue estimation on a quantum computer. In: NASA International Conference on Quantum Computing and Quantum Communications, pp. 174–188 (1998)
57. Mosca, M.: Quantum computer algorithms. PhD thesis, University of Oxford (1999)
58. Jozsa, R.: Quantum factoring, discrete logarithms, and the hidden subgroup problem. Comput. Sci. Eng. **3**(2), 34 (2001)
59. Cheung, K. K., Mosca, M.: Decomposing finite abelian groups. arXiv:cs/0101004 (2001)
60. Damgård, I.: QIP note: on the quantum Fourier transform and applications. Published on https://users-cs.au.dk/~ivan/fourier.pdf (2004). Accessed 26 June 2019
61. Van Dam, W., Hallgren, S., Ip, L.: Quantum algorithms for some hidden shift problems. In: Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms, Baltimore, Maryland, USA, pp. 489–498 (2003)
62. Van Dam, W., Hallgren, S., Ip, L.: Quantum algorithms for some hidden shift problems. SIAM J. Comput. **36**(3), 763–778 (2006)
63. Van Dam, W.: Quantum algorithms for weighing matrices and quadratic residues. Algorithmica. **34**(4), 413–428 (2002)
64. Van Dam, W., Hallgren, S.: Efficient quantum algorithms for shifted quadratic character problems. arXiv:quant-ph/0011067 (2000)
65. Childs, A.M., Schulman, L.J., Vazirani, U.V.: Quantum algorithms for hidden nonlinear structures. In: 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07), pp. 395–404 (2007)
66. Rötteler, M.: Quantum algorithms for highly non-linear boolean functions. In: Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2010, Austin, Texas, USA, pp. 448–457 (2010)
67. Gavinsky, D., Roetteler, M., Roland, J.: Quantum algorithm for the Boolean hidden shift problem. In: International Computing and Combinatorics Conference, pp. 158–167. Springer, Berlin (2011)
68. Ozols, M., Roetteler, M., Roland, J.: Quantum rejection sampling. ACM Trans. Comput. Theory (TOCT) **5**(3), 1–33 (2013)
69. Ettinger, M., Høyer, P.: On quantum algorithms for non-commutative hidden subgroups. Adv. Appl. Math. **25**(3), 239–251 (2000)
70. Kuperberg, G.: Another subexponential-time quantum algorithm for the dihedral hidden subgroup problem. arXiv:1112.3333 (2011)
71. Roetteler, M.: Quantum algorithms for abelian difference sets and applications to dihedral hidden subgroups. arXiv:1608.02005 (2016)

72. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing, pp. 197–206 (2008)
73. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. J. ACM (JACM). **56**(6), 34 (2009)
74. Shpilrain, V., Ushakov, A.: Thompsons group and public key cryptography. In: International Conference on Applied Cryptography and Network Security, pp. 151–163 (2005)
75. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, pp. 84–93 (2005)
76. Kuperberg, G.: A subexponential-time quantum algorithm for the dihedral hidden subgroup problem. SIAM J. Comput. **35**(1), 170–188 (2005)
77. Li, F., Bao, W., Fu, X., Zhang, Y., Li, T.: A reduction from LWE problem to dihedral coset problem. arXiv:1305.3769 (2013)
78. Eldar, L., Shor, P.W.: An efficient quantum algorithm for a variant of the closest lattice-vector problem. arXiv:1611.06999 (2016)
79. Eldar, L., Shor, P. W.: A discrete Fourier transform on lattices with quantum applications. arXiv:1703.02515 (2017)
80. Brakerski, Z., Kirshanova, E., Stehlé, D., Wen, W.: Learning with errors and extrapolated dihedral cosets. In: IACR International Workshop on Public Key Cryptography, pp. 702–727 (2018)
81. Grover, L., Rudolph, T.: Creating superpositions that correspond to efficiently integrable probability distributions. arXiv: quant-ph/0208112 (2002)