数理解析研究所講究録 1205 巻 2001 年 53-58

量子公開鍵暗号とその改良

田中 圭介 (Keisuke Tanaka)* 岡本龍明 (Tatsuaki Okamoto)*

Abstract

At CRYPTO2000, Okamoto, Tanaka, and Uchiyama [14] proposed a concept of *quantum public-key cryptosystems.* They suggested the public-key cryptosystems employing *quantum* Turing machines and *classical* (non-quantum) channels. In addition to the concept, they proposed a particular scheme for quantum public-key encryption based on knapsack problems with algebraic number fields. As described in their paper, the information rate of their scheme is not greater than $\frac{1}{2}$ and never reaches 1, even if they use the quadratic number field whose degree is at least 2. This scheme has the shortcomings on low information rates, and might have weakness. This paper proposes a scheme which increases the security and whose information rate is asymptotically 1, which can make up for the shortcomings of the Okamoto–Tanaka–Uchiyama scheme.

1 Introduction

The concept of public-key cryptosystems was introduced by Diffie and Hellman [6] based on classical Turing machines and classical channels. Since then, it has been widely studied by proposing particular schemes and/or proving the security of these schemes (e.g., [7]).

However, a new model of computing, a quantum Turing machine has been investigated since the 1980's. It seems reasonable to consider such a computing model since our world behaves quantum mechanically. Several recent results provide informal evidence that quantum Turing machines violate the feasible computation version of Church's thesis [5, 19, 18]. The most successful result in this field is Shor's (probabilistic) polynomial time algorithms for factoring and finding discrete logarithms [18].

Shor's result, in particular, greatly impacted practical public-key cryptosystems such as RSA, (multiplicative group/elliptic curve versions of) Diffie–Hellman, and ElGamal schemes, since almost all practical public-key cryptosystems are constructed on the factoring and/or discrete logarithm problem. Therefore, when quantum Turing machines are realized, we will lose almost all practical public-key cryptosystems.

Recently, at CRYPTO2000, Okamoto, Tanaka, and Uchiyama [14] proposed a possible solution for this problem, i.e., a concept of *quantum public-key cryptosystems.* They suggested the public-key cryptosystems employing *quantum* Turing machines and *classical* (non-quantum) channels. In addition to the concept, they proposed a particular scheme for quantum public-key encryption based on knapsack problems with algebraic number fields.

They claimed this scheme is secure by estimating the density and information rate on their scheme, and by considering possible attacks against their scheme. As described in their paper, the information rate of their scheme is not greater than $\frac{1}{2}$ and never reaches 1, even if they use the quadratic number field, whose degree is at least 2. This scheme has the shortcomings on low information rates, and might have weakness.

However, if the underlying algebraic number field is restricted to the field of rational integers or of degree 1, the information rate of their scheme is asymptotically 1. But as mentioned in Appendix 1 of their paper, they do not recommend to use it, since it has no freedom on choice of fields and is too simple to use even if no attack has been found in this case.

TT 情報流通プラットフォーム研究所 (NTT Information Sharing Platform Laboratories), 〒 239-0847 神奈川
の丘 1-1, Room 612A. `(keisuke,okamoto)@isl.ntt.co.jp`

In this paper, we propose a scheme whose information rate is asymptotically 1, which increases the security. Our scheme uses an iteration technique, and is an extension of the rational integer version of the Okamoto–Tanaka–Uchiyama scheme, and can make up for the shortcomings of the Okamoto–Tanaka–Uchiyama scheme.

2 Related works

Our scheme is a combination of the rational integer version of the Okamoto–Tanaka–Uchiyama scheme and the Merkle–Hellman iterated knapsack scheme based on super-increasing sequences [10]. The Okamoto–Tanaka–Uchiyama scheme is closely related to the Merkle–Hellman multiplicative trapdoor knapsack scheme [10], the Merkle–Hellman iterated knapsack scheme based on super-increasing sequences [10], and the Chor–Rivest scheme [3].

Even if no attack has been found for the rational integer version of the Okamoto–Tanaka–Uchiyama scheme, its related scheme has attacks as follows. The Merkle–Hellman multiplicative scheme was broken by Odlyzko [12] under some condition and has also been broken due to its low-density (asymptotically its density is zero) The Merkle–Hellman iterated scheme was also broken by Adleman [1]. Typical realizations of the Chor–Rivest scheme were cryptanalyzed by Schnorr–Hoerner [16] and Vaudenay [20], because of the known low cardinality of the subset-sum and the symmetry of the trapdoor information.

Also notice that the Merkle–Hellman iterated knapsack scheme based on super-increasing sequences was broken by Adleman [1] using a reduction in order to apply the super-increasing sequence attack.

Quantum Public-Key Encryption

Let us mention the model of quantum public-key encryption proposed in [14].

Definition 1 (Okamoto–Tanaka–Uchiyama) *A* quantum public-key encryption *scheme consists of three probabilistic polynomial-time quantum Turing machines, $(G, E, D)$, as follows:*

1. *$G$ is a probabilistic polynomial-time quantum Turing machine for generating keys. That is, $G$, on input 1^n, outputs $(e, d)$ with overwhelming probability in $n$ (taken over the classical coin flips and quantum observation of $G$), where $e$ is a public-key, $d$ is a secret-key, and $n$ is a security parameter. (W.o.l.g., we suppose $|e| = |d| = n$.)*
2. *$E$ is an encryption function that produces ciphertext $c$, and $D$ is a decryption function. For every message $m$ of size $|m| = n$, every polynomial $poly$, and all sufficiently large $n$,*

   $$\Pr[D(E(m, e), d) = m] > 1 - 1/poly(n).$$

   The probability is taken over the (classical) coin flips and quantum observation of $(G, E, D)$.

Note that all variables in this definition are classical strings, and no quantum channel between any pair of parties is assumed.

3 Proposed Scheme

Our scheme is a combination of the rational integer version of the Okamoto–Tanaka–Uchiyama scheme and the Merkle–Hellman iterated knapsack scheme based on super-increasing sequences [10].

3.1 Proposed scheme

Key generation

1. Fix size parameters $n$, $k$ from $\mathbb{Z}$.
2. Randomly choose a prime $p$, a generator $g$ of the group $(\mathbb{Z}/p\mathbb{Z})^\times$, and $n$ co-primes $p_1, \ldots, p_n \in \mathbb{Z}/p\mathbb{Z}$ such that $\prod_{j=1}^{k} p_{i_j} < p$ for any subset $\{p_{i_1}, p_{i_2}, \ldots, p_{i_k}\}$ from $\{p_1, p_2, \ldots, p_n\}$.
3. Use Shor's algorithm for finding discrete logarithms to get integers $a_1, \ldots, a_n \in \mathbb{Z}/(p-1)\mathbb{Z}$ satisfying $p_i \equiv g^{a_i} \pmod{p}$, for each $1 \leq i \leq n$.

4. Randomly choose a integer $d \in \mathbb{Z}/(p-1)\mathbb{Z}$.
5. Compute $b'_i = (a_i + d) \bmod (p-1)$, for each $1 \le i \le n$.
6. Randomly choose a prime p' such that $p' \ge \sum_{i=1}^{n} b'_i$.
7. Randomly choose integers c' and d' such that $c', d' \in \mathbb{Z}/p'\mathbb{Z}$.
8. Compute $b_i = c'b'_i + d' \bmod p'$, for each $1 \le i \le n$.
9. The public key is $(n, k, b_1, b_2, \ldots, b_n)$, and the secret key is $(g, c', d, d', p, p', p_1, p_2, \ldots, p_n)$.

Encryption

1. Fix the length of plaintext M to $\lfloor \log \binom{n}{k} \rfloor$.
2. Encode M into a binary string $m = (m_1, m_2, \ldots, m_n)$ of length n and Hamming weight k (i.e., having exactly k 1's) as follows:
   (a) Set $l \leftarrow k$.
   (b) For i from 1 to n do the following:
   If $M \ge \binom{n-i}{l}$ then set $m_i \leftarrow 1$, $M \leftarrow M - \binom{n-i}{l}$, $l \leftarrow l-1$. Otherwise, set $m_i \leftarrow 0$. (Notice that $\binom{l}{0} = 1$ for $l \ge 0$, and $\binom{0}{l} = 0$ for $l \ge 1$.)
3. Compute the ciphertext c by $c = \sum_{i=1}^{n} m_i b_i$.

Decryption

1. Compute $r' = (c - kd')/c' \bmod p'$.
2. Compute $r = (r' - kd) \bmod (p-1)$.
3. Compute $u = g^r \bmod p$.
4. Find the factors of u. If p_i is a factor, then set $m_i \leftarrow 1$. Otherwise, $m_i \leftarrow 0$.
5. Decode m to the plaintext M as follows:
   (a) Set $M \leftarrow 0$, $l \leftarrow k$.
   (b) For i from 1 to n do the following:
   If $m_i = 1$, then set $M \leftarrow M + \binom{n-i}{l}$ and $l \leftarrow l-1$.

3.2 Correctness and remarks

1 [Decryption] We show that the decryption works. We observe that

$$\begin{aligned} r' &\equiv (c - kd')/c' \equiv ((\sum_{i=1}^{n} m_i b_i) - kd')/c' \pmod{p'} \\ &\equiv \sum_{i=1}^{n} m_i b'_i \pmod{p'} \\ &= \sum_{i=1}^{n} m_i b'_i, \end{aligned}$$

and

$$\begin{aligned} u &\equiv g^r \equiv g^{r'-kd} \equiv g^{(\sum_{i=1}^{n} m_i b'_i) - kd} \pmod{p} \\ &\equiv g^{\sum_{i=1}^{n} m_i a_i} \pmod{p} \\ &\equiv \prod_{i=1}^{n} (g^{a_i})^{m_i} \pmod{p} \\ &\equiv \prod_{i=1}^{n} p_i^{m_i} \pmod{p} \\ &= \prod_{i=1}^{n} p_i^{m_i}. \end{aligned}$$

Since we choose n co-primes $p_1, \ldots, p_n$, a product of a subset in $p_1, \ldots, p_n$ can be uniquely factorized. Thus, a ciphertext is uniquely decrypted.

2 [Density and information rate] We here estimate the density and information rate of our scheme (see Section 3.3 for the definition of density and information rate). By the prime number theorem, we have the following relations: $|b_i| \approx |p|$ ($|x|$ is the size of x), $k \times |p_i| \approx |p|$, and $|p_i| \approx \log n$. Accordingly, ignoring minor terms, we obtain $|b_i| \approx k \log n$. Hence the density D of our scheme is estimated by $\frac{n}{k \log n}$, and the information rate R by $\frac{\binom{n}{k}}{k \log n} \approx \frac{k \log n - k \log k}{k \log n}$. If we choose $k = 2^{(\log n)^c}$ for a constant $c < 1$, the rate R is asymptotically 1, and density D is asymptotically ∞. Notice again that the information rate of the Okamoto–Tanaka–Uchiyama scheme is asymptotically 1/2 and never reaches 1.

3 [Shor's algorithm] Key generation uses Shor's algorithm for finding discrete logarithms. This is known to be a polynomial time algorithm, thus it fits our scheme.

4 [Knapsack problem] Our scheme is based on the knapsack problem, which is a typical NP-hard problem. Although Shor's result demonstrates the positive side of the power of quantum Turing machines, the limitation of the power of quantum Turing machines is also known. Bennett, Bernstein, Brassard, and Vazirani [2] show that relative to an oracle chosen uniformly at random, with probability 1, class NP cannot be solved on a quantum Turing machine in time $o(2^{n/2})$. Although this result does not rule out the possibility that NP $\subseteq$ BQP, many researchers believe that it is hard to find a probabilistic polynomial-time algorithm to solve an NP-complete problem even in the quantum Turing machine model, or conjecture that NP $\not\subseteq$ BQP.

5 [Coding] We next mention about the encoding scheme used in the encryption and decryption. This scheme is well known in literature on combinatorics (see [4]). This scheme is also employed by the Chor–Rivest cryptosystem and Okamoto–Tanaka–Uchiyama cryptosystem.) This encoding scheme is used mainly for avoiding the low-density attacks.

6 [Complexity] Here we mention about the time complexity needed for the key generation as well as the encryption and decryption. The most difficult part in the key generation is the computation of discrete logarithms at line 3. In particular, we compute n discrete logarithms $a_1, \ldots, a_n$ in the field $\mathbb{Z}/p\mathbb{Z}$. For the encryption, once we get the encoded string by line 2 in the encryption, all we need to do is to add k integers, each smaller than p. For the decryption, we perform the modular exponentiation $g^r \bmod p$ in line 3. This dominates the running time of the decryption. Raising a generator g to a power in the range up to p takes at most $2 \times \log p$ modular multiplications by using a standard multiplication technique. Notice that only the key generation (i.e., off-line stage) requires quantum mechanism, and the encryption and decryption (i.e., on-line stage) are very efficient with classical mechanisms.

3.3 Security Consideration

We provide an initial analysis for the security of our scheme by considering several possible attacks.

We can use quantum computers also for attacks in our setting. As far as we know, despite recent attempts at designing efficient quantum algorithms for problems where no efficient classical probabilistic algorithm is known, all known such quantum algorithms are for some special cases of the hidden subgroup problem [11]. Let f be a function from a finitely generated group G_1 to a finite set such that f is constant on the cosets of a subgroup G_2. Given a way of computing f, a hidden subgroup problem is to find G_2 (i.e., a generating set for G_2). The problems of factoring and finding discrete logarithms can be formulated as instances of the hidden subgroup problems.

There is also a result by Grover [8] for database search. He shows that the problem of finding an entry with the target value can be searched in $O(\sqrt{N})$ time, where N is the number of entries in the database. This result implies NP-complete problems can be solved in $O(\sqrt{N})$ time.

However, if we do not put a structure in the database, i.e., we need to ask oracles for the contents in the database, it is known that we cannot make algorithms whose time complexity is $o(\sqrt{N})$. Thus, it is widely believed that NP-complete problems cannot be solved in polynomial time even with quantum computers.

3.3.1 Finding secret keys from public keys

Recall that we have the public key $(n, k, b_1, b_2, \ldots, b_n)$, and the secret key $(g, c', d, d', p, p', p_1, p_2, \ldots, p$

First, in a passive attack setting, the attacker has only information on the public key. The information on n and k only exposes problem size.

Second, we could guess a subset of p_i's, since we have chosen roughly n primes out of cn smallest primes, where c is a constant. Suppose we find a subset of p_i's. In order to use them in the attack by Odlyzko for multiplicative knapsack cryptosystems [12], the size of the subset must be fairly large. In addition, it is necessary to find the correspondences between the elements of the subset and b_i's. Here we observe that b_i's seem to be random because of the discrete-log relation in our function. Thus, it seems impossible for any reasonable relation between public keys and private keys to be made without knowing g, c', d, d', p, p', so the critical attacks of directly finding public keys from secret keys seem to be difficult.

The Okamoto–Tanaka–Uchiyama cryptosystem does not employ secret parameters c', d' and p'. We claim that these parameters can increase security on our scheme. Adleman's attack is known to be effective for the Merkle–Hellman iterated knapsack scheme based on super-increasing sequences [1]. His attack uses simultaneous inequations to get over additional parameters used in iterated stages, but it still uses the weak property on super-increasing sequences (see also [17]). In contrast to the super-increasing case, our scheme does not suffer from this attack, and can add three secret parameters in order to increase the security.

3.3.2 Finding plaintexts from ciphertexts

For many knapsack-type cryptosystems, the low-density attack is known to be effective. Thus, it might be effective against our scheme. A low-density attack finds plaintexts from ciphertexts by directly solving feasible solutions to the subset-sum problems that the cryptosystem is based on.

The subset-sum problem is, given positive rational integers c and $a_1, \ldots, a_n$ to solve the equation $c = \sum_{i=1}^{n} m_i a_i$ with each $m_i \in \{0, 1\}$. Let $a = \{a_1, \ldots, a_n\}$. The density $d(a)$ of a knapsack system is defined to be $d(a) = \frac{n}{\log(\max_i a_i)}$. Density is an approximate measure of the information rate for knapsack-type cryptosystems. According to Orton [15], the shortest vector in a lattice solves almost all subset-sum problems whose density is less than 0.9408 with reasonable problem size. If we choose appropriate parameters for our scheme, the density is not less than 1 (see Section 3.2).

It is known that the algorithms for finding the shortest vector in a lattice can be used to find the solutions to the subset-sum problems. The LLL algorithm plays an important role in this kind of attack. However, it is not known that the LLL algorithm can be improved with the quantum mechanism. Incidentally, as far as we know, for any approximation algorithm, it is not known that its approximation ratio can be improved by the addition of the quantum mechanism.

Information rate R is defined to be $\frac{\log |M|}{N}$, where $|M|$ is the size of message space and N is the number of bits in a cipher text. If we select appropriate parameters, the information rate of our scheme is asymptotically 1 (see Section 3.2).

The subset-sum problem which our scheme is based on is a typical NP-hard problem. Notice again that it is widely believed that NP-complete problems cannot be solved in polynomial time even with quantum computers. Thus, our scheme with appropriate parameters does not seem to be open to successful crucial attacks that find plaintexts from ciphertexts even if quantum computers are used.

4 Concluding Remarks

Our scheme has two possible extensions. One is a generalization with an arbitrary number of iterations. This generalization does not increase the density or information rate of the scheme. Another direction is an extension from the field of rational integers to algebraic number fields similar to [14]. However, this generalization decreases the information rate of the scheme. Our scheme can be also employed to realize standard (non-quantum) public-key encryption based on conventional (non-quantum) algorithms [13]. We utilize the Chinese remainder theorem technique in the key generation to compute the discrete logarithms very efficiently even if conventional (non-quantum) algorithms are used.

References

[1] ADLEMAN, L. On Breaking the Iterated Merkle–Hellman public key cryptosystem. In *Advances in Cryptology—CRYPTO'82* (1982), pp. 303–308.

[2] BENNETT, C. H., BERNSTEIN, E., BRASSARD, G., AND VAZIRANI, U. Strengths and weaknesses of quantum computing. *SIAM J. Comput. 26*, 5 (Oct. 1997), 1510–1523.

[3] CHOR, B., AND RIVEST, R. L. A knapsack-type public key cryptosystem based on arithmetic in finite fields. *IEEE Trans. on Information Theory 34* (1988), 901–909.

[4] COVER, T. M. Enumerative source encoding. *IEEE Trans. on Information Theory IT-19* (1973), 901–909.

[5] DEUTSCH, D., AND JOZSA, R. Rapid solution of problems by quantum computation. *Proc. R. Soc. Lond. A 439* (1992), 553–558.

[6] DIFFIE, W., AND HELLMAN, M. New directions in cryptography. *IEEE Trans. on Information Theory IT-22*, 6 (1976), 644–654.

[7] GOLDREICH, O. On the foundations of modern cryptography. In *Advances in Cryptology—CRYPTO '97* (17–21 Aug. 1997), B. S. Kaliski Jr., Ed., vol. 1294 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 46–74.

[8] GROVER, L. K. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing* (Philadelphia, Pennsylvania, 22–24 May 1996), pp. 212–219.

[9] GUILLOU, L. C., AND QUISQUATER, J.-J., Eds. *Advances in Cryptology—EUROCRYPT 95* (21–25 May 1995), vol. 921 of *Lecture Notes in Computer Science*, Springer-Verlag.

[10] MERKLE, R. C., AND HELLMAN, M. E. Hiding information and signatures in trapdoor knapsacks. *IEEE Trans. on Information Theory 24* (1978), 525–530.

[11] MOSCA, M., AND EKERT, A. The Hidden Subgroup Problem and Eigenvalue Estimation on a Quantum Computer. quant-ph/9903071, (1999).

[12] ODLYZKO, A. M. Cryptanalytic attacks on the multiplicative knapsack cryptosystem and on Shamir's fast signature scheme. *IEEE Trans. on Information Theory IT-30* (1984), 594–601.

[13] OKAMOTO, T., AND TANAKA, K. A New Approach to Knapsack Cryptosystems. manuscript (2000).

[14] OKAMOTO, T., TANAKA, K., AND UCHIYAMA, S. Quantum Public-Key Cryptosystems. In *Advances in Cryptology—CRYPTO2000* (2000), pp. 147–165.

[15] ORTON, G. A Multiple-Iterated Trapdoor for Dense Compact Knapsacks. In *Advances in Cryptology—EUROCRYPT'94* (1994), pp. 112–130.

[16] SCHNORR, C. P., AND HÖRNER, H. H. Attacking the Chor–Rivest cryptosystem by improved lattice reduction. In Guillou and Quisquater [9], pp. 1–12.

[17] SHAMIR, A. A Polynomial Time Algorithm for Breaking the Basic Merkle–Hellman Cryptosystem. *IEEE Trans. Inf. Theory IT-30*, 5 (1984), 145–152.

[18] SHOR, P. W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput. 26*, 5 (Oct. 1997), 1484–1509.

[19] SIMON, D. R. On the power of quantum computation. *SIAM J. Comput. 26*, 5 (Oct. 1997), 1474–1483.

[20] VAUDENAY, S. Cryptanalysis of the Chor–Rivest cryptosystem. In *Advances in Cryptology—CRYPTO'98* (1998), pp. 243–256.