Computational Indistinguishability between Quantum States and Its Cryptographic Application*

Akinori Kawachi[1] Takeshi Koshiba[2] Harumichi Nishimura[3] Tomoyuki Yamakami[4]

[1] Department of Mathematical and Computing Sciences, Tokyo Institute of Technology
kawachi@is.titech.ac.jp
[2] Division of Mathematics, Electronics and Informatics
Graduate School of Science and Engineering, Saitama University
koshiba@tcs.ics.saitama-u.ac.jp
[3] Department of Mathematics and Information Sciences
Graduate School of Science, Osaka Prefecture University
hnishimura@mi.s.osakafu-u.ac.jp
[4] ERATO-SORST Quantum Computation and Information Project
Japan Science and Technology Agency
yamakami@qci.jst.go.jp

Abstract

We introduce a computational problem of distinguishing between two specific quantum states as a new cryptographic problem to design a quantum cryptographic scheme that is "secure" against any polynomial-time quantum adversary. Our problem $\mathrm{QSCD}_{\mathit{ff}}$ is to distinguish between two types of random coset states with a hidden permutation over the symmetric group of finite degree. This naturally generalizes the commonly-used distinction problem between two probability distributions in computational cryptography. As our major contribution, we show three cryptographic properties: (i) $\mathrm{QSCD}_{\mathit{ff}}$ has the trapdoor property; (ii) the average-case hardness of $\mathrm{QSCD}_{\mathit{ff}}$ coincides with its worst-case hardness; and (iii) $\mathrm{QSCD}_{\mathit{ff}}$ is computationally at least as hard in the worst case as the graph automorphism problem. These cryptographic properties enable us to construct a quantum public-key cryptosystem, which is likely to withstand any chosen plaintext attack of a polynomial-time quantum adversary. We further discuss a generalization of $\mathrm{QSCD}_{\mathit{ff}}$, called QSCD_{cyc}, and introduce a multi-bit encryption scheme relying on the cryptographic properties of QSCD_{cyc}.

Keywords: quantum cryptography, computational indistinguishability, trapdoor property, worst-case/average-case equivalence, graph automorphism problem, quantum public-key cryptosystem.

*The preliminary version appeared in the Proceedings of EUROCRYPT 2005, Lecture Notes in Computer Science, Vol.3994, pp.268–284, Aahus, Denmark, May 22–26, 2005.

1 Introduction

Since Diffie and Hellman [16] first used a computationally intractable problem to design a key exchange protocol, computational cryptography has been extensively studied; especially, a number of practical cryptographic systems (e.g., public-key cryptosystems (PKCs), bit commitment schemes (BCSs), pseudorandom generators, and digital signature schemes) have been proposed under popular intractability assumptions, such as the hardness of the integer factorization problem (IFP) and the discrete logarithm problem (DLP), for which no efficient classical (i.e., deterministic or probabilistic) algorithm have been found. Using the power of quantum computation, however, we can efficiently solve various number-theoretic problems, including IFP (and the quadratic residuosity problem) [45], DLP (and the Diffie-Hellman problem) [11, 28, 45], and the principal ideal problem [23]. Therefore, a *quantum adversary* (i.e., an adversary who runs a quantum computer) can easily break the cryptosystems whose security proofs heavily rely on the computational hardness of these problems.

Fighting against such a powerful quantum adversary, a new area of cryptography, so-called *quantum cryptography*, has emerged in the past two decades. In 1984, Bennett and Brassard [8] proposed a *quantum key distribution scheme* via a quantum communication channel. Its unconditional security was later proven by Mayers [35]. Nonetheless, as Mayers [34] and Lo and Chau [32] independently demonstrated, quantum mechanics cannot make all cryptographic schemes information-theoretically secure as we had hoped; in particular, they proved that no quantum BCS can be both concealing and binding unconditionally. Therefore, "computational" approaches are still important in quantum cryptography. Along this line, a number of quantum cryptographic properties have been discussed from the complexity-theoretic point of view [1, 13, 14, 15, 17, 41].

A quantum computer is known to be capable of breaking the RSA cryptosystem and other well-known classical cryptosystems. It is therefore imperative to discover computationally-hard problems from which a secure quantum cryptosystem is constructed against any polynomial-time quantum adversary. For instance, the subset sum (knapsack) problem and the shortest vector problem are a basis to knapsack-based cryptosystems [26, 41] as well as lattice-based cryptosystems [4, 42]. Since it is currently unknown whether these problems withstand any attack of quantum adversaries, we need to continue searching for better intractable problems that can guard their associated quantum cryptosystems against any powerful quantum adversary.

This paper introduces the new notion of *computational indistinguishability between quantum states*, which generalizes the classical indistinguishability notion between two probability distributions [9, 19, 48]. In particular, we present a distinction problem, called $\mathrm{QSCD}_{\mathit{ff}}$ (quantum state computational distinction with fully flipped permutations), between specific ensembles of quantum states. $\mathrm{QSCD}_{\mathit{ff}}$ enjoys remarkable cryptographic properties as a building block of a secure quantum cryptosystem.

Definition 1.1 The *advantage* of a polynomial-time quantum algorithm $\mathcal{A}$ that distinguishes between two ensembles $\{\rho_0(l)\}_{l\in\mathbb{N}}$ and $\{\rho_1(l)\}_{l\in\mathbb{N}}$ of quantum states is the function $\delta_{\mathcal{A}}(l)$ defined as:

$$\delta_{\mathcal{A}}(l) = \left| \Pr_{\mathcal{A}}[\mathcal{A}(\rho_0(l)) = 1] - \Pr_{\mathcal{A}}[\mathcal{A}(\rho_1(l)) = 1] \right|$$

for two l-qubit quantum states $\rho_0(l)$ and $\rho_1(l)$, where the subscript $\mathcal{A}$ means that any output of

$\mathcal{A}$ is determined by measuring the final state of $\mathcal{A}$ in the standard computational basis. We say that two ensembles $\{\rho_0(l)\}_{l\in\mathbb{N}}$ and $\{\rho_1(l)\}_{l\in\mathbb{N}}$ are *computationally indistinguishable* if the advantage $\delta_{\mathcal{A}}(l)$ is negligible for any polynomial-time quantum algorithm $\mathcal{A}$; namely, for any polynomial p, any polynomial-time quantum algorithm $\mathcal{A}$, and any sufficiently large number l, it holds that $\delta_{\mathcal{A}}(l) < 1/p(l)$. The distinction problem between $\{\rho_0(l)\}_{l\in\mathbb{N}}$ and $\{\rho_1(l)\}_{l\in\mathbb{N}}$ is said to be *solvable with non-negligible advantage* if these ensembles are not computationally indistinguishable; that is, there exist a polynomial-time quantum algorithm $\mathcal{A}$ and a polynomial p such that

$$\left|\Pr_{\mathcal{A}}[\mathcal{A}(\rho_0(l))=1]-\Pr_{\mathcal{A}}[\mathcal{A}(\rho_1(l))=1]\right|>\frac{1}{p(l)}$$

for infinitely many numbers l.

The problem QSCD_{ff} asks whether we can distinguish between two sequences of identical samples of $\rho_\pi^+(n)$ and of $\rho_\pi^-(n)$ for each fixed hidden permutation π for each length parameter n of a certain form. Let S_n be the *symmetric group* of degree n and let $\mathcal{K}_n = \{\pi \in S_n : \pi^2 = id \text{ and } \forall i \in \{1,...,n\}[\pi(i) \neq i]\}$ for $n \in N$, where id stands for the identity permutation.

Definition 1.2 Let $N = \{2(2n'+1) : n' \in \mathbb{N}\}$. For each $\pi \in \mathcal{K}_n$, let $\rho_\pi^+(n)$ and $\rho_\pi^-(n)$ be two quantum states defined by

$$\rho_\pi^+(n) = \frac{1}{2n!}\sum_{\sigma\in S_n}(|\sigma\rangle+|\sigma\pi\rangle)(\langle\sigma|+\langle\sigma\pi|) \text{ and } \rho_\pi^-(n) = \frac{1}{2n!}\sum_{\sigma\in S_n}(|\sigma\rangle-|\sigma\pi\rangle)(\langle\sigma|-\langle\sigma\pi|).$$

The problem QSCD_{ff} is the distinction problem between two quantum states $\rho_\pi^+(n)^{\otimes k(n)}$ and $\rho_\pi^-(n)^{\otimes k(n)}$ for each parameter n in N, where k is a polynomial. For each fixed polynomial k, we use the succinct notation k-QSCD_{ff} instead.

To simplify our notation, we often drop the parameter n whenever n is clear from the context. For instance, we write $\rho_\pi^{+\otimes k}$ for $\rho_\pi^+(n)^{\otimes k(n)}$. More generally, k-QSCD_{ff} can be defined for any integer-valued function k. Note that Definition 1.2 uses the parameter n to express the "length" of the quantum states instead of the parameter l of Definition 1.1. There is, however, essentially no difference for polynomial-time indistinguishability since ρ_π^+ and ρ_π^- can be expressed with $O(n \log n)$ qubits and $k(n)$ is a polynomial in n. This parameter n is used to measure the computational complexity of our problem and is often referred to as the *security parameter* in the cryptographic context.

1.1 Our Contributions

This paper shows three cryptographic properties of QSCD_{ff} and its application to quantum cryptography. These properties are summarized as follows. (*i*) QSCD_{ff} has the *trapdoor property*; namely, for any given hidden permutation $\pi \in \mathcal{K}_n$, we can efficiently distinguish between ρ_π^+ and ρ_π^-. (*ii*) The average-case hardness of QSCD_{ff} over a randomly chosen permutation $\pi \in \mathcal{K}_n$ coincides with its worst-case hardness. (*iii*) QSCD_{ff} is computationally at least as hard in the worst case as the *graph automorphism problem* (GA), where GA is the graph-theoretical problem defined as:

Graph Automorphism Problem (GA):
input: an undirected graph $G = (V, E)$;
output: YES if G has a non-trivial automorphism, and NO otherwise.

Since there is no known efficient algorithmic solution for GA, the third property suggests that QSCD_{ff} should be hard to solve. In a certain restricted case, we can actually show without any assumption that no time-unbounded quantum algorithm can solve $o(n \log n)$-QSCD_{ff}. Making use of the aforementioned three cryptographic properties, we can design a computationally-secure quantum PKC where its security relies on the worst-case hardness of GA. The following subsection discusses in depth numerous advantages of using QSCD_{ff} as a basis of secure quantum cryptosystems.

Furthermore, we give a generalization of QSCD_{ff}, called QSCD_{cyc}, and show its cryptographic properties: (*i*) the trapdoor property and (*ii*) the equivalence between its average-case and worst-case hardness. This new problem becomes a basis for another public-key cryptosystem that can encrypt messages longer than those in QSCD_{ff}.

1.2 Comparison between Our Work and Previous Work

In recent literature, computational-complexity aspects of quantum states have been spotlighted in connection to quantum cryptography. For instance, the notion of statistical distinguishability between two quantum states was investigated by Watrous [47] and also Kobayashi [29] in the context of quantum zero-knowledge proofs. They proved that certain problems of statistically distinguishing between two quantum states are promise-complete for quantum zero-knowledge proof systems. Aharonov and Ta-Shma [2] also studied the computational complexity of quantum-state generation and showed its connection to quantum adiabatic computing as well as statistical zero-knowledge proofs. Note that our distinction problem QSCD_{ff} is also rooted in computational complexity theory.

In what follows, we briefly discuss various advantages of using QSCD_{ff} as a basis of quantum cryptosystems in comparison with existing cryptosystems and their underlying problems.

Average-Case Hardness versus Worst-Case Hardness. The efficient solvability of any given problem on average, in general, does not guarantee the problem to be solved efficiently in worst case. This makes it desirable to satisfy the following property: the average-case hardness of the problem is "equivalent" to its worst-case hardness under a certain type of polynomial-time reduction. Unfortunately, few cryptographic problems are known to enjoy this property.

Roughly, there are two categories of worst-case/average-case reductions discussed in the past literature. The first category is a strong reduction, which transforms an arbitrary instance of length n to a random instance of the same length or length polynomial in n. In this strong sense, Ajtai [3] found a remarkable connection between average-case hardness and worst-case hardness of certain variants of the shortest vector problem (SVP). He gave an efficient reduction from the problem of approximating the shortest vector in a given n-dimensional lattice in the worst case to the approximation problem of the shortest vector in a random lattice over a certain class of lattices with a large polynomial approximation factor. Later, Micciancio and Regev [36] gave the average-case/worst-case connection factor of approximately n for approximating SVP (see [10] and references therein for general worst-case/average-case reductions).

The second category is a weak reduction of Tompa and Woll [46], where the reduction is randomized only over a part of its instances. A typical example is DLP, which can be randomly reduced to itself by a reduction that maps instances to not all instances of the same length but rather to all instances of the same underlying group. Nonetheless, unknown is an efficient reduction from DLP with the worst-case prime to DLP with a random prime. Note that Shor's algorithm [45] efficiently solves DLP and the inverting problem of the RSA function with worst-case/average-case reductions of the second category. The graph isomorphism problem (GI) and GA—well-known graph-theoretical problems—also enjoy such reductions of the second category [46] although there is no known cryptosystem whose security relies on their hardness.

This paper, to the contrary, shows that QSCD_{ff} has a worst-case/average-case reduction of the first category. Our reduction depends only on the size of the instance unlike the reduction of DLP. In fact, our distinction problem QSCD_{ff} is the *first* cryptographic problem with a worst-case/average-case reduction of the first category. Moreover, there is no known efficient solution to QSCD_{ff} on a quantum computer. Our reduction is similar in flavor to the reductions of the aforementioned lattice problems.

Computational Hardness of Underlying Computational Problems. The hidden subgroup problem (HSP) has played a central role in recent discussions on the strength and limitation of quantum computation. The aforementioned IFP and DLP can be viewed as special cases of HSP on Abelian groups (AHSP). Kitaev [28] showed how to solve AHSP efficiently; in particular, he gave a polynomial-time algorithm for the quantum Fourier transformation over Abelian groups, which is a generalization of the quantum Fourier transformation used in Shor's algorithm [45]. Although an efficient quantum algorithm exists for AHSP, a simple application of currently known techniques may not be sufficient to solve HSP on non-Abelian groups. (Note that HSP on certain specific non-Abelian groups were already solved in [6, 18, 21, 25, 31, 38, 43].) Another important variant is the HSP on the dihedral groups (DHSP). Recently, Regev [43] demonstrated a quantum reduction from the unique shortest vector problem (uSVP) to a slightly different variant of DHSP. Note that uSVP is a basis of the lattice-based PKCs given in [4, 42]. For DHSP, Kuperberg [31] found a subexponential-time quantum algorithm. Although these results do not directly imply a subexponential-time quantum algorithm for uSVP, they may be a clue to find the desired algorithm in the end.

Our problem QSCD_{ff} is closely related to a much harder problem: HSP on the *symmetric groups* (SHSP). Note that no known subexponential-time quantum algorithm exists for SHSP. Hallgren et al. [25] introduced a distinction problem between certain two quantum states, similar to QSCD_{ff}, to discuss the computational intractability of SHSP by a "natural" extension of Shor's [45] algorithm with the quantum Fourier transformation. An efficient solution to this distinction problem gives an answer to a pending question on a certain special case of SHSP. To solve this distinction problem, as they showed, the so-called *weak Fourier sampling* on a single sample should require an exponential number of samples. This result was improved by Grigni et al. [21], who proved that we need exponentially-many samples even by a stronger method called the *strong Fourier sampling* on a single sample along with a random choice of the bases of the representations of S_n. Kempe and Shalev [27] further expanded [21, 25] for the computational hardness of SHSP using these quantum Fourier sampling methods. Moore et al. [39], on the contrary, demonstrated that, regardless of methods (like the above quantum Fourier sampling methods), any time-unbounded quantum algorithm on a single sample needs $\exp(\Omega(n))$ samples to solve the distinction problem. Even for the two sample case, Moore and Russell [37] argued that any time-unbounded quantum algorithm that simultaneously works over two samples should use $\exp(\Omega(\sqrt{n}/\log n))$ samples at best. More recently, Hallgren et al. [24] proved that no time-unbounded quantum algorithm solves the distinction problem even from $o(n\log n)$ samples. In this paper, we further show that the distinction problem is polynomial-time reducible to QSCD_{ff}. This immediately implies that we have no time-unbounded quantum algorithm for QSCD_{ff} from $o(n\log n)$ samples. Even with sufficiently many samples for QSCD_{ff}, there is no known subexponential-time quantum algorithms for QSCD_{ff} and thus finding such an algorithm seems a daunting task. This situation, on the contrary, indicates that our problem QSCD_{ff} should be more suitable than, e.g., uSVP as an underlying intractable problem founding a secure cryptosystem similar to the classical case of DLP over different groups; namely, DLP over $\mathbb{Z}_p^*$ (where p is a prime) is classically solvable in subexponential time whereas no known classical subexponential-time algorithm exists for DLP over certain groups in elliptic curve cryptography. It is generally believed that DLP over such groups is more reliable than DLP over $\mathbb{Z}_p^*$.

We prove that the computational complexity of QSCD_{ff} is lower-bounded by that of GA. Note that well-known upper bounds of GA are $\mathbf{NP}\cap$co-$\mathbf{AM}$ [20, 44], $\mathbf{SPP}$ [5], and $\mathbf{UAP}$ [12] but GA is not yet known to be in $\mathbf{NP}\cap$co-$\mathbf{NP}$. Since most cryptographic problems fall in $\mathbf{NP}\cap$co-$\mathbf{NP}$, few cryptographic systems are lower-bounded by the worst-case hardness of problems outside of $\mathbf{NP}\cap$co-$\mathbf{NP}$.

Quantum Computational Cryptography. Apart from PKCs, quantum key distribution gives a foundation to symmetric-key cryptology; for instance, the quantum key distribution scheme in [8] achieves unconditionally secure sharing of secret keys in symmetric-key cryptosystems (SKCs) through an authenticated classical communication channel. Undoubtedly, both SKCs and PKCs have their own advantages and disadvantages. Compared with SKCs, PKCs require less secret keys in a large-scale network; however, they often need certain intractability assumptions for their security proofs and are typically vulnerable to, e.g., the man-in-the-middle attack. As an immediate application of QSCD_{ff}, we propose a new computational quantum PKC whose security relies on the computational hardness of QSCD_{ff}.

Of many existing PKCs, few make their security proofs solely rely on the *worst-case* hardness of their underlying problems. Quantum adversaries can break many PKCs whose underlying problems are number-theoretic because fast quantum algorithms can solve these problems. Based on a certain subset of the knapsack problem, Okamoto et al. [41] proposed a quantum PKC, which withstands certain well-known quantum attacks. Our proposed quantum PKC also seems to fend a polynomial-time quantum adversary since we can reduce the problem GA to QSCD_{ff}, where GA is not known to be solved efficiently on a quantum computer.

2 Cryptographic Properties of $\mathrm{QSCD}_{\!f\!f}$

Through this section, we want to show three cryptographic properties of $\mathrm{QSCD}_{\!f\!f}$: (*i*) the trapdoor property, (*ii*) the equivalence between average-case hardness and worst-case hardness, and (*iii*) a reduction from $\mathrm{QSCD}_{\!f\!f}$ to other computationally-hard problems. These properties help us construct a quantum PKC in Section 3.

All the cryptographic properties of $\mathrm{QSCD}_{\!f\!f}$ are consequences of the following remarkable characteristics of the set $\mathcal{K}_n$ of the hidden permutations (although the definition of $\mathcal{K}_n$ seems somewhat artificial). (*i*) Each permutation $\pi \in \mathcal{K}_n$ is of order 2. This directly provides the trapdoor property of $\mathrm{QSCD}_{\!f\!f}$. (*ii*) For any $\pi \in \mathcal{K}_n$, the conjugacy class of π is equal to $\mathcal{K}_n$. This property enables us to prove the equivalence between the worst-case hardness and average-case hardness of $\mathrm{QSCD}_{\!f\!f}$. (*iii*) The problem GA is (polynomial-time Turing) equivalent to its subproblem with the promise that any given graph has a unique non-trivial automorphism in $\mathcal{K}_n$ or none at all. This equivalence is used to give a complexity-theoretic lower bound of $\mathrm{QSCD}_{\!f\!f}$; that is, the average-case hardness of $\mathrm{QSCD}_{\!f\!f}$ is lower-bounded by the worst-case hardness of GA. For these proofs, we introduce two new techniques: (*i*) a variant of the so-called *coset sampling method*, which is broadly used in extensions of Shor's algorithm (see, e.g., [43]) and (*ii*) a quantum version of the *hybrid argument*, which is a strong tool for many security reductions used in computational cryptography.

Now, let us assume the reader's familiarity with basics of quantum computation [40] and recall the two quantum states $\rho_\pi^+ = \frac{1}{2n!}\sum_{\sigma\in S_n}(|\sigma\rangle+|\sigma\pi\rangle)(\langle\sigma|+\langle\sigma\pi|)$ and $\rho_\pi^- = \frac{1}{2n!}\sum_{\sigma\in S_n}(|\sigma\rangle-|\sigma\pi\rangle)(\langle\sigma|-\langle\sigma\pi|)$ given for a permutation $\pi \in \mathcal{K}_n$. For convenience, let $\iota(n)$ (or simply ι) denote the maximally mixed state $\frac{1}{n!}\sum_{\sigma\in S_n}|\sigma\rangle\langle\sigma|$ over S_n, which will appear later as a technical tool.

2.1 Trapdoor Property

The first property to prove is that $\mathrm{QSCD}_{\!f\!f}$ enjoys the *trapdoor property*, which has played a key role in various cryptosystems in use. To prove this property, it suffices to present an efficient distinguishing algorithm between ρ_π^+ and ρ_π^- without knowing their hidden permutation $\pi \in \mathcal{K}_n$.

Theorem 2.1 (Distinguishing Algorithm) There exists a polynomial-time quantum algorithm that, for a hidden permutation $\pi \in \mathcal{K}_n$, distinguishes between $\rho_\pi^+(n)$ and $\rho_\pi^-(n)$ for any $n \in N$ with probability 1.

Proof. Fix n arbitrarily. Let χ be any given unknown state, which is either ρ_π^+ or ρ_π^-. The desired distinguishing algorithm for χ works as follows.

(D1) Prepare two quantum registers. The first register holds a control bit and the second register holds χ. Apply the Hadamard transformation H to the first register. The state of the system now becomes

$$H|0\rangle\langle 0|H \otimes \chi.$$

(D2) Apply the Controlled-π operator C_π to the two registers, where the operator C_π satisfies $C_\pi|0\rangle|\sigma\rangle = |0\rangle|\sigma\rangle$ and $C_\pi|1\rangle|\sigma\rangle = |1\rangle|\sigma\pi\rangle$ for any given $\sigma \in S_n$. Since $\pi^2 = id$ for every

$\pi \in \mathcal{K}_n$, the state of the entire system can be expressed as

$$\frac{1}{n!}\sum_{\sigma\in S_n} |\psi^+_{\pi,\sigma}\rangle\langle\psi^+_{\pi,\sigma}| \quad \text{if } \chi = \rho^+_\pi, \quad \text{and} \quad \frac{1}{n!}\sum_{\sigma\in S_n} |\psi^-_{\pi,\sigma}\rangle\langle\psi^-_{\pi,\sigma}| \quad \text{if } \chi = \rho^-_\pi,$$

where $|\psi^+_{\pi,\sigma}\rangle$ and $|\psi^-_{\pi,\sigma}\rangle$ are defined by

$$\begin{aligned} |\psi^\pm_{\pi,\sigma}\rangle &= C_\pi\left(\frac{1}{2}|0\rangle\left(|\sigma\rangle \pm |\sigma\pi\rangle\right) + |1\rangle\left(|\sigma\rangle \pm |\sigma\pi\rangle\right)\right) \\ &= \frac{1}{2}|0\rangle(|\sigma\rangle \pm |\sigma\pi\rangle) + \frac{1}{2}|1\rangle(|\sigma\pi\rangle \pm |\sigma\rangle). \end{aligned}$$

(D3) Apply the Hadamard transformation to the first register. If χ is either ρ^+_π or ρ^-_π, then the state of the system becomes either

$$(H\otimes I)|\psi^+_{\pi,\sigma}\rangle = \frac{1}{\sqrt{2}}|0\rangle\left(|\sigma\rangle + |\sigma\pi\rangle\right) \quad \text{or} \quad (H\otimes I)|\psi^-_{\pi,\sigma}\rangle = \frac{1}{\sqrt{2}}|1\rangle\left(|\sigma\rangle - |\sigma\pi\rangle\right).$$

Measure the first register in the computational basis. If the result is 0, then output YES; otherwise, output NO.

Clearly, the above procedure gives the correct answer with probability 1. □

2.2 Reduction from Worst Case to Average Case

We want to reduce the worst-case hardness of $\mathrm{QSCD}_{\mathit{ff}}$ to its average-case hardness. Such a reduction implies that $\mathrm{QSCD}_{\mathit{ff}}$ with a random permutation π is at least as hard as $\mathrm{QSCD}_{\mathit{ff}}$ with the permutation π of the highest complexity. Since the converse reduction is trivial, the average-case hardness of $\mathrm{QSCD}_{\mathit{ff}}$ is, in fact, polynomial-time Turing equivalent to its worst-case hardness.

Theorem 2.2 Let k be any polynomial. Assume that there exists a polynomial-time quantum algorithm $\mathcal{A}$ that solves k-$\mathrm{QSCD}_{\mathit{ff}}$ with non-negligible advantage for a uniformly random $\pi \in \mathcal{K}_n$; namely, there exists a polynomial p such that, for infinitely many security parameters n in N,

$$\left|\Pr_{\pi,\mathcal{A}}[\mathcal{A}(\rho^+_\pi(n)^{\otimes k(n)}) = 1] - \Pr_{\pi,\mathcal{A}}[\mathcal{A}(\rho^-_\pi(n)^{\otimes k(n)}) = 1]\right| > \frac{1}{p(n)},$$

where π is chosen uniformly at random from $\mathcal{K}_n$. Then, there exists a polynomial-time quantum algorithm $\mathcal{B}$ that solves k-$\mathrm{QSCD}_{\mathit{ff}}$ with non-negligible advantage for any permutation $\pi \in \mathcal{K}_n$.

Proof. Fix an arbitrary parameter $n \in N$ that satisfies the assumption of the theorem. For each $i \in \{1, 2, ..., k(n)\}$, let χ_i be the ith state of the given $k(n)$ states. Note that χ_i is in $\{\rho^+_\pi, \rho^-_\pi\}$. We build the desired worst-case algorithm $\mathcal{B}$ from the average-case algorithm $\mathcal{A}$ in the following way.

(R1) Choose a permutation $\tau \in S_n$ uniformly at random.

(R2) Apply τ to each χ_i, where $i \in \{1, ..., k\}$, from the right. If $\chi_i = \rho_\pi^+$, then we obtain the quantum state

$$\begin{aligned}\chi_i' &= \frac{1}{2n!}\sum_{\sigma\in S_n}(|\sigma\tau\rangle + |\sigma\tau\tau^{-1}\pi\tau\rangle)(\langle\sigma\tau| + \langle\sigma\tau\tau^{-1}\pi\tau|) \\ &= \frac{1}{2n!}\sum_{\sigma'\in S_n}(|\sigma'\rangle + |\sigma'\tau^{-1}\pi\tau\rangle)(\langle\sigma'| + \langle\sigma'\tau^{-1}\pi\tau|).\end{aligned}$$

When $\chi_i = \rho_\pi^-$, we instead obtain $\chi_i' = \frac{1}{2n!}\sum_{\sigma'\in S_n}(|\sigma'\rangle - |\sigma'\tau^{-1}\pi\tau\rangle)(\langle\sigma'| - \langle\sigma'\tau^{-1}\pi\tau|)$.

(R3) Invoke the average-case quantum algorithm $\mathcal{A}$ on the input $\bigotimes_{i=1}^k \chi_i'$.

(R4) Output the outcome of $\mathcal{A}$.

Note that $\tau^{-1}\pi\tau$ belongs to $\mathcal{K}_n$ for any τ. Moreover, there exists a $\tau \in S_n$ satisfying that $\tau^{-1}\pi\tau = \pi'$ for each $\pi' \in \mathcal{K}_n$. Hence, the conjugacy class of π is equal to $\mathcal{K}_n$. In addition, the number of all permutations $\tau \in S_n$ for which $\tau^{-1}\pi\tau = \pi'$ is independent of the choice of $\pi' \in \mathcal{K}_n$. These properties implies that $\tau^{-1}\pi\tau$ is indeed uniformly distributed over $\mathcal{K}_n$. Therefore, by feeding the input $\bigotimes_{i=1}^k \chi_i'$ to the algorithm $\mathcal{A}$, we achieve the desired non-negligible advantage of $\mathcal{A}$. □

2.3 Computational Hardness

The third property of QSCD$_{ff}$ relates to the computational hardness of QSCD$_{ff}$. We want to present two claims that witness its relative hardness. First, we prove that the computational complexity of QSCD$_{ff}$ is lower-bounded by that of GA by constructing an efficient reduction from GA to QSCD$_{ff}$. Second, we discuss a relationship between QSCD$_{ff}$ and SHSP and prove that QSCD$_{ff}$ cannot be solved from $o(n \log n)$ samples.

Now, we prove the first claim on the reducibility of GA to QSCD$_{ff}$. Our reduction from GA to QSCD$_{ff}$ consists of two parts: a reduction from GA to a variant of GA, called UniqueGA$_{ff}$, and a reduction from UniqueGA$_{ff}$ to QSCD$_{ff}$. To describe the desired reduction, we first introduce two variants of GA. Earlier, Köbler et al. [30] introduced the following *unique graph automorphism problem* (UniqueGA).

Unique Graph Automorphism Problem (UniqueGA):
input: an undirected graph $G = (V, E)$;
promise: G has either a unique non-trivial automorphism or no non-trivial automorphism;
output: YES if G has the non-trivial automorphism, and NO otherwise.

Note that UniqueGA is called (1GA, GA) as a promise problem in [30]. To establish a direct connection to QSCD$_{ff}$, we further introduce the *unique graph automorphism with fully-flipped permutation* (UniqueGA$_{ff}$).

Unique Graph Automorphism with Fully-Flipped Permutation (UniqueGA$_{ff}$):
input: an undirected graph $G = (V, E)$, where $|V| = n \in N$;

promise: G has either a unique non-trivial automorphism $\pi \in \mathcal{K}_n$ or no non-trivial automorphism;

output: YES if G has the non-trivial automorphism, and NO otherwise.

Note that the instance G of UniqueGA$_{ff}$ is defined only when the number n of nodes belongs to the set $N = \{2(2n'+1) : n \in \mathbb{N}\}$.

We prove two useful lemmas regarding UniqueGA$_{ff}$. The first lemma uses the so-called *coset sampling method*, which has been largely used in many extensions of Shor's algorithm.

Lemma 2.3 There exists a polynomial-time quantum algorithm that, given an instance G of UniqueGA$_{ff}$, generates a quantum state ρ_π^+ if G is an "YES" instance with its unique non-trivial automorphism π, or generates $\iota = \frac{1}{n!}\sum_{\sigma\in S_n} |\sigma\rangle\langle\sigma|$ if G is a "NO" instance.

Proof. Given an instance G of UniqueGA$_{ff}$, we first prepare the quantum state $\frac{1}{\sqrt{n!}}\sum_{\sigma\in S_n}|\sigma\rangle|\sigma(G)\rangle$, where $\sigma(G)$ is the graph resulting from relabeling its nodes according to each permutation σ. By discarding the second register, we obtain the unique quantum state χ in the first register. This χ satisfies $\chi = \rho_\pi^+$ if G is an "YES" instance with the unique non-trivial automorphism π, and $\chi = \iota$ otherwise, as requested. □

The second lemma requires a variant of the coset sampling method as a technical tool. The lemma in essence relies on the fact that the hidden π is an odd permutation. This is one of the special properties of $\mathcal{K}_n$.

Lemma 2.4 There exists a polynomial-time quantum algorithm that, given an instance G of UniqueGA$_{ff}$, generates the quantum state ρ_π^- if G is an "YES" instance with the unique non-trivial automorphism π or generates ι if G is a "NO" instance.

Proof. Similar to the algorithm of Lemma 2.3, we start with the quantum state $\frac{1}{\sqrt{n!}}\sum_{\sigma\in S_n}|\sigma\rangle|\sigma(G)\rangle$ in two registers. Compute the sign of each permutation in the first register and then invert its phase exactly when the permutation is odd. Consequently, we obtain the quantum state $\frac{1}{\sqrt{n!}}\sum_{\sigma\in S_n}(-1)^{\mathrm{sgn}(\sigma)}|\sigma\rangle|\sigma(G)\rangle$, where $\mathrm{sgn}(\sigma) = 0$ if σ is even, and $\mathrm{sgn}(\sigma) = 1$ otherwise. By discarding the second register, we obtain a certain quantum state, say, χ in the first register. Note that, since π is odd, if σ is odd (even, resp.) then $\sigma\pi$ is even (odd, resp.). Therefore, it follows that $\chi = \rho_\pi^-$ if G is an "YES" instance with the unique non-trivial automorphism π, and $\chi = \iota$ otherwise. □

We are now ready to present a reduction from GA to QSCD$_{ff}$. This concludes that QSCD$_{ff}$ is computationally at least as hard as GA for infinitely-many input lengths n.

Theorem 2.5 If there exist a polynomial k and a polynomial-time quantum algorithm that solves k-QSCD$_{ff}$ with non-negligible advantage, then there exists a polynomial-time quantum algorithm that solves GA in the worst case for infinitely-many input lengths n.

Proof. We first show that GA is polynomial-time Turing equivalent to UniqueGA$_{ff}$ and then give a reduction from UniqueGA$_{ff}$ to QSCD$_{ff}$. The reduction from GA to UniqueGA$_{ff}$ is similar to the one given by Köbler et al. [30], who presented a polynomial-time Turing reduction from GA to UniqueGA.

Their polynomial-time algorithm for GA invokes UniqueGA as an oracle on a promised input, which is a graph of even number of nodes with either the unique non-trivial automorphism without any fixed point or no non-trivial automorphism at all. Modifying the construction of their reduction, we can easily obtain our reduction from GA to UniqueGA$_{\mathit{ff}}$. Furthermore, it is possible to make the length n satisfy the equation $n = 2(2n' + 1)$ for a certain $n' \in \mathbb{N}$ by a slight modification of their argument. Therefore, we obtain the following lemma.

Lemma 2.6 UniqueGA$_{\mathit{ff}}$ is polynomial-time Turing equivalent to GA.

Actually, a much stronger statement holds. When a Turing reduction to a promise problem makes only queries that satisfy the promise, the reduction is called *smart* [22]. Such a smart reduction is desirable for a security reduction of a cryptosystem. Since the reduction from GA to UniqueGA in [30] is indeed smart, so is our reduction. For readability, we postpone the proof of Lemma 2.6 until Appendix.

From Lemma 2.6, it suffices to construct a reduction from UniqueGA$_{\mathit{ff}}$ to QSCD$_{\mathit{ff}}$. Assume that there exist two polynomials k, p and a polynomial-time quantum algorithm $\mathcal{A}$ such that, for infinitely many n's, $\mathcal{A}$ solves k-QSCD$_{\mathit{ff}}$ with advantage $1/p(n)$. Let us fix an arbitrary n for which $\mathcal{A}$ solves k-QSCD$_{\mathit{ff}}$ with advantage $1/p(n)$. For any given instance G of UniqueGA$_{\mathit{ff}}$, we perform the following procedure:

(S1) Generate two sequences $S^+ = (\chi^{+\otimes k}, ..., \chi^{+\otimes k})$ and $S^- = (\chi^{-\otimes k}, ..., \chi^{-\otimes k})$ of $8p^2(n)n$ instances from G using the algorithms of Lemmas 2.3 and 2.4, respectively.

(S2) Invoke $\mathcal{A}$ on each component in S^+ and S^- as an input. Let $R^+ = (\mathcal{A}(\chi^{+\otimes k}), ..., \mathcal{A}(\chi^{+\otimes k}))$ and $R^- = (\mathcal{A}(\chi^{-\otimes k}), ..., \mathcal{A}(\chi^{-\otimes k}))$ be the resulting sequences.

(S3) Output YES if the difference between the number of 1's in R^+ and that in R^- is at least $4p(n)n$; output NO otherwise.

Note that if G is an "YES" instance, then we have $S^+ = (\overbrace{\rho_\pi^{+\otimes k}, ..., \rho_\pi^{+\otimes k}}^{8p^2(n)n})$ and $S^- = (\overbrace{\rho_\pi^{-\otimes k}, ..., \rho_\pi^{-\otimes k}}^{8p^2(n)n})$; otherwise, we have $S^+ = S^- = (\overbrace{\iota^{\otimes k}, ..., \iota^{\otimes k}}^{8p^2(n)n})$. Therefore, as far as G is an "YES" instance, the numbers of 1's in R^+ and in R^- are clearly different.

Finally, we estimate the above difference. Let X^+ and X^- be two random variables respectively expressing the numbers of 1's in R^+ and in R^-. Assume that G is an "YES" instance. The Höffding bound implies $\Pr[|X^+ - X^-| > 4p(n)n] > 1 - 2e^{-n}$ since $|\Pr[\mathcal{A}(\rho_\pi^{+\otimes k}) = 1] - \Pr[\mathcal{A}(\rho_\pi^{-\otimes k}) = 1]| > 1/p(n)$ from our assumption. Similarly, when G is a "NO" instance, we have $\Pr[|X^+ - X^-| < 4p(n)n] > 1 - 2e^{-n}$. This guarantees that the above procedure solves UniqueGA$_{\mathit{ff}}$ efficiently. □

As noted in Section 1, our distinction problem QSCD$_{\mathit{ff}}$ is rooted in SHSP. It is known that a special case of SHSP is reduced to the distinction problem between $\{\rho_\pi^+(n)\}_{n\in N}$ and $\{\iota(n)\}_{n\in N}$. As Hallgren et al. [24] argued, this problem cannot be solved by any time-unbounded quantum algorithm over $o(n \log n)$ identical samples. Regarding our second claim, we want to show a close relationship between QSCD$_{\mathit{ff}}$ and this distinction problem between $\{\rho_\pi^+(n)\}_{n\in N}$ and $\{\iota(n)\}_{n\in N}$.

Before stating the second claim, we present an algorithm that converts ρ_π^+ to ρ_π^- for any fixed $\pi \in \mathcal{K}_n$. This algorithm is a key to the proof of the claim and further to the construction of a quantum PKC in the subsequent section.

Lemma 2.7 (Conversion Algorithm) There exists a polynomial-time quantum algorithm that, with certainty, converts $\rho_\pi^+(n)$ into $\rho_\pi^-(n)$ and keeps $\iota(n)$ as it is for any parameter $n \in N$ and any hidden permutation $\pi \in \mathcal{K}_n$.

Proof. First, recall the definition of $\mathrm{sgn}(\sigma)$. Let $\pi \in \mathcal{K}_n$ be any hidden permutation. For its corresponding quantum state ρ_π^+, the desired algorithm simply inverts its phase according to the sign of the permutation. This is done by performing the following transformation:

$$|\sigma\rangle + |\sigma\pi\rangle \longmapsto (-1)^{\mathrm{sgn}(\sigma)}|\sigma\rangle + (-1)^{\mathrm{sgn}(\sigma\pi)}|\sigma\pi\rangle.$$

Note that deciding the sign of a given permutation takes only polynomial time. Since π is odd, the above algorithm obviously converts ρ_π^+ to ρ_π^-. Moreover, the algorithm does not alter the quantum state ι. $\square$

A result similar to [24] also holds for QSCD_{ff} on the distinguishing hardness of two quantum states. Theorem 2.8 shows that QSCD_{ff} can be reduced to the above distinction problem in polynomial time. As an immediate consequence, no time-unbounded quantum algorithm can solve QSCD_{ff} from $o(n \log n)$ samples. The proof of the theorem requires a quantum version of the so-called *hybrid argument*.

Theorem 2.8 Let k be any polynomial. If there exists a quantum algorithm $\mathcal{A}$ such that

$$\left| \Pr_{\mathcal{A}}[\mathcal{A}(\rho_\pi^+(n)^{\otimes k(n)}) = 1] - \Pr_{\mathcal{A}}[\mathcal{A}(\rho_\pi^-(n)^{\otimes k(n)}) = 1] \right| > \varepsilon(n)$$

for any security parameter $n \in N$, then there exists a quantum algorithm $\mathcal{B}$ such that, for each $n \in N$,

$$\left| \Pr_{\mathcal{B}}[\mathcal{B}(\rho_\pi^+(n)^{\otimes k(n)}) = 1] - \Pr_{\mathcal{B}}[\mathcal{B}(\iota(n)^{\otimes k(n)}) = 1] \right| > \frac{\varepsilon(n)}{4}.$$

Proof. Fix $n \in N$ arbitrarily and we hereafter omit this parameter n. Assume that a quantum algorithm $\mathcal{A}$ distinguishes between $\rho_\pi^{+\otimes k}$ and $\rho_\pi^{-\otimes k}$ with advantage at least $\varepsilon(n)$. Let $\mathcal{A}'$ be the algorithm that applies the conversion algorithm of Lemma 2.7 to a given state χ (which is either $\rho_\pi^{+\otimes k}$ or $\iota^{\otimes k}$) and then feeds the resulting state χ' (either $\rho_\pi^{-\otimes k}$ or $\iota^{\otimes k}$) to $\mathcal{A}$. Note that $\mathcal{A}'(\rho_\pi^{+\otimes k}) = \mathcal{A}(\rho_\pi^{-\otimes k})$ and $\mathcal{A}'(\iota^{\otimes k}) = \mathcal{A}(\iota^{\otimes k})$ by our definition. It thus follows by the triangle inequality that

$$\left| \Pr_{\mathcal{A}}[\mathcal{A}(\rho_\pi^{+\otimes k}) = 1] - \Pr_{\mathcal{A}}[\mathcal{A}(\iota^{\otimes k}) = 1] \right| + \left| \Pr_{\mathcal{A}'}[\mathcal{A}'(\rho_\pi^{+\otimes k}) = 1] - \Pr_{\mathcal{A}'}[\mathcal{A}'(\iota^{\otimes k}) = 1] \right| > \varepsilon(n)$$

for any parameter $n \in N$. This inequality leads us to either

$$\left| \Pr_{\mathcal{A}}[\mathcal{A}(\rho_\pi^{+\otimes k}) = 1] - \Pr_{\mathcal{A}}[\mathcal{A}(\iota^{\otimes k}) = 1] \right| > \frac{\varepsilon(n)}{2}$$

or

$$\left|\Pr_{\mathcal{A}'}[\mathcal{A}'(\rho_\pi^{+\otimes k})=1]-\Pr_{\mathcal{A}'}[\mathcal{A}'(\iota^{\otimes k})=1]\right|>\frac{\varepsilon(n)}{2}.$$

To complete the proof, we design the desired algorithm $\mathcal{B}$ as follows: first choose either $\mathcal{A}$ or $\mathcal{A}'$ at random and then simulate the chosen algorithm. It is easy to verify that $\mathcal{B}$ distinguishes between $\rho_\pi^{+\otimes k}$ and $\iota^{\otimes k}$ with advantage at least $\varepsilon(n)/4$. □

3 Application to a Quantum Public-Key Cryptosystem

Section 2 has shown the useful cryptographic properties of $\mathrm{QSCD}_{\mathit{ff}}$. Founded on these properties, we wish to construct a quantum PKC where the computational hardness of $\mathrm{QSCD}_{\mathit{ff}}$ (which can be further reduced to the hardness of GA) guarantees its security. We start with an efficient quantum algorithm that generates ρ_π^+ from π.

Lemma 3.1 ($\boldsymbol{\rho_\pi^+}$-Generation Algorithm) There exists a polynomial-time quantum algorithm that, on input $\pi\in\mathcal{K}_n$, generates the quantum state ρ_π^+ with probability 1.

Proof. The desired generation algorithm uses two registers and is given below. It is straightforward to verify the correctness of the given algorithm and we omit the correctness proof.

(G1) Prepare the state $|0\rangle|id\rangle$ in two quantum registers.
(G2) Apply the Hadamard transformation to the first register to obtain the state $\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle)|id\rangle$.
(G3) Perform the Controlled-π on the both registers and we obtain the state $\frac{1}{\sqrt{2}}(|0\rangle|id\rangle+|1\rangle|\pi\rangle)$.
(G4) Subtract 1 from the content of the first register exactly when the second register contains π. This process gives rise to the state $\frac{1}{\sqrt{2}}(|0\rangle|id\rangle+|0\rangle|\pi\rangle)$.
(G5) Apply a uniformly random permutation σ to the content of the second register from the left. The whole quantum system becomes $\frac{1}{\sqrt{2}}(|0\rangle|\sigma\rangle+|0\rangle|\sigma\pi\rangle)$.
(G6) Output the content of the second register.

□

Hereafter, we describe our quantum PKC and give its security proof. For the security proof, we need to specify the model of adversary's attack. Of all attack models discussed in [7], we choose a quantum analogue of *the indistinguishability against the chosen plaintext attack (IND-CPA)* and adapt the following "weakest" scenario:

> Alice (sender) wants to send a classical single-bit message securely to Bob (receiver) via a quantum channel. Assume that Alice and Bob are capable of running polynomial-time quantum algorithms. Bob first generates a certain quantum state as an encryption key. Alice requests him for his encryption key and then encrypts her message using the key. By making a request to Bob, Eve (adversary) also obtains numerous copies of his encryption key. Therefore, we can assume that Eve's attack concentrates on Alice's message transmission phase through the quantum channel. Eve intercepts Alice's encrypted message via

the channel and tries to decrypt it using polynomially-many copies of Bob's encryption key by applying polynomial-time quantum algorithms.

Now, we explain our quantum PKC protocol in detail. Note that, in our protocol, Alice transmits a single-bit message to Bob using his $O(\log n)$-qubit-long encryption key. Our protocol consists of two phases: Bob's key transmission phase and Alice's message transmission phase. (See Figure 1.)

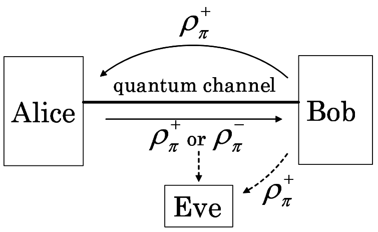

Figure 1: our public-key cryptosystem

Here is the precise description of our quantum PKC protocol.

[Key transmission phase]

(A1) Bob chooses a decryption key π uniformly at random from $\mathcal{K}_n$.

(A2) Bob generates sufficiently many copies of the encryption key ρ_π^+.

(A3) Alice obtains a copy of the encryption key from Bob.

[Message transmission phase]

(A4) Alice encrypts 0 or 1 into ρ_π^+ or ρ_π^-, respectively, and sends the encrypted message back to Bob.

(A5) Bob decrypts Alice's message using the decryption key π.

Step (A1) can be implemented by first choosing different transpositions uniformly at random and then letting π to be the product of these chosen transpositions. Step (A2) is done by the ρ_π^+-generation algorithm of Lemma 3.1. The conversion algorithm of Lemma 2.7 implements Step (A4) since Alice sends Bob either the received state ρ_π^+ or its converted state ρ_π^-. Finally, the distinguishing algorithm of Theorem 2.1 implements Step (A5).

The security of our PKC is proven by reducing GA to Eve's attack during the message transmission phase. Our reduction is a simple modification of the reduction given in Theorem 2.5.

Proposition 3.2 Let $\mathcal{A}$ be any polynomial-time quantum adversary who attacks our quantum PKC during the message transmission phase. Assume that there exist two polynomials $p(n)$ and $l(n)$ satisfying that

$$\left| \Pr_{\pi,\mathcal{A}}[\mathcal{A}(\rho_\pi^+, \rho_\pi^{+\otimes l(n)}) = 1] - \Pr_{\pi,\mathcal{A}}[\mathcal{A}(\rho_\pi^-, \rho_\pi^{+\otimes l(n)}) = 1] \right| > \frac{1}{p(n)}$$

for infinitely many parameters $n \in N$. Then, there exists a polynomial-time quantum algorithm that solves GA in the worst case with non-negligible probability for infinitely many n's.

Proof. The proposition immediately follows from the proof of Theorem 2.5 by replacing $\rho_\pi^{+\otimes k}$, $\rho_\pi^{-\otimes k}$, and $\iota^{\otimes k}$ in the proof with $(\rho_\pi^+, \rho_\pi^{+\otimes l(n)})$, $(\rho_\pi^-, \rho_\pi^{+\otimes l(n)})$, and $(\iota, \iota^{\otimes l(n)})$, respectively. $\square$

4 Generalization of $\mathrm{QSCD}_{\mathit{ff}}$

In our $\mathrm{QSCD}_{\mathit{ff}}$-based quantum PKC, Alice encrypts a single-bit message using an $O(n \log n)$-qubit encryption key. We wish to show how to increase the size of Alice's encryption message and construct a multi-bit quantum PKC built upon a generalization of $\mathrm{QSCD}_{\mathit{ff}}$, called QSCD_{cyc} (QSCD with cyclic permutations), which is the distinction problem among multiple ensembles of quantum states. Recall that Definition 1.1 has introduced the notion of computational indistinguishability between two ensembles of quantum states. This notion can be naturally generalized as follows to multiple quantum state ensembles.

Definition 4.1 We say that m ensembles $\{\rho_0(l)\}_{l\in\mathbb{N}}, \ldots, \{\rho_{m-1}(l)\}_{l\in\mathbb{N}}$ of quantum states are *computationally indistinguishable* if, for any distinct pair $i,j \in \mathbb{Z}_m$, the advantage between the two ensembles $\{\rho_i(l)\}_{l\in\mathbb{N}}$ and $\{\rho_j(l)\}_{l\in\mathbb{N}}$ is negligible for any polynomial-time quantum algorithm $\mathcal{A}$; namely, for any two ensembles $\{\rho_i(l)\}_{l\in\mathbb{N}}$ and $\{\rho_j(l)\}_{l\in\mathbb{N}}$, any polynomial p, any polynomial-time quantum algorithm $\mathcal{A}$, and any sufficiently large number l, we have

$$\left| \Pr_{\mathcal{A}}[\mathcal{A}(\rho_i(l)) = 1] - \Pr_{\mathcal{A}}[\mathcal{A}(\rho_j(l)) = 1] \right| < \frac{1}{p(l)}.$$

The distinction problem among $\{\rho_0(l)\}_{l\in\mathbb{N}}, \ldots, \{\rho_{m-1}(l)\}_{l\in\mathbb{N}}$ is said to be *solvable with non-negligible advantage* if the ensembles are not computationally indistinguishable; i.e., there exist two ensembles $\{\rho_i(l)\}_{l\in\mathbb{N}}$ and $\{\rho_j(l)\}_{l\in\mathbb{N}}$, a polynomial-time quantum algorithm $\mathcal{A}$ and a polynomial p such that

$$\left| \Pr_{\mathcal{A}}[\mathcal{A}(\rho_i(l)) = 1] - \Pr_{\mathcal{A}}[\mathcal{A}(\rho_j(l)) = 1] \right| > \frac{1}{p(l)}$$

for infinitely many numbers $l \in \mathbb{N}$.

We wish to define a specific distinction problem, denoted succinctly QSCD_{cyc} among m ensembles of quantum states. For any fixed $n \in \mathbb{N}$, assume that $m \geq 2$ and m divides n. For each $\sigma \in S_n$, $\pi \in \mathcal{K}_n^m$, and $s \in \mathbb{Z}_m$, let

$$|\Phi_{\pi,s}^{\sigma}\rangle = \frac{1}{\sqrt{m}} \sum_{t=0}^{m-1} \omega_m^{st} |\sigma\pi^t\rangle,$$

where $\omega_m = e^{2\pi i/m}$. Our new hidden permutation π consists of disjoint n/m cyclic permutations of length m; namely, π is of the form

$$\pi = (i_0\, i_1\, \cdots\, i_{m-1}) \cdots (i_{n-m}\, i_{n-m+1}\, \cdots\, i_{n-1}),$$

where $i_s, i_t \in \mathbb{Z}_m$ and $i_s \neq i_t$ if $s \neq t$ for any pair (s,t). Such a permutation π has the following properties: (i) π has no fixed points (i.e., $\pi(i) \neq i$ for any $i \in \mathbb{Z}_n$) and (ii) π is of order m (i.e., $\pi^m = id$). For convenience, denote by $\mathcal{K}_n^m \subseteq S_n$ the set of all such permutations. The distinction problem QSCD_{cyc} is finally defined in the following way.

Definition 4.2 The problem QSCD_{cyc} is the distinction problem among m ensembles $\{\rho_\pi^{(0)}(n)^{\otimes k(n)}\}_{n\in\mathbb{N}}$, ..., $\{\rho_\pi^{(m-1)}(n)^{\otimes k(n)}\}_{n\in\mathbb{N}}$ of quantum states, where k is a polynomial and the notation $\rho_\pi^{(s)}(n)$ denotes the mixed state $\frac{1}{n!}\sum_{\sigma\in S_n} |\Phi_{\pi,s}^{\sigma}\rangle\langle\Phi_{\pi,s}^{\sigma}|$ for each $\pi \in \mathcal{K}_n^m$. In particular, for any fixed k, we write k-QSCD_{cyc}.

As in the case of QSCD_{ff}, we also drop the parameter n wherever possible. Note that QSCD_{ff} coincides with QSCD_{cyc} with $m=2$ and $n=2(2n'+1)$ for a certain number $n'\in\mathbb{N}$.

The generalized problem QSCD_{cyc} also enjoys useful cryptographic properties. We first present the trapdoor property of QSCD_{cyc}. In the case of QSCD_{ff}, we embed only a single bit into the quantum states ρ_π^+ and ρ_π^-. This is possible because its trapdoor information π is a permutation of order two. Since π is of order $m\geq 2$ in QSCD_{cyc}, m bits can be embedded into the quantum states $\rho_\pi^{(0)},\ldots,\rho_\pi^{(m-1)}$.

Now, we present a distinguishing algorithm for $\rho_\pi^{(s)}$.

Theorem 4.3 (Generalized Distinguishing Algorithm) There exists a polynomial-time quantum algorithm that, for each $n\in\mathbb{N}$, $\pi\in\mathcal{K}_n^m$, and $s\in\mathbb{Z}_m$, decrypts $\rho_\pi^{(s)}(n)$ to s with exponentially-small error probability.

Proof. Let χ be any given quantum state of the form $\rho_\pi^{(s)}$ for a certain hidden permutation $\pi\in\mathcal{K}_n^m$ and a hidden parameter s. Note that χ is the mixture of pure states $|\Phi_{\pi,s}^\sigma\rangle$ over a randomly chosen $\sigma\in S_n$. It thus suffices to give a polynomial-time quantum algorithm that decrypts $|\Phi_{\pi,s}^\sigma\rangle$ to s for any fixed σ. Such an algorithm can be given by conducting the following *Generalized Controlled-π Test*, which is a straightforward generalization of the distinguishing algorithm given in Theorem 2.1.

[Generalized Controlled-π Test]

(D1') Prepare two quantum registers. The first register holds a control string, initially set to $|0\rangle$, and the second register holds the state $|\Phi_{\pi,s}^\sigma\rangle$. Apply the inverse Fourier transformation F_m^{-1} to the first register. Meanwhile, assume that we can perform the Fourier transformation exactly. The total system then becomes

$$\frac{1}{\sqrt{m}}\sum_{r=0}^{m-1}|r\rangle|\Phi_{\pi,s}^\sigma\rangle=\frac{1}{m}\sum_{r,t}\omega_m^{st}|r\rangle|\sigma\pi^t\rangle.$$

(D2') Apply π to the content of the second register from the right r times. The state of the total system now becomes

$$\frac{1}{m}\sum_{r,t}\omega_m^{st}|r\rangle|\sigma\pi^{r+t \bmod m}\rangle.$$

(D3') Apply the Fourier transformation F_m to the first register and we obtain the state

$$\begin{aligned}
&\frac{1}{m}\sum_{r,t}\frac{1}{\sqrt{m}}\sum_{r'=0}^{m-1}\omega_m^{rr'}|r'\rangle\omega_m^{st}|\sigma\pi^{r+t \bmod m}\rangle\\
=\;&\frac{1}{m^{3/2}}\sum_{r,r',t}\omega_m^{st+rr'}|r'\rangle|\sigma\pi^{r+t \bmod m}\rangle\\
=\;&\frac{1}{m^{3/2}}\sum_{r,t}\omega_m^{s(r+t)}|s\rangle|\sigma\pi^{r+t \bmod m}\rangle+\frac{1}{m^{3/2}}\sum_{r,t,r'\neq s}\omega_m^{st+rr'}|r'\rangle|\sigma\pi^{r+t \bmod m}\rangle\\
=\;&\frac{1}{\sqrt{m}}\sum_{t=0}^{m-1}\omega_m^{st}|s\rangle|\sigma\pi^t\rangle\;=\;|s\rangle|\Phi_{\pi,s}^\sigma\rangle.
\end{aligned}$$

(D4') Finally, measure the first register in the computational basis and output the result s in $\mathbb{Z}_m$.

The error probability of the above algorithm depends only on the precision of the Fourier transformation over $\mathbb{Z}_m$. As shown in [28], the quantum Fourier transformation can be implemented with exponentially-small error probability by the approximated quantum Fourier transformation. Therefore, the theorem follows. $\Box$

Similar to QSCD_{ff}, the average-case hardness of QSCD_{cyc} coincides with its worst-case hardness.

Theorem 4.4 Let k be any polynomial. Assume that there exists a polynomial-time quantum algorithm $\mathcal{A}$ that solves k-QSCD_{cyc} with non-negligible advantage for a uniformly random $\pi \in \mathcal{K}_n^m$; namely, there exist two numbers $s, s' \in \mathbb{Z}_m$ and a polynomial p such that, for infinitely many numbers $n \in \mathbb{N}$,

$$\left| \Pr_{\pi,\mathcal{A}}[\mathcal{A}(\rho_\pi^{(s)}(n)^{\otimes k(n)}) = 1] - \Pr_{\pi,\mathcal{A}}[\mathcal{A}(\rho_\pi^{(s')}(n)^{\otimes k(n)}) = 1] \right| > \frac{1}{p(n)},$$

where π is chosen uniformly at random from $\mathcal{K}_n^m$. Then, there exists a polynomial-time quantum algorithm $\mathcal{B}$ that solves k-QSCD_{cyc} with non-negligible advantage.

Proof. Applying a uniformly random permutation $\tau \in S_n$ to $|\Phi_{\pi,s}^\sigma\rangle$ from its right side and we obtain the state

$$\frac{1}{\sqrt{m}} \sum_{t=0}^{m-1} \omega_m^{st} |\sigma\pi^t\tau\rangle = \frac{1}{\sqrt{m}} \sum_{t=0}^{m-1} \omega_m^{st} |\sigma\tau\tau^{-1}\pi^t\tau\rangle = \frac{1}{\sqrt{m}} \sum_{t=0}^{m-1} \omega_m^{st} |\sigma\tau(\tau^{-1}\pi\tau)^t\rangle.$$

Note that $\frac{1}{n!}\sum_{\sigma\in S_n} |\Phi_{\tau^{-1}\pi\tau,s}^{\sigma\tau}\rangle\langle\Phi_{\tau^{-1}\pi\tau,s}^{\sigma\tau}|$ is an average-case instance of QSCD_{cyc} since $\tau^{-1}\pi\tau$ is distributed uniformly at random over $\mathcal{K}_n^m$. The rest of the proof follows by an argument similar to the proof of Theorem 2.2. $\Box$

We want to show a quantum algorithm that generates the quantum state $\rho_\pi^{(s)}$ efficiently from π and s. This generation algorithm will be used to generate encryption keys in our QSCD_{cyc}-based multi-bit quantum PKC.

Lemma 4.5 ($\rho_\pi^{(s)}$-Generation Algorithm) There exists a polynomial-time quantum algorithm that generates $\rho_\pi^{(s)}$ for any $s \in \mathbb{Z}_m$ and any $\pi \in \mathcal{K}_n^m$ with exponentially-small error probability.

Proof. The construction is based on a straightforward generalization of the ρ_π^+-generation algorithm. We use the approximated Fourier transformation [28] instead of the Hadamard transformation. Note that the Fourier transformation F_π over the cyclic group $\{id, \pi, \pi^2, ..., \pi^{m-1}\}$ can be efficiently approximated from π by an argument similar to the proof of Lemma 3.1 using the approximated Fourier transformation. Such approximation enables us to perform with exponentially-small error probability the following transformation:

$$F_\pi|\pi^s\rangle = \frac{1}{\sqrt{m}} \sum_{t=0}^{m-1} \omega_m^{st} |\pi^t\rangle.$$

Since the initial state $|\pi^s\rangle$ can be easily generated from π, we immediately obtain the approximation of $F_\pi|\pi^s\rangle$. By applying a uniformly-random permutation $\sigma \in S_n$ to the resulting state from the left, we obtain the desired state $\rho_\pi^{(s)}$ with exponentially-small error probability. $\square$

Toward the end of this section, we present our multi-bit quantum PKC.

[Key transmission phase]

(A1') Bob chooses a decryption key π uniformly at random from $\mathcal{K}_n^m$.

(A2') Bob generates the series $(\rho_\pi^{(0)}, ..., \rho_\pi^{(m-1)})$ of his encryption keys.

(A3') Alice obtains the entire encryption keys from Bob.

[Message transmission phase]

(A4') Alice picks up $\rho_\pi^{(s)}$ for her message $s \in \mathbb{Z}_m$ and sends $\rho_\pi^{(s)}$ back to Bob.

(A5') Bob decrypts Alice's encrypted message using his decryption key π.

By choosing cycles one by one sequentially, we can perform Step (A1'). The $\rho_\pi^{(s)}$-generation algorithm of Lemma 4.5 immediately implements Step (A2'). Note that Alice can encrypt her message s simply by choosing $\rho_\pi^{(s)}$ from the series $(\rho_\pi^{(0)}, ..., \rho_\pi^{(m-1)})$ of Bob's encryption keys. Finally, the generalized distinguishing algorithm in Theorem 4.3 achieves Step (A5').

A major drawback of our multi-bit encryption scheme is that Bob needs to send Alice all the encryption keys $(\rho_\pi^{(0)}, ..., \rho_\pi^{(m-1)})$ simply because of the lack of a sophisticated converting algorithm among different encryption keys without knowing a hidden decryption key π. For comparison, recall the conversion algorithm for the $\mathrm{QSCD}_{\mathit{ff}}$-based single-bit encryption scheme. This conversion algorithm utilizes the "parity" of σ and $\sigma\pi$ to invert their phase without using any information on π. More precisely, the algorithm implements the homomorphism f from S_n to $\{+1, -1\} \cong \mathbb{Z}/2\mathbb{Z}$ satisfying that $f(\sigma) = +1$ (-1, resp.) if σ is even (odd, resp.). Unfortunately, the same algorithm fails for QSCD_{cyc}. This is seen as follows. Let us assume, to the contrary, that there exists a homomorphism g mapping S_n to $\{1, \omega_m, ..., \omega_m^{m-1}\}$ ($\cong \mathbb{Z}/m\mathbb{Z}$). The *fundamental homomorphism theorem* implies that $S_n/\mathrm{Ker}(g) \cong \mathbb{Z}/m\mathbb{Z}$; namely, there exists an isomorphism from $\sigma\mathrm{Ker}(g)$ to $g(\sigma)$ for every $\sigma \in S_n$. Note that $\mathrm{Ker}(g)$ is a normal subgroup in S_n. It is known that such a normal subgroup in S_n equals either the trivial group $\{id\}$ or the alternation group A_n. Apparently, there is no isomorphism between $\{\sigma A_n : \sigma \in S_n\}$ and $\mathbb{Z}/m\mathbb{Z}$ nor isomorphism between $\{\sigma : \sigma \in S_n\}$ and $\mathbb{Z}/m\mathbb{Z}$ if $n > 4$ and $n \geq m > 2$. This contradicts our assumption.

5 Concluding Remarks

The computational distinction problem $\mathrm{QSCD}_{\mathit{ff}}$ has useful properties to design a quantum PKC whose security is guaranteed by the computational intractability of GA. Although GA is reducible to $\mathrm{QSCD}_{\mathit{ff}}$, there seems a large gap between the hardness of GA and that of $\mathrm{QSCD}_{\mathit{ff}}$ because all the combinatorial structures of input graphs in GA are completely lost in $\mathrm{QSCD}_{\mathit{ff}}$. It is therefore pressing to find a nice classical problem (for instance, the problems of finding a centralizer or finding a normalizer [33]) which almost matches the computational hardness of $\mathrm{QSCD}_{\mathit{ff}}$. Since no fast quantum algorithm is known for $\mathrm{QSCD}_{\mathit{ff}}$, discovering such an algorithm may require new tools and novel proof techniques in quantum complexity theory. Besides our quantum states $\{\rho_\pi^+(n), \rho_\pi^-(n)\}$ in $\mathrm{QSCD}_{\mathit{ff}}$, it is imperative to search for other simple quantum states whose computational indistinguishability is helpful to construct a more secure cryptosystem.

Similar to $\mathrm{QSCD}_{\mathit{ff}}$, QSCD_{cyc} owns useful cryptographic properties for which we have built a multi-bit quantum PKC. It is unfortunate that the intractability of QSCD_{cyc} and therefore the security of our multi-bit quantum PKC are not yet clear. If one proves that the worst-case hardness of QSCD_{cyc} is lower-bounded by, for instance, the hardness of GA, then our multi-bit quantum PKC might find more practical use.

Acknowledgments. The authors are grateful to Hirotada Kobayashi and Claude Crépeau for fruitful discussions, to John Watrous for useful comments on key ideas, to Donald Beaver, Louis Salvail, and the anonymous reviewers of EUROCRYPT 2005 for their valuable suggestions. The authors' thanks also go to Cristopher Moore for providing references to the historical account of hidden subgroup problems. This research was partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Young Scientists (B) No.17700007, 2005 and for Scientific Research on Priority Areas No.16092206, 2005.

References

[1] M. Adcock and R. Cleve. A quantum Goldreich-Levin theorem with cryptographic applications. In *Proceedings of the 19th Annual Symposium on Theoretical Aspects of Computer Science*, LNCS 2285, pages 323–334. Springer, 2002.

[2] D. Aharonov and A. Ta-Shma. Adiabatic quantum state generation and statistical zero knowledge. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing*, pages 20–29, 2003.

[3] M. Ajtai. Generating hard instances of lattice problems (extended abstract). In *Proceedings of the 28th annual ACM Symposium on Theory of Computing*, pages 99–108, 1996.

[4] M. Ajtai and C. Dwork. A public-key cryptosystem with worst-case/average-case equivalence. In *Proceedings of the 29th annual ACM Symposium on Theory of Computing*, pages 284–293, 1997. See also ECCC TR96-065.

[5] V. Arvind and P. P. Kurur. Graph isomorphism is in SPP. In *Proceedings of the 43rd IEEE Symposium on Foundations of Computer Science*, pages 743–750, 2002.

[6] D. Bacon, A. M. Childs, and W. van Dam. From optimal measurement to efficient quantum algorithms for the hidden subgroup problem over semidirect product groups. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, pages 469–478, 2005.

[7] M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway. Relations among notions of security for public-key encryption schemes. In *Advances in Cryptology - CRYPTO '98*, pages 26–45. Springer, 1998.

[8] C. H. Bennett and G. Brassard. Quantum cryptography: public key distribution and coin tossing. In *Proceedings of IEEE International Conference on Computers, Systems, and Signal Processing*, pages 175–179, 1984.

[9] M. Blum and S. Micali. How to generate cryptographically strong sequences of pseudo-random bits. *SIAM Journal on Computing*, 13(4):850–864, 1984.

[10] A. Bogdanov and L. Trevisan. On worst-case to average-case reductions for NP problems. In *Proceedings of the 44th IEEE Symposium on Foundations of Computer Science*, pages 308–317, 2004.

[11] D. Boneh and R. J. Lipton. Quantum cryptanalysis of hidden linear functions (extended abstract). In *Advances in Cryptology - CRYPTO '95*, LNCS 963, pages 424–437. Springer, 1995.

[12] M. Crâsmaru, C. Glaßer, K. W. Regan, and S. Sengupta. A protocol for serializing unique strategies. In *Proceedings of the 29th International Symposium on Mathematical Foundations of Computer Science*, LNCS 3153, pages 660–672. Springer, 2004.

[13] C. Crépeau, P. Dumais, D. Mayers, and L. Salvail. Computational collapse of quantum state with application to oblivious transfer. In *Proceedings of the 1st Theory of Cryptography Conference*, LNCS 2951, pages 374–393. Springer, 2004.

[14] C. Crépeau, F. Légaré, and L. Salvail. How to convert the flavor of a quantum bit commitment. In *Advances in Cryptology - EUROCRYPT '01*, LNCS 2045, pages 60–77. Springer, 2001.

[15] I. Damgård, S. Fehr, and L. Salvail. Zero-knowledge proofs and string commitments withstanding quantum attacks. In *Advances in Cryptology - CRYPTO '04*, LNCS 3152, pages 254–272. Springer, 2004.

[16] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22(6):644–654, 1976.

[17] P. Dumais, D. Mayers, and L. Salvail. Perfectly concealing quantum bit commitment from any quantum one-way permutation. In *Advances in Cryptology - EUROCRYPT 2000*, LNCS 1807, pages 300–315. Springer, 2000.

[18] M. Ettinger and P. Høyer. On quantum algorithms for noncommutative hidden subgroups. *Advances in Applied Mathematics*, 25:239–251, 2000.

[19] S. Goldwasser and S. Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270–299, 1984.

[20] S. Goldwasser and M. Sipser. Private coins versus public coins in interactive proof system. In S. Micali, editor, *Advances in Computing Research, Vol. 5: Randomness and Computation*, pages 73–90. JAI Press, 1989.

[21] M. Grigni, L. J. Schulman, M. Vazirani, and U. Vazirani. Quantum mechanical algorithms for the nonabelian hidden subgroup problem. *Combinatorica*, 24(1):137–154, 2004.

[22] J. Grollmann and A. L. Selman. Complexity measures for public-key cryptosystems. *SIAM Journal on Computing*, 17(2):309–335, 1988.

[23] S. Hallgren. Polynomial-time quantum algorithms for Pell's equation and the principal ideal problem. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, pages 653–658, 2002.

[24] S. Hallgren, C. Moore, M. Rötteler, A. Russell, and P. Sen. Limitations of quantum coset states for graph isomorphism. In *Proceedings of the 38th ACM Symposium on Theory of Computing*, 2006. To appear. See also quant-ph/0511148 and quant-ph/0511149.

[25] S. Hallgren, A. Russell, and A. Ta-Shma. The hidden subgroup problem and quantum computation using group representations. *SIAM Journal on Computing*, 32(4):916–934, 2003.

[26] R. Impagliazzo and M. Naor. Efficient cryptographic schemes provably as secure as subset sum. *Journal of Cryptology*, 9(4):199–216, 1996.

[27] J. Kempe and A. Shalev. The hidden subgroup problem and permutation group theory. In *Proceedings of the 16th ACM-SIAM Symposium on Discrete Algorithms*, pages 1118–1125, 2005.

[28] A. Kitaev. Quantum measurements and the abelian stabilizer problem. quant-ph/9511026, 1995.

[29] H. Kobayashi. Non-interactive quantum perfect and statistical zero-knowledge. In *Proceedings of the 14th Annual International Conference on Algorithms and Computation*, LNCS 2906, pages 178–188. Springer, 2003.

[30] J. Köbler, U. Schöning, and J. Torán. *The Graph Isomorphism Problem: Its Structural Complexity*. Birkhäuser Boston Inc., 1993.

[31] G. Kuperberg. A subexponential-time quantum algorithm for the dihedral hidden subgroup problem. *SIAM Journal on Computing*, 35(1):170–188, 2005.

[32] H.-K. Lo and H. F. Chau. Is quantum bit commitment really possible? *Physical Review Letters*, 78(17):3410–3413, 1997.

[33] E. M. Luks. Permutation groups and polynomial-time computation. In L. Finklestein and W. M. Kantor, editors, *Groups and Computation*, DIMACS series in Discrete Mathematics and Theoretical Computer Science, Vol. 5, pages 139–175. American Mathematical Society, 1993.

[34] D. Mayers. Unconditionally secure quantum bit commitment is impossible. *Physical Review Letters*, 78(17):3414–3417, 1997.

[35] D. Mayers. Unconditional security in quantum cryptography. *Journal of the ACM*, 48(3):351–406, 2001.

[36] D. Micciancio and O. Regev. Worst-case to average-case reductions based on gaussian measure. In *Proceedings of the 45th IEEE Symposium on Foundations of Computer Science*, pages 372–381, 2004.

[37] C. Moore and A. Russell. The symmetric group defies strong Fourier sampling: Part II. quant-ph/0501066, 2005.

[38] C. Moore, D. Rockmore, A. Russell, and L. J. Schulman. The hidden subgroup problem in affine groups: basis selection in Fourier sampling. In *Proceedings of the 15th ACM-SIAM Symposium on Discrete Algorithms*, pages 1106–1115, 2004.

[39] C. Moore, A. Russell, and L. J. Schulman. The symmetric group defies strong Fourier sampling. In *Proceedings of the 46th IEEE Symposium on Foundations of Computer Science*, pages 479–490, 2005. See also quant-ph/0501056 and quant-ph/0501066.

[40] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information.* Cambridge University Press, 2000.

[41] T. Okamoto, K. Tanaka, and S. Uchiyama. Quantum public-key cryptosystems. In *Advances in Cryptology - CRYPTO 2000*, LNCS 1880, pages 147–165. Springer, 2000.

[42] O. Regev. New lattice-based cryptographic constructions. *Journal of the ACM*, 51(6):899–942, 2004.

[43] O. Regev. Quantum computation and lattice problems. *SIAM Journal on Computing*, 33(3):738–760, 2004.

[44] U. Schöning. Graph isomorphism is in the low hierarchy. *Journal of Computer and System Sciences*, 37:312–323, 1988.

[45] P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, 1997.

[46] M. Tompa and H. Woll. Random self-reducibility and zero knowledge interactive proofs of possession of information. In *Proceedings of the 28th IEEE Symposium on Foundations of Computer Science*, pages 472–482, 1987.

[47] J. Watrous. Limits on the power of quantum statistical zero-knowledge. In *Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science*, pages 459–468, 2002.

[48] A. C.-C. Yao. Theory and applications of trapdoor functions (extended abstract). In *Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science*, pages 80–91, 1982.

Appendix: Reduction from GA to UniqueGA$_{ff}$

In this Appendix, we prove Lemma 2.6. Earlier, Köbler et al. [30] proved the polynomial-time Turing equivalence between GA and UniqueGA. We first review their reduction and then explain how to modify it to obtain the reduction from GA to UniqueGA$_{ff}$. Note that the reduction from UniqueGA$_{ff}$ to GA is trivial.

We begin with a technical tool and notations necessary to describe the reduction of Köbler et al. The reduction of Köbler et al. uses a technical tool called a *label* to distinguish each node of a given graph G from the others. The label j attached to node i consists of two chains, one of which is of length $2n+3$ connected to node i and the other is of length j connected to the $n+2$-nd node of the first chain. (See Figure 2.)

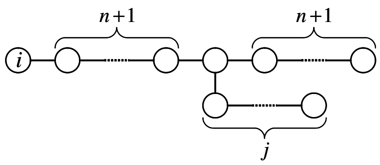

Figure 2: label

Note that the total size of the label j is $2n+j+3$. Let $G_{[i]}$ denote the graph obtained from G by attaching label 1 to node i. Similarly, $G_{[i_1,...,i_j]}$ is defined as the graph with labels $1,...,j$ respectively attached to nodes $i_1,...,i_j$. Note that any automorphism of $G_{[i]}$ maps the node i into itself and that any label adds no new automorphism into the modified graph. Let $Aut(G)$ be the automorphism group of the graph G and let $Aut(G)_{[1,...,i]}$ be the point-wise stabilizer of $\{1,...,i\}$ in $Aut(G)$, i.e., $Aut(G)_{[1,...,i]} = \{\sigma \in Aut(G) : \forall j \in \{1,...,i\}[\sigma(j) = j]\}$.

Köbler et al. proved the following theorem. For our later use, we give its proof.

Theorem 5.1 [30, Theorem 1.31] GA is polynomial-time Turing reducible to UniqueGA.

Proof. Given an oracle $\mathcal{O}$ for UniqueGA, the following algorithm solves GA in polynomial time. Let G be any given instance of GA.

(U1) Repeat (U2)-(U3) for each i starting with n down to 1.
(U2) Repeat (U3) for each j ranging from $i+1$ to n.
(U3) Invoke $\mathcal{O}$ with input graph $G_{[1,...,i-1,i]} \cup G_{[1,...,i-1,j]}$. If the outcome of $\mathcal{O}$ is YES, output YES and halt.
(U4) Output NO.

If G is an "YES" instance, there is at least one non-trivial automorphism. Take the largest number $i \in \{1,...,n\}$ such that there exists a number $j \in \{1,...,n\}$ and a non-trivial automorphism $\pi \in Aut(G)_{[1,...,i]}$ for which $\pi(i) = j$ and $i \neq j$. We claim that there is exactly one such non-trivial automorphism. This is seen as follows. First, note that $Aut(G)_{[1,...,i-1]}$ is expressed as $Aut(G)_{[1,...,i-1]} = \pi_1 Aut(G)_{[1,...,i]} + \cdots + \pi_d Aut(G)_{[1,...,i]}$. For any two distinct cosets $\pi_s Aut(G)_{[1,...,i]}$ and $\pi_t Aut(G)_{[1,...,i]}$ and for any two automorphisms $\sigma \in \pi_s Aut(G)_{[1,...,i]}$ and $\sigma' \in \pi_t Aut(G)_{[1,...,i]}$, it holds that $\sigma(i) \neq \sigma'(i)$. Since $|Aut(G)_{[1,...,i]}| = 1$ and there exists the unique coset $\pi_k Aut(G)$ such that $\sigma(i) = j$ for any $\sigma \in \pi_k Aut(G)$ by the definition of i, we obtain $|\pi_k Aut(G)_{[1,...,i]}| = 1$. This implies that the non-trivial automorphism π is unique. Note that the unique non-trivial automorphism interchanges two subgraphs $G_{[1,...,i-1,i]}$ and $G_{[1,...,i-1,j]}$. Therefore, the above algorithm successfully outputs YES at Step (U3).

On the contrary, if G is a "NO" instance, then for every distinct i and j, the modified graph has no non-trivial automorphism. Thus, the above algorithm correctly rejects such a graph G. □

Finally, we describe the reduction from GA to UniqueGA$_{\mathit{ff}}$ by slightly modifying the reduction given in the above proof.

Lemma 5.2 GA is polynomial-time Turing reducible to UniqueGA$_{\mathit{ff}}$.

Proof. We only need to change the number of nodes to invoke oracle UniqueGA$_{\mathit{ff}}$ in (U3). For such a change, we first modify the size of each label. Since the number m of all nodes $G_{[1,\ldots,i-1,i]} \cup G_{[1,\ldots,i-1,j]}$ is even, if there is no k such that $m = 2(2k+1)$ then we add one more node appropriately to the original labels. We then attach our modified labels of length $2n+i+4$ and $2n+j+4$ to nodes i and j, respectively. Obviously, this modified graph satisfies the promise of UniqueGA$_{\mathit{ff}}$. Our algorithm therefore works correctly for any instance of GA. □