CS-E4340: Cryptography **I-II 2022/2023**

Lecture 8: Introduction to Lattice-based Cryptography

Lecturer: Russell W.F. Lai

Abstract

This lecture aims to:

- introduce the two main family of assumptions – short integer solution (SIS) and learning with errors (LWE) – used in lattice-based cryptography,
- introduce the definition of public-key encryption (PKE),
- explain a construction of collision-resistant hash functions from the SIS assumption, and
- explain two constructions of PKE from the LWE assumption.

8.1 Background

Lattice-based cryptography is the use of conjectured hard problems over lattices (discrete additive subgroups of $\mathbb{R}^n$) as the foundation for secure cryptographic systems. While lattice algorithms have long been used for cryptanalysis (i.e. attacking cryptographic constructions), the constructive use of lattices in cryptography began in 1996 with the seminal work of Ajtai [Ajt96]. In this work, Ajtai introduced the Short Integer Solution (SIS) problem, showed that its average-case hardness can be reduced from the worst-case hardness of lattice problems, and constructed one-way functions based on its conjectured hardness. Fast forward to 2005, the Learning with Errors (LWE) problem was introduced by the important work of Regev [Reg05]. Syntactically similar to the SIS problem, the LWE problem enjoys similar worst-case-to-average-case hardness reductions. However, different from the SIS problem which so-far is only useful for constructing "minicrypt" [Imp95] primitives[1], efficient public-key encryption can be efficiently constructed based on the conjectured hardness of the LWE problem. The SIS and LWE problems are still heavily used as sources of hardness for constructing secure cryptographic systems today.

A few notable features of lattice-based cryptography are as follows.

- (Conjectured) Post-Quantum Security: Unlike widely deploy cryptographic systems based on the hardness of the RSA [RSA78] or discrete logarithm (e.g. [DH76]) problems, which are efficiently solvable by quantum computers due to Shor's algorithm [Sho97], quantum algorithms do not seem to solve lattice problems better than their classical counterparts.
- Security from Worst-case Hardness: The security of a cryptographic system is usually based on an assumption that certain class of computational problems is hard on average. In contrast, as mentioned above, the security of lattice-based cryptographic systems can often be proven assuming that at least one (e.g. the hardest) member of a class of lattice problems is hard.
- Algorithmic Simplicity and Parallelism: Operations in lattice-based cryptographic systems often boil down to modular arithmetic with relatively small moduli, which are highly parallelisable.

[1] Minicrypt = anything that can be constructed from onw-way functions.

8.2 Notation

Throughout this and the next lectures, we will heavily make use of the following notation.

- $\mathbb{Z}$: Ring of rational integers, i.e. $\mathbb{Z} \coloneqq \{\ldots, -1, 0, 1, \ldots\}$
- $[a, b]$: The discrete interval $\{a, a+1, \ldots, b\} \subseteq \mathbb{Z}$ for $a, b \in \mathbb{Z}$.
- $\mathbb{Z}_q \coloneqq \mathbb{Z}/q\mathbb{Z}$: Ring of rational integers modulo $q \in \mathbb{N}$, represented by $[-\lceil q/2 \rceil + 1, \lfloor q/2 \rfloor]$. We will abuse the notation and not distinguish $\mathbb{Z}_q$ and $[-\lceil q/2 \rceil + 1, \lfloor q/2 \rfloor]$. In this course, we always assume that q is prime, so that $\mathbb{Z}_q$ is a finite field.
- $\mathbf{A}$, $\mathbf{x}$: We will use bold capital letters to denote matrices, and bold lowercase letters to denote vectors.
- x_i: For a vector $\mathbf{x} \in \mathbb{Z}^n$, we write x_i for the i-th entry of $\mathbf{x}$.
- $\|\cdot\|$: The infinity norm over $\mathbb{R}^n$ induced to $\mathbb{Z}^n$, i.e. for $\mathbf{x} \in \mathbb{Z}^n$, $\|\mathbf{x}\| = \max_i |x_i|$. When taking the norm of a vector in $\mathbb{Z}_q^n$, we mean to take the norm of its representation in $\mathbb{Z}^n$.
- χ: A distribution over $\mathbb{Z}$. Typically χ is a zero-mean distribution supported by low-norm integers. In this course, we always consider χ to be the uniform distribution over $\mathbb{Z}_\beta$ for some $\beta \in \mathbb{Z}$. In the literature, χ is usually set to be a discrete Gaussian distribution for nicer hardness reductions.

8.3 Short Integer Solution (SIS) and Learning with Errors (LWE)

Suppose we are given a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ which is wide (i.e. $m > n$) and full rank (i.e. the rows of $\mathbf{A}$ are linearly independent). There are two basic computational problems associated to $\mathbf{A}$:

1. Preimage problem: Given an image vector $\mathbf{v} \in \mathbb{Z}_q^n$, express $\mathbf{v}$ as a linear combination of the columns of $\mathbf{A}$, i.e. find a preimage $\mathbf{u} \in \mathbb{Z}^m$ such that $\mathbf{A} \cdot \mathbf{u} = \mathbf{v} \bmod q$.
2. Decoding problem: Given a codeword vector $\mathbf{b}$ spanned by the rows of $\mathbf{A}$ recover the linear combination, i.e. given $\mathbf{b}^\intercal = \mathbf{s}^\intercal \cdot \mathbf{A} \bmod q$ for some $\mathbf{s} \in \mathbb{Z}_q^n$ recover $\mathbf{s}$.

Since $\mathbb{Z}_q$ is a field and $\mathbf{A}$ is full rank, a preimage problem always has a solution and the solution of a decoding problem is always unique. Furthermore, both problems can be solved efficiently, e.g. using Gaussian elimination. However, if we restrict the solutions of a preimage problem to fall into a small set $S \subseteq \mathbb{Z}^m$, or modify a decoding problem so that $\mathbf{b}^\intercal = \mathbf{s}^\intercal \cdot \mathbf{A} + \mathbf{e}^\intercal \bmod q$ is a noisy linear combination of the rows of $\mathbf{A}$, then the problems appear to be much harder.

8.3.1 Short Integer Solution (SIS)

The SIS problem [Ajt96] is the preimage problem with a random matrix $\mathbf{A} \leftarrow\!\!\$\ \mathbb{Z}_q^{n \times m}$ and a bounded-norm constraint.

Definition 8.1 (Short Integer Solution (SIS) Problem)**.** *Let $n, m, q, \beta \in \mathbb{N}$ with $n \leq m$ and $\beta \leq q$, and $\mathbf{v} \in \mathbb{Z}_q^n$. The $\mathsf{SIS}_{n,m,q,\beta,\mathbf{v}}$ problem is the following: Given a uniformly random matrix $\mathbf{A} \leftarrow\!\!\$\ \mathbb{Z}_q^{n \times m}$, find a vector $\mathbf{u} \in \mathbb{Z}^m$ satisfying*

$$\mathbf{A} \cdot \mathbf{u} = \mathbf{v} \bmod q \qquad \textit{and} \qquad \|\mathbf{u}\| \leq \beta.$$

Furthermore, if $\mathbf{v} = \mathbf{0}$, then $\mathbf{u} \neq \mathbf{0}$.

Definition 8.2 (Short Integer Solution (SIS) Assumption)**.** *Let $n, m, \log q, \log \beta \in \mathsf{poly}(\lambda)$ with $n \leq m$ and $\beta \leq q$, and $\mathbf{v} \in \mathbb{Z}_q^n$. The $\mathsf{SIS}_{n,m,q,\beta,\mathbf{v}}$ assumption states that for any PPT adversary $\mathcal{A}$ the $\mathsf{SIS}_{n,m,q,\beta,\mathbf{v}}$ problem is hard. That is, for any PPT adversary $\mathcal{A}$, it holds that*

$$\Pr\left[\begin{array}{l|l} \mathbf{A}\cdot\mathbf{u} = \mathbf{v} \bmod q & \mathbf{A} \leftarrow\!\$\ \mathbb{Z}_q^{n\times m} \\ \wedge\ 0 < \|\mathbf{u}\| \leq \beta & \mathbf{u} \leftarrow \mathcal{A}(\mathbf{A}, \mathbf{v}) \end{array}\right] \leq \mathsf{negl}(\lambda).$$

Parameters. It is easy to verify that the SIS problem becomes harder as the number of rows n increases (more constraints) and as the norm bound β decreases (more restrictive constraint). Likewise, the problem becomes easier as the number of columns m increases (higher degree of freedom). Typically, we can think of n as the security parameter, β as a polynomial in n, $q = \Theta(\beta \cdot n \cdot \log n)$, and $m = \Theta(n \cdot \log_\beta q)$.[2]

Variants. We mention a few notable variations of the SIS problem.

- In the literature, the SIS problem usually refers to the special case where $\mathbf{v} = \mathbf{0} \in \mathbb{Z}_q^n$. In this case, we drop $\mathbf{v}$ from the subscript and denote the problem by $\mathsf{SIS}_{n,m,q,\beta}$. The case where $\mathbf{v} \neq \mathbf{0}$ is usually referred to as the inhomogeneous short integer solution (ISIS) problem.
- The SIS problem can be defined with respect to any norm $\|\cdot\|$. Most commonly considered ones are the Euclidean norm $\|\mathbf{u}\| = \|\mathbf{u}\|_2 = \sqrt{\sum_i |u_i|^2}$ and the infinity norm $\|\mathbf{u}\| = \|\mathbf{u}\|_\infty = \max_i |u_i|$.
- The SIS problem can be defined over any ring $\mathcal{R}$ equipped with a norm $\|\cdot\|$, instead of $\mathbb{Z}$.

8.3.2 Learning with Errors (LWE)

There are two versions of the LWE problem [Reg05] – search and decision.

Search-LWE. The Search-LWE problem is the decoding problem with a random matrix $\mathbf{A} \leftarrow\!\$\ \mathbb{Z}_q^{n\times m}$ and where each entry of the noise vector $\mathbf{e} \leftarrow\!\$\ \chi^m$ is sampled independently from some noise distribution χ.

Definition 8.3 (Search-Learning with Errors (LWE) Problem)**.** *Let $n, m, q \in \mathbb{N}$ and χ be a distribution over $\mathbb{Z}$. The Search-$\mathsf{LWE}_{n,m,q,\chi}$ problem is the following: Let $\mathbf{A} \leftarrow\!\$\ \mathbb{Z}_q^{n\times m}$ and $\mathbf{s} \leftarrow\!\$\ \mathbb{Z}_q^n$ be uniformly random, and $\mathbf{e} \leftarrow\!\$\ \chi^m$ be consisting of m i.i.d. samples from the distribution χ. Let $\mathbf{b} \in \mathbb{Z}_q^m$ be computed as*

$$\mathbf{b}^\mathsf{T} \coloneqq \mathbf{s}^\mathsf{T}\cdot\mathbf{A} + \mathbf{e}^\mathsf{T} \bmod q.$$

Given $(\mathbf{A}, \mathbf{b})$, recover $\mathbf{s}$.

Definition 8.4 (Search-Learning with Errors (LWE) Assumption)**.** *Let $n, m, \log q \in \mathsf{poly}(\lambda)$ with $n \leq m$ and χ be a distribution over $\mathbb{Z}$ parametrised by λ. The Search-$\mathsf{LWE}_{n,m,q,\chi}$ assumption states that for any PPT adversary $\mathcal{A}$ the Search-$\mathsf{LWE}_{n,m,q,\chi}$ problem is hard. That is, for any PPT adversary $\mathcal{A}$, it holds that*

$$\Pr\left[\mathbf{s}^* = \mathbf{s} \;\middle|\; \begin{array}{l} \mathbf{A} \leftarrow\!\$\ \mathbb{Z}_q^{n\times m} \\ \mathbf{s} \leftarrow\!\$\ \mathbb{Z}_q^n,\ \mathbf{e} \leftarrow\!\$\ \chi^m \\ \mathbf{b}^\mathsf{T} \coloneqq \mathbf{s}^\mathsf{T}\cdot\mathbf{A} + \mathbf{e}^\mathsf{T} \bmod q \\ \mathbf{s}^* \leftarrow \mathcal{A}(\mathbf{A}, \mathbf{b}) \end{array}\right] \leq \mathsf{negl}(\lambda).$$

[2] Warning: This is not meant to be a recommendation for parameters selection.

The Search-LWE problem can also be interpreted as learning a secret linear function $\langle \mathbf{s}, \cdot \rangle \bmod q$ given m noisy samples $\{(\mathbf{a}_i, b_i)\}_{i=1}^m$. Packing $\{(\mathbf{a}_i, b_i)\}_{i=1}^m$ into columns gives $(\mathbf{A}, \mathbf{b}^\mathsf{T})$.

With a probabilistic argument and an appropriate choice of χ, one could show that Search-$\mathsf{LWE}_{n,m,q,\chi}$ has a unique solution except with probability negligible in n.[3]

Decision-LWE. Next, we introduce the Decision-LWE problem, which asks to distinguish whether the samples $\{(\mathbf{a}_i, b_i)\}_{i=1}^m$ are uniformly random or are of the form $b_i = \mathbf{s}^\mathsf{T} \cdot \mathbf{a}_i + e_i \bmod q$.

Definition 8.5 (Decision-Learning with Errors (LWE) Problem)**.** *Let $n, m, q \in \mathbb{N}$ and χ be a distribution over $\mathbb{Z}$. The Decision-$\mathsf{LWE}_{n,m,q,\chi}$ problem is the following: Let $\mathbf{A} \leftarrow\!\$\ \mathbb{Z}_q^{n \times m}$ and $\mathbf{s} \leftarrow\!\$\ \mathbb{Z}_q^n$ be uniformly random, and $\mathbf{e} \leftarrow\!\$\ \chi^m$ be consisting of m i.i.d. samples from the distribution χ. Let $\mathbf{b} \in \mathbb{Z}_q^m$ be either computed as*

$$\mathbf{b}^\mathsf{T} \coloneqq \mathbf{s}^\mathsf{T} \cdot \mathbf{A} + \mathbf{e}^\mathsf{T} \bmod q$$

or sampled uniformly at random as $\mathbf{b} \leftarrow\!\$\ \mathbb{Z}_q^m$. Given $(\mathbf{A}, \mathbf{b})$, distinguish how $\mathbf{b}$ was generated.

Definition 8.6 (Decision-Learning with Errors (LWE) Assumption)**.** *Let $n, m, \log q \in \mathsf{poly}(\lambda)$ with $n \leq m$ and χ be a distribution over $\mathbb{Z}$ parametrised by λ. The Decision-$\mathsf{LWE}_{n,m,q,\chi}$ assumption states that for any PPT adversary $\mathcal{A}$ the Decision-$\mathsf{LWE}_{n,m,q,\chi}$ problem is hard. That is, for any PPT adversary $\mathcal{A}$, it holds that*

$$\left| \Pr\left[b = 1 \;\middle|\; \begin{array}{l} \mathbf{A} \leftarrow\!\$\ \mathbb{Z}_q^{n \times m} \\ \mathbf{s} \leftarrow\!\$\ \mathbb{Z}_q^n,\ \mathbf{e} \leftarrow\!\$\ \chi^m \\ \mathbf{b}^\mathsf{T} \coloneqq \mathbf{s}^\mathsf{T} \cdot \mathbf{A} + \mathbf{e}^\mathsf{T} \bmod q \\ b \leftarrow \mathcal{A}(\mathbf{A}, \mathbf{b}) \end{array} \right] - \Pr\left[b = 1 \;\middle|\; \begin{array}{l} \mathbf{A} \leftarrow\!\$\ \mathbb{Z}_q^{n \times m} \\ \mathbf{b} \leftarrow\!\$\ \mathbb{Z}_q^m \\ b \leftarrow \mathcal{A}(\mathbf{A}, \mathbf{b}) \end{array} \right] \right| \leq \mathsf{negl}(\lambda).$$

Parameters. It is easy to verify that the LWE problems become harder as the number of rows n increases (longer secret) but easier as the number of columns m increases (more samples). Another factor which affects hardness is the ratio between the modulus q and the norm of the noise vector $\mathbf{e}$. The LWE problems become easier as this ratio increases.

Variants. As for SIS, we mention a few notable variations of the LWE problems.

- The LWE secret $\mathbf{s}$ can be sampled from a different distribution other than the uniform distribution over $\mathbb{Z}_q^n$. A particularly interesting choice is to sample $\mathbf{s} \leftarrow\!\$\ \chi^n$. This is called the "normal form" of LWE.
- The LWE problems can be defined over any ring $\mathcal{R}$ over which a meaningful error distribution χ can be defined, instead of $\mathbb{Z}$.

Relation between Search- and Decision-LWE. It is clear that Decision-LWE is easier than Search-LWE, since there is a trivial decision-to-search reduction: Use the Search-$\mathsf{LWE}_{n,m,q,\chi}$ oracle to find $\mathbf{s}$. If the oracle succeeds, then the given samples are LWE sampled. Otherwise, they are uniformly random samples. It turns out that Search-LWE is also not much harder than Decision-LWE, i.e. there exists a search-to-decision reduction. However, we will not look into this in this course.

[3] Consider $\mathbf{b}^\mathsf{T} = \mathbf{s}^\mathsf{T} \cdot \mathbf{A} + \mathbf{e}^\mathsf{T} \bmod q = \bar{\mathbf{s}}^\mathsf{T} \cdot \mathbf{A} + \bar{\mathbf{e}}^\mathsf{T} \bmod q$ which implies $\mathbf{A}^\mathsf{T} \cdot (\mathbf{s} - \bar{\mathbf{s}}) = \bar{\mathbf{e}} - \mathbf{e} \bmod q$.

8.4 Basic Cryptographic Applications

We study some basic applications of the SIS and LWE assumptions. As mentioned in the background section, the SIS assumption is typically used to build minicrypt [Imp95] primitives (more efficiently than generically from one-way functions), while the LWE assumption can be used to build a large variety of public-key primitives. As examples, we will look at Ajtai's construction [Ajt96; GGH96] of collision-resistant hash functions from the SIS assumption, Regev's construction [Reg05] of public-key encryption from the LWE assumption, and the dual-Regev public-key encryption scheme from the LWE assumption by Gentry, Peikert, and Vaikuntanathan [GPV08].

8.4.1 Ajtai's Collision-Resistant Hash Functions

From the SIS assumption, Ajtai [Ajt96] constructed the following one-way functions family. The family is parametrised by $n, m, \log q, \log \beta \in \mathsf{poly}(\lambda)$ and a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, which defines the function

$$\begin{aligned} f_{\mathbf{A}} : \mathbb{Z}_\beta^m &\to \mathbb{Z}_q^n \\ \mathbf{x} &\mapsto \mathbf{A} \cdot \mathbf{x} \bmod q. \end{aligned}$$

The (family) of function(s) $f_{\mathbf{A}}$ is sometimes called the "SIS function". Later, it was realised [GGH96] that Ajtai's functions family is actually collision-resistant.

Theorem 8.7. *If the $\mathsf{SIS}_{n,m,q,\beta}$ assumption holds, then $\{f_{\mathbf{A}} \mid \mathbf{A} \in \mathbb{Z}_q^{n \times m}\}$ is a family of collision-resistant hash functions, i.e. for any PPT adversary $\mathcal{A}$ it holds that*

$$\Pr\left[\{\mathbf{x}, \mathbf{y}\} \subseteq \mathbb{Z}_\beta^m \wedge f_{\mathbf{A}}(\mathbf{x}) = f_{\mathbf{A}}(\mathbf{y}) \wedge \mathbf{x} \neq \mathbf{y} \;\middle|\; \begin{array}{l} \mathbf{A} \leftarrow\!\$\; \mathbb{Z}_q^{n \times m} \\ (\mathbf{x}, \mathbf{y}) \leftarrow \mathcal{A}(\mathbf{A}) \end{array}\right] \leq \mathsf{negl}(\lambda).$$

Proof. If there exists a PPT adversary $\mathcal{A}$ which breaks the collision-resistance of the hash functions, we construct a PPT reduction which solves the $\mathsf{SIS}_{n,m,q,\beta}$ problem. Our reduction gets a uniformly random instance $\mathbf{A}$ of $\mathsf{SIS}_{n,m,q,\beta}$ and passes it to $\mathcal{A}$, who returns $(\mathbf{x}, \mathbf{y})$. The reduction then returns $\mathbf{u} \coloneqq \mathbf{x} - \mathbf{y}$. By assumption, we have $\{\mathbf{x}, \mathbf{y}\} \subseteq \mathbb{Z}_\beta^m$, $f_{\mathbf{A}}(\mathbf{x}) = f_{\mathbf{A}}(\mathbf{y})$, and $\mathbf{x} \neq \mathbf{y}$ with non-negligible probability. This means that, with non-negligible probability, the solution $\mathbf{u}$ satisfies $\|\mathbf{u}\| \leq \beta$, $\mathbf{A} \cdot \mathbf{u} = \mathbf{0} \bmod q$, and $\mathbf{u} \neq \mathbf{0}$, i.e. $\mathbf{u}$ is a valid solution to the $\mathsf{SIS}_{n,m,q,\beta}$ problem instance $\mathbf{A}$. □

Regularity. Another useful property of Ajtai's functions is that they exhibit good regularity properties: If there are sufficiently many columns, i.e. m is large enough, applying the function on uniformly random inputs give outputs which are close to uniform. Formally, the following is a special case of the leftover hash lemma [ILL89] which we state without proof.

Lemma 8.8 (Regularity/Leftover Hash Lemma [ILL89; GPV08])**.** *For $m > n \cdot \log_\beta q + \omega(\log n)$, the distributions*

$$\left\{(\mathbf{A}, \mathbf{y}) \;\middle|\; \begin{array}{l} \mathbf{A} \leftarrow\!\$\; \mathbb{Z}_q^{n \times m} \\ \mathbf{x} \leftarrow\!\$\; \mathbb{Z}_\beta^m \\ \mathbf{y} \coloneqq \mathbf{A} \cdot \mathbf{x} \bmod q \end{array}\right\} \qquad \textit{and} \qquad \left\{(\mathbf{A}, \mathbf{y}) \;\middle|\; \begin{array}{l} \mathbf{A} \leftarrow\!\$\; \mathbb{Z}_q^{n \times m} \\ \mathbf{y} \leftarrow\!\$\; \mathbb{Z}_q^n \end{array}\right\}$$

are statistically-close in n.

Note that for the SIS function to be compressing, which we want or collision-resistance is not too meaningful, we require $m > n \cdot \log_\beta q$, which almost coincides with the requirement in the leftover hash lemma (Lemma 8.8).

Normal Form of SIS. In the above construction, it takes $n \cdot m \cdot \log q$ bits to specify a hash function $f_{\mathbf{A}}$. It turns out that there is a way to slightly reduce the description size of the hash function to just $n \cdot (m-n) \cdot \log q$ bits without relying on stronger assumptions. Concretely, for a matrix $\bar{\mathbf{A}} \in \mathbb{Z}_q^{n \times (m-n)}$, define the function

$$\begin{aligned} \bar{f}_{\bar{\mathbf{A}}} : \mathbb{Z}_\beta^m &\to \mathbb{Z}_q^n \\ \mathbf{x} &\mapsto \left(\bar{\mathbf{A}} \| \mathbf{I}_n\right) \cdot \mathbf{x} \bmod q. \end{aligned}$$

The SIS problem where $\mathbf{A}$ is restricted to $\mathbf{A} = \left(\bar{\mathbf{A}} \| \mathbf{I}_n\right)$ is known as the (Hermite) normal form of SIS.

Theorem 8.9. *If the* $\mathsf{SIS}_{n,m,q,\beta}$ *assumption holds, then* $\left\{ \bar{f}_{\bar{\mathbf{A}}} \mid \bar{\mathbf{A}} \in \mathbb{Z}_q^{n \times (m-n)} \right\}$ *is a family of collision-resistant hash functions, i.e. for any PPT adversary* $\mathcal{A}$ *it holds that*

$$\Pr\left[\{\mathbf{x}, \mathbf{y}\} \subseteq \mathbb{Z}_\beta^m \wedge \bar{f}_{\bar{\mathbf{A}}}(\mathbf{x}) = \bar{f}_{\bar{\mathbf{A}}}(\mathbf{y}) \wedge \mathbf{x} \neq \mathbf{y} \;\middle|\; \begin{array}{l} \bar{\mathbf{A}} \leftarrow\!\!\$\; \mathbb{Z}_q^{n \times (m-n)} \\ (\mathbf{x}, \mathbf{y}) \leftarrow \mathcal{A}(\bar{\mathbf{A}}) \end{array} \right] \leq \mathsf{negl}(\lambda).$$

Proof. Instead of proving collision-resistance from scratch, we show a reduction from $\mathsf{SIS}_{n,m,q,\beta}$ to the normal form of $\mathsf{SIS}_{n,m,q,\beta}$. Let $\mathcal{A}$ be a PPT algorithm which solves the normal form of $\mathsf{SIS}_{n,m,q,\beta}$ with non-negligible probability. Our reduction gets a uniformly random instance $\mathbf{A}$ of $\mathsf{SIS}_{n,m,q,\beta}$ and does the following.

- Check that $\mathbf{A}$ is full-rank and abort if otherwise.
- Write $\mathbf{A} = (\mathbf{A}_0 \| \mathbf{A}_1)$ where $\mathbf{A}_0 \in \mathbb{Z}_q^{n \times (m-n)}$ and $\mathbf{A}_1 \in \mathbb{Z}_q^{n \times n}$. We can assume that $\mathbf{A}_1$ is invertible over $\mathbb{Z}_q$ without loss of generality.[4]
- Define $\bar{\mathbf{A}} \coloneqq \mathbf{A}_1^{-1} \cdot \mathbf{A}_0$.
- Pass $\bar{\mathbf{A}}$ to $\mathcal{A}$ and in return receive $\mathbf{u}$.
- Return $\mathbf{u}$.

We argue that $\mathbf{u}$ is a valid solution to the $\mathsf{SIS}_{n,m,q,\beta}$ problem instance $\mathbf{A}$ except with negligible probability.

First, from Lemma 8.10 which we will state and prove later, the probability that $\mathbf{A}$ is not full-rank is at most $\frac{n}{q^{m-n+1}}$, which is negligible in n. In what follows, assume that $\mathbf{A}$ is full-rank.

Next, we note that that $\bar{\mathbf{A}} \coloneqq \mathbf{A}_1^{-1} \cdot \mathbf{A}_0$ is a well-distributed normal-form $\mathsf{SIS}_{n,m,q,\beta}$ instance, because $\mathbf{A}_1^{-1}$ is invertible over $\mathbb{Z}_q$ and $\mathbf{A}_0$ is uniformly random over $\mathbb{Z}_q^{n \times (m-n)}$.

Finally, by assumption, we have $\left(\bar{\mathbf{A}} \| \mathbf{I}_n\right) \cdot \mathbf{u} = \mathbf{0} \bmod q$ and $\|\mathbf{u}\| \leq \beta$ with non-negligible probability. The former implies that $\mathbf{A} \cdot \mathbf{u} = \mathbf{0} \bmod q$. □

Lemma 8.10. *Let* $n, m, q \in \mathbb{N}$ *with* $m > n$. *It holds that* $\Pr\left[\mathbf{A} \text{ is full-rank} \;\middle|\; \mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}\right] \geq 1 - \frac{n}{q^{m-n+1}}$.

Proof. We show by counting the number of full-rank matrices in $\mathbb{Z}_q^{n \times m}$. There is only one choice of the first row of $\mathbf{A}$ which makes $\mathbf{A}$ not full-rank – the all-zero vector. Therefore, there are $q^m - 1$ choices for the first row. Fix any choice $\mathbf{a}_1$ of the first row. The second row cannot be a multiple of $\mathbf{a}_1$, else the matrix $\mathbf{A}$ is not full-rank. Therefore, there are $q^m - q$ choices for the second row. Continue in this way, we conclude that there are $q^m - q^{i-1}$ choices for the i-th row. Therefore, the number of full-rank matrices in $\mathbb{Z}_q^{n \times m}$ is

$$\prod_{i=1}^{n} (q^m - q^{i-1}) = q^{nm} \cdot \prod_{i=1}^{n} \left(1 - \frac{1}{q^{m-i+1}}\right) \geq q^{nm} \cdot \left(1 - \frac{1}{q^{m-n+1}}\right)^n \geq q^{nm} \cdot \left(1 - \frac{n}{q^{m-n+1}}\right).$$

A random matrix in $\mathbb{Z}_q^{n \times m}$ is therefore full-rank except with probability at most $\frac{n}{q^{m-n+1}}$. □

[4] If not, just permute the columns of $\mathbf{A}$.

8.4.2 Regev's Public-Key Encryption

Next, we study Regev's construction [Reg05] of public-key encryption (PKE) from the LWE assumption. Before that, let us formally define PKE. As one would expect, a PKE is similar to a symmetric-key encryption except that the key generation algorithm additionally outputs a public key and the encryption algorithm inputs the public key instead of the secret key.

Definition 8.11 (PKE)**.** *A public-key encryption (PKE) scheme for the message space $\mathcal{X}$ is a tuple of PPT algorithms* $(\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ *with the following syntax:*

- $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$*: The key generation algorithm generates a pair of public and secret keys* $(\mathsf{pk}, \mathsf{sk})$*.*
- $\mathsf{ctxt} \leftarrow \mathsf{Enc}(\mathsf{pk}, x)$*: The encryption algorithm takes a public key* pk *and a message $x \in \mathcal{X}$ and outputs a ciphertext* ctxt*.*
- $x \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathsf{ctxt})$*: The decryption algorithm takes a secret key* sk *and a ciphertext* ctxt *and outputs a message x.*

Definition 8.12 (Correctness)**.** *A PKE scheme* $(\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ *is correct if for any $\lambda \in \mathbb{N}$, any* $(\mathsf{pk}, \mathsf{sk}) \in \mathsf{KGen}(1^\lambda)$*, and any $x \in \mathcal{X}$, it holds that*

$$\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, x)) = x.$$

Similar to symmetric-key encryption, the basic notion of security of PKE is IND-CPA-security. The definition of IND-CPA-security of PKE is actually simpler than that of symmetric-key encryption because the encryption algorithm is now public-key, which means that the security experiment does not need to provide an encryption oracle to the adversary.

Definition 8.13 (IND-CPA)**.** *A PKE scheme $\Pi = (\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ has ciphertext indistinguishability under chosen-plaintext attacks (IND-CPA-secure) if for any (two-stage) PPT adversary $\mathcal{A}$ it holds that*

$$\left|\Pr\left[\text{IND-CPA}^0_{\Pi,\mathcal{A}}(1^\lambda) = 1\right] - \Pr\left[\text{IND-CPA}^1_{\Pi,\mathcal{A}}(1^\lambda) = 1\right]\right| \leq \mathsf{negl}(\lambda)$$

where the experiment $\text{IND-CPA}^b_{\Pi,\mathcal{A}}$ *for $b \in \{0,1\}$ is defined as follows:*

$\text{IND-CPA}^b_{\Pi,\mathcal{A}}(1^\lambda)$

$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$

$(x_0, x_1) \leftarrow \mathcal{A}(\mathsf{pk})$

$\mathsf{ctxt}^* \leftarrow \mathsf{Enc}(\mathsf{pk}, x_b)$

$b' \leftarrow \mathcal{A}(\mathsf{ctxt}^*)$

return b'

Scheme Description. We are now ready to describe Regev's construction of public-key encryption. In short, the public key consists of m LWE samples, and the secret key is an LWE secret. To encrypt, perform a random linear combination of the LWE samples to produce a new one, and use it to one-time-pad (an encoding of) the message. To decrypt, use the LWE secret to recover the LWE sample used for the one-time-pad. Formally, let $n, m, \log q, \log \beta \in \mathsf{poly}(\lambda)$ and let χ be the uniform distribution over $\mathbb{Z}_\beta$. Regev's construction is as follows.

Key Generation $\mathsf{KGen}(1^\lambda)$:

- Sample a uniformly random matrix $\mathbf{A} \leftarrow\!\!\$\ \mathbb{Z}_q^{n\times m}$, an LWE secret vector $\mathbf{s} \leftarrow\!\!\$\ \mathbb{Z}_q^n$, and a short noise vector $\mathbf{e} \leftarrow\!\!\$\ \chi^m$.
- Compute $\mathbf{b}^\mathsf{T} = \mathbf{s}^\mathsf{T} \cdot \mathbf{A} + \mathbf{e}^\mathsf{T} \bmod q$.
- Output $\mathsf{pk} \coloneqq (\mathbf{A}, \mathbf{b})$ and $\mathsf{sk} \coloneqq \mathbf{s}$.

Encryption $\mathsf{Enc}(\mathsf{pk}, x \in \{0,1\})$:

- Sample short vector $\mathbf{r} \leftarrow\!\!\$\ \chi^m$.
- Set $\mathbf{c}_0 \coloneqq \mathbf{A} \cdot \mathbf{r} \bmod q$.
- Set $c_1 \coloneqq \mathbf{b}^\mathsf{T} \cdot \mathbf{r} + \lfloor q/2 \rfloor \cdot x \bmod q$.
- Output $\mathsf{ctxt} \coloneqq (\mathbf{c}_0, c_1)$.

Decryption $\mathsf{Dec}(\mathsf{sk}, \mathsf{ctxt})$:

- Compute $\bar{x} \coloneqq c_1 - \mathbf{s}^\mathsf{T} \cdot \mathbf{c}_0 \bmod q$.
- If $|\bar{x}| < q/4$, output 0. Else, output 1.

Analysis. Next, let us analyse the correctness and IND-CPA-security of Regev's scheme.

Theorem 8.14 (Correctness)**.** *If $q > m \cdot \beta^2 + 1$, Regev's public-key encryption scheme is correct.*

Proof. Notice that for any encryption $\mathsf{ctxt} \coloneqq (\mathbf{c}_0, c_1)$ of $x \in \{0,1\}$, we have

$$\begin{aligned}\bar{x} &= c_1 - \mathbf{s}^\mathsf{T} \cdot \mathbf{c}_0 \\ &= \mathbf{b}^\mathsf{T} \cdot \mathbf{r} + \lfloor q/2 \rfloor \cdot x - \mathbf{s}^\mathsf{T} \cdot \mathbf{A} \cdot \mathbf{r} \bmod q \\ &= \mathbf{e}^\mathsf{T} \cdot \mathbf{r} + \lfloor q/2 \rfloor \cdot x \bmod q.\end{aligned}$$

If $x = 0$, we have

$$\|\bar{x}\| = \left\|\mathbf{e}^\mathsf{T} \cdot \mathbf{r} \bmod q\right\| \leq m \cdot \beta^2/4 < q/4.$$

If $x = 1$, we have

$$\|\bar{x}\| = \left\|\mathbf{e}^\mathsf{T} \cdot \mathbf{r} + \lfloor q/2 \rfloor \bmod q\right\| \geq (q-1)/2 - m \cdot \beta^2/4 > q/4.$$

□

Theorem 8.15 (IND-CPA-Security)**.** *If $m = \Omega(n \log_\beta q)$ and the Decision-$\mathsf{LWE}_{n,m,q,\chi}$ assumption holds, then Regev's public-key encryption scheme is IND-CPA-secure.*

Proof. Let Π denote Regev's public-key encryption scheme and let $\mathcal{A}$ be any PPT adversary. We prove by a standard "hybrid argument". That means, we define a sequence of hybrid security experiments where the first is identical to $\text{IND-CPA}^0_{\Pi,\mathcal{A}}$ and the last is identical to $\text{IND-CPA}^1_{\Pi,\mathcal{A}}$, and show that any two consecutive experiments are computationally or statistically indistinguishable from each other. Consequently, we have that $\text{IND-CPA}^0_{\Pi,\mathcal{A}}$ and $\text{IND-CPA}^1_{\Pi,\mathcal{A}}$ are computationally indistinguishable from each other, as desired.

The sequence of hybrids are as follows.

- Hyb_0: This experiment is identical to $\text{IND-CPA}^0_{\Pi,\mathcal{A}}$.

- Hyb_1: This experiment is almost identical to Hyb_0, except that the public key $\mathsf{pk} = (\mathbf{A}, \mathbf{b})$ is replaced by a uniformly sampled one.
- Hyb_2: This experiment is almost identical to Hyb_1, except that the terms $\mathbf{A} \cdot \mathbf{r} \bmod q$ and $\mathbf{b}^\mathsf{T} \cdot \mathbf{r} \bmod q$ in the challenge ciphertext are replaced by uniformly sampled ones.
- Hyb_3: This experiment is almost identical to Hyb_2, except that the message being encrypted is changed from x_0 to x_1.
- Hyb_4: This experiment is almost identical to Hyb_3, except that the challenge ciphertext is computed as in $\mathsf{Enc}(\mathsf{pk}, x_1)$.
- Hyb_5: This experiment is almost identical to Hyb_4, except that the public key $\mathsf{pk} = (\mathbf{A}, \mathbf{b})$ is sampled as in $\mathsf{KGen}(1^\lambda)$. This experiment is identical to $\text{IND-CPA}^1_{\Pi,\mathcal{A}}$.

The explicit definition of the hybrids, with the procedures of the construction inline, are given below.

$\mathsf{Hyb}_0(1^\lambda)$	$\mathsf{Hyb}_1(1^\lambda)$	$\mathsf{Hyb}_2(1^\lambda)$
$\mathbf{A} \leftarrow\!\!\$\ \mathbb{Z}_q^{n \times m}$	$\mathbf{A} \leftarrow\!\!\$\ \mathbb{Z}_q^{n \times m}$	$\mathbf{A} \leftarrow\!\!\$\ \mathbb{Z}_q^{n \times m}$
$\mathbf{s} \leftarrow\!\!\$\ \mathbb{Z}_q^n$		
$\mathbf{e} \leftarrow\!\!\$\ \chi^m$		
$\mathbf{b}^\mathsf{T} = \mathbf{s}^\mathsf{T} \cdot \mathbf{A} + \mathbf{e}^\mathsf{T} \bmod q$	$\mathbf{b} \leftarrow\!\!\$\ \mathbb{Z}_q^m$	$\mathbf{b} \leftarrow\!\!\$\ \mathbb{Z}_q^m$
$(x_0, x_1) \leftarrow \mathcal{A}(\mathbf{A}, \mathbf{b})$	$(x_0, x_1) \leftarrow \mathcal{A}(\mathbf{A}, \mathbf{b})$	$(x_0, x_1) \leftarrow \mathcal{A}(\mathbf{A}, \mathbf{b})$
$\mathbf{r} \leftarrow\!\!\$\ \chi^m$	$\mathbf{r} \leftarrow\!\!\$\ \chi^m$	$\mathbf{d}_0 \leftarrow\!\!\$\ \mathbb{Z}_q^n,\ d_1 \leftarrow\!\!\$\ \mathbb{Z}_q$
$\mathbf{c}_0 \coloneqq \mathbf{A} \cdot \mathbf{r} \bmod q$	$\mathbf{c}_0 \coloneqq \mathbf{A} \cdot \mathbf{r} \bmod q$	$\mathbf{c}_0 \coloneqq \mathbf{d}_0 \bmod q$
$c_1 \coloneqq \mathbf{b}^\mathsf{T} \cdot \mathbf{r} + \lfloor q/2 \rfloor \cdot x_0 \bmod q$	$c_1 \coloneqq \mathbf{b}^\mathsf{T} \cdot \mathbf{r} + \lfloor q/2 \rfloor \cdot x_0 \bmod q$	$c_1 \coloneqq d_1 + \lfloor q/2 \rfloor \cdot x_0 \bmod q$
return $b \leftarrow \mathcal{A}(\mathbf{c}_0, c_1)$	**return** $b \leftarrow \mathcal{A}(\mathbf{c}_0, c_1)$	**return** $b \leftarrow \mathcal{A}(\mathbf{c}_0, c_1)$

$\mathsf{Hyb}_5(1^\lambda)$	$\mathsf{Hyb}_4(1^\lambda)$	$\mathsf{Hyb}_3(1^\lambda)$
$\mathbf{A} \leftarrow\!\!\$\ \mathbb{Z}_q^{n \times m}$	$\mathbf{A} \leftarrow\!\!\$\ \mathbb{Z}_q^{n \times m}$	$\mathbf{A} \leftarrow\!\!\$\ \mathbb{Z}_q^{n \times m}$
$\mathbf{s} \leftarrow\!\!\$\ \mathbb{Z}_q^n$		
$\mathbf{e} \leftarrow\!\!\$\ \chi^m$		
$\mathbf{b}^\mathsf{T} = \mathbf{s}^\mathsf{T} \cdot \mathbf{A} + \mathbf{e}^\mathsf{T} \bmod q$	$\mathbf{b} \leftarrow\!\!\$\ \mathbb{Z}_q^m$	$\mathbf{b} \leftarrow\!\!\$\ \mathbb{Z}_q^m$
$(x_0, x_1) \leftarrow \mathcal{A}(\mathbf{A}, \mathbf{b})$	$(x_0, x_1) \leftarrow \mathcal{A}(\mathbf{A}, \mathbf{b})$	$(x_0, x_1) \leftarrow \mathcal{A}(\mathbf{A}, \mathbf{b})$
$\mathbf{r} \leftarrow\!\!\$\ \chi^m$	$\mathbf{r} \leftarrow\!\!\$\ \chi^m$	$\mathbf{d}_0 \leftarrow\!\!\$\ \mathbb{Z}_q^n,\ d_1 \leftarrow\!\!\$\ \mathbb{Z}_q$
$\mathbf{c}_0 \coloneqq \mathbf{A} \cdot \mathbf{r} \bmod q$	$\mathbf{c}_0 \coloneqq \mathbf{A} \cdot \mathbf{r} \bmod q$	$\mathbf{c}_0 \coloneqq \mathbf{d}_0 \bmod q$
$c_1 \coloneqq \mathbf{b}^\mathsf{T} \cdot \mathbf{r} + \lfloor q/2 \rfloor \cdot x_1 \bmod q$	$c_1 \coloneqq \mathbf{b}^\mathsf{T} \cdot \mathbf{r} + \lfloor q/2 \rfloor \cdot x_1 \bmod q$	$c_1 \coloneqq d_1 + \lfloor q/2 \rfloor \cdot x_1 \bmod q$
return $b \leftarrow \mathcal{A}(\mathbf{c}_0, c_1)$	**return** $b \leftarrow \mathcal{A}(\mathbf{c}_0, c_1)$	**return** $b \leftarrow \mathcal{A}(\mathbf{c}_0, c_1)$

By the Decision-$\mathsf{LWE}_{n,m,q,\chi}$ assumption, we have

$$\left|\Pr\left[\mathsf{Hyb}_0(1^\lambda) = 1\right] - \Pr\left[\mathsf{Hyb}_1(1^\lambda) = 1\right]\right| \leq \mathsf{negl}(\lambda)$$

and

$$\left|\Pr\left[\mathsf{Hyb}_4(1^\lambda) = 1\right] - \Pr\left[\mathsf{Hyb}_5(1^\lambda) = 1\right]\right| \leq \mathsf{negl}(\lambda).$$

By the leftover hash lemma (Lemma 8.8), we have

$$\left|\Pr\left[\mathsf{Hyb}_1(1^\lambda) = 1\right] - \Pr\left[\mathsf{Hyb}_2(1^\lambda) = 1\right]\right| \leq \mathsf{negl}(\lambda)$$

and

$$\left|\Pr\left[\mathsf{Hyb}_3(1^\lambda)=1\right]-\Pr\left[\mathsf{Hyb}_4(1^\lambda)=1\right]\right|\leq \mathsf{negl}(\lambda).$$

Finally, we realise that $\Pr\left[\mathsf{Hyb}_2(1^\lambda)=1\right]=\Pr\left[\mathsf{Hyb}_3(1^\lambda)=1\right]$ because d_1 is uniformly random over $\mathbb{Z}_q$, which is an additive group, and adding anything to a random group element gives a random element. □

8.4.3 The Dual-Regev Public-Key Encryption

In Regev's construction, the public key consists of LWE samples and the secret key is the LWE secret, which is likely unique. A ciphertext is simply a short linear combination of the LWE samples. Correspondingly, in the security proof, the LWE assumption is first used to switch the public key to a fake one, and the leftover hash lemma is then used to argue the statistical uniformity of (fake) ciphertexts.

Gentry, Peikert, and Vaikuntanathan [GPV08] discovered an alternative way to construct a PKE which is in a sense "dual" to Regev's construction. In this dual-Regev construction, the public key is instead a SIS instance $(\mathbf{A},\mathbf{v})$ and the secret key is a solution of the SIS instance. To encrypt, an LWE sample is produced using the matrix $(\mathbf{A}\|\mathbf{v})$ which is then used to one-time-pad the message. Formally, let $n,m,\log q,\log\beta\in\mathsf{poly}(\lambda)$ and let χ be the uniform distribution over $\mathbb{Z}_\beta$. The dual-Regev construction is as follows.

Key Generation $\mathsf{KGen}(1^\lambda)$:

- Sample a uniformly random matrix $\mathbf{A}\leftarrow\!\!\$\ \mathbb{Z}_q^{n\times m}$ and a short vector $\mathbf{u}\leftarrow\!\!\$\ \chi^m$.
- Compute $\mathbf{v}\coloneqq\mathbf{A}\cdot\mathbf{u}\bmod q$.
- Output $\mathsf{pk}\coloneqq(\mathbf{A},\mathbf{v})$ and $\mathsf{sk}\coloneqq\mathbf{u}$.

Encryption $\mathsf{Enc}(\mathsf{pk},x\in\{0,1\})$:

- Sample an LWE secret vector $\mathbf{s}\leftarrow\!\!\$\ \mathbb{Z}_q^n$, a short noise vector $\mathbf{e}_0\leftarrow\!\!\$\ \chi^m$, and another short noise $e_1\leftarrow\!\!\$\ \chi$.
- Set $\mathbf{c}_0^\mathsf{T}\coloneqq\mathbf{s}^\mathsf{T}\cdot\mathbf{A}+\mathbf{e}_0^\mathsf{T}\bmod q$.
- Set $c_1\coloneqq\mathbf{s}^\mathsf{T}\cdot\mathbf{v}+e_1+\lfloor q/2\rfloor\cdot x\bmod q$.
- Output $\mathsf{ctxt}\coloneqq(\mathbf{c}_0,c_1)$.

Decryption $\mathsf{Dec}(\mathsf{sk},\mathsf{ctxt})$:

- Compute $\bar{x}\coloneqq c_1-\mathbf{c}_0^\mathsf{T}\cdot\mathbf{u}\bmod q$.
- If $|\bar{x}|<q/4$, output 0. Else, output 1.

For a sufficiently large m, the dual-Regev PKE has an interesting property that a public key could correspond to (possibly exponentially) many secret keys. This property turns out to be beneficial for the construction of more advanced encryption schemes.

We leave the correctness and security statements and their proofs as exercises.

References

[Ajt96] Miklós Ajtai. "Generating Hard Instances of Lattice Problems (Extended Abstract)". In: *28th ACM STOC*. ACM Press, May 1996, pp. 99–108. DOI: 10.1145/237814.237838.

[DH76] Whitfield Diffie and Martin E. Hellman. "New Directions in Cryptography". In: *IEEE Transactions on Information Theory* 22.6 (1976), pp. 644–654.

[GGH96] Oded Goldreich, Shafi Goldwasser, and Shai Halevi. *Collision-Free Hashing from Lattice Problems*. Cryptology ePrint Archive, Report 1996/009. https://eprint.iacr.org/1996/009. 1996.

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. "Trapdoors for hard lattices and new cryptographic constructions". In: *40th ACM STOC*. Ed. by Richard E. Ladner and Cynthia Dwork. ACM Press, May 2008, pp. 197–206. DOI: 10.1145/1374376.1374407.

[ILL89] Russell Impagliazzo, Leonid A. Levin, and Michael Luby. "Pseudo-random Generation from one-way functions (Extended Abstracts)". In: *21st ACM STOC*. ACM Press, May 1989, pp. 12–24. DOI: 10.1145/73007.73009.

[Imp95] R. Impagliazzo. "A personal view of average-case complexity". In: *Proceedings of Structure in Complexity Theory. Tenth Annual IEEE Conference*. 1995, pp. 134–147. DOI: 10.1109/SCT.1995.514853.

[Reg05] Oded Regev. "On lattices, learning with errors, random linear codes, and cryptography". In: *37th ACM STOC*. Ed. by Harold N. Gabow and Ronald Fagin. ACM Press, May 2005, pp. 84–93. DOI: 10.1145/1060590.1060603.

[RSA78] Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems". In: *Communications of the Association for Computing Machinery* 21.2 (1978), pp. 120–126.

[Sho97] Peter W. Shor. "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer". In: *SIAM J. Comput.* 26.5 (Oct. 1997), pp. 1484–1509. ISSN: 0097-5397. DOI: 10.1137/S0097539795293172. URL: https://doi.org/10.1137/S0097539795293172.