# On the Learning With Errors Problem

Shreya Sharma

June 2026

**Abstract**

Quantum computing has fundamentally altered modern cryptography by demonstrating that widely–used public–key systems based on integer factorization and the discrete logarithm problem are vulnerable to efficient quantum algorithms. This paper examines why those assumptions fail under quantum computation and why lattice-based cryptography, specifically the Learning with Errors (LWE) problem, has emerged as a leading post-quantum alternative. We begin by developing the mathematical foundations regarding finite fields, linear algebra over finite fields, and the Quantum Fourier Transform. We then contrast this with systems of noisy linear equations, demonstrating that the introduction of small random errors transforms an efficiently solvable linear algebra problem into one with no known efficient classical or quantum solution. The geometric interpretation of LWE is developed through lattices and the Bounded Distance Decoding problem, followed by Regev's worst-case-to-average-case reduction associating the average-case hardness of LWE to worst-case lattice problems such as GapSVP and SIVP. By indicating progression from exact algebraic structre to approximate geometric hardness, this paper explains the mathematical foundations and security rationale for one of the principal candidates for post-quantum cryptography.

## 1 Introduction

Modern cryptography rests on computational asymmetry: an operation efficient to evaluate and, conjecturally, intractable to invert. Classically, this asymmetry originates from number theory, namely, integer factorization and the discrete logarithm problem, upon which the Rivest-Shamir-Adleman (RSA) and Elliptic Curve Cryptography (ECC) encryption algorithms depend. The advent of quantum computation removed the basis for this confidence. Let us begin, perhaps surprisingly for a paper concerning linear algebra, with quantum computers. The reason will become clear shortly; it is the specific aspect of the quantum threat which informs us around what mathematical object could plausibly replace the broken cryptographic assumptions and thus motivates all that follows. Quantum computation is, essentially, linear algebra over a complex vector space that grows exponentially in size [NC00]. To be precise about this, we need a few definitions. We will use Dirac notation, otherwise called *ket notation* after the symbol $|\cdot\rangle$, which is the standard representation of quantum mechanics. A single classical bit is a physical device that takes one of two values: 0 or 1. A quantum bit, or *qubit*, is the quantum analogue of a classical bit. We write the two "classical" states as $|0\rangle$ and $|1\rangle$, which are names for the standard basis vectors of $\mathbb{C}^2$:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

The vertical bar and angle bracket together form the *ket*, which indicates the object inside is a vector in a complex Hilbert space. The dual object $\langle\psi|$ is the conjugate transpose, so

$$\langle 0|0\rangle = 1, \qquad \langle 1|1\rangle = 1, \qquad \langle 0|1\rangle = 0,$$

and the two basis states are orthonormal. We must establish the qubit's key distinction from the classical bit: a qubit can exist in a superposition,

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,, \qquad |\alpha|^2 + |\beta|^2 = 1,$$

where $\alpha, \beta \in \mathbb{C}$, and $\alpha$ and $\beta$ themselves are called *amplitudes*. The qubit inhabits both classical states simultaneously, weighted by these amplitudes. The normalization condition

$$|\alpha|^2 + |\beta|^2 = 1$$

ensures measurement yields 0 with probability $|\alpha|^2$ and 1 with probability $|\beta|^2$, which collapses the superposition to the observed outcome. Before measurement, both possibilities coexist. For $n$ qubits, the state space is the tensor product

$$(\mathbb{C}^2)^{\otimes n},$$

of dimension $2^n$. A general $n$-qubit state is a superposition over a

$$|\psi\rangle = \sum_{x \in \{0,1\}^n} \alpha_x |x\rangle, \qquad \sum_x |\alpha_x|^2 = 1.$$

The number of complex amplitudes $\alpha_x$ grows exponentially with $n$. For example, 300 qubits carry a superposition with $2^{300}$ amplitudes, which is more than the number of atoms in the observable universe. This is, in a sense, the source of quantum computing's potential power. Computation on qubits proceeds through quantum gates, which correspond to unitary linear maps

$$U : (\mathbb{C}^2)^{\otimes n} \to (\mathbb{C}^2)^{\otimes n}$$

satisfying

$$UU^\dagger = I.$$

Unitarity is important because it preserves the normalization

$$\sum_x |\alpha_x|^2 = 1$$

throughout the computation and ensures every quantum operation is reversible. Among the fundamental single-qubit gates, the Hadamard gate $H$ maps

$$|0\rangle \mapsto \frac{|0\rangle + |1\rangle}{\sqrt{2}}$$

and

$$|1\rangle \mapsto \frac{|0\rangle - |1\rangle}{\sqrt{2}},$$

creating an equal superposition of both states in a single step. Here, we can establish that a sequence of such gates manipulates all $2^n$ amplitudes simultaneously so amplitudes can interfere constructively, i.e. reinforcing correct answers, and destructively, i.e. canceling wrong ones, before a final measurement extracts a classical result. No classical computer can maintain $2^n$ amplitudes with fewer than $2^n$ bits of memory; a quantum computer does so with only $n$ physical qubits.

## 1.1 Grover's Algorithm

Not every quadratic improvement over classical exhaustive search comes from algebraic structure. In 1996, Lov Grover showed that searching through an unstructured list of $N$ items can be performed in

$$O(\sqrt{N})$$

quantum steps rather than

$$O(N)$$

classical ones [Gro96]. The algorithm works by repeatedly applying a "Grover diffusion" operator that amplifies the amplitude of the correct item relative to all others; after $O(\sqrt{N})$ iterations, the correct item dominates the superposition and measurement finds it with high probability. For a secret vector

$$s \in \mathbb{F}_q^n,$$

Grover's algorithm can search exhaustively for $s$ in time

$$O\big(\sqrt{q^n}\big) = O\Big(q^{n/2}\Big).$$

This is faster than brute-force classical search, but it is still exponential in $n$. It does not break the hardness assumption; it only halves the effective bit-security, which is why lattice-based cryptographic standards such as ML-KEM (Kyber) use dimensions $n \geq 256$ to absorb Grover's attack with room to spare [Nat24]. The truly catastrophic, exponential, "quantum speedup" which breaks RSA entirely comes from a different mechanism altogether: Shor's algorithm.

## 1.2 Shor's Algorithm: "Exponential Speedup" Through Periodicity

Peter Shor's 1994 algorithm is the result that upended public-key cryptography [Sho97]. It runs in time polynomial in $\log N$, which is exponentially faster than every known classical factoring algorithm, and its mechanism is worth understanding in some detail because the same mechanism will appear, inverted, at the heart of LWE's security proof. Here, the central insight is a reduction: integer factorization can be reduced to the problem of finding the multiplicative order of a random element. Let

$$N = pq$$

for unknown primes $p$ and $q$, and choose

$$a \in \mathbb{Z}_N^\times$$

uniformly at random with

$$\gcd(a, N) = 1$$

Define

$$f(x) = a^x \bmod N.$$

Because $\mathbb{Z}_N^\times$ is a finite group of order

$$\varphi(N) = (p-1)(q-1),$$

the function $f$ is periodic; there exists the smallest positive integer $r$, called the multiplicative order of $a$ modulo $N$, satisfying

$$a^r \equiv 1 \pmod{N},$$

and therefore

$$f(x + r) = f(x)$$

for all $x$. If we can find $r$, and if $r$ is even while

$$a^{r/2} \not\equiv -1 \pmod{N},$$

then

$$a^r - 1 = \big(a^{r-2} - 1\big)\big(a^{r-2} + 1\big) \equiv 0 \pmod{N},$$

and at least one of

$$\gcd(a^{r/2} \pm 1, N)$$

yields a nontriwithl factor of $N$. The bottleneck is entirely in finding $r$. Classically, this appears to require exponential time. The Quantum Fourier Transform (QFT) changes this. On a quantum computer, we prepare a uniform superposition all over values of $x$ and compute $f(x)$ simultaneously. The resulting state encodes the periodicity of $f$ in its amplitudes. The QFT,

$$|x\rangle \longmapsto \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i k x / N} |k\rangle,$$

where k is an index that runs over every basis state in the Fourier basis:

$$k \in \{0, 1 \ldots, N-1\},$$

maps the computational basis into a frequency basis. Applied to a periodic state, it concentrates probability on multiples of $N/r$. Classical continued-fraction algorithms then recover $r$ in polynomial time. The complete algorithm runs in

$$O((\log N)^3)$$

quantum gate operations. Through this result, we can establish, soberingly, that classical hardness of an algebraic problem is not preserved under passage to a quantum model whenever that hardness rests on abelian group structure amenable to Fourier analysis. Factoring and discrete logarithms are both reducible to hidden subgroup problems over $\mathbb{Z}$, and the hidden subgroup problem over any finite abelian group is solvable in quantum polynomial time. Less formally, any problem whose difficulty boils down to "find the hidden period of some group operation" fails to the same QFT-based attack that breaks factoring, regardless of the particular group involved. A cryptographic assumption intended to resist quantum adversaries must therefore derive its hardness from a source structurally distinct from periodicity. The Learning with Errors (LWE) problem, introduced by Oded Regev (2005), is built from linear equations over a finite field $\mathbb{F}_q$-superficially the least promising candidate, since its underlying algebraic object, an exact linear system, is solved by Gaussian elimination in cubic time [Reg05]. The central claim to be established over the remainder of this paper is the following contrast: the noiseless system

$$As = b$$

admits an efficient solution algorithm with no known quantum or classical obstruction, while the perturbed system

$$As + e = b,$$

for $e$ a suitably distributed error term, does not. The paper is organized into five parts. *Finite Fields* develops finite fields from the quotient construction $\mathbb{Z}/q\mathbb{Z}$ up through the structure of $\mathbb{F}_q^\times$. *Vector Spaces over Finite Fields* develops vector spaces over a finite field, establishing dimension as a well-defined invariant. *Linear Systems* solves the noiseless linear system $As = b$ completely, in both its algorithmic and "information theoretic??" aspects, and isolates exactly the point at which a single additive perturbation invalidates every argument given. *Lattices and the Hardness of LWE* develops the geometric "apparatus": lattices, discrete Gaussian distributions, bounded distance decoding, and Regev's worst-case-to-average-case reduction, required to convert that single perturbation into a cryptographic hardness assumption, and closes by explaining why the resulting assumption resists the attack of this Introduction.

### 1.3 A Remark on What "Hardness" Will Mean

It is worth fixing the terminology before proceeding, since the word "hard" will be used in two related but distinct senses below. A problem is *worst-case hard* is every algorithm fails on some instance, chosen adversarially; this is the sense in which SVP on a general lattice is conjectured hard in *Lattices and the Hardness of LWE*. A problem is *average-case hard* relative to a distribution $\mathcal{D}$ if every efficient algorithm fails with high probability on an instant drawn from $\mathcal{D}$; this is the sense relevant to a cryptographic assumption, since an attacker does not get to choose the public key. Factoring and discrete logarithm are only known to be average-case hard, with no proof connecting this to any worst-case statement. The principal structural advantage of LWE is a reduction in the opposite direction: average-case hardness of LWE is derived from worst-case hardness of lattice problems, so that breaking a random LWE instance would solve the lattice problem on every instance, not merely a typical one.

## 2 Finite Fields

Cryptographic algorithms require an algebraic structure that is finite, closed under arithmetic, and supports division. This section constructs that structure from first principles, beginning with the integers and ending with the field $\mathbb{F}_q$ on which every later definition depends.

## 2.1 Modular Arithmetic as a Quotient Ring

### 2.1.1 Congruence and the Quotient Set

**Definition 2.1.** *For $q \geq 2$ and $a, b \in \mathbb{Z}$, write*

$$a \equiv b \pmod{q}$$

*if $q \mid (a - b)$.*

**Proposition 2.2.** *Congruence modulo $q$ is an equivalence relation on $\mathbb{Z}$.*

*Proof.* Reflexivity: $a - a = 0 = 0 \cdot q$. Symmetry: $a - b = kq \implies b - a = (-k)q$. Transitivity: $a - b = kq$ and $b - c = \ell q \implies a - c = (k + \ell)q$. Therefore, congruence modulo $q$ is reflexive, symmetric, and transitive. □

The equivalence classes

$$[a] = \{\, b \in \mathbb{Z} : a \equiv b \pmod{q} \,\}$$

partition $\mathbb{Z}$ into exactly $q$ classes, represented by $\{0, 1, \ldots, q - 1\}$, yielding the quotient set $\mathbb{Z}/q\mathbb{Z}$.

### 2.1.2 Well-Definedness of the Ring Operations

Define addition and multiplication on $\mathbb{Z}/q\mathbb{Z}$ by

$$[a] + [b] := [a + b], qquad[a][b] := [ab]$$

These operations are well-defined independently of the representatives chosen. Indeed, if $a \equiv a'$ and $b \equiv b' \pmod{q}$, then $q \mid (a - a')$ and $q \mid (b - b')$. Consequently,

$$q \mid \big((a + b) - (a' + b')\big)$$

and

$$ab - a'b' = a(b - b') + b'(a - a'),$$

so $q \mid (ab - a'b'$. Thus, the equivalence class obtained after addition or multiplication is independent of the representatives selected. Under these operations, $\mathbb{Z}/q\mathbb{Z}$ is a commutative ring with identity [1], associative and distributive because $\mathbb{Z}$ itself is, and is in fact the quotient of $\mathbb{Z}$ by the ideal

$$q\mathbb{Z} = \{\, kq : k \in \mathbb{Z} \,\}$$

in the usual ring-theoretic sense: $q\mathbb{Z}$ is closed under addition and absorbs multiplication by arbitrary integers, which are exactly the conditions required for the cosets $a + q\mathbb{Z}$ to inherit a well-defined arithmetic.

**Example 2.3.** *To compute $17 \cdot 23 \pmod{5}$, well-definedness permits reducing first: $17 \equiv 2 \pmod{5}$ and $23 \equiv 3 \pmod{5}$, so $17 \cdot 23 \equiv 2 \cdot 3 = 6 \equiv 1 \pmod{5}$. This agrees with the direct computation $17 \cdot 23 = 391 = 78 \cdot 5 + 1$. The proposition assures us that every choice of representatives produces the same result, which licenses working with small remainders throughout this paper rather than with the original integers.*

## 2.2 The Quotient Map and the General Quotient Construction

The map

$$\pi : \mathbb{Z} \longrightarrow \mathbb{Z}/q\mathbb{Z}, \qquad \pi(a) = [a],$$

is a surjective ring homomorphism with kernel $q\mathbb{Z}$. The first isomorphism theorem identifies $\mathbb{Z}/q\mathbb{Z}$ with the image of $\pi$, and the demonstrates quotienting a ring by an ideal produces another ring because the ideal is closed under addition and absorbs multiplication by arbitrary ring elements. Informally, we can determine $\pi$ discards information irrelevant to arithmetic modulo $q$, how many complete multiples of $q$ an integer contains, while preserving everything relevant to addition and multiplication.

**Remark 2.4.** *The same construction reappears in higher rank. If $\Lambda' \subseteq \Lambda \subset \mathbb{R}^m$ are full rank-lattices, then the quotient $\Lambda/\Lambda'$ is again a finite abelian group whose order equals the index $[\Lambda : \Lambda']$, computable as the absolute value of the determinant of the corresponding change-of-basis matrix. The case treated here, $\mathbb{Z}/q\mathbb{Z}$, is the rank-one instance of this construction. Its higher-dimensional analogue, $\mathbb{Z}^{>}/\shortparallel\mathbb{Z}^{>}$, reappears in Lattices and the Hardness of LWE.4ish as the ambient group inside which the LWE lattice $\Lambda(A)$ is defined, and the quotient map $\pi$ there plays exactly the role it does here; it is the operation an attacker confined to exact field arithmetic cannot invert, because it has already discarded the real-valued information the attacker needs.*

## 2.3 Invertibility and the Field $\mathbb{F}_p$

### 2.3.1 Zero Divisors and the Necessity of Primality

The construction $\mathbb{Z}/q\mathbb{Z}$ is a field if and only if $q$ is prime [DF04]. If $q = de$ is a nontriwithl factorization, with $1 < d, e < q$, then $[d][e] = [de] = [q] = [0]$ while $[d] \neq [0]$ and $[e] \neq [0]$; a commutative ring with such zero divisors cannot be a field, since in a field $[d][e] = [0]$ forces one of $[d]$ or $[e]$ to vanish. The converse direction, that primality suffices, is the content of the following proposition.

**Definition 2.5.** *A field is a set $F$ with operations $+$ and $\cdot$ such that $F, +$ is an abelian group with identity $0$, $(F^\times, \cdot)$ is an abelian group wth identity $1$, where $F^\times = F \setminus \{0\}$ and multiplication distributes over addition.*

### 2.3.2 The Multiplicative Group $\mathbb{F}_p^\times$ is Cyclic

**Proposition 2.6.** *The construction $\mathbb{F}_p^\times$ is a cyclic group of order $p - 1$.*

*Proof.* Every finite subgroup of the multiplicative group of a field is cyclic. To see this for $\mathbb{F}_p^\times$, let $N(d)$ denote the number of elements of order exactly $d$, for $d \mid (p-1)$. The polynomial $x^d - 1$ has at most $d$ roots in a field, so the number of elements $x$ satisfying $x^d = 1$ is at most $d$. These elements are the union of the cyclic subgroups created by elements whose orders divide $d$, and, a sort argument with the Mobius function (or directly: if $N(d) > 0$), fixing one element $g$ of order $d$ shows all $d$ roots of $x^d - 1$ are powers of $g$, of which exactly $\varphi(d)$ have order $d$, so $N(d) \in \{0, \varphi(d)\}$) shows $N(d) \in \{0, \varphi(d)\}$ for every divisor $d$ of $p - 1$. Since

$$\sum_d d \mid (p-1)\varphi(d) = p - 1 = \sum_{d|(p-1)} N(d),$$

and each $N(d) \leq \varphi(d)$, equality forces $N(d) = \varphi(d)$ for every divisor $d$ of $p - 1$. In particular, $N(p-1) = \varphi(p-1) \geq 1$, so we conclude that an element of order $p - 1$ exists. □

**Remark 2.7.** *Cyclicity of $\mathbb{F}_p^\times$ is not used directly in the linear-algebraic results of Vector Spaces over Finite Fields and Linear Systems, which depend only on the field axioms, but it is the fundamental to the discrete logarithm problem of the Introduction (the hardness of inverting $x \mapsto g^x$ for a generator $g$ of $\mathbb{F}_p^\times$). It is recorded here so the contrast with LWE's hardness source is available for the discussion in Search and Decision are Equivalent: discrete logarithm's hardness is a periodicity statement about the cyclic group $\mathbb{F}_p^\times$, while LWE's, as will be shown, is not a periodicity statement about any group at all.*

### 2.3.3 Prime Power Fields

More generally, for any prime power $q = p^k$ there exists a field of order $q$, unique up to isomorphism, obtained as the splitting field of $x^q - x$ over $\mathbb{F}_p$. We have no need of this generality in what follows, since the moduli arising in lattice-based cryptography are taken to be prime; because $\mathbb{F}_q$ is a field, every nonzero element is invertible, allowing the algorithm to compute inverses directly rather than through the extension-field arithmetic required for prime-power fields, but we record this notion because the uniqueness statement is what enables speaking of the field $\mathbb{F}_q$ without ambiguity throughout the remainder of the paper. We now have a complete arithmetic: the finite field $\mathbb{F}_q$, supporting addition, multiplication, and, importantly, division, all in polynomial time. The next step is to assemble vectors out of field elements and ask: when does a collection of vectors provide a useful "coordinate system"? That question, the subject of *Vector Spaces over Finite Fields*, will establish background for our secret-hiding construction in *Linear Systems*.

# 3 Vector Spaces over Finite Fields

## 3.1 The Space $\mathbb{F}_q^n$

**Definition 3.1.** *For a field $\mathbb{F}_q$ and integer $n \geq 1$,*

$$\mathbb{F}_q^n = \{(a_1, \ldots, a_n) : a_i \in \mathbb{F}_q\},$$

*with component-wise addition and scalar multiplication;*

$$|\mathbb{F}_q^n| = q^n.$$

The axioms of a vector space over $\mathbb{F}_q$ are identical to those over $\mathbb{R}$: closure, associativity and commutativity of vector addition, existence of a zero vector and additive inverses, and compatibility of scalar multiplication with field multiplication and vector addition. Every algebraic consequence of those axioms, independent of any reference to an ordering or to a norm, transfers verbatim. Linear independence, span, and dimension are defined exactly as in the real case: a set $v_1, \ldots, v_k \subset \mathbb{F}_q^n$ is independent if

$$\sum c_i v_i = 0$$

forces every $c_i = 0$, and spans a subspace $W$ if every element of $W$ is such a sum.

**Remark 3.2.** *This point deserves emphasis, because Lattices and the Hardness of LWE will demonstrate it is precisely where the analogy with $\mathbb{R}^n$ collapses. We observe $\mathbb{F}_q$ had no compatible total order and no notion of magnitude beyond the triwithl discrete metric (every nonzero element is, qua field element, exactly as far from 0 as any other); an error term which is geometrically minuscule when its representative is lifted to $\mathbb{Z}$ becomes, upon reduction mod $q$, an algebraically arbitrary field element. Everything proved in this section is thus, exact and unconditional; it holds for every nonzero vector, large or small, and the entire motivation of Lattices and the Hardness of LWE is to reintroduce, from outside the field, the metric structure which this section deliberately does without.*

## 3.2 Existence of a Basis

**Proposition 3.3.** *Every finite-dimensional vector space $V$ over $\mathbb{F}_q$ has a basis [Axl15].*

*Proof.* Let $S \subseteq V$ be a maximal linearly independent subset, which exists since $V$ is finite and hence admits no infinite independent subset. If $\text{span}(S) \neq V$, choose $v \in V \setminus \text{span}(S)$; then $S \cup \{v\}$ is independent (any dependency

$$\sum c_i s_i + cv = 0$$

with $c \neq 0$ would express $v \in \text{span}(S)$), contradicting maximality of $S$. Hence, $\text{span}(S) = V$ and $S$ is a basis. □

## 3.3 Dimension is Well-Defined

### 3.3.1 The Steinitz Exchange Lemma

**Proposition 3.4.** *If $\{v_1, \ldots, v_m\} \subseteq V$ is linearly independent and $\{w_1, \ldots, w_k\}$ spans $V$, then $m \leq k$.*

The proof trades spanning vectors for independent ones, one at a time: each $v_i$ replaces some $w_j$ in the spanning set without destroying the spanning property, because $v_i$ is not redundant with the $v$'s already exchanged in. If this trading can be carried out for all $m$ vectors $v_i$, the spanning set could not have had fewer than $m$ members to begin with.

*Proof.* Induction on $m$. The case $m = 0$ is triwithl. Suppose, after $m - 1$ exchanges and reindexing,

$$\{v_1, \ldots, v_{m-1}, w_m, \ldots, w_k\}$$

spans $V$. Write

$$v_m = \sum_{i<m} c_i v_i + \sum_{j \geq m} d_j w_j.$$

If $d_j = 0$ for all $j \geq m$, then $v_m \in \operatorname{span}(v_1, \ldots, v_{m-1})$, contradicting independence of $\{v_1, \ldots, v_m\}$. Hence $d_j \neq 0$ for some $j \geq m$; reindex so $j = m$. Since $d_m \in \mathbb{F}_q^\times$, the proposition that $\mathbb{F}_q$ is a field implies that $d_m$ is invertible, allowing us to solve for $w_m$ in terms of $v_1, \ldots, v_m, w_{m+1}, \ldots, w_k$, and substitution shows

$$\{v_1, \ldots, v_m, w_{m+1}, \ldots, w_k\}$$

spans $V$. This requires $m \leq k$. □

**Corollary 3.5.** *Any two bases of $V$ have equal cardinality.*

*Proof.* Apply Proposition 3.4 to a pair of bases, in each direction, regarding one as independent and the other as spanning; the two resulting inequalities force equality. □

In other words, no matter which basis one chooses, it always has the same number of elements. This is what establishes dimension a property of the space $V$ itself, instead of any particular basis used to describe it.

#### 3.3.2 Dimension, Coordinates, and the Identification with $\mathbb{F}_q^n$

This common cardinality, $\dim V$, together with the bijection

$$V \longrightarrow \mathbb{F}_q^{\dim V}$$

induced by a choice of basis (sending a vector to its tuple of coefficients), justifies treating an arbitrary finite-dimensional $\mathbb{F}_q$-vector space as $\mathbb{F}_q^n$ without loss of generality. The bijection respects both vector addition and scalar multiplication (it is a linear isomorphism) so any statement provable for $\mathbb{F}_q^n$ transfers immediately to $V$.

### 3.4 Dimension as a Security Parameter

In the cryptographic application of *Linear Systems* and *From Exact to Approximate: Lattices and the Hardness of LWE*, the secret $s \in \mathbb{F}_q^n$ has $\dim = n$, and $n$ serves as the principal security parameter. Two notions we have established through this section make this precise. Firstly, by Corollary 3.5, $n$ is an invariant of the secret's space rather than of any particular coordinate system an attacker happens to use, so no change of basis can shrink the apparent size of the secret. Secondly, and more importantly, $n$ bounds, through Theorem 4.3 below, the amount of linear information about $s$ that a fixed number of equations can disclose: each additional equation can reduce $\dim \ker(A)$ by at most one, so no fewer than $n$ independent equations can pin $s$ down to a single point. In *Lattices and the Hardness of LWE*, we will demonstrate once noise is introduced this bound on information disclosed per equation no longer converts into a bound on time required to recover $s$; the two, identical for the exact system of *Linear Systems*, become entirely decoupled. We now have all the geometry we need: a finite-dimensional vector space $\mathbb{F}_q^n$, characterized completely by its dimension $n$, with a well-defined notion of basis and coordinate. Through *Linear Systems*, we introduce the central question of the first half of this paper: what happens when we try to hide a secret vector $s \in \mathbb{F}_q^n$ by presenting it as the solution to a linear system? The answer, that we cannot, that the exact system leaks everything, will motivate the one small change which makes all the difference.

## 4 Linear Systems

Let $A \in \mathbb{F}_q^{m \times n}$, $x \in \mathbb{F}_q^n$, and $b \in \mathbb{F}_q^m$, and consider the linear system

$$Ax = b \tag{1}$$

## 4.1 Solving the System: Gaussian Elimination

### 4.1.1 Elementary Row Operations

Three operations on the augmented matrix $[A \,|\, b]$ preserve the solution set of (1). We establish them to be interchanging two rows, scaling a row by an element of $\mathbb{F}_q^\times$, and adding a scalar multiple of one row to another. Each is invertible (the inverse operation is again of the same type), so the solution set of the transformed system equals that of the original.

### 4.1.2 Complexity of Elimination

**Theorem 4.1.** *Equation (4.1) can be solved, or determined inconsistent, using $O(mn \cdot \min(m,n))$ field operations, i.e. $O(mn \cdot \min(m,n) \cdot \log^2 q)$ bit operations.*

*Proof.* Gaussian elimination reduces $[A \mid b]$ to row-echelon form through the operations of *Elementary Row Operations*. Every such operation is well-defined over $\mathbb{F}_q$ because we have guaranteed a multiplicative inverse for any nonzero pivot. Without this assurance, our operation of scaling a row by a pivot's inverse would not be available, and elimination would stall. Eliminating the entries below each of the $\min(m,n)$ pivots costs $O(mn)$ field operations per pivot (update at most $m$ rows, each of length at most $n$), giving $O(mn \cdot \min(m,n))$ field operations in total; back substitution contributes a lower-order term, $O(n^2)$. Each field operation in $\mathbb{F}_q$ costs $O(\log^2 q)$ bit operations through the conditional if $p$ is prime, $\mathbb{Z}/p\mathbb{Z}$ is a field. So, the overall procedure runs in time polynomial in $m$, $n$, and $\log q$. □

**Example 4.2.** *Over $\mathbb{F}_5$, with*

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \qquad b = \begin{pmatrix} 1 \\ 2 \end{pmatrix},$$

*row 2 minus 3·row 1 gives*

$$\begin{bmatrix} 1 & 2 \\ 0 & 4-6 \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 0 & -2 \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 0 & 3 \end{bmatrix} \pmod 5,$$

*with corresponding right-hand side*

$$(1,\, 2-3) = (1,-1) = (1,4).$$

*Scaling row 2 by $3^{-1} = 2 \pmod 5$ gives*

$$[0,1 \mid 8] = [0,1 \mid 3]$$

*Back-substitution:*

$$x_2 = 3, \qquad x_1 = 1 - 2 \cdot 3 = -5 \equiv 0 \pmod 5.$$

*We can check directly, that*

$$Ax = (1 \cdot 0 + 2 \cdot 3,\ 3 \cdot 0 + 4 \cdot 3) = (6, 12) \equiv (1, 2) \pmod 5,$$

*confirming the solution.*

## 4.2 The Structure of the Solution Set

### 4.2.1 Rank-Nullity

**Theorem 4.3.** *Let*

$$r = \operatorname{rank}(A) := \dim(\operatorname{colspace}(A)).$$

*The solution set of (4.1), if nonempty, is a coset*

$$x_0 + \ker(A)$$

*of*

$$\ker(A) = \{x \in \mathbb{F}_q^n : Ax = 0\}$$

*with*

$$\dim \ker(A) = n - r.$$

*Hence (4.1) has either no solutions nor exactly*

$$q^{n-r}$$

*solutions.*

Therefore, we find that once one solution $x_0$ to $Ax = b$ is known, every other solution is obtained by adding to it some vector that $A$ sends to 0, and the dimension count below indicates how many such extra solutions there are, $q^{n-r}$ of them, growing geometrically because the rank $r$ falls short of $n$.

*Proof.* If $Ax_0 = b$, then $Ax = b \iff A(a - x_0) = 0 \iff x \in x_0 + \ker(A)$, establishing the coset structure, and $\ker(A)$ is a subspace by linearity of $x \mapsto Ax$. For the dimension count, extend a basis $\{u_1, \ldots, u_{n-r}\}$ of $\ker(A)$ (Proposition 3.3) to a basis $\{u_1, \ldots, u_{n-r}, v_1, \ldots, v_k\}$ of $\mathbb{F}_q^n$, where $k = n - (n - r)$. The images $\{Av_1, \ldots, Av_k\}$ span colspace$(A)$, since $Au_i = 0$, and they are independent: a relation $\sum c_j Av_j = 0$ gives $\sum c_j v_j \in \ker(A) \cap \text{span}(v_1, \ldots, v_k) = \{0\}$, forcing all $c_j = 0$ by independence of the $v_j$. Hence $k = r$, i.e. $\dim \ker(A) = n - r$, and $|x_0 + \ker(A)| = |\ker(A)| = q^{n-r}$. □

**Corollary 4.4.** *If $r = n$, equation (4.1) has at most one solution; combined with Theorem 4.1, this unique solution, when it exists, is computable in time polynomial in $m$, $n$, and $\log q$.*

When $m = n$, full rank coincides with invertibility of $A$, and the unique solution is $x = A^{-1}b$, with $A^{-1}$ computable by applying Theorem 4.1's elimination procedure to $[A \mid I_n]$ and reading off the right-hand block once the left block has been reduced to $I_n$.

### 4.2.2 Almost Every Square Matrix is Invertible

**Proposition 4.5.** *The number of invertible matrices in $\mathbb{F}_q^{n\times n}$ is $\prod_{i=0}^{n-1}(q^n - q^i)$. Consequently, if $A$ is drawn uniformly at random, then* $\Pr[A \text{ is invertible}] = \prod_{k=1}^{n}(1 - q^{-k}) \geq 1 - \sum_{k=1}^{n}(1 - q^{-k}) \geq 1 - \frac{1}{q-1}$.

Through the following proof, we build an invertible matrix one row at a time and count the legal choices at each step. A row is legal if each newly chosen row must lie outside the span of the previously chosen ones, since any such dependency would make the matrix singular.

*Proof.* Construct $A$ row by row. The first row admits any nonzero vector, $q^n - 1$ choices. Given that rows $1, \ldots, i$ are independent and span a subspace of cardinality $q^i$, the $(i + 1)$-st row must remain outside this subspace to preserve independence, leaving $q^n - q^i$ choices. The product over $i = 0, \ldots, n - 1$ gives us the stated count. Normalizing by $q^{n^2}$ and bounding the resulting product below by $a - \sum_k q^{-k}$, then by the geometric series $\sum_{k=1}^{\infty} q^{-k} = \frac{1}{q-1}$ yields the inequality. □

**Remark 4.6.** *As an illustration, when $q = 3$, the limiting probability that a random square matrix is singular is approximately* 0.44*. More generally, this failure probability decreases geometrically as $q$ increases. Since practical lattice cryptosystems use large moduli $q$, typically on the order of several thousand or larger, a uniformly square matrix is invertible with overwhelming probability.*

## 4.3 The Total Insecurity of the Noiseless System

Proposition 4.5 quantifies the cryptographic vulnerability of (4.1) not through asserting this heuristically, but precisely. For $q \geq 3$, a uniformly random square matrix is invertible with probability at least $1/2$, and this probability tends to be the positive constant $\prod_{k=1}^{\infty}(1 - q^{-k})$ as $n \to \infty$. There is no system, in $n$ or in $q$, in which an adversary's task of acquiring a full-range system becomes harder. On the contrary, $m$ independent equations reduce the candidate set for $s$ from $q^n$ to $q^{n-\text{rank}(A)}$ by Theorem 4.3, deterministically. Once $m \geq n$ equations of full rank have accumulated, the coset $x_0 + \ker(A)$ is a single point, and Theorem 4.1 recovers $s$ exactly in polynomial time. Every additional independent equation reduces the solution set by an exact factor of $q$, with no asymptotic floor above $q^0 = 1$. The system $As = b$ discloses $s$ completely once

$\text{rank}(A) = n$, and it does so using an algorithm, Gaussian elimination, which predates digital computation itself and faces no quantum or classical obstruction whatsoever.

## 4.4 Introducing Noise: Where the Argument Breaks

This is the "failure mode" which LWE is built to avoid. Replacing (4.1) with

$$As + e = b, \qquad e \in \mathbb{F}_q^m, \tag{4.2}$$

for $e$ drawn from a distribution concentrated near 0, alters the problem by a single additive term, yet invalidates every step of the argument above.

### 4.4.1 The Coset Description Fails

The coset description of Theorem 4.3 no longer identifies $s$, since $b - As \neq 0$ in general; $b$ lies in the coset $As + e$, not within $As + \ker(A)$, and these two cosets coincide only when $e \in \ker(A)$, an event with probability $q^{r-m} \leq q^{-(m-n)}$ when $A$ has rank $n$, small once $m$ exceeds $n$ by even a constant amount, and in particular not an event Theorem 4.3 has any way of detecting in advance.

### 4.4.2 Elimination Has No Notion of Approximate Satisfaction

Elimination over $\mathbb{F}_q$ has no means of separating the perturbation $e$ from the signal $As$; viewed as elements of $\mathbb{F}_q$, the two are algebraically indistinguishable to any procedure operating exactly. Exact linear algebra, in fact, has no notion of approximately satisfying an equation; row reduction either finds a solution or certifies that none exist, with nothing in-between, so even a single coordinate of noise is enough to send a real solution out of reach. Theorem 4.1's algorithm, run on (4.2), reports that the system, is inconsistent, which is true, and useless to an attacker who already knows it was generated from some $s$.

### 4.4.3 What Must Be Reintroduced

What replaces row reduction is geometry. The error $e$, although it destroys the algebraic coset structure of $\mathbb{F}_q^n$, is small in a metric sense one lifted to $\mathbb{Z}$, and it is this residual structure absent from $\mathbb{F}_q$ itself, by Remark 3.2, that determined whether $b$ can be decoded back to $As$ at all. Making this precise requires we leave the finite field and enter the lattice $\mathbb{Z}^m$, which is what we do in *Lattices and the Hardness of LWE*. We have now completely solved the exact problem, and we have identified precisely where our argument breaks when noise enters. The coset structure of Theorem 4.3 collapses, and Gaussian elimination has no system to separate signal from error. Through the *Lattices and the Hardness of LWE*, we develop lattices, Gaussian distributions, and the reduction which relates LWE to worst-case lattice problems; we conclude by explaining why the QFT attack of the *Introduction* finds no purchase there.

# 5 Lattices and the Hardness of LWE

We now leave the finite field entirely and enter a different mathematical space, where vectors have lengths, points can be close or far, and the geometric relationship between a target and its nearest lattice point is our focus. It is here where LWE's hardness resides. Through *Linear Systems*, we located where the exact linear system stops working once noise is introduced. In *Linear Systems*, we supply what replaces it: a geometric theory of lattices, the discrete Gaussian distributions that model the LWE error term, the reduction of LWE to a well-studied geometric decoding problem, and subsequently, the theorem which lifts an average-case statement about that decoding problem to a worst-case one. We close by returning to the QFT of the *Introduction* and explaining why it does not break the resulting assumption.

## 5.1 Lattices

### 5.1.1 Definition and Basic Properties

**Definition 5.1.** *A lattice $\Lambda \subset \mathbb{R}^n$ of rank $n$ is the set*

$$\Lambda = \left\{ \sum_{i=1}^{n} z_i b_i : z_i \in \mathbb{Z} \right\}$$

*generated by a basis $B = (b_1, \ldots, b_n)$ of linearly independent vectors in $\mathbb{R}^n$.*

A lattice is a discrete additive subgroup of $\mathbb{R}^n$ [MG02]. It is closed under addition and negation (immediate from the definition, since $\mathbb{Z}$ is closed under these), and every bounded region of $\mathbb{R}^n$ contains only finitely many points of $\Lambda$. The latter property is what separates a lattice from an arbitrary $\mathbb{Z}$-submodule of $\mathbb{R}^n$ and is what makes the successive minima of *Successive Minima and the Shortest Vector Problem* well-defined finite qualities instead of infima which are never attained. Two bases $B, B'$ generate the same lattice if and only if

$$B' = BU$$

for some $U \in GL_n(\mathbb{Z})$, a matrix with integer entries and determinant $\pm 1$. Consequently, $|\det B|$ is an invariant of $\Lambda$ itself, written $\det(\Lambda)$. Geometrically, $\det(\Lambda)$ is the volume of the fundamental parallelepiped

$$\left\{ \sum t_i b_i : t_i \in [0, 1) \right\},$$

a single representative of each coset of $\Lambda$ in $\mathbb{R}^n$ under translation.

### 5.1.2 Successive Minima and the Shortest Vector Problem

**Definition 5.2.** *The $i$-th successive minimum $\lambda_i(\Lambda)$ is the radius of the smallest closed ball centered at the origin containing $i$ linearly independent vectors of $\Lambda$. In particular,*

$$\lambda_1(\Lambda) = \min_{v \in \Lambda \setminus \{0\}} \|v\|$$

*is the length of a shortest nonzero lattice vector.*

With our vocabulary of lattice problems in hand, we are ready to describe the distributions that model LWE's error term. The key object is a discrete analogue of the Gaussian distribution that we develop next in *Discrete Gaussian Distributions*. The Shortest Vector Problem (SVP) asks to compute a vector achieving $\lambda_1(\Lambda)$. The decision variant $\text{GapSVP}_\gamma$ asks, given $\Lambda$ and a target $d$, to decide whether

$$\lambda_1(\Lambda) \leq d$$

or

$$\lambda_1(\Lambda) > \gamma d,$$

promising that one of the two holds and asking which. The Shortest Independent Vectors Problem, $\text{SIVP}_\gamma$, asks to output $m$ linearly independent lattice vectors each of length at most

$$\gamma \lambda_n(\Lambda).$$

Both SVP and SIVP are believed to require time exponential in $n$ for general $\Lambda$, classically and quantumly, once the approximation factor $\gamma$ is taken to be polynomial in $n$. The best known algorithms— lattice basis reduction (Lenstra-Lenstra-Lovász, or LLL, and its generalization Block Korkine-Zolotarev, or BKZ) combined with enumeration or sieving— run in time

$$2^{\Theta(n)}$$

for constant for polynomially bounded $\gamma$, and no subexponential quantum algorithm is known.

### 5.1.3 Minkowski's Theorem

**Theorem 5.3** (Minkowski)**.** *For any lattice $\Lambda \subset \mathbb{R}^n$,*

$$\lambda_1(\Lambda) \leq \sqrt{n}\,\det(\Lambda)^{1/n}.$$

The bound follows from Minkowski's convex body theorem: any convex, symmetric region of volume exceeding $2^n \det(\Lambda)$ contains a nonzero lattice point. Applying this to a ball of radius slightly larger than $\frac{\sqrt{n}\,\det(\Lambda)^{1/n}}{\sqrt{/pi}}$ (whose volume, by the standard formula for the volume of an $n$-ball, exceeds $2^n \det(\Lambda)$ for $n$ large) produces a nonzero lattice vector of the stated length, up to the dimension-dependent constant absorbed into the $\sqrt{n}$ in the statement above. We omit the routine volume computation.

**Remark 5.4.** *Minkowski's theorem guarantees a short vector exists without exhibiting it, and the gap between this existence guarantee and efficient algorithm for actually finding such a vector is the gap that SVP's conjectured hardness occupies. This is comparable to, at the level of lattices, the gap in Linear Systems between the existence of a solution to (4.1) (guaranteed whenever $b \in \text{colspace}(A)$ and its computability (guaranteed unconditionally there by Theorem 4.1); the entire point of Lattices and the Hardness of LWE is that for lattices this second guarantee missing.*

### 5.1.4 The Dual Lattice

**Definition 5.5.** *The dual of a lattice $\Lambda \subset \mathbb{R}^n$ is $\Lambda^* = \{y \in \mathbb{R}^n : \langle x, y\rangle \in \mathbb{Z} \text{ for all } x \in \Lambda\}$.*

Here, $\Lambda^*$ is again a lattice of rank $n$, with $\det(\Lambda^*) = \frac{1}{\det(\Lambda)}$ and $(\Lambda^*)^* = \Lambda$. If $B$ is a basis of $\Lambda$, then $(B^{-1})^T$ is a basis of $\Lambda^*$. The dual lattice is used only in *The Worst-Case-to-Average-Case Reduction* below, where the worst-case-to-average-case reduction is most naturally phrased as a statement about sampling on $\Lambda^*$ and converting the result into information about $\Lambda$.

## 5.2 Discrete Gaussian Distributions

### 5.2.1 Definition

For $s > 0$ define the Gaussian weight function $\rho_s(x) = \exp\!\left(-\frac{\pi\|x\|^2}{s^2}\right)$ on $\mathbb{R}^n$. For a lattice $\Lambda$, the discrete Gaussian distribution $D_{\Lambda,s}$ assigns to each $x \in \Lambda$ the probability $D_{\Lambda,s}(x) = \frac{\rho_s(x)}{\rho_s(\Lambda)}$, where $\rho_s(\Lambda) = \sum_{v\in\Lambda} \rho_s(v)$. Before we introduce the formal LWE definition, it is worth understanding what these Gaussian distributions look like and why they are the right choice for modeling small errors. This distribution is the discrete analogue of the continuous Gaussian, restricted to lattice points and renormalized. It is the foundational distribution for error term $\chi$ used in Definition 5.7.

### 5.2.2 Concentration

A sample $x \sim D_{\Lambda,s}$ satisfies $\|x\| \leq s\sqrt{n}$ with probability $1 - 2^{-\Omega(n)}$, a discrete analogue of the concentration of measure enjoyed by the continuous Gaussian on $\mathbb{R}^n$. This is the fact that makes $\|e\| = 0(s\sqrt{m})$ a reliable bound instead of an expectation, on the size of an LWE error vector, and is used without further comment in *Bounded Distance Decoding and the Exact Failure of Linear Methods.*

### 5.2.3 The Smoothing Parameter

**Definition 5.6.** *For $\varepsilon > 0$, the smoothing parameter $\eta_\varepsilon(\Lambda)$ is the smallest $s > 0$ such that $\rho_{1/s}(\Lambda^* \setminus \{0\}) \leq \varepsilon$.*

For $s \geq \eta_\varepsilon(\Lambda)$, the distribution of ($x$ mod $\Lambda$) for $x$ sampled from the continuous Gaussian of parameter $s$ is within statistical distance $\varepsilon$ of uniform on $\mathbb{R}^n/\Lambda$. Equivalently, $D_{\Lambda,s}$ reduced modulo a sublattice becomes statistically indistinguishable from uniform once $s$ crosses this threshold. This permits reduction of *The Worst-Case-to-Average-Case Reduction* to create LWE-like samples that are simultaneously close to a genuine lattice point (so that decoding them solves a lattice problem) and statistically indistinguishable from uniform noise (so that an LWE oracle cannot detect and thus reject them as anomalous).

## 5.3 The LWE Distribution: Formal Definitions

### 5.3.1 Search-LWE and Decision-LWE

**Definition 5.7.** *Search-LWE Fix a security parameter $n$, a prime modulus $q = q(n)$, and an error distribution $\chi$ on $\mathbb{Z}_q$, typically $\chi = D_{\mathbb{Z},\alpha q}$ reduced modulo $q$, for a noise rate $\alpha = \alpha(n) \in (0,1)$. For $a \in \mathbb{F}_q^n$, let $A_{s,\chi}$ be the distribution on $\mathbb{F}_q^n \times \mathbb{F}_q$ we can obtain by drawing $a \in \mathbb{F}_q^n$ uniformly, $e \leftarrow \chi$, and outputting $(a, \langle a, s\rangle + e$.*

$$\text{Search-LWE}_{n,q,\chi}$$

*asks: given $m = \text{poly}(n)$ independent samples $(a_i, b_i) \sim A_{s,\chi}$, recover $s$.*

**Definition 5.8.** *Decision-LWE* $\text{Decision-LWE}_{n,q,\chi}$ *asks to distinguish, with non-negligible advantage, between $m$ samples drawn from*

$$A_{s,\chi}$$

*for a fixed uniform $s$, and $m$ samples drawn uniformly from $\mathbb{F}_q^n \times \mathbb{F}_q$.*

Stacking the $m$ samples as rows reproduces the system of *Linear Systems*. Writing $A \in \mathbb{F}_q^{m\times n}$ for the matrix of the $a_i$'s and $b = (b_1 \ldots, b_m)$, Search-LWE is the problem of recovering $s$ from $b = As + e$, equation (4.2), and Decision-LWE is the problem of telling $As + e$ apart from a uniformly random vector. Theorem 4.1 solves both instantly when $e = 0$. The entire context of *Bounded Distance Decoding and the Exact Failure of Linear Methods*, *The Worst-Case-to-Average-Case Reduction*, and *Search and Decision are Equivalent* is what happens to that algorithm, and to every other known algorithm, once $e \neq 0$.

### 5.3.2 A Worked Numerical Instance

**Example 5.9.** *Take $n = q, \quad q = 97, \quad s = (3, 61)$. Drawing $a_1 = (12, 40)$, the noiseless linear product is $\langle a_1, s\rangle = 12 \cdot 3 + 40 \cdot 61 = 36 + 2449 = 2476 \equiv 2476 - 25 \cdot 97 = 2476 - 2425 = 51 \pmod{97}$. A noise term $e_1 = 2$, drawn from a distribution concentrated near 0, perturbs this to $b_1 = 53$. Given $(a_1, b_1) = ((12, 40), 53)$ alone, an attacker sees one linear equation in two unknowns modulo 97 with an unknown additive shift of unknown but small size. Even granted many such pairs, Bounded Distance Decoding and the Exact Failure of Linear Methods will show that recovering $s = (3, 61)$ from these is an instance of bounded-distance decoding on a $1 + m$-dimensional lattice (rather than a linear-algebra problem), and that this is what makes the problem hard despite each individual equation, appearing, superficially, as an equation from Linear Systems.*

## 5.4 Bounded Distance Decoding and the Exact Failure of Linear Methods

### 5.4.1 The LWE Lattice

Define the $m$-dimensional integer lattice associated to $A$, $\Lambda(A) = \{x \in \mathbb{Z}^m : x \equiv A^T y \pmod q \text{ for some } y \in \mathbb{Z}^n\} \supseteq q\mathbb{Z}^m$, a sublattice of $\mathbb{Z}^m$ of rank $m$ containing $q\mathbb{Z}^m$, so that $|\mathbb{Z}^m/\Lambda(A)| = q^n$ with overwhelming probability over the choice of $A$. This is a consequence of Proposition 4.5, since $A^T$ had rank $n$ with overwhelming probability, and rank $n$ implies its $q^n$ possible values $a^T y \pmod q$, as $y$ ranges over $\mathbb{F}_q^n$, are generally distinct. The vector $As$, lifted to integer representatives in $[0, q)^m$, is by construction a point of $\Lambda(A)$. The LWE instance $b = AS + e \pmod q$ is therefore a lattice point displayed by the error vector $e$.

### 5.4.2 LWE as Bounded Distance Decoding

**Definition 5.10.** *BDD The Bounded Distance Decoding problem* $\text{BDD}_{\Lambda,d}$ *asks: given a lattice $\Lambda$ and a target $t = v + e$ for some $v \in \Lambda$ and $\|e\| \leq d < \frac{\lambda_1(\Lambda)}{2}$, recover $v$ (equivalently, recover $e$).*

We now have everything we need to state the central geometric reinterpretation of LWE:

**Proposition 5.11.** $\text{Search-LWE}_{n,q,\chi}$ *is an instance of* $\text{BDD}_{\Lambda(A,d)}$ *with target $b$ and $d = 0(\alpha q\sqrt{m})$.*

*Proof.* By *Concentration*, $\|e\| \leq \alpha q\sqrt{m}$ with overwhelming probability over $\chi$. The target $b = As + e$, lifted to integer representatives, is a point of $\mathbb{Z}^m$ displaced from the lattice point $As \in \Lambda(A)$ by $e$; recovering $As$ from $b$, and hance, because $A$ has rank $n$ with overwhelming probabiluty, recovering $s$ uniquely through Corollary

4.4 applied to $As =$ (recovered point) is the decoding problem of Definition 5.10, provided $\alpha q\sqrt{m} < \frac{\lambda_1(\Lambda(A))}{2}$, which the parameter choices of *The Worst-Case-to-Average-Case Reduction* are established to guarantee. □

## 5.5 Why Row Reduction Cannot be Salvaged

The reason Theorem 4.1 cannot be repaired is now exact (rather than heurestic). Row reduction operates entirely inside $\mathbb{F}_q^n$. It sees $b$ only as an element of the quotient $\mathbb{Z}^m/q\mathbb{Z}^m$ and has no system to test whether a given coset representative is close to a lattice point of $\Lambda(A)$, because closeness is a statement about the embedding of $\Lambda(A)$ in $\mathbb{R}^m$ that the arithmetic of $\mathbb{F}_q$ does not encode. By Remark 3.2, every nonzero element of $\mathbb{F}_q$ is, as a field element, equally far from every other. Decoding requires reintroducing the real embedding $\mathbb{R}^m \supset \mathbb{Z}^m \supset \Lambda(A)$ that the quotient map $\pi$ of *The Quotient Map and the General Quotient Construction* was specifically built to discard. No sequence of the three elementary row operations of *Elementary Row Operations*, each of which is defined purely in terms of $\mathbb{F}_q$-arithmetic, can recover information that $\pi$ has already thrown away.

## 5.6 The State of the Art in Lattice Algorithms

This does not mean BDD is unconditionally hard [Pei16]. It is solvable in time $2^{O(n)}$ by basis reduction (LLL, producing a basis with vectors no more than exponentially longer than $\lambda_1$ in the worst case, in time polynomial in the bit-length if the input, or, BKZ, trading more time for shorter output vectors) followed by nearest-plane or enumeration-based decoding, and substantial cryptanalytic effort (the BKZ family of algorithms, sieving algorithms, and their quantum-accelerated variants) is devoted to pushing the practical limits of this exponential-time approach downward in the constant in the exponent. The choice of parameters $n$, $q$, and $\alpha$ is what places the resulting BDD instance in the system where every such algorithm, lattice reduction included, requires time $2^{\Omega(n)}$ for the values of $n$ used in practice, while keeping $\|e\|$ small enough that a legitimate party who additionally knows a short basis for $\Lambda(A)$ can decode efficiently. Hardness, in other words, is not asserted of the noisy linear systems in isolation, but is inherited from the conjectured hardness of BDD/SVP on $\Lambda(A)$, *The Worst-Case-to-Average-Case Reduction* makes this inheritance rigorous in the worst case.

## 5.7 The Worst-Case-to-Average-Case Reduction

### 5.7.1 Why Average-Case Hardness is Not Enough

*Bounded Distance Decoding and the Exact Failure of Linear Methods* leaves a gap: hardness of BDD on the specific, randomly generated lattice $\Lambda(A)$ is an average-case statement, and everage-case hardness has historically proven fragile. Factoring and discrete logarithm (Remark 2.9) are themselves average-case-hard relative to a fixed instance distribution, with no worst-case aspect, which is part of why structural attacks (the number field sieve; Shor's algorithm itself) have repeatedly succeeded against them by exploiting structure present in every instance of the distribution actually used. LWE's hardness claim is stronger.

### 5.7.2 The Problems GapSVP and SIVP

Before stating the reduction we must introduce the two lattice problems it reduces from. Both GapSVP and SIVP are problems concerning the geometry of an arbitrary lattice $\Lambda \subset \mathbb{R}^n$ given as a worst-case input, meaning an adversary has chosen the hardest lattice they can find, and both are parameterized by an approximation factor $\gamma = \gamma(n) \geq 1$, which controls how accurate our answer must be. The force of Regev's theorem rests on the claim that these problems are difficult.

**The Gap Shortest Vector Problem (GapSVP)** Recall from *Successive Minima and the Shortest Vector Problem* that $\Lambda_1(\Lambda)$ denotes the length of the shortest nonzero vector in $\Lambda$. The SVP asked us, given a basis for $\Lambda$, compute $\lambda_1(\Lambda)$. This problem is NP-hard under randomized reductions and is believed to remain hard to approximate for polynomial approximation factors $\gamma$. However, it is a search problem, requiring the algorithm to produce an explicit shortest vector, which is inconvenient when creating reductions. The decision

version, GapSVP$_\gamma$, avoids this difficulty. We are given a basis $B$ for $\Lambda$ and a threshold $d > 0$, together with the guarantee that exactly one of the following holds:

$$\textbf{YES instance:} \qquad \lambda_1(\Lambda) \le d,$$

or

$$\textbf{NO instance:} \qquad \lambda_1(\Lambda) > \gamma d.$$

The task is simply to determine which case holds. The guarantee excludes all lattices satisfying $d < \lambda_1(\Lambda) \le \gamma d$, so the algorithm never needs to distinguish borderline cases. When $\gamma = 1$, the problem is equivalent to the exact SVP. As $\gamma$ increases, the two cases become more widely separated (and thus, easier to distinguish). Indeed, for $\gamma = \sqrt{n}$, the problem lies in $\mathbf{NP} \cap \mathbf{coNP}$ and is widely believed not to be NP-hard for polynomial $\gamma$, although no polynomial-time algorithm is known. Regev's reduction operates here, where $\gamma = \widetilde{O}\big(\frac{n}{\alpha}\big)$.

**Example 5.12.** *Consider the lattice formed by*

$$b_1 = (1, 0), \qquad b_2 = (0.5, 10).$$

*The shortest nonzero lattice vector is $b_1$, so $\lambda_1(\Lambda) = 1$. A* GapSVP$_3$ *instance with threshold $d = 1$ asks whether $\lambda_1(\Lambda) \le 1$ or $\lambda_1(\Lambda) > 3$, The correct answer is YES, as we observe from vector $b_1$. In high dimensions, however, where a basis may consist of hundreds of highly skewed vectors spanning $\mathbb{R}^{500}$, even determining whether such a short vector exists becomes extremely difficult.*

**The Shortest Independent Vectors Problem (SIVP)** Where GapSVP concerns a single shortest vector, the SIVP asks for an entire collection of short, linearly independent vectors. Recall that $\lambda_n(\Lambda)$ denotes the $n$th successive minimum of $\Lambda$, the smallest radius containing $n$ linearly independent lattice vectors. Formally,

$$\lambda_n(\Lambda) = \min\left\{r > 0 : \exists\, v_1, \ldots, v_n \in \Lambda \text{ linearly independent with } \|v_i\| \le r \text{ for all } i\right\}.$$

Given a basis for $\Lambda$, the problem SIVP$_\gamma$ asks us to output linearly independent lattice vectors $v_1, \ldots, v_n \in \Lambda$ satisfying $\max_{1 \le i \le n} \|v_i\| \le \gamma(n)\lambda_n(\Lambda)$. Thus, our objective is to create an approximately shortest basis for the lattice. SIVP is stronger than GapSVP in the following sense. A solution to SIVP$_\gamma$ immediately yields a solution to GapSVP$_{\gamma\sqrt{n}}$, since the shortest vector among $v_1, \ldots, v_n$ has length at most $\gamma\lambda_n(\Lambda) \le \gamma\sqrt{n}\,\lambda_1(\Lambda)$ by Minkowski's theorem. The converse implication is not known, making SIVP the more difficult problem. This asymmetry explains why both problems appear in Theorem 5.13. Regev's reduction first solves SIVP$_\gamma$ on the worst-caes lattice and then derives a solution to GapSVP$_{\gamma'}$ for a slightly larger approximation factor $\gamma'$.

## 5.8 Statement of the Reduction

**Theorem 5.13.** *Regev, 2005, informal statement Let $n$ be a security parameter, $q = q(n)$ a prime, and $\alpha = \alpha(n) \in (0, 1)$ a noise rate satisfying $\alpha q \ge 2\sqrt{n}$. There exists a probabilistic polynomial-time reduction, making use of a quantum subroutine, from* GapSVP$_\gamma$ *and* SIVP$_\gamma$ *on arbitrary $n$-dimensional lattices, for $\gamma - \widetilde{0}(n/\alpha)$, to* Decision-LWE$_{n,q,\chi}$ *for $\chi$ the discrete Gaussian error distribution of rate $\alpha$ [Reg09].*

The reduction does not run on $\Lambda(A)$ At all. Instead, it runs on an arbitrary, worst-case lattice $L$ of rank $n$, supplied adversarially, and uses an assumed Decision-LWE oracle to solve SIVP on $L$. The reduction is iterative, decreasing a width parameter $s$ from an initially large value down to $0(\sqrt{n}) \cdot \lambda_1(L)$ over polynomially many rounds. Three facts make each round possible.

## 5.9 The Three Components of the Reduction

(i) Discrete Gaussian sampling reduces to BDD. Given a sufficiently short basis of $L$, we can sample from $D_{L,s}$ for $s$ above the smoothing parameter $\eta_\varepsilon(L)$ of Definition 5.6; conversely, given many independent samples from $D_{L,s}$, one can find vectors progressively closer to $\lambda_1(L)$, since a discrete Gaussian at a fine enough width concentrates near the shortest vectors by *Concentration*.

(ii) A single round takes a current discrete Gaussian sample of $L$ at width $s$ and uses the Decision-LWE oracle, applied to a derived distribution on the dual lattice $L^*$ of The Dual Lattice, to test statistical hypotheses allowing the reduction to produce a sample of $L$ at a strictly smaller width $s' < s$. For this step, we require a quantum subroutine, preparing a quantum superposition over a coset of $L^*$, weighted by $\rho_s$, and applying a QFT over the continuous torus $\mathbb{R}^n/L^*$ to convert the width-reduction step into a phase-estimation-type measurement. Structurally, this is the same primitive as the QFT of the *Introduction*, but applied here as the system of a security proof instead as an attack.

(iii) Iterating the width reduction polynomially many times drives $s$ down to $0(\sqrt{n}) \cdot \lambda_1(L)$, at which point the resulting discrete Gaussian samples demonstrate a shortest vector of $L$ directly, solving SIVP, or (with a small additional argument) decide $\text{GapSVP}_\gamma$.

## 5.10 The QFT as Proof Tool, Not as Attack

The role of the QFT here inverts the moral of the *Introduction*. There, the QFT was the attacker's tool, exploiting an abelian periodic structure that factoring and discrete-log assumptions could not avoid having. Here, the QFT is the reduction's tool used not to extract a hidden period from the LWE instance itself (LWE has none, by construction, once $e \neq 0$ destroys the coset structure of Theorem 4.3), and instead to convert worst-case geometric information about an unrelated lattice $L$ into a hardness guarantee for the average-case algebraic problem of *The LWE Distribution: Formal Definitions*. No step of the reduction extracts a period from $A_{s,\chi}$. The quantum subroutine acts entirely on the lattice $L^*$, which carries no information about any particular LWE secret. This is why the hidden-subgroup attack of the *Introduction* has no purchase on LWE. The only periodicity available to a QFT-based attacker would be that of the function $a \mapsto \langle a, s\rangle$, and the additive error $e$ establishes this function exactly constant on no coset of any nontriwithl subgroup of $\mathbb{F}_q^n$. Here, $e$ is, by Definition 5.7, supported on all of $\mathbb{F}_q$ with the discrete Gaussian's tail, so $\langle a, s\rangle + e$ agrees with no group homomorphism on any subgroup of positive index. The system of this hidden subgroup, which requires the target function to be exactly constant on cosets, simply does not apply.

**Remark 5.14.** *A classical variant of Theorem 5.13 was later given by Peikert (2009) for moduli $q$ of a special form ($q$ close to a power of 2), at the cost of a somewhat worse approximation factor $\gamma$; the quantum step described above remains necessary for the reduction in its original generality, and it is this asymmetry (a classical attack is not known to exist, while the strongest hardness guarantee currently requires a quantum proof technique) that makes LWE a genuinely post-quantum, and not merely classically unbroken, assumption.*

## 5.11 Search and Decision are Equivalent

**Theorem 5.15.** *For $q$ prime and polynomial in $n$,* $\text{Search-LWE}_{n,q,\chi}$ *and* $\text{Decision-LWE}_{n,q,\chi}$ *are equivalent up to a* $\text{poly}(n, q)$ *overhead.*

We can thus establish that a search oracle solves decision is immediate: recover $s$ and check whether $\langle a_i, s\rangle - b_i$ is distributed as $\chi$-noise or its uniform. The converse, substantive direction, proceeds by a hybrid argument recovering $s$ one coordinate at a time.

## 5.12 Recovering One Coordinate

Fix a Decision-LWE distinguisher $D$ with advantage $\varepsilon$. To recover $s_1$, the first coordinate of $s$, an algorithm guesses a candidate $g \in \mathbb{F}_q$ and, for each of the $q$ possible shifts $k \in \mathbb{F}_q$, transforms each sample $(a, b) = (a_1, \ldots, a_n, b)$ into $(a_1, \ldots, a_n,\ b - ka_!)$, feeding the shifted samples to $D$. If $g = s_1$, then the shift by $k = 0$ leaves the distribution exactly $\mathcal{A}_{s,\chi}$, while a shift by any $k \neq 0$ introduces a perturbation that is uniform and independent of the original noise, statistically close to the uniform distribution $D$ is meant to detect against. If $g \neq s_1$, the situation reverses; $D$'s response as a function of $k$ therefore distinguishes a correct guess $g = s_1$ from an incorrect one with advantage at least $\varepsilon$, and trying all $q$ candidates and retaining the one whose response pattern matches the LWE case identifies $s_1$ using $O(q)$ calls to $D$.

## 5.13 Iterating Over All Coordinates

Repeating the coordinate-recovery procedure of *Recovering One Coordinate* for each of the $n$ coordinates, "masking out" previously recovered coordinates' contributions before testing the next, costs $O(nq)$ calls to $D$ in total, and a standard amplification argument (repeating the whole procedure $O(\mathrm{poly}(n)/\varepsilon^2)$ times and taking a majority vote, justified by a Chernoff bound on the number of correct individual trials) leads the failure probability to be negligible. The full procedure, detailed in Regev's original reduction, is unconditional and entirely classical. In contrast to Theorem 5.13, it requires no quantum subroutine. It is, in fact, the only step in the entire chain of reductions presented here that converts one aspect of LWE hardness into another without any appeal to any unproven geometric assumption beyond LWE's own decision hardness. Notably, factoring and discrete logarithm have no comparable unconditional search-to-decision equivalence, since recovering a discrete logarithm and distinguishing a Diffie-Hellman tuple from random are not known to be polynomial-time equivalent in general.

## 5.14 From Hardness to a Cryptosystem

The contrast we established in *Linear Systems* (easy exactly, hard approximately), is not merely a complexity-theoretic curiosity once it is shown, as in *From Hardness to a Cryptosystem*, to support an encryption scheme whose correctness depends on the very smallness of $e$ that makes decryption-without-the-key infeasible.

### 5.14.1 Key Generation

Sample $A \in \mathbb{F}_q^{m\times n}$ uniformly and $s \in \mathbb{F}_q^n$ uniformly (the secret key); sample $e \in \mathbb{F}_q^m$ from $\chi^m$ and set $b = As + e \in \mathbb{F}_q^m$ (the public key, together with $A$).

### 5.14.2 Encryption

To encrypt a bit $\mu \in \{0,1\}$, sample a uniformly random subset $T \subseteq \{1,\dots,m\}$ (equivalently, a random vector $r \in \{0,1\}^m$), and output the pair $(u,c) = \left(\sum_{i\in T} a_i^T,\ \sum_{i\in T} b_i + \mu\left\lfloor \frac{q}{2}\right\rfloor\right) \in \mathbb{F}_q^n \times \mathbb{F}_q$, where we denote $a_i$ as the $i$-th row of $A$. This is a randomized re-encoding of ($As+e$ restricted to $T$) plus, in the case $\mu = 1$, an offset of $q/2$ designed to be recoverable after decryption but not to be confused with the accumulated noise.

### 5.14.3 Decryption and Correctness

Given $(u,c)$ and the secret key $s$, compute

$$c - \langle u, s\rangle = \sum_{i\in T}(b_i - \langle a_i, s\rangle) + \mu\left\lfloor \frac{q}{2}\right\rfloor.$$

If the accumulated noise $\sum_{i\in T} e_i$—a sum of $|T| \le m$ independent draws from $\chi$, hence itself concentrated by *Concentration*, around 0 with standard dewithtion $O(\alpha q\sqrt{m})$— has magnitude less than $q/4$, then $c - \langle u, s\rangle$ lies within $q/4$ of 0 if $\mu = 0$ and within $q/4$ of $\lfloor q/2\rfloor$ is $\mu = 1$, and these two regions are disjoint. Rounding to the nearer of $\{0, \lfloor q/2\rfloor\}$ recovers $\mu$ correctly except with negligible probability. This is why the noise rate $\alpha$ cannot be taken arbitrarily large; correctness requires $\alpha q\sqrt{m} = o(q)$, i.e. $\alpha = o\left(\frac{1}{\sqrt{m}}\right)$, while *The State of the Art in Lattice Algorithms* requires $\alpha q$ large enough (concretely $\alpha q \ge 2\sqrt{n}$, as in Theorem 5.13) that BDD on $\Lambda(A)$ remains exponentially hard. The entire parameter selection problem for LWE-based cryptography is the problem of satisfying both inequalities simultaneously, and that a wide range of $(n, q, \alpha)$ does so is what Theorem 5.13 and *The State of the Art in Lattice Algorithms* jointly certify for us.

### 5.14.4 Security

An eavesdropper sees $(A, b)$ and $(u, c)$. Semantic security, that $c$ reveals no more about $\mu$ than its length does, follows directly from Decision-LWE, since $(A, b)$ is, by Definition 5.8, indistinguishable from a uniformly random pair, and a uniformly random $b$ makes $c = \sum_{i\in T} b_i + \mu\left\lfloor \frac{q}{2}\right\rfloor$ uniform on $\mathbb{F}_q$ regardless of $\mu$, by a standard argument that a uniform element plus any fixed shift is again uniform. Reduction from breaking semantic security to solving Decision-LWE is "black-box" and loses only a polynomial factor, so Theorem 5.13 transfers its worst-case lattice hardness guarantee to the cryptosystem essentially without loss.

### 5.15 A Remark on Efficiency: Ring-LWE

The scheme of *From Hardness to a Cryptosystem* requires a public key of size $O(mn \log q)$ field elements, quadratic in the security parameter if $m = \Theta(n)$; this is large compared to, say, an RSA public key. Replacing $\mathbb{F}_q^n$ with the quotient ring $\mathbb{F}_q[x]/(x^n + 1)$ and the matrix $A$ with multiplication by a single random ring element $a$, in the Ring-LWE variant introduced by Lyubashevsky, Peikert, and Regev (2010), reduces key size to $O(n \log q)$ while a reduction analogous to Theorem 5.13 connects its hardness to worst-case problems on a restricted class of lattices with the extra algebraic composition of ideals in the ring. We do not develop Ring-LWE further here, beyond noting its aforementioned extra algebraic composition is the kind of regularity a hidden-subgroup-style quantum attack would, in principle, have the best chance of exploiting were one to exist, and that no such attack is currently known is itself informative evidence, though not proof, for the conjecture closing this paper below.

## 6 What Survives Quantum Computation, and Why

Through the *Introduction*, we isolated the mechanism used by every known exponential quantum speedup over classical cryptanalysis: a hidden, periodic structure on an abelian group, exposed by interference under the Quantum Fourier Transform. Through *Finite Fields*, *Vector Spaces over Finite Fields*, and *Linear Systems*, we then showed that an exact linear system over a finite field, in isolation offers no resistance to attack of any kind, as Gaussian elimination already solves it in classical polynomial time. Rank–Nullity (Theorem 4.3) guarantees that $n$ independent equations determine an $n$-dimensional secret completely, and the density result of Proposition 4.5 guarantees that an attacker accumulates such equations with overwhelming probability essentially for free. In other words, linear algebra was never a candidate for a post-quantum hardness assumption, and by *The Total Insecurity of the Noiseless System*, it was never even a candidate for a classical one.

The single additive perturbation introduced in equation (4.2) changes this completely. We established through *The Total Insecurity of the Noiseless System* that the perturbed system no longer names a coset of a subspace of $\mathbb{F}_q^n$, by *The Coset Description Fails*, it identifies a lattice point shifted by the error vector in the real embedding $\mathbb{Z}^m \supset \Lambda(A)$, from a lattice by a short but nonzero vector, and recovering the secret becomes an instance of Bounded Distance Decoding as in *Bounded Distance Decoding and the Exact Failure of Linear Methods* on a high-dimensional lattice, a geometric with no known subexponential algorithm, classical or quantum, once the dimension and approximation factor are chosen as in *Decryption and Correctness*. Theorem 5.13 then upgrades this average-case geometric hardness into a worst-case one; breaking LWE for almost any choice of $A$ would yield an algorithm solving the Shortest Vector Problem on every $n$-dimensional lattice, including those specifically constructed to be hard, with a reduction that, notably, uses the QFT itself, but as a proof tool as we observe in *Statement of the Reduction* and *The Three Components of the Reduction*. It is the precise sense in which LWE's hardness is structurally distinct from that of factoring and discrete logarithms, as promised at the close of the *The QFT as Proof Tool, Not as Attack*.

The function $\langle a, s \rangle + e$ that an LWE attacker must invert is, once $e$ is included, constant on no nontriwithl coset of any subgroup of $\mathbb{F}_q^n$ (textitIntroduction), and is therefore not an instance of any Hidden Subgroup Problem to which Shor's algorithm, or any of its generalizations, applies. It is not a stronger numerical coincidence, nor a larger search space, that protects LWE; instead, it is the substitution of a periodicity-based hardness source, which quantum computation is generically equipped to defeat, for a geometric one, for which no analogous quantum primitive is known to exist. The search-to-decision equivalence of *Search and Decision are Equivalent* and the cryptosystem of *From Hardness to a Cryptosystem* confirm that, this geometric hardness source, once established, is as useful for building cryptography as the periodicity-based sources of the *Introduction* were. It supports an encryption scheme whose correctness and security trade off against each other through the single noise-rate parameter $\alpha$, akin to the discrete logarithm and factoring assumptions supported Diffie–Hellman and RSA.

Whether this protection is permanent is not a question linear algebra, exact or approximate, can answer. GapSVP and SIVP are conjectured, not proven, to lie outside **BQP**. Theorem 5.13 shows that any future quantum algorithm breaking LWE would also solve worse-case lattice problems, which raises the cost of an

attack but does not rule one out. The contrast (easy exactly, hard approximately) we established through this paper should thus accordingly be read as an account of where that conjecture currently stands, and of which steps in the cryptographic argument are unconditional (in *Finite Fields*, *Vector Spaces over Finite Fields*, and *Linear Systems*) and the search-to-decision equivalence of *Search and Decision are Equivalent*) versus conjectural (the assumed hardness of GapSVP/SIVP through Theorem 5.13, and by extension every cryptographic use of LWE), not as a proof the conjecture must continue to hold.

## Acknowledgments

The author thanks Ethan Zhou for helpful discussions, and Simon Rubinstein-Salzedo for providing the environment and opportunity for this work.

## References

[Axl15] Sheldon Axler. *Linear Algebra Done Right*. Springer, 3rd edition, 2015.

[DF04] David S. Dummit and Richard M. Foote. *Abstract Algebra*. John Wiley & Sons, 3rd edition, 2004.

[Gro96] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC)*, pages 212–219, 1996.

[MG02] Daniele Micciancio and Shafi Goldwasser. *Complexity of Lattice Problems: A Cryptographic Perspective*. Kluwer Academic Publishers, 2002.

[Nat24] National Institute of Standards and Technology. Fips 203: Module-lattice-based key-encapsulation mechanism standard. Technical report, National Institute of Standards and Technology, U.S. Department of Commerce, 2024.

[NC00] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.

[Pei16] Chris Peikert. A decade of lattice cryptography. *Foundations and Trends in Theoretical Computer Science*, 10(4):283–424, 2016.

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC)*, pages 84–93, 2005.

[Reg09] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM*, 56(6):Article 34, 2009.

[Sho97] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, 1997.