Efficient Collision-Resistant Hashing from Worst-Case Assumptions on Cyclic Lattices*

Chris Peikert† Alon Rosen‡

August 11, 2006

Abstract

The generalized knapsack function is defined as $f_{\mathbf{a}}(\mathbf{x}) = \sum_i a_i \cdot x_i$, where $\mathbf{a} = (a_1, \ldots, a_m)$ consists of m elements from some ring R, and $\mathbf{x} = (x_1, \ldots, x_m)$ consists of m coefficients from a specified subset $S \subseteq R$. Micciancio (FOCS 2002) proposed a specific choice of the ring R and subset S for which inverting this function (for random $\mathbf{a}, \mathbf{x}$) is at least as hard as solving certain worst-case problems on cyclic lattices.

We show that for a different choice of $S \subset R$, the generalized knapsack function is in fact *collision-resistant*, assuming it is infeasible to approximate the shortest vector in n-dimensional cyclic lattices up to factors $\tilde{O}(n)$. For slightly larger factors, we even get collision-resistance for *any* $m \geq 2$. This yields *very* efficient collision-resistant hash functions having key size and time complexity almost linear in the security parameter n. We also show that altering S is necessary, in the sense that Micciancio's original function is *not* collision-resistant (nor even universal one-way).

Our results exploit an intimate connection between the linear algebra of n-dimensional cyclic lattices and the ring $\mathbb{Z}[\alpha]/(\alpha^n - 1)$, and crucially depend on the factorization of $\alpha^n - 1$ into irreducible cyclotomic polynomials. We also establish a new bound on the discrete Gaussian distribution over general lattices, employing techniques introduced by Micciancio and Regev (FOCS 2004) and also used by Micciancio in his study of compact knapsacks.

1 Introduction

A function family $\{f_a\}_{a \in A}$ is said to be *collision-resistant* if given a uniformly chosen $a \in A$, it is infeasible to find elements $x_1 \neq x_2$ so that $f_a(x_1) = f_a(x_2)$. Collision-resistant hash functions are one of the most widely-employed cryptographic primitives. Their applications include integrity checking, user and message authentication, commitment protocols, and more.

Many of the applications of collision-resistant hashing tend to invoke the hash function only a small number of times. Thus, the efficiency of the function has a direct effect on the efficiency of the application that uses it. This is in contrast to primitives such as one-way functions, which typically must be invoked many times in their applications (at least when used in a black-box way) [9].

Collision-resistance can be obtained from many well-studied complexity assumptions, but the resulting hash functions are not efficient enough for practical use. Instead, faster *heuristic* constructions such as MD5 and SHA-1 are often employed. Unfortunately, recent cryptanalytic analysis

*A preliminary version of this paper appeared in the Third Theory of Cryptography Conference (TCC 2006).

†MIT Computer Science and AI Laboratory (CSAIL), Cambridge, MA

‡DEAS, Harvard, Cambridge, MA. Part of this work done while at MIT CSAIL.

of many popular hash functions casts doubt on the heuristic approach [26, 25]. This presents the theoretical community with a great opportunity and challenge: propose a *practical* hash function with *rigorous* security guarantees.

In this paper we present an *efficient* collision-resistant hash function whose security is based on a well-defined and plausible complexity assumption.

1.1 Generalized Knapsacks

Our constructions are based on a generalization of the well-known *knapsack* function. For a ring R, key $\mathbf{a} = (a_1, \ldots, a_m) \in R^m$, and input $\mathbf{x} = (x_1, \ldots, x_m)$, the generalized knapsack function is defined as

$$f_{\mathbf{a}}(\mathbf{x}) = \sum_{i=1}^{m} a_i \cdot x_i,$$

where each x_i is restricted to some large subset $S \subseteq R$. This generalization was proposed by Micciancio, who suggested a specific choice of the ring R and subset S for which inverting the function (for random $\mathbf{a}, \mathbf{x}$) is at least as hard as solving certain worst-case problems on *cyclic* lattices [15].

Knapsacks have a long and infamous history in cryptography; we refer the interested reader to Micciancio's account of various knapsack proposals and their cryptanalysis [15]. The bottom line is that even though many knapsack systems have been broken heuristically, there is still no *asymptotically-efficient* attack on the general function.

Micciancio's result might be viewed as an indication that knapsack functions (or at least, some version of them) are secure after all. In this paper, we continue Micciancio's line of study, and show that, for a different choice of $S \subset R$, the generalized knapsack function can enjoy even stronger cryptographic properties.

1.2 Lattices, Hardness, and Cryptography

Lattices are a great source of cryptographic hardness. First of all, lattices have been subject to hundreds of years of mathematical scrutiny, which lends support to conjectures on the computational hardness of problems related to lattices. Indeed, many lattice problems are NP-hard to approximate for small factors, e.g. the closest vector [24, 4, 7] and shortest vector problems [2, 5, 16, 13].

Secondly, lattices admit *worst-case* to *average-case* reductions. In his groundbreaking result, Ajtai first constructed a one-way function [1], which was later observed to also be collision-resistant [11]. Public-key cryptosystems [12, 3, 20, 21] soon followed, based on presumably stronger worst-case assumptions. As a bonus, these constructions tended to be asymptotically more efficient than those based on, e.g., modular exponentiation.

An interesting special case is presented by *cyclic* lattices. A lattice Λ is said to be cyclic if for any vector $\mathbf{x} \in \Lambda$, its *cyclic rotation* also belongs to Λ. The cyclic rotation of $\mathbf{x} = (x_0, \ldots, x_{n-1})^T \in \mathbb{R}^n$ is defined as $(x_{n-1}, x_0, \ldots, x_{n-2})^T$.

Micciancio's work [15] opened the door to the use of cyclic lattices as a new source of hardness assumptions, and motivates their study from a computational perspective. Currently no hardness results are known for problems on cyclic lattices (even in their exact versions), and the additional structure may indeed reduce the underlying hardness. However, state-of-the-art lattice algorithms appear not to benefit from cyclicity, and it seems reasonable to conjecture that standard problems on cyclic lattices are intractable, at least for small approximation factors.

1.3 Our Results

Our main result is that certain instantiations of the generalized knapsack function are collision-resistant, assuming it is infeasible to approximate the shortest vector in cyclic lattices up to factors $\tilde{O}(n)$ almost linear in the dimension n.

Assuming hardness for slightly larger approximation factors $n^{1+\epsilon}$, our functions remain secure even when m is taken to be a *constant*. The functions have key size almost linear in the security parameter n and can be evaluated with m Fast Fourier Transform operations, making them potentially practical. To motivate our choice of knapsack function, we also show that Micciancio's original one-way function is not collision-resistant, nor even universal one-way.

In the course of proving our main results, we formulate special worst-case problems on cyclic lattices, and relate them to the more standard lattice problems. Most interestingly, we demonstrate that for cyclic lattices of *prime* dimension n, the short *independent* vectors problem SIVP reduces to (a slight variant of) the shortest vector problem SVP with only a factor of 2 loss in approximation factor. For general lattices, the best known reduction loses a $\sqrt{n}$ factor [17]; furthermore, that reduction performs manipulations on its input lattice that can destroy the cyclicity property. Hence our reduction can be seen as the first connection between SIVP and SVP on cyclic lattices.

Finally, in using the Gaussian techniques of [18], we also establish a new bound on the discrete Gaussian distribution over general lattices, which may be of independent interest.

1.4 Techniques and Ideas

The overarching theme of our paper is the tight relationship shared by cyclic lattices, the algebra of polynomials modulo $(\alpha^n - 1)$, and linear algebra in $\mathbb{R}^n$.

Cyclic lattices are closed under *cyclic convolution* with integer vectors. Furthermore, the lattice points naturally correspond to polynomials in $\mathbb{Z}[\alpha]/(\alpha^n - 1)$. Because convolution is equivalent to polynomial multiplication in $\mathbb{Z}[\alpha]/(\alpha^n - 1)$, this implies that integer cyclic lattices are isomorphic to *ideals* in $\mathbb{Z}[\alpha]/(\alpha^n - 1)$.

The divisors of $(\alpha^n - 1)$ in $\mathbb{Z}[\alpha]$ correspond to special *cyclotomic* linear subspaces of $\mathbb{R}^n$. These subspaces admit a natural partitioning into complementary pairs of orthogonal subspaces. Even more importantly, the subspaces are closed under *cyclic rotation* of vector coordinates, and under certain other conditions, these rotations are *linearly independent*. These facts imply a new connection between the SIVP and SVP problems in cyclic lattices.

The security of our knapsack function comes from using all this structure to impose an algebraic restriction on the function domain. Looking ahead to the security reduction, this restriction ensures that collisions in the function are very likely to yield "useful" and short lattice points in a desired subspace.

1.5 Comparison with Related Work

This work takes its inspiration from, and is most similar to, Micciancio's work on cyclic lattices [15]. However, while our knapsack function is very similar to Micciancio's, the reduction used to establish collision-resistance differs in many significant ways. First of all, Micciancio's function is proven to be one-way, while ours is collision-resistant. On the other hand, Micciancio relies on a presumably weaker worst-case assumption than we do. Our stronger assumption, combined with our algebraic view of cyclic lattices, makes our security reduction tighter and conceptually simpler.

	Security	Efficiency	Lattice Class	Assumption	Approx. Factor
Ajtai [1]	CRHF	$O(n^2)$	General	SVP etc.	$\text{poly}(n)$
Cai, Nerurkar [6]	CRHF	$O(n^2)$	General	SVP etc.	$n^{4+\epsilon}$
Micciancio [15]	OWF	$\tilde{O}(n)$	Cyclic	GDD	$n^{1+\epsilon}$
Micciancio, Regev [18]	CRHF	$O(n^2)$	General	SVP etc.	$\tilde{O}(n)$
This work	CRHF	$\tilde{O}(n)$	Cyclic	SVP etc.	$\tilde{O}(n)$

Figure 1: Comparison of results in lattice-based cryptographic functions with worst-case to average-case security reductions, to date. "Efficiency" means the key size and computation time, as a function of the lattice dimension n. "Security" denotes the function's main cryptographic property.

Figure 1 gives a comparison of our work with other major results in worst-case to average-case reductions, in chronological order. Important considerations in these works include: provable security properties of the cryptographic function, efficiency of that function, class of lattice on which the function is based, type of worst-case problem that is assumed to be hard for that class of lattice, and its hardness of approximation factor. Our work compares very favorably in many of these considerations, at the cost of a qualitatively stronger assumption.

The actual worst-case assumption underlying our hash function is that SVP is hard on cyclic lattices for all sufficiently large *prime* dimensions n. Therefore, the discovery of an efficient algorithm for SVP on, say, all *even* dimensions would have no immediate effect on the security of our hash function. Conveniently, the *concrete* hardness of the cyclic lattice problems we study appears to be *greatest* when the dimension is prime! More specifically: problems in composite dimensions n seem to reduce to problems in the smaller prime (or prime-power) dimensions dividing n.

In an independent and concurrent work, Lyubashevsky and Micciancio [14] have obtained exceedingly similar results, but expressed in different mathematical language. In particular, by making many of the same algebraic insights, they construct collision-resistant hash functions with nearly identical parameters, based on a worst-case hardness assumption that can be shown to be equivalent to ours. They also present a more general algebraic framework for constructing hash functions, which can be related to problems in algebraic number theory. Due to its generality, their framework may have the potential to admit better constructions, though its current best application essentially matches ours.

2 Preliminaries

In this section we present basic definitions and results about statistical distance, hash functions, cyclic lattices, cyclotomic polynomials and Gaussian probability distributions. In many places we follow [18] almost verbatim.

For any real $a \geq 0$, $\lfloor a \rfloor$ denotes the largest integer not greater than a and $\lfloor a \rceil$ denotes the closest integer to a (i.e., $\lfloor a \rceil = \lfloor a + 1/2 \rfloor$). For any reals $a, b \geq 0$, $[a, b)$ denotes the set of all reals $a \leq r < b$. The uniform probability distribution over a set S is denoted $U(S)$. We let I denote $U([0,1))$. A function $f(n)$ is said to be negligible (denoted $f(n) = n^{-\omega(1)}$) if for every $c > 0$ there exists an n_0 such that $|f(n)| < 1/n^c$ for all $n > n_0$.

The set of real numbers is denoted by $\mathbb{R}$, and the quotient ring of integers modulo a positive integer p is denoted by $\mathbb{Z}_p$. For a value $v \in \mathbb{Z}_p$, $|v|$ denotes the absolute value of the unique integer $r \in (-p/2, p/2]$ representing v's residue class. We use bold lower case letters (e.g., $\mathbf{x}$) to denote vectors and bold upper case letters (e.g., $\mathbf{A}$) to denote matrices. Vectors are represented as columns and we use $(\cdot)^T$ to denote matrix transposition. We adopt the convention that vector indices are *zero-based*, i.e. for $\mathbf{x} \in \mathbb{R}^n$ we write $\mathbf{x} = (x_0, \ldots, x_{n-1})^T$. The ith coordinate of $\mathbf{x}$ is denoted x_i or $(\mathbf{x})_i$, depending on context. The *Euclidean norm* of a vector $\mathbf{x}$ (in either $\mathbb{R}^n$ or $\mathbb{Z}_p^n$) is the quantity $\|\mathbf{x}\| = (\sum_i |x_i|^2)^{1/2}$. The Euclidean norm of a matrix $\mathbf{S} = (\mathbf{s}_1, \ldots, \mathbf{s}_t)$ is $\|\mathbf{S}\| = \max_i \|\mathbf{s}_i\|$. Other norms used in this paper (for vectors in either $\mathbb{R}^n$ or $\mathbb{Z}_p^n$) are the ℓ_1 norm $\|\mathbf{x}\|_1 = \sum_i |x_i|$ and the ℓ_∞ norm $\|\mathbf{x}\|_\infty = \max_i |x_i|$, which are similarly extended to matrices. These norms are related through the following inequalities, valid for any n-dimensional vector $\mathbf{x} \in \mathbb{R}^n$:

$$\begin{array}{ccccc} \|\mathbf{x}\| & \leq & \|\mathbf{x}\|_1 & \leq & \sqrt{n}\|\mathbf{x}\| \\ \|\mathbf{x}\|_\infty & \leq & \|\mathbf{x}\| & \leq & \sqrt{n}\|\mathbf{x}\|_\infty \end{array}$$

2.1 Statistical distance

We use the standard definitions of statistical distance and state some simple facts that will be used in the analysis of our reductions.

Definition 2.1 (Statistical distance). Let X and Y be two discrete random variables over a (countable) set A. The statistical distance between X and Y is the quantity

$$\Delta(X, Y) = \frac{1}{2} \sum_{a \in A} |\Pr[X = a] - \Pr[Y = a]| \,.$$

In the case of continuous random variables, the statistical distance between X and Y is

$$\Delta(X, Y) = \frac{1}{2} \int_A |\delta_X(a) - \delta_Y(a)| \, da,$$

where δ_X and δ_Y are the probability density functions of X and Y respectively.

Statistical distance satisfies the usual properties of distance functions, i.e., $\Delta(X, Y) \geq 0$, $\Delta(X, Y) = \Delta(Y, X)$, and $\Delta(X, Z) \leq \Delta(X, Y) + \Delta(Y, Z)$. Two random variables X, Y are said to be identically distributed (written $X \equiv Y$) if and only if $\Delta(X, Y) = 0$.

The following proposition shows that applying a (possibly randomized) function to two distributions does not increase the statistical distance.

Proposition 2.2. *Let X, Y be two random variables taking values over a common set A. For any (possibly randomized) function f with domain A, the statistical distance between $f(X)$ and $f(Y)$ is at most*

$$\Delta(f(X), f(Y)) \leq \Delta(X, Y)$$

We shall also use the following property of statistical distance.

Proposition 2.3. *Let $X_1, \ldots, X_k$ and $Y_1, \ldots, Y_k$ be two lists of totally independent random variables. Then,*

$$\Delta((X_1, \ldots, X_k), (Y_1, \ldots, Y_k)) \leq \sum_{i=1}^{k} \Delta(X_i, Y_i).$$

2.2 One-way and hash function families

A function family $\{f_a : X \to R\}_{a \in A}$ is a collection of functions (indexed by a set of keys A) with a common domain X and range R. A function ensemble is a sequence $\{f_a : X_n \to R_n\}_{a \in A_n}$ of function families indexed by a security parameter $n \in \mathbb{N}$. We assume that $\log |A_n|, \log |X_n|$ and $\log |R_n|$ are all polynomial in n, that elements in the sets A_n, X_n and R_n can be efficiently represented and (almost) uniformly sampled in polynomial time, and that membership in the sets can be decided in polynomial time. We also require that f_a can be efficiently computable, namely there exists a polynomial time algorithm that on input $n, a \in A_n$, and $x \in X_n$, outputs $f_a(x)$.

Definition 2.4 (One-way function). A function ensemble $\{f_a : X_n \to R_n\}_{a \in A_n}$ is *one-way* if, for any probabilistic polynomial-time algorithm $\mathcal{A}$, the probability that $f_a(\mathcal{A}(1^n, a, f_a(x))) = f_a(x)$ (when $a \in A_n$ and $x \in X_n$ are selected at random) is negligible in n.

The function families $\mathcal{H} = \{f_a : X \to R\}_{a \in A}$ considered in this paper have the property that the domain size $|X|$ is strictly bigger than the range size $|R|$. Such families are generically called *hash* function families. For many cryptographic applications it is useful to require that, given a uniformly chosen key $a \in A$ to f_a, it is infeasible to find elements $x_1 \neq x_2 \in X$ such that $f_a(x_1) = f_a(x_2)$ (a.k.a. *collisions* of f_a).

Definition 2.5 (Collision-resistant hash function). A hash function ensemble $\{f_a : X_n \to R_n\}_{a \in A_n}$ is *collision-resistant* if, for any probabilistic-polynomial time algorithm $\mathcal{A}$, the probability (over a uniformly chosen $a \in A_n$) that $\mathcal{A}(1^n, a)$ outputs $x_1, x_2 \in X_n$ such that $x_1 \neq x_2$ and $f_a(x_1) = f_a(x_2)$ is negligible in n.

A somewhat weaker requirement from a hash function ensemble is that for *any* $x_1 \in X_n$ it is infeasible, given a uniformly chosen key $a \in A_n$, to find an $x_2 \in X_n$ such that $x_2 \neq x_1$ and $f_a(x_1) = f_a(x_2)$.[1] Hash function families that satisfy this requirement are called *universal one-way.*

Universal one-way hash functions were suggested in [19] as a tool for constructing digital signatures. They can be constructed (in a black-box way) from any one-way function [19, 22], albeit black box constructions of this kind are inherently inefficient [10]. There exists no black-box construction of collision resistant hash functions from one-way functions (or even permutations) [23].

2.3 Lattices

A *lattice* in $\mathbb{R}^n$ is the set of all integer combinations

$$\Lambda = \left\{ \sum_{i=1}^{d} c_i \mathbf{b}_i \mid c_i \in \mathbb{Z} \text{ for } 1 \leq i \leq d \right\}$$

of d linearly independent vectors $\mathbf{b}_1, \ldots, \mathbf{b}_d \in \mathbb{R}^n$. We say that the lattice *spans* the d-dimensional subspace of $\mathbb{R}^n$ generated by $\mathbf{b}_1, \ldots, \mathbf{b}_d$. The set of vectors $\mathbf{b}_1, \ldots, \mathbf{b}_d$ is called a *basis* for the lattice, which can be written in matrix form as $\mathbf{B} = [\mathbf{b}_1 | \cdots | \mathbf{b}_d]$ with the basis vectors as columns. The lattice generated by $\mathbf{B}$ is denoted $\mathcal{L}(\mathbf{B})$. For any basis $\mathbf{B}$, we define the *fundamental parallelepiped* $\mathcal{P}(\mathbf{B}) = \{\mathbf{B} \cdot \mathbf{x} : \forall\, i, 0 \leq x_i < 1\}$.

[1]One could also consider a more “uniform” version in which x_1 is first chosen by the adversary. For simplicity of exposition we choose to adopt the non-uniform version.

The *minimum distance* $\lambda_1(\Lambda)$ of a lattice Λ is the length of the shortest nonzero lattice vector: $\lambda_1(\Lambda) = \min_{0 \neq \mathbf{x} \in \Lambda} \|\mathbf{x}\|$. More generally, the ith successive minimum $\lambda_i(\Lambda)$ is the smallest radius r such that the closed ball $\overline{\mathcal{B}}(r) = \{\mathbf{x} : \|\mathbf{x}\| \leq r\}$ contains i linearly independent lattice vectors.

Let H be a subspace of $\mathbb{R}^n$ and let Λ be a lattice that spans H. Then we define the *dual lattice* $\Lambda^* = \{\mathbf{x} \in H \mid \forall\, \mathbf{v} \in \Lambda, \langle \mathbf{x}, \mathbf{v} \rangle \in \mathbb{Z}\}$.

Cyclic lattices and convolution. For any $\mathbf{x} = (x_0, \ldots, x_{n-1})^T \in \mathbb{R}^n$, define the *rotation of* $\mathbf{x}$, denoted as $\mathrm{rot}(\mathbf{x})$, to be the vector $(x_{n-1}, x_0, \ldots, x_{n-2})^T$; similarly $\mathrm{rot}^i(\mathbf{x}) = \mathrm{rot}(\cdots \mathrm{rot}(\mathbf{x}) \cdots)$ is defined to be the rotation of $\mathbf{x}$, taken i times. A lattice Λ is *cyclic* if for all $\mathbf{x} \in \Lambda$, $\mathrm{rot}(\mathbf{x}) \in \Lambda$. For any integer $d \geq 1$, define the *rotation matrix* $\mathrm{Rot}^d(\mathbf{x})$ to be the matrix $[\mathbf{x}|\mathrm{rot}(\mathbf{x})|\cdots|\mathrm{rot}^{d-1}(\mathbf{x})]$. ($\mathrm{Rot}^n(\mathbf{x})$ is known as the *circulant matrix* of $\mathbf{x}$.)

For any ring R, the (cyclic) convolution product of $\mathbf{x}, \mathbf{y} \in R^n$ is the vector $\mathbf{x} \otimes \mathbf{y} = \mathrm{Rot}^n(\mathbf{x}) \cdot \mathbf{y}$, with entries

$$(\mathbf{x} \otimes \mathbf{y})_k = \sum_{i+j=k \bmod n} x_i \cdot y_j.$$

Observe that in a cyclic lattice Λ, the convolution of any $\mathbf{x} \in \Lambda$ with any integer vector $\mathbf{y} \in \mathbb{Z}^n$ is also in the lattice: $\mathbf{x} \otimes \mathbf{y} \in \Lambda$. This is because all the columns of $\mathrm{Rot}^n(\mathbf{x})$ are in Λ, and any integer combination of points in Λ is also in Λ.

The convolution product is commutative, associative, and distributive over vector addition; also, it satisfies the following inequalities, valid for any n-dimensional vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$:

$$\begin{aligned} \|\mathbf{x} \otimes \mathbf{y}\|_\infty &\leq \|\mathbf{x}\| \cdot \|\mathbf{y}\| \\ \|\mathbf{x} \otimes \mathbf{y}\|_\infty &\leq \|\mathbf{x}\|_1 \cdot \|\mathbf{y}\|_\infty \end{aligned}$$

2.4 Polynomial Rings and Linear Algebra

Convolution and polynomial multiplication are intimately related. Specifically, for any ring R, we identify an element $(x_0, \ldots, x_{n-1}) = \mathbf{x} \in R^n$ with the polynomial $\mathbf{x}(\alpha) \in R[\alpha]/(\alpha^n - 1)$ defined as $\mathbf{x}(\alpha) = x_0 + x_1\alpha + \ldots + x_{n-1}\alpha^{n-1}$. Then it is easy to show that for any $\mathbf{x}, \mathbf{y} \in R^n$, $\mathbf{x} \otimes \mathbf{y}$ is identified with $\mathbf{x}(\alpha) \cdot \mathbf{y}(\alpha) \in R[\alpha]/(\alpha^n - 1)$. In words, convolution of two vectors is equivalent to taking the product of their polynomials modulo $\alpha^n - 1$. Throughout the paper, we will switch between vector and polynomial notation as is convenient.

In the following lemma, we relate the algebra of $\mathbb{R}[\alpha]/(\alpha^n - 1)$ to the linear algebra of $\mathbb{R}^n$.

Lemma 2.6. *Let* $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$ *with* $\mathbf{a}(\alpha) \cdot \mathbf{b}(\alpha) = 0 \bmod (\alpha^n - 1)$. *Then* $\langle \mathbf{a}, \mathbf{b} \rangle = 0$.

Proof. Let $\mathbf{F}$ be the $n \times n$ matrix with (zero-indexed) entries given by

$$(\mathbf{F})_{j,k} = \frac{e^{2\pi i jk/n}}{\sqrt{n}} = \frac{\omega^{jk}}{\sqrt{n}},$$

where ω is the principal nth root of unity ($\mathbf{F}$ is known as a *Fourier matrix*). It is well-known that $\mathbf{F}$ is a unitary matrix, so $\langle \mathbf{a}, \mathbf{b} \rangle = \langle \mathbf{Fa}, \mathbf{Fb} \rangle$. By definition, $(\mathbf{Fa})_i = \mathbf{a}(\omega^i)/\sqrt{n}$ and $(\mathbf{Fb})_i = \mathbf{b}(\omega^i)/\sqrt{n}$. Now because $\mathbf{a}(\alpha)\mathbf{b}(\alpha)$ is divisible by $\alpha^n - 1$, then $\mathbf{a}(\omega^i) \cdot \mathbf{b}(\omega^i) = 0$ (in $\mathbb{C}$) for every i. Therefore

$$\langle \mathbf{a}, \mathbf{b} \rangle = \langle \mathbf{Fa}, \mathbf{Fb} \rangle = \frac{1}{n} \sum_{i=1}^{n} \mathbf{a}(\omega^i)\mathbf{b}(\omega^i) = 0. \qquad \square$$

In the polynomial ring $\mathbb{Z}[\alpha]$, $(\alpha^n - 1)$ has a special structure: it uniquely factors into the product of *cyclotomic polynomials* (see e.g. [8] for a detailed treatment). For integer $k \geq 1$, the kth cyclotomic polynomial $\Phi_k(\alpha)$ is defined:

$$\Phi_k(\alpha) = \prod_{\substack{1 \leq c \leq k \\ (c,k)=1}} (\alpha - e^{2\pi i c/k}),$$

where (c, k) denotes the greatest common divisor of c and k. The cyclotomic polynomial $\Phi_k(\alpha)$ is irreducible in $\mathbb{Z}[\alpha]$, has integer coefficients, and has degree $\phi(k)$ (where ϕ denotes Euler's totient function). The factorization of $\alpha^n - 1$ in $\mathbb{Z}[\alpha]$ is: $\alpha^n - 1 = \prod_{\substack{k \mid n \\ k \geq 1}} \Phi_k(\alpha)$.

In the following lemmas, we establish connections between cyclotomic polynomials and the linear algebra of integer cyclic lattices:

Lemma 2.7. *Let $\mathbf{c} \in \mathbb{Z}^n$, and suppose $\Phi(\alpha) \in \mathbb{Z}[\alpha]$ divides $(\alpha^n - 1)$ and is coprime to $\mathbf{c}(\alpha)$. Then $\mathbf{c}, \mathrm{rot}(\mathbf{c}), \ldots, \mathrm{rot}^{\deg(\Phi)-1}(\mathbf{c})$ are linearly independent.*

Proof. Suppose that there exist $t_0, \ldots, t_{\deg(\Phi)-1} \in \mathbb{R}$ such that $\sum_{i=0}^{\deg(\Phi)-1} t_i \mathrm{rot}^i(\mathbf{c}) = 0$. Define $\mathbf{t} = (t_0, t_1, \cdots, t_{\deg(\Phi)-1}, 0, \cdots, 0)^T$, so $\mathbf{c} \otimes \mathbf{t} = 0$ (where the convolution is performed in $\mathbb{R}^n$). Therefore in $\mathbb{R}[\alpha]$, $(\alpha^n - 1)$ divides $\mathbf{c}(\alpha)\mathbf{t}(\alpha)$.

We recall two basic facts from field theory (see, e.g., [8, Proposition 9, Chapter 13]): first, $\Phi_k(\alpha)$ is the *minimal polynomial*[2] of any primitive kth root of unity, and has exactly the primitive kth roots of unity as its roots. Second, the minimal polynomial of any algebraic number ζ divides any polynomial $p(\alpha) \in \mathbb{Q}[\alpha]$ such that $p(\zeta) = 0$.

Now, because $\Phi(\alpha) \,|\, (\alpha^n - 1)$, $\Phi(\alpha)$ is a product of cyclotomic polynomials. Because $\Phi(\alpha)$ is coprime to $\mathbf{c}(\alpha)$ and $\mathbf{c}(\alpha) \in \mathbb{Z}[\alpha] \subset \mathbb{Q}[\alpha]$, none of the roots of $\Phi(\alpha)$ are roots of $\mathbf{c}(\alpha)$. Therefore all the roots of $\Phi(\alpha)$ must be roots of $\mathbf{t}(\alpha)$. Because $\deg(\mathbf{t}(\alpha)) < \deg(\Phi)$, we must have $\mathbf{t} = 0$. □

Suppose $\Phi(\alpha) \in \mathbb{Z}[\alpha]$ divides $\alpha^n - 1$, i.e. $\Phi(\alpha)$ is a product of cyclotomic polynomials. We define the *cyclotomic subspace*

$$H_\Phi = \{\mathbf{x} \in \mathbb{R}^n : \Phi(\alpha) \text{ divides } \mathbf{x}(\alpha) \text{ in } \mathbb{R}[\alpha]\}.$$

Lemma 2.8. *H_Φ is closed under* rot*: that is, if $\mathbf{c} \in H_\Phi$, then $\mathrm{rot}(\mathbf{c}) \in H_\Phi$.*

Proof. Observe that the vector $\mathrm{rot}(\mathbf{c})$ is identified with the residue $\alpha \cdot \mathbf{c}(\alpha) \bmod (\alpha^n - 1)$. Let $\alpha \cdot \mathbf{c}(\alpha) = Q(\alpha) \cdot (\alpha^n - 1) + R(\alpha)$, for $Q(\alpha), R(\alpha) \in \mathbb{R}[\alpha]$, where $\deg(R(\alpha)) < n$. Then because $\Phi(\alpha) \,|\, \alpha \cdot \mathbf{c}(\alpha)$ and $\Phi(\alpha) \,|\, Q(\alpha) \cdot (\alpha^n - 1)$, it must be that $\Phi(\alpha) \,|\, R(\alpha)$. Therefore $\Phi(\alpha)$ divides $\mathrm{rot}(\mathbf{c})(\alpha)$ in $\mathbb{R}[\alpha]$, as desired. □

Lemma 2.9. *H_Φ is a linear subspace of $\mathbb{R}^n$ of dimension $n - \deg(\Phi)$.*

Proof. It is evident that H_Φ is closed under addition and scalar multiplication, so it is a linear subspace. To establish the dimension, define $\overline{\Phi}(\alpha) = (\alpha^n - 1)/\Phi(\alpha)$. By Lemma 2.6, because $\Phi(\alpha) \cdot \overline{\Phi}(\alpha) = 0 \bmod (\alpha^n - 1)$, H_Φ and $H_{\overline{\Phi}}$ are orthogonal subspaces. Therefore $\dim(H_\Phi) + \dim(H_{\overline{\Phi}}) \leq n$.

By Lemma 2.7, the vectors $\Phi, \mathrm{rot}(\Phi), \ldots, \mathrm{rot}^{\deg(\overline{\Phi})-1}(\Phi)$ are linearly independent. By Lemma 2.8, they all lie in H_Φ. Therefore $\dim(H_\Phi) \geq \deg(\overline{\Phi}) = n - \deg(\Phi)$. Symmetrically, $\dim(H_{\overline{\Phi}}) \geq n - \deg(\overline{\Phi})$. All three inequalities can be satisfied only with equality, hence $\dim(H_\Phi) = n - \deg(\Phi)$. □

[2]The minimal polynomial of an algebraic number ζ is the unique irreducible monic (i.e., with leading coefficient 1) polynomial $p(\alpha) \in \mathbb{Q}[\alpha]$ of minimum degree such that $p(\zeta) = 0$.

2.5 Gaussian Distributions

For any d-dimensional subspace H of $\mathbb{R}^n$, any $\mathbf{c} \in H$ and any $s > 0$, define

$$\rho_{H,s,\mathbf{c}}(\mathbf{x}) = \begin{cases} \exp(-\pi\|(\mathbf{x}-\mathbf{c})/s\|^2) & \text{if} \quad \mathbf{x} \in H \\ 0 & \text{if} \quad \mathbf{x} \notin H \end{cases}$$

to be the Gaussian function (over H) centered at $\mathbf{c}$, with radius s. By normalizing $\rho_{s,\mathbf{c}}$ by its total measure $\int_{\mathbf{x}\in H} \rho_{s,\mathbf{c}}(\mathbf{x})d\mathbf{x} = s^d$, we get a continuous distribution with density function

$$D_{H,s,\mathbf{c}}(\mathbf{x}) = \frac{\rho_{H,s,\mathbf{c}}(\mathbf{x})}{s^d}.$$

The center $\mathbf{c}$ is taken to be zero when not explicitly specified.

Given an orthonormal basis (consisting of d vectors in $\mathbb{R}^n$) for H, $D_{H,s,\mathbf{c}}$ can be written as the sum of d orthogonal 1-dimensional Gaussian distributions, each along one of the basis vectors. Therefore sampling from $D_{H,s,\mathbf{c}}$ can be efficiently approximated. For simplicity we will assume that our algorithms can work with infinite-precision real numbers and sample from Gaussians exactly.

The Fourier transform. For a d-dimensional subspace H of $\mathbb{R}^n$, the Fourier transform (over H) of a function $h : H \to \mathbb{C}$ is a function $\hat{h} : H \to \mathbb{C}$, defined as $\hat{h}(\mathbf{w}) = \int_{\mathbf{x}\in H} h(\mathbf{x})e^{-2\pi i\langle \mathbf{x},\mathbf{w}\rangle}\, d\mathbf{x}$. It follows directly from the definition that if, for all $\mathbf{x} \in H$, h satisfies $h(\mathbf{x}) \equiv g(\mathbf{x}+\mathbf{v})$ for some $\mathbf{v} \in H$ and some function $g : H \to \mathbb{R}$, then $\hat{h}(\mathbf{w}) = e^{2\pi i\langle \mathbf{v},\mathbf{w}\rangle}\hat{g}(\mathbf{w})$. The Fourier transform of a Gaussian function (over H, centered at 0) is another Gaussian (also centered at 0); specifically, $\widehat{\rho_{H,s}} = s^d \cdot \rho_{H,1/s}$.

2.6 Gaussian Measures on Lattices

For any countable set S and any function f, define $f(S) = \sum_{x\in S} f(x)$. For a lattice $\Lambda \subset H$ that spans H and for any $\mathbf{x} \in \Lambda$, define

$$D_{\Lambda,s,\mathbf{c}}(\mathbf{x}) = \frac{D_{H,s,\mathbf{c}}(\mathbf{x})}{D_{H,s,\mathbf{c}}(\Lambda)}$$

to be the conditional probability of $\mathbf{x}$ sampled from $D_{H,s,\mathbf{c}}$, given $\mathbf{x} \in \Lambda$.

One fact connecting lattices and the Fourier transform is the Poisson summation formula:

Lemma 2.10. *Let H be a subspace of $\mathbb{R}^n$. For any lattice $\Lambda \subset H$ that spans H and any "well-behaved"[3] function f, $f(\Lambda) = \det(\Lambda^*)\hat{f}(\Lambda^*)$, where $\hat{f}$ is the Fourier transform (over H) of f.*

The smoothing parameter. Micciancio and Regev [18] defined a new lattice parameter related to Gaussian measures, called the *smoothing parameter*. The following is a generalization of their definition to lattices of possibly less than full rank:

Definition 2.11 (Smoothing parameter). Let H be a subspace of $\mathbb{R}^n$. For a lattice $\Lambda \subset H$ that spans H and positive real $\epsilon > 0$, the smoothing parameter $\eta_\epsilon(\Lambda)$ is defined to be the smallest s such that $\rho_{H,1/s}(\Lambda^*\backslash\{0\}) \le \epsilon$.

[3]The precise condition is technical, but all functions we consider are well-behaved.

The name "smoothing parameter" is justified by the following fact (stated formally in Lemma 2.12): if random noise chosen from a Gaussian distribution of radius $\eta_\epsilon(\Lambda)$ is added to a lattice Λ that spans H, the resulting distribution is almost uniform over H.

Lemma 2.12 ([18], Lemma 4.1, generalized to subspaces). *For any subspace H of $\mathbb{R}^n$, lattice $\mathcal{L}(\mathbf{B})$ that spans H, $\mathbf{c} \in H$, and $s \geq \eta_\epsilon(\mathcal{L}(\mathbf{B}))$, we have*

$$\Delta(D_{H,s,\mathbf{c}} \bmod \mathcal{P}(\mathbf{B}),\ U(\mathcal{P}(\mathbf{B}))) \leq \epsilon/2.$$

Micciancio and Regev also establish relationships between η_ϵ and other standard lattice parameters like λ_n. Here we generalize to lattices of possibly less than full rank:

Lemma 2.13 ([18], Lemma 3.3, generalized to subspaces). *For any super-logarithmic function $f(n) = \omega(\log n)$, there exists a negligible function $\epsilon(n)$ such that: for any d-dimensional subspace H of $\mathbb{R}^n$ and lattice Λ that spans H, $\eta_\epsilon(\Lambda) \leq \sqrt{f(n)} \cdot \lambda_d(\Lambda)$.*

Finally, we will need to bound the norm of the convolution of two vectors, where one of the vectors is chosen from a discrete Gaussian distribution.

Lemma 2.14 ([15], Lemma 3.2, generalized to subspaces). *For any d-dimensional subspace H of $\mathbb{R}^n$, lattice Λ that spans H, positive reals $\epsilon \leq 1/3$, $s \geq 2\eta_\epsilon(\Lambda)$ and vectors $\mathbf{c}, \mathbf{x} \in H$,*

$$E_{\mathbf{v} \sim D_{\Lambda,s,\mathbf{c}}}\left[\|(\mathbf{v} - \mathbf{c}) \otimes \mathbf{x}\|^2\right] \leq s^2 \cdot d \cdot \|\mathbf{x}\|^2.$$

2.7 A New Lemma on Gaussian Distributions Over Lattices

In [18] it is shown that, for a full-rank lattice Λ and large enough s, $D_{\Lambda,s,\mathbf{c}}$ behaves very much like $D_{\mathbb{R}^n,s,\mathbf{c}}$, i.e. their moments are similar. In this work, we will need a different fact about $D_{\Lambda,s,\mathbf{c}}$, specifically, a bound on its maximum value over all points in Λ.

In order to prove such a bound, we need a lemma which is implicit in [18]:

Lemma 2.15 ([18]). *Let H be a d-dimensional subspace of $\mathbb{R}^n$, and Λ be a lattice that spans H. For any $s \geq \eta_\epsilon(\Lambda)$ and any $\mathbf{c} \in H$:*

$$s^d \det(\Lambda^*) \cdot (1 - \epsilon) \quad \leq \quad \rho_{H,s,\mathbf{c}}(\Lambda) \quad \leq \quad s^d \det(\Lambda^*) \cdot (1 + \epsilon).$$

Now we are ready to bound the maximum value of $D_{\Lambda,s,\mathbf{c}}(\cdot)$:

Lemma 2.16. *Let H be a d-dimensional subspace of $\mathbb{R}^n$ and let Λ be a lattice that spans H. For any $\epsilon > 0$, $s \geq 2 \cdot \eta_\epsilon(\Lambda)$, $\mathbf{y} \in \Lambda$, and $\mathbf{c} \in H$,*

$$D_{\Lambda,s,\mathbf{c}}(\mathbf{y}) \leq 2^{-d} \cdot \frac{1+\epsilon}{1-\epsilon}.$$

Proof. First, observe

$$D_{\Lambda,s,\mathbf{c}}(\mathbf{y}) = \frac{\rho_{H,s,\mathbf{c}}(\mathbf{y})}{\rho_{H,s,\mathbf{c}}(\Lambda)} \leq \frac{1}{s^d \det(\Lambda^*) \cdot (1-\epsilon)},$$

because $\rho_{H,s,\mathbf{c}}(\mathbf{y}) \leq 1$ and by Lemma 2.15. Now we also have

$$1 \leq \rho_{H,s/2}(\Lambda) \leq (s/2)^d \det(\Lambda^*) \cdot (1 + \epsilon),$$

again by Lemma 2.15 and because $s/2 \geq \eta_\epsilon(\Lambda)$. Combining the inequalities, we get the result. □

3 Worst-Case Problems on Cyclic Lattices

In this section we introduce a variety of worst-case computational problems on cyclic lattices, and exhibit some (worst-case to worst-case) reductions among them. We specify these problems in their *search* versions, rather than as decisional problems. Due to the algebraic nature of cyclic lattices and our hash function, we will find it useful to formulate problems that ask for short lattice vectors *within a specified cyclotomic subspace* of $\mathbb{R}^n$; as a group, we call these *cyclotomic* problems. After defining these problems, we show that certain cyclotomic problems are as hard as the more standard problems on cyclic lattices.

When formulating computational lattice problems it is customary to assume that the input basis contains integer entries (and we do so implicitly in all the problem definitions below). This restriction is without loss of generality, because rational entries can always be multiplied by their least common denominator, which just scales the lattice by some constant.

For generality, the problems below are parameterized by some arbitrary function ζ of the input lattice, and the quality of a solution is measured relative to ζ. Typically, ζ will be some appropriate lattice parameter, e.g. λ_1 or the lattice's smoothing parameter.

3.1 Definitions

Definition 3.1 (SubSIVP). The *cyclotomic (generalized) short independent vectors problem*, $\textsc{SubSIVP}^{\zeta}_{\gamma}$, given an n-dimensional full-rank cyclic lattice basis $\mathbf{B}$ and an integer polynomial $\Phi(\alpha) \neq 0 \bmod (\alpha^n - 1)$ that divides $\alpha^n - 1$, asks for a set of $\dim(H_\Phi)$ linearly independent (sub)lattice vectors $\mathbf{S} \subset \mathcal{L}(\mathbf{B}) \cap H_\Phi$ such that $\|\mathbf{S}\| \leq \gamma(n) \cdot \zeta(\mathcal{L}(\mathbf{B}) \cap H_\Phi)$.

Definition 3.2 (SubSVP). The *cyclotomic (generalized) short vector problem*, $\textsc{SubSVP}^{\zeta}_{\gamma}$, given an n-dimensional full-rank cyclic lattice basis $\mathbf{B}$ and an integer polynomial $\Phi(\alpha) \neq 0 \bmod (\alpha^n - 1)$ that divides $\alpha^n - 1$, asks for a (sub)lattice vector $\mathbf{c} \in \mathcal{L}(\mathbf{B}) \cap H_\Phi$ such that $\|\mathbf{c}\| \leq \gamma(n) \cdot \zeta(\mathcal{L}(\mathbf{B}) \cap H_\Phi)$.

Definition 3.3 (SubIncSVP). The *cyclotomic incremental (generalized) short vector problem*, $\textsc{SubIncSVP}^{\zeta}_{\gamma}$, given an n-dimensional full-rank cyclic lattice basis $\mathbf{B}$, an integer polynomial $\Phi(\alpha) \neq 0 \bmod (\alpha^n - 1)$ that divides to $\alpha^n - 1$, and a nonzero (sub)lattice vector $\mathbf{c} \in \mathcal{L}(\mathbf{B}) \cap H_\Phi$ such that $\|\mathbf{c}\| > \gamma(n) \cdot \zeta(\mathcal{L}(\mathbf{B}) \cap H_\Phi)$, asks for a nonzero (sub)lattice vector $\|\mathbf{c}'\| \in \mathcal{L}(\mathbf{B}) \cap H_\Phi$ such that $\|\mathbf{c}'\| \leq \|\mathbf{c}\|/2$.

Note that Definitions 3.2 and 3.3 are slightly more general than the standard (incremental) shortest vector problems, because their approximation factors are relative to an arbitrary function ζ of the sublattice, rather than λ_1.

The standard well-studied lattice problems (on cyclic lattices) are simply special cases of the above problems. For example, the *shortest vector problem* SVP_γ is simply $\textsc{SubSVP}^{\zeta}_{\gamma}$ with $\zeta = \lambda_1$ and $\Phi(\alpha) = 1$. The *generalized independent vectors problem* $\text{GIVP}^{\zeta}_{\gamma}$, as described by Micciancio, is simply $\textsc{SubSIVP}^{\zeta}_{\gamma}$ with $\Phi(\alpha) = 1$. The *shortest independent vectors problem* SIVP_γ is $\text{GIVP}^{\zeta}_{\gamma}$ with $\zeta = \lambda_n$.

3.2 Reductions Among Problems

In this section we give some standard (worst-case to worst-case) reductions among the the cyclotomic problems defined above, and the more standard lattice problems from the literature.

Micciancio coined the term *lattice-preserving* to describe a reduction from problem A to problem B which invokes its B-oracle only on the lattice specified in the instance of problem A. Following in this vein, we define a *sublattice-preserving* reduction between two *cyclotomic* problems to have the property that all calls to the B oracle are on the same cyclic lattice *and* cyclotomic subspace as specified in the problem A instance.

Proposition 3.4. *For any $\zeta, \gamma(n)$, there is a deterministic, polynomial-time sublattice-preserving reduction from* $\textsc{SubSVP}_\gamma^\zeta$ *to* $\textsc{SubIncSVP}_\gamma^\zeta$.

Proof. Given an instance $(\mathbf{B}, \Phi(\alpha))$ of $\textsc{SubSVP}_\gamma^\zeta$, we will use the following basic strategy: starting from some (possibly very long) nonzero $\mathbf{c} \in \mathcal{L}(\mathbf{B}) \cap H_\Phi$, iteratively reduce the length of $\mathbf{c}$ by invoking the oracle for $\textsc{SubIncSVP}_\gamma^\zeta$ on $(\mathbf{B}, \Phi(\alpha), \mathbf{c})$ until the oracle fails, which indicates that $\|\mathbf{c}\| \leq \gamma(n) \cdot \zeta(\mathcal{L}(\mathbf{B}) \cap H_\Phi)$.

It now suffices to show how to find such an initial $\mathbf{c}$ and bound its norm (and hence, the number of iterations). We claim that for some i, $\mathbf{c}(\alpha) = \mathbf{b}_i(\alpha)\Phi(\alpha) \bmod (\alpha^n - 1)$ is nonzero. For suppose not: then by Lemma 2.6, $\Phi \neq 0$ is orthogonal to $\mathbf{b}_i$ for every i, so the space spanned by $\mathbf{B}$ is not full-dimensional, which contradicts the assumption that $\mathbf{B}$ is full-rank.

Now, because $\Phi(\alpha)$ divides $\alpha^n - 1$, it is the product of cyclotomic factors of $\alpha^n - 1$. All such factors are computable in time $\text{poly}(n)$, and there are at most n such factors, so any $\Phi(\alpha)$ has coefficients of length $\text{poly}(n)$. This implies that $\|\mathbf{c}\| \leq 2^{\text{poly}(n)}$, so the number of iterations in the reduction is $\text{poly}(n)$. ☐

The following lemma will help us reduce problems asking for *many linearly independent vectors* to problems asking for a *single vector*:

Lemma 3.5. *Let $\Phi(\alpha) \in \mathbb{Z}[\alpha]$ equal $(\alpha^n - 1)/\Phi_k(\alpha)$ for some $k \,|\, n$. Then for any cyclic lattice $\Lambda \subseteq \mathbb{Z}^n$ and any nonzero $\mathbf{c} \in \Lambda \cap H_\Phi$, vectors*

$$\mathbf{c},\ \text{rot}(\mathbf{c}),\ \ldots,\ \text{rot}^{\deg(\Phi_k)-1}(\mathbf{c})$$

are linearly independent. As a consequence,

$$\lambda_1(\Lambda \cap H_\Phi) = \cdots = \lambda_{\dim(H_\Phi)}(\Lambda \cap H_\Phi).$$

Proof. Because $\mathbf{c} \neq 0$, $\mathbf{c}(\alpha) \in \mathbb{Z}[\alpha]$, and $\Phi(\alpha) \,|\, \mathbf{c}(\alpha)$, $\mathbf{c}(\alpha)$ is not divisible by $\Phi_k(\alpha)$. Then by Lemma 2.7, the rotations of $\mathbf{c}$ are linearly independent. Now let $\mathbf{c} \in \Lambda \cap H_\Phi$ be such that $\|\mathbf{c}\| = \lambda_1(\Lambda \cap H_\Phi)$. By Lemma 2.9, $\dim(H_\Phi) = \deg(\Phi_k)$. Because $\|\text{rot}^i(\mathbf{c})\| = \|\mathbf{c}\|$ for any i, the result follows. ☐

Corollary 3.6. *For any $\zeta, \gamma(n)$, there exists a deterministic, polynomial-time sublattice-preserving reduction from* $\textsc{SubSIVP}_\gamma^\zeta$ *instances $(\mathbf{B}, \Phi(\alpha))$ where $\Phi(\alpha) = (\alpha^n - 1)/\Phi_k(\alpha)$ for some $k \,|\, n$ to* $\textsc{SubSVP}_\gamma^\zeta$*, which makes exactly one oracle call.*

When the dimension n of a cyclic lattice is *prime*, $\alpha^n - 1$ factors as $\Phi_n(\alpha) \cdot \Phi_1(\alpha)$. In this case, there is a very tight connection between SIVP and SVP (in an appropriate subspace):

Proposition 3.7. *For any $\gamma(n)$, there is a deterministic, polynomial-time lattice-preserving reduction from* $\text{SIVP}_{\max(n,2\gamma)}$ *on a cyclic lattice of prime dimension n to* $\textsc{SubSVP}_\gamma^{\lambda_1}$*. The reduction makes exactly one oracle call, on an instance for which $\Phi(\alpha) = \Phi_1(\alpha) = \alpha - 1$.*

Proof. The main idea behind the proof is as follows: first, we use the SubSVP oracle to find a short vector in $\mathcal{L}(\mathbf{B}) \cap H_{\Phi_1}$, then rotate it to yield $n-1$ linearly independent vectors. For the nth vector, we take the shortest vector in $\mathcal{L}(\mathbf{B}) \cap H_{\Phi_n}$, which can be found efficiently; furthermore, it is an n-approximation to the shortest vector in $\mathcal{L}(\mathbf{B}) \backslash H_{\Phi_1}$.

We now give the full proof. Given an integer lattice basis $\mathbf{B}$ of a cyclic lattice of prime dimension n, invoke the SubSVP oracle on $(\mathbf{B}, \Phi_1(\alpha))$, yielding a lattice vector $\mathbf{c} \in \mathcal{L}(\mathbf{B}) \cap H_{\Phi_1}$ such that $\|\mathbf{c}\| \leq \gamma(n) \cdot \lambda_1(\mathcal{L}(\mathbf{B}) \cap H_{\Phi_1})$. Looking ahead, the rotations of $\mathbf{c}$ will provide $n-1$ linearly independent vectors of length $\|\mathbf{c}\|$, however we will need one more vector (outside H_{Φ_1}) to solve SIVP.

Now let $s_i = \sum_{j=1}^{n} (\mathbf{b}_i)_j = \mathbf{b}_i(1)$ for $i = 1, \ldots n$. Because $\alpha - 1$ cannot divide every $\mathbf{b}_i(\alpha)$ (otherwise $\mathcal{L}(\mathbf{B}) \subset H_{\Phi_1}$, so $\mathcal{L}(\mathbf{B})$ would not be full-rank), some s_i must be non-zero. Let $g = \gcd(s_1, \ldots, s_n) \neq 0$, and let $\mathbf{g} = (g, g, \ldots, g)$. Output the vectors $\mathbf{S} = (\mathbf{c}, \mathrm{rot}(\mathbf{c}), \ldots, \mathrm{rot}^{n-2}(\mathbf{c}), \mathbf{g})$.

To prove correctness of the reduction, we first show that $\mathbf{g} \in \mathcal{L}(\mathbf{B})$. Note that for every i, $\mathbf{s}_i = \mathbf{b}_i \otimes (1, 1, \ldots, 1) = (s_i, s_i, \ldots, s_i) \in \mathcal{L}(\mathbf{B})$. By the extended Euclidean algorithm, $\mathbf{g}$ is an integer combination of the $\mathbf{s}_i$ vectors, hence $\mathbf{g} \in \mathcal{L}(\mathbf{B})$.

Claim 3.8. *The vectors in* $\mathbf{S}$ *are linearly independent.*

Proof. Because n is prime, $(\alpha^n - 1)/\Phi_1(\alpha) = \Phi_n(\alpha)$ is irreducible in $\mathbb{Z}[\alpha]$, so by Lemma 3.5 the $n-1$ rotations of $\mathbf{c}$ in $\mathbf{S}$ are linearly independent. Further, $\mathbf{g} \notin H_{\Phi_1}$ while $\mathrm{rot}^i(\mathbf{c}) \in H_{\Phi_1}$ for every i (Lemma 2.8), so $\mathbf{S}$ consists of n linearly independent vectors from $\mathcal{L}(\mathbf{B})$. ☐

We now analyze the approximation factor of the reduction. First, we bound $\lambda_n(\mathcal{L}(\mathbf{B}))$:

Claim 3.9.

$$\lambda_n(\mathcal{L}(\mathbf{B})) \geq \max\left(\frac{g}{\sqrt{n}}, \frac{\lambda_1(\mathcal{L}(\mathbf{B}) \cap H_{\Phi_1})}{2}\right).$$

Proof. Let $\mathbf{T}$ be some full-rank set of nonzero vectors in $\mathcal{L}(\mathbf{B})$ such that $\|\mathbf{T}\| = \lambda_n(\mathcal{L}(\mathbf{B}))$. Then $\mathbf{T}$ must contain some $\mathbf{u} \in \mathcal{L}(\mathbf{B}) \backslash H_{\Phi_1}$, because $\dim(H_{\Phi_1}) = n-1$. Let $\mathbf{u} = \sum_{i=1}^{n} a_i \mathbf{b}_i$ for integers $a_1, \ldots, a_n$. Because $\Phi_1(\alpha)$ does not divide $\mathbf{u}(\alpha)$, $\mathbf{u}(1) = \sum_{j=1}^{n} \mathbf{u}_j \neq 0$. Further, $\mathbf{u}(1) = \sum_{i=1}^{n} a_i \mathbf{b}_i(1)$, so g divides $\mathbf{u}(1)$. Therefore $\|\mathbf{u}\|_1 \geq |\mathbf{u}(1)| \geq g$, which implies $\lambda_n(\mathcal{L}(\mathbf{B})) = \|\mathbf{T}\| \geq \|\mathbf{u}\| \geq \|\mathbf{u}\|_1/\sqrt{n} \geq g/\sqrt{n}$.

Furthermore, $\mathbf{T}$ must contain some $\mathbf{v} \in \mathcal{L}(\mathbf{B}) \backslash H_{\Phi_n}$, because $\dim(H_{\Phi_n}) = 1$. Now $\mathbf{v}' = \mathrm{rot}(\mathbf{v}) - \mathbf{v}$ is identified with the polynomial $(\alpha - 1) \cdot \mathbf{v}(\alpha) \bmod (\alpha^n - 1)$, so $0 \neq \mathbf{v}' \in \mathcal{L}(\mathbf{B}) \cap H_{\Phi_1}$. Then by the triangle inequality we have

$$\lambda_1(\mathcal{L}(\mathbf{B}) \cap H_{\Phi_1}) \leq \|\mathbf{v}'\| \leq 2\|\mathbf{v}\| \leq 2\|\mathbf{T}\| = 2\lambda_n(\mathcal{L}(\mathbf{B})).$$ ☐

Now, $\|\mathbf{S}\| = \max(g\sqrt{n}, \gamma(n) \cdot \lambda_1(\mathcal{L}(\mathbf{B}) \cap H_{\Phi_1}))$. By taking both cases of $\|\mathbf{S}\|$ and invoking Claim 3.9 with each, we get

$$\frac{\|\mathbf{S}\|}{\lambda_n(\mathcal{L}(\mathbf{B}))} \leq \max(n, 2\gamma(n)).$$

This completes the proof of Proposition 3.7. ☐

We also have, for arbitrary (not necessarily prime) n, a reduction from SVP to SubSVP:

Proposition 3.10. *For any* $\gamma(n)$*, there is a deterministic, polynomial-time lattice-preserving reduction from* $\mathrm{SVP}_{\max(n,\gamma)}$ *to* $\mathrm{SubSVP}_{\gamma}^{\lambda_1}$*. The reduction calls the oracle exactly once, on an instance for which* $\Phi(\alpha) = \Phi_1(\alpha) = \alpha - 1$.

Proof. The reduction and proof of correctness are very similar to the one from the proof of Proposition 3.7: on input $\mathbf{B}$, call the SubSVP oracle on $(\mathbf{B}, \Phi_1(\alpha))$, yielding a vector $\mathbf{c} \in \mathcal{L}(\mathbf{B}) \cap H_{\Phi_1}$ such that $\|\mathbf{c}\| \le \gamma(n) \cdot \lambda_1(\mathcal{L}(\mathbf{B}) \cap H_{\Phi_1})$. Additionally, construct the vector $\mathbf{g}$ as above, and output the shorter of $\mathbf{c}$ and $\mathbf{g}$.

Using reasoning as above, we can show that $\lambda_1(\mathcal{L}(\mathbf{B})) \ge \min(g/\sqrt{n}, \lambda_1(\mathcal{L}(\mathbf{B}) \cap H_{\Phi_1}))$. Then by considering both cases of $\lambda_1(\mathcal{L}(\mathbf{B}))$, we can show that

$$\frac{\min(\|\mathbf{g}\|, \|\mathbf{c}\|)}{\lambda_1(\mathcal{L}(\mathbf{B}))} \le \max(n, \gamma(n)). \qquad \square$$

4 Generalized Compact Knapsacks

Definition 4.1 ([15], Definition 4.1). For any ring R, subset $S \subset R$ and integer $m \ge 1$, the generalized knapsack function family $\mathcal{H}(R, S, m) = \{f_{\mathbf{a}} : S^m \to R\}_{\mathbf{a} \in R^m}$ is defined by

$$f_{\mathbf{a}}(\mathbf{x}) = \sum_{i=1}^{m} x_i \cdot a_i.$$

In our knapsack function for security parameter n, R is the ring $R = (\mathbb{Z}_p^n, +, \otimes)$ of n-dimensional vectors over $\mathbb{Z}_p$, where $p = n^{O(1)}$ but *need not be prime*, with vector addition and convolution product $\otimes$.

This choice of ring admits very efficient implementations of the knapsack function: using a Fast Fourier Transform algorithm (which works for any n), convolution can be performed in $O(n \log n)$ operations in $\mathbb{Z}_p$, and addition of two vectors takes time $O(n \log p) = O(n \log n)$. Furthermore, by choosing a p such that $\mathbb{Z}_p$ has an element of multiplicative order n, we can compute the Fourier transform mod p using modular (rather than floating-point) arithmetic. The resulting time complexity of the function is $O(m \cdot n \cdot \mathrm{poly}(\log n))$, with key size $O(m \cdot n \log n)$.

4.1 How to Find Collisions

Here we show how to find collisions in the compact knapsack function when $S = [0, D]^n$ for some $D = p^{\Theta(1)}$, for which Micciancio proved that the function was one-way (under suitable assumptions). Our attacks actually do more than just find arbitrary collisions; in fact, they find second preimages for many elements of the domain, thereby violating the definition of universal one-wayness as well. In the following we write $\mathbf{X} \in S^m \subset \mathbb{Z}_p^{n \times m}$ as an element of the domain, and $\mathbf{A} \in R^m = \mathbb{Z}_p^{n \times m}$ as a uniformly-chosen key.

First observe that $f_{\mathbf{A}}$ is linear: $f_{\mathbf{A}}(\mathbf{X}) + f_{\mathbf{A}}(\mathbf{X}') = f_{\mathbf{A}}(\mathbf{X} + \mathbf{X}')$. Therefore, for any fixed $\mathbf{X}'$ such that $\|\mathbf{X}'\|_\infty < D$ and a random key $\mathbf{A}$, to find a collision with $\mathbf{X}'$ it suffices to find a nonzero $\mathbf{X} \in S^m$ such that $f_{\mathbf{A}}(\mathbf{X}) = \mathbf{0}$ and $\|\mathbf{X}\|_\infty = 1$. In fact, our attack will be even stronger: we demonstrate a *fixed* $\mathbf{X} \ne 0$, *oblivious* to the key $\mathbf{A}$, for which $f_{\mathbf{A}}(\mathbf{X}) = \mathbf{0}$ with non-negligible probability (over the choice of $\mathbf{A}$).

We define $\mathbf{X}$ by its representation as an m-tuple of polynomials in the ring $\mathbb{Z}_p[\alpha]/(\alpha^n - 1)$. In this polynomial representation, $f_{\mathbf{A}}(\mathbf{X})$ corresponds to $\sum_{i=1}^{m} \mathbf{x}_i(\alpha) \cdot \mathbf{a}_i(\alpha) \bmod (\alpha^n - 1)$. For any small positive integer divisor q of n (including $q = 1$), we can define $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_m)$ as follows: let

$$\mathbf{x}_1(\alpha) = \frac{\alpha^n - 1}{\alpha^q - 1} = \alpha^{n-q} + \alpha^{n-2q} + \cdots + 1,$$

and let $\mathbf{x}_j(\alpha) = 0$ for all $j \neq 1$. Then $\mathbf{X} \in S^m$, $\|\mathbf{X}\|_\infty = 1$, and $f_{\mathbf{A}}(\mathbf{X})$ corresponds to $\mathbf{a}_1(\alpha) \cdot \mathbf{x}_1(\alpha)$. Now suppose $\mathbf{a}_1(\alpha)$ is divisible by $\alpha^q - 1$, which happens with probability $1/p^q$ over the uniform choice of $\mathbf{A}$. Then $f_{\mathbf{A}}(\mathbf{X}) = 0$ because $(\alpha^n - 1)$ divides $\mathbf{a}_1(\alpha) \cdots \mathbf{x}_1(\alpha)$.

4.2 How to Achieve Collision-Resistance

The essential fact enabling the above attack is that $(\alpha^n - 1)$ is not *irreducible* in $\mathbb{Z}_p[\alpha]$, so $\mathbb{Z}_p[\alpha]/(\alpha^n - 1)$ is not an *integral domain*. That is, for many non-zero $\mathbf{a}(\alpha)$, it is easy to find non-zero $\mathbf{x}(\alpha)$ (having small coefficients) such that $\mathbf{a}(\alpha) \cdot \mathbf{x}(\alpha) = 0 \bmod (\alpha^n - 1)$. In particular, when we examine $\mathbf{a}(\alpha), \mathbf{x}(\alpha) \bmod (\alpha^n - 1)$ in their Chinese remainder representations, each of the components is zero for either $\mathbf{a}(\alpha)$ or $\mathbf{x}(\alpha)$ (or both).

To circumvent our particular attack, we can enforce an *algebraic constraint* on $\mathbf{X}$. Informally, we require every $\mathbf{x}_i(\alpha)$ to be divisible *over* $\mathbb{Z}[\alpha]$ by $\frac{\alpha^n - 1}{\Phi_k(\alpha)}$ for some fixed $k \mid n$. Then in the Chinese remainder representation, all but one component of $\mathbf{x}_i(\alpha)$ is zero, so the evaluation of $f_{\mathbf{A}}(\mathbf{X})$ is essentially performed mod $\Phi_k(\alpha)$.

Note that while $\Phi_k(\alpha)$ is irreducible over $\mathbb{Z}[\alpha]$, it *may* still be reducible over $\mathbb{Z}_p[\alpha]$. Therefore constraining $\mathbf{X}$ in the above way may *not* necessarily place the calculation of $f_A(\mathbf{X})$ in an integral domain. Furthermore, the constraint is crafted specifically to prevent our attack, but not to prevent any other potential attacks on the function that may remain undiscovered. Nevertheless (and perhaps quite surprisingly), it proves to be exactly what is needed to attain collision-resistance, as our security reduction will demonstrate.

Formally, we consider the generalized compact knapsack function where the set $S = S_{D,\Phi} \subset \mathbb{Z}_p^n$ for some bound D on the max-norm of $\mathbf{X}$ (recall that $\|\mathbf{x}\|_\infty \in [0, p/2]$ for any $\mathbf{x} \in \mathbb{Z}_p^n$), and $\Phi(\alpha) = \frac{\alpha^n - 1}{\Phi_k(\alpha)}$ for some $k \mid n$. For a value $v \in \mathbb{Z}_p$, define $v_{\mathbb{Z}}$ to be the unique integer in the range $(-p/2, p/2]$ representing v as a residue, and for a vector $\mathbf{x} \in \mathbb{Z}_p^n$ define the vector $\mathbf{x}_{\mathbb{Z}} \in \mathbb{Z}^n$ similarly. Now we define $S_{D,\Phi}$ as:

$$S_{D,\Phi} = \{\mathbf{x} \in \mathbb{Z}_p^n : \|\mathbf{x}\|_\infty \leq D \text{ and } \Phi(\alpha) \text{ divides } \mathbf{x}_{\mathbb{Z}}(\alpha) \text{ in } \mathbb{Z}[\alpha]\}. \tag{1}$$

4.3 How to Get a (Useful) Hash Function

In order to verify that our knapsack is a *hash* function, we must compare the size of the domain $S_{D,\Phi}^m$ to the size of the function's range. In addition, practical usage requires efficient one-to-one encodings of *bit strings* into elements of the domain, and of range elements back to bit strings.

Both tasks are most easily done when n is prime and $\Phi(\alpha) = \alpha - 1$. Given a string $w \in \{0,1\}^\ell$, where $\ell = m \cdot (n-1) \cdot \lfloor \log D \rfloor$, encode w in the following way: first, break w into m chunks representing vectors $\mathbf{w}_i \in [0, D-1]^{n-1}$ for $i = 1, \ldots, m$. For each i, and for $j = 0, \ldots, n-2$, let $(\mathbf{x}_i)_j = \pm(\mathbf{w}_i)_j$, where the signs are iteratively chosen to satisfy the invariant that every partial sum $\sum_{k=0}^{j} (x_i)_k \in [-D, D]$. Finally, for every i let $(\mathbf{x}_i)_{n-1} = -\sum_{j=0}^{n-2} (x_i)_j \in [-D, D]$, so that $\mathbf{x}_i(1) = \sum_{j=0}^{n-1} (\mathbf{x}_i)_j = 0$, hence $\alpha - 1$ divides $\mathbf{x}_i(\alpha)$ and $\|\mathbf{x}_i\|_\infty \leq D$.

To encode the output, first notice that $\alpha - 1$ divides $\mathbf{y}(\alpha)$, where $\mathbf{y} = f_{\mathbf{A}}(\mathbf{X})$. Therefore it is sufficient to write $(\mathbf{y})_j$ in binary for $j = 0, \ldots, n-2$. This can be done using $(n-1) \cdot \lceil \log p \rceil$ bits. Therefore, the function shrinks its input by a factor of $\frac{m \lfloor \log D \rfloor}{\lceil \log p \rceil}$, which for appropriate choices of parameters is larger than 1.

5 The Main Reduction

Due to the reductions among worst-case problems on cyclic lattices explored in Section 3.2, the security of our hash function can be established by reducing the worst-case problem $\textsc{SubIncSVP}_\gamma^{\eta_\epsilon}$ to finding collisions in $\mathcal{H}(\mathbb{Z}_p^n, S_{D,\Phi}, m)$. Because collision-resistance is meaningful even for functions that do not shrink their input, we exhibit a general reduction in Theorem 5.1, then consider special cases of hash functions in the corollaries that follow.

Theorem 5.1. *For any polynomially-bounded functions $D(n)$, $m(n)$, $p(n)$ and negligible function $\epsilon(n)$ such that $p(n) \geq 8n^{2.5} \cdot m(n)D(n)$ and $\gamma(n) \geq 16n \cdot m(n)D(n)$, there is a probabilistic polynomial-time reduction from $\textsc{SubIncSVP}_\gamma^{\eta_\epsilon}$ instances $(\mathbf{B}, \Phi(\alpha), \mathbf{c})$ where $\frac{\alpha^n - 1}{\Phi(\alpha)} = \Phi_k(\alpha)$ for some $k \mid n$ to finding collisions in $\mathcal{H}(\mathbb{Z}_{p(n)}^n, S_{D(n),\Phi}, m(n))$.*

Roadmap to the proof. First we describe a reduction that, given a collision-finding oracle $\mathcal{F}$, attempts to solve SubIncSVP. The remainder of the proof is a series of claims that establish the correctness of the reduction. Claim 5.2 shows that the reduction feeds $\mathcal{F}$ a properly-distributed input. Claim 5.3 establishes that the reduction's output vector is in the proper sublattice. Claims 5.4 and 5.5 show that, with good likelihood, the output is both nonzero and significantly shorter than the input lattice vector (respectively).

Proof. Assume that $\mathcal{F}$ finds collisions in the specified hash family, for infinitely many n and $\Phi(\alpha)$, with probability at least $1/q(n)$ for some polynomial $q(\cdot)$. For shorthand, we will abbreviate $H = H_\Phi$ and let $d = \dim(H)$ throughout the proof. We assume wlog that $d \geq 3$, because efficient algorithms are known for SVP when $d = 1, 2$ (we omit details).

Our reduction proceeds as follows: on input $(\mathbf{B}, \mathbf{c})$ where $\mathbf{c} \in \mathcal{L}(\mathbf{B}) \cap H$,

1. For $i = 1$ to m,
   - Generate uniform $\mathbf{v}_i \in \mathcal{L}(\mathbf{B}) \cap H \cap \mathcal{P}(\mathrm{Rot}^d(\mathbf{c}))$. (See [17] for algorithms.)
   - Generate noise $\mathbf{y}_i \in H$ according to $D_{H,s}$ for $s = 2\|\mathbf{c}\|/\gamma(n)$. Let $\mathbf{y}_i' = \mathbf{y}_i \bmod \mathbf{B}$.
   - Choose $\mathbf{b}_i$ (as described below) so that $\mathrm{Rot}^n(\mathbf{c}) \cdot \mathbf{b} = \mathbf{v}_i + \mathbf{y}_i'$, and let $\mathbf{a}_i = \lfloor \mathbf{b}_i \cdot p \rceil$. Choosing $\mathbf{b}_i$ is done by breaking it into two parts: $\mathbf{b}_i^1 = ((\mathbf{b}_i)_0, \ldots, (\mathbf{b}_i)_{d-1})^T$, and $\mathbf{b}_i^2 = ((\mathbf{b}_i)_d, \ldots, (\mathbf{b}_i)_{n-1})^T$. First, pick $\mathbf{b}_i^2$ according to $I^{n-d} = U([0,1))^{n-d}$. Then solve for $\mathbf{b}_i^1$ as follows: let $\mathbf{G} \in \mathbb{R}^{d \times n}$ be such that $\mathbf{G} \cdot \mathrm{Rot}^d(\mathbf{c}) = \mathbf{I}_d$, the $d \times d$ identity matrix. (Such a $\mathbf{G}$ exists because $\mathrm{Rot}^d(\mathbf{c})$ has column rank d, and it can be found via Gaussian elimination.) Then $\mathbf{b}_i^1 = \mathbf{G} \cdot (\mathbf{v}_i + \mathbf{y}_i' - \mathbf{w}_i)$, where $\mathbf{w}_i = \mathrm{Rot}^n(\mathbf{c}) \cdot (0, \ldots, 0, (\mathbf{b}_i)_d, \ldots, (\mathbf{b}_i)_{n-1})^T$.

2. Give $\mathbf{A} = (\mathbf{a}_1 \bmod p, \ldots, \mathbf{a}_m \bmod p)$ to the collision-finding oracle $\mathcal{F}$. Get a collision $\mathbf{X} \neq \mathbf{X}'$ such that $\|\mathbf{X}\|_\infty, \|\mathbf{X}'\|_\infty \leq D$, and $\Phi(\alpha)$ divides every $\mathbf{x}_i(\alpha), \mathbf{x}_i'(\alpha)$. Let $\mathbf{Z} = \mathbf{X} - \mathbf{X}'$, and note that $\|\mathbf{Z}\|_\infty \leq 2D$ and $\Phi(\alpha)$ divides every $\mathbf{z}_i(\alpha)$.

3. Output the vector

$$\begin{aligned} \mathbf{c}' &= \sum_{i=1}^{m} (\mathbf{v}_i + \mathbf{y}_i' - \mathbf{y}_i) \otimes \mathbf{z}_i - \mathbf{c} \otimes \frac{\sum_{i=1}^m \mathbf{a}_i \otimes \mathbf{z}_i}{p} &(2)\\ &= \sum_{i=1}^{m} (\mathbf{v}_i + \mathbf{y}_i' - \mathbf{y}_i - \frac{\mathbf{c} \otimes \mathbf{a}_i}{p}) \otimes \mathbf{z}_i. &(3)\end{aligned}$$

The following claim follows from Lemma 2.12 and straightforward manipulations of statistical distance:

Claim 5.2. *The probability that $\mathcal{F}$ outputs a valid collision is non-negligible:*

$$\Pr[(\mathbf{X}, \mathbf{X}') \text{ is a valid collision}] \geq 1/q(n) - m(n) \cdot \epsilon(n)/2.$$

Proof. It suffices to bound the statistical distance $\Delta(\mathbf{A}, U(\mathbb{Z}_p^{nm}))$ by $m\epsilon/2$. Each $\mathbf{a}_i$ is independently generated, so by the triangle inequality, $\Delta(\mathbf{A}, U(\mathbb{Z}_p^{nm})) \leq m \cdot \Delta(\mathbf{a}_i \bmod p, U(\mathbb{Z}_p^n))$. Now $\mathbf{a}_i \bmod p = \lfloor (\mathbf{b}_i \bmod 1) \cdot p \rceil$, so $\Delta(\mathbf{a}_i \bmod p, U(\mathbb{Z}_p^n)) \leq \Delta(\mathbf{b}_i \bmod 1, I^n)$.

Let $\mathbf{b}_i^1 = ((\mathbf{b}_i)_0, \ldots, (\mathbf{b}_i)_{d-1})^T$, and $\mathbf{b}_i^2 = ((\mathbf{b}_i)_d, \ldots, (\mathbf{b}_i)_{n-1})^T$. By construction, $\mathbf{b}_i^2$ is uniform over $[0,1)^{n-d}$. Additionally, we have

$$\mathbf{b}_i^1 = \mathbf{G} \cdot (\mathbf{v}_i + \mathbf{y}'_i - \mathbf{w}_i) = \mathbf{G} \cdot (\mathbf{v}_i + \mathbf{y}'_i) - \mathbf{G} \cdot \mathbf{w}_i, \tag{4}$$

where $\mathbf{w}_i$ is a function of $\mathbf{b}_i^2$. Notice that $\mathbf{y}'_i$ is distributed according to $D_{H,s} \bmod \mathcal{P}(\mathbf{B})$, so by Lemma 2.12,

$$\Delta(\mathbf{y}'_i, U(\mathcal{P}(\mathbf{B}))) \leq \epsilon/2.$$

Because $\mathbf{v}_i$ is uniform over $\mathcal{L}(\mathbf{B}) \cap H \cap \mathcal{P}(\mathrm{Rot}^d(\mathbf{c}))$, we get

$$\Delta(\mathbf{v}_i + \mathbf{y}'_i \bmod \mathrm{Rot}^d(\mathbf{c}), U(\mathcal{P}(\mathrm{Rot}^d(\mathbf{c})))) \leq \epsilon/2,$$

which by definition of $\mathbf{G}$ implies

$$\Delta(\mathbf{G} \cdot (\mathbf{v}_i + \mathbf{y}'_i) \bmod 1, I^d) \leq \epsilon/2.$$

By Equation (4), we have that conditioned on *any* value $\mathbf{v} \in [0,1)^{n-d}$,

$$\Delta(\{\mathbf{b}_i^1 \bmod 1 \mid \mathbf{b}_i^2 = \mathbf{v}\}, I^d) \leq \epsilon/2.$$

Putting it all together and using $\delta_{\mathbf{b}_i^2}(\mathbf{v}) = \delta_{I^{n-d}}(\mathbf{v})$,

$$\begin{aligned}
\Delta(\mathbf{b}_i \bmod 1, I^n) &= \frac{1}{2} \int_{\mathbf{v} \in [0,1)^n} \left| \delta_{(\mathbf{b}_i^1 \bmod 1, \mathbf{b}_i^2)}(\mathbf{v}) - \delta_{I^n}(\mathbf{v}) \right| d\mathbf{v} \\
&= \frac{1}{2} \int_{\substack{\mathbf{u} \in [0,1)^d \\ \mathbf{v} \in [0,1)^{n-d}}} \left| \delta_{\{\mathbf{b}_i^1 \bmod 1 | \mathbf{b}_i^2 = \mathbf{v}\}}(\mathbf{u}) \cdot \delta_{\mathbf{b}_i^2}(\mathbf{v}) - \delta_{I^d}(\mathbf{u}) \cdot \delta_{I^{n-d}}(\mathbf{v}) \right| d\mathbf{u}\, d\mathbf{v} \\
&= \frac{1}{2} \int_{\mathbf{v} \in [0,1)^{n-d}} \delta_{I^{n-d}}(\mathbf{v}) \int_{\mathbf{u} \in [0,1)^d} \left| \delta_{\{\mathbf{b}_i^1 \bmod 1 | \mathbf{b}_i^2 = \mathbf{v}\}}(\mathbf{u}) - \delta_{I^d}(\mathbf{u}) \right| d\mathbf{u}\, d\mathbf{v} \\
&= \frac{1}{2} \int_{\mathbf{v} \in [0,1)^{n-d}} \delta_{I^{n-d}}(\mathbf{v}) \cdot 2\Delta(\{\mathbf{b}_i^1 \bmod 1 \mid \mathbf{b}_i^2 = \mathbf{v}\}, I^d)\, d\mathbf{v} \\
&\leq \frac{\epsilon}{2} \int_{\mathbf{v} \in [0,1)^{n-d}} \delta_{I^{n-d}}(\mathbf{v})\, d\mathbf{v} = \frac{\epsilon}{2},
\end{aligned}$$

as desired. ∎

Claim 5.3. *If $\mathcal{F}$ outputs a valid collision, $\mathbf{c}' \in \mathcal{L}(\mathbf{B}) \cap H$.*

Proof. First observe that $\mathcal{L}(\mathbf{B}) \cap H$ is a sublattice of $\mathcal{L}(\mathbf{B})$. We now examine the terms in Equation (2). By construction, $\mathbf{v}_i + \mathbf{y}'_i - \mathbf{y}_i \in \mathcal{L}(\mathbf{B}) \cap H$, and $\mathbf{z}_i \in \mathbb{Z}^n$, so the first summation is in $\mathcal{L}(\mathbf{B}) \cap H$. Next, $f_{\mathbf{A}}(\mathbf{Z}) = \sum_i \mathbf{a}_i \otimes \mathbf{z}_i = 0 \bmod p$ by the assumption that $\mathcal{F}$ outputs a valid collision, so $\frac{\sum_i \mathbf{a}_i \otimes \mathbf{z}_i}{p} \in \mathbb{Z}^n$. Since $\mathbf{c} \in \mathcal{L}(\mathbf{B}) \cap H$, the second term of Equation (2) is also in $\mathcal{L}(B) \cap H$. $\square$

Claim 5.4. *Conditioned on $\mathcal{F}$ outputting a collision,* $\Pr[\mathbf{c}' \neq 0] \geq 3/4$.

Proof. The main idea: because $\mathbf{c}' \in H$, $\mathbf{c}' = 0$ iff $\Phi_k(\alpha)$ divides $\mathbf{c}'(\alpha)$. Because $\Phi_k(\alpha)$ is irreducible, we can show that $\mathbf{c}'(\alpha) = 0 \bmod \Phi_k(\alpha)$ only when a sample from $D_{\mathcal{L}(\mathbf{B}) \cap H, s, -\mathbf{y}'_1}$ hits a certain target lattice point exactly. By Lemma 2.16, the probability of this event is small.

Throughout the proof we implicitly condition all probabilities on the event that $\mathcal{F}$ outputs a collision. Because $\Phi(\alpha)$ divides $\mathbf{c}'(\alpha)$ and $\Phi(\alpha) \cdot \Phi_k(\alpha) = (\alpha^n - 1)$, by Equation (3) we get

$$\mathbf{c}' = 0 \iff \sum_{i=1}^{m} \left(\mathbf{v}_i(\alpha) + \mathbf{y}'_i(\alpha) - \mathbf{y}_i(\alpha) + \frac{\mathbf{c}(\alpha)\mathbf{a}_i(\alpha)}{p} \right) \cdot \mathbf{z}_i(\alpha) = 0 \bmod \Phi_k(\alpha).$$

Since $\mathbf{Z} \neq 0$, there exists i such that $\mathbf{z}_i \neq 0$; assume without loss of generality that $i = 1$. Then let $\mathbf{h}(\alpha) = \sum_{i>1} (\mathbf{v}_i(\alpha) + \mathbf{y}'_i(\alpha) - \mathbf{y}_i(\alpha) + \frac{\mathbf{c}(\alpha) \cdot \mathbf{a}_i(\alpha)}{p}) \cdot \mathbf{z}_i(\alpha)$ and rearrange terms, yielding

$$\left(\mathbf{v}_1(\alpha) + \mathbf{y}'_1(\alpha) - \mathbf{y}_1(\alpha) + \frac{\mathbf{c}(\alpha) \cdot \mathbf{a}_1(\alpha)}{p} \right) \cdot \mathbf{z}_1(\alpha) = -\mathbf{h}(\alpha) \bmod \Phi_k(\alpha). \tag{5}$$

Now because $\mathbf{z}_1 \neq 0$ and $\Phi(\alpha)$ divides $\mathbf{z}_1(\alpha)$, it must be that $\mathbf{z}_1(\alpha) \neq 0 \bmod \Phi_k(\alpha)$. Since $\mathbb{Z}[\alpha]/\Phi_k(\alpha)$ is an integral domain, there exists at most one element $\mathbf{w}(\alpha) \in \mathbb{Z}[\alpha]/\Phi_k(\alpha)$ such that $\mathbf{w}(\alpha) \cdot \mathbf{z}_1(\alpha) = -\mathbf{h}(\alpha) \bmod \Phi_k(\alpha)$. If no such $\mathbf{w}(\alpha)$ exists, then $\mathbf{c}' \neq 0$ always, and we're done. If such a $\mathbf{w}(\alpha)$ exists, then $\mathbf{c}' = 0$ only when the multiplicand of $\mathbf{z}_1(\alpha)$ in Equation (5) equals $\mathbf{w}(\alpha)$. Then $\mathbf{c}' = 0$ only if:

$$(\mathbf{y}'_1 - \mathbf{y}_1)(\alpha) = \mathbf{w}(\alpha) - \frac{\mathbf{c}(\alpha) \cdot \mathbf{a}_1(\alpha)}{p} - \mathbf{v}_1(\alpha) \bmod \Phi_k(\alpha).$$

Now, $\mathbf{y}_1$ is independent of $\mathbf{v}_1$ and the coins of $\mathcal{F}$. Furthermore, conditioned on $\mathbf{y}'_1$, $\mathbf{y}_1$ is independent of $\mathbf{h}, \mathbf{z}_1$, and $\mathbf{a}_1$, because these variables depend only on $\mathbf{y}'_1$ and other independent coins. Therefore by averaging over these variables, it suffices to bound

$$M = \max_{\mathbf{h}'(\alpha)} \Pr\left[(\mathbf{y}'_1 - \mathbf{y}_1)(\alpha) = \mathbf{h}'(\alpha) \bmod \Phi_k(\alpha) \mid \mathbf{y}'_1 \right].$$

Because $\Phi(\alpha)$ divides $(\mathbf{y}'_1 - \mathbf{y}_1)(\alpha)$,

$$M = \max_{\mathbf{h}'(\alpha)} \Pr\left[(\mathbf{y}'_1 - \mathbf{y}_1)(\alpha) = \mathbf{h}'(\alpha) \bmod (\alpha^n - 1) \mid \mathbf{y}'_1 \right].$$

Now given $\mathbf{y}'_1$, $\mathbf{y}_1 - \mathbf{y}'_1$ is distributed according to $D_{\mathcal{L}(\mathbf{B}) \cap H_\Phi, s, -\mathbf{y}'_1}$ because $\mathbf{y}_1 - \mathbf{y}'_1 \mathcal{L}(\mathbf{B}) \cap H_\Phi$. By Lemma 2.16 and because $d \geq 3$,

$$M \leq 2^{-d} \cdot \frac{1+\epsilon}{1-\epsilon} \leq 1/4$$

for sufficiently large n. $\square$

Claim 5.5. *Conditioned on $\mathcal{F}$ outputting a collision,* $\Pr\left[\|\mathbf{c}'\| \leq \frac{\|\mathbf{c}\|}{2} \right] \geq 1/2$.

Proof. Throughout the proof we implicitly condition all probabilities on the event that $\mathcal{F}$ outputs a collision. First, it is sufficient to establish the bound $E[\|\mathbf{c}'\|] \leq \frac{\|\mathbf{c}\|}{4}$, because by Markov's inequality, this implies $\Pr\left[\|\mathbf{c}'\| > \frac{\|\mathbf{c}\|}{2}\right] \leq 1/2$. Now by Equation (2) and the triangle inequality,

$$\|\mathbf{c}'\| \quad \leq \quad \sum_{i=1}^{m} \left\|(\mathbf{v}_i + \mathbf{y}'_i - \frac{\mathbf{c} \otimes \mathbf{a}_i}{p}) \otimes \mathbf{z}_i\right\| + \sum_{i=1}^{m} \|\mathbf{y}_i \otimes \mathbf{z}_i\|. \tag{6}$$

Now using the fact that $\mathrm{Rot}^n(\mathbf{c}) \cdot \mathbf{b}_i = \mathbf{v}_i + \mathbf{y}'_i$, we get

$$\mathbf{v}_i + \mathbf{y}'_i - \frac{\mathbf{c} \otimes \mathbf{a}_i}{p} = \frac{\mathrm{Rot}^n(\mathbf{c}) \cdot \mathbf{b}_i \cdot p - \mathrm{Rot}^n(\mathbf{c}) \cdot \mathbf{a}_i}{p} = \frac{\mathrm{Rot}^n(\mathbf{c})(\mathbf{b}_i \cdot p - \mathbf{a}_i)}{p}.$$

Since $\|\mathbf{b}_i \cdot p - \mathbf{a}_i\|_\infty \leq 1/2$, we get

$$\left\|\mathbf{v}_i + \mathbf{y}'_i - \frac{\mathbf{c} \otimes \mathbf{a}_i}{p}\right\|_\infty \leq \frac{n\|\mathbf{c}\|}{2p}.$$

Now we use the fact that $\|\mathbf{z}_i\|_1 \leq 2n \cdot D$, yielding

$$\left\|(\mathbf{v}_i + \mathbf{y}'_i - \frac{\mathbf{c} \otimes \mathbf{a}_i}{p}) \otimes \mathbf{z}_i\right\|_\infty \leq \left\|\mathbf{v}_i + \mathbf{y}'_i - \frac{\mathbf{c} \otimes \mathbf{a}_i}{p}\right\|_\infty \cdot \|\mathbf{z}_i\|_1 \leq \frac{n^2\|\mathbf{c}\|D}{p}.$$

Finally, using the fact that $\|\mathbf{w}\| \leq \sqrt{n}\|\mathbf{w}\|_\infty$ for any n-dimensional vector $\mathbf{w}$ and summing over $i = 1, \ldots, m$, we get that the first summation in Equation (6) is at most $\frac{mn^{2.5}\|\mathbf{c}\|D}{p}$.

Next we analyze the second term of Equation (6). Conditioned on $\mathbf{y}'_i$, the distribution of $\mathbf{y}_i - \mathbf{y}'_i \in \mathcal{L}(\mathbf{B}) \cap H$ is $D_{\mathcal{L}(\mathbf{B}) \cap H, s, -\mathbf{y}'_i}$, and is independent of $\mathbf{A}, \mathbf{Z}$, and the coins of $\mathcal{F}$. Recall that $s = 2\|\mathbf{c}\|/\gamma(n) > 2\eta_\epsilon(\mathcal{L}(\mathbf{B}) \cap H)$, by assumption on the input to SubIncSVP. Also recall that $\mathbf{y}_i$ is chosen according to $D_{H,s}$, and that $\mathbf{z}_i \in H$. So by Lemma 2.14,

$$\begin{aligned}
E\left[\|\mathbf{y}_i \otimes \mathbf{z}_i\|^2 \mid \mathbf{y}'_i\right] \quad &= \quad E_{(\mathbf{y}_i - \mathbf{y}'_i) \leftarrow D_{\mathcal{L}(\mathbf{B}) \cap H, s, -\mathbf{y}'_i}}\left[\|((\mathbf{y}_i - \mathbf{y}'_i) - (-\mathbf{y}'_i)) \otimes \mathbf{z}_i\|^2\right] \\
&\leq \quad s^2\|\mathbf{z}_i\|^2 \cdot d \\
&\leq \quad s^2 n^2 D^2.
\end{aligned}$$

Because $\mathrm{Var}[X] = E[X^2] - E[X]^2 \geq 0$ for any random variable X, it must be that $E\left[\|\mathbf{y}_i \otimes \mathbf{z}_i\| \mid \mathbf{y}'_i\right] \leq n \cdot s \cdot D$. Adding up and averaging over all $\mathbf{y}'_i$, we get

$$\sum_{i=1}^{m} E\left[\|\mathbf{y}_i \otimes \mathbf{z}_i\|\right] \leq m \cdot n \cdot s \cdot D = \frac{2m \cdot n \cdot \|\mathbf{c}\| \cdot D}{\gamma(n)}.$$

Combining everything, we get:

$$\begin{aligned}
E[\|\mathbf{c}'\|] \quad &\leq \quad \frac{m \cdot n^{2.5} \cdot \|\mathbf{c}\| \cdot D}{p} + \frac{2m \cdot n \cdot \|\mathbf{c}\| \cdot D}{\gamma(n)} \\
&= \quad \|\mathbf{c}\| \cdot \left(\frac{m \cdot n^{2.5} \cdot D}{p} + \frac{2m \cdot n \cdot D}{\gamma(n)}\right).
\end{aligned}$$

Using the hypotheses $p \geq 8mn^{2.5}D$ and $\gamma(n) \geq 16mnD$, we get $E[\|\mathbf{c}'\|] \leq \|\mathbf{c}\|/4$, as desired. ☐

Then by Claims 5.4 and 5.5 and the union bound, we get that (conditioned on $\mathcal{F}$ producing a collision) the probability that $\mathbf{c}'$ is a solution to the SubIncSVP instance is at least 1/4. By Claim 5.2, the reduction solves SubIncSVP in the worst case with non-negligible probability, which can be amplified to high probability by standard repetition techniques. This completes the proof. $\square$

Putting it all together. Using the relationship between η_ϵ and λ_{n-1}, restricting n to be prime, and setting the knapsack parameters appropriately, we get collision-resistant hash functions:

Corollary 5.6. *For any $m(n) = \Theta(\log n)$, there exist $D(n) = \Theta(1)$ and $p(n) = n^{2.5+\Theta(1)}$ such that: $\mathcal{H}(\mathbb{Z}^n_{p(n)}, S_{D(n),\Phi_1(\alpha)}, m(n))$ is a hash function ensemble for which finding collisions for infinitely many* prime *n is at least as hard as solving* SVP_γ *with high probability in the worst case for infinitely many prime n within a factor $\gamma(n) = n \cdot \mathrm{poly}(\log n)$.*

Proof. We can choose $D(n)$ and $p(n)$ such that $\frac{m(n)\log D(n)}{\log p(n)} = \Theta(1)$ is greater than 1 (yielding a hash function) and satisfying the hypothesis of Theorem 5.1. Because n is prime, $(\alpha^n - 1)/\Phi_n(\alpha) = \Phi_1(\alpha)$, so by Theorem 5.1 and Lemma 3.4 we have an algorithm for $\mathrm{SubSVP}^{\eta_\epsilon(n)}_{\Theta(n \log n)}$ in H_{Φ_1}. By Lemma 2.13, this is an algorithm for $\mathrm{SubSVP}^{\lambda_{n-1}}_{n \cdot \mathrm{poly}(\log n)}$ in H_{Φ_1}. Again because n is prime, by Lemma 3.5 we have $\lambda_{n-1} = \lambda_1$ on $\mathcal{L}(\mathbf{B}) \cap H_{\Phi_1}$, so (finally) by Proposition 3.10 we get an algorithm for $\mathrm{SVP}_{n \cdot \mathrm{poly}(\log n)}$. $\square$

Corollary 5.7. *For any constant $\delta > 0$, there exist $D(n) = n^{\Theta(1)}$, $p(n) = n^{2.5+\Theta(1)}$, and $m(n) = \Theta(1)$ such that: $\mathcal{H}(\mathbb{Z}^n_{p(n)}, S_{D(n),\Phi_1(\alpha)}, m(n))$ is a hash function ensemble for which finding collisions for infinitely many* prime *n is at least as hard as solving* SVP_γ *with high probability in the worst case for infinitely many prime n within a factor $\gamma(n) = n^{1+\delta}$.*

Proof. We can choose $D(n) = \Theta(n^{\delta/2})$ and a large enough $m(n) = \Theta(1)$ so that $\frac{m(n)\log D(n)}{\log p(n)} > 1$. The chain of reductions is the same as in the proof of Corollary 5.6, yielding an SVP algorithm with approximation factor $n \cdot m(n) \cdot D(n) \cdot \mathrm{poly}(\log n) \leq n^{1+\delta}$. $\square$

6 Acknowledgements

We thank the anonymous reviewers for their helpful and thorough comments, and especially for a simplified proof of Lemma 2.16.

References

[1] M. Ajtai. Generating hard instances of lattice problems (extended abstract). In *Proc. 28th Annual ACM Symposium on Theory of Computing (STOC 1996)*, pages 99–108, 1996.

[2] M. Ajtai. The shortest vector problem in L_2 is NP-hard for randomized reductions (extended abstract). In *Proc. 30th Annual ACM Symposium on Theory of Computing (STOC 1998)*, pages 10–19, 1998.

[3] M. Ajtai and C. Dwork. A public-key cryptosystem with worst-case/average-case equivalence. In *Proc. 29th Annual ACM Symposium on Theory of Computing (STOC 1997)*, pages 284–293, 1997.

[4] S. Arora, L. Babai, J. Stern, and Z. Sweedyk. The hardness of approximate optima in lattices, codes, and systems of linear equations. *J. Computer and System Sciences*, 54(2):317–331, 1997.

[5] J.-Y. Cai and A. Nerurkar. Approximating the SVP to within a factor $(1+1/\dim^{\epsilon})$ is NP-hard under randomized reductions. *Jounal of Computer and System Sciences*, 59(2):221–239, 1999.

[6] J.-Y. Cai and A. P. Nerurkar. An improved worst-case to average-case connection for lattice problems. In *Proc. 38th Annual Symposium on Foundations of Computer Science (FOCS 1997)*, page 468, 1997.

[7] I. Dinur, G. Kindler, and S. Safra. Approximating-CVP to within almost-polynomial factors is NP-hard. In *Proc. 39th Annual Symposium on Foundations of Computer Science (FOCS 1998)*, pages 99–111. IEEE Computer Society, 1998.

[8] D. S. Dummit and R. M. Foote. *Abstract Algebra.* Prentice Hall, Upper Saddle River, NJ, USA, second edition, 1999.

[9] R. Genarro, Y. Gertner, J. Katz, and L. Trevisan. Bounds on the efficiency of generic cryptographic constructions. *SIAM J. Computing*, 35(1):217–246, 2005.

[10] R. Gennaro and L. Trevisan. Lower bounds on the efficiency of generic cryptographic constructions. In *Proc. 41st Annual Symposium on Foundations of Computer Science (FOCS 2000)*, page 305. IEEE Computer Society, 2000.

[11] O. Goldreich, S. Goldwasser, and S. Halevi. Collision-free hashing from lattice problems. Electronic Colloquium on Computational Complexity (ECCC) Report TR96-042, 1996.

[12] O. Goldreich, S. Goldwasser, and S. Halevi. Public-key cryptosystems from lattice reduction problems. In *Proc. 17th Annual Conference on Advances in Cryptology (CRYPTO 1997)*, pages 112–131. Springer-Verlag, 1997.

[13] S. Khot. Hardness of approximating the shortest vector problem in lattices. In *Proc. 45th Symposium on Foundations of Computer Science (FOCS 2004)*, pages 126–135. IEEE Computer Society, 2004.

[14] V. Lyubashevsky and D. Micciancio. Generalized compact knapsacks are collision resistant. Electronic Colloquium on Computational Complexity (ECCC) Report TR05-142, 2005.

[15] D. Micciancio. Generalized compact knapsaks, cyclic lattices, and efficient one-way functions from worst-case complexity assumptions. In *Proc. 43rd Annual Symposium on Foundations of Computer Science (FOCS 2002).*

[16] D. Micciancio. The shortest vector problem is NP-hard to approximate to within some constant. *SIAM J. Computing*, 30(6):2008–2035, Mar. 2001.

[17] D. Micciancio and S. Goldwasser. *Complexity of Lattice Problems: a cryptographic perspective*, volume 671 of *The Kluwer International Series in Engineering and Computer Science.* Kluwer Academic Publishers, Boston, Massachusetts, 2002.

[18] D. Micciancio and O. Regev. Worst-case to average-case reductions based on Gaussian measure. pages 371–381.

[19] M. Naor and M. Yung. Universal one-way hash functions and their cryptographic applications. In *Proc. 21st Annual ACM Symposium on Theory of Computing (STOC 1989)*, pages 33–43, 1989.

[20] O. Regev. New lattice-based cryptographic constructions. *J. ACM*, 51(6):899–942, 2004.

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

[22] J. Rompel. One-way functions are necessary and sufficient for secure signatures. In *Proc. 22nc Annual ACM Symposium on Theory of Computing (STOC 1990)*, pages 387–394, 1990.

[23] D. R. Simon. Finding collisions on a one-way street: Can secure hash functions be based on general assumptions? In *EUROCRYPT*, pages 334–345, 1998.

[24] P. van Emde Boas. Another NP-complete problem and the complexity of computing short vectors in a lattice. Technical Report 81-04, University of Amsterdam, 1981.

[25] X. Wang, Y. L. Yin, and H. Yu. Finding collisions in the full SHA-1. In *CRYPTO*, 2005.

[26] X. Wang and H. Yu. How to break MD5 and other hash functions. In *EUROCRYPT*, pages 19–35, 2005.