Pseudorandom Knapsacks and the Sample Complexity of LWE Search-to-Decision Reductions*

Daniele Micciancio and Petros Mol

Department of Computer Science & Engineering
University of California, San Diego
{daniele, pmol}@cs.ucsd.edu

Abstract. We study the pseudorandomness of bounded knapsack functions over arbitrary finite abelian groups. Previous works consider only specific families of finite abelian groups and 0-1 coefficients. The main technical contribution of our work is a new, general theorem that provides sufficient conditions under which pseudorandomness of bounded knapsack functions follows directly from their one-wayness. Our results generalize and substantially extend previous work of Impagliazzo and Naor (J. Cryptology 1996).
As an application of the new theorem, we give sample preserving search-to-decision reductions for the Learning With Errors (LWE) problem, introduced by (Regev, STOC 2005) and widely used in lattice-based cryptography. Concretely, we show that, for a wide range of parameters, m LWE samples can be proved indistinguishable from random just under the hypothesis that search LWE is a one-way function for the same number m of samples.

1 Introduction

The Learning With Errors (LWE) problem, introduced by Regev in [31], is the problem of recovering a secret n-dimensional integer vector $\mathbf{s} \in \mathbb{Z}_q^n$, given a collection of perturbed random equations $\mathbf{a}_i\mathbf{s} \approx b_i$ where $\mathbf{a}_i \in \mathbb{Z}_q^n$ is chosen uniformly at random and $b_i = \mathbf{a}_i\mathbf{s} + e_i$ for some small, randomly chosen error term e_i. In recent years, LWE has been used to substantially expand the scope of lattice based cryptography, yielding solutions to many important cryptographic tasks, including public key encryption secure against passive [31, 20, 29] and active attacks [30, 28], (hierarchical) identity based encryption [14, 10, 1, 2], digital signatures [14, 10], oblivious transfer protocols [29], several forms of leakage resilient encryption [5, 6, 11, 16], homomorphic encryption [13] and more.

* This research was supported in part by NSF under grants CNS-0831536 and CNS-0716790. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. This is an extended abstract of the work. For the full version, see the authors' webpage.

The versatility of the LWE problem in the construction of a plethora of cryptographic applications is due in large part to its pseudorandomness properties: as proved in [31], if recovering (with high[1] probability) the secret $\mathbf{s}$ from the samples $(\mathbf{a}_i, \mathbf{a}_i\mathbf{s} + e_i)$ is computationally hard, then it is also hard to distinguish the LWE samples $(\mathbf{a}_i, \mathbf{a}_i\mathbf{s}+e_i)$ from uniformly random ones $(\mathbf{a}_i, b_i)$ where the $b_i \in \mathbb{Z}_q$ are chosen uniformly and independently at random. In other words, any efficient distinguisher (between the LWE and uniform distributions) can be turned into an inverter that recovers the secret $\mathbf{s}$, with only a polynomial slow-down.

On the theoretical side, cryptography based on LWE is supported by deep worst-case/average-case connections [31, 28], showing that any algorithm that solves LWE (on the average) can be efficiently converted into a (quantum) algorithm to solve the (worst-case) hardest instances of several famous lattice approximation problems which are believed to be intractable, including approximating the minimum distance of a lattice within factors that grow polynomially in the dimension, and related problems [23]. It should be remarked that, while such proofs of security based on worst-case lattice assumptions provide a solid theoretical justification for the probability distributions used in LWE cryptography, they are hardly useful in practice: in order to get meaningful estimates on the hardness of breaking LWE cryptography, it is generally more useful and appropriate to conjecture the average-case hardness of solving LWE, and use that as a starting point. In fact, all recent work aimed at determining appropriate key sizes and security parameters [26, 22, 33] follows this approach, and investigates experimentally the concrete hardness of solving LWE on the average.

In light of that, LWE is best formulated as the problem of inverting the one-way function family (indexed by a random matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$, where m is the number of samples) that maps the secret $\mathbf{s}$ and error vector $\mathbf{e}$ to $\mathbf{A}\mathbf{x} + \mathbf{e}$. The search-to-decision reduction of [31] shows that if the LWE function family is one-way, then it is also a good pseudorandom generator. However, the reduction in [31] somehow hides a very important detail: the value of m for which the function is assumed to be one-way is much higher (still polynomially related) to the value of m for which the output of the function is pseudorandom. While theoretical results based on worst-case lattice problems are fairly insensitive to the value of m (i.e., the number of samples used in the LWE instance), this number becomes more important and relevant when considering concrete attacks on the average-case hardness of LWE.

For instance, recent algorithmic results [7], show that when the errors e_i are sufficiently small, the LWE problem can be solved in subexponential (or even polynomial) time, provided a sufficiently large (but still polynomial) number of samples is available. Therefore, for certain ranges of the parameters, the number of available samples can have a significant impact on the computational hardness of the LWE problem. Likewise, some lattice attacks perform better in practice when given many (typically $\omega(n)$) samples [26]. However, LWE-based encryption schemes (e.g., see [22]) typically expose only a small number of samples (say,

[1] Due to the self-reducibility properties of the LWE problem, here "high" can be interpreted in a variety of ways, ranging from "nonnegligible" to "very close to 1".

comparable to the dimension n of the LWE secret $\mathbf{s}$) during key generation and encryption. Fixing the number of available samples to a small value may significantly reduce the effectiveness of concrete attacks, and increase our confidence in the security of the schemes.

It should also be noted that when the number of available samples is above a certain threshold, one can efficiently generate an arbitrary number of additional samples [14, 6, 32], but at the cost of increasing the magnitude of the errors. So, for certain other ranges of the parameters the impact of increasing the number of samples may not be as critical as in [7]. Still, even in such situations, using a large number of samples comes at the price of lowering the quality of the samples, which can negatively impact the concrete security and performance of LWE-based cryptographic functions.

This motivates the following question: how big of a blow-up in the number of samples is required to prove the pseudorandomness of the LWE problem, based on the conjectured hardness of its search (secret recovery) version? The main result of this paper is that, perhaps surprisingly, in most common applications of LWE in cryptography, no such blow-up is necessary at all: there is a *sample preserving* reduction from solving the search LWE problem (with nonnegligible success probability) to the problem of distinguishing the LWE distribution from random (with nonnegligible advantage). At the core of our result is a general theorem about the pseudorandomness of the bounded knapsacks over arbitrary groups, that substantially extends previous results in the area and might be of independent interest.

Contributions and Applications. Let $(G, +)$ be a finite abelian group, and $\mathbf{g} = (g_1, \ldots, g_m) \in G^m$ a sequence of group elements chosen uniformly at random. The group elements $\mathbf{g}$ define a knapsack function $f_{\mathbf{g}}(\mathbf{x})$ that maps the vector $\mathbf{x} \in \mathbb{Z}^m$ to the group element $f_{\mathbf{g}}(\mathbf{x}) = \sum_i x_i g_i$. If the input $\mathbf{x}$ is restricted to vectors with small entries, then for a large variety of groups G, $f_{\mathbf{g}}$ is conjectured to be a one-way function family, i.e., a family of functions that are hard to invert on average when the key $\mathbf{g}$ is chosen uniformly at random. For example, when the input $\mathbf{x}$ is restricted to the set $\{0,1\}^m$ of binary vectors, inverting $f_{\mathbf{g}}$ is the famous subset-sum problem, which is conjectured to be hard to solve on average, and has been extensively studied in cryptography. In a classic paper [18], Impagliazzo and Naor showed that for some specific, but representative, choices of the group G, if the subset-sum function is one-way, then it is also a pseudorandom generator, i.e., it is computationally hard to distinguish $(\mathbf{g}, f_{\mathbf{g}}(\mathbf{x}))$ from a uniformly random element of G^{m+1}, when $\mathbf{g} \in G^m$ and $\mathbf{x} \in \{0,1\}^m$ are chosen uniformly at random. We generalize the results of [18] in two respects:

- We consider functions over arbitrary groups G. Only groups of the form $\mathbb{Z}_N$ were considered in [18], and for two specific (but representative) choices of N (prime and power of 2).
- We consider input coefficients x_i that take values from a set $\{0, \ldots, s\}$ (or $\{-s, \ldots, s\}$) for any (polynomially bounded) s. Moreover, we consider *arbi-*

trary input distributions. By contrast, the results in [18] hold for inputs $\mathbf{x}$ distributed *uniformly* with coefficients in $\{0, 1\}$.

Both extensions are essential for the sample-preserving search-to-decision LWE reduction presented in Section 4.2, which requires the pseudorandomness of the knapsack function over vector groups $G = \mathbb{Z}_q^k$, and for inputs $\mathbf{x}$ following a nonuniform (Gaussian) distribution over a sufficiently large set $\{-s, \ldots, s\}$. Our main technical result (Theorem 2) shows that for any group G and input distribution $\mathcal{X}$, the output of the knapsack function is pseudorandom provided the following two conditions hold:

1. $f_{\mathbf{g}}$ is a one-way function family with respect to input distribution $\mathcal{X}$, and
2. certain folded versions of $f_{\mathbf{g}}$ (where both the key $\mathbf{g}$ and the output $f_{\mathbf{g}}(\mathbf{x})$ are projected onto a quotient group $G_d = G/dG$ for some $d \in \mathbb{Z}$,) have pseudorandom output.

The second condition above may seem to make the statement in the theorem vacuous, as it asserts the pseudorandomness of $f_{\mathbf{g}}$ assuming the pseudorandomness of (certain other versions of) $f_{\mathbf{g}}$. The power of the theorem comes from the fact that the quotient groups G_d considered are very small, so small that in many important settings the output of the folded knapsack function $f_{\mathbf{g}}(\mathbf{x}) \bmod dG$ is *statistically* close to uniform. As a technical tool, we provide upper bounds on the statistical distance between the distribution $(\mathbf{g}, f_{\mathbf{g}}(\mathbf{x})) \bmod dG$ and the uniform distribution over G_d^{m+1} (Lemma 4). We use these bounds to show that for many interesting groups and input distributions, the output of the folded knapsack function is statistically close to uniform. Therefore, as a corollary to the main theorem, we get that one-wayness of the bounded knapsack function implies that knapsacks are good pseudorandom generators. Specific groups and input distributions for which this holds include among others:

- Groups whose order contains only large prime factors, larger than the maximum value of the input coefficients. Cyclic groups with prime order and vector groups $\mathbb{Z}_p^k$ for prime p fall into this category. This result generalizes those in [18] from uniform binary input to arbitrary input distributions.
- Distributions that, when folded (modulo small divisors of the order of G,) maintain high entropy relative to the size of the quotient group G/dG. (See Theorem 6.) Groups of the form $G = \mathbb{Z}_{2^\ell}^k$ and uniform input distribution over $\mathbb{Z}_{2^i}^m$ for some $i < \ell$ satisfy this requirement. This parameter set is a very attractive choice in practice since both group operations and input sampling are particularly efficient and easy to implement.

Using the duality between LWE and the knapsack problem [35, 25], we obtain sample preserving search-to-decision reductions for LWE for several interesting choices of the modulus q and input distribution, which include (among others):

- $q = 2$ and *any* error distribution. This directly proves the pseudorandomness of the well-known Learning Parity with Noise (LPN) problem, as already established in [9, 19], but in a sample-preserving manner.

– prime q and *any polynomially bounded* error distribution.
– power-of-prime modulus $q = p^e$ for p large enough so that the error distribution is concentrated over $\{-(p-1)/2, \ldots, (p-1)/2\}$.
– $q = p^e$ for small prime p and uniform error distribution over $\mathbb{Z}_{p^i}$ $(i < e)$.

These results subsume (see below) several previous pseudorandomness results for LWE [31, 6] and LPN [19] but with an important difference. While the proofs in [31, 6, 19] require that LWE (resp. LPN) is hard to solve for a very large number of samples, our reductions are *sample preserving*: the pseudorandomness of LWE (resp. LPN) holds, provided the same problem is one-way for the *same* number of samples. We remark that previous results are often phrased as reductions from solving the LWE search problem *with high probability*, to solving the LWE decision problem *with nonnegligible* advantage, combining the search-to-decision reduction and success probability amplification into a single statement. By contrast, our reduction shows how to solve the LWE search problem with *nonnegligible probability*. Our results subsume previous work in the sense that the LWE search problem can be solved with high probability by first invoking our reduction, and then amplifying the success probability using standard repetition techniques. Of course, any such success probability amplification would naturally carry the cost of a higher sample complexity. We remark that a close inspection of worst-case to average-case reductions for LWE [31, 28] shows that these reductions directly support the conjecture that LWE is a *strong* one-way function. As already discussed, worst-case to average-case reductions do not provide quantitatively interesting results, and are best used as qualitative arguments to support the conjecture that certain problems are computationally hard on average. Under the standard conjecture that search LWE is a strong one-way function, the results in this paper offer a fairly tight, and sample preserving proof that LWE is also a good pseudorandom generator, which can be efficiently used for the construction of many other lattice based public key cryptographic primitives. By contrast, it is not known how to take advantage of the strong one-wayness of LWE within previous search-to-decision reductions, resulting in a major degradation of the parameters. Of course, if we change the complexity assumption, and as a starting point we use the worst-case hardness of lattice problems or the assumption that LWE is only a *weak* one-way function, then our reduction will also necessarily incur a large blow up in sample complexity through amplification, and lead to quantitatively uninteresting results.

2 Preliminaries

We use $\mathbb{N}, \mathbb{C}, \mathbb{T}$ for the sets of natural, complex and complex numbers of unit magnitude respectively. We use lower case for scalars, upper case for sets, bold lower case for vectors and bold upper case for matrices. We use calligraphic letters for probability distributions and (possibly randomized) algorithms. For $s \in \mathbb{N}$, the set of the first s nonnegative integers is denoted $[s] = \{0, 1, \ldots, s-1\}$.

2.1 Probability

We write $x \leftarrow \mathcal{X}$ for the operation of selecting x according to a probability distribution $\mathcal{X}$ or by running probabilistic algorithm $\mathcal{X}$. We use $\{(x, x') \mid x \leftarrow \mathcal{X}, x' \leftarrow \mathcal{X}\}$ to denote the probability distribution obtained by drawing two samples from $\mathcal{X}$ independently at random. For any probability distribution $\mathcal{X}$ over set X and any value $x \in X$, $\Pr\{x \leftarrow \mathcal{X}\}$ is the probability associated to x by distribution $\mathcal{X}$. The uniform distribution over a set A is denoted $\mathcal{U}(A)$, and the support of a distribution $\mathcal{X}$ is denoted $[\mathcal{X}] = \{x \in X \mid \Pr\{x \leftarrow \mathcal{X}\} > 0\}$. The *collision probability* of $\mathcal{X}$ is the probability $\mathsf{Col}(\mathcal{X}) = \Pr\{x = x' \mid x \leftarrow \mathcal{X}, x' \leftarrow \mathcal{X}\} = \sum_{x \in [\mathcal{X}]} \Pr\{x \leftarrow \mathcal{X}\}^2$ that two independent identically distributed samples from $\mathcal{X}$ take the same value. The *mode* of $\mathcal{X}$ is the probability of the most likely value, i.e. $\mathsf{mode}(\mathcal{X}) = \max_{x \in X} \Pr\{x \leftarrow \mathcal{X}\}$. It is easy to see that $\mathsf{Col}(\mathcal{X}) \leq \mathsf{mode}(\mathcal{X})$.

The *statistical distance* $\Delta(\mathcal{X}, \mathcal{Y})$ between distributions $\mathcal{X}$ and $\mathcal{Y}$, defined over the same set X, is the quantity $\frac{1}{2} \sum_{x \in X} |\Pr\{x \leftarrow \mathcal{X}\} - \Pr\{x \leftarrow \mathcal{Y}\}|$. The statistical distance satisfies $\Delta(f(\mathcal{X}), f(\mathcal{Y})) \leq \Delta(\mathcal{X}, \mathcal{Y})$ for any (possibly probabilistic) function f. Two distributions $\mathcal{X}, \mathcal{Y}$ are ϵ-*close* if $\Delta(\mathcal{X}, \mathcal{Y}) \leq \epsilon$. They are (t, ϵ)-*indistinguishable* if $\Delta(\mathcal{D}(\mathcal{X}), \mathcal{D}(\mathcal{Y})) \leq \epsilon$ for *any* probabilistic predicate $\mathcal{D}\colon X \to \{0, 1\}$ (called the *distinguisher*) computable in time at most t. Otherwise, we say that $\mathcal{X}, \mathcal{Y}$ are (t, ϵ)-*distinguishable*. When $\mathcal{Y} = \mathcal{U}(X)$ is the uniform distribution, we use $\Delta_U(\mathcal{X}) = \Delta(\mathcal{X}, \mathcal{U}(X))$ as an abbreviation and say that $\mathcal{X}$ is ϵ-*random* (resp. (t, ϵ)-*pseudorandom*) if it is ϵ-close (resp. (t, ϵ)-close) to $\mathcal{U}(X)$.

Function families. A function family $(F, \mathcal{X})$ is a collection $F = \{f_i\colon X \to R\}_{i \in I}$ of functions indexed by $i \in I$ with common domain X and range R, together with a probability distribution $\mathcal{X}$ over the input set $X \supseteq [\mathcal{X}]$. For simplicity, in this paper we always assume that the set of functions is endowed with the *uniform* probability distribution $\mathcal{U}(F)$. Each function family $(F, \mathcal{X})$ naturally defines a probability distribution

$$\mathcal{F}(F, \mathcal{X}) = \{(f, f(x)) \mid f \leftarrow \mathcal{U}(F), x \leftarrow \mathcal{X}\} \tag{1}$$

obtained by selecting a function at random and evaluating it at a random input.

A function family $\mathcal{F} = (F, \mathcal{X})$ is called (t, ϵ)-*one-way* if there is no (probabilistic) algorithm $\mathcal{I}$ running in time at most t such that $\Pr\{f(x) = y \mid (f, y) \leftarrow \mathcal{F}(F, \mathcal{X}), x \leftarrow \mathcal{I}(f, y)\} \geq \epsilon$. In this paper it is convenient to use the related notion of "uninvertible function". A (t, ϵ)-*inverter* for a function family $(F, \mathcal{X})$ is a (probabilistic) algorithm $\mathcal{I}$ running in time at most t such that $\Pr\{x = y \mid f \leftarrow \mathcal{U}(F), x \leftarrow \mathcal{X}, y \leftarrow \mathcal{I}(f, f(x))\} \geq \epsilon$. If there exists a (t, ϵ)-inverter for a function family $(F, \mathcal{X})$, then we say that $(F, \mathcal{X})$ is (t, ϵ)-invertible. A function family such that there is no (t, ϵ)-inverter is called (t, ϵ)-*uninvertible*. In this paper, we deal with function families that are (almost) injective, i.e. with overwhelming probability over $f \leftarrow \mathcal{U}(F)$ and $x \leftarrow \mathcal{X}$, there exists no $x' \neq x$ such that $f(x) = f(x')$. When this is the case, then one-wayness and uninvertibility

are equivalent notions. A (t, ϵ)-*pseudorandom generator family* is a function family $(F, \mathcal{X})$ such that the associated distribution (1) is (t, ϵ)-pseudorandom, i.e., it is (t, ϵ)-indistinguishable from the uniform distribution $\mathcal{U}(F \times R)$.

Asymptotics. We use n as a (security) parameter that controls all other parameters. Unless otherwise stated, any other parameter (say m) will be polynomially related to n. We use standard asymptotic notation $O(\cdot), \Omega(\cdot), o(\cdot), \omega(\cdot)$, etc. We write $\mathsf{negl}(n)$ for the set of negligible functions and $\mathsf{poly}(n)$ for the set of polynomially bounded functions. In the asymptotic computational complexity setting, one often considers probability ensembles, i.e., sequences $\mathcal{X} = (\mathcal{X}_n)_{n\in\mathbb{N}}$ of probability distributions over possibly different sets $X_n \supseteq [\mathcal{X}_n]$. Two distributions ensembles $\mathcal{X} = (\mathcal{X}_n)_{n\in\mathbb{N}}$ and $\mathcal{Y} = (\mathcal{Y}_n)_{n\in\mathbb{N}}$ are *statistically close* (denoted $\mathcal{X} \overset{s}{\approx} \mathcal{Y}$) if $\mathcal{X}_n$ and $\mathcal{Y}_n$ are $\epsilon(n)$-close for some negligible function $\epsilon(n) = \mathsf{negl}(n)$. The ensembles $\mathcal{X}$ and $\mathcal{Y}$ are *computationally indistinguishable* (denoted $\mathcal{X} \overset{c}{\approx} \mathcal{Y}$) if $\mathcal{X}_n$ and $\mathcal{Y}_n$ are $(t(n), \epsilon(n))$-indistinguishable for $\epsilon(n) = \mathsf{negl}(n)$ and any $t(n) = \mathsf{poly}(n)$ under a uniform sequence $(\mathcal{D}_n \colon X_n \to \{0,1\})_{n\in\mathbb{N}}$ of distinguishers. Definitions for function families are also extended in the obvious way to function family ensembles $\mathcal{F} = (\mathcal{F}_n)_n$ in the asymptotic setting by taking $\epsilon(n) = \mathsf{negl}(n)$ and $t(n) = \mathsf{poly}(n)$, and considering uniform sequences of distinguishing algorithms. In particular, a function family ensemble $\mathcal{F} = (\mathcal{F}_n)_n$ is *one-way* if $\mathcal{F}_n$ is $(t(n), \epsilon(n))$-one-way for $\epsilon(n) = \mathsf{negl}(n)$ and any $t(n) = \mathsf{poly}(n)$. It is *pseudorandom* if the associated (asymptotic) distribution (1) is $(t(n), \epsilon(n))$-pseudorandom, i.e., it is $(t(n), \epsilon(n))$-indistinguishable from the uniform distribution $\mathcal{U}(F_n \times R_n)$.

Discrete Gaussian Distributions. Gaussian-like distributions play a central role in the Learning With Errors (LWE) problem. For each sample $(\mathbf{a}, b = \mathbf{a} \cdot \mathbf{s} + e)$, the distribution χ from which e is drawn, is a normal distribution over the integers. Below, we focus mainly on the *discrete* Gaussian distribution and provide bounds on its collision probability. Those bounds are used in establishing the search-to-decision reduction for LWE. Similar bounds can be established for the *discretized* Gaussian (defined in [31]).

The *Gaussian function* on $\mathbb{R}^m$ with parameter r and center $\mathbf{c}$ is defined as $\forall \mathbf{x} \in \mathbb{R}^m,\ \rho_{r,\mathbf{c}}(\mathbf{x}) = exp(-\pi\|\mathbf{x} - \mathbf{c}\|^2/r^2)$. The *discrete Gaussian distribution* over a countable set S is defined as

$$\forall \mathbf{x} \in S,\ \mathcal{D}_{S,r,\mathbf{c}} = \frac{\rho_{r,\mathbf{c}}(\mathbf{x})}{\sum_{\mathbf{y}\in S} \rho_{r,\mathbf{c}}(\mathbf{y})}$$

Here, we are interested in vectors $\mathbf{x} = (x_1, \ldots, x_m) \in \mathbb{Z}^m$ distributed according to $\mathcal{D}_{\mathbb{Z}^m,r}$ ($\mathbf{c} = \mathbf{0}$). In that case, each coordinate x_i of $\mathbf{x}$ is identically and independently distributed according to the 1-dimensional Gaussian $\mathcal{D}_{\mathbb{Z},r}$. For our search-to-decision reduction of LWE with discrete Gaussian error distribution, we need to consider the folded (1-dimensional) distribution $\mathcal{D}_{\mathbb{Z},r} \bmod d$. The following lemma bounds the collision probability of this distribution.

Lemma 1. *For any $r > 0$ and $d \in \mathbb{Z}$, we have $\mathsf{Col}(\mathcal{D}_{\mathbb{Z},r} \bmod d) \leq \frac{1}{r} + \frac{1}{d}$. Furthermore, if $r = d \cdot \omega(\sqrt{\log n})$, then $\mathsf{Col}(\mathcal{D}_{\mathbb{Z},r} \bmod d) \leq \frac{1}{d} + \mathsf{negl}(n)$.*

2.2 Groups and Knapsack function families.

In this work, by group we always mean *finite abelian group*. We use additive notation for groups; 0_G is the *identity element*, $|G|$ is the *order* (size) of G and M_G its *exponent*, i.e. the smallest non-zero integer e such that $e \cdot g = 0_G$ for all $g \in G$. We use the dot product notation $\mathbf{x} \cdot \mathbf{y} = \sum_i x_i \cdot y_i$ both for the inner product of two vectors $\mathbf{x}, \mathbf{y} \in R^n$ with elements in a ring R, and also to take integer linear combinations $\mathbf{x} \in \mathbb{Z}^n$ of a vector $\mathbf{y} \in G^n$ with elements in an additive group. For $\mathbf{x} = (x_1, \ldots, x_n) \in R^n$ and $a \in R$, we also define $\mathbf{x} \cdot a = (x_1 \cdot a, \ldots, x_n \cdot a)$.

For any group G and (positive) integer d, we use G_d to denote the quotient group G/dG where dG is the subgroup $\{d \cdot g \,\big|\, g \in G\}$, in analogy with the usual notation $\mathbb{Z}_d = \mathbb{Z}/d\mathbb{Z}$ for the group of integers modulo d. Likewise, for an element $g \in G$, we use $g \bmod dG$ (or just $g \bmod d$) for the image of g under the natural homomorphism from G to G_d. For any integer vector $\mathbf{w} = (w_1, \ldots, w_r) \in \mathbb{Z}^r$, we write $\gcd_G(\mathbf{w}) = \gcd(w_1, \ldots, w_r, M_G)$ for the greatest common divisor of the elements of $\mathbf{w}$ and the group exponent.

Lemma 2. *For any group G and integer vector $\mathbf{w} \in \mathbb{Z}^r$, $\{\mathbf{w} \cdot \mathbf{g} \mid \mathbf{g} \leftarrow \mathcal{U}(G^r)\} = \mathcal{U}(\gcd_G(\mathbf{w}) \cdot G)$. In particular, $\Pr\left[\, \mathbf{w} \cdot \mathbf{g} = 0_G \,\middle|\, \mathbf{g} \leftarrow \mathcal{U}(G^r) \,\right] = \frac{1}{|\gcd_G(\mathbf{w}) \cdot G|}$.*

Knapsack Families. For any group G and input distribution $\mathcal{X}$ over $\mathbb{Z}^m$, the knapsack family $\mathcal{K}(G, \mathcal{X})$ is the function family with input distribution $\mathcal{X}$ and set of functions $f_{\mathbf{g}} \colon [\mathcal{X}] \to G$ indexed by $\mathbf{g} \in G^m$ and defined as $f_{\mathbf{g}}(\mathbf{x}) = \mathbf{g} \cdot \mathbf{x} \in G$. We will often use $\mathbf{g}$ instead of $f_{\mathbf{g}}$ to describe a member function drawn from $\mathcal{K}(G, \mathcal{X})$. When $G, \mathcal{X}$ are clear from the context we will simply write $\mathcal{K}$. We often consider folded knapsack families $\mathcal{K}(G_d, \mathcal{X})$ over quotient groups G_d. For brevity, when G and $\mathcal{X}$ are clear from the context, we will write $\mathcal{K}_d$ instead of $\mathcal{K}(G_d, \mathcal{X})$. The following lemma shows that the distribution $\mathcal{F}(\mathcal{K}_d)$ associated to a folded knapsack function family is closely related to the distribution

$$\mathcal{F}_d(\mathcal{K}) = \{(\mathbf{g}, g + h) \mid (\mathbf{g}, g) \leftarrow \mathcal{F}(\mathcal{K}), h \leftarrow \mathcal{U}(d \cdot G)\}. \tag{2}$$

Lemma 3. *For any knapsack family $\mathcal{K}$ and $d \in \mathbb{Z}$, $\Delta_U(\mathcal{F}_d(\mathcal{K})) = \Delta_U(\mathcal{F}(\mathcal{K}_d))$. Also, $\mathcal{F}_d(\mathcal{K})$ is pseudorandom if and only if $\mathcal{F}(\mathcal{K}_d)$ is pseudorandom.*

For a group H such that $\mathcal{K}(H, \mathcal{X})$ compresses its input, Lemma 4 provides an upper bound on the statistical distance between $\mathcal{F}(\mathcal{K}(H, \mathcal{X}))$ and $\mathcal{U}(H^m \times H)$ by generalizing the Leftover Hash Lemma [17] to (non-necessarily universal) knapsack function families $\mathcal{K}(H, \mathcal{X})$ over arbitrary groups.

Lemma 4 (LHL, generalized). *For any finite abelian group H and integer d,*

$$\Delta_U(\mathcal{F}(\mathcal{K}(H, \mathcal{X}))) \leq \frac{1}{2} \sqrt{\sum_{1 < d \mid M} |H_d| \cdot \Pr\{\gcd_H(\mathbf{x} - \mathbf{y}) = d \mid \mathbf{x} \leftarrow \mathcal{X}, \mathbf{y} \leftarrow \mathcal{X}\}} \tag{3}$$

where M is the exponent of H and $d > 1$ ranges over all divisors of M.

2.3 Fourier Analysis and Learning

Fourier analysis has been used extensively in learning theory, especially in the context of learning functions defined over the boolean hypercube (see [21, 8, 27] for some examples). In Cryptography, two noteworthy examples are the Kushilevitz-Mansour [21] formulation of the proof of the Goldreich-Levin [15] hard-core predicate for any one-way function and the proofs of hard-core predicates for several number-theoretic one-way functions by Akavia, Goldwasser and Safra [4].

Below we review some basic facts from Fourier analysis focusing on the discrete Fourier transform over finite abelian groups. We restrict the presentation to what is needed and refer the interested reader to [3, 34] for more details.

Fourier Basics. Let H be a finite abelian group and $h_1, h_2 : H \to \mathbb{C}$ be functions from H to the complex numbers. The *inner product* of h_1 and h_2 is defined as

$$\langle h_1, h_2 \rangle = \mathop{\mathbb{E}}_{x \leftarrow \mathcal{U}(H)} \left[h_1(x)\overline{h_2(x)} \right] = \frac{1}{|H|} \sum_{x \in H} h_1(x)\overline{h_2(x)}$$

where $\bar{z}$ is the conjugate of $z \in \mathbb{C}$. The ℓ_2-norm and ℓ_∞-norm of h are defined as

$$\|h\|_2 = \sqrt{\langle h, h \rangle} \quad \text{and} \quad \|h\|_\infty = \max_{x \in H} |h(x)|.$$

The set of *characters* of H (denoted as $char(H)$) is the set of all the *homomorphisms* from H to the complex numbers of unit magnitude $\mathbb{T}$. Namely,

$$char(H) = \{\chi : H \to \mathbb{T} \mid \forall x, y \in H,\ \chi(x + y) = \chi(x) \cdot \chi(y)\}$$

When H is a *vector group*, i.e. $H \simeq \mathbb{Z}_k^\ell$, and $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_\ell) \in H$, then the character $\chi_{\boldsymbol{\alpha}} \colon H \to \mathbb{T}$ is defined as $\chi_{\boldsymbol{\alpha}}(\mathbf{x}) = (\omega_k)^{\sum_{i=1}^{\ell} \alpha_i x_i} = \omega_k^{\mathbf{x} \cdot \boldsymbol{\alpha}}$.

FOURIER TRANSFORM. The *Fourier transform* of a function $h \colon H \to \mathbb{C}$ is the function $\widehat{h} \colon H \to \mathbb{C}$ defined as $\widehat{h}(\boldsymbol{\alpha}) = \langle h, \chi_{\boldsymbol{\alpha}} \rangle$. The Fourier transform measures the correlation of h with the characters in H.
The *energy* of a Fourier coefficient $\boldsymbol{\alpha}$ is defined as the square of its norm ($|\widehat{h}(\boldsymbol{\alpha})|^2$) while the *total energy* of h is defined as $\sum_{\boldsymbol{\alpha} \in H} |\widehat{h}(\boldsymbol{\alpha})|^2$. Parseval's identity says that $\sum_{\boldsymbol{\alpha} \in H} |\widehat{h}(\boldsymbol{\alpha})|^2 = \|h\|_2^2$.

Learning Heavy Fourier Coefficients. Let $\tau \in \mathbb{R}$, $\boldsymbol{\alpha} \in H$ and $h \colon H \to \mathbb{C}$ where H is a finite abelian group. Following the notation and terminology from [3], we say that $\boldsymbol{\alpha}$ is a τ-significant (or τ-heavy) Fourier coefficient of h if $|\widehat{h}(\boldsymbol{\alpha})|^2 \geq \tau$. The set of τ-significant Fourier coefficients of h is denoted by $Heavy_\tau(h)$, that is $Heavy_\tau(h) = \{\boldsymbol{\alpha} \in H \mid |\widehat{h}(\boldsymbol{\alpha})|^2 \geq \tau\}$. The following Theorem provides the conditions for learning heavy Fourier coefficients of functions defined over arbitrary finite groups and will be used in the proof of our main result.

Theorem 1. *(Significant Fourier Transform,[3, Theorem 3.3]) There exists a probabilistic algorithm ($\mathcal{SFT}$) that on input a threshold τ and given query access to a function $h\colon H \to \mathbb{C}$, returns all τ-heavy Fourier coefficients of h in time $\mathsf{poly}(\log|H|, 1/\tau, \|h\|_\infty)$ with probability[2] at least $2/3$.*

3 Pseudorandomness of Knapsack Functions

In this section we establish the connection between the search and decision problems for families of bounded knapsack functions. The following theorem summarizes our main result.

Theorem 2 (Main). *Let $\mathcal{X}$ be a distribution over $[s]^m \subset \mathbb{Z}^m$ for some $s = \mathsf{poly}(n)$ and G be a finite abelian group. If $\mathcal{K}(G, \mathcal{X})$ is one-way and $\mathcal{K}(G_d, \mathcal{X})$ is pseudorandom for all $d < s$, then $\mathcal{K}(G, \mathcal{X})$ is pseudorandom.*

The proof of Theorem 2 makes use of the intermediate notion of (un)predictability defined below. Informally, for any $\ell \in \mathbb{N}$, a ℓ-predictor for a function family $(F, \mathcal{X})$ is a weak form of inverter algorithm that on input a function $f \in F$, a target value $f(\mathbf{x})$ and a query vector $\mathbf{r} \in \mathbb{Z}_\ell^m$, attempts to recover the value of $\mathbf{x} \cdot \mathbf{r} \pmod{\ell}$, rather than producing the entire input $\mathbf{x}$.

Definition 1. *For any $\ell \in \mathbb{N}$ and function family $(F, \mathcal{X})$ with domain $[\mathcal{X}] \subseteq \mathbb{Z}^m$, a* ℓ-predictor *for $(F, \mathcal{X})$ is a probabilistic algorithm $\mathcal{P}$ that on input $(f, y, \mathbf{r}) \in F \times R \times \mathbb{Z}_\ell^m$ outputs a value $\mathcal{P}(f, y, \mathbf{r}) \in \mathbb{Z}_\ell$ which is intended to be a guess for $\mathbf{x} \cdot \mathbf{r} \pmod{\ell}$. The error distribution of a predictor $\mathcal{P}$ is defined as*

$$\mathcal{E}_\ell(\mathcal{P}) = \{\mathbf{x} \cdot \mathbf{r} - \mathcal{P}(f, f(\mathbf{x}), \mathbf{r}) \pmod{\ell} \mid f \leftarrow \mathcal{U}(F), \mathbf{x} \leftarrow \mathcal{X}, \mathbf{r} \leftarrow \mathcal{U}(\mathbb{Z}_\ell^m)\}.$$

The bias *of a ℓ-predictor $\mathcal{P}$ is the quantity $\left|\sum_{k \in \mathbb{Z}_\ell} \Pr\{k \leftarrow \mathcal{E}_\ell(\mathcal{P})\} \cdot \omega_\ell^{-k}\right|$. If $\mathcal{P}$ runs in time t and has bias* at least *ϵ, we say that $\mathcal{P}$ is (t, ϵ)-biased. A function family $(F, \mathcal{X})$ that admits a (t, ϵ)-biased ℓ-predictor is* (t, ϵ, ℓ)-predictable*.*

The proof of Theorem 2 proceeds in two steps. In the first step (Lemma 5) we show that a certain (non-trivial) predictor for $\mathcal{K}$ implies a non-trivial inverter for $\mathcal{K}$. This step uses Fourier analysis and holds true for *any* function family (and not just for $\mathcal{K}$) with domain $[\mathcal{X}] \subseteq \mathbb{Z}^m$. In the second step (Proposition 2), we prove that if there exists a distinguisher for $\mathcal{K}(G, \mathcal{X})$, but no distinguisher for $\mathcal{K}(G_d, \mathcal{X})$ for small d, then there exists a predictor for $\mathcal{K}(G, \mathcal{X})$. This step is specific to knapsack families and depends on both the underlying group G and the distribution $\mathcal{X}$. The two steps combined yield Theorem 2. Sections 3.1 and 3.2 are devoted to each step of the reduction.

[2] The success probability is taken over the internal randomness of the $\mathcal{SFT}$ algorithm only, and can be amplified using standard repetition techniques. However, this is not needed in our context, so for simplicity we fix the success probability to $2/3$.

3.1 From Predictability to Invertibility.

Proving that predictability implies invertibility is not specific to knapsack families. Rather, it holds for any function family $(F, \mathcal{X})$ with $F\colon X \to G$ where $X \subseteq \mathbb{Z}^m$ and G is a finite abelian group. Lemma 5 provides the conditions under which predictability implies invertibility.

Lemma 5. *Let $(F, \mathcal{X})$ be a function family with $[\mathcal{X}] \subseteq [s]^m \subset \mathbb{Z}^m$ for some $s = \mathsf{poly}(n)$. If $(F, \mathcal{X})$ is (t, ϵ, ℓ)-predictable for some $\ell > s$, then $(F, \mathcal{X})$ is $(\mathsf{poly}(n, \log \ell, 1/\epsilon) \cdot t, \frac{\epsilon}{3})$-invertible.*

Proof (Sketch). We use Fourier analysis and the $\mathcal{SFT}$ algorithm from Theorem 1. Let $\mathcal{P}$ be a ℓ-predictor for $\mathcal{F}$ that runs in time t. Roughly speaking, the inverter $\mathcal{I}$ on input $(f, f(\mathbf{x}))$ for some $f \leftarrow \mathcal{U}(F)$, simulates the execution of $\mathcal{SFT}$ in order to find $\mathbf{x}$. For every query $\mathbf{r} \in \mathbb{Z}_\ell^m$ issued by $\mathcal{SFT}$, $\mathcal{I}$ invokes $\mathcal{P}$ on appropriate input and sends back the result to $\mathcal{SFT}$. It turns out that, if the predictor $\mathcal{P}$ is (t, ϵ)-biased for some "sufficiently large" bias ϵ, then $\mathcal{I}$ simulates to $\mathcal{SFT}$ a (deterministic) function $h : \mathbb{Z}_\ell^m \to \mathbb{C}$ which is *highly correlated* with the character $\chi_{\mathbf{x}}(\cdot)$, that is, the function h $\mathcal{SFT}$ is given access to (through $\mathcal{I}, \mathcal{P}$) is such that $\widehat{h}(\mathbf{x})$ is "sufficiently heavy"[3] and therefore $\mathcal{SFT}$ will include $\mathbf{x}$ in the list it returns.

3.2 From Distinguishability to Predictability.

We now proceed into proving that, under certain conditions, a distinguisher $\mathcal{D}$ for $\mathcal{K}(G, \mathcal{X})$ with noticeable distinguishing advantage implies a predictor for $\mathcal{K}(G, \mathcal{X})$ with noticeable bias. At a high level, the predictor works as follows: on input a modulus ℓ, function $\mathbf{g} \leftarrow \mathcal{U}(G^m)$, $y = \mathbf{g} \cdot \mathbf{x} \in G$ and $\mathbf{r} \in \mathbb{Z}_\ell^m$, it first makes a guess for the inner product $\mathbf{x} \cdot \mathbf{r} \bmod \ell$; it then uses that guess to modify the knapsack instance $(\mathbf{g}, y)$, and finally invokes the distinguisher $\mathcal{D}$ on the modified instance $(\mathbf{g}', y')$. To conclude, the output of $\mathcal{D}$ is used to determine whether the initial guess was correct or not. The same technique has been used by Impagliazzo and Naor in [18]. However, in the setting considered in [18] – subset-sum over a cyclic group of prime order[4] – the reduction is rather straightforward: if the guess for $\mathbf{x} \cdot \mathbf{r}$ is correct, then the modified knapsack instance $(\mathbf{g}', y')$ is distributed according to $\mathcal{F}(\mathcal{K}(G, \mathcal{X}))$, whereas if the guess is wrong, the distribution of $(\mathbf{g}', y')$ is (statistically close to) uniform. Therefore, a distinguisher with noticeable advantage implies almost immediately a 2-predictor with noticeable bias.

When considering general (not necessarily cyclic) abelian groups with possibly composite order and distributions $\mathcal{X}$ with $[\mathcal{X}] \not\subseteq \{0, 1\}^m$, several technical difficulties arise. Unlike [18], if the guess for $\mathbf{x} \cdot \mathbf{r}$ is wrong, then the distribution of

[3] In the context of polynomial time reductions "sufficiently high" and "sufficiently heavy" is to be interpreted as noticeable in the security parameter.

[4] [18] also consider cyclic groups with power-of-2 order but this makes their analysis only slightly more complicated.

$(\mathbf{g}', y')$ can be statistically far from uniform. In fact, $(\mathbf{g}', y')$ can be distributed according to $\mathcal{F}_d(\mathcal{K}(G, \mathcal{X}))$ for any divisor d of the group exponent M_G. Depending on the order and structure of the underlying group, and the output distribution of the distinguisher $\mathcal{D}$ on the various auxiliary distributions $\mathcal{F}_d(\mathcal{K}(G, \mathcal{X}))$, the technical details of the reduction differ significantly. As a warm-up, we first state a weak form of our main Theorem.

Proposition 1. *If $\mathcal{K}(G, \mathcal{X})$ is (t, δ)-distinguishable from uniform for some noticeable δ, but $\mathcal{K}(G_d, \mathcal{X})$ is pseudorandom for all $d \leq 2ms^2$ then there is a $\mathsf{poly}(n)$-bounded prime $p \geq s$ and a polynomial[5] $q(\cdot)$ such that $\mathcal{K}(G, \mathcal{X})$ is $(O(t + m), 1/q(n), p)$-predictable.*

Even though Proposition 1 already gives search-to-decision reductions for some interesting families $\mathcal{K}$, it is not strong enough to establish Theorem 2 in its full generality. This is achieved in Proposition 2. Theorem 2 then follows directly if we combine Proposition 2 and Lemma 5.

Proposition 2. *If $\mathcal{K}(G, \mathcal{X})$ is (t, δ)-distinguishable from random for some noticeable δ, but $\mathcal{K}(G_d, \mathcal{X})$ is pseudorandom for all $d < s$, then there exists a polynomially bounded $d^* \geq s$ and polynomial $q(\cdot)$ such that $\mathcal{K}$ is $(O(t+m), 1/q(n), d^*)$-predictable.*

Proof. For simplicity, we write $\mathcal{K}$ (resp. $\mathcal{K}_d$) instead of $\mathcal{K}(G, \mathcal{X})$ (resp. $\mathcal{K}(G_d, \mathcal{X})$). We use $\mathcal{F}_d(\mathcal{K})$ (as in (2)) for all auxiliary distributions. For a distinguisher $\mathcal{D}$, $\mathsf{prob}_d^{\mathcal{D}} := \Pr[\, \mathcal{D}(\mathcal{F}_d(\mathcal{K})) = 1 \,]$. Notice that $\Pr[\, \mathcal{D}(\mathcal{U}(G^m \times G)) = 1 \,] = \mathsf{prob}_1^{\mathcal{D}}$ and $\Pr[\, \mathcal{D}(\mathcal{F}(\mathcal{K})) = 1 \,] = \mathsf{prob}_{M_G}^{\mathcal{D}}$. The *distinguishing advantage* of $\mathcal{D}$ between distributions $\mathcal{F}_{d_1}(\mathcal{K})$ and $\mathcal{F}_{d_2}(\mathcal{K})$ is defined as $\mathsf{Adv}^{\mathcal{D}}(\mathcal{F}_{d_1}(\mathcal{K}), \mathcal{F}_{d_2}(\mathcal{K})) = |\mathsf{prob}_{d_1}^{\mathcal{D}} - \mathsf{prob}_{d_2}^{\mathcal{D}}|$. When one of the two distribution is $\mathcal{U}(G^m \times G)$, we write $\mathsf{Adv}_d^{\mathcal{D}}$ instead of $\mathsf{Adv}^{\mathcal{D}}(\mathcal{F}_d(\mathcal{K}), \mathcal{F}_1(\mathcal{K}))$. We often write $a \equiv_c b$ instead of $a \equiv b \pmod{c}$ and define $\delta_{ij} = 1$ if $i = j$ and 0 otherwise.

By hypothesis, there exists distinguisher $\mathcal{D}$ and polynomial $t(\cdot)$ such that $|\mathsf{Adv}_{M_G}^{\mathcal{D}}| \geq \frac{1}{t(n)}$ and $|\mathsf{Adv}_{d'}^{\mathcal{D}}| = \mathsf{negl}(n)\ \forall d' < s$. If $\mathsf{Adv}_{d'}^{\mathcal{D}} = \mathsf{negl}(n)\ \forall d' < 2ms^2$ then proof follows directly from Proposition 1. Else, there exists d with $s \leq d < 2ms^2$ (notice that since both s and m are polynomially bounded in n, so is d) and polynomial $w(\cdot)$ such that $|\mathsf{Adv}_d^{\mathcal{D}}| \geq \frac{1}{w(n)}$. Let d^* be the *smallest* divisor of d such that[6] $|\mathsf{Adv}_{d^*}^{\mathcal{D}}| \geq \frac{d^{*3}}{d^3 w(n)}$ (in particular, this implies that $|\mathsf{Adv}_{d'}^{\mathcal{D}}| < \frac{d'^3}{d^3 w(n)}$ for all $d' \mid d^*$). Since $\frac{d^{*3}}{d^3 w(n)} \geq \frac{1}{\mathsf{poly}(n)}$ and $|\mathsf{Adv}_{d'}^{\mathcal{D}}| = \mathsf{negl}(n)\ \forall d' < s$, it should be the case that $d^* \geq s$. Consider now the predictor $\mathcal{P}$ shown in Algorithm 1 ($\mathcal{P}$ tries to guess the inner product $\mathbf{r} \cdot \mathbf{x} \pmod{d^*}$).

---

[5] We only care about the predicting advantage being noticeable and do not seek to optimize it as a function of the distinguishing advantage. We simply mention that the success probability ϵ of the predictor is $\epsilon \geq \delta/4ms^2$.

[6] such a d^* always exists. Indeed d itself satisfies this condition and is a divisor of itself.

input : $(\mathbf{g}, y, \mathbf{r})$ // $y = \mathbf{g} \cdot \mathbf{x},\ \mathbf{r} \leftarrow \mathcal{U}(\mathbb{Z}_{d^*}^m)$
output: $guess \in \mathbb{Z}_{d^*}$

1 Pick $c \leftarrow \mathcal{U}(\mathbb{Z}_{d^*})$;
2 Pick $g_1 \leftarrow \mathcal{U}(G), g_2 \leftarrow \mathcal{U}(G)$;
3 $\bar{\mathbf{g}} \leftarrow \mathbf{g} - \mathbf{r} \cdot g_1$ // $\mathbf{r} \cdot g_1 = (r_1 \cdot g_1, \ldots, r_m \cdot g_1)$;
4 Run $\mathcal{D}$ on input $(\bar{\mathbf{g}}, y - c \cdot g_1 + d^* \cdot g_2)$;
5 **if** $\mathcal{D}$ *returns 1* **then**
6 $guess \leftarrow c$;
7 **else**
8 $guess \leftarrow \mathcal{U}(\mathbb{Z}_{d^*} \setminus c)$;
9 **end**
10 return $guess$

Algorithm 1: Predictor for strong reduction (Proposition 2)

It can be checked that, if $\mathcal{P}$'s guess for $\mathbf{r} \cdot \mathbf{x} \pmod{d^*}$ (line 1) is correct, then the input distribution to $\mathcal{D}$ (line 4) is exactly $\mathcal{F}_{d^*}(\mathcal{K})$. Otherwise the input distribution to $\mathcal{D}$ is $\mathcal{F}_{d'}(\mathcal{K})$ for some $d' \mid d^*$ with $d' < d^*$.

It only remains to compute the *bias* of $\mathcal{P}$. First notice that

$$\begin{aligned} Pr\{k \leftarrow \mathcal{E}_{d^*}(\mathcal{P})\} &= \Pr\left[\, guess \equiv_{d^*} v - k \,\right] \\ &= \sum_{j=0}^{d^*-1} \Pr\left[\, guess \equiv_{d^*} v - k \,\middle|\, c \equiv_{d^*} v - j \,\right] \Pr\left[\, c \equiv_{d^*} v - j \,\right] \\ &= \frac{1}{d^*} \sum_{j=0}^{d^*-1} \Pr\left[\, guess \equiv_{d^*} v - k \,\middle|\, c \equiv_{d^*} v - j \,\right] \end{aligned} \tag{4}$$

Conditioning on $\mathcal{D}$'s output and after doing some calculations, we get

$$Pr\{k \leftarrow \mathcal{E}_{d^*}(\mathcal{P})\} = \frac{1}{d^*} + \frac{1}{d^*} \mathsf{prob}^{\mathcal{D}}_{\gcd(k,d^*)} - \frac{1}{d^*(d^*-1)} \sum_{j \neq k} \mathsf{prob}^{\mathcal{D}}_{\gcd(j,d^*)}$$

which implies that

$$Pr\{k \leftarrow \mathcal{E}_{d^*}(\mathcal{P})\} - Pr\{1 \leftarrow \mathcal{E}_{d^*}(\mathcal{P})\} = \frac{\mathsf{prob}^{\mathcal{D}}_{\gcd(k,d^*)} - \mathsf{prob}^{\mathcal{D}}_{1}}{d^* - 1} = \frac{\mathsf{Adv}^{\mathcal{D}}_{\gcd(k,d^*)}}{d^* - 1}$$

Using this and the fact that $\sum_{k=0}^{d^*-1} \omega_{d^*}^{-k} = 0$ we get that

$$\begin{aligned} \left| \sum_{k=0}^{d^*-1} Pr\{k \leftarrow \mathcal{E}_{d^*}(\mathcal{P})\} \cdot \omega_{d^*}^{-k} \right| &= \frac{1}{d^* - 1} \left| \sum_{k=0}^{d^*-1} \mathsf{Adv}^{\mathcal{D}}_{\gcd(k,d^*)} \omega_{d^*}^{-k} \right| \\ &\geq \frac{1}{d^* - 1} \left[\left| \mathsf{Adv}^{\mathcal{D}}_{d^*} \right| - \sum_{k=1}^{d^*-1} \left| \mathsf{Adv}^{\mathcal{D}}_{\gcd(k,d^*)} \right| \right] \end{aligned} \tag{5}$$

Next we bound $\sum_{k=1}^{d^*-1} \left|\mathsf{Adv}^{\mathcal{D}}_{\gcd(k,d^*)}\right|$. Define $\Phi(d^*,k) = \{1 \le i < d^* : \gcd(i,d^*) = k\}$ and let[7] $\phi(d^*,k) = |\Phi(d^*,k)|$. Clearly $\phi(d^*,d') \le \frac{d^*}{d'}$ $\forall d' \mid d^*$. So

$$\sum_{k=1}^{d^*-1} \left|\mathsf{Adv}^{\mathcal{D}}_{\gcd(k,d^*)}\right| \le \sum_{\substack{d' \mid d^* \\ d'<d^*}} \phi(d^*,d') \left|\mathsf{Adv}^{\mathcal{D}}_{\gcd(k,d^*)}\right| \le \frac{d^*}{d^3 w(n)} \sum_{\substack{d' \mid d^* \\ d'<d^*}} d'^2$$

where in the last inequality we used the fact that for all proper divisors d' of d^*, $|\mathsf{Adv}^{\mathcal{D}}_{d'}| < \frac{d'^3}{d^3 w(n)}$. Replacing back in (5) we finally get

$$\begin{aligned}\left|\sum_{k=0}^{d^*-1} Pr\{k \leftarrow \mathcal{E}_{d^*}(\mathcal{P})\} \cdot \omega_{d^*}^{-k}\right| &\ge \frac{1}{d^*-1}\left[\frac{d^{*3}}{d^3 w(n)} - \frac{d^*}{d^3 w(n)} \sum_{\substack{d' \mid d^* \\ d'<d^*}} d'^2\right] \\ &\ge \frac{d^{*3}}{d^3(d^*-1)w(n)}\left(2 - \frac{\pi^2}{6}\right) \ge \frac{1}{q(n)}\end{aligned}$$

for some polynomial $q(\cdot)$. In the last inequality we used the fact that for any $d \in \mathbb{N}$, $\sum_{r|d, r<d} r^2 \le (\pi^2/6) \cdot d^2$.

4 Implications and applications

Theorem 2 provides explicit criteria for checking if a knapsack family is pseudorandom. For a group G and input distribution $\mathcal{X}$, one needs only to check whether the folded families $\mathcal{K}_d = \mathcal{K}(G_d, \mathcal{X})$ are pseudorandom. As it turns out, for many choices of $(G, \mathcal{X})$, the folded knapsack functions $\mathcal{K}_d$ *compress* their input and map $\mathcal{X}$ to a distribution which is *statistically close* to uniform over G_d. More specifically, $\Delta_U(\mathcal{F}(\mathcal{K}(G_d, \mathcal{X}))) = \mathsf{negl}(n)$, and $\mathcal{K}(G_d, \mathcal{X})$ is pseudorandom in a strong statistical sense. Below, we provide some representative examples focusing on those that are most interesting in applications.

4.1 Specific groups and input distributions

We start with groups G whose order does not contain any factors that are smaller than the maximum value the input can take, i.e. $[\mathcal{X}] \subseteq [s]^m$ and any prime factor of $|G|$ is at least as large as s. In this case, a direct interpretation of Theorem 2 reveals that one-wayness implies pseudorandomness for *any* input distribution.

Corollary 1. *Let p be the smallest prime factor of $|G|$ and $\mathcal{X}$ be such that $[\mathcal{X}] \subseteq [p]^m$. If $\mathcal{K}(G, \mathcal{X})$ is one-way, then it is also pseudorandom.*

Corollary 1 is already very powerful. For instance, in the standard subset sum problem we have $[\mathcal{X}] = \{0,1\}^m \subseteq [p]^m$ for any prime p. Therefore, Corollary 1

[7] This is a generalization of Euler's totient function.

significantly generalizes the results from [18] and [12]. More specifically, it asserts that *any* knapsack family $\mathcal{K}(G, \mathcal{X})$ with $[\mathcal{X}] \subseteq \{0,1\}^m$ is pseudorandom provided it is one-way, for *any* abelian group G. Other interesting groups Corollary 1 is directly applicable to include groups with prime order, vector groups $\mathbb{Z}_p^k$ for prime p and generally groups $\mathbb{Z}_{p^e}^k$ where p is a prime such that $[\mathcal{X}] \subseteq [p]^m$.

For groups with small prime factors (smaller than s, where $[\mathcal{X}] \subseteq [s]^m$), the connection between one-wayness and pseudorandomness is more subtle: *search to decision* equivalence can be shown only for some input distributions and groups G. We summarize a few such examples focusing on *vector groups*, i.e. $G = \mathbb{Z}_q^k$ both for simplicity and because those groups are most interesting from a cryptographic viewpoint (see Section 4.2). Throughout, we assume $m - k = \omega(\log n)$.

For a vector group $G = \mathbb{Z}_q^k$ consider the folded knapsack function $\mathcal{K}_d = \mathcal{K}(G_d, \mathcal{X})$. First notice that $M_G = q$ and $dG = d\mathbb{Z}_q^k = \gcd(d, q) \cdot \mathbb{Z}_q^k$. By Theorem 2, proving pseudorandomness of $\mathcal{K}(G, \mathcal{X})$ amounts to proving that $\mathcal{K}_d$ are pseudorandom for all $d < s$ with $d \mid q$. In fact, below we study cases where $\mathcal{K}_d$ are *statistically* random, i.e. $\Delta_U(\mathcal{F}(\mathcal{K}(\mathbb{Z}_d^k, \mathcal{X}))) = \mathsf{negl}(n)$ for all divisors $d < s$ of q. Lemma 6 provides sufficient conditions for pseudorandomness expressed in terms of the statistical properties of $\mathcal{X}$ and the factorization of q. The statistical properties of $\mathcal{X}$ can be better expressed by defining the *d-folded* distribution $\mathcal{X}_d = \{\mathbf{x} \pmod d \mid \mathbf{x} \leftarrow \mathcal{X}\}$. Lemma 6 then requires that, for every "small" divisor of q, the d-folded distribution $\mathcal{X}_d$ has collision probability sufficiently smaller than a quantity that depends exclusively on d^k, the order of the quotient group $G_d = \mathbb{Z}_d^k$. The proof follows almost immediately from Theorem 2 and Lemma 4.

Lemma 6. *If $\mathcal{K} = \mathcal{K}(\mathbb{Z}_q^k, \mathcal{X})$ is one-way, $[\mathcal{X}] \subseteq [s]^m$ and $d^k \cdot \mathsf{Col}(\mathcal{X}_d) = \mathsf{negl}(n)$ for all $d \mid q$ with $d < s$, then $\mathcal{K}(\mathbb{Z}_q^k, \mathcal{X})$ is also pseudorandom.*

Below, we present 2 natural families of distributions which have small collision probability when "folded" over small d. Search to decision reductions for the corresponding knapsack families follow directly from Lemma 6 and the bounds on the collision probability of the two distributions. Lemmas 7 and 8 provide formal statements.

Uniformly Folded Distributions. For a given vector group G we say that a distribution $\mathcal{X}$ with $[\mathcal{X}] \subseteq [s]^m$ is *uniformly folded* with respect to G, if $\mathcal{X}_d = (\mathcal{X} \bmod d) \overset{s}{\approx} \mathcal{U}(\mathbb{Z}_d^m)$ for all $d < s$ such that $d \mid M_G$. When $G = \mathbb{Z}_q^k$, one such example is $\mathcal{X} = \mathcal{U}(\mathbb{Z}_q^m)$ or $\mathcal{X} = \mathcal{U}(\mathbb{Z}_{p^i}^m)$ when $q = p^e$ for some $e > i$.

Lemma 7. *If $\mathcal{K}(\mathbb{Z}_q^k, \mathcal{X})$ is one-way and $\mathcal{X}$ is uniformly folded with respect to $\mathbb{Z}_q^k$, (with $[\mathcal{X}] \subseteq [s]^m$ for $s = \mathsf{poly}(n)$), then it is also pseudorandom.*

Gaussian. Gaussian-like distributions are typically used for sampling the error in LWE-based cryptographic constructions. The following lemma establishes the search-to-decision reduction for knapsack families defined over $\mathbb{Z}_q^k$ and discrete Gaussian input distribution. Qualitatively similar results hold for *discretized* (rounded) Gaussians.

Lemma 8. *Let r be the Gaussian parameter with*[8] $\omega(\log n) \leq r \leq \mathsf{poly}(n)$. *If $\mathcal{K}(\mathbb{Z}_q^k, \mathcal{D}_{\mathbb{Z}^m,r})$ is one-way then it is also pseudorandom provided that either*
(a) q is prime or
(b) q is composite and there exists a function $\beta(n) = \omega(\sqrt{\log n})$ such that all divisors d of q lie outside the interval $[r/\beta(n), r \cdot \beta(n)]$.

4.2 Applications to LWE

In this section, we show how our results for knapsack functions imply similar search-to-decision reductions for the Learning With Errors (LWE) problem with the interesting feature of being *sample-preserving*. Following existing LWE literature, we use n for the length of the secret vector $\mathbf{s}$, m for the number of samples, q for the modulus and χ for the error distribution. Let n, m, q be positive integers and χ a distribution with $[\chi] \subseteq \mathbb{Z}_q$. For a vector $\mathbf{s} \in \mathbb{Z}_q^n$, define the distribution

$$\mathcal{A}_{\mathbf{s},\chi} = \{(\mathbf{a}, \mathbf{a} \cdot \mathbf{s} + e) \mid \mathbf{a} \leftarrow \mathcal{U}(\mathbb{Z}_q^n), e \leftarrow \chi\}.$$

The LWE problem is the problem of recovering $\mathbf{s}$ given m samples from distribution $\mathcal{A}_{\mathbf{s},\chi}$. In its decisional version (DLWE), one is given m samples drawn independently at random either from $\mathcal{A}_{\mathbf{s},\chi}$ (for some secret $\mathbf{s}$) or from $\mathcal{U}(\mathbb{Z}_q^n \times \mathbb{Z}_q)$. The goal is to tell the two distributions apart with noticeable probability.

We are interested in reductions from LWE to DLWE that *preserve* all the parameters n, m, q, χ, including the *number of samples* m. Sample-preserving reductions are more naturally described using matrix notation for the LWE problem. Given a collection of m LWE samples $(\mathbf{a}_i, b_i) \leftarrow \mathcal{A}_{\mathbf{s},\chi}$, we can combine them in a matrix $\mathbf{A}$ having the vectors $\mathbf{a}_i$ as rows, and a column vector $\mathbf{b}$ with entries equal to b_i. That is, $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}$ where $\mathbf{e} \leftarrow \chi^m$. With this notation, we want to prove that any algorithm that distinguishes $(\mathbf{A}, \mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e})$ from $\mathcal{U}(\mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m)$ can be used to recover the secret $\mathbf{s}$. Notice that once the secret $\mathbf{s}$ has been recovered, one can also recover the error vector $\mathbf{e} = \mathbf{b} - \mathbf{A}\mathbf{s}$. So, we can equivalently define LWE as the problem of recovering both $\mathbf{s}$ and $\mathbf{e}$. This is exactly the problem of inverting the following function family.

Definition 2. *Let n, m, q and χ defined as above.* $\mathrm{LWE}(n, m, q, \chi)$ *is the function family $(F, \mathcal{X})$ where $\mathcal{X} = \{(\mathbf{s}, \mathbf{e}) \mid \mathbf{s} \leftarrow \mathcal{U}(\mathbb{Z}_q^n), \mathbf{e} \leftarrow \chi^m\}$, and F is the set of functions $f_{\mathbf{A}}$ indexed by $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ and defined as $f_{\mathbf{A}}(\mathbf{s}, \mathbf{e}) = \mathbf{A}\mathbf{s} + \mathbf{e}$.*

The decision LWE is the problem of distinguishing $\mathcal{F}(\mathrm{LWE}(n, m, q, \chi))$ from $\mathcal{U}(\mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m)$. However, $\mathrm{LWE}(n, m, q, \chi)$ is not a knapsack family. In order to apply the results from Section 3, we exploit the duality between the LWE problem and an associated knapsack function family described in the following lemmas.

Lemma 9. *For any*[9] $n, m \geq n + \omega(\log n), q$ *and χ, there is a polynomial time reduction from the problem of inverting* $\mathrm{LWE}(n, m, q, \chi)$ *with probability ϵ, to the problem of inverting $\mathcal{K}(\mathbb{Z}_q^{m-n}, \chi^m)$ with probability $\epsilon' = \epsilon + \mathsf{negl}(n)$.*

[8] In typical instantiations, $r = \Omega(n^\theta)$ for some constant $\theta > 0$.

[9] The requirement $m \geq n + \omega(\log n)$ is a standard assumption in the context of LWE, where typically $m \geq n + \Omega(n)$.

Proof (Sketch). The transformation from the LWE problem into an equivalent knapsack problem requires that the matrix $\mathbf{A}$ be nonsingular, i.e., the rows of $\mathbf{A}$ generate $\mathbb{Z}_q^n$. When $\mathbf{A} \leftarrow \mathcal{U}(\mathbb{Z}_q^{m \times n})$, this is true except with probability at most $1/p^{m-n-1}$, where p is the smallest prime factor of q. So, for $m \geq n + \omega(\log n)$, $\Pr[\ \mathbf{A}$ singular $\ ] = \mathsf{negl}(n)$. We can therefore assume $\mathbf{A}$ has been chosen at random, but conditioned on the property that it is nonsingular.

Consider now the set that contains all vectors $\mathbf{g}$ such that $\mathbf{gA} = 0 \pmod q$. Under the assumption that $\mathbf{A}$ is nonsingular, this set is generated by the rows of a matrix $\mathbf{G} \in \mathbb{Z}_q^{(m-n) \times m}$ that can be efficiently computed from $\mathbf{A}$ using linear algebra. We can further randomize $\mathbf{G}$ by left-multiplying it by a random unimodular matrix $\mathbf{U} \in \mathbb{Z}_q^{(m-n)\times(m-n)}$. Finally, if $\mathbf{A}$ is chosen at random among all nonsingular matrices, then this randomized $\mathbf{G}$ is also distributed uniformly at random among all matrices whose *columns* generate $\mathbb{Z}_q^{m-n}$. As before, the distribution of $\mathbf{G}$ is within negligible statistical distance from $\mathcal{U}(\mathbb{Z}_q^{(m-n)\times m})$, so we can treat the columns of $\mathbf{G}$ as random elements from the vector group $G = \mathbb{Z}_q^{m-n}$. Finally, we set $\mathbf{c} = \mathbf{Gb} = \mathbf{GAs} + \mathbf{Ge} = \mathbf{Ge}$, so the distribution $(\mathbf{G}, \mathbf{c})$ is *statistically close* to a *random instance* of the knapsack problem with group $G = \mathbb{Z}_q^{m-n}$ and input distributed according to the error distribution χ^m.

Lemma 10. *For any $n, m \geq n + \omega(\log n), q$ and χ, there is a polynomial time reduction from the problem of distinguishing $\mathcal{F}(\mathcal{K}(\mathbb{Z}_q^{m-n}, \chi^m))$ from uniform with advantage ϵ to the problem of distinguishing $\mathcal{F}(\mathrm{LWE}(n, m, q, \chi))$ from uniform with advantage $\epsilon' = \epsilon + \mathsf{negl}(n)$.*

Proof (Proof Sketch). The reduction reverses the steps taken to transform LWE into knapsack. We start from a pair $(\mathbf{G}, \mathbf{c})$. As before, we can assume that the columns of $\mathbf{G}$ generate $\mathbb{Z}_q^{m-n}$. Next, by linear algebra, we compute a matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ whose columns generate the set of vectors $\mathbf{a}$ such that $\mathbf{Ga} = \mathbf{0} \pmod q$. As before, we can randomize $\mathbf{A}$ by right-multiplying it by a random unimodular matrix $\mathbf{U} \in \mathbb{Z}_q^{n \times n}$ to obtain $\mathbf{A'}$. We also map $\mathbf{c}$ to $\mathbf{A's'} + \mathbf{r}$ where $\mathbf{s'} \leftarrow \mathcal{U}(\mathbb{Z}_q^n)$ and $\mathbf{r}$ is a random solution to the equation $\mathbf{Gr} = \mathbf{c}$. It can be checked that this transformation maps the knapsack distribution $(\mathbf{G}, \mathbf{c} = \mathbf{Ge})$ to the LWE distribution $(\mathbf{A'}, \mathbf{A's} + \mathbf{e})$ (with uniformly random $\mathbf{s}$), when $\mathbf{G}$ and $\mathbf{A'}$ are chosen at random subject to the constraint that they are nonsingular. The transformation also maps the uniform distribution to a (statistically close to) uniform distribution.

LWE: From Search to Decision Sample-preserving search to decision reductions for LWE are immediately obtained combining the reductions from Lemma 9 and Lemma 10 with the results from Section 3 on $\mathcal{K}(\mathbb{Z}_q^{m-n}, \chi^m)$. Similarly to the knapsack case, the reductions *do not hold unconditionally*; rather they hold for specific, yet very broad, moduli q and error distributions χ. Below we give a general statement for the search to decision reduction parametrized by n, m, q and χ. Upon giving the statement, we provide specific instantiations of the error distribution χ and the modulus q for which the statement holds. Throughout, we assume that $m \geq n + \omega(\log n)$.

Proposition 3. *Assume there exists an efficient algorithm $\mathcal{D}$ that distinguishes between $\mathcal{F}(\mathrm{LWE}(n, m, q, \chi))$ and $\mathcal{U}(\mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m)$ with noticeable advantage. Then there exists an efficient algorithm $\mathcal{I}$ that inverts $\mathrm{LWE}(n, m, q, \chi)$ with noticeable success probability.*

The following "assignments" provide examples of q and χ that make the above statement true.

- prime $q = \Theta(n^c)$ for *constant* c and $\chi = \mathcal{D}_{\mathbb{Z},r}$. The search to decision reduction of the corresponding bounded knapsack problem follows directly from Corollary 1. Setting q and χ as above is typical for instantiations of LWE-based cryptographic applications.
- $q = p^e$ for prime $p = \mathsf{poly}(n)$, and $\chi = \mathcal{D}_{\mathbb{Z},r}$ for "sufficiently narrow" standard deviation (more specifically, it is required that $r = o(\frac{p}{\log n})$). Again, the search to decision reduction of the bounded knapsack problem stems from Corollary 1. We note that this case provides a sample-preserving version of the search to decision reduction proved in [6].
- $q = p^e = \mathsf{poly}(n)$, with $\chi = \mathcal{U}(\mathbb{Z}_{p^i})$ for some $i < e$. Pseudorandomness of the knapsack instance stems directly from Lemma 7. Search to decision reduction for LWE with such noise distribution appears to be new; no such (even non-sample-preserving) reduction has previously appeared in the literature.

5 Open Problems

Our work leaves many interesting open questions. To start with, sample-preserving search to decision reductions for LWE with *bounded* noise as considered in this work, don't seem to extend to the unbounded noise regime, i.e. when each coefficient e_i of the error vector $\mathbf{e}$ of LWE is drawn from a set with *superpolynomial* size. We note that such search to decision reductions are known [28] but are *not* sampling preserving. These reductions rely heavily on a Chinese Reminder Theorem (CRT) approach: using a *perfect*[10] distinguisher, they first learn the secret modulo p_i with *overwhelming success probability* for each *polynomially bounded* prime factor p_i of the modulus q; they then use the CRT to recover the entire secret. In sample preserving reductions, where only an *imperfect* distinguisher can be afforded by the available number of samples, learning the secret modulo p_i can be performed in a much looser, list-decoding sense: the secret modulo p_i is included in the corresponding lists L_i but among possibly *many* other elements. And the only way to check which of the list elements corresponds to the secret modulo p_i seems to be by forming first the entire secret using CRT and then verifying that the result is the LWE secret. Thus, one has to solve superpolynomially many CRT instances before recovering the correct value of the secret. It would be nice to extend the list-decoding approach to work even in that case.

[10] By perfect here we mean a distinguisher with advantage almost 1. Getting a perfect distinguisher out of an imperfect one (one with only a nonnegligible advantage) is the main reason for the blowup in the number of samples the reduction consumes.

As an additional motivation, we mention that extending our sample preserving reductions to the unbounded error setting is likely to have implications to the search to decision equivalence of the newly introduced Ring LWE (R-LWE) problem [24]. R-LWE is an algebraic variant of LWE that leads to much more efficient constructions than standard LWE while still enjoying strong security guarantees. Much like LWE with unbounded noise, existing search to decision reductions [24] decompose the secret (which is an element from a ring R) modulo $\mathfrak{q}_i$ where $\mathfrak{q}_i$ are prime ideal factors.

Our work also highlights the importance of understanding the hardness of LWE under various noise distributions. Current hardness proofs for search LWE [31] based on worst-case lattice problems rely on the noise following a Gaussian distribution. Can lattice-based hardness results for search LWE be extended to noise distributions other than Gaussian? Can we show similar lattice-based hardness results if the noise is distributed uniformly at random modulo 2^i? The latter case is very attractive from a practical viewpoint since arithmetic modulo 2 and sampling from uniform distributions can be implemented very efficiently.

References

1. Shweta Agrawal, Dan Boneh, and Xavier Boyen. Efficient Lattice (H)IBE in the Standard Model. In *EUROCRYPT*, pages 553–572, 2010.
2. Shweta Agrawal, Dan Boneh, and Xavier Boyen. Lattice Basis Delegation in Fixed Dimension and Shorter-Ciphertext Hierarchical IBE. In *CRYPTO*, pages 98–115, 2010.
3. Adi Akavia. *Learning Noisy Characters, Multiplication Codes and Hardcore Predicates*. PhD thesis, MIT, February 2008.
4. Adi Akavia, Shafi Goldwasser, and Shmuel Safra. Proving Hard-Core Predicates Using List Decoding. In *FOCS*, pages 146–157, 2003.
5. Adi Akavia, Shafi Goldwasser, and Vinod Vaikuntanathan. Simultaneous Hardcore Bits and Cryptography against Memory Attacks. In *TCC*, pages 474–495, 2009.
6. Benny Applebaum, David Cash, Chris Peikert, and Amit Sahai. Fast Cryptographic Primitives and Circular-Secure Encryption Based on Hard Learning Problems. In *CRYPTO*, pages 595–618, 2009.
7. Sanjeev Arora and Rong Ge. New algorithms for learning in presence of errors. In *ICALP*, 2011. Available at http://www.eccc.uni-trier.de/report/2010/066/.
8. Avrim Blum, Merrick L. Furst, Jeffrey C. Jackson, Michael J. Kearns, Yishay Mansour, and Steven Rudich. Weakly Learning DNF and Characterizing Statistical Query Learning using Fourier Analysis. In *STOC*, pages 253–262, 1994.
9. Avrim Blum, Merrick L. Furst, Michael J. Kearns, and Richard J. Lipton. Cryptographic Primitives Based on Hard Learning Problems. In *CRYPTO*, pages 278–291, 1993.
10. David Cash, Dennis Hofheinz, Eike Kiltz, and Chris Peikert. Bonsai Trees, or How to Delegate a Lattice Basis. In *EUROCRYPT*, pages 523–552, 2010.
11. Yevgeniy Dodis, Shafi Goldwasser, Yael Tauman Kalai, Chris Peikert, and Vinod Vaikuntanathan. Public-Key Encryption Schemes with Auxiliary Inputs. In *TCC*, pages 361–381, 2010.
12. Jean-Bernard Fischer and Jacques Stern. An efficient pseudo-random generator provably as secure as syndrome decoding. In *EUROCRYPT*, pages 245–255, 1996.

13. Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. A Simple BGN-Type Cryptosystem from LWE. In *EUROCRYPT*, pages 506–522, 2010.
14. Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for Hard Lattices and New Cryptographic Constructions. In *STOC*, pages 197–206, New York, NY, USA, 2008. ACM.
15. Oded Goldreich and Leonid A. Levin. A Hard-Core Predicate for All One-Way Functions. In *STOC*, pages 25–32, 1989.
16. Shafi Goldwasser, Yael Tauman Kalai, Chris Peikert, and Vinod Vaikuntanathan. Robustness of the Learning with Errors Assumption. In *ICS*, 2010.
17. R. Impagliazzo and D. Zuckerman. How to Recycle Random Bits. In *FOCS*, pages 248–253, Washington, DC, USA, 1989. IEEE Computer Society.
18. Russell Impagliazzo and Moni Naor. Efficient Cryptographic Schemes Provably as Secure as Subset Sum. *J. Cryptology*, 9(4):199–216, 1996.
19. Jonathan Katz, Ji Sun Shin, and Adam Smith. Parallel and Concurrent Security of the HB and HB^+ Protocols. *J. Cryptology*, 23(3):402–421, 2010.
20. Akinori Kawachi, Keisuke Tanaka, and Keita Xagawa. Multi-bit Cryptosystems Based on Lattice Problems. In *Public Key Cryptography*, pages 315–329, 2007.
21. Eyal Kushilevitz and Yishay Mansour. Learning Decision Trees Using the Fourier Sprectrum. In *STOC*, pages 455–464, 1991.
22. Richard Lindner and Chris Peikert. Better Key Sizes (and Attacks) for LWE-Based Encryption. In *CT-RSA*, pages 319–339, 2011.
23. Vadim Lyubashevsky and Daniele Micciancio. On bounded distance decoding, unique shortest vectors, and the minimum distance problem. In *CRYPTO*, pages 577–594, 2009.
24. Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On Ideal Lattices and Learning with Errors over Rings. In *EUROCRYPT*, pages 1–23, 2010.
25. Daniele Micciancio. Duality in Lattice Based Cryptography. In *Public Key Cryptography*, 2010. Invited Talk.
26. Daniele Micciancio and Oded Regev. Lattice-Based Cryptography. In *Post Quantum Cryptography*, pages 147–191. Springer Publishing Company, 2009.
27. Elchanan Mossel, Ryan O'Donnell, and Rocco A. Servedio. Learning Juntas. In *STOC*, pages 206–212, 2003.
28. Chris Peikert. Public-Key Cryptosystems from the Worst-Case Shortest Vector Problem. In *STOC*, pages 333–342, New York, NY, USA, 2009. ACM.
29. Chris Peikert, Vinod Vaikuntanathan, and Brent Waters. A Framework for Efficient and Composable Oblivious Transfer. In *CRYPTO*, pages 554–571, Berlin, Heidelberg, 2008. Springer-Verlag.
30. Chris Peikert and Brent Waters. Lossy Trapdoor Functions and Their Applications. In *STOC*, pages 187–196, New York, NY, USA, 2008. ACM.
31. Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of ACM*, 56(6):34, September 2009. Preliminary version in STOC 2005.
32. Oded Regev. The Learning with Errors Problem (Invited Survey). In *IEEE Conference on Computational Complexity*, pages 191–204, 2010.
33. Markus Rückert and Michael Schneider. Estimating the Security of Lattice-based Cryptosystems. Technical Report 2010/137, IACR ePrint archive, 2010.
34. Daniel Stefankovic. Fourier Transform in Computer Science. Master's thesis, University of Chicago, October 2000.
35. Damien Stehlé, Ron Steinfeld, Keisuke Tanaka, and Keita Xagawa. Efficient Public Key Encryption Based on Ideal Lattices. In *ASIACRYPT*, pages 617–635, 2009.