Simple Lattice Trapdoor Sampling from a Broad Class of Distributions

Vadim Lyubashevsky[1]* and Daniel Wichs[2] **

[1] Inria/ENS, Paris
[2] Northeastern University

Abstract. At the center of many lattice-based constructions is an algorithm that samples a short vector $\mathbf{s}$, satisfying $[\mathbf{A}|\mathbf{AR}-\mathbf{HG}]\mathbf{s} = \mathbf{t} \bmod q$ where $\mathbf{A}, \mathbf{AR}, \mathbf{H}, \mathbf{G}$ are public matrices and $\mathbf{R}$ is a trapdoor. Although the algorithm crucially relies on the knowledge of the trapdoor $\mathbf{R}$ to perform this sampling efficiently, the distribution it outputs should be independent of $\mathbf{R}$ given the public values. We present a new, simple algorithm for performing this task. The main novelty of our sampler is that the distribution of $\mathbf{s}$ does not need to be Gaussian, whereas all previous works crucially used the properties of the Gaussian distribution to produce such an $\mathbf{s}$. The advantage of using a non-Gaussian distribution is that we are able to avoid the high-precision arithmetic that is inherent in Gaussian sampling over arbitrary lattices. So while the norm of our output vector $\mathbf{s}$ is on the order of $\sqrt{n}$ to n - times larger (the representation length, though, is only a constant factor larger) than in the samplers of Gentry, Peikert, Vaikuntanathan (STOC 2008) and Micciancio, Peikert (EUROCRYPT 2012), the sampling itself can be done very efficiently. This provides a useful time/output trade-off for devices with constrained computing power. In addition, we believe that the conceptual simplicity and generality of our algorithm may lead to it finding other applications.

1 Introduction

At the core of many lattice-based cryptosystems is the many-to-one one-way function $f_{\mathbf{A}}(\mathbf{s}) = \mathbf{A}\mathbf{s} \bmod q$, where $\mathbf{A} \in \mathbb{Z}_q^{m\times n}$ is a random (short & fat) matrix and $\mathbf{s} \in \mathbb{Z}_q^m$ is a "short" vector. The works of [Ajt96,Ajt99] showed that this function is one-way under a worst-case lattice assumption, and moreover, that there is a way to sample a random $\mathbf{A}$ along with a trapdoor that allows one to invert the function $f_{\mathbf{A}}$. However, since the function $f_{\mathbf{A}}$ is many-to-one, the choice of which pre-image we sample might depend on which trapdoor for $\mathbf{A}$ we use. Not leaking information about the trapdoor, which is used as a secret key in cryptographic schemes, is essential for security – both "provable" and actual. Some early lattice schemes, such as GGH [GGH97] and NTRU [HHGP$^+$03] signatures, did not have security proofs, and it was subsequently shown that obtaining a small amount of signature samples was enough to completely recover the secret key [NR09].

* Partially supported by the French ANR-13-JS02-0003 "CLE" Project
** Supported by NSF grants 1347350, 1314722, 1413964.

The first algorithm which was able to sample pre-images of $f_{\mathbf{A}}$ without leaking any information about the trapdoor was devised in the breakthrough work of Gentry, Peikert, and Vaikuntanathan [GPV08]. It was able to output such a pre-image $\mathbf{s}$ according to a discrete Gaussian distribution using a short basis of the lattice $\mathcal{L}_q^{\perp}(\mathbf{A}) := \{\mathbf{v} \in \mathbb{Z}^m : \mathbf{A}\mathbf{v} = \mathbf{0} \bmod q\}$ as a trapdoor. Following the intuition of the two-sided sampler of Agrawal, Boneh, and Boyen [ABB10], Micciancio and Peikert introduced a sampling procedure that did not explicitly require a short basis of the underlying lattice [MP12]. In particular, instead of sampling a uniformly random matrix, they sampled a statistically close matrix $\mathbf{A}' = [\mathbf{A}|\mathbf{A}\mathbf{R} - \mathbf{H}\mathbf{G}]$, where $\mathbf{A}$ is a uniformly random (short & fat) matrix over $\mathbb{Z}_q$, $\mathbf{R}$ is a matrix with small coefficients, $\mathbf{H}$ is any matrix invertible over $\mathbb{Z}_q$, and $\mathbf{G}$ is a special (publicly-known) matrix that allows one to easily compute a small vector $\mathbf{x}$ satisfying $\mathbf{G}\mathbf{x} = \mathbf{t} \bmod q$ for any $\mathbf{t} \in \mathbb{Z}_q^n$. We think of $\mathbf{A}, \mathbf{A}\mathbf{R}, \mathbf{G}, \mathbf{H}$ as publicly known and think of $\mathbf{R}$ as a secret trapdoor. They showed how to sample a short pre-image $\mathbf{s} = (\mathbf{s}_1, \mathbf{s}_2)$ given some target $\mathbf{t}$ so as to satisfy:

$$f_{\mathbf{A}'}(\mathbf{s}) = \mathbf{A}'\mathbf{s} = [\mathbf{A}|\mathbf{A}\mathbf{R} - \mathbf{H}\mathbf{G}]\mathbf{s} = \mathbf{A}\mathbf{s}_1 + (\mathbf{A}\mathbf{R} - \mathbf{H}\mathbf{G})\mathbf{s}_2 = \mathbf{t} \bmod q. \qquad (1)$$

Furthermore, they ensure that the distribution of $\mathbf{s}$ does not depend on the trapdoor $\mathbf{R}$.

The intuition for how the Micciancio-Peikert sampler produces short vectors while hiding the exact value of $\mathbf{R}$ is as follows. If we define $\mathbf{A}' = [\mathbf{A}|\mathbf{A}\mathbf{R} - \mathbf{H}\mathbf{G}]$ and $\mathbf{R}' = \begin{bmatrix} \mathbf{R} \\ -\mathbf{I} \end{bmatrix}$, then $\mathbf{A}'\mathbf{R}' = \mathbf{H}\mathbf{G}$. To sample an $\mathbf{s}$ such that $\mathbf{A}'\mathbf{s} = \mathbf{t} \bmod q$, one first samples a vector $\mathbf{w}$ from a particular distribution, then samples a discrete Gaussian vector $\mathbf{z}$ satisfying $\mathbf{G}\mathbf{z} = \mathbf{H}^{-1}(\mathbf{t} - \mathbf{A}'\mathbf{w}) \bmod q$, and finally outputs $\mathbf{s} = \mathbf{w} + \mathbf{R}'\mathbf{z}$. One can check that $\mathbf{A}'\mathbf{s} = \mathbf{A}'\mathbf{w} + \mathbf{A}'\mathbf{R}'\mathbf{z} = \mathbf{t} - \mathbf{H}\mathbf{G}\mathbf{z} = \mathbf{t} \bmod q$. The main part of [MP12] consisted of proving that the distribution of $\mathbf{s}$ is independent of $\mathbf{R}$. If $\mathbf{z}$ is a discrete Gaussian with a large-enough standard deviation, then the distribution of $\mathbf{R}'\mathbf{z}$ is also a discrete Gaussian with covariance matrix approximately $\mathbf{R}'\mathbf{R}'^T$. Then, if the distribution of $\mathbf{w}$ was also a discrete Gaussian with covariance matrix $s^2\mathbf{I} - \mathbf{R}'\mathbf{R}'^T$, the covariance matrix of the distribution of $\mathbf{s} = \mathbf{w} + \mathbf{R}'\mathbf{z}$ is very close to the sum of the covariance matrices [Pei10], which is $s^2\mathbf{I}$, and is therefore independent of $\mathbf{R}$.[1]

Both the GPV and the Micciancio-Peikert samplers strongly rely on the Gaussian nature of the output distribution. The GPV algorithm samples vectors along the Gram-Schmidt decomposition of the trapdoor, which ends up being Gaussian due to the rotational invariance of the Gaussian distribution. Similarly, the Micciancio-Peikert sampler crucially relies on the convolution theorem for Gaussian distributions, which is also an implicit consequence of the rotational invariance.

[1] The matrix $\mathbf{H}$ does not in any way help in the inversion procedure. It is present only because it is very useful in constructions of various schemes such as CCA-secure encryption, digital signatures, and identity-based encryption schemes (we refer the reader to [MP12] for more details).

1.1 Our result

Our main result is another sampler (which is very much inspired by the works of [ABB10] and [MP12]) that outputs an $\mathbf{s}$ satisfying Equation (1) that does not inherently rely on the special properties of any distribution. For example, it is able to output an $\mathbf{s} = (\mathbf{s}_1, \mathbf{s}_2)$ where both $\mathbf{s}_1$ and $\mathbf{s}_2$ come from (different) uniform distributions, or the distribution of $\mathbf{s}_2$ could be uniform while $\mathbf{s}_1$ is a discrete Gaussian. The algorithm is also conceptually very simple. To sample an $\mathbf{s}$, we rewrite $\mathbf{A}\mathbf{s}_1 + (\mathbf{AR} - \mathbf{HG})\mathbf{s}_2 = \mathbf{t} \bmod q$ as $\mathbf{G}\mathbf{s}_2 = \mathbf{H}^{-1}(\mathbf{A}(\mathbf{s}_1 + \mathbf{R}\mathbf{s}_2) - \mathbf{t}) \bmod q$. We first pick the variable $\mathbf{y}$ corresponding to $\mathbf{s}_1 + \mathbf{R}\mathbf{s}_2$ according to some distribution P_y. Once this $\mathbf{y}$ is fixed, the value of $\mathbf{s}_2$ is deterministically determined via the equation $\mathbf{G}\mathbf{s}_2 = \mathbf{H}^{-1}(\mathbf{A}\mathbf{y} - \mathbf{t}) \bmod q$ according to some rule – for example, if $\mathbf{G}$ is the "powers-of-2" matrix (see (4) in Section 3.2), then $\mathbf{s}_2$ is just a concatenation of the binary decompositions of each coefficient of $\mathbf{H}^{-1}(\mathbf{A}\mathbf{y} - \mathbf{t}) \bmod q$.[2] Once $\mathbf{s}_2$ is chosen, the value of $\mathbf{s}_1$ is uniquely determined to be $\mathbf{s}_1 = \mathbf{y} - \mathbf{R}\mathbf{s}_2$. At this point, outputting $\mathbf{s} = (\mathbf{s}_1, \mathbf{s}_2)$ would leak $\mathbf{R}$, and we need to use rejection sampling to break the dependency. The idea is similar to that in [Lyu09,Lyu12], except that in our case $\mathbf{y}$ and $\mathbf{s}_2$ are dependent on each other (in particular, the only entropy in the whole algorithm is in $\mathbf{y}$ and in the coins that are eventually used for the rejection sampling) and one needs a more careful argument to show that the distribution of $(\mathbf{s}_1, \mathbf{s}_2)$ can have a distribution that is independent of $\mathbf{R}$.

The main advantage of our sampler lies in its conceptual simplicity and the generality of its output distribution. A hidden cost of discrete Gaussian sampling over arbitrary lattices is that it requires the storage of, and computation with, vectors of real numbers (for example, the Gram-Schmidt orthogonalization in [GPV08] or the square root of the covariance matrix in [Pei10,MP12]) with precision of at least the security parameter. This could pose a serious implementation obstacle on devices in which storage space and/or computational power are at a premium.[3] Using our new sampling algorithm, on the other hand, we can choose the distributions for $\mathbf{s}_1$ and $\mathbf{s}_2$ to be uniform, and then one only needs to perform uniform sampling over $\mathbb{Z}_q$ and the rejection sampling part of the algorithm simply involves checking whether all the coefficients are in a particular interval (see the first example in Section 3.2). If a little more processing power or storage capacity is available, we can change the distribution of $\mathbf{s}_1$ to a discrete Gaussian over $\mathbb{Z}^m$, which will make the outputs slightly shorter but will require some additional resources for doing discrete Gaussian sampling over $\mathbb{Z}^m$ (cf. [DDLL13,DG14,PDG14]) and for the rejection sampling step (see the second example in Section 3.1).

[2] One could choose $\mathbf{s}_2$ according to some (non-deterministic) distribution instead, but we do not at present see any reason to do so.

[3] It should be pointed out that the signature schemes of [Pei10,MP12] can do a lot of the necessary high-precision computations "offline" before receiving the message to be signed. In such an "online/offline" model, this can save on the computation time during the online phase, but the storage space still remains an issue.

The main disadvantage of our sampler is that the norm of the produced vector $\mathbf{s}_1$, and therefore the norm of the entire vector $\mathbf{s} = (\mathbf{s}_1, \mathbf{s}_2)$ of our algorithm, is larger by at least a $\sqrt{n}$ factor than of the ones produced by the samplers of [GPV08,MP12] (see the examples in Section 3.2). In practice, having the norms of the outputs be larger by a factor of $O(\sqrt{n})$ results in the bit-length of the output $\mathbf{s}$ to increase by a factor of 2 or 3 (e.g. compare [GLP12] to [DDLL13]). Therefore we believe that our sampler provides a time vs. size trade-off that is most useful in instances where Gaussian sampling over arbitrary lattices is either not possible or prohibitively expensive.

2 Preliminaries

Let X, Y be distributions or random variables with support S. We define their *statistical distance* by $\mathsf{SD}(X,Y) = \frac{1}{2}\sum_{s \in S} |\Pr[X = s] - \Pr[Y = s]|$. We write $X \approx_\varepsilon Y$ and say that X and Y are ε-statistically close to denote that $\mathsf{SD}(X,Y) \leq \varepsilon$. For a random variable X, we define the min-entropy of X as $H_\infty(X) := -\log(\max_x \Pr[X = x])$.

Lemma 2.1 (Leftover Hash Lemma). *[HILL99,NZ96] Let $\mathcal{H} = \{h : D \to R\}$ be a universal hash function family, meaning that for all $x \neq y \in D$ we have $\Pr_{h \leftarrow \mathcal{H}}[h(x) = h(y)] \leq 1/|R|$. Let X be any random variable with support D and min-entropy $H_\infty(X) \geq \log(|R|) + 2\log(1/\varepsilon)$. Then $(h, h(x))$ is ε-statistically close to (h, r) where $h \overset{\$}{\leftarrow} \mathcal{H}, x \overset{\$}{\leftarrow} X$ and $r \overset{\$}{\leftarrow} R$.*

In particular, for a prime q, setting $D = \mathbb{Z}_q^m$, $R = \mathbb{Z}_q^n$ and $\mathcal{H} = \{h_{\mathbf{A}}(\mathbf{x}) := \mathbf{A}\mathbf{x} \mid \mathbf{A} \in \mathbb{Z}_q^{n \times m}\}$, for any distribution $\mathbf{x}$ over $\mathbb{Z}_q^m$ having entropy $H_\infty(\mathbf{x}) \geq n\log(q) + 2\log(1/\varepsilon)$ we have $(\mathbf{A}, \mathbf{A}\mathbf{x})$ is ε-statistically close to $(\mathbf{A}, \mathbf{r})$ where $\mathbf{A} \overset{\$}{\leftarrow} \mathbb{Z}_q^{n \times m}$ and $\mathbf{r} \overset{\$}{\leftarrow} \mathbb{Z}_q^n$.

Lemma 2.2. *[Lyu12, Lemma 4.7]*

Let f, g be probability distributions with the property that

$$\exists M \in \mathbb{R}^+ \textit{ such that, } \Pr_{z \overset{\$}{\leftarrow} f}[Mg(z) \geq f(z)] \geq 1 - \varepsilon$$

then the distribution of the output of the following algorithm $\mathcal{A}$:

```
1: $z \overset{\$}{\leftarrow} g$
2: output $z$ with probability $\min\left(\frac{f(z)}{Mg(z)}, 1\right)$
```

is within statistical distance ε/M of the distribution of the following algorithm $\mathcal{F}$:

```
1: $z \overset{\$}{\leftarrow} f$
2: output $z$ with probability $1/M$
```

Moreover, the probability that $\mathcal{A}$ outputs something is at least $(1 - \varepsilon)/M$.

Ideal Distribution

```
1: Generate $\mathbf{s}_2 \xleftarrow{\$} P_2$
2: Generate $\mathbf{s}_1 \xleftarrow{\$} P_1 | \mathbf{A}\mathbf{s}_1 = \mathbf{t} + (\mathbf{H}\mathbf{G} - \mathbf{A}\mathbf{R})\mathbf{s}_2 \bmod q$
3: Output $\mathbf{s} = (\mathbf{s}_1, \mathbf{s}_2)$
```

Real Distribution

```
1: Generate $\mathbf{y} \xleftarrow{\$} P_y$
2: Compute $\mathbf{s}_2 \leftarrow \mathbf{G}^{-1}\big(\mathbf{H}^{-1}(\mathbf{A}\mathbf{y} - \mathbf{t}) \bmod q\big)$
3: Compute $\mathbf{s}_1 \leftarrow \mathbf{y} - \mathbf{R}\mathbf{s}_2$
4: Output $\mathbf{s} = (\mathbf{s}_1, \mathbf{s}_2)$ with probability $\frac{P_1(\mathbf{s}_1)}{M \cdot P_y(\mathbf{s}_1 + \mathbf{R}\mathbf{s}_2)}$
5: If nothing was output, GOTO 1.
```

Fig. 1. Ideal and Real Distributions

3 The Sampling Algorithm

Given matrices $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, $\mathbf{R} \in \mathbb{Z}_q^{m \times l}$, $\mathbf{G} \in \mathbb{Z}_q^{n \times l}$, $\mathbf{H} \in \mathbb{Z}_q^{n \times n}$ and a target $\mathbf{t} \in \mathbb{Z}_q^n$, we would like to output a short vector $\mathbf{s} = (\mathbf{s}_1, \mathbf{s}_2)$ that satisfies

$$[\mathbf{A} | \mathbf{A}\mathbf{R} - \mathbf{H}\mathbf{G}]\mathbf{s} = \mathbf{A}\mathbf{s}_1 + (\mathbf{A}\mathbf{R} - \mathbf{H}\mathbf{G})\mathbf{s}_2 = \mathbf{t} \bmod q \tag{2}$$

and furthermore, the distribution of $\mathbf{s}$ is independent of $\mathbf{R}$ given the "public" values $\mathbf{A}, \mathbf{A}\mathbf{R}, \mathbf{H}, \mathbf{G}$. In other words, if we think of $\mathbf{R}$ as a trapdoor needed to perform the sampling, we want to ensure that the sample $\mathbf{s}$ should not reveal anything about the trapdoor $\mathbf{R}$.

We present a general framework for performing such sampling with many different choices on the distribution of $\mathbf{s}_1, \mathbf{s}_2$. The framework is defined in terms of three component distributions that we call P_1, P_2 and P_y. Using these three distributions, we compare between an efficiently sampleable distribution which uses $\mathbf{R}$ and which we call the "real distribution", and an "ideal distribution" which is not efficiently sampleable but does not use $\mathbf{R}$. We present the ideal and real distributions in Figure 1. The ideal distribution directly samples $\mathbf{s}_2$ from the desired distribution P_2, and then samples $\mathbf{s}_1$ from some distribution P_1 conditioned on the fact that Equation (2) should be satisfied. Clearly, since $\mathbf{R}$ is never used in the sampling procedure of the ideal distribution, it does not depend on $\mathbf{R}$, but only on the publicly-available information.

The real distribution that our sampling algorithm samples from, first generates a $\mathbf{y}$ from an intermediate distribution P_y. This $\mathbf{y}$ will now deterministically determine both $\mathbf{s}_2$ and $\mathbf{s}_1$. To compute $\mathbf{s}_2$, we first compute $\mathbf{H}^{-1}(\mathbf{A}\mathbf{y} - \mathbf{t}) \bmod q \in \mathbb{Z}_q^n$, and then find an $\mathbf{s}_2$ in the support of P_2 such that $\mathbf{G}\mathbf{s}_2 = \mathbf{H}^{-1}(\mathbf{A}\mathbf{y} - \mathbf{t}) \bmod q$. By our choice of $\mathbf{G}$, this value of $\mathbf{s}_2$ will be unique and easily computable, and we denote it by $\mathbf{s}_2 = \mathbf{G}^{-1}(\mathbf{H}^{-1}(\mathbf{Ay}-\mathbf{t}) \bmod q)$.[4] We then compute $\mathbf{s}_1$ as $\mathbf{y}-\mathbf{R}\mathbf{s}_2$. At this point, the distribution of $(\mathbf{s}_1, \mathbf{s}_2)$ is not as in the ideal distribution. To correct the discrepancy, we use rejection sampling, and output $(\mathbf{s}_1, \mathbf{s}_2)$ with probability $\frac{P_1(\mathbf{s}_1)}{M \cdot P_y(\mathbf{s}_1+\mathbf{R}\mathbf{s}_2)}$ where M is some positive real (if this fraction is greater than 1, we define the probability to be 1).

In Section 3.1, we state the relationships between the matrices and the distributions that are required for our sampling algorithm to produce a distribution statistically close to the ideal distribution. Then in Section 3.2, we give two illustrative examples of instantiations over general lattices and polynomial rings.

3.1 Requirements and security proof

Theorem 3.1. *Consider matrices* $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, $\mathbf{R} \in \mathbb{Z}_q^{m\times l}$, $\mathbf{G} \in \mathbb{Z}_q^{n\times l}$, *and* $\mathbf{H} \in \mathbb{Z}_q^{n\times n}$, *and distributions* P_1, P_2, P_y *over* $\mathbb{Z}^m$, $\mathbb{Z}^l$, *and* $\mathbb{Z}^m$ *respectively, such that the following four conditions are satisfied:*

1. *For the two distributions* $\mathbf{s} \xleftarrow{\$} P_1$ *and* $\mathbf{s} \xleftarrow{\$} P_y$, *the statistical distance between* $\mathbf{As} \bmod q$ *and the uniform distribution over* $\mathbb{Z}_q^n$ *is at most* $2^{-(n \log q+\lambda)}$.
2. $\mathbf{H}$ *is invertible modulo* q.
3. *The function* $\mathbf{G}$ *mapping the support of* P_2 *to* $\mathbb{Z}_q^n$, *defined by* $\mathbf{G}(\mathbf{s}) = \mathbf{Gs}$, *is 1-to-1 and onto and is efficiently invertible via a function* $\mathbf{G}^{-1}$. *Furthermore,* P_2 *is uniformly random over its support.*
4. $\Pr_{(\mathbf{x}_1,\mathbf{x}_2)\xleftarrow{\$} IdealDistribution}\left[\frac{P_1(\mathbf{x}_1)}{P_y(\mathbf{x}_1+\mathbf{R}\mathbf{x}_2)} \le M\right] \ge 1 - 2^{-\lambda}$ *for some positive* $M \ge 1$.

then the outputs of the ideal distribution and the real distribution are ε*-close for* $\varepsilon = \lambda \cdot (2M + 1) \cdot 2^{-\lambda}$, *whenever* $\lambda > 4$. *Furthermore, the expected number of iterations of the sampling algorithm is* $\approx M$.

Proof. We first describe an intermediate distribution, which we will call the *hybrid distribution* defined as follows.

Hybrid Distribution without Rejection:

```
    Generate $\mathbf{s}_2 \xleftarrow{\$} P_2$
    Generate $\mathbf{y} \xleftarrow{\$} P_y|(\mathbf{Ay} = \mathbf{t} + \mathbf{HG}\mathbf{s}_2 \bmod q)$
    Compute $\mathbf{s}_1 \leftarrow \mathbf{y} - \mathbf{R}\mathbf{s}_2$
    Output $\mathbf{s} = (\mathbf{s}_1, \mathbf{s}_2)$
```

We also define a "Hybrid Distribution with Rejection" which first samples $(\mathbf{s}_1, \mathbf{s}_2)$ from the above-described Hybrid Distribution and then outputs it with probability $\frac{P_1(\mathbf{s}_1)}{M \cdot P_y(\mathbf{s}_1+\mathbf{R}\mathbf{s}_2)}$, else tries again.

[4] This is an abuse of notation since $\mathbf{G}^{-1}$ is not a matrix but rather a deterministic function satisfying $\mathbf{G} \cdot \mathbf{G}^{-1}(\mathbf{z}) = \mathbf{z}$.

Lemma 3.2. *Let f be the probability density function of the Ideal Distribution and g be the probability density function of the Hybrid Distribution (without rejection). Then*

$$\frac{f(\mathbf{x}_1,\mathbf{x}_2)}{g(\mathbf{x}_1,\mathbf{x}_2)} = \frac{P_1(\mathbf{x}_1)}{P_y(\mathbf{x}_1+\mathbf{R}\mathbf{x}_2)}(1+\delta) \text{ for some } \delta \ : \ -2\cdot 2^{-\lambda} \le \delta \le 3\cdot 2^{-\lambda}$$

In particular, for $\lambda > 4$, this means that the Ideal Distribution is $\lambda(M+1)2^{-\lambda}$ close to the Hybrid Distribution with Rejection.

Proof.

$$\begin{aligned}
\frac{f(\mathbf{x}_1,\mathbf{x}_2)}{g(\mathbf{x}_1,\mathbf{x}_2)} &= \frac{\Pr_{\mathbf{s}_2 \xleftarrow{\$} P_2}[\mathbf{s}_2 = \mathbf{x}_2]\cdot \Pr_{\mathbf{s}_1 \xleftarrow{\$} P_1}[\mathbf{s}_1 = \mathbf{x}_1 | \mathbf{A}\mathbf{s}_1 = \mathbf{t} + (\mathbf{HG}-\mathbf{AR})\mathbf{x}_2 \bmod q]}{\Pr_{\mathbf{s}_2 \xleftarrow{\$} P_2}[\mathbf{s}_2 = \mathbf{x}_2]\cdot \Pr_{\mathbf{y} \xleftarrow{\$} P_y}[\mathbf{y} = \mathbf{x}_1 + \mathbf{R}\mathbf{x}_2 | \mathbf{A}\mathbf{y} = \mathbf{t} + \mathbf{HG}\mathbf{x}_2 \bmod q]} \\
&= \frac{\Pr_{\mathbf{s}_1 \xleftarrow{\$} P_1}[\mathbf{s}_1 = \mathbf{x}_1]\cdot \Pr_{\mathbf{y} \xleftarrow{\$} P_y}[\mathbf{A}\mathbf{y} = \mathbf{t} + \mathbf{HG}\mathbf{x}_2 \bmod q]}{\Pr_{\mathbf{y} \xleftarrow{\$} P_y}[\mathbf{y} = \mathbf{x}_1 + \mathbf{R}\mathbf{x}_2]\cdot \Pr_{\mathbf{s}_1 \xleftarrow{\$} P_1}[\mathbf{A}\mathbf{s}_1 = \mathbf{t} + (\mathbf{HG}-\mathbf{AR})\mathbf{x}_2 \bmod q]} \\
&= \frac{P_1(\mathbf{x}_1)}{P_y(\mathbf{x}_1+\mathbf{R}\mathbf{x}_2)} \cdot \frac{\Pr_{\mathbf{y} \xleftarrow{\$} P_y}[\mathbf{A}\mathbf{y} = \mathbf{t} + \mathbf{HG}\mathbf{x}_2 \bmod q]}{\Pr_{\mathbf{s}_1 \xleftarrow{\$} P_1}[\mathbf{A}\mathbf{s}_1 = \mathbf{t} + (\mathbf{HG}-\mathbf{AR})\mathbf{x}_2 \bmod q]} \\
&= \frac{P_1(\mathbf{x}_1)}{P_y(\mathbf{x}_1+\mathbf{R}\mathbf{x}_2)} \cdot \frac{q^{-n}+\delta_1}{q^{-n}+\delta_y} \qquad \text{where } -q^{-n}2^{-\lambda} \le \delta_1, \delta_y \le q^{-n}2^{-\lambda} \qquad (3) \\
&= \frac{P_1(\mathbf{x}_1)}{P_y(\mathbf{x}_1+\mathbf{R}\mathbf{x}_2)}(1+\delta) \text{ where } -2\cdot 2^{-\lambda} \le \delta \le 3\cdot 2^{-\lambda}
\end{aligned}$$

Line 3 follows from the requirement of Theorem 3.1 that the distributions of $\mathbf{A}\mathbf{y} \bmod q$ and $\mathbf{A}\mathbf{s}_1 \bmod q$ are $q^{-n}2^{-\lambda}$-close to uniformly random over $\mathbb{Z}_q^n$ when $\mathbf{y} \xleftarrow{\$} P_y$ and $\mathbf{s}_1 \xleftarrow{\$} P_1$. This proves the first part of the lemma.

For the second part of the lemma, we define one more hybrid distribution, that we call the "Hybrid Distribution with Ideal Rejection". It is the same as the "Hybrid Distribution with Rejection", but we set the rejection probability to $\gamma_{ideal} = \min\left(\frac{f(\mathbf{s}_1,\mathbf{s}_2)}{M(1+\delta^+)g(\mathbf{s}_1,\mathbf{s}_2)}, 1\right)$ for $\delta^+ = 3\cdot 2^{-\lambda}$. This is instead of the rejection probability $\gamma_{real} = \min\left(\frac{P_1(\mathbf{s}_1)}{MP_y(\mathbf{s}_1+\mathbf{R}\mathbf{s}_2)}, 1\right)$ used in the original "Hybrid Distribution with Rejection" (in real life, we don't know γ_{ideal} exactly, and therefore are forced to use γ_{real} as an approximation). Note that by the first part of the lemma, $\gamma_{ideal} = \min\left(\frac{(1+\delta)P_1(\mathbf{s}_1)}{M(1+\delta^+)P_y(\mathbf{s}_1+\mathbf{R}\mathbf{s}_2)}, 1\right)$ for some $\delta \in [-2\cdot 2^{-\lambda}, \delta^+]$ and therefore $|\gamma_{real} - \gamma_{ideal}| \le 5(2^{-\lambda})$. Furthermore, in the "Hybrid Distribution with Ideal Rejection", the rejection sampling step only occurs at most λM times with probability $1 - \left(1 - \frac{1}{(1+\delta^+)M}\right)^{\lambda M} \ge 1 - 2^{-\lambda}$. Therefore, the statistical distance between the “Hybrid Distribution with Rejection” and the “Hybrid Distribution with Ideal Rejection” is at most $\lambda M \cdot 2^{-\lambda}$.

Next, we show that the “Hybrid Distribution with Ideal Rejection” is $\lambda \cdot 2^{-\lambda}$-statistically close to the ideal distribution. This relies on the rejection sampling lemma (Lemma 2.2). We note that by requirement (4) of the theorem: we have for $\delta^+ = 3 \cdot 2^{-\lambda}$:

$$\begin{aligned}\Pr_{(\mathbf{s}_1,\mathbf{s}_2)\overset{\$}{\leftarrow} f}\left[M(1+\delta^+)g(\mathbf{s}_1,\mathbf{s}_2) \geq f(\mathbf{s}_1,\mathbf{s}_2)\right] = & \Pr_{(\mathbf{s}_1,\mathbf{s}_2)\overset{\$}{\leftarrow} f}\left[\frac{f(\mathbf{s}_1,\mathbf{s}_2)}{g(\mathbf{s}_1,\mathbf{s}_2)} \leq M(1+\delta^+)\right] \\ \geq & \Pr_{(\mathbf{s}_1,\mathbf{s}_2)\overset{\$}{\leftarrow} f}\left[\frac{P_1(\mathbf{s}_1)}{P_y(\mathbf{s}_1+\mathbf{R}\mathbf{s}_2)} \leq M\right] \\ \geq & 1 - 2^{-\lambda}\end{aligned}$$

Therefore, by the “rejection sampling lemma” (Lemma 2.2), each iteration of the “Hybrid Distribution with Ideal Rejection” outputs something which is $2^{-\lambda}/M$ statistically close to outputting a sample from the ideal distribution with probability $\frac{1}{M(1+\delta^+)}$. Furthermore, the rejection sampling step only occurs at most λM times with probability $1 - \left(1 - \frac{1}{(1+\delta^+)M}\right)^{\lambda M} \geq 1 - 2^{-\lambda}$. Therefore, the statistical distance between the “Hybrid Distribution with Ideal Rejection” and the “Ideal Distribution” is at most $\lambda \cdot 2^{-\lambda}$.

Combining the above, the statistical distance between the “Hybrid Distribution with Rejection” and the “Ideal Distribution” is at most $\lambda M \cdot 2^{-\lambda} + \lambda \cdot 2^{-\lambda} = \lambda(M+1)2^{-\lambda}$ as claimed.

□

Now we introduce another distribution that we call the “Real Distribution without Rejection” which matches the real distribution without the rejection sampling step:

```
Real Distribution without Rejection:
    Generate $\mathbf{y} \overset{\$}{\leftarrow} P_y$
    Compute $\mathbf{s}_2 \leftarrow \mathbf{G}^{-1}\left(\mathbf{H}^{-1}(\mathbf{A}\mathbf{y} - \mathbf{t}) \bmod q\right)$
    Compute $\mathbf{s}_1 \leftarrow \mathbf{y} - \mathbf{R}\mathbf{s}_2$.
    Output $(\mathbf{s}_1, \mathbf{s}_2)$.
```

Lemma 3.3. *The statistical distance between the “Real Distribution without Rejection” and the “Hybrid Distribution without Rejection” is at most $2^{-\lambda}$. In particular, this also means that the statistical distance between the “Real Distribution with Rejection” and the “Hybrid Distribution with Rejection” is at most $\lambda M \cdot 2^{-\lambda}$.*

Proof. Let us define a randomized function $f(\mathbf{u})$ which gets as input $\mathbf{u} \in \mathbb{Z}_q^n$ and does the following:

Sample: $\mathbf{s}_2 \xleftarrow{\$} P_2 | \mathbf{t} + \mathbf{H}\mathbf{G}\mathbf{s}_2 = \mathbf{u} \bmod q$.
Sample: $\mathbf{y} \xleftarrow{\$} P_y | \mathbf{A}\mathbf{y} = \mathbf{u} \bmod q$.
Compute: $\mathbf{s}_1 \leftarrow \mathbf{y} - \mathbf{R}\mathbf{s}_2$.
Output: $(\mathbf{s}_1, \mathbf{s}_2)$.

It is easy to see that, when $\mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^n$, then $f(\mathbf{u})$ is equivalent to the "Hybrid Distribution". This is because the distribution of $\mathbf{t} + \mathbf{H}\mathbf{G}\mathbf{s}_2 \bmod q$ for $\mathbf{s}_2 \leftarrow P_2$ is indeed uniformly random (due to requirements (2) and (3) of Theorem 3.1).

On the other hand, if we instead sample $\mathbf{u}$ by choosing $\mathbf{y}' \xleftarrow{\$} P_y$ and setting $\mathbf{u} := \mathbf{A}\mathbf{y}' \bmod q$ then $f(\mathbf{u})$ is equivalent to the "Real Distribution without Rejection".

Therefore, the statistical distance between the hybrid and real distributions without rejection sampling is the statistical distance between $\mathbf{A}\mathbf{y}' \bmod q$: $\mathbf{y}' \xleftarrow{\$} P_y$ and the uniform distribution over $\mathbb{Z}_q^n$. By definition, this is at most $2^{-(n \log q + \lambda)} \leq 2^{-\lambda}$.

The distributions with rejection sampling just depend on at most λM copies of the corresponding distributions without rejection sampling with overwhelming probability $1 - 2^{-\lambda}$ (using the same argument as in the previous lemma) and therefore we can use the hybrid argument to argue that the statistical distance between them is at most $\lambda M \cdot 2^{-\lambda}$. □

Combining the above lemmas, proves the theorem. □

3.2 Two examples

We will now give two examples of matrices and distributions that satisfy the requirements of Theorem 3.1. Even though the examples are specific in the choices of parameters, we believe that they illustrate the techniques needed to apply our algorithm in other scenarios.

In both examples, we will let q be some prime[5], set $m = l = n\lceil \log q \rceil$, and define $\mathbf{G}$ as the matrix

$$\mathbf{G} = \begin{bmatrix} 1\ 2\ 4 \ldots 2^{\lceil \log q \rceil} & & & \\ & 1\ 2\ 4 \ldots 2^{\lceil \log q \rceil} & & \\ & & \ldots & \\ & & & 1\ 2\ 4 \ldots 2^{\lceil \log q \rceil} \end{bmatrix} \qquad (4)$$

Notice that with this $\mathbf{G}$, for every element $\mathbf{t} \in \mathbb{Z}^n$ with coefficients between 0 and $q-1$ there is a unique vector $\mathbf{s}_2 \in \{0,1\}^m$ such that $\mathbf{G}\mathbf{s}_2 = \mathbf{t}$ (without reduction modulo q). We denote this vector by $\mathbf{s}_2 = \mathbf{G}^{-1}(\mathbf{t})$, but note that this

[5] The requirement that q is prime only comes from the use of the leftover hash-lemma, and it can be relaxed. For example, it suffices that the smallest prime divisor of q is at least as large as $2\|\mathbf{s}\|_\infty$. Alternatively, if q is a product of primes and $\mathbf{s}$ has high entropy modulo each of the primes, then we can use the leftover-hash lemma for each prime divisor separately. For simplicity, we only mention these relaxations in passing and concentrate on the case when q is prime in what follows.

is an abuse of notation as the matrix $\mathbf{G}$ is not actually an invertible matrix. The distribution P_2 in our examples will simply be the distribution of $\mathbf{s}_2 = \mathbf{G}^{-1}(\mathbf{t})$ for a uniformly random $\mathbf{t} \in \mathbb{Z}^n$ having coefficients between 0 and $q-1$. Such a choice of $\mathbf{G}$ and P_2 satisfy requirement (3) of the theorem.

We will choose our matrix $\mathbf{A}$ at random from $\mathbb{Z}_q^{n\times m}$ and the distributions P_1, P_y to have min-entropy at least $3n\log q + 4\lambda$.[6] This ensures that, by the Leftover Hash Lemma (Lemma 2.1), the statistical distance between $(\mathbf{A}, \mathbf{A}\cdot\mathbf{s} \bmod q)$ and $(\mathbf{A}, \mathbf{r})$ where $\mathbf{s} \xleftarrow{\$} P_1$ (resp. P_y) and $\mathbf{r} \xleftarrow{\$} \mathbb{Z}_q^n$, is bounded by $2^{-(n\log q+2\lambda)}$. Let's say that a fixed matrix $\mathbf{A}$ is "good for P_1" (resp. "good for P_y") if the statistical distance between $\mathbf{A}\mathbf{s}$ and $\mathbf{r}$ is at most $\varepsilon = 2^{-n\log q+\lambda}$ when $\mathbf{s} \xleftarrow{\$} P_1$ (resp. $\mathbf{s} \xleftarrow{\$} P_y$). Then, by Markov's inequality, the probability that a random $\mathbf{A}$ is good for P_1 (resp. good for P_y) is at least $1-2^{-\lambda}$. Let's say that $\mathbf{A}$ is "good" if it is good for both P_1 and P_y. By the union bound, a random $\mathbf{A}$ is good with probability at least $1 - 2\cdot 2^{-\lambda}$. Therefore if the distributions P_1, P_y have min-entropy at least $3n\log q + 4\lambda$, requirement (1) of the theorem is satisfied with probability at least $1-2\cdot 2^{-\lambda}$.

In both examples, we will take the matrix $\mathbf{R}$ to be uniformly random from $\{-1,0,1\}^{m\times m}$. The important property we will need from $\mathbf{R}$ is that the norm of $\mathbf{R}\mathbf{s}_2$ is not too large.

By the Chernoff bound, we obtain that for any $\mathbf{s}_2 \in \{0,1\}^m$, there exists a $k_\infty = \Theta\left(\sqrt{\lambda m}\right)$ such that

$$\Pr_{\mathbf{R}\xleftarrow{\$}\{-1,0,1\}^{m\times m}}[\|\mathbf{R}\mathbf{s}_2\|_\infty \le k_\infty] \ge 1 - 2^{-2\lambda}. \tag{5}$$

For the distribution P_2 over $\{0,1\}^m$, we say that a fixed matrix $\mathbf{R}$ is ℓ_∞-good if $\Pr_{\mathbf{s}_2\xleftarrow{\$}P_2}[\|\mathbf{R}\mathbf{s}_2\|_\infty > k_\infty] \le 2^{-\lambda}$. By the above equation we have

$$\Pr_{\mathbf{s}_2\xleftarrow{\$}P_2, \mathbf{R}\xleftarrow{\$}\{-1,0,1\}^{m\times m}}[\|\mathbf{R}\mathbf{s}_2\|_\infty > k_\infty] \le 2^{-2\lambda}$$

and therefore by Markov inequality, a random $\mathbf{R}$ is ℓ_∞-good with probability $1-2^{-\lambda}$.

We can also establish a bound on the ℓ_2 norm of $\mathbf{R}\mathbf{s}_2$. By [Ver10, Theorem 5.39], for all $\mathbf{s}_2 \in \{0,1\}^m$, there exists a $k_2 = \Theta(m + \sqrt{\lambda m}) = \Theta(m)$ such that

$$\Pr_{\mathbf{R}\xleftarrow{\$}\{-1,0,1\}^{m\times m}}[\|\mathbf{R}\mathbf{s}_2\| \le k_2] \ge 1 - 2^{-2\lambda}. \tag{6}$$

For the distribution P_2 over $\{0,1\}^m$, we say that a fixed matrix $\mathbf{R}$ is ℓ_2-good if $\Pr_{\mathbf{s}_2\xleftarrow{\$}P_2}[\|\mathbf{R}\mathbf{s}_2\| > k_2] \le 2^{-\lambda}$. By the same reasoning as above, a random $\mathbf{R}$ is ℓ_2-good with probability $1-2^{-\lambda}$.

[6] This entropy lower bound is not really a restriction on the distributions P_1 and P_y. The distributions that we will need to pick to satisfy property (4) of the theorem will easily meet this bound.

We now proceed to show how to pick the distributions P_1 and P_y to satisfy the requirement (4) of the theorem. We will assume that the randomly-chosen $\mathbf{A}$ and $\mathbf{R}$ are good for these distributions (as defined above), which happens with overwhelming probability $1 - 4 \cdot 2^{-\lambda}$. In our first example, both P_1 and P_y will be uniform distributions in some cube. The advantage of such distributions is that they are particularly easy to sample. Our second example will have both of these distributions be discrete Gaussians over $\mathbb{Z}^m$. The advantage of using discrete Gaussians rather than the uniform distribution is that the norm of $\mathbf{s}_1$ will end up being smaller. The disadvantage is that sampling the discrete Gaussian distribution over $\mathbb{Z}^m$ is a more involved procedure than sampling the uniform distribution over $\mathbb{Z}_q$. Still, sampling a discrete Gaussian over $\mathbb{Z}^m$ is more efficient and requires less precision than sampling such a distribution over an arbitrary lattice.

Example for P_1 and P_y being uniform in an m-dimensional integer cube. We define the distribution P_1 (respectively P_y) to be the uniform distribution over all vectors $\mathbf{x} \in \mathbb{Z}^m$ such that $\|\mathbf{x}\|_\infty \leq mk_\infty$ (respectively $\|\mathbf{x}\|_\infty \leq mk_\infty + k_\infty$). And we set the constant

$$M = \left(\frac{2mk_\infty + 2k_\infty + 1}{2mk_\infty + 1}\right)^m \approx e. \tag{7}$$

We will now show that the above choices satisfy the necessary requirements of Theorem 3.1. First, we will lower-bound the entropies of P_1 and P_y.

$$\begin{aligned} H_\infty(P_y) > H_\infty(P_1) &= -\log\left(\frac{1}{(2mk_\infty + 1)^m}\right) \\ &> m \log m > n \log n \log q > 3n \log q + 4\lambda, \end{aligned}$$

and so the first requirement of the theorem is satisfied.

We will conclude by showing that requirement (4) of the theorem is also satisfied. First, it's easy to see that for any $\mathbf{x}_1$ output by the ideal distribution, $P_1(\mathbf{x}_1) = \frac{1}{(2mk_\infty+1)^m}$. Additionally, for any $\mathbf{x}_1$ in the support of P_1, if $\|\mathbf{R}\mathbf{x}_2\|_\infty \leq k_\infty$, then $\mathbf{x}_1 + \mathbf{R}\mathbf{x}_2$ is in the support of P_2, and so $P_y(\mathbf{x}_1 + \mathbf{R}\mathbf{x}_2) = \frac{1}{(2mk_\infty+2k_\infty+1)^m}$. Therefore if $\|\mathbf{R}\mathbf{x}_2\|_\infty \leq k_\infty$, we have

$$\frac{P_1(\mathbf{x}_1)}{P_y(\mathbf{x}_1 + \mathbf{R}\mathbf{x}_2)} = \left(\frac{2mk_\infty + 2k_\infty + 1}{2mk_\infty + 1}\right)^m < \left(1 + \frac{1}{m}\right)^m < e, \tag{8}$$

and so if we let f be the ideal distribution, then

$$\Pr_{(\mathbf{x}_1,\mathbf{x}_2) \overset{\$}{\leftarrow} f}\left[\frac{P_1(\mathbf{x}_1)}{P_y(\mathbf{x}_1 + \mathbf{R}\mathbf{x}_2)} \leq M\right] \geq \Pr_{(\mathbf{x}_1,\mathbf{x}_2) \overset{\$}{\leftarrow} f}\left[\|\mathbf{R}\mathbf{x}_2\|_\infty \leq k_\infty\right] \geq 1 - 2^{-\lambda},$$

where the last inequality follows by our choice of k_∞.

Note that since $\mathbf{s}_1$ is chosen to have coordinates of size $mk_\infty = \Theta(m^{1.5}\sqrt{\lambda})$, we have $\|\mathbf{s}\| \approx \|\mathbf{s}_1\| = \Theta(m^2\sqrt{\lambda})$. We also point out that the rejection sampling part of our sampler in Figure 1 is actually very simple and one does not in fact need to compute any probability distributions or even the value of M in equation (7) – simply looking at the infinity norm of $\mathbf{s}_1$ is enough. If $\mathbf{s}_1$ (in line 3) is outside the support of P_1 (i.e. $\|\mathbf{s}_1\|_\infty > mk_\infty$), then $P_1(\mathbf{s}_1) = 0$ and we always reject. On the other hand, if $\|\mathbf{s}_1\|_\infty \le mk_\infty$, then $\frac{P_1(\mathbf{x}_1)}{MP_y(\mathbf{x}_1+\mathbf{R}\mathbf{x}_2)} = 1$ (by (7) and (8)), and we always accept.

Example for P_1 and P_y being discrete Gaussians over $\mathbb{Z}^m$. The discrete Gaussian distribution with standard deviation σ over $\mathbb{Z}^m$ is defined as

$$D^m_\sigma(\mathbf{x}) = \frac{e^{-\|\mathbf{x}\|^2/2\sigma^2}}{\sum\limits_{\mathbf{v}\in\mathbb{Z}^m} e^{-\|\mathbf{v}\|^2/2\sigma^2}}.$$

In this example, we will define both P_1 and P_y to be distributions D^m_σ for $\sigma = 2k_2\sqrt{\lambda}$, and we set the constant $M = e^{1+1/8\lambda}$. We will first lower-bound the min-entropy of P_1. Notice that the heaviest element of the distribution is $\mathbf{0}$, and from the proof of [Lyu12, Lemma 4.4], we have $\sum\limits_{\mathbf{v}\in\mathbb{Z}^m} e^{-\|\mathbf{v}\|^2/2\sigma^2} > (\sqrt{2\pi}\sigma - 1)^m$. Thus,

$$H_\infty(P_1) = -\log\left(D^m_\sigma(\mathbf{0})\right) = -\log\left(\frac{1}{\sum\limits_{\mathbf{v}\in\mathbb{Z}^m} e^{-\|\mathbf{v}\|^2/2\sigma^2}}\right) > m\log\sigma > 3n\log q + 4\lambda.$$

We will now move on to prove that requirement (4) of Theorem 3.1 is also satisfied. First, we write

$$\frac{P_1(\mathbf{x}_1)}{P_y(\mathbf{x}_1+\mathbf{R}\mathbf{x}_2)} = \frac{e^{-\|\mathbf{x}_1\|^2/2\sigma^2}}{e^{-\|\mathbf{x}_1+\mathbf{R}\mathbf{x}_2\|^2/2\sigma^2}} = e^{(2\langle\mathbf{x}_1,\mathbf{R}\mathbf{x}_2\rangle + \|\mathbf{R}\mathbf{x}_2\|^2)/2\sigma^2} \le e^{\frac{\langle\mathbf{x}_1,\mathbf{R}\mathbf{x}_2\rangle}{\sigma^2}+1/8\lambda},$$

where the last inequality follows from our assumption that the random $\mathbf{R}$ satisfies the condition in Equation (6).

We now would like to upper-bound the above quantity when $\mathbf{x}_1, \mathbf{x}_2$ are distributed according to the ideal distribution. If we let $\mathbf{t}' = \mathbf{t} + (\mathbf{HG} - \mathbf{AR})\mathbf{x}_2$, then the probability that the absolute value of the dot product $\langle\mathbf{x}_1, \mathbf{R}\mathbf{x}_2\rangle$ is less than some arbitrary positive real r is

$$\begin{aligned}
&\Pr_{\mathbf{x}_1 \xleftarrow{\$} P_1}\left[|\langle\mathbf{x}_1,\mathbf{R}\mathbf{x}_2\rangle| \le r \,|\, \mathbf{A}\mathbf{x}_1 = \mathbf{t}'\right] \\
&\qquad = \frac{\Pr\limits_{\mathbf{x}_1 \xleftarrow{\$} P_1}\left[\mathbf{A}\mathbf{x}_1 = \mathbf{t}' \mid |\langle\mathbf{x}_1,\mathbf{R}\mathbf{x}_2\rangle| \le r\right] \cdot \Pr\limits_{\mathbf{x}_1 \xleftarrow{\$} P_1}\left[|\langle\mathbf{x}_1,\mathbf{R}\mathbf{x}_2\rangle| \le r\right]}{\Pr\limits_{\mathbf{x}_1 \xleftarrow{\$} P_1}\left[\mathbf{A}\mathbf{x}_1 = \mathbf{t}'\right]}
\end{aligned}$$

By [Lyu12, Lemma 4.3], we have that for $r = 2k_2\sigma\sqrt{\lambda}$,

$$\Pr_{\mathbf{x}_1 \overset{\$}{\leftarrow} P_1} [|\langle \mathbf{x}_1, \mathbf{R}\mathbf{x}_2 \rangle| \le r] > 1 - 2^{-\lambda}.$$

Furthermore,

$$\begin{aligned} H_\infty(P_1 \mid \ |\langle \mathbf{x}_1, \mathbf{R}\mathbf{x}_2 \rangle| \le r) &\ge H_\infty(P_1) - \log(1 - 2^{-\lambda}) \\ &> H_\infty(P_1) - 1 \\ &> m \log \sigma \\ &> 3n \log q + 4\lambda, \end{aligned}$$

which allows us to apply the Leftover Hash Lemma (Lemma 2.1) to conclude that

$$\Pr_{\mathbf{x}_1 \overset{\$}{\leftarrow} P_1} [|\langle \mathbf{x}_1, \mathbf{R}\mathbf{x}_2 \rangle| \le r | \mathbf{A}\mathbf{x}_1 = \mathbf{t}'] \ge \frac{q^{-n} - \delta_1}{q^{-n} + \delta_1} \cdot (1 - 2^{-\lambda}) \ge (1 - \delta)(1 - 2^{-\lambda})$$

where $\delta_1 \le q^{-n}2^{-\lambda}$ and $\delta < 3 \cdot 2^{-\lambda}$. If we let f be the ideal distribution, then putting everything together, we obtain that

$$\begin{aligned} &\Pr_{(\mathbf{x}_1, \mathbf{x}_2) \overset{\$}{\leftarrow} f} \left[\frac{P_1(\mathbf{x}_1)}{P_y(\mathbf{x}_1 + \mathbf{R}\mathbf{x}_2)} \le e^{1+1/8\lambda} \right] \\ &\ge \Pr_{(\mathbf{x}_1, \mathbf{x}_2) \overset{\$}{\leftarrow} f} \left[\frac{2\langle \mathbf{x}_1, \mathbf{R}\mathbf{x}_2 \rangle + \|\mathbf{R}\mathbf{x}_2\|^2}{2\sigma^2} \le 1 + 1/8\lambda \right] \\ &= \Pr_{\mathbf{x}_1 \overset{\$}{\leftarrow} P_1} [|\langle \mathbf{x}_1, \mathbf{R}\mathbf{x}_2 \rangle| \le 2k_2\sigma\sqrt{\lambda} | \mathbf{A}\mathbf{x}_1 = \mathbf{t}'] \\ &\ge (1 - 3 \cdot 2^{-\lambda}) \cdot (1 - \lambda) > 1 - 4 \cdot 2^{-\lambda}. \end{aligned}$$

Since $\mathbf{s}_1$ is chosen from D_σ^m for $\sigma = 2k\sqrt{\lambda} = \Theta(m\sqrt{\lambda})$, the norm of $\mathbf{s}_1$ is tightly concentrated around $\Theta(m^{1.5}\sqrt{\lambda})$ [Ban93]. Therefore choosing the distribution P_1 to be a discrete Gaussian rather than uniform (as in the previous example), allowed us to keep the distribution P_2 of $\mathbf{s}_2$ exactly the same, while reducing the expected length of the vector $\mathbf{s}_1$.

Sampling over polynomial rings. Just like the sampler of [MP12], ours also naturally extends to sampling vectors over polynomial rings $R = \mathbb{Z}[x]/(f(x))$, where $f(x)$ is a monic polynomial with integer coefficients. This allows the sampler to be used in constructions of more efficient lattice primitives based on the hardness of Ring-SIS [PR06,LM06] and Ring-LWE [LPR13a].

For sampling over polynomial rings, one can keep all the notation exactly the same, simply taking care that all additions and multiplications that were done over the rings $\mathbb{Z}$ and $\mathbb{Z}_q$ are now done over the rings $\mathbb{Z}[x]/(f(x))$ and $\mathbb{Z}_q[x]/(f(x))$. The only thing to be careful about is the application of the leftover hash lemma for satisfying part (1) of Theorem 3.1. If the ring is a field (i.e.

$f(x)$ is irreducible over $\mathbb{Z}_q$), then everything is very simple because the function mapping $\mathbf{s}$ to $\mathbf{As}$ is still universal. If, on the other hand, $f(x)$ does split, then the function becomes an *almost* universal hash function whose universality may degrade with the number of terms into which $f(x)$ splits. In particular, if $f(x)$ splits into many terms, then it may in fact be impossible to reach the necessary statistical distance for satisfying condition (1), and one will instead need to use different distributions and leftover hash lemmas, (cf. [Mic07, Theorem 4.2], [SS11, Theorem 2], [LPR13b, Theorem 7.4]).

4 Acknowledgements

We thank the anonymous reviewers for their useful suggestions.

References

[ABB10] Shweta Agrawal, Dan Boneh, and Xavier Boyen. Efficient lattice (H)IBE in the standard model. In *EUROCRYPT*, pages 553–572, 2010.

[Ajt96] Miklós Ajtai. Generating hard instances of lattice problems (extended abstract). In *STOC*, pages 99–108, 1996.

[Ajt99] Miklós Ajtai. Generating hard instances of the short basis problem. In *ICALP*, pages 1–9, 1999.

[Ban93] Wojciech Banaszczyk. New bounds in some transference theorems in the geometry of numbers. *Mathematische Annalen*, 296:625635, 1993.

[DDLL13] Léo Ducas, Alain Durmus, Tancrède Lepoint, and Vadim Lyubashevsky. Lattice signatures and bimodal gaussians. In *CRYPTO (1)*, pages 40–56, 2013.

[DG14] Nagarjun C. Dwarakanath and Steven D. Galbraith. Sampling from discrete gaussians for lattice-based cryptography on a constrained device. *Appl. Algebra Eng. Commun. Comput.*, 25(3):159–180, 2014.

[GGH97] Oded Goldreich, Shafi Goldwasser, and Shai Halevi. Public-key cryptosystems from lattice reduction problems. In *CRYPTO*, pages 112–131, 1997.

[GLP12] Tim Güneysu, Vadim Lyubashevsky, and Thomas Pöppelmann. Practical lattice-based cryptography: A signature scheme for embedded systems. In *CHES*, pages 530–547, 2012.

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *STOC*, pages 197–206, 2008.

[HHGP$^+$03] Jeffrey Hoffstein, Nick Howgrave-Graham, Jill Pipher, Joseph H. Silverman, and William Whyte. Ntrusign: Digital signatures using the ntru lattice. In *CT-RSA*, pages 122–140, 2003.

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM J. Comput.*, 28(4):1364–1396, 1999.

[LM06] Vadim Lyubashevsky and Daniele Micciancio. Generalized compact knapsacks are collision resistant. In *ICALP (2)*, pages 144–155, 2006.

[LPR13a] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. *J. ACM*, 60(6):43, 2013. Preliminary version appeared in EUROCRYPT 2010.

[LPR13b] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. A toolkit for ring-lwe cryptography. In *EUROCRYPT*, pages 35–54, 2013.

[Lyu09] Vadim Lyubashevsky. Fiat-Shamir with aborts: Applications to lattice and factoring-based signatures. In *ASIACRYPT*, pages 598–616, 2009.

[Lyu12] Vadim Lyubashevsky. Lattice signatures without trapdoors. In *EUROCRYPT*, pages 738–755, 2012.

[Mic07] Daniele Micciancio. Generalized compact knapsacks, cyclic lattices, and efficient one-way functions. *Computational Complexity*, 16(4):365–411, 2007.

[MP12] Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In *EUROCRYPT*, pages 700–718, 2012.

[NR09] Phong Q. Nguyen and Oded Regev. Learning a parallelepiped: Cryptanalysis of GGH and NTRU signatures. *J. Cryptology*, 22(2):139–160, 2009.

[NZ96] Noam Nisan and David Zuckerman. Randomness is linear in space. *J. Comput. Syst. Sci.*, 52(1):43–52, 1996.

[PDG14] Thomas Pöppelmann, Léo Ducas, and Tim Güneysu. Enhanced lattice-based signatures on reconfigurable hardware. In *CHES*, pages 353–370, 2014.

[Pei10] Chris Peikert. An efficient and parallel gaussian sampler for lattices. In *CRYPTO*, pages 80–97, 2010.

[PR06] Chris Peikert and Alon Rosen. Efficient collision-resistant hashing from worst-case assumptions on cyclic lattices. In *TCC*, pages 145–166, 2006.

[SS11] Damien Stehlé and Ron Steinfeld. Making NTRU as secure as worst-case problems over ideal lattices. In *EUROCRYPT*, pages 27–47, 2011.

[Ver10] Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. *CoRR*, abs/1011.3027, 2010.