Ring-LWE

Noah Stephens-Davidowitz
(for Vinod Vaikuntanathan's class)

1 Ring-LWE basics and some properties of $\mathbb{Z}[x]/(x^n+1)$

1.1 From Ring-SIS to Ring-LWE

In the previous lecture, we introduced the polynomial ring $R := \mathbb{Z}[x]/(x^n+1)$ for n a power of two. We then replaced the SIS problem, which was to find non-zero $\boldsymbol{e} \in \{-1,0,1\}^m$ such that $A\boldsymbol{e} = \boldsymbol{0} \bmod q$ for random $A \in \mathbb{Z}_q^{n\times m}$, by the Ring-SIS problem, which asks us to find $e_1,\ldots,e_\ell \in R_{\{-1,0,1\}}$ not all zero such that $a_1e_1+\cdots+a_\ell e_\ell = 0 \bmod qR$ for random $a_i \in R_q$. Here, $R_{\{-1,0,1\}}$ is the set of polynomials in R with coefficients in $\{-1,0,1\}$ and $R_q := R/(qR)$ is the ring R with coefficients reduced modulo q.

We built a more efficient collision-resistant hash function whose security is equivalent to Ring-SIS. And, we related this security to a worst-case problem on ideals over R—i.e., additive subgroups $\mathcal{I} \subseteq R$ that are closed under multiplication by x. In particular, we defined the norm of a ring element as the ℓ_2 norm of its coefficient vector, and we defined γ-IdealSVP, which is just γ-SVP restricted to ideal lattices. We then showed that γ-IdealSVP reduces to Ring-SIS for appropriate parameters.

Now, we bring these ideas from the SIS setting to the LWE setting. In particular, the problem (search) LWE asks us to find $\boldsymbol{s} \in \mathbb{Z}_q^n$ given $(A, \boldsymbol{s}^T A + \boldsymbol{e}^T \bmod q)$, where $A \in \mathbb{Z}_q^{n\times m}$ and $\boldsymbol{s} \in \mathbb{Z}_q^n$ are uniformly random and $\boldsymbol{e} \in \mathbb{Z}^m$ is chosen from some error distribution on short vectors. We will define Ring-LWE in a similarly natural way. We will see that the hardness of Ring-LWE implies more efficient public-key cryptography, and that this hardness can be based on the worst-case hardness of the worst-case ideal lattice problem γ-IdealBDD (which we will define later). Because we will rely very heavily on special properties of our specific ring $R = \mathbb{Z}[x]/(x^n+1)$ for n a power of two, we only define Ring-LWE over this specific ring. Everything presented here can be generalized, but doing so requires quite a bit more work [LPR10].[1]

Definition 1. *For integers $\ell, q \geq 2$, power of two n, and an error distribution χ over short elements in R, the (average-case, search) Ring-LWE problem is defined as follows. The input is $a_1,\ldots,a_\ell \in R_q$ sampled independently and uniformly at random together with $b_1,\ldots,b_n \in R_q$, where $b_i := a_i \cdot s + e_i \bmod qR$ for $s \in R_q$, and $e_i \sim \chi$. The goal is to output s.*

Notice that we take s to be worst-case, rather than uniformly random. This is without loss of generality, since we can trivially randomize s if necessary. Just like before, we will also need the

[1] The "right" notion of Ring-LWE for more general rings has a more sophisticated definition based on the canonical embedding of a number field. In particular, the naive coefficient embedding in which the norm of a ring element is just the norm of its coefficient vector does not behave nicely for general rings. In the special case when $R = \mathbb{Z}[x]/(x^n+1)$ for n a power of two, the canonical embedding and coefficient embedding are identical (up to scaling and rotation), so we can largely ignore these issues.

decisional version of the problem, which asks us to distinguish the (a_i, b_i) from uniformly random and independent elements of R_q.

1.2 Basic properties

Ring-LWE inherits many of LWE's nice properties. In particular, Ring-LWE is equivalent to the planted variant of Ring-SIS, and the hardness of Ring-LWE (both search and decision) remains unchanged if we sample the secret s from the error distribution χ (at the expense of one sample). One can prove both of these facts in more-or-less the same way that we proved the corresponding facts for plain LWE, at least for appropriate choices of q.

For example, given ℓ Ring-LWE samples $(a_1, b_1), \ldots, (a_\ell, b_\ell)$ with $b_i := a_i s + e_i$, we can try to convert them into $\ell - 1$ Ring-LWE samples with the secret sampled from the error distribution as follows. We assume that one of the a_i is invertible in R_q (i.e., there exists an element $a_i^{-1} \in R_q$ such that $a_i a_i^{-1} = 1$, which happens with non-negligible probability, as shown in [LPR13, Claim 2.25]). Then, $a_j a_i^{-1} b_i = a_j s + a_j a_i^{-1} e_i$, and $a_j a_i^{-1} b_i - b_j = a_j a_i^{-1} e_i + e_j$. We can therefore create the new samples $(a_j a_i^{-1}, a_j a_i^{-1} b_i - b_j)$ for all $j \neq i$, which are $\ell - 1$ valid Ring-LWE samples with secret e_i and error e_j, as needed.

1.3 Encryption

Recall that we saw both a secret-key encryption scheme and a public-key encryption scheme from plain LWE. Both of these schemes have natural analogues in the Ring-LWE world. Just like our Ring-SIS-based hash function, these schemes are remarkably efficient.

The secret-key encryption scheme is as follows. Both this scheme and the public-key scheme naturally use $R_{\{0,1\}}$ as their message space, i.e., polynomials with $\{0, 1\}$ coefficients. (Compare this to the one-bit message space that we obtained for LWE.)

- **Key generation:** The secret key is simply a uniformly random element $s \in R_q$.
- **Encryption:** To encrypt $m \in R_{\{0,1\}}$, compute (a, b) for $b := a \cdot s + e + \lfloor q/2 \rceil \cdot m \bmod qR$, where $a \in R_q$ is chosen uniformly at random and $e \sim \chi$.
- **Decryption:** To decrypt (a, b), compute $b - a \cdot s \bmod qR = \lfloor q/2 \rceil \cdot m + e \bmod qR$. Round each coefficient to either $q/2$ or zero, whichever is closest (where we assume that our representation modulo qR uses coefficients in $[q]$), and interpret 0 as 0 and $q/2$ as 1.

Clearly, this scheme is correct if and only if the coefficients of e are smaller than roughly $q/4$. Furthermore, the CPA-security of the scheme is immediate from Ring-LWE. And, this scheme is quite efficient, encrypting n-bit messages using roughly $n \log q$-bit ciphertexts, with encryption and decryption in time $n \cdot \operatorname{poly}\log(n, q)$. As far as we know, this scheme is $2^{\Omega(n)}$ secure for appropriate parameters, so that we may take n only linear in the security parameter.

The public-key encryption scheme is as follows.

- **Key generation:** The secret key is a short secret $s \sim \chi$. The public key is $(\widehat{a}, y)$ for $\widehat{a} \in R_q$ uniformly random and $y := \widehat{a} \cdot s + e \bmod qR$, where $e \sim \chi$.
- **Encryption:** To encrypt $m \in R_{\{0,1\}}$, compute (a, b), where $a := \widehat{a}r + x \bmod q$ and $b := yr + x' + \lfloor q/2 \rceil m \bmod q$ for $r, x, x' \sim \chi$.

- **Decryption:** To decrypt (a, b), compute $b - a \cdot s \bmod qR = \lfloor q/2 \rceil m + er + x' - xs \bmod q$ and again do our rounding procedure to find m.

Clearly, this scheme is correct if and only if $er + x' - xs$ is less than $q/4$. (So, we can take our error to have size roughly $\sqrt{q}/2$.) Security follows from a proof similar to the one for plain LWE in our first lecture. I.e., we use the hardness of decisional Ring-LWE with short secrets once to show that the public key can be replaced by uniformly random ring elements and then again to show that the element b in the ciphertext can also be replaced by a uniformly random ring element.

Again, we note the remarkable efficiency of this scheme. As far as we know, it is $2^{\Omega(n)}$ secure and all operations are computable in time $n \cdot \operatorname{poly}\log(n, q)$. Taking $q = \operatorname{poly}(n)$ gives a public-key encryption scheme with key generation, encryption, and decryption all computable in time quasilinear in the security parameter. And, Lyubashevsky, Peikert, and Regev proved that breaking this scheme is at least as hard as a certain worst-case ideal lattice problem [LPR10]—even an ideal lattice problem that is plausibly $2^{\Omega(n)}$ hard. (We will only prove a weaker worst-case to average-case reduction, with a worst-case problem solvable in $2^{o(n)}$ time and an exponential modulus q.)

1.4 Reduction modulo ideals and Chinese Remainder Theorem

Recall that for an element $r \in R$ in some ring R (e.g., $R = \mathbb{Z}$), we define equivalence of $s_1, s_2 \in R$ modulo r by $s_1 = s_2 \bmod r$ if and only if there exists an $r' \in R$ with $s_1 = s_2 + r'r$. Equivalently, $s_1 = s_2 \bmod r$ if and only if there exists an *ideal* element $y \in rR := \{r' \cdot r \ : \ r' \in R\}$ in the ideal rR generated by r such that $s_1 = s_2 + y$. This is an equivalence relation because the ideal is closed under addition, which also implies that it respects addition. It respects multiplication because the ideal is closed under multiplication by any ring element. I.e., if $s_1 = s_2 + y$ for $y \in rR$, then $xs_1 = xs_2 + xy$, which implies that $xs_1 = xs_2 \bmod r$, since $xy \in rR$ also.

This immediately shows that we can also reduce modulo an arbitrary ideal $\mathcal{I}$, not just an ideal generated by a single element. I.e., we define $s_1 = s_2 \bmod \mathcal{I}$ if and only if there exists $y \in \mathcal{I}$ such that $s_1 = s_2 + y$. (This is a big part of the reason why ideals are such important objects in the study of rings, as opposed to, say, subrings.) Just like before, addition and multiplication are well defined modulo $\mathcal{I}$, and we write $R/\mathcal{I}$ for the ring of equivalence classes modulo $\mathcal{I}$.[2]

We will need something slightly more general. For an ideal $\mathcal{J} \subseteq \mathcal{I}$ (e.g, $\mathcal{J} = q\mathcal{I}$), we can again define the quotient $\mathcal{I}/\mathcal{J}$. This quotient is also a ring, and we can define multiplication by x in $\mathcal{I}/\mathcal{J}$ in the obvious way.

We can now present the Chinese Remainder Theorem over R. (A far more general theorem holds here over a very large class of rings.) We say that two ideals $\mathcal{I}$ and $\mathcal{J}$ are *coprime* if there exists $y \in \mathcal{I}, z \in \mathcal{J}$ such that $y + z = 1$.

Theorem 1.1 (Chinese Remainder Theorem for R)**.** *For any pairwise coprime ideals $\mathcal{I}_1, \ldots, \mathcal{I}_k \subseteq R$ over R, let $\mathcal{I} := \bigcap \mathcal{I}_j$. Then, $R/\mathcal{I}$ is isomorphic (as a ring) to the direct product*

$$\frac{R}{\mathcal{I}_1} \times \frac{R}{\mathcal{I}_2} \times \cdots \times \frac{R}{\mathcal{I}_k} \ .$$

Indeed, an isomorphism is given by the natural map

$$r \mapsto (r \bmod \mathcal{I}_1, r \bmod \mathcal{I}_2, \ldots, r \bmod \mathcal{I}_k) \ ,$$

[2] In fact, we have already been sneakily using this notation, writing $\bmod qR$, rather than $\bmod q$.

and it can be efficiently inverted.

Furthermore, in the special case when $\mathcal{I} = qR$ for a prime q, we may take the $\mathcal{I}_j$ to be the ideal generated by q and the jth irreducible factor of $x^n + 1$ modulo q. Then, the quotients $R/\mathcal{I}_j$ are actually fields of characteristic q.

In particular, turning back to the question of invertibility of a_i from Section 1.2, we see that at least for prime q, $a \in R_q$ is invertible unless $a = 0 \bmod \mathcal{I}_j$ for some j (since the quotients are fields and therefore do not have zero divisors). Since the quotient has size at least q, this happens with probability at most $1/q$. Because of the product structure guaranteed by the Chinese Remainder Theorem, we then see that a is invertible with probability at least $(1 - 1/q)^n$.

2 Search to decision

We will prove the following search-to-decision reduction for Ring-LWE, which was originally proven by Lyubashevsky, Peikert, and Regev [LPR10]. We say that a polynomial *splits mod q* if it is the product of distinct linear factors modulo q. We say that an error distribution χ over R is *spherically symmetric* if the probability of sampling a ring element from χ depends only on its norm.

Theorem 2.1. *For a prime $q \geq 2$, integer $\ell \geq 2$, power of two n such that $x^n + 1$ splits mod q, and spherically symmetric error distribution χ, there is a reduction from search Ring-LWE to decision Ring-LWE that runs in time $q \cdot \mathrm{poly}(n, \ell)$*

Many new issues arise in the ring setting. Therefore, the proof is quite a bit more difficult than the relatively easy proof for plain LWE. Indeed, lurking behind this reduction is quite a bit of Galois theory. (We refer the reader to [LPR10] for a much more thorough discussion.) Furthermore, the result is not entirely satisfying for at least two reasons.

First, the running time proportional to q is unfortunate, since our worst-case to average-case reduction will only work for exponentially large moduli q. Recall that we had this issue in the plain LWE case as well, but there we mentioned that modulus-switching techniques can be used to reduce exponential q to polynomial q (with a large loss in parameters) [BLP+13]. However, nothing similar is known in the Ring-LWE setting. Indeed, the only hardness results known for Ring-LWE with small q use a quantum reduction [LPR10, PRS17], which we will not present here.

Second, the fact that our polynomial $x^n + 1$ splits mod q is a bit worrisome, since we saw an attack on Ring-SIS when the polynomial modulus has a high-degree factor with small coefficients over the integers. *That* attack does extend to Ring-LWE, but as far as we know, there is no attack that exploits a modulus q over which $x^n + 1$ factors. In particular, though we will factor $x^n + 1$ modulo q, we will not find high-degree factors with small coefficients. Indeed, the worst-case to average-case reduction in [LPR10] shows that, if Ideal-SVP with appropriate parameters is hard for a quantum computer, then Ring-LWE is also hard for any sufficiently large modulus q, regardless of whether $x^n + 1$ splits modulo q (and we will prove a weaker version of this result).

Remark. *Actually, for worst-case hardness, we can dispense with search-to-decision reductions entirely. In [PRS17], we showed with Peikert and Regev that the worst-case to average-case reduction for Ring-LWE (which we will see below) can be modified slightly to "go straight to decision." So, while we do not quite show that decision is necessarily as hard as search, we do show worst-case hardness for decision that is just as strong as the corresponding results for search. Of course, the reduction in [PRS17] has its own challenges, so we do not present it here.*

2.1 Where we're going

Recall that our search-to-decision reduction for plain LWE worked by guessing the coordinates of the secret vector $\boldsymbol{s} \in \mathbb{Z}_q^n$ one by one. One might therefore hope to find a similar reduction for Ring-LWE that works by guessing the *coefficients* of the secret ring element $s \in R_q$ one by one. However, it is not at all clear how to do this. In the plain LWE case, we crucially used the fact that knowing a coordinate of $\boldsymbol{s}$ allows us to compute $\langle \boldsymbol{a}, \boldsymbol{s} \rangle \bmod q$ for some $\boldsymbol{a} \in \mathbb{Z}_q^n$. (Namely, the standard basis vector corresponding to the relevant coordinate.) However, knowing just one coefficient of s (or even $n-1$ coefficients of s) does not allow us to compute $a \cdot s \bmod qR$ for any non-zero $a \in R_q$.

We will need to develop a few tools in the next few subsections to correct this. The high-level structure is as follows. First, we show how to use a different coordinate system, based on the Chinese Remainder Theorem, to make multiplication coordinate-wise. This is nice because it allows us to guess a coordinate in a meaningful way. However, when we guess wrong, we will not end up with uniformly random samples. Instead, we will get Ring-LWE samples that are uniform in just one coordinate, and it is not immediately clear how to use a decision oracle to distinguish these two cases. In order to get around this, we will show the existence of very special functions that essentially allow us to "swap" coordinates. Finally, we will use a hybrid argument together with these tools to prove that hardness of search Ring-LWE implies hardness of decision Ring-LWE.

2.2 The CRT embedding (which is very different from the coefficient embedding!)

Our first task is to find a coordinate system in which multiplication is coordinate-wise. E.g., in these coordinates, the product of $(s_1, s_2, \ldots, s_n)$ with $(1, 0, 0, \ldots, 0)$ should simply be $(s_1, 0, 0, \ldots, 0)$. Indeed, since $x^n + 1$ splits modulo q, the Chinese Remainder Theorem tells us that R_q is isomorphic as a ring to the ring $\mathbb{Z}_q^n$ under coordinate-wise multiplication. So, we can in fact write ring elements $a, s \in R_q$ in a coordinate system such that $a \cdot s = (a_1 s_1, \cdots a_n s_n)$. We call this the *CRT embedding*, in contrast to the *coefficient embedding* in which we view the ring elements as polynomials. We recall that the Chinese Remainder Theorem guarantees that we can move efficiently between these two embeddings. (Indeed, this is accomplished via an invertible linear map over the field $\mathbb{Z}_q$.)

It might seem a bit silly to have gone through all of the trouble of defining Ring-LWE over polynomial rings just to end up working with $\mathbb{Z}_q^n$ under coordinate-wise multiplication! But, we stress that the error distribution looks quite different in the CRT embedding. (If we used as an error distribution that is short in the CRT embedding, the resulting Ring-LWE problem would be easy.) To see this, let's consider the smallest non-trivial example. The polynomial $x^2 + 1$ splits modulo 13 as $x^2 + 1 = (x + 5)(x - 5) \bmod 13$, so an element $ax + b \in \mathbb{Z}_{13}/(x^2 + 1)$ has CRT representation $(5a + b, b - 5a) \in \mathbb{Z}_{13}^2$. (Check this!) Therefore, if our initial error distribution is, say, uniform over polynomials with $a, b \in \{-1, 0, 1\}$, then in the CRT embedding, our error distribution is uniform over the rather strange set $\{(0,0), \pm(5,-5), \pm(1,1), \pm(6,-4), \pm(6,-4)\}$, which in particular contains quite long elements, relative to $q = 13$. I.e., the mapping from the coefficient embedding to the CRT embedding is a linear transformation with large distortion (to the extent that one can define "distortion" over a finite vector space). So, while we *can* equivalently define Ring-LWE in terms of $\mathbb{Z}_q^n$ (when $x^n + 1$ splits modulo q), we would end up with a much less natural error distributions. In particular, the error distributions obtained from our worst-case to average-case reduction would be rather strange and depend on q in complicated ways.

2.3 When and how $x^n + 1$ splits modulo q

We now consider when $x^n + 1$ splits modulo q and show that the factors take a nice form. Notice that $x^n + 1$ is the minimal polynomial over $\mathbb{Z}$ of the (complex) primitive $2n$th roots of unity $e^{k\pi i/(2n)}$ for odd k. I.e., $x^n + 1$ splits over $\mathbb{C}$ precisely because $\mathbb{C}$ contains such elements. In analogy with this, suppose that $z \in \mathbb{Z}_q^*$ is a primitive $2n$th root of unity modulo q. That is, suppose $z^{2n} = 1 \bmod q$ but $z^k \neq 1 \bmod q$ for all $0 < k < 2n$. Then, clearly $z^n \neq 1 \bmod q$ is a square root of 1 in $\mathbb{Z}_q$. Since $\mathbb{Z}_q$ is a field, the only square roots of 1 are ± 1, so we must have $z^n = -1 \bmod q$. I.e., $z^n + 1 = 0 \bmod q$.

Furthermore, for any odd k, z^{kn} is also a primitive $2n$th root of unity. So, $z, z^3, z^5, \ldots, z^{2n-1} \in \mathbb{Z}_q^n$ are all roots of $x^n + 1$ modulo q. Indeed, they are distinct because $z^k \neq 1$ for $0 < k < 2n$. Finally, since $\mathbb{Z}_q$ is a field, there is only one non-zero polynomial over $\mathbb{Z}_q$ of degree n with these roots, and we must have $x^n + 1 = (x - z)(x - z^3)(x - z^5) \cdots (x - z^{2n-1}) \bmod q$. I.e., $x^n + 1$ splits modulo q.

So, $x^n + 1$ splits modulo a prime q if (and only if) there is an element of order $2n$ in $\mathbb{Z}_q^*$ (i.e., a primitive $2n$th root of unity modulo q). To find such a prime, we recall that $\mathbb{Z}_q^*$ is cyclic of order $q - 1$, so that it has an element of order $2n$ if and only if $2n$ divides $q - 1$. Therefore, $x^n + 1$ splits modulo a prime q if (and only if) $q = 1 \bmod 2n$. The Prime Number Theorem in arithmetic progressions guarantees that such primes exist and can be found efficiently. And, when this is the case, the factors of $x^n + 1$ modulo q can be written as $x - z^k$ for all odd $1 \leq k \leq 2n - 1$.

2.4 Some very special automorphisms τ_k

The above discussion shows a very natural way to think of the coordinates CRT embedding. Each coordinate in the CRT embedding of a polynomial $p(x)$ is simply $p(x) \bmod \mathcal{I}_i = p(z^{2i-1}) \bmod \mathcal{I}_i$ for some $i \in [n]$, where z is some fixed primitive $2n$th root of unity modulo q and $\mathcal{I}_i$ is the ideal generated by q and $x - z^{2i-1}$. It is therefore natural to order the coordinates in the CRT embedding so that the ith coordinate is $p(z^{2i-1})$. We then observe a nice symmetry of the CRT embedding. Let $k := (2i-1)^{-1}(2j-1) \bmod 2n$ (where we have used the fact that all odd numbers have an inverse modulo $2n$). Then, we see that the ith CRT coordinate of $p(x) \in R_q$ is the jth CRT coordinate of $p(x^k)$.

So, we define $\tau_k : R_q \to R_q$ for odd k such that $\tau_k(p(x)) := p(x^k)$. We see that τ_k can be viewed as a certain permutation of the coordinates in the CRT embedding. It is therefore a ring automorphism (i.e., it is a bijection respecting addition and multiplication). In fact, it also preserves norms in the coefficient embedding! I.e., $\|\tau_k(p(x))\| = \|p(x^k)\| = \|p(x)\|$, which can be seen by observing that τ_k simply permutes the coordinates of $p(x)$ (and flips some of their signs). Such maps are very rare,[3] and very useful. The next lemma extracts the specific property that we will need from them.

Lemma 2.2. *The maps $\tau_k : R_q \to R_q$ as described above are efficiently computable ring automorphisms preserving the norm (in the coefficient embedding). Furthermore, τ_k acts on the CRT embedding by permuting the coordinates, and for each $i, j \in [n]$, there is an efficiently computable k such that τ_k maps the ith CRT coordinate to the jth CRT coordinate.*

[3] As we've described these maps here, they only exist for our specific choice of R_q! They can, however, be generalized to more rings if we work in the canonical embedding rather than the coefficient embedding [LPR10].

2.5 The reduction

We can now finally present our reduction. As we discussed above, we can guess the coordinate s_1 and replace the Ring-LWE sample (a_i, b_i) by $(a_i + \alpha_i v_1 \bmod qR, b + \alpha_i \sigma_1 v_1 \bmod qR)$, where $v_1 = (1, 0, 0, \ldots, 0)^T$ in the CRT embedding, $\sigma_1 \in \mathbb{Z}_q$ is our guess for the first coordinate s_1 of s in the CRT embedding, and $\alpha_i \in \mathbb{Z}_q$ is uniformly random. Clearly, when our guess σ_1 is correct, the result is still a valid Ring-LWE sample with the same secret s, and the same error. However, when σ_1 is not correct, the result is not uniformly random. Instead, the first coordinate in the CRT embedding is uniformly random, but the remaining coordinates are completely unchanged.

To fix this, we use a hybrid argument together with the special maps τ_k. In particular, we let $\mathsf{Ring\text{-}LWE}_j$ be the variant of decision Ring-LWE that asks us to distinguish Ring-LWE samples in which the first $j - 1$ coordinates in the CRT embedding are replaced by uniformly random noise from Ring-LWE samples in which the first j CRT coordinates are replaced by uniformly random noise. To show the hardness of decision Ring-LWE, it suffices to show the hardness of $\mathsf{Ring\text{-}LWE}_j$ for each j.

Notice that the above argument lets us use an oracle for $\mathsf{Ring\text{-}LWE}_1$ to learn the first coordinate s_1 in the CRT embedding of the secret s of a Ring-LWE instance. More generally, we can use an oracle for $\mathsf{Ring\text{-}LWE}_j$ to find the jth coordinate s_j. So, to finish our proof, we need to show how the ability to find the jth coordinate s_j in the CRT embedding allows us to find all coordinates s_i. This is where we use the maps τ_k. In particular, Lemma 2.2 lets us find a k such that τ_k maps the ith coordinate to the jth coordinate in the CRT embedding. Since τ_k is a ring automorphism, it converts Ring-LWE samples with secret s to Ring-LWE samples with secret $\tau_k(s)$. Furthermore, since τ_k preserves the norm and the error distribution χ is spherically symmetric, τ_k preserves the error distribution.

So, our full reduction from search Ring-LWE to $\mathsf{Ring\text{-}LWE}_j$ behaves as follows. For each $i = 1, \ldots, n$, we use our $\mathsf{Ring\text{-}LWE}_j$ oracle to find the ith coordinate s_i of s in the CRT embedding by first computing $k = (2i - 1)^{-1}(2j - 1) \bmod 2n$ such that τ_k maps the ith CRT coordinate to the jth CRT coordinate, as in Lemma 2.2. Let $v_j \in R_q$ be the element whose coordinates in the CRT embedding are $(0, 0, \ldots, 1, 0, \ldots, 0)$, where the 1 is in the jth position. For each $\sigma \in \mathbb{Z}_q$, we replace our Ring-LWE samples (a_ℓ, b_ℓ) by $(\tau_k(a_\ell) + \alpha_\ell v_j, \tau_k(b_\ell) + \sigma \alpha_\ell v_j + \widetilde{e}_\ell)$, where $\alpha_\ell \in \mathbb{Z}_q$ is uniformly random, and $\widetilde{e}_\ell \in R_q$ has its first $j - 1$ coordinates uniformly random in the CRT embedding and last $n - j + 1$ coordinates equal to zero. If $\sigma = s_i$, then the resulting distribution

$$(\tau_k(a_\ell) + \alpha_\ell v_j, \tau_k(a_\ell)\tau_k(s_\ell) + \alpha_\ell \sigma v_j + \tau_k(e_\ell) + \widetilde{e}_\ell)$$

will be exactly the YES case of $\mathsf{Ring\text{-}LWE}_j$ with secret $\tau_k(s)$—i.e., the first $j - 1$ coordinates will be uniformly random and the last $n - j + 1$ coordinates will correspond to valid Ring-LWE samples. Otherwise, the distribution will be exactly the NO case—i.e., the jth coordinate will also be uniformly random.

3 The worst-case to average-case reduction

We can now move on to the worst-case to average-case reduction for (search) Ring-LWE. Fortunately, very little of the math from the previous section is needed for this part. We will, however, assume a basic familiarity with the worst-case to average-case reduction for plain LWE, as presented in the earlier lectures or as described in [Pei09] (or in [Reg09] as the "classical part").

Recall that we defined the Bounded Distance Decoding problem (BDD) as the problem that asks us to find a lattice vector $\boldsymbol{y} \in \mathcal{L}$ that is closest to some target $\boldsymbol{t}$, given the promise that this vector is actually very close, $\mathrm{dist}(\boldsymbol{t}, \mathcal{L}) \ll \lambda_1(\mathcal{L})$. We define the analogous problem for ideal lattices over R. (We take our target $t \in \mathbb{R}[x]/(x^n+1)$, but we do not need many properties of this structure. We just need that it is a ring that contains R and that we can round from $\mathbb{R}[x]/(x^n+1)$ to R in the natural way.)

Definition 2. *For a power of two n and an approximation factor $\alpha < 1/2$, the α-IdealBDD problem over $R = \mathbb{Z}[x]/(x^n+1)$ is defined as follows. The input is (a basis for) an ideal $\mathcal{I}$ over R and a target $t \in \mathbb{R}[x]/(x^n+1)$ such that $\mathrm{dist}(t, \mathcal{I}) \le \alpha\lambda_1(\mathcal{I})$. The goal is to output the (unique) element $y \in \mathcal{I}$ with $\|y - t\| \le \alpha\lambda_1(\mathcal{I})$.*

In the plain LWE case, we reduced BDD to LWE, and then we reduced GapSVP to BDD, so that we were able to prove hardness from a more standard lattice problem. We could do the same thing here, but recall that γ-IdealGapSVP for $\gamma > \sqrt{n}$. So, our worst-case problem will simply be IdealBDD. (There is actually a quantum reduction that reduces IdealSVP to IdealBDD with appropriate parameters [LPR10, PRS17], but we will not present this here.) In particular, we prove the following theorem. (To make the proof easier, we have chosen the rather extreme noise parameter $\sigma > n^{\omega(1)}\alpha q$, which makes the theorem vacuous unless $\alpha < n^{-\omega(1)}$. More careful analysis of essentially the same reduction gives $\sigma \ge \mathrm{poly}(n, \ell)\alpha q$ for some fixed polynomial and therefore allows for $\alpha = 1/\mathrm{poly}(n, \ell)$. Removing the dependence on ℓ takes much more work [LPR10].)

Theorem 3.1 (Weak variant of [LPR10])**.** *For any power of two n, $q \ge 2^n$, and any $\alpha < n^{-\omega(1)}$, there is an efficient reduction from α-IdealBDD over $R = \mathbb{Z}[x]/(x^n+1)$ to Ring-LWE over R with noise sampled from the discrete Gaussian $D_{R,\sigma}$ over R with parameter $\sigma > n^{\omega(1)}(n + \alpha q)$.*

3.1 The inverse ideal and discrete Gaussian samples

As in the reduction for plain LWE, a key tool will be the dual $\mathcal{L}^*$ of our worst-case lattice $\mathcal{L}$. Recall that $\mathcal{L}^*$ is defined as the set of vectors that have integral inner product with every lattice vector,

$$\mathcal{L}^* := \{\boldsymbol{w} \in \mathbb{Q}^n \ : \ \forall \boldsymbol{y} \in \mathcal{L}, \ \langle \boldsymbol{w}, \boldsymbol{y} \rangle \in \mathbb{Z}\} \ .$$

One can check that $\mathcal{L}^*$ is itself a (scaling of a) lattice. The important property of $\mathcal{L}^*$ that we use in the reduction is that, for a BDD target $\boldsymbol{t} \in \mathbb{R}^n$, we can write

$$\langle \boldsymbol{w}, \boldsymbol{t} \rangle = \langle \boldsymbol{w}, \boldsymbol{y} \rangle + \langle \boldsymbol{w}, \boldsymbol{e} \rangle \ ,$$

where $\langle \boldsymbol{w}, \boldsymbol{y} \rangle$ is an integer and $\langle \boldsymbol{w}, \boldsymbol{e} \rangle$ is relatively small.

In the context of ideals, we will instead work with the *inverse* ideal

$$\mathcal{I}^{-1} := \{w \in \mathbb{Q}[x]/(x^n+1) \ : \ \forall y \in \mathcal{I}, \ w \cdot y \in R\} \ .$$

(There is a notion of a dual ideal that is *different* than this notion, whose definition only makes sense in the canonical embedding. So, we avoid calling $\mathcal{I}^{-1}$ the "dual ideal." Again, we are relying here on the very special properties of the ring $\mathbb{Z}[x]/(x^n+1)$ for n a power of two in order to simply things.) To see that this is an ideal, we simply need to observe that (1) it is closed under addition, and (2) it is closed under multiplication by x. Both facts are immediate from the relevant definitions. E.g., $(xw) \cdot y = w \cdot (xy) \in R$ for any $w \in \mathcal{I}^{-1}$.[4] Furthermore, we have the ring analogue of the above identity,

$$wt = wy + we \ ,$$

where $wy \in R$ and we is short.

As in the plain LWE case, our vectors w will be sampled from the discrete Gaussian distribution $D_{\mathcal{I}^{-1},\sigma'}$, defined by

$$\Pr_{W \sim D_{\mathcal{I}^{-1},\sigma'}}[W = w] \propto e^{-\pi \|w\|^2/\sigma'^2}$$

for all $w \in \mathcal{I}^{-1}$. (Since this is a probability distribution, we only need to define this up to the constant of proportionality.)

We will only need some very basic properties from $D_{\mathcal{I}^{-1},\sigma'}$. First, we can sample efficiently from $D_{\mathcal{I}^{-1},\sigma'}$ for $\sigma' > 2^n \cdot \lambda_n(\mathcal{I}^{-1})$. Second, for $\sigma' > \sqrt{n}q \cdot \lambda_n(\mathcal{I}^{-1})$, a sample $w \sim D_{\mathcal{I}^{-1},\sigma'}$ is statistically close to uniformly random modulo $q\mathcal{I}^{-1}$. Third, a sample $w \sim D_{\mathcal{I}^{-1},\sigma'}$ is not too long, i.e. except with negligible probability we have $\|w\| < \sqrt{n}\sigma'$. We have seen all of these properties in previous lectures, and none of them depend on the ideal structure at all.

3.2 Mapping $\mathcal{I}^{-1}/(q\mathcal{I}^{-1})$ to R_q

We noted in the previous section that $w \sim D_{\mathcal{I}^{-1},\sigma'}$ is essentially uniformly random modulo $q\mathcal{I}^{-1}$ for appropriate σ'. We would like to create Ring-LWE samples (a, b) where $a \in R_q = R/(qR)$ somehow corresponds to this coset. I.e., we would like to map $\mathcal{I}^{-1}/(q\mathcal{I}^{-1})$ to $R/(qR)$. In the plain LWE world, this was simply a matter of computing the coordinates $\mathbf{B}^{-1}\boldsymbol{w} \bmod q$ modulo q in some basis $\mathbf{B}$ for $\mathcal{L}^*$ of the dual lattice vector $\boldsymbol{w} \in \mathcal{L}^*$.

In the ring case, we must be more careful because our map must preserve the multiplicative structure of $\mathcal{I}^{-1}/(q\mathcal{I}^{-1})$ and R_q. We will need a bijective map $\theta : \mathcal{I}^{-1}/(q\mathcal{I}^{-1}) \to R_q$ that is linear (i.e., $\theta(w_1 + w_2) = \theta(w_1) + \theta(w_2) \bmod qR$ for any $w_1, w_2 \in \mathcal{I}^{-1}/(q\mathcal{I}^{-1})$) and respects multiplication by any ring element, i.e., $\theta(xw) = x\theta(w) \bmod qR$ for all $w \in \mathcal{I}^{-1}/(q\mathcal{I}^{-1})$. (Formally, this is an isomorphism of R modules.)

Lemma 3.2. *For $q \geq 2$ and an ideal $\mathcal{I} \subseteq R$, let $z \in \mathcal{I}$ be any element such that there exists $\widehat{w} \in \mathcal{I}^{-1}$ and $\widehat{r} \in R$ such that $\widehat{w}z + \widehat{r}q = 1$. Then, the map $\theta_z : \mathcal{I}^{-1}/(q\mathcal{I}^{-1}) \to R_q$ defined by $\theta_z(w) = zw \bmod qR$ is a linear bijection satisfying $\theta_z(xw) = x\theta_z(w) \bmod qR$ for all $w \in \mathcal{I}^{-1}/(q\mathcal{I}^{-1})$.*

Furthermore, such a z always exists and can be found efficiently given (a basis for) $\mathcal{I}$ and q, and θ_z and its inverse are efficiently computable.

Proof. It follows from the basic properties of multiplication that θ_z is linear and respects multiplication. So, we only need to prove that θ_z is a bijection. Indeed, multiplication by $\widehat{w}$ is the inverse of θ_z. I.e., for each $r \in R$, we have $\theta_z(\widehat{w}r) = z\widehat{w}r \bmod qR = r \bmod qR$, where we have used the fact that $\widehat{w}r = 1 \bmod qR$. Therefore, θ_z is surjective. To prove that it is bijective, it suffices to note that the domain and range have the same size, q^n. The fact that θ_z is efficiently computable is trivial, and the fact that it can be inverted efficiently follows from the fact that it is a linear bijection. (I.e., we can write down the matrix corresponding to θ_z, which is guaranteed to have an inverse. So, we can simply compute its inverse.)

[4] Since $\mathcal{I}^{-1} \not\subseteq R$, it is technically only a *fractional* ideal. I.e., there exists some denominator $z \in \mathbb{Z}$ and some ideal $\mathcal{J} \subseteq R$ such that $\mathcal{I}^{-1} = z^{-1}\mathcal{J}$.

For the proof that z exists and can be found efficiently, see [LPR10, Lemma 2.14]. It relies on the Chinese Remainder Theorem together with the factorization of an ideal into the product (i.e., intersection) of prime ideals (i.e., ideals $\mathcal{P}$ such that for any ideal $\mathcal{J}$, either $\mathcal{P}$ and $\mathcal{J}$ are coprime or $\mathcal{J} \subseteq \mathcal{P}$). □

3.3 The reduction

Recall that in the plain LWE reduction, we sampled $\boldsymbol{w}_i \sim D_{\mathcal{L}^*,\sigma'}$ for appropriately chosen parameter $s > 0$ and created LWE samples

$$(\mathbf{B}^{-1}\boldsymbol{w}_i \bmod q, \lfloor\langle \boldsymbol{w}_i, \boldsymbol{t}\rangle\rceil + \widetilde{e}_i \bmod q) ,$$

where $\mathcal{L} \subseteq \mathbb{Z}^n$ is the input lattice to the BDD instance, $\boldsymbol{t} \in \mathbb{R}^n$ is the input target, and $\widetilde{e}_i$ is some extra noise that we add.

In the Ring-LWE world, the natural analogue is as follows. We take as input (a basis for) an ideal $\mathcal{I} \subseteq R$ over R and a target $t \in \mathbb{R}[x]/(x^n+1)$. We sample $w_1, \ldots, w_\ell \sim D_{\mathcal{I}^{-1},\sigma'}$ from the discrete Gaussian for $\sqrt{n}q\lambda_n(\mathcal{I}^{-1}) < \sigma' \leq 2\sqrt{n}q\lambda_n(\mathcal{I}^{-1})$.[5]

We then create Ring-LWE samples (a_i, b_i) with $b_i := \lfloor w_i \cdot t\rceil + \widetilde{e}_i \bmod qR$, where $\widetilde{e}_i \sim \chi$ is some additional noise and $\lfloor\cdot\rceil$ just means rounding the coefficients to integers. To create a_i, we find $z \in \mathcal{I}$ with $\theta_z : \mathcal{I}^{-1}/(q\mathcal{I}^{-1}) \to R_q$ as guaranteed by Lemma 3.2. We then set $a_i := \theta_z(w_i) = zw_i \bmod qR$. Notice that the fact that w_i is statistically close to uniformly random modulo $q\mathcal{I}^{-1}$ together with the fact that θ_z is a bijection immediately implies that $a_i \in R_q$ is statistically close to uniformly random.

It remains to study the distribution of b_i. We can write

$$\lfloor w_i \cdot t\rceil + \widetilde{e}_i \bmod qR = w_i y + \lfloor w_i e\rceil + \widetilde{e}_i \bmod qR ,$$

where $y \in \mathcal{I}$ is a closest lattice vector to t, and $e := y - t$, and we have used the fact that $w_i y \in R$ to write $\lfloor w_i y + w_i e\rceil = w_i y + \lfloor w_i e\rceil$. We are promised that $\|e\| \leq \alpha\lambda_1(\mathcal{I})$.

We first study the error term $\lfloor w_i e\rceil + \widetilde{e}_i$. We just need to show that $\|\lfloor w_i e\rceil\| \leq \sigma/n^{\omega(1)}$ is short, since $\widetilde{e}_i$ is sampled from a Gaussian with parameter σ (and it is a basic fact that a Gaussian with parameter σ is within statistical distance $O(d/\sigma)$ of the same Gaussian shifted by d). Indeed, recall from the previous lecture that $\|w_i e\| \leq \sqrt{n}\|w_i\|\|e\|$ (which follows immediately from Cauchy-Schwarz). We noted earlier that $\|w_i\| \leq \sqrt{n}\sigma \leq 2nq\lambda_n(\mathcal{I}^{-1})$ with high probability. And, by assumption $\|e\| \leq \alpha\lambda_1(\mathcal{I}) \leq \alpha n/\lambda_n(\mathcal{I}^{-1})$, where the second inequality is the transference theorem that we discussed in an earlier lecture [Ban93]. Putting all of this together, we see that $\|\lfloor w_i e\rceil\| \leq n + \|w_i e_i\| \leq n + 2\alpha n^{2.5}q$ with high probability. Since we took $\sigma > n^{\omega(1)}(n + \alpha q)$, our error is statistically close to a Gaussian with parameter σ. (I.e., the additional noise from $w_i e$ does not affect the overall distribution much.)

Next, we turn to $w_i y$. Recall from Lemma 3.2 that we can write $1 = \widehat{w}z + \widehat{r}q$ for some $\widehat{w} \in \mathcal{I}^{-1}$ and $\widehat{r} \in R$. Therefore,

$$w_i y = w_i y(\widehat{w}z + \widehat{r}q) = w_i z\widehat{w}y + w_i y\widehat{r}q = a_i\widehat{w}y \bmod qR ,$$

where we have used the definition of $a_i = w_i z \bmod qR$ and the fact that $w_i y \in R$, since $w_i \in \mathcal{I}^{-1}$ and $y \in \mathcal{I}$.

[5] We can try many different parameters until we find one that happens to work, using the 2^n-approximation to $\lambda_n(\mathcal{I}^{-1})$ given by the LLL algorithm to guarantee that we need only try $O(n)$ different parameters.

Finally, we note that $\widehat{w}y \in R$ is independent of i, so we simply define $s := \widehat{w}y \bmod qR$, and we notice that the (a_i, b_i) are valid Ring-LWE samples with secret s. We then simply note that $zs = y \bmod q\mathcal{I}$, so that we can recover $y \bmod q\mathcal{I}$ from s. So, our Ring-LWE oracle allows us to find $y \bmod q\mathcal{I}$. To finish, we need to find y given t and $y \bmod q\mathcal{I}$, which is equivalent to solving IdealBDD with parameter $\alpha' = \alpha/q < 2^{-n}$. This can be solved efficiently, e.g., by rounding the coordinates of the target in an LLL-reduced basis.

3.4 A note on the error

If we had not chosen the rather extreme approximation factor $\alpha = n^{-\omega(1)}$, then we would have had to study the error $w_i e$ more carefully. In fact, this error is not really "average case" in the sense that it depends fundamentally on e. For example, recall that e is a polynomial and suppose that it has some root, perhaps over the integers. Then, clearly, $w_i e$ must have the same root, so it lies in a subspace and cannot be distributed like a spherical Gaussian, regardless of the distribution of w_i. In general, the distribution of $w_i e$ will be close to a discrete Gaussian over R with some covariance matrix that depends on e. (In the canonical embedding, which we have avoided defining, each coordinate will be an independent Gaussian whose parameter is proportional to the corresponding coordinate of e.) I.e., we will *not* get the same error distribution regardless of the input IdealBDD instance.

Lyubashevsky, Peikert, and Regev offer two solutions for this [LPR10]. The first is to solve this problem "in the lattice regime," by rerandomizing the target t. I.e., we reduce IdealBDD to a variant of IdealBDD in which the target is sampled from a Gaussian distribution. This still does not quite allow us to reduce to Ring-LWE with fixed spherical Gaussian error. Instead, this gives us a distribution over covariance matrices such that Ring-LWE with error given by the Gaussian whose covariance is sampled from this distribution is worst-case hard with high probability. This is enough to build cryptography, but naturally this is not used in practice.

The second solution in [LPR10] is to use noise flooding in the RingLWE instance itself. I.e., we take $\widetilde{e}$ to be a spherical Gaussian that is significantly larger than $\|w_i e\|$, just like we did. However, to avoid having to take $\alpha = n^{-\omega(1)}$, we allow the noise parameter to depend on the number of samples ℓ in the Ring-LWE instance. We then get a distribution $w_i e + \widetilde{e}$ that is *not* within negligible statistical distance of a spherical Gaussian, but the two distributions will still have large overlap, even over ℓ samples.

4 NTRU

Finally, we mention a different elegant way to build public-key encryption using polynomial rings, such as R_q, the NTRU encryption scheme, due to Hoffstein, Pipher, and Silverman [HPS98]. Historically, NTRU predates LWE by nearly a decade and Ring-LWE by about 15 years. As far as we know, it is more-or-less as secure as Ring-LWE-based schemes for most reasonable parameter settings. However, unlike Ring-LWE-based schemes, NTRU comes with no worst-case hardness guarantee. We present it here because (1) it is pretty; (2)one of the relatively few concrete assumptions known to imply public-key cryptography; and (3) people who work in lattice-based cryptography should know what NTRU is.

As before, we work over $R := \mathbb{Z}[x]/(x^n + 1)$ for power-of-two n with $R_q := R/(qR)$ for some modulus $q = \mathrm{poly}(n)$. A "typical" element in R is invertible modulo $3R$ (i.e., the polynomial $x^n + 1$

does not have low-degree factors modulo 3),[6] and we may, e.g., take q to be prime to guarantee the same modulo qR. (NTRU can be defined over any polynomial ring, and it is often actually defined over $\mathbb{Z}[x]/(x^n-1)$. This causes some annoying issues related to those that we observed in the Ring-SIS lecture. They can be overcome, but we ignore this issue by using our preferred ring.)

- **Key generation:** Sample two short polynomials $g, f \in R$. E.g., sample them uniformly at random from $R_{\{0,1\}}$. If f is not invertible modulo both qR and $3R$, we resample it. Otherwise, we denote these respective inverses by f_q^{-1} and f_3^{-1}. The public key is $h := gf_q^{-1} \bmod qR$, and the private key is f, g.
- **Encryption:** Let $m \in R_{\{-1,0,1\}}$ be some ternary message. The encryption algorithm computes the ciphertext $c := 3rh + m \bmod qR$, where r is some random short polynomial.
- **Decryption:** Given a ciphertext c, we compute $fc = 3rg + fm \bmod qR$. As long as q is sufficiently large, this element $3rg + fm$ should have small coefficients relative to q. I.e., by choosing our representative of $3rg+fm \bmod qR$ to have coefficients in the interval $(-q/2, q/2]$, we can actually recover $3rg + fm \in R$, not just its coset in R_q. This allows us to reduce the result modulo $3R$ to recover fm. Finally, we multiply by f_3^{-1} to find m, which is uniquely determined by its coset modulo $3R$.

The security of NTRU is typically proven under the assumption that the public key h is indistinguishable from random. However, there is no known reduction from a more standard computational problem to the problem of distinguishing h from random. For most choices of parameters, however, our best attack on NTRU is a lattice attack that searches for a short vector in the so-called NTRU lattice, spanned by the basis

$$\begin{pmatrix} I_n & 0 \\ \mathrm{Rot}(\boldsymbol{h}) & qI_n \ , \end{pmatrix} \in \mathbb{Z}^{2n\times 2n}$$

where

$$\mathrm{Rot}\begin{pmatrix} h_1 \\ h_2 \\ \vdots \\ h_n \end{pmatrix} := \begin{pmatrix} h_1 & -h_n & -h_{n-1} & \cdots & -h_2 \\ h_2 & h_1 & -h_n & \cdots & -h_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{n-1} & h_{n-2} & h_{n-3} & \cdots & -h_n \\ h_n & h_{n-1} & h_{n-2} & \cdots & h_1 \end{pmatrix} ,$$

as in the previous lecture, and $\boldsymbol{h}$ is the coefficient vector of the public key h. Notice that the NTRU lattice contains the short vector $(\boldsymbol{f}, \boldsymbol{g}) \in \mathbb{Z}^{2n}$ corresponding to the secret key. Indeed, any short enough vector in this lattice can be used to break the NTRU encryption scheme.

References

[Ban93] Wojciech Banaszczyk. New bounds in some transference theorems in the geometry of numbers. *Mathematische Annalen*, 296(4), 1993.

[6] It's factorization into irreducible polynomials is $x^n + 1 = (x^{n/2} + x^{n/4} - 1)(x^{n/2} - x^{n/4} - 1)$ modulo 3. The fact that these polynomials are irreducible is equivalent to saying that a finite field of characteristic 3 contains a primitive $2n$th root of unity if and only if it has size 3^m for m divisible by $n/2$, i.e., that $2n$ divides $3^m - 1$ if and only if $n/2$ divides m (since the multiplicative group of a finite field is cyclic). I was frustrated by my inability to find a nice enough proof of this, so I asked on Math StackExchange and got some very nice answers [Nic]—three very nice proof as of the last time I checked, as well as my own rather clunky proof.

[BLP+13] Zvika Brakerski, Adeline Langlois, Chris Peikert, Oded Regev, and Damien Stehlé. Classical hardness of learning with errors. In *STOC*, 2013.

[HPS98] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. Ntru: A ring-based public key cryptosystem. In *ANTS*, 1998.

[LPR10] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. In *Eurocrypt*, 2010.

[LPR13] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. A toolkit for ring-lwe cryptography. In *Eurocrypt*, 2013.

[Nic] Nicer proof that $2{̂n+2}$ divides $3m̂-1$ if and only if $2{̂n}$ divides m.

[Pei09] Chris Peikert. Public-key cryptosystems from the worst-case shortest vector problem: Extended abstract. In *STOC*, STOC '09, New York, NY, USA, 2009.

[PRS17] Chris Peikert, Oded Regev, and Noah Stephens-Davidowitz. Pseudorandomness of ring-lwe for any ring and modulus. In *STOC*, 2017.

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