Lattice cryptography: from cryptanalysis to New Foundations

Woerden, W.P.J. van

CHAPTER 10

Remarkable Lattices & Cryptography

This chapter is based on the joint work 'On the Lattice Isomorphism Problem, Quadratic Forms, Remarkable Lattices, and Cryptography', with Léo Ducas, published at Eurocrypt 2022. The last section summarizes the follow-up joint work 'HAWK: Module LIP makes Lattice Signatures Fast, Compact and Simple', with Léo Ducas, Eamonn W. Postlethwaite and Ludo Pulles.

10.1 Introduction

The central idea in lattice-based cryptography is the concept of a bad basis acting as a public key, and a good basis of the same lattice acting as a secret key. With a good basis one can efficiently decode or Gaussian sample at low width, while with a bad basis this is hard. This discrepancy in hardness allows to create a backdoor that can be used for encryption schemes, signature schemes and more.

The creation of such a basis key-pair is precisely where the hardness assumptions such as NTRU [HPS98] and the Learning with Error (LWE) problem [Reg09] come in. They allow us to securely generate a somewhat random lattice along with a (partial) good basis. These assumptions are very versatile and allow to instantiate many different kinds of cryptographic schemes. However, from a decoding perspective, these good bases can only decode up to a radius $\Omega(\sqrt{n})$ from the optimal Minkowski bound, as they essentially reduce to the case of the trivial lattice $\mathbb{Z}^n$. Due to this non-optimal decoding distance we can break these schemes by solving SVP in some dimension $\beta \leq n/2+o(n)$, which forces the dimension to be at least 2 times larger than a direct SVP attack would imply.

In contrast, there do exist many remarkable lattices for which we can efficiently decode at much lower approximation factors. For example, Ducas and Pierrot [DP19] showed that the Chor-Rivest cryptosystem [CR88] was implicitly relying on a family of lattices for which it is possible to efficiently decode errors up to a radius within a factor of $O(\log n)$ from optimal. Recently, lattice families reaching a factor of $O(\sqrt{\log n})$ were found [MP21; BP22]. We cannot use such remarkable lattice directly, everyone, including an adversary, can decode in them. To use such remarkable lattices in cryptographic protocols we need a way to hide their remarkable structure, without destroying their decoding capabilities.

At repeated occasions [CR88; Len91; OTU00; SCM01; LLXY20], and over more than 30 years, exactly this has been tried, for example by porting the original public-key encryption scheme of McEliece [McE78] from codes to lattices: instead of giving a public bad basis of the lattice $\mathcal{L} \subset \mathbb{R}^n$ itself, one gives a bad basis of the permuted lattice $\pi(\mathcal{L}) := \{(y_{\pi^{-1}(i)})_i \in \mathbb{R}^n : \mathbf{y} \in \mathcal{L}\}$ for some secret random permutation π. One can efficiently decode in $\pi(\mathcal{L})$ via $\mathcal{L}$ (only) when π is known. Unfortunately this code-style way of hiding the lattice is also susceptible for code-style attacks, leading to subexponential attacks [CR88; Lap21; Odl90]; either by Information Set Decoding techniques or by guessing or brute-forcing some coordinates. Due to these attacks, this approach has not been very popular lately.

These attacks are enabled by the fact that these schemes do barely more than re-randomize the lattice by applying a permutation of the coordinates. Such permutations are isometries, i.e., lattice isomorphisms, but those are not the only ones. Since the isometry group $\mathcal{O}_n(\mathbb{R})$ acting on lattices is much larger than the one acting on codes, applying a random isometry from this larger group should convincingly thwart those code-style attacks: the canonical coordinate system becomes irrelevant.

All these remarks point toward the Lattice Isomorphism Problem (LIP) as a potential theoretical platform for finally getting this natural approach properly formalized, and hopefully, truly "lattice-based" in the cryptanalytic sense: the best known attack should be based on generic lattice reduction algorithms such as LLL [LLL82] and BKZ [Sch87]. As seen in Section 9.5 the current state of the art on LIP supports this hypothesis: all known algorithms [PP85; PS97; HR14; DHVW20] rely on finding short vectors. This is the case even for algorithms specialized to the trivial lattice $\mathbb{Z}^n$ [Szy03]. Experimental studies [BM21] do show that the basis randomization step requires care, in order to not introduce any weak instances.

While instantiating LIP with $\mathbb{Z}^n$ may already give rise to secure cryptosystems, the end goal of this work is to enable lattice-based cryptosystems that could be be even more secure than those based on LWE and SIS in similar dimensions, by instantiating the constructed schemes with remarkably decodable lattices.

10.1.1 Contributions

We introduce a formal and convenient framework for LIP-based cryptography, from which we build three cryptographic schemes: an identification scheme based on search-LIP (sLIP), a (passively secure) Key Encapsulation Mechanism (KEM) based on distinguish-LIP (ΔLIP), and a signature scheme also based on ΔLIP. In more detail:

- In Chapter 9 we discussed LIP, recalled the quadratic form formalism, and rephrased the LIP problem in terms of quadratic forms to conveniently avoid real numbers. Here, with the use of Gaussian Sampling [GPV08; Pei10], we define an average-case distribution for LIP and establish a worst-case to average-case reduction within an isomorphism class (Section 10.2). This addresses the concerns raised by Blanks and Miller [BM21] and formalizes their heuristic countermeasure.

- The above cryptographic foundations are directly inspired by the Zero-Knowledge proof of lattice non-isomorphism of Haviv and Regev [HR14]. We further extend on their techniques by proposing a Zero-Knowledge proof of knowledge (ZKPoK) of a lattice isomorphism (Section 10.3). This directly implies the existence of an identification scheme based on sLIP.

- We propose a KEM scheme (Section 10.4) and a hash-then-sign signature scheme (Section 10.5), both based on ΔLIP. Perhaps surprisingly, and unlike the original scheme of McEliece for codes, we circumvent the additional assumption that decoding a certain class of random lattices is hard. This is done via a lossyness argument [PW11] for the KEM, and a dual argument for the signature scheme.

- We review the state of the art for solving LIP (Section 10.6). In particular we note that all known algorithms require lattice reduction, and we quantify the required approximation factor.

- We discuss natural instantiations for each scheme given any remarkable lattice in Section 10.7. This section handles the construction of the auxiliary lattice appearing in ΔLIP for the lossyness arguments to get through.

10.1.2 Potential advantages

The KEM. To instantiate the KEM, consider a lattice $\mathcal{L}$ (w.l.o.g., of volume 1) such that:

- the minimal distance is within a factor f from Minkowski's bound: $\lambda_1(\mathcal{L}) \geq \Omega(\sqrt{n}/f)$,

- there exists an efficient algorithm that can decode errors in $\mathcal{L}$ up to radius $\rho < \lambda_1(\mathcal{L})/2$ within a factor f' from Minkowski's bound: $\rho \geq \Omega(\sqrt{n}/f')$.[1]

- the dual minimal distance is within a factor f^* from Minkowski's bound: $\lambda_1(\mathcal{L}^*) \geq \Omega(\sqrt{n}/f^*)$.

[1] Note that uniqueness of decoding implies $f' \geq 2f$.

Then, the instantiated KEM appears to resist concrete lattice reduction attacks down to an approximation factor $O(\max(f, f^*, f'))$. For a security reduction to ΔLIP we have to create a pair of lattices, which increases the approximation factor to $O(\max(f, f^*) \cdot f^* \cdot f')$. More specifically, it's security is based on ΔLIP for two lattices whose primal and dual first minima are all within a factor $O(\max(f, f^*) \cdot f^* \cdot f')$ from Minkowski's bound.

The trivial lattice $\mathbb{Z}^n$ gives all three factors f, f', f^* of the order $\Theta(\sqrt{n})$. The Barnes-Wall lattice improves all three factors down to $\Theta(\sqrt[4]{n})$ [MN08].

The endgame would be to instantiate with lattices for which all three factors f, f', f^* would be very small. In particular, one would naturally turn to recent work on decoding the Chor-Rivest lattices [CR88; DP19; LLXY20; Lap21] and the Barnes-Sloane lattices [MP21] giving $f = \text{polylog}(n)$ and $f' = \text{polylog}(n)$, but unfortunately their dual gaps are not that good: $f^* \geq \Theta(\sqrt{n})$. Indeed, both Chor-Rivest and Barnes-Sloane are integer lattices $\mathcal{L} \subset \mathbb{Z}^n$ with single exponential volume $\det(\mathcal{L}) = c^n$: their duals $\mathcal{L}^*$ have a Minkowski bound of $\Theta(\sqrt{n}/\det(\mathcal{L})^{1/n}) = \Theta(\sqrt{n})$, but contain all of $\mathbb{Z}^n \subset \mathcal{L}^*$, including vectors of norm 1.

Note, nevertheless that there is no geometric impossibility to the existence of the desired remarkably decodable lattice: random lattices have $f = O(1)$ and $f^* = O(1)$; so unique decoding is in principle possible down to $f' = f/2 = O(1)$. Unfortunately the the best known decoding algorithm for such lattices takes exponential time.

The signature scheme. The same principle also applies to our signature scheme, but this time with respect to Gaussian sampling rather than decoding: lattices with tight sampling (and large dual minimal distance) would lead to a scheme resisting attacks down to very small approximation factors. Unfortunately, even ignoring the constraint on the dual lattice, we do not know of any lattice much better than $\mathbb{Z}^n$ for efficient Gaussian sampling. However, instantiated with $\mathbb{Z}^n$ our scheme still has an interesting feature: not having to deal with any Gram-Schmidt or Cholesky matrices over the real numbers. This may be a worthy practical advantage over currently proposed hash-then-sign signature schemes [GPV08].

	Primal	Dual	Decoding
Decodable Lattice	f	f^*	f'
Random Lattice	$\Theta(1)$	$\Theta(1)$	$2^{\Theta(n)}$
$\mathbb{Z}^n$	$\Theta(\sqrt{n})$	$\Theta(\sqrt{n})$	$\Theta(\sqrt{n})$
NTRU [HPS98]	$\Omega(\alpha)$	$\Omega(\alpha)$	$\Omega(n/\alpha)$
LWE [Ajt99; AP11; MP12]	$\Omega(1)$	$\Omega(\alpha)$	$\Omega(n/\alpha)$
Prime Lattice [CR88; DP19]	$\Theta(\log n)$	$\Omega(\sqrt{n})$	$\Theta(\log n)$
Barnes-Sloane [MP21]	$\Theta(\sqrt{\log n})$	$\Omega(\sqrt{n})$	$\Theta(\sqrt{\log n})$
Reed-Solomon [BP22]	$\Theta(\sqrt{\log n})$	$\Omega(\sqrt{n})$	$\Theta(\sqrt{\log n})$
Barnes-Wall [MN08]	$\Theta(\sqrt[4]{n})$	$\Theta(\sqrt[4]{n})$	$\Theta(\sqrt[4]{n})$

Table 10.1: Some decodable lattices and their primal gap $f = \mathrm{gh}(\mathcal{L})/\lambda_1(\mathcal{L})$, dual gap $f^* = \mathrm{gh}(\mathcal{L}^*)/\lambda_1(\mathcal{L}^*)$ and efficient decoding gap $f' = \mathrm{gh}(\mathcal{L})/\rho$. We have $1 \leq \alpha \leq n$.

The identification scheme. Because sLIP seems super-exponentially hard in the dimension for well chosen lattices (i.e., that have a large kissing number), it might be secure to instantiate our ZKPoK with a rather small lattice dimension, maybe down to about a hundred (see the challenge in Table 10.2). This is more a theoretical curiosity than a practical advantage —the protocol still needs soundness amplification, where each round requires exchanging $\tilde{O}(n^2)$ bits.

10.1.3 Open questions

A KEM with polylog*-approximation factor security.* Is there any family of lattices that can be efficiently decoded within a polylog factor from Minkowski's bound such as [CR88; DP19; LLXY20; Lap21; MP21], but whose dual would also have an equally large minimal distance?

Tight Gaussian Sampling for signatures. Is there any family of lattices $\mathcal{L}$ (of volume 1) in which one can efficiently sample a discrete Gaussian with small parameter $\sigma < o(\sqrt{n})$, if not $\sigma = \text{polylog}(n)$ (with exponential smoothing $\sigma > \eta_{2^{-n}}(\mathcal{L})$)? And if so, do they and their dual have a large minimal distance? Note that quantumly, this question is related to the previous one via the reduction of Regev [Reg09]: decoding in the primal for a large radius gives Gaussian sampling in the dual for a small width. But a classical algorithm would be much preferable.

Concrete instantiation with simple lattices. Instantiated with $\mathbb{Z}^n$, our signature scheme has the advantage of not requiring any Gram-Schmidt or Cholesky decomposition, contrary to existing hash-then-sign signature schemes, and may therefore be of practical interest. It could also be reasonable to instantiate our KEM with the lattice of Barnes and Wall, by using the decoder of Micciancio and Nicolesi [MN08].

Module-LIP. Lastly, it also seems natural to explore structured variants of LIP, where both the lattice and the isometry should be structured. We note that for every ideal lattice in complex-multiplication number fields, a classical polynomial time algorithm for LIP is known [GS02; LS14]. Could the module variant be secure? Can our constructions gain a linear factor on key sizes from such a variant? And are there remarkably decodable lattices that are also modules over certain number fields? For example the repeated-difference lattices (Craig's lattices [CS13]) are ideal lattices in cyclotomic number fields with large minimal distances, but a polynomial decoding algorithm for them remains to be discovered.

Remark 171. The last two open questions were partially answered positively in a follow-up work [DPPW22]. In this work we build a competitive signature scheme, named HAWK, based on the simple lattice $\mathbb{Z}^n$, instantiated as a rank 2 module. In Section 10.8 we give a short summary of this work, and show that it improves upon FALCON in several ways.

10.2 LIP and self-reducibility

In this section we lay the foundation for using the Lattice Isomorphism Problem in cryptography. We present an average-case distribution for any quadratic form equivalence class, show how to sample from it, and conclude with a worst-case to average-case reduction. Note that the worst-case to average-case reduction is realized *within* an equivalence class $[\mathbf{Q}] = \{\mathbf{U}^\top \mathbf{Q}\mathbf{U} : \mathbf{U} \in \mathcal{GL}_n(\mathbb{Z})\}$. But first we start with some adaptation of discrete Gaussian sampling to the quadratic form setting.

10.2.1 Discrete Gaussians and sampling

Discrete Gaussian sampling has been fundamental to the development of lattice based cryptography, by allowing to return short or nearby lattice vectors without leaking information about the secret key [GPV08]. We rephrase the relevant definitions and propositions in the quadratic form language.

Distribution. For any quadratic form $\mathbf{Q} \in \mathcal{S}_n^{>0}$ we define the Gaussian function on $\mathbb{R}^n$ with parameter $s > 0$ and center $\mathbf{c} \in \mathbb{R}^n$ by

$$\forall \mathbf{x} \in \mathbb{R}^n, \rho_{\mathbf{Q},s,\mathbf{c}}(\mathbf{x}) := \exp(-\pi \left\|\mathbf{x} - \mathbf{c}\right\|_{\mathbf{Q}}^2 / s^2).$$

The discrete Gaussian distribution is obtained by restricting the continuous Gaussian distribution to a discrete lattice. In the quadratic form setting the discrete lattice will always be $\mathbb{Z}^n$, but with the geometry induced by the quadratic form. For any quadratic form $\mathbf{Q} \in \mathcal{S}_n^{>0}$ we define the discrete Gaussian distribution $\mathcal{D}_{\mathbf{Q},s,\mathbf{c}}$ with center $\mathbf{c} \in \mathbb{R}^n$ and parameter $s > 0$ by

$$\Pr_{X \sim \mathcal{D}_{\mathbf{Q},s,\mathbf{c}}}[X = \mathbf{x}] := \frac{\rho_{\mathbf{Q},s,\mathbf{c}}(\mathbf{x})}{\rho_{\mathbf{Q},s,\mathbf{c}}(\mathbb{Z}^n)} \text{ if } \mathbf{x} \in \mathbb{Z}^n, \text{ and 0 otherwise.}$$

If the center $\mathbf{c}$ is not denoted we have $\mathbf{c} = \mathbf{0}$. An important property of the discrete gaussian distribution is the smoothing parameter, i.e., how much gaussian noise $s > 0$ is needed to ‘smooth out’ the discrete structure.

Definition 172 (Smoothing Parameter). *For a quadratic form* $\mathbf{Q} \in \mathcal{S}_n^{>0}$ *and* $\epsilon > 0$ *we define the smoothing parameter* $\eta_\epsilon(\mathbf{Q})$ *as the minimal* $s > 0$ *such that* $\rho_{\mathbf{Q}^{-1},1/s}(\mathbb{Z}^n) \leq 1 + \epsilon$.

The smoothing parameter is a central quantity for gaussians over lattice, for example it permits to control the variations of $\rho_{\mathbf{Q},s,\mathbf{c}}(\mathbb{Z}^n)$ is over all centers $\mathbf{c}$.

Lemma 173 ([MR07]). *For any quadratic form* $\mathbf{Q} \in \mathcal{S}_n^{>0}$, $\epsilon > 0$, *center* $\mathbf{c} \in \mathbb{R}^n$ *and parameter* $s > \eta_\epsilon(\mathbf{Q})$ *we have:*

$$(1-\epsilon)\frac{s^n}{\sqrt{\det(\mathbf{Q})}} \leq \rho_{\mathbf{Q},s,\mathbf{c}}(\mathbb{Z}^n) \leq (1+\epsilon)\frac{s^n}{\sqrt{\det(\mathbf{Q})}}.$$

Note that the smoothing parameter $\eta_\epsilon(\mathbf{Q})$ is an invariant property of the similarity class $[\mathbf{Q}]$, and so we might also denote $\eta_\epsilon([\mathbf{Q}])$ for a similarity class. While computing or even approximating the exact smoothing parameter is hard, we can obtain sufficient bounds via the dual form.

Lemma 174 (Smoothing bound [MR07]). *For any* $\epsilon > 0$ *and any quadratic form* $\mathbf{Q} \in \mathcal{S}_n^{>0}$ *we have* $\eta_{2^{-n}}(\mathbf{Q}) \leq \sqrt{n}/\lambda_1(\mathbf{Q}^{-1})$, *and*

$$\eta_\epsilon(\mathbf{Q}) \leq \|\tilde{\mathbf{B}}_\mathbf{Q}\| \cdot \sqrt{\ln(2n(1+1/\epsilon))/\pi}.$$

Above the smoothing parameter the discrete gaussian distribution is in some sense 'well behaved' and we have the following tailbound that one would expect from a Gaussian distribution.

Lemma 175 (Tailbound [MR07, Lemma 4.4]). *For any quadratic form* $\mathbf{Q} \in \mathcal{S}_n^{>0}$, $\epsilon \in (0,1)$, *center* $\mathbf{c} \in \mathbb{R}^n$ *and parameter* $s \geq \eta_\epsilon(\mathbf{Q})$, *we have*

$$\Pr_{\mathbf{x} \sim \mathcal{D}_{\mathbf{Q},s,\mathbf{c}}}[\|\mathbf{x}-\mathbf{c}\|_\mathbf{Q} > s\sqrt{n}] \leq \frac{1+\epsilon}{1-\epsilon} \cdot 2^{-n}.$$

A constant factor above the smoothing parameter we can furthermore lower bound the min-entropy of the distribution.

Lemma 176 (Min-entropy [PR06]). *For any quadratic form* $\mathbf{Q} \in \mathcal{S}_n^{>0}$, *positive* $\epsilon > 0$, *center* $\mathbf{c} \in \mathbb{R}^n$, *parameter* $s \geq 2\eta_\epsilon(\mathbf{Q})$, *and for every* $\mathbf{x} \in \mathbb{Z}^n$, *we have*

$$\Pr_{X \sim \mathcal{D}_{\mathbf{Q},s,\mathbf{c}}}[X = \mathbf{x}] \leq \frac{1+\epsilon}{1-\epsilon} \cdot 2^{-n}.$$

Gaussian Sampling. While the discrete Gaussian distribution already is an important theoretical tool, for many practical purposes we want to actually sample (close to) the distribution in an efficient manner. In their breakthrough work Gentry et al. [GPV08] showed that Klein's [Kle00] randomized Babai's nearest plane algorithm does exactly that. Given a lattice basis one can sample statistically close to the discrete Gaussian distribution with parameters depending on the shortness of the (Gram-Schmidt) basis; a better reduced basis allows for a lower Gaussian width s. To simplify later proofs we use an exact sampling algorithm by Brakerski et al. [Bra+13].

Lemma 177 (Discrete Sampling [Bra+13, Lemma 2.3]). *There is a polynomial-time algorithm* $\mathbf{DiscreteSample}(\mathbf{Q}, s, \mathbf{c})$ *that given a quadratic form* $\mathbf{Q} \in \mathcal{S}_n^{>0}$, *center* $\mathbf{c} \in \mathbb{R}^n$, *and a parameter* $s \geq \|\tilde{\mathbf{B}}_{\mathbf{Q}}\| \cdot \sqrt{\ln(2n+4)/\pi}$, *returns a sample distributed as* $\mathcal{D}_{\mathbf{Q},s,\mathbf{c}}$.

10.2.2 An average-case distribution

First, we define our average-case distribution within an equivalence class $[\mathbf{Q}]$, which can be seen as an extension of the techniques used by Haviv and Regev [HR14] to show that LIP lies in SZK. While in their work they use a discrete Gaussian sampler [GPV08] to sample a generating set of the lattice, we extend this by a linear algebra step that returns a canonically distributed lattice basis — or, in our case, a quadratic form.

In hindsight, this algorithm appears very similar to the heuristic approach of [BM21], but the use of Gaussian sampling formally guarantees that the output distribution solely depends on the lattice and not on the specific input basis — or, in our case, depends only on the class of the input quadratic form.

We start with the linear algebra step, that, given a quadratic form and a set of short vectors of full rank, returns a well-reduced equivalent form.

Lemma 178 (Adapted from [MG02, Lemma 7.1]). *There is a polynomial time algorithm* $(\mathbf{R}, \mathbf{U}) \leftarrow \mathbf{Extract}(\mathbf{Q}, \mathbf{Y})$ *that on input a quadratic form* $\mathbf{Q}$, *and vectors* $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_m) \in \mathbb{Z}^{n \times m}$ *such that*

$\mathrm{rk}(\mathcal{L}(\mathbf{Y})) = n$, *outputs a transformation* $\mathbf{U} \in \mathcal{GL}_n(\mathbb{Z})$ *and a quadratic form* $\mathbf{R} = \mathbf{U}^\top \mathbf{Q} \mathbf{U}$ *equivalent to* $\mathbf{Q}$ *such that* $\left\| \tilde{\mathbf{B}}_{\mathbf{R}} \right\| \leq \max_i \|\mathbf{y}_i\|_{\mathbf{Q}}$.

Proof. First let $\mathbf{U} \in \mathcal{GL}_n(\mathbb{Z})$ be the unique transformation such that $\mathbf{T} = \mathbf{U}^{-1}\mathbf{Y}$ is the canonical upper-diagonal Hermite Normal Form of $\mathbf{Y}$. Let $\mathbf{R} = \mathbf{U}^\top \mathbf{Q} \mathbf{U}$ and note that $\mathbf{R}$ is equivalent to $\mathbf{Q}$. Denote the column vectors of $\mathbf{U}$ by $\mathbf{u}_1, \ldots, \mathbf{u}_n$. Let $p_1, \ldots, p_n$ be the pivot columns of $\mathbf{T}$. Because $\mathbf{T}$ is upper triangular and in Hermite Normal Form we have $\mathbf{y}_{p_i} = \sum_{j=1}^{i} \mathbf{T}_{j,p_i} \mathbf{u}_j$, where $\mathbf{T}_{i,p_i} \geq 1$. In particular we have that $\mathrm{span}(\mathbf{y}_{p_i}, \ldots, \mathbf{y}_{p_k}) = \mathrm{span}(\mathbf{u}_1, \ldots, \mathbf{u}_k)$. Let $\tilde{\mathbf{y}}_{p_i}$ and $\tilde{\mathbf{u}}_i$ be the i-th Gram-Schmidt vector of $(\mathbf{y}_{p_1}, \ldots, \mathbf{y}_{p_n})$ and $\mathbf{U}$ respectively w.r.t. $\mathbf{Q}$. Note that $\tilde{\mathbf{y}}_{p_i} = \mathbf{T}_{i,p_i} \cdot \tilde{\mathbf{u}}_i$, and thus $\|\tilde{\mathbf{u}}_i\|_{\mathbf{Q}} = \|\tilde{\mathbf{y}}_{p_i}\|_{\mathbf{Q}} / \mathbf{T}_{i,p_i} \leq \|\tilde{\mathbf{y}}_{p_i}\|_{\mathbf{Q}} \leq \|\mathbf{y}_{p_i}\|_{\mathbf{Q}}$. We conclude by $\left\| \tilde{\mathbf{B}}_{\mathbf{R}} \right\| = \max_i \|\tilde{\mathbf{u}}_i\|_{\mathbf{Q}} \leq \max_i \|\mathbf{y}_i\|_{\mathbf{Q}}$. □

For our final distribution to be well-defined we need that the extracted quadratic form only depends on the geometry of the input vectors, and not on the particular representative $\mathbf{Q}$.

Lemma 179. *Let* $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_m) \in \mathbb{Z}^{n \times m}$ *have full rank* n. *If* $(\mathbf{R}, \mathbf{U}) \leftarrow \mathbf{Extract}(\mathbf{Q}, \mathbf{Y})$, *and for some unimodular* $\mathbf{V} \in \mathcal{GL}_n(\mathbb{Z})$ *we have* $(\mathbf{R}', \mathbf{U}') \leftarrow \mathbf{Extract}(\mathbf{V}^\top \mathbf{Q} \mathbf{V}, \mathbf{V}^{-1}\mathbf{Y})$, *then* $\mathbf{R}' = \mathbf{R}$, *and* $\mathbf{U}' = \mathbf{V}^{-1} \cdot \mathbf{U}$.

Proof. From the canonicity of the Hermite Normal Form we immediately obtain that $(\mathbf{U}')^{-1}\mathbf{V}^{-1}\mathbf{Y} = \mathbf{T} = \mathbf{U}^{-1}\mathbf{Y}$, and thus $\mathbf{U}' = \mathbf{V}^{-1} \cdot \mathbf{U}$. It follows that $\mathbf{R}' = (\mathbf{V}^{-1} \cdot \mathbf{U})^\top \mathbf{V}^\top \mathbf{Q} \mathbf{V} (\mathbf{V}^{-1} \cdot \mathbf{U}) = \mathbf{U}^\top \mathbf{Q} \mathbf{U} = \mathbf{R}$. □

Now we can define our average-case distribution for a Gaussian parameter $s > 0$.

Definition 180. *Given a quadratic form* $\mathbf{Q} \in \mathcal{S}_n^{>0}$, *we define the Gaussian form distribution* $\mathcal{D}_s([\mathbf{Q}])$ *over* $[\mathbf{Q}]$ *with parameter* $s > 0$ *algorithmically as follows:*

1. *Let* $C := 1 - (1 + e^{-\pi})^{-1} > 0$, *and* $m := \left\lceil \frac{2n}{C} \right\rceil$. *Sample* m *vectors* $(\mathbf{y}_1, \ldots, \mathbf{y}_m) =: \mathbf{Y}$ *from* $\mathcal{D}_{\mathbf{Q},s}$. *Repeat until their span has full rank* n.

2. $(\mathbf{R}, \mathbf{U}) \leftarrow \mathbf{Extract}(\mathbf{Q}, \mathbf{Y})$.

3. *Return* $\mathbf{R}$.

Proof. We have to show that the distribution is well-defined over the equivalence class $[\mathbf{Q}]$, i.e., for any input representative $\mathbf{Q}' \in [\mathbf{Q}]$ the output distribution should be identical. Let $\mathbf{Q}' = \mathbf{V}^\top \mathbf{Q} \mathbf{V} \in [\mathbf{Q}]$. Note that step 1 only depends on the geometry of the vectors w.r.t. $\mathbf{Q}'$. Therefore if step 1 on input $\mathbf{Q}$ finishes with $\mathbf{Y}$, then on input $\mathbf{Q}'$ it finishes step 1 with $\mathbf{Y}' = \mathbf{V}^{-1}\mathbf{Y}$ with the same probability. By Lemma 179 step 2 extracts the same quadratic form $\mathbf{R}$ in both cases. So the distribution is independent of the input representative $\mathbf{Q}' \in [\mathbf{Q}]$, and thus it is well-defined over $[\mathbf{Q}]$. □

Given the algorithmic definition of $\mathcal{D}_s([\mathbf{Q}])$, an efficient sampling algorithm follows with only a few adaptations. Firstly, we need to efficiently sample from $\mathcal{D}_{\mathbf{Q},s}$ which puts some constraints on the parameter s depending on the reducedness of the representative $\mathbf{Q}$. Secondly, we need to show that step 1 does not have to be repeated very often. For this we require the additional constraint $s \geq \lambda_n(\mathbf{Q})$.

Algorithm 16: Sampling from $\mathcal{D}_s([\mathbf{Q}])$.

```
Data: A quadratic form $\mathbf{Q} \in \mathcal{S}_n^{>0}(\mathbb{Z}^n)$, and a parameter
      $s \geq \max\{\lambda_n(\mathbf{Q}), \|\tilde{\mathbf{B}}_\mathbf{Q}\| \cdot \sqrt{\ln(2n+4)/\pi}\}$.
Result: Sample $\mathbf{R} = \mathbf{U}^\top \mathbf{Q} \mathbf{U}$ from $\mathcal{D}_s([\mathbf{Q}])$, with a
        transformation $\mathbf{U} \in \mathcal{GL}_n(\mathbb{Z})$.
$C \leftarrow 1 - (1 + e^{-\pi})^{-1}$, $m \leftarrow \lceil \frac{2n}{C} \rceil$;
do
    Sample $\mathbf{y}_1, \ldots, \mathbf{y}_m \leftarrow \mathcal{D}_{\mathbf{Q},s}$ ;        // Using Lemma 177
    $\mathbf{Y} \leftarrow (\mathbf{y}_1, \ldots, \mathbf{y}_m)$;
while rk$(\mathbf{Y}) < n$;
$(R, U) \leftarrow$ Extract$(\mathbf{Q}, \mathbf{Y})$;
```

Lemma 181. *For any quadratic form $\mathbf{Q} \in \mathcal{S}_n^{>0}(\mathbb{Z})$, and parameter*

$$s \geq \max\{\lambda_n(\mathbf{Q}), \|\tilde{\mathbf{B}}_\mathbf{Q}\| \cdot \sqrt{\ln(2n+4)/\pi}\},$$

Algorithm 16 runs in expected polynomial time and returns $(\mathbf{R}, \mathbf{U}) \in [\mathbf{Q}] \times \mathcal{GL}_n(\mathbb{Z})$, where $\mathbf{R}$ is a sample from $\mathcal{D}_s([\mathbf{Q}])$, and $\mathbf{U}$, conditioned on $\mathbf{R}$, is uniform over $\mathrm{Isom}(\mathbf{Q}, \mathbf{R})$. *In particular $\mathbf{R} = \mathbf{U}^\top \mathbf{Q} \mathbf{U}$.*

Proof. By Lemmas 177 and 178 every step in Algorithm 16 runs in polynomial time. What remains is to show that the number of iterations is polynomially bounded in expectation. Let the random variable T be the number of iterations before a set of full rank vectors is found, we have to bound $\mathbb{E}[T]$.

If $\mathrm{rk}(\mathbf{y}_1, \ldots, \mathbf{y}_i) < n$, then because $s \geq \lambda_n(\mathbf{Q})$ we have by [HR14, Lemma 5.1] that every newly sampled vector $\mathbf{y}_{i+1} \leftarrow \mathcal{D}_{\mathbf{Q},s}$ is not in the span of $\mathbf{y}_1, \ldots, \mathbf{y}_i$ with constant probability at least $C := 1 - (1 + e^{-\pi})^{-1} > 0$. Let $m := \left\lceil \frac{2n}{C} \right\rceil$. The failure probability p_{fail} of not finding n linearly independent vectors is upper bounded by a binomial experiment with success probability C, where we reach at most $n-1$ wins in m trials. By Hoeffding's inequality we have the tail bound $p_{\text{fail}} \leq \exp(-2m \cdot (C - \frac{n-1}{m})^2) \leq \exp(-mC) \leq \exp(-2n) \leq e^{-2}$. The expected number of iterations $\mathbb{E}[T]$ is then bounded by the mean of a geometric distribution with success probability $1 - p_{\text{fail}}$, i.e.,

$$\mathbb{E}[T] \leq 1/(1 - p_{\text{fail}}) \leq 1/(1 - e^{-2}) < 2.$$

Suppose that the algorithm runs and finishes with a full rank matrix $\mathbf{Y}$, and returns $(\mathbf{R}, \mathbf{U}) \leftarrow \mathbf{Extract}(\mathbf{Q}, \mathbf{Y})$. For any automorphism $\mathbf{V} \in \mathrm{Aut}(\mathbf{Q})$, i.e., such that $\mathbf{V}^\top \mathbf{Q} \mathbf{V} = \mathbf{Q}$, it would have been just as likely that the final full rank matrix equalled $\mathbf{VY}$, because the samples from $\mathcal{D}_{\mathbf{Q},s}$ and the stopping condition only depend on the geometry of the vectors w.r.t. $\mathbf{Q}$. Then, by Lemma 179, we have:

$$\mathbf{Extract}(\mathbf{Q}, \mathbf{VY}) = \mathbf{Extract}((\mathbf{V}^{-1})^\top \mathbf{Q} \mathbf{V}^{-1}, \mathbf{VY}) = (\mathbf{R}, \mathbf{VU}),$$

and thus the algorithm would have returned $\mathbf{VU}$ with the same probability as $\mathbf{U}$, which makes the returned transformation uniform over the set of isomorphisms $\{\mathbf{VU} : \mathbf{V} \in \mathrm{Aut}(\mathbf{Q})\}$ from $\mathbf{Q}$ to $\mathbf{R}$. □

For (exponentially) large parameters s we can always efficiently sample from the average-case distribution by first LLL reducing the representative.

Lemma 182. *Given any quadratic form $\mathbf{Q} \in \mathcal{S}_n^{>0}(\mathbb{Z})$ we can sample from $\mathcal{D}_s([\mathbf{Q}])$ (together with a transformation) in polynomial time for $s \geq 2^{\Theta(n)} \cdot \lambda_n([\mathbf{Q}])$.*

Proof. Run the LLL algorithm on $\mathbf{Q}$ to obtain a representative $\mathbf{Q}' \in [\mathbf{Q}]$ for which $\|\tilde{\mathbf{B}}_{\mathbf{Q}'}\| \leq 2^{\Theta(n)} \cdot \lambda_n([\mathbf{Q}])$. Then apply Lemma 181 on $\mathbf{Q}'$.

For a sample $\mathbf{Q}''$, combining the unimodular transformations obtained from LLL and the sampling gives us a unimodular $\mathbf{U} \in \mathcal{GL}_n(\mathbb{Z})$ such that $\mathbf{Q}'' = \mathbf{U}^\top \mathbf{Q} \mathbf{U}$. □

Lemma 183. *For any quadratic form $\mathbf{Q} \in \mathcal{S}_n^{>0}$, parameter $\epsilon \in (0,1)$, and $s \geq \max\{\lambda_n(\mathbf{Q}), \eta_\epsilon(\mathbf{Q})\}$, we have*

$$\Pr_{\mathbf{Q}' \sim \mathcal{D}_s([\mathbf{Q}])}[\|\tilde{\mathbf{B}}_{\mathbf{Q}'}\| > s\sqrt{n}] \leq \frac{1+\epsilon}{1-\epsilon} \cdot 100n \cdot 2^{-n}.$$

Proof. Given full rank vectors $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_m) \in \mathbb{Z}^{n \times m}$ the extractor returns a quadratic form $\mathbf{Q}'$ such that $\|\tilde{\mathbf{B}}_{\mathbf{Q}'}\| \leq \max_i \|\mathbf{y}_i\|_{\mathbf{Q}}$ and thus we can just focus on the norms $\|\mathbf{y}_i\|_{\mathbf{Q}}$ of the sampled vectors. Let the random variable $T \geq 1$ be the number of iterations before a set of full rank vectors is found. From the proof of Lemma 181 we have $\mathbb{E}[T] \leq 2$. After t iterations we have sampled $t \cdot m$ vectors $\mathbf{x}_1, \ldots, \mathbf{x}_{t \cdot m}$, and by Lemma 175 we have $\|\mathbf{x}_i\| > s\sqrt{n}$ with probability at most $(1+\epsilon)/(1-\epsilon) \cdot 2^{-n}$ for each of them. By a union bound we conclude

$$\begin{aligned}
\Pr\left[\max_{1 \leq i \leq T \cdot m} \|\mathbf{y}_i\|_{\mathbf{Q}} > s\sqrt{n}\right] &= \sum_{t=1}^{\infty} \Pr[T = t] \cdot \Pr[\max_{1 \leq i \leq t \cdot m} \|\mathbf{y}_i\|_{\mathbf{Q}} > s\sqrt{n}] \\
&\leq \underbrace{\sum_{t=1}^{\infty} \Pr[T = t] \cdot t}_{\mathbb{E}[T]} \cdot m \cdot \frac{1+\epsilon}{1-\epsilon} \cdot 2^{-n} \\
&\leq 2 \cdot \left\lceil \tfrac{2n}{C} \right\rceil \cdot \frac{1+\epsilon}{1-\epsilon} \cdot 2^{-n} \leq 100n \cdot \frac{1+\epsilon}{1-\epsilon} \cdot 2^{-n}.
\end{aligned}$$

□

10.2.3 Average case LIP

The above definition of a distribution over a class which is efficiently sampleable from any representative of that class leads us to a natural average-case version of both versions of LIP. It is parametrized by a width parameter $s > 0$.

Definition 184 (Average-case search LIP: ac-sLIP$_s^{\mathbf{Q}}$). *For $s > 0$ and a quadratic form $\mathbf{Q} \in \mathcal{S}_n^{>0}$, the problem* ac-sLIP$_s^{\mathbf{Q}}$ *is, given a quadratic form sampled as $\mathbf{Q}' \leftarrow \mathcal{D}_s([\mathbf{Q}])$, to compute a unimodular $\mathbf{U} \in \mathcal{GL}_n(\mathbb{Z})$ such that $\mathbf{Q}' = \mathbf{U}^\top \mathbf{Q} \mathbf{U}$.*

Definition 185 (Average-case distinguishing LIP: ac-ΔLIP$_s^{\mathbf{Q}_0,\mathbf{Q}_1}$). *For two quadratic forms $\mathbf{Q}_0, \mathbf{Q}_1 \in \mathcal{S}_n^{>0}$ and parameter $s > 0$ the problem* ac-ΔLIP$_s^{\mathbf{Q}_0,\mathbf{Q}_1}$ *is, given a quadratic form sampled as $\mathbf{Q}' \leftarrow \mathcal{D}_s([\mathbf{Q}_b])$ where $b \in \{0,1\}$ is a uniform random bit, to recover b.*

Trivially the average-case variants can be reduced to their respective worst-case variants. In the following section we show that the reverse is also true.

10.2.4 A worst-case to average-case reduction

In general, lattice problems become easier when given a short basis and harder when given a long basis. Similarly, one would expect that ac-sLIP$_s^{\mathbf{Q}}$ and ac-ΔLIP$_s^{\mathbf{Q}_0,\mathbf{Q}_1}$ become harder when the parameter $s > 0$ increases. In fact, when s is large enough, the average-case problem becomes at least as hard as any worst-case instance, making the average-case and worst-case problems equivalent.

Lemma 186 (ac-sLIP$_s^{\mathbf{Q}} \geq$ wc-sLIP$^{\mathbf{Q}}$ for large s). *Given an oracle that solves* ac-sLIP$_s^{\mathbf{Q}}$ *for some $s \geq 2^{\Theta(n)} \cdot \lambda_n(\mathbf{Q})$ in time T_0 with probability $\epsilon > 0$, we can solve* wc-sLIP$^{\mathbf{Q}}$ *with probability at least ϵ in time $T = T_0 + poly(n, \log s)$.*

Proof. Given any (worst-case) instance $\mathbf{Q}' \in [\mathbf{Q}]$, apply Lemma 182 to sample $\mathbf{Q}'' \leftarrow \mathcal{D}_s([\mathbf{Q}])$ for some $s \geq 2^{O(n)} \cdot \lambda_n([\mathbf{Q}])$, together with a $\mathbf{U}''$ such that $\mathbf{Q}'' = \mathbf{U}''^\top \mathbf{Q}' \mathbf{U}''$. Note that $\mathcal{D}_s([\mathbf{Q}]) = \mathcal{D}_s([\mathbf{Q}'])$; we can therefore apply our ac-sLIP$_s^{\mathbf{Q}}$-oracle once to $\mathbf{Q}''$ and obtain $\mathbf{U} \in \mathcal{GL}_n(\mathbb{Z})$ such that $\mathbf{Q}'' = \mathbf{U}^\top \mathbf{Q} \mathbf{U}$. Now for $\mathbf{U}' := \mathbf{U}\mathbf{U}''^{-1} \in \mathcal{GL}_n(\mathbb{Z})$ we have:

$$\mathbf{U}'^\top \mathbf{Q} \mathbf{U}' = (\mathbf{U}''^{-1})^\top \mathbf{U}^\top \mathbf{Q} \mathbf{U} \mathbf{U}''^{-1} = (\mathbf{U}''^{-1})^\top \mathbf{Q}'' \mathbf{U}''^{-1} = \mathbf{Q}'.$$

So given an ac-sLIP$_s^{\mathbf{Q}}$-oracle we can solve wc-sLIP$^{\mathbf{Q}}$. □

To allow for more efficient schemes we would like to decrease the parameter $s > 0$ in the worst-case to average-case reduction. We can do so at the cost of a stronger lattice reduction algorithm than LLL.

Lemma 187. *Given an oracle that solves* $\text{ac-sLIP}_s^{\mathbf{Q}}$ *for some* $s \geq \lambda_n(\mathbf{Q})$ *in time* T_0 *with probability* $\epsilon > 0$*, we can solve* $\text{wc-sLIP}^{\mathbf{Q}}$ *with probability at least* $\frac{1}{2}$ *in time*

$$T = \frac{1}{\epsilon}(T_0 + \mathrm{poly}(n, \log s)) + C\left(n, \frac{s}{\lambda_n(\mathbf{Q}) \cdot \sqrt{\ln(2n+4)/\pi}}\right),$$

where $C(n, f)$ *is the cost of solving the Shortest Independent Vector Problem in dimension* n *(SIVP, [Reg09]) within an approximation factor of* f*.*

Proof. Since the f-approx-SIVP oracle returns n linearly independent vectors of norm at most $f \cdot \lambda_n(\mathbf{Q})$, we can construct an equivalent form $\mathbf{Q}' \in [\mathbf{Q}]$ with $\|\tilde{\mathbf{B}}_{\mathbf{Q}'}\| \leq f \cdot \lambda_n(\mathbf{Q})$, using Lemma 178. We do this once at cost $C(n, f)$. For $f := s/(\lambda_n(\mathbf{Q}) \cdot \sqrt{\ln(2n+4)/\pi})$ we obtain that $s \geq \|\tilde{\mathbf{B}}_{\mathbf{Q}'}\| \cdot \sqrt{\ln(2n+4)/\pi}$. Therefore using $\mathbf{Q}'$ we can sample efficiently from $\mathcal{D}_s([\mathbf{Q}])$. The rest of the proof is similar to that of Lemma 186. Additionally the reduction succeeds with some probability $\epsilon > 0$, so we need to repeat it $\frac{1}{\epsilon}$ times to obtain a success probability of at least $\frac{1}{2}$. Note that each additional sample can be computed in polynomial time from the same representative $\mathbf{Q}'$. □

Remark 188. Note that the overhead is entirely additive, in particular it does not suffer from the $\frac{1}{\epsilon}$ amplification. So, despite the reduction not being polynomial time, one can still afford extraordinary large overheads. Concretely, for example, a hardness of 2^{100} for f-SIVP would barely affect the hardness reduction, if the hardness of ac-sLIP is 2^{128}. This situation is quite different from the usual inefficient reductions found in the literature, where the overhead is multiplicative.

In Lemma 187, the SIVP oracle can be instantiated by a variant of the BKZ algorithm [Sch87]. With a sub-linear blocksize of $\beta := n/\log(n)$ we could decrease s to a quasi-polynomial factor $\exp(\log^2(n)) \cdot \lambda_n(\mathbf{Q})$, with only a subexponential additive cost to the reduction. For security based on exponential hardness (e.g., $T_0/\epsilon = \exp(\Omega(n))$) this would still be meaningful, while the lower parameters imply a more efficient scheme with only a poly-logarithmic bitlength for the integer entries of the manipulated matrices.

Going down to polynomial factors $s = \mathrm{poly}(n) \cdot \lambda_n(\mathbf{Q})$ (and hence single logarithmic integer bitlength) would require a linear blocksize $\beta := \Theta(n)$, and an exponential cost 2^{cn}. For small constants $c > 0$ such that cn is smaller than the security parameter the reduction would still be meaningful. However, for provable algorithms this constant c is generally too large to be useful. As we want to keep our reduction non-heuristic in this initial work, we will leave this regime for further research.

Using a similar strategy, one can also establish a worst-case to average-case reduction for ΔLIP. Note that, because it is a distinguishing problem, the advantage amplification requires $O(1/\alpha^2)$ calls to the average-case oracle.

Lemma 189 (ac-$\Delta\mathrm{LIP}_s^{\mathbf{Q}_0,\mathbf{Q}_1} \geq$ wc-$\Delta\mathrm{LIP}^{\mathbf{Q}_0,\mathbf{Q}_1}$ for large s). *Given an oracle that solves* ac-$\Delta\mathrm{LIP}_s^{\mathbf{Q}_0,\mathbf{Q}_1}$ *for some*

$$s \geq 2^{\Theta(n)} \cdot \max\{\lambda_n(\mathbf{Q}_0), \lambda_n(\mathbf{Q}_1)\}$$

in time T_0 with advantage $\alpha > 0$, we can solve wc-$\Delta\mathrm{LIP}^{\mathbf{Q}_0,\mathbf{Q}_1}$ *with advantage α in time $T = T_0 + \mathrm{poly}(n, \log s)$.*

Lemma 190. *Given an oracle that solves* ac-$\Delta\mathrm{LIP}_s^{\mathbf{Q}_0,\mathbf{Q}_1}$ *in time T_0 for some $s \geq \max\{\lambda_n(\mathbf{Q}_0), \lambda_n(\mathbf{Q}_1)\}$ with advantage $\alpha > 0$, we can solve* wc-$\Delta\mathrm{LIP}^{\mathbf{Q}_0,\mathbf{Q}_1}$ *with advantage at least $\frac{1}{4}$ in time*

$$\begin{aligned} T =& \frac{1}{\alpha^2}(T_0 + \mathrm{poly}(n, \log s)) \\ &+ C\left(n, \frac{s}{\max\{\lambda_n(\mathbf{Q}_0), \lambda_n(\mathbf{Q}_1)\} \cdot \sqrt{\ln(2n+4)/\pi}}\right), \end{aligned}$$

where $C(n, f)$ is the cost of solving the Shortest Independent Vector Problem in dimension n (SIVP, [Reg09]) within an approximation factor of f.

10.3 Zero Knowledge Proof of Knowledge

Recall from Section 2.7.2 that a ZKPoK protocol allows one to prove knowledge of a solution to some problem without revealing the solution. In our case, given public quadratic forms $\mathbf{Q}_0, \mathbf{Q}_1 \in \mathcal{S}_n^{>0}(\mathbb{Z})$

we want to show knowledge of a LIP solution $\mathbf{U} \in \mathcal{GL}_n(\mathbb{Z})$ such that $\mathbf{Q}_1 = \mathbf{U}^\top \mathbf{Q}_0 \mathbf{U}$, without revealing any information about $\mathbf{U}$. The protocol is defined in Figure 10.1.

On a high level, the protocol of Haviv and Regev [HR14], as well as ours, is very similar to protocols for other types of isomorphisms, in particular protocols for graph isomorphism [GMW91] and for code isomorphism [BMPS20].

A notable difference, however, is that both these type of protocols [GMW91; BMPS20] rely on the action of a finite group (permutations), allowing to show zero-knowledgness by uniformity of the distribution over an orbit. Since in our case, the group acting $\mathcal{GL}_n(\mathbb{Z})$ is not finite, not even compact, it cannot admit such uniform distribution. It is perhaps surprising to see that uniformity is in fact not required; our earlier defined average-case distribution can be used instead.

10.3.1 The Σ-protocol

Besides efficiency we have to check the standard properties of a sigma protocol as defined in Section 2.7.2: completeness, special soundness and honest-verifier zero-knowledge.

Efficiency and completeness. To show the efficiency of the prover we have to check that Algorithm 16 runs in polynomial time. Indeed, by Lemma 181, this is the case because

$$s \geq \max\left\{\lambda_n([\mathbf{Q}_0]), \|\tilde{\mathbf{B}}_{\mathbf{Q}_0}\| \cdot \sqrt{\ln(2n+4)/\pi}\right\}.$$

To show the efficiency of the verifier we have to show that the check $\mathbf{W} \in \mathcal{GL}_n(\mathbb{Z})$ can be done efficiently. This is the case, since $\mathbf{W} \in \mathcal{GL}_n(\mathbb{Z})$ if and only if $\mathbf{W}$ is integral and $\det(\mathbf{W}) = \pm 1$, both of which are easy to check in polynomial time.

To prove completeness of the protocol Σ, we have to show that the verifier accepts, whenever the prover executes the protocol honestly. If the prover is honest then we have $\mathbf{W} := \mathbf{U}^{-c} \cdot \mathbf{V} \in \mathcal{GL}_n(\mathbb{Z})$ because $\mathbf{U}$ and $\mathbf{V}$ are both unimodular by definition. Additionally we have

$$\mathbf{Q}' = \mathbf{V}^\top \mathbf{Q}_0 \mathbf{V} = \underbrace{(\mathbf{V}^\top (\mathbf{U}^{-c})^\top)}_{\mathbf{W}^\top} \underbrace{((\mathbf{U}^c)^\top \mathbf{Q}_0 \mathbf{U}^c)}_{\mathbf{Q}_c} \underbrace{(\mathbf{U}^{-c}\mathbf{V})}_{\mathbf{W}} = \mathbf{W}^\top \mathbf{Q}_c \mathbf{W},$$

Zero Knowledge Proof of Knowledge Σ

Consider two equivalent public quadratic forms $\mathbf{Q}_0, \mathbf{Q}_1 \in \mathcal{S}_n^{>0}(\mathbb{Z})$ and a secret unimodular $\mathbf{U} \in \mathcal{GL}_n(\mathbb{Z})$ such that $\mathbf{Q}_1 = \mathbf{U}^\top \mathbf{Q}_0 \mathbf{U}$. Given the public parameter

$$s \geq \max\left\{\lambda_n([\mathbf{Q}_0]), \max\left\{\|\tilde{\mathbf{B}}_{\mathbf{Q}_0}\|, \|\tilde{\mathbf{B}}_{\mathbf{Q}_1}\|\right\} \cdot \sqrt{\ln(2n+4)/\pi}\right\},$$

we define the following protocol Σ that gives a zero-knowledge proof of knowledge of an isomorphism between $\mathbf{Q}_0$ and $\mathbf{Q}_1$:

Prover | Verifier

Sample $\mathbf{Q}' \leftarrow \mathcal{D}_s([\mathbf{Q}_0])$ by Alg. 16, together with $\mathbf{V}$ s.t. $\mathbf{Q}' = \mathbf{V}^\top \mathbf{Q}_0 \mathbf{V}$

$$\xrightarrow{\mathbf{Q}'}$$

Sample $c \leftarrow \mathcal{U}(\{0,1\})$

$$\xleftarrow{c}$$

Compute $\mathbf{W} = \mathbf{U}^{-c} \cdot \mathbf{V}$

$$\xrightarrow{\mathbf{W}}$$

Check if $\mathbf{W} \in \mathcal{GL}_n(\mathbb{Z})$, and $\mathbf{Q}' = \mathbf{W}^\top \mathbf{Q}_c \mathbf{W}$.

Figure 10.1: A zero knowledge proof of knowledge sigma protocol for knowledge of a solution to search LIP.

and thus the verifier accepts.

Special Soundness. For special soundness, we have to show that two accepting conversations with the same initial message, but a different challenge, allow to compute an isomorphism from $\mathbf{Q}_0$ to $\mathbf{Q}_1$. Suppose we have two such accepting conversations $(\mathbf{Q}', 0, \mathbf{W}_0)$ and $(\mathbf{Q}', 1, \mathbf{W}_1)$ of Σ where the first message $\mathbf{Q}'$ is identical. The acceptance implies that $\mathbf{W}_0, \mathbf{W}_1 \in \mathcal{GL}_n(\mathbb{Z})$ and $\mathbf{W}_0^\top \mathbf{Q}_0 \mathbf{W}_0 = \mathbf{Q}' = \mathbf{W}_1^\top \mathbf{Q}_1 \mathbf{W}_1$. Therefore

$\mathbf{U}' := \mathbf{W}_0\mathbf{W}_1^{-1} \in \mathcal{GL}_n(\mathbb{Z})$ gives an isomorphism from $\mathbf{Q}_0$ to $\mathbf{Q}_1$, as

$$\begin{aligned}\mathbf{U}'^{\top}\mathbf{Q}_0\mathbf{U}' &= (\mathbf{W}_1^{-1})^{\top}(\mathbf{W}_0^{\top}\mathbf{Q}_0\mathbf{W}_0)\mathbf{W}_1^{-1}\\ &= (\mathbf{W}_1^{-1})^{\top}(\mathbf{W}_1^{\top}\mathbf{Q}_1\mathbf{W}_1)\mathbf{W}_1^{-1} = \mathbf{Q}_1.\end{aligned}$$

We conclude that Σ has the special soundness property.

Honest-verifier zero-knowledge. To show that a honest verifier cannot learn anything from the interaction, we must show that there exists an efficient simulator that can produce accepting conversations following the same distribution, but *without* knowledge of a secret isomorphism between $\mathbf{Q}_0$ and $\mathbf{Q}_1$. We create such a simulator that given the public input $\mathbf{Q}_0, \mathbf{Q}_1$, outputs an accepting conversation with the same probability distribution as it would have been between a honest prover and verifier. Note that the first message $\mathbf{Q}'$ is always distributed as $\mathcal{D}_s([\mathbf{Q}_0])$, the challenge c as $\mathcal{U}(\{0,1\})$, and $\mathbf{V}$ is uniform over the set of isomorphisms from $\mathbf{Q}_0$ to $\mathbf{Q}'$, by Lemma 181. Because $\mathbf{U}$ is an isomorphism from $\mathbf{Q}_0$ to $\mathbf{Q}_1$ we have, given the challenge c, that $\mathbf{W} = \mathbf{U}^{-c}\cdot\mathbf{V}$ is uniform over the set of isomorphisms from $\mathbf{Q}_c$ to $\mathbf{Q}'$.

To simulate this we first sample the uniformly random challenge $c \leftarrow \mathcal{U}(\{0,1\})$. If $c = 0$ we can proceed the same as in Σ itself, e.g., sample $\mathbf{Q}' \leftarrow \mathcal{D}_s([\mathbf{Q}_0])$ using Algorithm 16, together with a $\mathbf{V}$ such that $\mathbf{Q}' = \mathbf{V}^{\top}\mathbf{Q}_0\mathbf{V}$, and set $\mathbf{W} := \mathbf{V}$. The final conversation $(\mathbf{Q}', 0, \mathbf{W})$ is accepting and follows by construction the same distribution as during an honest execution conditioned on challenge $c = 0$.

If $c = 1$ we use the fact that $[\mathbf{Q}_0] = [\mathbf{Q}_1]$, and that we can use Algorithm 16 with representative $\mathbf{Q}_1$ as input instead of $\mathbf{Q}_0$. So again we obtain $\mathbf{Q}' \leftarrow \mathcal{D}_s([\mathbf{Q}_1]) = \mathcal{D}_s([\mathbf{Q}_0])$ following the same distribution, but now together with a unimodular $\mathbf{W} \in \mathcal{GL}_n(\mathbb{Z})$ such that $\mathbf{Q}' = \mathbf{W}^{\top}\mathbf{Q}_1\mathbf{W}$. The conversation $(\mathbf{Q}', 1, \mathbf{W})$ is accepting by construction, and $\mathbf{Q}'$ follows the same distribution $\mathcal{D}_s([\mathbf{Q}_0])$. Additionally by Lemma 181 the transformation $\mathbf{W}$ is indeed uniform over the set of isomorphisms from $\mathbf{Q}_1$ to $\mathbf{Q}'$.

We conclude that Σ has the honest-verifier zero-knowledge property.

10.3.2 Identification scheme

The Zero Knowledge Proof of Knowledge in the previous section is worst-case in the sense that, given any two equivalent forms $\mathbf{Q}_0, \mathbf{Q}_1 \in \mathcal{S}_n^{>0}(\mathbb{Z})$ and a secret isomorphism $\mathbf{U} \in \mathcal{GL}_n(\mathbb{Z})$ from $\mathbf{Q}_0$ to $\mathbf{Q}_1$, we can show knowledge of such an isomorphism. However, to turn this Σ-protocol into an identification scheme (see [Dam02]), we need to define a distribution of $\mathbf{U} \in \mathcal{GL}_n(\mathbb{Z})$ (or alternatively of $\mathbf{Q}_1$ w.r.t $\mathbf{Q}_0$). Finding an isomorphism between $\mathbf{Q}_0$ and $\mathbf{Q}_1$ is at most as hard as solving either ac-sLIP$_s^{\mathbf{Q}_0}$ or ac-sLIP$_s^{\mathbf{Q}_1}$ for parameter s as in Σ. Therefore, a natural choice is to have $\mathbf{Q}_1$ distributed according to $\mathcal{D}_{s'}([\mathbf{Q}_0])$ for some parameter $s' \geq \max\{\lambda_n([\mathbf{Q}_0]), \|\tilde{\mathbf{B}}_{\mathbf{Q}_0}\| \cdot \sqrt{\ln(2n+4)/\pi}\}$, which we can efficiently sample from using Algorithm 16. The security of our identification scheme is then solely based on the hardness of ac-sLIP$_{s'}^{\mathbf{Q}_0}$.

10.4 Key Encapsulation Mechanism

In this section we construct a Key Encapsulation Mechanism (KEM) with a security proof based on the hardness of ΔLIP. See Figure 10.2 for the scheme. In short, we will need a quadratic form $\mathbf{S}$ along with an efficient decoder up to some radius $\rho < \lambda_1(\mathbf{S})/2$. The public key will consist of a long equivalent form $\mathbf{P} := \mathbf{U}^\top \mathbf{S} \mathbf{U} \leftarrow \mathcal{D}_s([\mathbf{S}])$, while the unimodular transformation $\mathbf{U}$ will be the secret key. Knowledge of the transformation $\mathbf{U}$ allows to decode w.r.t. $\mathbf{P}$ via $\mathbf{S}$; without any loss in decoding performance. The generated key will be a random error $\mathbf{e}$ of norm $\|\mathbf{e}\|_{\mathbf{P}} \leq \rho$, and it can be encapsulated as the syndrome $\bar{\mathbf{e}} := \mathbf{e} \bmod \mathbb{Z}^n \in [0,1)^n$. The receiver with knowledge of the secret transformation $\mathbf{U}$ can recover $\mathbf{e}$ by decoding via $\mathbf{S}$. The correctness follows from the fact that the decoding is unique due to $\rho < \lambda_1(\mathbf{S})/2$.

For the security we assume that it is (computationally) hard to differentiate between $\mathbf{P} \leftarrow \mathcal{D}_s([\mathbf{S}])$ and some random sample $\mathbf{R} \leftarrow \mathcal{D}_s([\mathbf{Q}])$ from a special class $[\mathbf{Q}]$, corresponding to a lattice admitting a dense sublattice. This assumption allows us to replace $\mathbf{P}$ by $\mathbf{R}$ in the security proof, which completely breaks the uniqueness of the decoding. That is, the syndrome $\bar{\mathbf{e}}$ has many (say $\exp(\Omega(\lambda))$) nearby points w.r.t. $\mathbf{R}$, and retrieving the exact original point becomes statistically hard.

Key Encapsulation Scheme

Let $\rho < \lambda_1(\mathbf{S})/2$ and let $\mathbf{S} \in \mathcal{S}_n^{>0}(\mathbb{Z})$ be a quadratic form with an efficient decoder **Decode** with decoding radius ρ. Let $\mathcal{E} : \frac{1}{q}\mathbb{Z}^n \times \{0,1\}^z \to \{0,1\}^\ell$ be a $(\ell, \mathrm{negl}(n))$-extractor for some $\ell = \Theta(n)$. Given the public parameters

$$\begin{aligned} s &\geq \max\{\lambda_n(\mathbf{S}), \|\tilde{\mathbf{B}}_{\mathbf{S}}\| \cdot \sqrt{\ln(2n+4)/\pi}\}, \text{ and} \\ q &:= \left\lceil \frac{s \cdot n}{\rho} \cdot \sqrt{\ln(2n+4)/\pi} \right\rceil, \end{aligned}$$

we define the KEM $\mathcal{K} := (\mathbf{Gen}, \mathbf{Encaps}, \mathbf{Decaps})$ as follows:

- $(pk, sk) \leftarrow \mathbf{Gen}(1^n)$: on input 1^n do:
  1. Sample $\mathbf{P} \leftarrow \mathcal{D}_s([\mathbf{S}])$ using Alg. 16, together with $\mathbf{U}$ such that $\mathbf{P} = \mathbf{U}^\top \mathbf{S} \mathbf{U}$.
  2. Output $(pk, sk) = (\mathbf{P}, \mathbf{U})$.
- $(c, k) \leftarrow \mathbf{Encaps}(pk)$: on input a public key $\mathbf{P} = pk$ do:
  1. Sample $\mathbf{e} \leftarrow \frac{1}{q}\mathcal{D}_{\mathbf{P}, q\rho/\sqrt{n}} \in \frac{1}{q}\mathbb{Z}^n$ using Lemma 177.
  2. Compute $\mathbf{c} \leftarrow \mathbf{e} \bmod \mathbb{Z}^n$ s.t. $\mathbf{c} \in \mathbb{T}_q^n = \{0, \frac{1}{q}, \ldots, \frac{q-1}{q}\}^n$.
  3. Sample a random extractor seed $Z \leftarrow \{0,1\}^z$.
  4. Compute $k \leftarrow \mathcal{E}(\mathbf{e}, Z)$.
  5. Output (c, k) where $c := (\mathbf{c}, Z)$.
- $k \leftarrow \mathbf{Decaps}(sk, c)$: on input $c = (\mathbf{c}, Z)$ and a secret key $\mathbf{U} := sk$ do:
  1. Compute $\mathbf{y} \leftarrow \mathbf{Decode}(\mathbf{S}, \mathbf{U}\mathbf{c})$ s.t. $\|\mathbf{y} - \mathbf{U}\mathbf{c}\|_{\mathbf{S}} \leq \rho$, output $\perp$ on failure.
  2. Compute $k \leftarrow \mathcal{E}(\mathbf{c} - \mathbf{U}^{-1}\mathbf{y}, Z)$.
  3. Output k.

Figure 10.2: A key encapsulation scheme based on LIP.

Efficiency and correctness. We consider the efficiency and correctness of the KEM $\mathcal{K} := (\mathbf{Gen}, \mathbf{Encaps}, \mathbf{Decaps})$ instantiated with quadratic form $\mathbf{S} \in \mathcal{S}_n^{>0}(\mathbb{Z})$ and public parameter

$$s \geq \max\{\lambda_n(\mathbf{S}), \|\tilde{\mathbf{B}}_\mathbf{S}\| \cdot \sqrt{\ln(2n+4)/\pi}\}.$$

By the above constraint on s, Algorithm 16 will run in polynomial-time by Lemma 181. Furthermore by Lemma 183 we have with overwhelming probability that

$$q\rho/\sqrt{n} \geq s\sqrt{n} \cdot \sqrt{\ln(2n+4)/\pi} \geq \|\tilde{\mathbf{B}}_\mathbf{P}\| \cdot \sqrt{\ln(2n+4)/\pi},$$

and thus we can efficiently sample from $\mathcal{D}_{\mathbf{P}, q\rho/\sqrt{n}}$ by Lemma 177.

In order to prove correctness, note that in the key encapsulation algorithm the sampled error $\mathbf{e}$ has norm at most $\|\mathbf{e}\|_\mathbf{P} \leq \rho$ except with negligible probability by Lemma 175. We denote the encapsulated key by $k := \mathcal{E}(\mathbf{e}, Z)$, where Z denotes the randomness extractor's seed. Because $\rho < \lambda_1(\mathbf{S})/2$ the vector $\mathbf{x} := \mathbf{c} - \mathbf{e} \in \mathbb{Z}^n$ is the unique closest vector to $\mathbf{c}$ with respect to $\mathbf{P}$, which makes $\mathbf{Ux}$ the unique closest vector to $\mathbf{Uc}$ with respect to $\mathbf{S} = (\mathbf{U}^{-1})^\top \mathbf{P} \mathbf{U}^{-1}$. In the decapsulation the decoder computes the unique vector $\mathbf{y}$ at distance at most ρ from $\mathbf{Uc}$, which implies that $\mathbf{y} = \mathbf{Ux}$. So indeed the output $k' := \mathcal{E}(\mathbf{c} - \mathbf{U}^{-1}\mathbf{y}, Z) = \mathcal{E}(\mathbf{c} - \mathbf{x}, Z) = \mathcal{E}(\mathbf{e}, Z) = k$ equals the encapsulated key with overwhelming probability.

CPA security. To show that our KEM is CPA-secure we fall back to a lossy trapdoor argument as in [PW11]. Under the hardness of decisional LIP we can replace our unique ρ-decodable quadratic form by one that is far from uniquely decodable. For the latter it is enough to have a dense sublattice.

Lemma 191. *Let $\mathbf{Q} \in \mathcal{S}_n^{>0}(\mathbb{Z})$ be a quadratic form with a rank r sublattice $\mathbf{D} \cdot \mathbb{Z}^r \subset \mathbb{Z}^n$. For positive $\epsilon > 0$, center $\mathbf{c} \in \mathbb{R}^n$, parameter $\rho \geq \sqrt{n} \cdot 2\eta_\epsilon([\mathbf{D}^\top \mathbf{Q} \mathbf{D}])$, and for every $\mathbf{x} \in \mathbb{Z}^n$ we have*

$$\Pr_{X \sim \mathcal{D}_{\mathbf{Q}, \rho/\sqrt{n}, \mathbf{c}}}[X = \mathbf{x}] \leq \frac{1+\epsilon}{1-\epsilon} \cdot 2^{-r}.$$

Proof. Let $\mathbf{y} := \mathbf{x} - \mathbf{c} \in \mathbb{R}^n$, and decompose $\mathbf{y} =: \mathbf{y}_\mathbf{D} + \mathbf{y}_{\mathbf{D}^\perp}$ where $\mathbf{y}_\mathbf{D} \in \operatorname{span}(\mathbf{D}\mathbb{Z}^r)$, and $\mathbf{y}_{\mathbf{D}^\perp}$ is orthogonal to $\mathbf{y}_\mathbf{D}$ w.r.t $\mathbf{Q}$. Then, writing

$s := \rho/\sqrt{n}$, we have

$$\begin{aligned}\Pr_{X\sim\mathcal{D}_{\mathbf{Q},s,\mathbf{c}}}[X=\mathbf{x}] &= \frac{\rho_{\mathbf{Q},s,\mathbf{c}}(\mathbf{x})}{\rho_{\mathbf{Q},s,\mathbf{c}}(\mathbb{Z}^n)} = \frac{\rho_{\mathbf{Q},s}(\mathbf{y})}{\rho_{\mathbf{Q},s}(\mathbf{y}+\mathbb{Z}^n)} \leq \frac{\rho_{\mathbf{Q},s}(\mathbf{y})}{\rho_{\mathbf{Q},s}(\mathbf{y}+\mathbf{D}\mathbb{Z}^r)} \\ &= \frac{\rho_{\mathbf{Q},s}(\mathbf{y}_{\mathbf{D}^\perp})\cdot\rho_{\mathbf{Q},s}(\mathbf{y}_{\mathbf{D}})}{\rho_{\mathbf{Q},s}(\mathbf{y}_{\mathbf{D}^\perp})\cdot\rho_{\mathbf{Q},s}(\mathbf{y}_{\mathbf{D}}+\mathbf{D}\mathbb{Z}^r)} = \frac{\rho_{\mathbf{Q},s}(\mathbf{y}_{\mathbf{D}})}{\rho_{\mathbf{Q},s}(\mathbf{y}_{\mathbf{D}}+\mathbf{D}\mathbb{Z}^r)}.\end{aligned}$$

Since we can write $\mathbf{y}_{\mathbf{D}} = \mathbf{D}\mathbf{z}$ for some $\mathbf{z}\in\mathbb{R}^r$, the above equals $\Pr_{X\sim\mathcal{D}_{\mathbf{D}^\top\mathbf{Q}\mathbf{D},s,\mathbf{z}}}[X=\mathbf{0}]$, which, by Lemma 176, is bounded by $\frac{1+\epsilon}{1-\epsilon}\cdot 2^{-r}$ because $s \geq 2\eta_\epsilon([\mathbf{D}^\top\mathbf{Q}\mathbf{D}])$.

□

Theorem 192. *We consider the KEM* $\mathcal{K} := (\mathbf{Gen},\mathbf{Encaps},\mathbf{Decaps})$ *instantiated with quadratic form* $\mathbf{S}\in\mathcal{S}_n^{>0}(\mathbb{Z})$*, decoding radius* ρ*, and public key parameter* $s>0$*. Let* $\mathbf{Q}\in\mathcal{S}_n^{>0}(\mathbb{Z})$ *be a quadratic form with a dense rank* $r=\Theta(n)$ *sublattice* $D\mathbb{Z}^r\subset\mathbb{Z}^n$*, in particular such that* $\eta_{\frac{1}{2}}(\mathbf{D}^\top\mathbf{Q}\mathbf{D})\leq\rho/(2\sqrt{n})$*. Then* $\mathcal{K}$ *is CPA-secure if* ac-$\Delta\mathrm{LIP}_s^{\mathbf{S},\mathbf{Q}}$ *is hard.*

Proof. Let $\mathcal{A}$ be a probabilistic polynomial-time adversary. We present two games $\mathbf{Game}_1$ and $\mathbf{Game}_2$, where $\mathbf{Game}_1$ is the regular CPA-security game with the original scheme, and $\mathbf{Game}_2$ is almost identical but with the only change that the public key is drawn from $\mathcal{D}_s([\mathbf{Q}])$ instead of $\mathcal{D}_s([\mathbf{S}])$. By the hardness of ac-$\Delta\mathrm{LIP}_s^{\mathbf{S},\mathbf{Q}}$ the two games are computationally indistinguishable, and due to the dense sublattice we can conclude that winning $\mathbf{Game}_2$ with a non-negligible advantage is statistically impossible.

Let the key-size $\ell=\Theta(n)$ be such that $\ell\leq r-\log_2(3)$. The original KEM CPA game $\mathbf{Game}_1$ is as follows [KL20]:

- $\mathbf{Gen}(1^n)$ is run to obtain a public key $pk=\mathbf{P}$. Then $\mathbf{Encaps}(pk)$ is run to generate (c,k) with $k\in\{0,1\}^\ell$.
- A uniform bit $b\in\{0,1\}$ is chosen. If $b=0$, set $\hat{k}:=k$, if $b=1$, choose a uniform $\hat{k}\in\{0,1\}^\ell$.
- Given $(pk, c=(\mathbf{c},Z),\hat{k})$ the adversary $\mathcal{A}$ wins the experiment if b is guessed correctly.

The only difference between $\mathbf{Game}_1$ and $\mathbf{Game}_2$ is that in $\mathbf{Game}_2$ we sample the public key $\mathbf{P}$ from $\mathcal{D}_s([\mathbf{Q}])$ instead of $\mathcal{D}_s([\mathbf{S}])$. Note that $\mathbf{Game}_1$ and $\mathbf{Game}_2$ both only use public information and thus by the hardness of ac-$\Delta\mathrm{LIP}_s^{\mathbf{S},\mathbf{Q}}$ the two are computationally indistinguishable by $\mathcal{A}$.

Now we take a look at $\mathbf{Game}_2$. Consider the output $(c = (\mathbf{c}, Z), k) \leftarrow \mathbf{Encaps}(pk)$ where $pk := \mathbf{Q}' \in [\mathbf{Q}]$. For any fixed $\mathbf{c}$ we have by construction that $k := \mathcal{E}(\mathbf{e}, Z)$, where $\mathbf{e} \leftarrow \frac{1}{q}\mathcal{D}_{\mathbf{Q}',q\rho/\sqrt{n}}$ under the condition that $\mathbf{e} = \mathbf{c} \bmod \mathbb{Z}^n$. Equivalently we could say that $\mathbf{e} \leftarrow \mathbf{c} - \mathcal{D}_{\mathbf{Q}',\rho/\sqrt{n},\mathbf{c}}$, then, by Lemma 191, we know that $\mathbf{e}$ has a min-entropy of at least $r - \log_2(3) \geq l$, and thus $k := \mathcal{E}(\mathbf{e}, Z) \in \{0,1\}^\ell$ is negligibly close to uniform and independent of c. So, in $\mathbf{Game}_2$ we have that $\hat{k}$ is negligibly close to uniform, independent of c and the choice of $b \in \{0,1\}$, making it impossible for $\mathcal{A}$ to guess b with non-negligible advantage.

□

10.5 Signature scheme

Similar to the Key Encapsulation Mechanism we propose in Figure 10.3 a *hash-then-sign* signature scheme based on ΔLIP. The main requirement is a quadratic form $\mathbf{S}$ along with an efficient discrete Gaussian sampling algorithm of smallish width $\rho/\sqrt{n} \geq \eta_{2^{-\Theta(n)}}(\mathbf{S})$.

Again the public key will consist of some lesser reduced form $\mathbf{P} := \mathbf{U}^\top\mathbf{S}\mathbf{U} \leftarrow \mathcal{D}_s([\mathbf{S}])$ equivalent to $\mathbf{S}$, where the unimodular transformation $\mathbf{U}$ will form the secret key. To sign a message we use a full domain hash to obtain a uniform coset $\mathbf{t} + \mathbb{Z}^n$, The signature then consists of a nearby vector $\boldsymbol{\sigma} \leftarrow \mathcal{D}_{\mathbf{P},\rho/\sqrt{n},\mathbf{t}}$ w.r.t. the form $\mathbf{P}$. The nearby vector is obtained via $\mathbf{S}$ by using the secret transformation $\mathbf{U}$.

The security assumption is similar, but in some way dual to that of the KEM. Again assume that it is computationally hard to differentiate between $\mathbf{P}$ and some special class of forms $[\mathbf{Q}]$; however in this case $\mathbf{Q}$ must admit a sparse projection (equivalently, their dual should contain a dense lattice). The sparsity implies that a uniformly random target $\mathbf{t}$ does not have a nearby vector with overwhelming probability, making the signage vacuously hard.

Signature Scheme

Let $\mathbf{S} \in \mathcal{S}_n^{>0}(\mathbb{Z})$ be a quadratic form together with a sampling algorithm **DiscreteSample** that allows to sample statistically close to $\mathcal{D}_{\mathbf{P},\rho/\sqrt{n}}(\mathbf{t} + \mathbb{Z}^n)$ for some parameter $\rho/\sqrt{n} \geq \eta_{2^{-\Theta(n)}}([\mathbf{S}])$ and any target $\mathbf{t} \in \mathbb{T}_q^n$. Let $\mathcal{H} : \mathcal{M} \to \mathbb{T}_q^n$ be a full domain hash function (modeled as a random oracle). Given the public parameters

$$s \geq \max\{\lambda_n(\mathbf{S}), \|\tilde{\mathbf{B}}_{\mathbf{S}}\| \cdot \sqrt{\ln(2n+4)/\pi}\}, \text{ and}$$
$$q := \left\lceil \frac{s \cdot n}{\rho} \cdot \sqrt{\ln(2n+4)/\pi} \right\rceil,$$

we define the signature scheme $\mathcal{S} := (\mathbf{Gen}, \mathbf{Sign}, \mathbf{Verify})$ as follows:

- $(pk, sk) \leftarrow \mathbf{Gen}(1^n)$: on input 1^n do:
  1. Sample $\mathbf{P} \leftarrow \mathcal{D}_s([\mathbf{S}])$ using Alg. 16, together with $\mathbf{U}$ s.t. $\mathbf{P} = \mathbf{U}^\top \mathbf{S} \mathbf{U}$.
  2. Output $(pk, sk) = (\mathbf{P}, \mathbf{U}) \in \mathcal{S}_n^{>0}(\mathbb{Z}) \times \mathcal{GL}_n(\mathbb{Z})$.
- $\boldsymbol{\sigma} \leftarrow \mathbf{Sign}(sk, m)$: on input a message m and a secret key $\mathbf{U} := sk$ do:
  1. Compute $\mathbf{t} \leftarrow \mathcal{H}(m)$.
  2. Sample $\boldsymbol{\sigma}' \leftarrow \mathcal{D}_{\mathbf{S},\rho/\sqrt{n},\mathbf{Ut}}$ using **DiscreteSample**.
  3. Compute $\boldsymbol{\sigma} \leftarrow \mathbf{U}^{-1}\boldsymbol{\sigma}'$.
  4. Output $\boldsymbol{\sigma} \in \mathbb{Z}^n$.
- $b := \mathbf{Verify}(pk, m, \boldsymbol{\sigma})$: on input a public key $\mathbf{P} = pk$, a message m and a signature $\boldsymbol{\sigma}$ do:
  1. Compute $\mathbf{t} \leftarrow \mathcal{H}(m)$.
  2. If $\boldsymbol{\sigma} \in \mathbb{Z}^n$, and $\|\mathbf{t} - \boldsymbol{\sigma}\|_{\mathbf{P}} \leq \rho$, output $b = 1$.
  3. Otherwise, output $b = 0$.

Figure 10.3: A signature scheme based on LIP.

Correctness. To prove correctness, we mainly have to check that the returned signature $\boldsymbol{\sigma} \in \mathbb{Z}^n$ is indeed close to $\mathbf{t} := \mathcal{H}(m)$ w.r.t $\mathbf{P}$. Because $\mathbf{P} = \mathbf{U}^\top \mathbf{S} \mathbf{U}$ we have:

$$\|\boldsymbol{\sigma} - \mathbf{t}\|_{\mathbf{P}} = \|\mathbf{U}(\boldsymbol{\sigma} - \mathbf{t})\|_{\mathbf{S}} = \|\boldsymbol{\sigma}' - \mathbf{U}\mathbf{t}\|_{\mathbf{S}},$$

and by Lemma 175 we have with overwhelming probability that

$$\|\boldsymbol{\sigma} - \mathbf{t}\|_{\mathbf{P}} = \|\boldsymbol{\sigma}' - \mathbf{U}\mathbf{t}\|_{\mathbf{S}} \leq \rho/\sqrt{n} \cdot \sqrt{n} = \rho,$$

concluding the correctness.

Security. For the security proof we first consider a class of quadratic forms for which the signage is vacuously hard, e.g., for a random target $\mathbf{t} \in \mathbb{R}^n/\mathbb{Z}^n$ there exists no nearby vector.

Lemma 193. *Let $\mathbf{Q} \in \mathcal{S}_n^{>0}(\mathbb{Z})$ be a quadratic form with a dense rank k sublattice $\mathbf{D}\mathbb{Z}^k \subset \mathbb{Z}^n$, in particular such that $\rho/\sqrt{k} \leq 1/(\sqrt{8\pi e} \cdot \det(\mathbf{D}^\top \mathbf{Q} \mathbf{D})^{1/2k})$. Then for the dual form $\mathbf{Q}^{-1}$ we have*

$$\Pr_{\mathbf{t} \sim \mathcal{U}([0,1]^n)}[|(\mathbf{t} + \mathcal{B}^n_{\mathbf{Q}^{-1},\rho}) \cap \mathbb{Z}^n| \geq 1] \leq 2^{-k}.$$

Proof. Let $V := \operatorname{span}(\mathbf{D}) \subset \mathbb{R}^n$ such that the orthogonal projection w.r.t. $\mathbf{Q}^{-1}$ of $\mathbb{Z}^n$ onto V defines a projected lattice $\mathbf{C}\mathbb{Z}^k := \pi_{\mathbf{Q}^{-1},V}(\mathbb{Z}^n)$ of rank k, with $\det(\mathbf{C}^\top \mathbf{Q}^{-1} \mathbf{C}) \geq 1/\det(\mathbf{D}^\top \mathbf{Q} \mathbf{D})$. Because a projection is non-increasing in length we have

$$\begin{aligned}
&\Pr_{\mathbf{t} \sim \mathcal{U}(\mathbb{R}^n/\mathbb{Z}^n)}[|(\mathbf{t} + \mathcal{B}^n_{\mathbf{Q}^{-1},\rho}) \cap \mathbb{Z}^n| \geq 1] \\
&\leq \Pr_{\mathbf{t} \sim \mathcal{U}(\mathbb{R}^k/\mathbb{Z}^k)}[|(\mathbf{t} + \mathcal{B}^k_{\mathbf{C}^\top \mathbf{Q}^{-1} \mathbf{C},\rho}) \cap \mathbb{Z}^n| \geq 1] = (*).
\end{aligned}$$

Then using Markov's inequality we can bound the above by

$$\begin{aligned}
(*) \leq \mathbb{E}_{\mathbf{t} \sim \mathcal{U}(\mathbb{R}^k/\mathbb{Z}^k)}[|(\mathbf{t} + \mathcal{B}^k_{\mathbf{C}^\top \mathbf{Q}^{-1} \mathbf{C},\rho}) \cap \mathbb{Z}^n|] &= \frac{\operatorname{Vol}_{\mathbf{C}^\top \mathbf{Q}^{-1} \mathbf{C}}(\mathcal{B}^k_{\mathbf{C}^\top \mathbf{Q}^{-1} \mathbf{C},\rho})}{\operatorname{Vol}_{\mathbf{C}^\top \mathbf{Q}^{-1} \mathbf{C}}(\mathbb{R}^k/\mathbb{Z}^k)} \\
&\leq \frac{(2\pi e/k)^{k/2} \cdot \rho^k}{\sqrt{\det(\mathbf{C}^\top \mathbf{Q}^{-1} \mathbf{C})}} \leq 2^{-k}.
\end{aligned}$$

□

Theorem 194. *We consider the signature scheme* $\mathcal{S} := (\mathbf{Gen}, \mathbf{Sign}, \mathbf{Verify})$ *instantiated with quadratic form* $\mathbf{S} \in \mathcal{S}_n^{>0}(\mathbb{Z})$*, sampling parameter* ρ*, and public key parameter* $s > 0$*. Let* $\mathbf{Q} \in \mathcal{S}_n^{>0}(\mathbb{Z})$ *be a quadratic form with a dense rank* $k = \Theta(n)$ *sublattice* $\mathbf{D}\mathbb{Z}^k \subset \mathbb{Z}^n$*, in particular such that* $2\rho/\sqrt{k} \leq (\sqrt{8\pi e} \cdot \det(\mathbf{D}^\top \mathbf{Q} \mathbf{D}^\top)^{1/k})^{-1}$*. Then* $\mathcal{S}$ *is EUF-CMA secure if* ac-$\Delta\mathrm{LIP}_s^{\mathbf{S},\mathbf{Q}^{-1}}$ *is hard.*

Proof. Let $\mathcal{A}$ be a probabilistic polynomial-time adversary. We present three games $\mathbf{Game}_1, \mathbf{Game}_2, \mathbf{Game}_3$ where $\mathbf{Game}_1$ is the regular EUF-CMA game with the original scheme, $\mathbf{Game}_2$ reprograms the random oracle to generate valid signatures without knowledge of the secret key, and $\mathbf{Game}_3$ samples the public key from $[\mathbf{Q}^{-1}]$ instead of $[\mathbf{S}]$. By a standard smoothness argument the adversary's view of $\mathbf{Game}_1$ and $\mathbf{Game}_2$ is statistically indistinguishable, and $\mathbf{Game}_2$ and $\mathbf{Game}_3$ are indistinguishable by the hardness of ac-$\Delta\mathrm{LIP}_s^{\mathbf{S},\mathbf{Q}^{-1}}$. Then we conclude by Lemma 193 that the problem of forging a signature in $\mathbf{Game}_3$ is statistically hard.

The original EUF-CMA game $\mathbf{Game}_1$ is as follows [KL20]:

- $\mathbf{Gen}(1^n)$ is run to obtain keys $(pk = \mathbf{P}, sk = \mathbf{U})$.
- Adversary $\mathcal{A}$ is given $pk = \mathbf{P}$ and access to an oracle $\mathbf{Sign}(sk, \cdot)$. The adversary then outputs $(m, \boldsymbol{\sigma})$ where m was not queried before to the oracle.
- $\mathcal{A}$ succeeds if and only if $\mathbf{Verify}(pk, m, \boldsymbol{\sigma}) = 1$.

To show that our signature scheme $\mathcal{S}$ is EUF-CMA secure we have to show that $\mathbf{Game}_1$ succeeds only with negligible probability. We assume that the adversary queries the oracle on $l = \mathrm{poly}(n)$ distinct[2] message $m_1, \ldots, m_l$. In $\mathbf{Game}_1$ the secret key is used to obtain a valid signature $(m_i, \boldsymbol{\sigma}_i)$ where $\boldsymbol{\sigma}_i \leftarrow \mathcal{D}_{\mathbf{P},\rho/\sqrt{n},\mathcal{H}(m_i)}$. In $\mathbf{Game}_2$ instead we first sample a random error $\mathbf{e}_i \leftarrow \frac{1}{q} \cdot \mathcal{D}_{\mathbf{P},q\rho/\sqrt{n}}$. By Lemma 183 we have $q\rho/\sqrt{n} \geq \|\tilde{\mathbf{B}}_{\mathbf{P}}\| \cdot \sqrt{\ln(2n+4)/\pi}$ with overwhelming probability, and thus by Lemma 177 we can do the sampling without using the secret key. Then we reprogram the random oracle such that $\mathcal{H}(m_i) := \mathbf{t}_i = \mathbf{e} \bmod \mathbb{Z}^n \in \mathbb{T}_q$, and return the signature pair $(m_i, \boldsymbol{\sigma}_i := \mathbf{t}_i - \mathbf{e}_i)$. Note that the probability that $\mathbf{t}_i$ equals any target $\mathbf{t} \in \mathbb{T}_q^n$ is proportional

[2] this can be enforced by salting messages or by derandomization.

to $\rho_{\mathbf{P},\rho/\sqrt{n},\mathbf{t}}(\mathbb{Z}^n)$. So $\mathbf{t}_i$ is close to uniform by Lemma 173 because $\rho/\sqrt{n} \geq \eta_{2^{-\Theta(n)}}([\mathbf{S}]) = \eta_{2^{-\Theta(n)}}([\mathbf{P}])$, and thus the random oracle is still simulated correctly. Additionally the conditional probability of $\boldsymbol{\sigma}_i$ conditioned on $\mathbf{t}_i$ is exactly the same as in $\mathbf{Game}_1$, so we can conclude that $\mathbf{Game}_1$ and $\mathbf{Game}_2$ are statistically indistinguishable from the adversary's point of view.

The only difference between $\mathbf{Game}_2$ and $\mathbf{Game}_3$ is that in $\mathbf{Game}_3$ we sample the public key $\mathbf{P}$ from $\mathcal{D}_s([\mathbf{Q}^{-1}])$ instead of $\mathcal{D}_s([\mathbf{S}])$. Note that $\mathbf{Game}_2$ and $\mathbf{Game}_3$ both only use public information and thus by the hardness of ac-$\Delta\mathrm{LIP}_s^{\mathbf{S},\mathbf{Q}^{-1}}$ the two are computationally indistinguishable.

To conclude note that for any message m we obtain a random target $\mathbf{t} := \mathcal{H}(m) \in \mathbb{T}_q^n$. Let $\mathbf{e}'$ be uniform over the Babai nearest plane region defined by $\mathbf{P}$, then $\|\mathbf{e}'\|_{\mathbf{P}} \leq \frac{\sqrt{n}}{2}\|\tilde{\mathbf{B}}_{\mathbf{P}}\|$, and $\mathbf{t}' := \mathbf{t} + \frac{1}{q}\mathbf{e}'$ is uniform over $\mathbb{R}^n/\mathbb{Z}^n$. By Lemma 193 the uniformly random target $\mathbf{t}'$ lies at distance at least 2ρ from $\mathbb{Z}^n$ w.r.t. $\mathbf{P}$ with overwhelming probability. So for $\mathbf{t}$ we have with overwhelming probability that:

$$\begin{aligned}\mathrm{dist}_{\mathbf{P}}(\mathbf{t},\mathbb{Z}^n) &\geq \mathrm{dist}_{\mathbf{P}}(\mathbf{t}',\mathbb{Z}^n) - \left\|\frac{1}{q}\mathbf{e}'\right\|_{\mathbf{P}} \geq 2\rho - \frac{\sqrt{n}\cdot\|\tilde{\mathbf{B}}_{\mathbf{P}}\|}{2q} \\ &\geq 2\rho - \rho/(2\sqrt{\ln(2n+4)/\pi}) > \rho.\end{aligned}$$

Therefore it is statistically impossible for the adversary to return a valid signature for m, and thus to win $\mathbf{Game}_3$. □

10.6 Cryptanalysis

Equivalent quadratic forms $\mathbf{Q}, \mathbf{Q}' := \mathbf{U}^t\mathbf{Q}\mathbf{U}$ (for some $\mathbf{U} \in \mathcal{GL}_n(\mathbb{Z})$) share many common properties, and these invariants can be used to decide that two quadratic forms cannot be equivalent, or can guide the search for an isomorphism.

10.6.1 Invariants

Recall from Section 9.3 that we named the invariants that are easy to compute *arithmetic* invariants, and those that are hard to compute *geometric* invariants. The arithmetic invariants for an integral form

$\mathbf{Q} \in \mathcal{S}_n^{>0}(\mathbb{Z})$, such as the determinant, gcd, parity and the genus, provide an efficiently computable fingerprint $\mathrm{ari}(\mathbf{Q})$. The fingerprint is essentially only useful to answer the ΔLIP problem in the negative. When instantiating ΔLIP for cryptosystems, we should therefore make sure that these fingerprints match.

The geometric invariants, such as $|\operatorname{Min}(\mathbf{Q})|$ or more generally (part of) the Theta-series, appears to involve finding or even enumerating short vectors; in particular they are plausibly hard to compute.

10.6.2 Algorithms for distinguish-LIP and hardness conjecture

In Section 10.7, we will use ΔLIP with quadratic forms that have different minimal distances $\lambda_1(\mathbf{Q}_0) < \lambda_1(\mathbf{Q}_1)$. However we will be careful to ensure that their arithmetic invariant match $\mathrm{ari}(\mathbf{Q}_0) = \mathrm{ari}(\mathbf{Q}_1)$ to not make the problem trivial.

Approximate-SVP oracle. An f-approx-SVP oracle applied to a form $\mathbf{Q}$ finds a short vector of length at most $f \cdot \lambda_1(\mathbf{Q})$. So ΔLIP is no harder than f-approx-SVP for $f = \lambda_1(\mathbf{Q}_1)/\lambda_1(\mathbf{Q}_0) > 1$ in any of those lattices.

Unusual-SVP via lattice reduction. However even when the gap between $\lambda_1(\mathbf{Q}_0)$ and $\lambda_1(\mathbf{Q}_1)$ is small, the minimal vectors may individually still be unusually short, which make them significantly easier to find than in a random lattice. This is usually formalized via the f-unique-SVP problem, but many instances of interest do not have such a gap between λ_1 and λ_2. In fact the lattice $\mathbb{Z}^n$, the the lattices of Barnes-Wall and Barnes-Sloane, all have $\lambda_1 = \lambda_2 = \cdots = \lambda_n$. But practical and heuristic studies have showed that uniqueness is not that relevant to lattice attacks [AD21]. We therefore introduce yet another lattice problem, called *unusual-SVP* to discuss such instances. A formal complexity reduction between unusual-SVP and unique-SVP matching or approaching the heuristic state of the art appears to be a valuable research objective, but is beyond the scope of the present article.

We define f-unusual-SVP: find a minimal vector under the promise that $\lambda_1(\mathbf{Q}) \leq \mathrm{gh}(\mathbf{Q})/f$, where the Gaussian Heuristic $\mathrm{gh}(\mathbf{Q})$ is a heuristic estimate for $\lambda_1(\mathbf{Q})$ given by:

$$\mathrm{gh}(\mathbf{Q}) := \det(\mathbf{Q})^{1/2n} \cdot \frac{1}{\sqrt{\pi}} \cdot \Gamma(1+n/2)^{1/n} \approx \det(\mathbf{Q})^{1/2n} \cdot \sqrt{\frac{n}{2\pi e}}.$$

State of the art lattice reduction techniques find these unusually short vector more easily than longer vectors with length around $\mathrm{gh}(\mathbf{Q})$, where (heuristically) the hardness is directly driven by the ratio $f = \mathrm{gh}(\mathbf{Q})/\lambda_1(\mathbf{Q})$ [AD21]. E.g., for $f = O(n^\alpha)$ with $\alpha \geq 0$, we can heuristically solve f-unusual-SVP by running BKZ-β with blocksize $\beta = \frac{n}{2\alpha+1} + o(n)$. Given a form $\mathbf{Q}' \in [\mathbf{Q}_0] \cup [\mathbf{Q}_1]$, we parametrize the lattice reduction algorithm to find a unusual short vector with length $\min\{\lambda_1(\mathbf{Q}_0), \lambda_1(\mathbf{Q}_1)\}$, then depending on success we learn that either $\mathbf{Q}' \in [\mathbf{Q}_0]$ or $\mathbf{Q}' \in [\mathbf{Q}_1]$.

Conclusion. To conclude, let us also note that any of the above attack can also be run over the dual. To state a hardness conjecture capturing these attacks we define the primal-dual gap to the Gaussian Heuristic as:

$$\mathrm{gap}(\mathbf{Q}) = \max\left\{\frac{\mathrm{gh}(\mathbf{Q})}{\lambda_1(\mathbf{Q})}, \frac{\mathrm{gh}(\mathbf{Q}^{-1})}{\lambda_1(\mathbf{Q}^{-1})}\right\}.$$

Note that this quantity might be slightly lower than 1 (but no lower than 1/2 by Minkowski's bound): there might exist excellent lattice packings beating the Gaussian Heuristic. We will be assuming[3] $\mathrm{gap}(\mathbf{Q}_i) \geq 1$, which implies that $\lambda_1(\mathbf{Q}_i)/\lambda_1(\mathbf{Q}_{1-i}) \leq \mathrm{gap}(\mathbf{Q}_i)$ for $i = 0, 1$, therefore also capturing the approximate-SVP approach.

In all the attacks above, one first searches for vectors no larger than $f \cdot \lambda_1(\mathbf{Q}_i)$ w.r.t. $\mathbf{Q}_i$ for $f = \mathrm{gap}(\mathbf{Q}_i)$, hence the following conjecture.

Conjecture 195 (Hardness of ΔLIP (Strong)). *For any two classes of quadratic forms* $[\mathbf{Q}_0], [\mathbf{Q}_1]$ *of dimension* n*, with* $\mathrm{ari}([\mathbf{Q}_0]) = \mathrm{ari}([\mathbf{Q}_1])$*, and* $1 \leq \mathrm{gap}([\mathbf{Q}_i]) \leq f$*, the best attack against* wc-$\Delta\mathrm{LIP}^{\mathbf{Q}_0,\mathbf{Q}_1}$ *requires solving* f*-approx-SVP in the worst-case from either* $[\mathbf{Q}_0]$ *or* $[\mathbf{Q}_1]$.

[3] That is, we do not make a hardness conjecture for such exceptionally dense lattice packings. Such a regime has never been considered in practical cryptanalysis and would deserve specific attention.

This conjecture is meant to offer a comparison point with existing lattice-based cryptography in terms of the approximating factor. Beyond contradicting this assumption, we also invite cryptanalysis effort toward concrete comparison of f-approx-SVP on those instances to SIS and LWE with the same approximation factor f.

If one only wishes to argue exponential security in n of the schemes proposed in this paper, a sufficient conjecture is the following.

Conjecture 196 (Hardness of ΔLIP (Mild)). *For any two classes of quadratic forms* $[\mathbf{Q}_0], [\mathbf{Q}_1]$ *of dimension* n, *with* $\mathrm{ari}([\mathbf{Q}_0]) = \mathrm{ari}([\mathbf{Q}_1])$, *and* $\mathrm{gap}([\mathbf{Q}_i]) \leq \mathrm{poly}(n)$, $\text{wc-}\Delta\mathrm{LIP}^{\mathbf{Q}_0,\mathbf{Q}_1}$ *is* $2^{\Theta(n)}$*-hard.*

Note that the conjectures above are very strong and 'best-case' over the choice of the isomorphism classes. That is, even though we may only want to use ΔLIP for specific choices of isomorphism classes, we gladly invite cryptanalysis effort on ΔLIP for any choice of isomorphism classes.

We would also like to motivate any reductions from more standard lattice assumptions to LIP, though given current knowledge on LIP, this may be hard to attain. A more reasonable goal might be to generalize the search-to-decision reduction of Szydlo [Szy03], which is currently limited to solving sLIP for the trivial lattice $\mathbb{Z}^n$ given a decisional LIP oracle for a few special lattices.

10.6.3 Algorithms for search-LIP and challenges

While the mentioned arithmetic and geometric invariants allow to semi-decide LIP, the search version requires more effort. Recall from Section 9.5 that it typically proceeds by enumerating sets of short vectors for both quadratic forms, and then to backtrack search for isometries between these sets. The hardest instances appear to be those with many minimal vectors, and in particular those with $\lambda_1(\mathbf{Q}) = \ldots = \lambda_n(\mathbf{Q})$.

Given the current state of the art, it seems difficult to give a precise conjecture for the hardness of search-LIP, as it may depend on the minimal distance, the kissing number, and the size and structure of the automorphism group. Given that all known approaches require finding at least one short vector, we can conjecture exponential hardness for

	Source	Dim.	Kissing Number
Λ_{24}	[NS11; CS13]	24	$196560 \approx 2^{17.6}$
MW_{44}	[NS11]	44	$2708112 \approx 2^{21.4}$
$\mathrm{P}_{48\mathrm{n}}$	[NS11]	48	$52416000 \approx 2^{25.6}$
Ne_{64}	[NS11]	64	$138458880 \approx 2^{27.0}$
Γ_{72}	[NS11]	72	$6218175600 \approx 2^{32.5}$
MW_{128}	[NS11]	128	$218044170240 \approx 2^{37.7}$
BW_n	[CS13, Sec 6.5]	$n = 2^m$	$2^{m^2/2+O(m)}$
V_n	[Vlă19]	n	$\geq 2^{0.0338n-o(n)}$

Table 10.2: Some challenge lattices for search-LIP

lattices with polynomial $\mathrm{gap}([\mathbf{Q}])$, bearing in mind that some instances may be much harder, with a complexity up to $n^{O(n)}$.

Conjecture 197 (Hardness of sLIP). *For any class* $[\mathbf{Q}]$ *of quadratic forms of dimension* n *such that* $\mathrm{gap}([\mathbf{Q}]) \leq \mathrm{poly}(n)$*,* wc-sLIP$^{\mathbf{Q}}$ *is* $2^{\Theta(n)}$*-hard.*

To motivate further study of the hardness of LIP, we provide a list of challenge lattices for LIP in Table 10.2, selected from known lattices with large kissing numbers. More remarkable lattices may be found in the online catalogue of Nebe and Sloane, in particular the section on the kissing number [NS11].

For these large kissing number challenges it might be more enlightening to separate the search for short vectors and the isomorphism reconstruction. We therefore invite the cryptanalist to even assume that sampling uniformly a random shortest vector comes at unit cost. The search for all shortest vectors in these specific lattices may also be harder in practice than for random lattices. Indeed, for these challenges, the kissing number is significantly larger than the $\approx (4/3)^{n/2}$ many short vectors provided by heuristic sieving algorithms [NV08].

10.7 Instantiating from remarkable lattices

To instantiate our KEM and signature scheme, we do not only need a lattice with efficient decoding or sampling; we also need a second lattice with a specific property to instantiate the ΔLIP problem and argue security. This section deals with how the ΔLIP pair is constructed from a single remarkable lattice, while keeping the gaps small. Note that this is just one generic way of doing this; for specific lattices there might exist better instantiations.

10.7.1 Key Encapsulation Mechanism

To instantiate our KEM we need two quadratic forms: a form $\mathbf{S}$ along with an efficient decoder that can decode up to some distance $\rho < \lambda_1(\mathbf{S})/2$, and a form $\mathbf{Q}$ with a dense rank $k = \Theta(n)$ sublattice $\mathbf{D} \cdot \mathbb{Z}^k \subset \mathbb{Z}^n$ such that $\eta_{\frac{1}{2}}(\mathbf{D}^t\mathbf{Q}\mathbf{D}) \leq \rho/(2\sqrt{n})$. For simplicity of notation we move to the lattice point of view.

We assume to have an n-dimensional lattice $\mathcal{L}$ for which $\mathrm{gap}(\mathcal{L}) \leq f = f(n)$ is bounded, and for which we can decode up to $\rho = \Theta(1/f) \cdot \mathrm{gh}(\mathcal{L}) < \lambda_1(\mathcal{L})/2$. I.e., a lattice for which the primal, dual and decoding gap are bounded by f. We consider a general construction leading to a $2n$-dimensional primary lattice $\mathcal{L}_{\mathbf{S}}$ and secondary lattice $\mathcal{L}_{\mathbf{Q}}$ with primal-dual gaps bounded by $O(f^3)$ and such that $\mathcal{L}_{\mathbf{Q}}$ has a dense enough sublattice to instantiate our KEM.

By Lemma 174 and due to the bounded gap of $\mathcal{L}$, we have

$$\eta_{\frac{1}{2}}(\mathcal{L}) \leq \eta_{2^{-n}}(\mathcal{L}) \leq \frac{\sqrt{n}}{\lambda_1(\mathcal{L}^*)} \leq \frac{\sqrt{n} \cdot f}{\mathrm{gh}(\mathcal{L}^*)} = \Theta(f \cdot \det(\mathcal{L})^{1/n}).$$

Now let $g = \Theta(f^2)$ be a positive integer and consider the lattices:

$$\mathcal{L}_{\mathbf{S}} := g \cdot \mathcal{L} \oplus (g+1) \cdot \mathcal{L}, \text{ and } \mathcal{L}_{\mathbf{Q}} := \mathcal{L} \oplus g(g+1)\mathcal{L}.$$

By construction the first lattice $\mathcal{L}_{\mathbf{S}}$ is geometrically similar to (two orthogonal copies of) the original lattice. In particular we can still decode $\mathcal{L}_{\mathbf{S}}$ up to radius $\rho' := g \cdot \rho = \Theta(g/f) \cdot \mathrm{gh}(\mathcal{L})$. Furthermore, the second lattice $\mathcal{L}_{\mathbf{Q}}$ contains by construction a dense sublattice $\mathcal{L} \subset \mathcal{L}_{\mathbf{Q}}$, where the parameter g allows to tune the precise (relative) density.

Invariants match. Both lattices have determinant $g^n(g+1)^n \det(\mathcal{L})^2$. Due to the coprimality of g and $g+1$ we still have $\gcd(\mathcal{L}_{\mathbf{S}}) = \gcd(\mathcal{L}_{\mathbf{Q}}) = \gcd(\mathcal{L})$, and similarly for the parity. It remains to check rational equivalence and p-adic equivalence for all primes p. Let $\mathbf{R}$ denote a quadratic form representing $\mathcal{L}$. Up to integral equivalence, we have:

$$\mathbf{S} := \begin{pmatrix} g^2\mathbf{R} & 0 \\ 0 & (g+1)^2\mathbf{R} \end{pmatrix} \qquad \mathbf{Q} := \begin{pmatrix} \mathbf{R} & 0 \\ 0 & g^2(g+1)^2\mathbf{R} \end{pmatrix}.$$

Let $\mathbf{I}_n$ be the $n \times n$ identity matrix and consider the transformations:

$$\mathbf{U}_1 := \begin{pmatrix} g^{-1}\mathbf{I}_n & 0 \\ 0 & g\mathbf{I}_n \end{pmatrix} \qquad \mathbf{U}_2 := \begin{pmatrix} 0 & (g+1)\mathbf{I}_n \\ (g+1)^{-1}\mathbf{I}_n & 0 \end{pmatrix}.$$

Then $\mathbf{Q} = \mathbf{U}_1^t\mathbf{S}\mathbf{U}_1$ over $\mathbb{Q}$: this implies $[\mathbf{S}]_{\mathbb{Q}} = [\mathbf{Q}]_{\mathbb{Q}}$. For any prime p we have that $\gcd(g,p) = 1$ or $\gcd(g+1,p) = 1$. So g or $(g+1)$ is invertible over the p-adic integers $\mathbb{Z}_p$, and thus $\mathbf{U}_1 \in \mathcal{GL}_d(\mathbb{Z}_p)$ exists and $\mathbf{Q} = \mathbf{U}_1^t\mathbf{S}\mathbf{U}_1$ over $\mathbb{Z}_p$ or $\mathbf{U}_2 \in \mathcal{GL}_d(\mathbb{Z}_p)$ exists and $\mathbf{Q} = \mathbf{U}_2^t\mathbf{S}\mathbf{U}_2$ over $\mathbb{Z}_p$. In any case, we have established $[\mathbf{S}]_{\mathbb{Z}_p} = [\mathbf{Q}]_{\mathbb{Z}_p}$, which concludes the comparison of arithmetic invariants: $\operatorname{ari}(\mathbf{S}) = \operatorname{ari}(\mathbf{Q})$.

Dense sublattice. We now check the requirements for Theorem 192, namely that $\eta_{\frac{1}{2}}(\mathcal{L}) \leq \rho'/(2\sqrt{2n})$, where $\rho' = \Theta(g/f) \cdot \operatorname{gh}(\mathcal{L})$ is the decoding radius of $\mathcal{L}_{\mathbf{S}}$. Given that $\eta_{\frac{1}{2}}(\mathcal{L}) \leq \Theta(f \cdot \operatorname{gh}(\mathcal{L})/\sqrt{n})$, it is sufficient if

$$\Theta(f \cdot \operatorname{gh}(\mathcal{L})/\sqrt{n}) \leq \rho'/(2\sqrt{2n}) = \Theta(g/f) \cdot \operatorname{gh}(\mathcal{L})/\sqrt{n},$$

and thus we can conclude that some $g = \Theta(f^2)$ indeed suffices.

Following the conclusions from the cryptanalysis in Section 10.6.2 and more specifically Conjecture 195, we take a look at the primal-dual gap for $\mathcal{L}_{\mathbf{S}}$ and $\mathcal{L}_{\mathbf{Q}}$. We have that $\operatorname{gap}(\mathcal{L}_{\mathbf{S}}) = \Theta(\operatorname{gap}(\mathcal{L})) \leq O(f)$, and $\operatorname{gap}(\mathcal{L}_{\mathbf{Q}}) = \Theta(g \cdot \operatorname{gap}(\mathcal{L})) \leq O(f^3)$.

More specifically, following the same computation above but for a primal gap of f, dual gap of f^*, and a decoding gap of $f' \geq 2f$ we would have $g = \Theta(f^* \cdot f')$ and obtain a final primal-dual gap of $O(\max(f, f^*) \cdot f^* \cdot f')$.

10.7.2 Signature scheme

Our signature scheme can be instantiated with any lattice for which we can sample efficiently at small Gaussian widths, following a similar ΔLIP pair as above.

Namely, we assume to have a lattice $\mathcal{L}$ with $\mathrm{gap}(\mathcal{L}) \leq f$ and such that we can sample a discrete Gaussian over $\mathcal{L}$ efficiently with parameter $\rho/\sqrt{n} = \Theta(\eta_{2^{-\Theta(n)}}(\mathcal{L}))$ close to the smoothing bound. Similarly to the KEM we set $\mathcal{L}_{\mathbf{S}} := g \cdot \mathcal{L} \oplus (g+1) \cdot \mathcal{L}$, and $\mathcal{L}_{\mathbf{Q}^{-1}} = \mathcal{L} \oplus g(g+1) \cdot \mathcal{L}$ for some integer $g \geq 1$. In particular, as in the KEM, we do have $\mathrm{ari}(\mathbf{S}) = \mathrm{ari}(\mathbf{Q}^{-1})$ with $\mathbf{Q}^{-1}$ instead of $\mathbf{Q}$.

Then for the dual we have $\mathcal{L}_{\mathbf{Q}} = \mathcal{L}^* \oplus \frac{1}{g(g+1)}\mathcal{L}^*$, with $\frac{1}{g(g+1)}\mathcal{L}^*$ as a dense sublattice. The constraint of Theorem 194 boils down to the inequality $\Theta(g \cdot f \cdot \det(\mathcal{L})^{1/n}) \leq \Theta(g^2 \det(\mathcal{L})^{1/n})$, and thus some $g = \Theta(f)$ suffices. The final primal-dual gap of $\mathcal{L}_{\mathbf{S}}$ and $\mathcal{L}_{\mathbf{Q}^{-1}}$ is then bounded by $O(f^2)$.

The simplest lattice for which we have very efficient samplers is of course the integer lattice $\mathbb{Z}^n$, leading to a gap of $O(n)$ via the above construction. Instantiating our scheme with this lattice would lead to an interesting signature scheme where there is no need to compute any Cholesky decomposition, even for signing, and that could be fully implemented with efficient integer arithmetic.

We refer to our last open question (Section 10.1.3) regarding lattices with a tighter Gaussian sampler, in order to obtain a signature scheme with a better underlying approximation factor.

10.7.3 Getting down to $O(f)$

The general constructions presented turn a well-decodable or well-sampleable lattice $\mathcal{L}$ with gaps f into a primary and secondary lattice with gap $O(f^3)$ and $O(f^2)$ to instantiate our KEM and signature scheme respectively. We suggest here that these losses from $O(f)$ to $O(f^3)$ and $O(f^2)$ respectively might be an artifact of the security proof and our generic instantiation construction.

Suppose we can generate a random lattice $\mathcal{L}_{\mathbf{Q}}$ such that $\mathrm{ari}(\mathcal{L}_{\mathbf{Q}}) = \mathrm{ari}(\mathcal{L})$; without the arithmetic constraint we would have with overwhelming probability that $\mathrm{gap}(\mathcal{L}_{\mathbf{Q}}) = O(1)$ (but even $O(f)$ would suffice). Let's assume that the constraint does not affect this gap. Then similar to the scheme of McEliece, by adding the extra security assumption that it is hard to decode in $\mathcal{L}_{\mathbf{Q}}$ (or hard to sample for the signature scheme), we could remove the lossyness argument from the security proof and directly instantiate our schemes with the pair $(\mathcal{L}, \mathcal{L}_{\mathbf{Q}})$, leading to a gap of $O(f)$.

10.7.4 Remarkable lattices

In Table 10.1 we show a list of some remarkable lattices that have efficient decoding algorithms together with their primal, dual and decoding gap. These are interesting candidates for instantiating our KEM, especially those that reach poly-logarithmic decoding gaps. Unfortunately, for all candidates at least one gap exceeds the poly-logarithmic regime, and thus we do not yet obtain a KEM that is secure up to poly-logarithmic approximation factors. Note, however, that the best decoders in the list are very recent, and that there is no (obvious) geometric obstacle for the existence of an efficiently decodable lattice reaching poly-logarithmic primal, dual and decoding gaps.

For the signature scheme we have yet to find a lattice for which one can do discrete Gaussian sampling significantly better than the generic randomized Babai approach. Potentially the list decoding [GP12] algorithm for the Barnes-Wall lattice can be turned into a sampler. From a practical perspective, the lattice $\mathbb{Z}^n$ is extremely interesting, as it allows to sample very efficiently (and in parallel), with similar or even better parameters than those based on LWE and NTRU trapdoor lattices.

10.8 HAWK: fast, compact and simple

The end goal of LIP based cryptography is to build lattice-based schemes that significantly improve upon the state-of-the-art based on LWE, SIS and NTRU assumptions. Until then, we show here that LIP is already competitive with the state-of-the-art using a lattice as simple as $\mathbb{Z}^d$.

We shortly present and summarize the signature scheme Hawk [DPPW22], as a practical instantiation of the signature scheme from Section 10.5. Hawk is a hash-then-sign signature scheme similar to the to be standardized scheme Falcon. Hawk significantly improves over Falcon in its simplicity, and lack of floating point operations

	[Pre+20] Falcon 512	This work Hawk 512	Gain $\left(\frac{\text{Falcon}}{\text{Hawk}}\right)$
AVX2 **KeyGen**	7.95 ms	4.25 ms	×1.87
Reference **KeyGen**	19.32 ms	13.14 ms	×1.47
AVX2 **Sign**	193 µs	50 µs	× 3.9
Reference **Sign**	2449 µs	168 µs	×14.6
AVX2 **Verify**	50 µs	19 µs	×2.63
Reference **Verify**	53 µs	178 µs	×0.30
Secret key (bytes)	1281	1153	×1.11
Public key (bytes)	897	1006 ± 6	×0.89
Signature (bytes)	652 ± 3	542 ± 4	×1.21

Table 10.3: Performance of Falcon and Hawk for $n = 512, 1024$ on an Intel® Core™ i5-4590 @3.30GHz processor with TurboBoost disabled. The **Sign** timings correspond to batch usage.

in signing and verification. This is all due to the simplicity of the underlying lattice $\mathbb{Z}^d$ compared to the more complicated NTRU lattice of Falcon.

Discrete sampling in (cosets of) $\mathbb{Z}^d$ is almost trivial and can be done coordinate wise. The Gaussian parameter s, and thus the signatures, can also be a relative factor 1.17 smaller than that of Falcon due to the orthogonality of $\mathbb{Z}^d$. To make the sampling even more practically efficient and easy to implement in constant time we restrict the target to the cosets $\{0, \frac{1}{2}\}^d + \mathbb{Z}^d$, such that we can use two precomputed tables for the discrete Gaussian: one for $\mathbb{Z}$ and one for $\frac{1}{2} + \mathbb{Z}$. Note that any short coset representative $\mathbf{t}$ also gives a short vector $2\mathbf{t} \in \mathbb{Z}^d$. With the appropriate parameters the concrete cryptanalysis shows that these vectors can be leaked safely.

To limit the size of the keys and signatures, and the arithmetical operations, we instantiate $\mathbb{Z}^d$ as a rank 2 module lattice. The number ring $R = \mathbb{Z}[X]/(X^n + 1)$ for $n = d/2$ is naturally orthogonal under the trace norm and can thus be identified with $\mathbb{Z}^n$. Then we simply take $R^2 = R \oplus R$ as a rank 2 R-module which can be identified with

$\mathbb{Z}^d$. A basis $\mathbf{B} \in R^{2\times 2}$ of R^2 consists of only 4 ring elements. Instead of quadratic forms we have to work with hermitian forms $\mathbf{Q} = \mathbf{B}^*\mathbf{B}$, where $\mathbf{B}^*$ denotes the adjoint transpose of $\mathbf{B}$. The inner product and norm w.r.t. $\mathbf{Q}$ can be defined in an analogues way from the original trace inner product.

What remains is to discuss a sampler for the hermitian forms $\mathbf{Q}$ corresponding to R^2 serving as a public key. Given that $\mathbf{I}_2$ is a basis of R^2 the basis transformations and the bases themself coincide. I.e., we only have to consider how to sample bases $\mathbf{B} \in \mathcal{SL}_2(R)$. Surprisingly the natural way to do this is similar to the key generation of Falcon. First we sample a vector $(f, g)^\top \in R^2$ each element according to a discrete Gaussian, which we then have to complete with another vector $(F, G)^\top \in \mathbb{R}^2$ to a full basis $\mathbf{B}$ with determinant $f \cdot G - g \cdot F = 1$. The Falcon key generation does exactly the same, except that there the final basis must have determinant q, and Falcon's efficient algorithms for this can simply be adapted to suit the needs of Hawk. They key generation is thus as follows: sample $(f, g) \in R^2$, extend to a basis $\mathbf{B}$ of R^2, and return $(\mathbf{sk} = \mathbf{B}, \mathbf{pk} = \mathbf{B}^*\mathbf{B})$. A similar worst-case to average-case reduction as in Section 10.2 can be shown for the resulting distribution.

The key and signature sizes can be further reduced. For example the public key $\mathbf{Q}$ is hermitian: the symmetry, the self-adjointness of the diagonal elements, and the relation $\det(\mathbf{Q}) = 1$, allow to throw away (and later reconstruct) a lot of information. Similarly we can throw away the first half of the signature, and reconstruct the other half simply by size-reduction with the hermitian form. The resulting scheme is very competitive with Falcon, see Table 10.3. The key and signature sizes for NIST security level 1 are about the same, while Hawk is about 1.5× to 4× faster with **KeyGen**, **Sign** and **Verify** than Falcon. But Hawk really shines on low-end hardware without floating-point support. Due to the simplicity and floating-point free sampling over $\mathbb{Z}^n$ the reference **Sign** implementation is more than 14× faster than Falcon. Solving a big disadvantage of the Falcon scheme.