Refined Security Estimation for LWE with Hints via a Geometric Approach

Dana Dachman-Soled[1] *, Huijing Gong[2], Tom Hanson[1], and Hunter Kippen[1] **

[1] University of Maryland
{danadach, thanson, hkippen}@umd.edu
[2] Intel Labs
huijing.gong@intel.com

Abstract. The Distorted Bounded Distance Decoding Problem (DBDD) was introduced by Dachman-Soled et al. [Crypto '20] as an intermediate problem between LWE and unique-SVP (uSVP). They presented an approach that reduces an LWE instance to a DBDD instance, integrates side information (or "hints") into the DBDD instance, and finally reduces it to a uSVP instance, which can be solved via lattice reduction. They showed that this principled approach can lead to algorithms that perform better than ad-hoc algorithms that do not rely on lattice reduction.
The current work focuses on new methods for integrating hints into a DBDD instance. We introduce a variant of DBDD which we coin Ellipsoidal Bounded Distance Decoding (EBDD), and view an EBDD instance as providing the promise that the correct solution is the unique lattice point contained in an ellipsoid. We then view "hints" as geometric operations on the EBDD ellipsoid. Our approach allows us to introduce two new types of hints: (1) Inequality hints, corresponding to the region of intersection of an ellipsoid and a halfspace; (2) Combined hints, corresponding to the region of intersection of two ellipsoids. Since the regions in (1) and (2) are not necessarily ellipsoids, we replace them with approximations. We also consider compatibility of our approach with "perfect," "approximate," "modular," and "short vector" hints from the prior work.
We apply our techniques to the decryption failure and side-channel attack settings. We show that "inequality hints" can be used to model decryption failures, and that our new approach yields a geometric analogue of the "failure boosting" technique of D'anvers et al. [ePrint, '18]. We also show that "combined hints" can be used to fuse information from a decryption failure and a side-channel attack, resulting in reduced hardness of the resulting uSVP instance, compared to a naive combination of the information. We provide experimental data for both applications. The code that we have developed to implement the integration of hints and hardness estimates extends the Toolkit from prior work and has been released publicly.

* This project is supported in part by NSF grant #CNS-1453045 (CAREER), by financial assistance awards 70NANB15H328 and 70NANB19H126 from the U.S. Department of Commerce, National Institute of Standards and Technology, and by Intel through the Intel Labs Crypto Frontiers Research Center.

** Supported in part by the Clark Doctoral Fellowship from the Clark School of Engineering, University of Maryland, College Park

1 Introduction

LWE-based cryptosystems are among the foremost candidates for post-quantum standardization and, as such, are expected to be deployed in the next few years. It is therefore critical to understand the *concrete security* of LWE, i.e., exactly how much computational cost is needed to solve an LWE instance for a particular choice of parameters. Parameters for standardized cryptosystems are typically set so that the *best known (quantum) attack* in a given computational model requires some minimum amount of time (e.g. a common target is "128-bit security"). If the state-of-the-art algorithm for solving LWE is significantly improved, parameter settings of all cryptosystems relying on LWE must be modified in order to retain their security guarantees.

Currently, one of the commonly used algorithms for LWE follow this template: (1) Embed the LWE instance into a uSVP instance, which asks to find the shortest non-zero vector in a *lattice*, and then (2) solve the uSVP instance using a type of algorithm known as *lattice reduction*. In this work, we consider algorithms that follow the above template, and our goal is to develop improved methods for the first step–in the case that additional side information about the LWE secret or error is available. While side-channel information is not considered as part of the standard security model, it remains an important practical consideration, especially for standardized cryptosystems which will be widely deployed in a range of settings. Indeed, Round 3 of the NIST post-quantum cryptography (PQC) standardization effort focused attention on resistance of candidate implementations to side-channel attacks [1].

Dachman-Soled et al. [17] created a toolkit for integrating so-called "hints" into uSVP instances that can then be solved via lattice-reduction algorithms. To achieve this, they introduced an intermediate lattice problem known as DBDD (Distorted Bounded Distance Decoding). A DBDD instance consists of three parts: A lattice Λ, a mean vector $\boldsymbol{\mu}$, and a covariance matrix $\boldsymbol{\Sigma}$. The original lattice Λ represents the lattice obtained through Kannan's embedding–which is a way to construct a lattice in which the LWE secret/error is the shortest non-zero vector. Subsequently, side information can sometimes be used to sparsify or reduce the dimension of the original lattice. The remaining parts of the instance $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, correspond to a mean vector and covariance matrix, and these represent distributional information known about the LWE secret/error. $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ originally represents the fact that the secret/error is drawn from a distribution with known mean/covariance determined by the specifications of the cryptosystem. Subsequently, it captures the conditional distribution on the secret/error, given the side information, in cases where this conditional distribution remains well approximated by a Gaussian. Thus, certain types of information on the structure or on the distribution of the secret can be integrated into a DBDD instance, starting with the original instance, and then modifying Λ, and/or $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ appropriately. A DBDD instance can then be converted into a uSVP instance using *homogenization*—centering the ellipsoid at the origin—and *isotropization*—applying a linear transformation that simultaneously transforms the ellipsoid into a ball and transforms the lattice into a different lattice with higher volume. Finally, the

resulting uSVP instance is fed into the BKZ [44] lattice reduction algorithm to obtain the shortest non-zero vector in the transformed lattice. This short non-zero vector allows direct recovery of the LWE secret/error. [17] demonstrated their methodology with numerous examples, and provided an open-source implementation to predict the security decay (i.e. reduction in the BKZ blocksize, β, required for key recovery) of an LWE instance given a set of hints.

The approach of the current work is to provide an alternate geometric interpretation to the distributional approach considered in [17]. We begin by viewing the solution of an LWE instance (with secret of dimension n and error of dimension m, for a total dimension $d = n + m$) as the (unique) integer point contained in an ellipsoid that is constructed from the given LWE instance. Note that such an instance can again be specified by the lattice $\Lambda = \boldsymbol{I}_d$, as well as by the center $\boldsymbol{\mu}$ and the positive semidefinite "shape" matrix $\boldsymbol{\Sigma}$ defining the ellipsoid. Thus, we obtain an initial embedding with different dimension than a DBDD instance, which we define to be an instance of a slightly modified problem, EBDD (Ellipsoidal Bounded Distance Decoding). As with DBDD, an EBDD instance can be converted into an SVP instance by first converting it to a DBDD instance (by embedding it in a space of one higher dimension), and then using *homogenization* and *isotropization*. Where our approach differs, however, is that we now view "hints" as geometric operations on the EBDD ellipsoid, as opposed to viewing a hint as inducing a conditional probability distribution (represented by a mean and covariance) on the secret/error. Viewing hints as inducing conditional distributions meant that only hints corresponding to information $Z = z$ such that the distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \mid Z = z$ was (well approximated by) a Gaussian (as was the case for perfect and conditional approximate hints, for example), could be integrated into a DBDD instance. For EBDD instances, however, we need only maintain the invariant that the lattice point corresponding to the correct solution is contained inside the ellipsoid that is part of the instance. This gives us additional flexibility to integrate types of hints for which $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \mid Z = z$ is *not* (well approximated by) a Gaussian.

1.1 Our Contributions

We introduce two new types of hints:

Inequality Hints. Here we consider the leakage of the information that $\langle \boldsymbol{v}, \boldsymbol{s} \rangle \geq \gamma$, where $\boldsymbol{v}$ is known and $\boldsymbol{s}$ is the LWE secret/error vector. The approach from prior work to integrate this type of hint would be to consider the distribution on $\boldsymbol{s}$, conditioned on knowledge of $\langle \boldsymbol{v}, \boldsymbol{s} \rangle \geq \gamma$. As this is no longer (well approximated by) a Gaussian distribution, the prior approach could not directly handle these types of hints. Given our geometric perspective, this hint now corresponds to the information that the LWE secret is contained in the intersection of the initial ellipsoid and the halfspace $\{\boldsymbol{x} \in \mathbb{R}^d \mid \langle \boldsymbol{x}, \boldsymbol{v} \rangle \geq \gamma\}$. Unfortunately, the geometric perspective does not seem to help, as this region of intersection is also not an ellipsoid! Instead, we *approximate* the region of intersection with an ellipsoid. We use the fact that one can efficiently compute the minimal volume ellipsoid (called

the Löwner-John ellipsoid) that circumscribes the intersection of an ellipsoid and a halfspace [12]. Using this circumscribed ellipsoid in our EBDD instance maintains our required invariant, yet the new ellipsoid has *smaller volume* (under the constraints given in Section 2.3.1), making the resulting uSVP problem easier. See Section 4.1 for details on integration of inequality hints and Section 5.1 for validation of our β estimates for these hints.

Inequality hints are useful in the decryption failure setting since the information that is learned from a decryption failure is exactly of the form of an inequality hint. We apply our approach to reduce the predicted β value required for key recovery, given a fixed number of decryption failures. We further describe a new, geometric-based failure boosting technique[3] obtained from our approach. See Section 5.2 for details and experimental results.

Combined Hints. Combined hints provide a way to "fuse" information from two DBDD or EBDD instances into a single instance. To motivate this type of hint, consider a situation where we have two sources of side information for a single LWE secret/error, such as data from decryption failures, and data from a side-channel attack. The information from these sources is captured by the two DBDD or EBDD instances $(\Lambda, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $(\Lambda, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ (for purposes of this example we assume the two lattices are equal in the two instances, but our techniques extend to the case in which the lattice differ).

One might consider using the conditional approximate hints of [17] to integrate the information from the second instance $(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ into the first instance $(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$. However, the formulas for conditional approximate hints given by [17] require both distributions to be Gaussian. In cases where the secret/error coordinates are independent, it may be possible to use the *a posteriori* approximate hints of [17]. This approach essentially erases certain information from $(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$, and replaces it with corresponding information from $(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$. However, we would like a way to combine information from the two instances more effectively, rather than replacing one with the other.

Even when no distributional information is available, given the promise of the two DBDD or EBDD instances, we can conclude that the LWE secret/error vector $\boldsymbol{s}$ lies in the intersection of the two ellipsoids corresponding to $(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$. This region is not necessarily an ellipsoid, so we cannot simply obtain a new DBDD or EBDD instance by intersecting the ellipsoids. Instead, we adopt the "fusion" approach [43,48] which is to find the convex combination of the two ellipsoids that optimizes the volume of the resulting ellipsoid. The optimal convex combination has the following properties: (1) It is an ellipsoid, (2) It is guaranteed to contain the intersection of the two ellipsoids, and (3) It does not contain points that are outside both ellipsoids. See Section 4.2 for details on integration of combined hints and discussion of when the resulting ellipsoid achieves smaller volume than both input ellipsoids. Validation of our β estimates for these hints can be found in Section 5.1.

[3] The term "failure boosting" (see [20]) refers to techniques that use information from previous decryption failures to increase the failure rate for subsequent queries.

We illustrate our approach by using it to fuse information from decryption failures and side-channel leakage, reducing the predicted β value required to recover the secret as compared to the naive approach of combining the information. See Section 5.3 for details and experimental results.

Compatibility with perfect hints. Once a DBDD instance or EBDD instance has evolved via the integration of inequality or combined hints, we can no longer make the Gaussian assumption from the prior work. This means that if there are additional perfect hints, they can no longer be integrated using the prior method. We present a new algorithm for integrating "perfect hints" into an LWE instance that does not require any distributional assumptions. A perfect hint is the leakage of the information that $\langle \boldsymbol{v}, \boldsymbol{s} \rangle = \gamma$, where $\boldsymbol{v}$ is known and $\boldsymbol{s}$ is the LWE secret/error vector. We now view this hint as consisting of the information that the LWE secret lies in the the intersection of the current DBDD or EBDD ellipsoid and the hyperplane $H := \{\boldsymbol{s} : \langle \boldsymbol{v}, \boldsymbol{s} \rangle = \gamma\}$. We note that the resulting intersection is itself an ellipsoid, thus maintaining our invariant that the DBDD or EBDD instance consists of a lattice and ellipsoid such that the LWE secret is the (unique) lattice point contained in the ellipsoid. We also propose a different way to deal with non-homogenized perfect hints. All perfect hints from [17] were manipulated so that the incorporated hint was $\langle \boldsymbol{v}', \boldsymbol{s}' \rangle = 0$ with $\gamma = 0$. This was needed in order to maintain the invariant that the lattice part of the DBDD instance remains a lattice, and not a lattice coset. Our alternative technique allows us to deal directly with non-homogenized perfect hints, and may be of independent interest. See Section 4.3 for more details.

Experimental results show that our β estimates improve accuracy when more hints are integrated, compared to the estimates of [17]. The actual β needed to recover the secret are the same across the two techniques, since the generated instances differ only by a scaling factor. See Section 5.1 for validation of our β estimates for these hints and comparison with the β estimates from [17].

Compatibility with approximate hints. Given an evolved DBDD or EBDD instance $(\Lambda, \boldsymbol{\mu}, \boldsymbol{\Sigma})$, and $\ell \leq d$ number of approximate hints[4] each having independent error of standard deviation σ_e, we can write the hints as $\boldsymbol{s}\boldsymbol{V} \approx \boldsymbol{\gamma}$, where $\boldsymbol{V}$ is a $d \times \ell$ matrix with each column corresponding to a hint vector. The hints can be integrated by considering the set $\{\boldsymbol{x} : \|\boldsymbol{s}\boldsymbol{V} - \boldsymbol{\gamma}\|^2 \leq \ell \cdot \sigma_e^2\}$, which defines a (possibly degenerate) ellipsoid with mean and shape matrix $(\boldsymbol{\mu}', \boldsymbol{\Sigma}')$. We then apply a *combined hint* on $(\Lambda, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ and $(\Lambda, \boldsymbol{\mu}', \boldsymbol{\Sigma}')$, to obtain the new instance $(\Lambda, \boldsymbol{\mu}'', \boldsymbol{\Sigma}'')$.

Compatibility with modular hints. We sketch how modular hints can be incorporated into an evolved DBDD or EBDD instance, where the secret distribution is no longer Gaussian. Assume we are given a DBDD or EBDD instance $(\Lambda, \boldsymbol{\mu}, \boldsymbol{\Sigma})$, and a hint $\langle \boldsymbol{v}, \boldsymbol{s} \rangle \equiv \gamma \mod k$, where $\boldsymbol{s}$ denotes the LWE secret/error.

[4] ℓ should be large enough that concentration bounds hold for the noise on the aggregate set of hints.

We add a variable c' such that $\langle \boldsymbol{v}, \boldsymbol{s} \rangle - c' \cdot k = \gamma$, where the equation is over the reals. Since $\boldsymbol{s}$ is contained in the ellipsoid defined by $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, we have that $c' \cdot k + \gamma$ is bounded by $\langle \boldsymbol{v}, \boldsymbol{\mu} \rangle \pm \sqrt{r \boldsymbol{v} \boldsymbol{\Sigma} \boldsymbol{v}^T}$ (where r is the rank of $\boldsymbol{\Sigma}$). Therefore, we can first consider the DBDD or EBDD instance $(\Lambda', \boldsymbol{\mu}', \boldsymbol{\Sigma}')$, where $\Lambda' = \Lambda \times \mathbb{Z}$, $\boldsymbol{\mu}' = (\boldsymbol{\mu} || \frac{1}{k}(\langle \boldsymbol{v}, \boldsymbol{\mu} \rangle - \gamma))$ and $\boldsymbol{\Sigma}'$ has dimension one larger than $\boldsymbol{\Sigma}$, with the final row and column all 0 except the bottom right corner set to $\frac{r \boldsymbol{v} \boldsymbol{\Sigma} \boldsymbol{v}^T}{k^2}$. Note that $(\boldsymbol{s} || c')$ is guaranteed to be contained in the corresponding rank-scaled ellipsoid. Finally, we apply our new perfect hint algorithm for hint $\langle \boldsymbol{v}, \boldsymbol{s} \rangle - c' \cdot k = \gamma$. This results in a new DBDD or EBDD instance $(\Lambda'', \boldsymbol{\mu}'', \boldsymbol{\Sigma}'')$ with dimension equal to the original DBDD or EBDD instance.

If some distributional information is known about the instance $(\Lambda, \boldsymbol{\mu}, \boldsymbol{\Sigma})$, one can potentially find a $(\boldsymbol{\mu}', \boldsymbol{\Sigma}')$ where $\boldsymbol{\Sigma}'$ has a smaller bottom right corner coordinate, and for which $(\boldsymbol{s} || c')$ is still guaranteed to be contained in the corresponding rank-scaled ellipsoid. For example, if $\boldsymbol{\Sigma}$ is the original LWE distribution then $\langle \boldsymbol{v}, \boldsymbol{s} \rangle$ is a Gaussian with variance $\boldsymbol{v} \boldsymbol{\Sigma} \boldsymbol{v}^T$. One can choose a constant $h \ll \sqrt{r}$ such that $\langle \boldsymbol{v}, \boldsymbol{s} \rangle \leq h \sqrt{\boldsymbol{v} \boldsymbol{\Sigma} \boldsymbol{v}^T}$ with probability $1 - \epsilon$. The bottom right corner of Σ' can then be set to $\frac{h^2 \boldsymbol{v} \boldsymbol{\Sigma} \boldsymbol{v}^T}{k^2}$, with the guarantee that the secret is contained in the rank-scaled ellipsoid with probability $1 - \epsilon$. We defer implementation of compatible modular hints to future work.

Compatibility with short vector hints. Our new approach remains compatible with the approach of [17] for short vector hints. See Section 4.4 for more details on the adjustment that must be made.

Toolkit Extension. Alongside this paper, we release an extension to the original python/sage 9.0 toolkit from [17][5] We provide an updated API (which simplifies further extensions to the toolkit), and several new class files. The new EBDD.sage class is *fully-featured* implementation of an EBDD instance. It maintains all information about the instance: the lattice Λ, and the (rank-scaled) ellipsoid $E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ as hints are integrated. We leave the lightweight implementations of this extension to future work, as the full implementation is presently required to perform accurate estimation of the hardness loss resulting from our (more-general) geometry-based hints.

1.2 Related Work

Concrete security of Lattice Based Cryptosystems. Two LWE attack templates considered in the literature are known as the *primal*, and *dual* attacks. Both of these attack templates reduce the task of breaking LWE to solving an SVP instance. The SVP problem is a long-standing problem that has attracted much attention from the cryptographic as well as quantum communities. The current asymptotically best SVP algorithms (for classical and quantum computers) include [3,10,36,28]. In practice, the BKZ algorithm [44] was found to perform well

[5] The updated toolkit can be found at https://github.com/hunterkipt/Geometric-LWE-Estimator.

on parameter regimes of interest, though it is not amenable to provable guarantees on its asymptotic performance. The BKZ algorithm on dimension d includes as a core subroutine an SVP solving step on a smaller block-size $\beta \ll d$. It is compatible with both classical and quantum algorithms for solving the smaller blocksize SVP-β instances. NIST post quantum (PQC) candidates have used the runtime estimates for the BKZ algorithm to inform the setting of their concrete parameters [1]. Several works have sought to create models to accurately predict the behavior of the BKZ algorithm in parameter regimes of interest [5,8,16]. Finally, there has been some work on comparing the lattice reduction-based algorithms described up to now with combinatorial algorithms [4,2].

Sice-Channel Attacks (SCA). There are various ways in which an attacker can obtain "side-channel" information about the secret key of a cryptographic scheme, greatly reducing the security of the scheme, or even allowing for a full key recovery. These methods include timing attacks [34], power analysis attacks [35], cache side-channel attacks [46], and microarchitectural attacks [33,37]. Side-channel attacks on NIST PQC candidates include attacks on (earlier versions of) Dilithium, which was recently announced as a selected digital signature algorithm [41,42], qTESLA [42], NTRUEncrypt [45], as well as Rainbow, NTRU, and McEliece [47]. Template attacks were introduced by [15], who used a device identical to the target to generate a precise "template" of the noise. When noisy side-channel data is obtained, the template can be used to learn information about the secret. Bos et al. [13] applied this approach to FrodoKEM, a Round 3 PQC alternate candidate, simulating a single trace power attack using ELMO [38]. We use their side channel attack as the starting point of our experiments in Section 5.3. Other research has focused on active side-channel attacks, where faults are injected during computation involving the secret key, such as RowHammer. Such attacks were performed on the LUOV signature scheme, a Round 2 PQC candidate [39], as well as Dilithium [29].

Decryption Failures. A decryption failure is when the decryption process returns an incorrect message on a validly encrypted ciphertext. Since most lattice-based cryptographic KEM schemes have a non-zero decryption failure rate, several prior works have investigated the possibility of decryption failure attacks. Specifically, decryption failures leak information about the secret key, and in some cases can be used to fully recover the secret. These attacks were first applied on CPA-secure schemes [32,24,9,40,21,22]. However, CCA-secure schemes use a Fujisaki-Okamoto transform which protects against such attacks and ensures that even a malicious attacker can only cause decryption failures with extremely low probability. Several methods have been suggested to boost the rate at which decryption failures occur, thereby lowering the complexity of the attack [20,18,26,11,19]. Recently, Fahr et al. [23] combined SCA and Decryption Failure attacks by using a Rowhammer attack—which induces bit flips in memory—to artificially boost the failure rate of NIST PQC finalist FrodoKEM. This allowed an end-to-end key recovery attack on Frodo-640.

1.3 Organization

In Section 2, we present notation and provide necessary background in linear algebra (Section 2.2), geometry (Section 2.3), and lattices (Section 2.4). Background on the ellipsoid method can be found in Appendix A.

Section 3 defines the DBDD and EBDD problems and gives the reduction from EBDD to DBDD. The reduction from DBDD to uSVP is given in Section 3.1, and Section 3.2 presents security estimates for the uSVP problem. Section 3.3 presents the reduction from from LWE to EBDD.

Section 4 introduces inequality hints (Section 4.1), combined hints (Section 4.2), and revisits perfect (Section 4.3) and short vector (Section 4.4) hints. Missing proofs can be found in Appendix B.

Section 5 presents experimental validation of our β predictions (Section 5.1), applications of our new types of hints to the decryption failure setting (Section 5.2), and to combining decryption failure and side-channel information (Section 5.3).

2 Preliminaries

2.1 Notation

We use bold lower case letters to denote vectors, and bold upper case letters to denote matrices. We use row notation for vectors, and start indexing from 1. We denote by $\boldsymbol{I}_d$ the d-dimensional identity matrix and denote by $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$ the inner product of vectors $\boldsymbol{x}, \boldsymbol{y}$ of the same dimension. We denote by $(\boldsymbol{x}||\boldsymbol{y})$ the concatenation of two row vectors $\boldsymbol{x}, \boldsymbol{y}$. For $\boldsymbol{v} \in \mathbb{R}^d$, $\|\boldsymbol{v}\|$ denotes the ℓ_2 norm of the vector. For a vector $\boldsymbol{v}$, we use both v_i and $\boldsymbol{v}[i]$ to denote the i-th coordinate of the vector. For a matrix $\boldsymbol{M}$ we use $\boldsymbol{M}[i][j]$ to denote the (i, j)-th position of the matrix. Random variables—i.e. variables whose values depend on outcomes of a random experiment—are denoted with lowercase calligraphic letters e.g. $\mathcal{a}, \mathcal{b}, \mathcal{e}$, while random vectors are denoted with uppercase calligraphic letters e.g. $\mathcal{C}, \mathcal{X}, \mathcal{Z}$.

2.2 Linear Algebra

Definition 2.1 (Positive Semidefinite)**.** *A $n \times n$ symmetric real matrix $\boldsymbol{M}$ is positive semidefinite if the scalar quantity $\boldsymbol{x}\boldsymbol{M}\boldsymbol{x}^T \geq 0 \forall \boldsymbol{x} \in \mathbb{R}^n$; if so, we write $\boldsymbol{M} \geq 0$. Given two $n \times n$ real matrices $\boldsymbol{A}$ and $\boldsymbol{B}$, we note that $\boldsymbol{A} \geq \boldsymbol{B}$ if $\boldsymbol{A} - \boldsymbol{B}$ is positive semidefinite.*

Definition 2.2. *$\boldsymbol{M}$ is a square root of $\boldsymbol{\Sigma}$, denoted $\sqrt{\boldsymbol{\Sigma}}$, if $\boldsymbol{M}^T \cdot \boldsymbol{M} = \boldsymbol{\Sigma}$*

As in the prior work [17], we make use of a generalized notion of the inverse and determinant, where these operations are restricted to operate on the row span of the input matrix. For $\boldsymbol{X} \in \mathbb{R}^{d \times k}$ (with any $d, k \in \mathbb{N}$), we denote by $\boldsymbol{\Pi}_{\boldsymbol{X}}$ the orthogonal projection matrix onto $\mathsf{Span}(\boldsymbol{X})$. More formally, let $\boldsymbol{Y}$ be a maximal set of independent row-vectors of $\boldsymbol{X}$; the orthogonal projection matrix is given by $\boldsymbol{\Pi}_{\boldsymbol{X}} = \boldsymbol{Y}^T \cdot (\boldsymbol{Y} \cdot \boldsymbol{Y}^T)^{-1} \cdot \boldsymbol{Y}$. Its complement (the projection

orthogonally to $\mathsf{Span}(\boldsymbol{X})$) is denoted by $\boldsymbol{\Pi}_{\boldsymbol{X}}^{\perp} := \boldsymbol{I}_d - \boldsymbol{\Pi}_{\boldsymbol{X}}$. We naturally extend the notation $\boldsymbol{\Pi}_F$ and $\boldsymbol{\Pi}_F^{\perp}$ to subspaces $F \subset \mathbb{R}^d$. By definition, the projection matrices satisfy $\boldsymbol{\Pi}_F^2 = \boldsymbol{\Pi}_F$, $\boldsymbol{\Pi}_F^T = \boldsymbol{\Pi}_F$ and $\boldsymbol{\Pi}_F \cdot \boldsymbol{\Pi}_F^{\perp} = \boldsymbol{\Pi}_F^{\perp} \cdot \boldsymbol{\Pi}_F = \mathbf{0}$.

Definition 2.3 (Restricted Inverse and Determinant [17])**.** *Let $\boldsymbol{\Sigma}$ be a symmetric matrix. We denote a restricted inverse denoted $\boldsymbol{\Sigma}^{\sim}$ as*

$$\boldsymbol{\Sigma}^{\sim} := (\boldsymbol{\Sigma} + \boldsymbol{\Pi}_{\boldsymbol{\Sigma}}^{\perp})^{-1} - \boldsymbol{\Pi}_{\boldsymbol{\Sigma}}^{\perp}$$

It satisfies $\mathsf{Span}(\boldsymbol{\Sigma}^{\sim}) = \mathsf{Span}(\boldsymbol{\Sigma})$ and $\boldsymbol{\Sigma} \cdot \boldsymbol{\Sigma}^{\sim} = \boldsymbol{\Pi}_{\boldsymbol{\Sigma}}$.

We denote by $\mathsf{rdet}(\boldsymbol{\Sigma})$ the restricted determinant: $\mathsf{rdet}(\boldsymbol{\Sigma}) := \mathsf{det}(\boldsymbol{\Sigma} + \boldsymbol{\Pi}_{\boldsymbol{\Sigma}}^{\perp})$.

2.3 Geometry

Definition 2.4 (Ellipsoid [25])**.** *A set $E \subseteq \mathbb{R}^d$ is a (possibly degenerate) **ellipsoid** if there exist a vector $\boldsymbol{\mu} \in \mathbb{R}^d$ and a positive (semi-)definite $d \times d$-matrix $\boldsymbol{\Sigma}$ such that*

$$E = E(\boldsymbol{\mu}, \boldsymbol{\Sigma}) := \{\boldsymbol{x} \in \boldsymbol{\mu} + \mathsf{Span}(\boldsymbol{\Sigma}) \,|\, (\boldsymbol{x} - \boldsymbol{\mu})\boldsymbol{\Sigma}^{\sim}(\boldsymbol{x} - \boldsymbol{\mu})^T \leq 1\} \tag{1}$$

Definition 2.4 generalizes the traditional non-degenerate ellipsoid. Note that if $\boldsymbol{\Sigma}$ is full rank, then $\boldsymbol{\mu} + \mathsf{Span}(\boldsymbol{\Sigma}) = \mathbb{R}^d$, and the restricted inverse becomes the regular matrix inverse. Equivalently, a (non-degenerate) ellipsoid can be described by the norm $\|\cdot\|_{\boldsymbol{\Sigma}}$ on $\mathbb{R}^d$

$$E(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \{\boldsymbol{x} \in \mathbb{R}^d \,|\, \|\boldsymbol{x} - \boldsymbol{\mu}\|_{\boldsymbol{\Sigma}} \leq 1\}$$

thus, the ellipsoid $E(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the unit ball around $\boldsymbol{\mu}$ in the vector space $\mathbb{R}^d$ endowed with the norm $\|\cdot\|_{\boldsymbol{\Sigma}}$. In particular, the unit ball around $\mathbf{0}$ in the traditional Euclidean norm is $E(\mathbf{0}, \boldsymbol{I}_d)$. As $\boldsymbol{\Sigma}$ is positive definite, the matrix square root exists. As such, we can express an ellipsoid via the following relation

$$E(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \boldsymbol{\Sigma}^{1/2} E(\mathbf{0}, \boldsymbol{I}_d) + \boldsymbol{\mu}$$

making every ellipsoid the image of the unit ball under a bijective affine transformation. These alternative views of an ellipsoid can also be generalized to work with the degenerate case in a similar fashion to the generalized definition.

Definition 2.5 (Volume of a full-rank ellipsoid)**.** *A full-rank ellipsoid $E(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ of dimension d has volume $\mathsf{Vol}(E(\boldsymbol{\mu}, \boldsymbol{\Sigma})) = \sqrt{\det(\boldsymbol{\Sigma})} \cdot V_d$, where V_d is the volume of the d-dimensional unit ball.*

Definition 2.6 (Ellipsoid norm)**.** *Let $\boldsymbol{x} \in \boldsymbol{\mu} + \mathsf{Span}\boldsymbol{\Sigma}$. We define the* ellipsoid norm *of $\boldsymbol{x}$ with respect to ellipsoid $E(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ to be the quantity $(\boldsymbol{x} - \boldsymbol{\mu})\boldsymbol{\Sigma}^{\sim}(\boldsymbol{x} - \boldsymbol{\mu})^T$. Note that $\boldsymbol{x}$ is contained in $E(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ if and only if its ellipsoid norm with respect to $E(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is at most 1.*

Remark 1 (Ellipsoid Scaling). Throughout the paper, we make use of two different ellipsoid scalings. For ellipsoid operations defined by the ellipsoid method (Section 2.3.1) or ellipsoid fusion (Section 4.2), we make use of the traditional scaling factor of 1 in (1). However, the invariant of the DBDD problem (section 3) requires that the ellipsoid be scaled such that the right hand side of (1) is $\mathsf{Rank}(\boldsymbol{\Sigma})$. To remain consistent with prior work [17], we will treat these ellipsoids as a separate object, the *rank-scaled ellipsoid*.

Definition 2.7 (Rank-scaled Ellipsoid)**.** *A set $E^{(\mathsf{Rank})} \subseteq \mathbb{R}^d$ is a (possibly degenerate) **rank-scaled ellipsoid** if there exist a vector $\boldsymbol{\mu} \in \mathbb{R}^d$ and a positive semidefinite $d \times d$-matrix $\boldsymbol{\Sigma}$ such that*

$$E^{(\mathsf{Rank})} = E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) := \{\boldsymbol{x} \in \boldsymbol{\mu} + \mathsf{Span}(\boldsymbol{\Sigma}) \,|\, (\boldsymbol{x}-\boldsymbol{\mu})\boldsymbol{\Sigma}^{\sim}(\boldsymbol{x}-\boldsymbol{\mu})^T \leq \mathsf{Rank}(\boldsymbol{\Sigma})\} \tag{2}$$

Converting a traditional ellipsoid into a rank-scaled ellipsoid follows from the definition. Given a rank-scaled ellipsoid, $E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, it is equivalent to the traditional ellipsoid $E(\boldsymbol{\mu}, \boldsymbol{\Sigma} \cdot \mathsf{Rank}(\boldsymbol{\Sigma}))$. As the mean of the ellipsoid remains the same, let

$$\mathcal{F} : \mathbb{R}^{d \times d} \mapsto \mathbb{R}^{d \times d},\ \mathcal{F}(\boldsymbol{\Sigma}) = \boldsymbol{\Sigma} \cdot \mathsf{Rank}(\boldsymbol{\Sigma}) \tag{3}$$

denote the transformation between the covariance matrices.

Definition 2.8 (Hyperplane)**.** *A set $H \subseteq \mathbb{R}^d$ is a **hyperplane** if there exist a vector $\boldsymbol{v} \in \mathbb{R}^d$ and a scalar threshold $\gamma \in \mathbb{R}$ such that*

$$H = H(\boldsymbol{v}, \gamma) := \{\boldsymbol{x} \in \mathbb{R}^d \,|\, \langle \boldsymbol{x}, \boldsymbol{v} \rangle = \gamma\}$$

Definition 2.9 (Halfspace)**.** *Without loss of generality, A set $H^{\leq} \subseteq \mathbb{R}^d$ is a **halfspace** if there exist a vector $\boldsymbol{v} \in \mathbb{R}^d$ and a scalar threshold $\gamma \in \mathbb{R}$ such that*

$$H^{\leq} := \{\boldsymbol{x} \in \mathbb{R}^d \,|\, \langle \boldsymbol{x}, \boldsymbol{v} \rangle \leq \gamma\}$$

2.3.1 Ellipsoid Halfspace and Hyperplane Intersection The algorithms in some of our applications are reminiscent of the *ellipsoid method*, the first provably polynomial time algorithm for solving linear programs [31]. While our goal is to solve an *integer program*—a harder problem than linear programming—it is well-known (s.f. [30]) that the ellipsoid method can be combined with lattice reduction to solve integer programs. In practice, however, this method is both inefficient and prone to numerical errors. So we must make crucial changes for the approach to be viable in our setting (see Section 5.2). For an overview of the ellipsoid method, see Appendix A.

The main update procedure of the ellipsoid method calculates the Löwner-John ellipsoid corresponding to the intersection of an ellipsoid and halfspace. Given an ellipsoid $E(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ and a halfspace $\{\boldsymbol{x} \in \mathbb{R}^d \,|\, \langle \boldsymbol{x}, \boldsymbol{v} \rangle \leq \gamma\}$ (where $\boldsymbol{v} \in$

$\mathsf{Span}(\boldsymbol{\Sigma})$), the Löwner-John ellipsoid of the intersection $E(\boldsymbol{\mu}', \boldsymbol{\Sigma}')$ is:

$$\begin{aligned} \boldsymbol{\mu}' &= \boldsymbol{\mu} - \tau \frac{\boldsymbol{v}\boldsymbol{\Sigma}}{\sqrt{\boldsymbol{v}\boldsymbol{\Sigma}\boldsymbol{v}^T}} \\ \boldsymbol{\Sigma}' &= \delta \left(\boldsymbol{\Sigma} - \sigma \frac{\boldsymbol{\Sigma}\boldsymbol{v}^T\boldsymbol{v}\boldsymbol{\Sigma}}{\boldsymbol{v}\boldsymbol{\Sigma}\boldsymbol{v}^T} \right) \end{aligned} \tag{4}$$

This expression generalizes the computation of multiple Löwner-John ellipsoids based on ellipsoid-X intersections. The exact intersection performed depends on the values of the three variables δ, σ, and τ. For a geometric interpretation of the effects of varying these parameters, see the survey on the ellipsoid method by Bland, Goldfarb, and Todd [12].

For an ellipsoid-halfspace intersection,

$$\tau = \frac{1 + r\alpha}{r + 1} \qquad \sigma = \frac{2(1 + r\alpha)}{(r + 1)(1 + \alpha)} \qquad \delta = \frac{r^2}{r^2 - 1}(1 - \alpha^2) \tag{5}$$

where r is the rank of $\boldsymbol{\Sigma}$, and α is a distorted measure of the distance between the center of the ellipsoid and the separating hyperplane.

$$\alpha = \frac{\boldsymbol{v}\boldsymbol{\mu}^T - \gamma}{\sqrt{\boldsymbol{v}\boldsymbol{\Sigma}\boldsymbol{v}^T}} \tag{6}$$

When $-1 < \alpha \leq 1$, the separating hyperplane intersects the ellipsoid. If $\alpha = 0$, the separating hyperplane bisects the ellipsoid through its center. The optimal circumscription when $-1 < \alpha < -1/r$ is simply the starting ellipsoid.

Ellipsoid-Hyperplane Intersection. The intersection between an ellipsoid $E(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ and a hyperplane $H(\boldsymbol{v}, \gamma)$, where $\boldsymbol{v} \in \mathsf{Span}(\boldsymbol{\Sigma})$ can be obtained by plugging appropriate parameters into the formula for parallel cuts given in [12]. Doing so yields τ, δ, and σ:

$$\tau = \alpha \qquad \sigma = 1 \qquad \delta = \frac{r}{r - 1}(1 - \alpha^2) \tag{7}$$

where α remains the same as in (6). An ellipsoid-hyperplane intersection is itself an ellipsoid of one fewer dimension. As $\sigma = 1$, the rank one update of $\boldsymbol{\Sigma}$ in (4) reduces the rank of the intersection $E(\boldsymbol{\mu}', \boldsymbol{\Sigma}')$ by 1, and ensures it is flat in the direction of $\boldsymbol{v}$. Here $-1 < \alpha \leq 1$, with no additional restrictions, as the hyperplane need simply intersect the starting ellipsoid.

It is possible to prove a tighter bound so that $\delta = (1 - \alpha^2)$, which is the setting of δ we will use in our implementation.

2.3.2 Ellipsoid Fusion The intersection of two ellipsoids is not generally an ellipsoid, so as in the ellipsoid method, some optimal approximation must be used to compute a representation of the intersection efficiently. There are multiple measures of an ellipsoid's size that could be optimized to produce a

good approximation. For our framework, we adopt the Ellipsoid Fusion procedure proposed by Ros et al. [43]. Ros et al. propose a measure based on the volume of a *convex combination* of the two input ellipsoids. This is done through the minimization of the determinant of the combined ellipsoid's covariance matrix.

Theorem 2.10 (Theorem 2 in [43]). *Given two (possibly degenerate) ellipsoids, $E(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $E(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$, whose intersection is a nonempty bounded region, the region defined by*

$$\{\boldsymbol{x} \,|\, \lambda(\boldsymbol{x}-\boldsymbol{\mu}_1){\boldsymbol{\Sigma}_1}^{\sim}(\boldsymbol{x}-\boldsymbol{\mu}_1)^T + (1-\lambda)(\boldsymbol{x}-\boldsymbol{\mu}_2){\boldsymbol{\Sigma}_2}^{\sim}(\boldsymbol{x}-\boldsymbol{\mu}_2)^T \leq 1\},$$

is a real ellipsoid, $E^{\lambda}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma})$, which coincides with $E(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ or $E(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ for $\lambda = 1$ or $\lambda = 0$ respectively; and it is given by

$$\left.\begin{array}{l} \boldsymbol{\Sigma} = k\boldsymbol{X}^{\sim} \\ \boldsymbol{X} = \lambda{\boldsymbol{\Sigma}_1}^{\sim} + (1-\lambda){\boldsymbol{\Sigma}_2}^{\sim} \\ \boldsymbol{\mu}_0 \Pi_{\boldsymbol{X}} = (\boldsymbol{\mu}_1\lambda{\boldsymbol{\Sigma}_1}^{\sim} + \boldsymbol{\mu_2}(1-\lambda){\boldsymbol{\Sigma}_2}^{\sim})\,\boldsymbol{X}^{\sim} \\ k = 1 - \lambda(1-\lambda)(\boldsymbol{\mu}_2-\boldsymbol{\mu}_1){\boldsymbol{\Sigma}_2}^{\sim}\boldsymbol{X}^{\sim}{\boldsymbol{\Sigma}_1}^{\sim}(\boldsymbol{\mu}_2-\boldsymbol{\mu}_1)^T \end{array}\right\}$$

for $\lambda \in [0,1]$.

Note that $\boldsymbol{\mu}_0$ can be set arbitrarily so long as $\boldsymbol{\mu}_0\Pi_{\boldsymbol{X}}$ satisfies the above. While this combination is not necessarily the optimal circumscription, it does not contain points that are in neither $E(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ nor $E(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$.

Definition 2.11 (Ellipsoid Fusion (Def. 5 in [43]))**.** *The fusion of $E(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $E(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$, whose intersection is a nonempty bounded region is $E^{\tilde{\lambda}}(\boldsymbol{\Sigma}, \boldsymbol{\mu}_0)$ for the value of $\tilde{\lambda} \in [0,1]$ that minimizes its volume.*

Theorem 2.12 (Fusion (Theorem 3 in [43])). *The fusion of $E(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $E(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ is: $E(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$; or $E(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$; or it is $E^{\tilde{\lambda}}(\boldsymbol{\Sigma}, \boldsymbol{\mu}_0)$ where $\tilde{\lambda}$ is the only root in $[0,1]$ of the following polynomial of degree $2n-1$:*

$$\begin{aligned} k(\mathsf{rdet}(\boldsymbol{X}))\mathsf{Trace}(\boldsymbol{X}({\boldsymbol{\Sigma}_1}^{\sim} - {\boldsymbol{\Sigma}_2}^{\sim})) - n(\mathsf{rdet}(\boldsymbol{X}))^2(2\boldsymbol{\mu}_0{\boldsymbol{\Sigma}_1}^{\sim}\boldsymbol{\mu}_1^T - \\ 2\boldsymbol{\mu}_0{\boldsymbol{\Sigma}_2}^{\sim}\boldsymbol{\mu}_2 + \boldsymbol{\mu}_0({\boldsymbol{\Sigma}_2}^{\sim} - {\boldsymbol{\Sigma}_1}^{\sim})\boldsymbol{\mu}_0^T - \boldsymbol{\mu}_1{\boldsymbol{\Sigma}_1}^{\sim}\boldsymbol{\mu}_1^T + \boldsymbol{\mu}_2{\boldsymbol{\Sigma}_2}^{\sim}\boldsymbol{\mu}_2^T) \end{aligned} \quad (8)$$

2.4 Lattice Preliminaries

A *lattice*, denoted by Λ, is a discrete additive subgroup of $\mathbb{R}^d$. It is generated by taking the set of all integer linear combinations of r (where $r \leq d$) linearly independent basis vectors $\{\boldsymbol{b}_j\} \subset \mathbb{R}^d$. Namely,

$$\Lambda := \left\{\sum\nolimits_j z_j\boldsymbol{b}_j : z_j \in \mathbb{Z}\right\}.$$

We say that d is the *dimension* of Λ and r is its rank. A lattice is *full rank* if $r = d$. A matrix $\boldsymbol{B}$ whose rows are the basis vectors $\{\boldsymbol{b}_j\}$ is called a *basis* of the lattice. The *determinant* or *volume* of a lattice Λ is defined as $\mathsf{Vol}(\Lambda) := \sqrt{\mathsf{det}(\boldsymbol{B}\boldsymbol{B}^T)}$.

Definition 2.13 (Unique Shortest Vector Problem)**.** *For a lattice Λ and for $i \in [\mathsf{Rank}(\Lambda)]$, let $\lambda_i(\Lambda)$ denote the i-th successive minimum (the smallest radius r such that the ball $B(\mathbf{0}, r)$ contains i independent points in the lattice). The unique shortest vector problem (*uSVP*) is the following:*

Given a lattice Λ in which $\lambda_1(\Lambda)$ is significantly shorter than $\lambda_2(\Lambda)$, find a nonzero vector $\boldsymbol{s} \in \Lambda$ where $\|\boldsymbol{s}\| = \lambda_1(\Lambda)$.

Definition 2.14 (Search LWE Problem with short secrets)**.** *Let n, m, and q be positive integers, let $\mathcal{X}$ be a distribution over $\mathbb{Z}$. The search* LWE *problem is:*

Given $(\boldsymbol{A} \in \mathbb{Z}_q^{m \times n}, \boldsymbol{b} = \boldsymbol{z}\boldsymbol{A}^T + \boldsymbol{e})$*, where:*
- $\boldsymbol{A} \in \mathbb{Z}_q^{m \times n}$ *is sampled uniformly at random*
- $\boldsymbol{z} \leftarrow \mathcal{X}$*, and* $\boldsymbol{e} \leftarrow \mathcal{X}$ *are sampled with independent and identically distributed coefficients from the distribution* $\mathcal{X}$

Find $\boldsymbol{z}$

3 The DBDD and EBDD Problems

Definition 3.1 (Distorted Bounded Distance Decoding problem)**.** *Let $\Lambda \subset \mathbb{R}^{d+1}$ be a lattice, $\boldsymbol{\Sigma} \in \mathbb{R}^{(d+1)\times(d+1)}$ be a symmetric matrix and $\boldsymbol{\mu} \in \mathsf{Span}(\Lambda) \subset \mathbb{R}^{(d+1)}$ such that*

$$\mathsf{Span}(\boldsymbol{\Sigma}) \subsetneq \mathsf{Span}(\boldsymbol{\Sigma} + \boldsymbol{\mu}^T \cdot \boldsymbol{\mu}) = \mathsf{Span}(\Lambda). \tag{9}$$

The Distorted Bounded Distance Decoding problem $\mathsf{DBDD}_{\Lambda,\boldsymbol{\mu},\boldsymbol{\Sigma}}$ *is:*

Given $\boldsymbol{\mu}, \boldsymbol{\Sigma}$ *and a basis of* Λ*.*
Find *the unique vector* $\boldsymbol{x} \in \Lambda \cap E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

where $E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the (possibly degenerate) rank-scaled ellipsoid

$$E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) := \{\boldsymbol{x} \in \boldsymbol{\mu} + \mathsf{Span}(\boldsymbol{\Sigma}) \mid (\boldsymbol{x} - \boldsymbol{\mu})\boldsymbol{\Sigma}^{\sim}(\boldsymbol{x} - \boldsymbol{\mu})^T \leq \mathsf{Rank}(\boldsymbol{\Sigma})\}.$$

In [17], $E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ corresponds to knowing that the secret vector $\boldsymbol{x}$ to be recovered follows a Gaussian distribution of variance $\boldsymbol{\Sigma}$ and mean $\boldsymbol{\mu}$, and the expected value of $(\boldsymbol{x} - \boldsymbol{\mu})\boldsymbol{\Sigma}^{\sim}(\boldsymbol{x} - \boldsymbol{\mu})^T$ for a Gaussian $\boldsymbol{x}$ of variance $\boldsymbol{\Sigma}$ and mean $\boldsymbol{\mu}$ is $\mathsf{Rank}(\boldsymbol{\Sigma})$. In the current work, we do not view the ellipsoid in the DBDD instance as stemming from the covariance matrix of a multivariate Gaussian distribution. Rather, we view the ellipsoid as defining a region containing a feasible solution to a certain constraint satisfaction problem over the reals. Then we restrict the solutions to those that are also contained in some lattice.

Beyond the different perspective on the DBDD problem described above, it will also be useful for us to consider instances of dimension one lower than DBDD instances. We therefore define a new problem, EBDD.

Definition 3.2 (Ellipsoidal Bounded Distance Decoding problem)**.** *Let $\Lambda \subset \mathbb{R}^d$ be a lattice, $\boldsymbol{\Sigma} \in \mathbb{R}^{d\times d}$ be a symmetric matrix and $\boldsymbol{\mu} \in \mathsf{Span}(\Lambda) \subset \mathbb{R}^d$. such that*

$$\mathsf{Span}(\boldsymbol{\Sigma}) = \mathsf{Span}(\boldsymbol{\Sigma} + \boldsymbol{\mu}^T \cdot \boldsymbol{\mu}) = \mathsf{Span}(\Lambda). \tag{10}$$

The Ellipsoidal Bounded Distance Decoding problem $\mathsf{EBDD}_{\Lambda,\boldsymbol{\mu},\boldsymbol{\Sigma}}$ *is:*

Given $\boldsymbol{\mu}, \boldsymbol{\Sigma}$ *and a basis of* Λ.
Find *the unique vector* $\boldsymbol{x} \in \Lambda \cap E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

where $E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ *denotes the (possibly degenerate) rank-scaled ellipsoid*

$$E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) := \{\boldsymbol{x} \in \mathsf{Span}(\boldsymbol{\Sigma}) \mid (\boldsymbol{x} - \boldsymbol{\mu})\boldsymbol{\Sigma}^{\sim}(\boldsymbol{x} - \boldsymbol{\mu})^T \leq \mathsf{Rank}(\boldsymbol{\Sigma})\}.$$

Converting EBDD to DBDD. We can convert an EBDD instance $(\Lambda, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ into a DBDD instance $(\Lambda', \boldsymbol{\mu}', \boldsymbol{\Sigma}')$ where

$$\begin{aligned}
\Lambda' &= \{(\boldsymbol{x}||z) \in \mathbb{R}^{d+1} : \boldsymbol{x} \in \Lambda, z \in \mathbb{Z}\} \\
\boldsymbol{\mu}' &= (\boldsymbol{\mu}||1) \in \mathbb{R}^{d+1} \\
\boldsymbol{\Sigma}'[i][j] &:= \begin{cases} \boldsymbol{\Sigma}[i][j] & \text{if } i, j \leq d \\ 0 & \text{if } i = d+1 \text{ or } j = d+1. \end{cases}
\end{aligned}$$

3.1 Reduction from DBDD to uSVP

Following [17], the conversion of a DBDD instance $(\Lambda, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ into a uSVP instance proceeds in two steps known as *homogenization* and *isotropization*.

Homogenization: The homogenization procedure takes an ellipsoid that is centered at $\boldsymbol{\mu}$ and converts it into an ellipsoid centered at $\mathbf{0}$. The zero-centered ellipsoid *contains* the ellipsoid centered at $\boldsymbol{\mu}$ (see [17] for the proof of this claim). The volume of the ellipsoid remains the same[6], and the rank of its covariance matrix goes up by 1. Specifically, the conversion is as follows:

$$(\Lambda, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \mapsto (\Lambda, \mathbf{0}, \boldsymbol{\Sigma}' := \boldsymbol{\Sigma} + \boldsymbol{\mu}^T \cdot \boldsymbol{\mu}).$$

Isotropization: The isotropization procedure converts the covariance matrix $\boldsymbol{\Sigma}'$ into an isotropic matrix (i.e. with all its eigenvalues equal to 1), by applying an appropriate linear transformation to the input space. We then perform the same linear transformation on the lattice. Specifically, the conversion is as follows:

$$(\Lambda, \mathbf{0}, \boldsymbol{\Sigma}') \mapsto (\Lambda \cdot M, \mathbf{0}, M \cdot \boldsymbol{\Sigma}' \cdot M^T),$$

where $M = \sqrt{\boldsymbol{\Sigma}'}^{\sim}$. The above can be simplified to

$$(\Lambda \cdot M, \mathbf{0}, M \cdot \boldsymbol{\Sigma}' \cdot M^T) = (\Lambda \cdot M, \mathbf{0}, \boldsymbol{\Pi}_{\boldsymbol{\Sigma}'}) = (\Lambda \cdot M, \mathbf{0}, \boldsymbol{\Pi}_{\Lambda}),$$

see [17] for details on the above simplification. After homogenization and isotropization, we obtain the uSVP instance $\Lambda \cdot \boldsymbol{M}$ (consisting of a lattice only). To complete the reduction, note that from a given solution, $\boldsymbol{x}$, to the $\mathsf{uSVP}_{\Lambda \cdot \boldsymbol{M}}$ problem, one can derive the solution, $\boldsymbol{x}' = \boldsymbol{x} \cdot \boldsymbol{M}^{\sim}$, to the $\mathsf{DBDD}_{\Lambda, \boldsymbol{\mu}, \boldsymbol{\Sigma}}$ problem.

[6] This assumes that $\boldsymbol{\mu}'$–corresponding to the first d coorindates of $\boldsymbol{\mu} \in \mathsf{Span}(\boldsymbol{\Sigma})$ and the final coordinate of $\boldsymbol{\mu}$ is equal to 1, which is the case for DBDD instances obtained from EBDD instances.

3.2 Security estimates of uSVP

We briefly recap the way concrete hardness estimates are computed for a given uSVP instance. Specifically, we consider an attack that consists of applying BKZ-β to the uSVP lattice Λ for an appropriate block size parameter β. The cost of the attack grows with β, and, as in [17], we will treat β itself as a measurement of the security level in a unit called the *bikz*. Bikz-to-bit conversion can be performed using a conversion factor based on the current best algorithms for SVP in lattices of rank β. Typically, it is assumed that 1 bikz ≈ 0.265 bits. As in [17], the concrete security estimates given in this paper only concern the pure lattice attacks via the uSVP embedding discussed above.

Predicting β for a uSVP instance The state-of-the-art predictions for solving uSVP using BKZ were given in [7,5]: For a lattice Λ of dimension $\dim(\Lambda)$, it is predicted that BKZ-β can solve a uSVP_Λ instance with secret $(e||s)$ when

$$\sqrt{\beta/\dim(\Lambda)} \cdot \|(e||s)\| \leq \delta_\beta^{2\beta-\dim(\Lambda)-1} \cdot \mathsf{Vol}(\Lambda)^{1/\dim(\Lambda)} \tag{11}$$

where δ_β is the so called root-Hermite-Factor of BKZ-β. For $\beta \geq 50$, the Root-Hermite-Factor is predictable using the Gaussian Heuristic [16]:

$$\delta_\beta = \left((\pi\beta)^{\frac{1}{\beta}} \cdot \frac{\beta}{2\pi e}\right)^{1/(2\beta-2)}. \tag{12}$$

In [17], the uSVP instances obtained were always isotropic and centered so that the secret has covariance $\boldsymbol{\Sigma} = \boldsymbol{I}$ (or $\boldsymbol{\Sigma} = \boldsymbol{\Pi}_\Lambda$ if Λ is not of full rank) and $\boldsymbol{\mu} = \boldsymbol{0}$. In this case, $\|(\boldsymbol{e}||\boldsymbol{s})\|^2 = \mathsf{Rank}(\boldsymbol{\Sigma}) = \dim(\Lambda)$, in expectation, and β can be estimated as the minimum integer that satisfies

$$\sqrt{\beta} \leq \delta_\beta^{2\beta-\dim(\Lambda)-1} \cdot \mathsf{Vol}(\Lambda)^{1/\dim(\Lambda)}. \tag{13}$$

Importantly, in our case where we do not enforce distributional assumptions, we can no longer assume that after isotropization the secret has covariance $\boldsymbol{\Sigma} = \boldsymbol{I}$ and $\boldsymbol{\mu} = \boldsymbol{0}$, rather, we just know that the secret is contained in the ellipsoid $E^{(\mathsf{Rank})}(\boldsymbol{0}, \boldsymbol{I})$, but its norm could be far smaller. Therefore, when performing our final hardness estimates, we sometimes need to take the length of the shortest vector into account (i.e. we will use equation (11)) in order to accurately predict β. Throughout the paper, whenever this is the case, we will make note of it. The default is to use the prediction from equation (13), which returns β that is at least as large as β from (11). As in [17], while β must be an integer as a BKZ parameter, we provide a continuous value.

Remark 2. To predict security, one does not need the basis of Λ, but only its dimension and its volume. Similarly, it is not necessary to explicitly compute the isotropization matrix $\boldsymbol{M}$ of Section 3.1: $\mathsf{Vol}(\Lambda \cdot \boldsymbol{M}) = \det(\boldsymbol{M})\mathsf{Vol}(\Lambda) = \det(\boldsymbol{\Sigma}')^{-1/2}\mathsf{Vol}(\Lambda)$.

Remark 3. Given a DBDD instance $(\Lambda, \boldsymbol{\mu}, \boldsymbol{\Sigma})$, it is important to note that as the volume of the rank-scaled ellipsoid $E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ *decreases*, the volume of the lattice $\Lambda \cdot \boldsymbol{M}$ after homogenization and isotropization *increases*. Applying the hardness estimate from (13), this makes the resulting uSVP instance *easier* to solve. Our goal, therefore, when integrating "hints" is to ensure that the volume of the rank-scaled ellipsoid $E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ *decreases* as much as possible.

3.3 Obtaining an initial EBDD embedding

Recall that, in the prior work, Kannan's embedding was used to reduce LWE to DBDD. We next present a somewhat different embedding of LWE in EBDD.

The basic EBDD embedding. Consider an LWE instance $\boldsymbol{s}\boldsymbol{A}^T + \boldsymbol{e} = \boldsymbol{b} \mod q$. We can remove the mod q and transform the above to a system of equations over the integers by adding the vector of variables $\boldsymbol{c}$:

$$\boldsymbol{s}\boldsymbol{A}^T + \boldsymbol{e} - q\boldsymbol{c} = \boldsymbol{b}.$$

Note that given an LWE instance $\boldsymbol{A}, \boldsymbol{b}$ and a solution $(\boldsymbol{c}||\boldsymbol{s})$, there is an *affine* transformation to obtain a solution modulo q of the form $(\boldsymbol{e}||\boldsymbol{s})$. Specifically, $\boldsymbol{e} = q\boldsymbol{c} - \boldsymbol{s}\boldsymbol{A}^T + \boldsymbol{b}$. Further, we assume that we can (w.h.p.) upper bound the squared norm of $(\boldsymbol{e}||\boldsymbol{s})$ by $\sigma^2(n+m) = \sigma^2 \cdot d$ (e.g. in standard LWE σ^2 is the variance of $\boldsymbol{s}, \boldsymbol{e}$). In matrix notation, we define $\boldsymbol{B}$ as:

$$\boldsymbol{B} := \begin{bmatrix} q\boldsymbol{I}_m & 0 \\ -\boldsymbol{A}^T & \boldsymbol{I}_n \end{bmatrix} \tag{14}$$

We obtain the following constraint on the solution $(\boldsymbol{c}||\boldsymbol{s})$ of the transformed system: $\left\| \left((\boldsymbol{c}||\boldsymbol{s})\,\boldsymbol{B} + (\boldsymbol{b}||\boldsymbol{0}) \right) \right\|^2 \leq \sigma^2 \cdot d$. The above defines a rank-scaled ellipsoid $\boldsymbol{E}$ with center $(-\boldsymbol{b}||\boldsymbol{0})\,\boldsymbol{B}^{-1}$:

$$E^{(\mathsf{Rank})}((-\boldsymbol{b}||\boldsymbol{0}, \sigma^2(\boldsymbol{B}\boldsymbol{B}^T)^{-1}) :=$$
$$\left\{ (\boldsymbol{c}||\boldsymbol{s}) \in \mathbb{R}^{n+m} : \left((\boldsymbol{c}||\boldsymbol{s}) - (-\boldsymbol{b})||\boldsymbol{0}) \right) \boldsymbol{B}^{-1} \right) \frac{1}{\sigma^2} \boldsymbol{B}\boldsymbol{B}^T \left((\boldsymbol{c}||\boldsymbol{s}) - (\boldsymbol{b}||\boldsymbol{0})\,\boldsymbol{B}^{-1} \right)^T \leq d \right\}.$$

Our EBDD instance is therefore: $\left(\mathbb{Z}^d, (-\boldsymbol{b}||\boldsymbol{0})\,\boldsymbol{B}^{-1}, \sigma^2(\boldsymbol{B}\boldsymbol{B}^T)^{-1} \right)$.

Incorporating a center and shape matrix for $(\boldsymbol{e}||\boldsymbol{s})$. We consider here the case that we are given a center vector $(\boldsymbol{\mu}_e||\boldsymbol{\mu}_s) \in \mathsf{Span}(\boldsymbol{\Sigma})$, and a shape matrix $\boldsymbol{\Sigma}$, along with the guarantee that w.h.p. $((\boldsymbol{e}||\boldsymbol{s}) - (\boldsymbol{\mu}_e||\boldsymbol{\mu}_s))\,\boldsymbol{\Sigma}^{\sim}\,((\boldsymbol{e}||\boldsymbol{s}) - (\boldsymbol{\mu}_e||\boldsymbol{\mu}_s))^T \leq \mathsf{Rank}(\boldsymbol{\Sigma})$. As a special case, the above guarantee holds when $(\boldsymbol{e}||\boldsymbol{s}) \sim \mathcal{N}((\boldsymbol{\mu}_e||\boldsymbol{\mu}_s), \boldsymbol{\Sigma})$ follow a multivariate Gaussian distribution. Using the same $\boldsymbol{B}$ as in (14), we obtain the constraint:

$$\left\| \left(\left((\boldsymbol{c}||\boldsymbol{s})\,\boldsymbol{B} + (\boldsymbol{b}||\boldsymbol{0}) \right) - (\boldsymbol{\mu}_e||\boldsymbol{\mu}_s) \right) \sqrt{\boldsymbol{\Sigma}^{\sim}} \right\|^2 \leq \mathsf{Rank}(\boldsymbol{\Sigma}).$$

This gives the rank-scaled ellipsoid:

$$E^{(\mathsf{Rank})}(((\boldsymbol{\mu}_e - \boldsymbol{b})||\boldsymbol{\mu}_s)\boldsymbol{B}^{-1}, (\boldsymbol{B}^T)^{-1}\boldsymbol{\Sigma}(\boldsymbol{B})^{-1}) := \Big\{(\boldsymbol{c}||\boldsymbol{s}) \in \mathsf{Span}((\boldsymbol{B}^T)^{-1}\boldsymbol{\Sigma}(\boldsymbol{B})^{-1}) :$$

$$\big((\boldsymbol{c}||\boldsymbol{s}) - ((\boldsymbol{\mu}_e - \boldsymbol{b})||\boldsymbol{\mu}_s)\,\boldsymbol{B}^{-1}\big)\boldsymbol{B}\boldsymbol{\Sigma}^{\sim}\boldsymbol{B}^T\big((\boldsymbol{c}||\boldsymbol{s}) - ((\boldsymbol{\mu}_e - \boldsymbol{b})||\boldsymbol{\mu}_s)\,\boldsymbol{B}^{-1}\big)^T \leq \mathsf{Rank}(\boldsymbol{\Sigma})\Big\}.$$

Our EBDD instance is now: $\left(\mathbb{Z}^d, ((\boldsymbol{\mu}_e - \boldsymbol{b})||\boldsymbol{\mu}_s)\,\boldsymbol{B}^{-1}, (\boldsymbol{B}^T)^{-1}\boldsymbol{\Sigma}(\boldsymbol{B})^{-1}\right)$. We can now apply hints to our initial EBDD instance.

Remark 4. Our EBDD embedding extends to $\boldsymbol{s}$ sampled from any distribution $\mathcal{S}$ whose support is contained in a lattice, and to $\boldsymbol{A}^T \in \mathbb{R}^{n\times m}, \boldsymbol{e} \in \mathbb{R}^m$ which are real-valued. Thus, our embedding captures the *Continuous LWE Problem* for secret distributions $\mathcal{S}$ as above [14,27].

4 Hints

4.1 Inequality Hints

An inequality hint on the secret $(\boldsymbol{c}||\boldsymbol{s})$ is the knowledge of $\boldsymbol{v} \in \mathbb{R}^d$ and $l \in \mathbb{R}$, such that $\langle(\boldsymbol{c}||\boldsymbol{s}), \boldsymbol{v}\rangle \leq \gamma$. In other words, inequality hints correspond to the knowledge that the secret lies on one side of a halfspace.

The process for integrating inequality hints relies on the ellipsoid-halfspace intersection procedure of the ellipsoid method (4,5). Given an EBDD (resp. DBDD) instance $\mathsf{EBDD}_{\Lambda,\boldsymbol{\mu},\boldsymbol{\Sigma}}$ (resp. $\mathsf{DBDD}_{\Lambda,\boldsymbol{\mu},\boldsymbol{\Sigma}}$), an inequality hint with $\boldsymbol{v} \in \mathsf{Span}(\boldsymbol{\Sigma})$ produces a new instance $\mathsf{EBDD}_{\Lambda',\boldsymbol{\mu}',\boldsymbol{\Sigma}'}$ (resp. $\mathsf{DBDD}_{\Lambda',\boldsymbol{\mu}',\boldsymbol{\Sigma}'}$),

$$\Lambda' = \Lambda \tag{15}$$

$$\boldsymbol{\mu}' = \boldsymbol{\mu} - \left(\frac{1+r\alpha}{r+1}\right)\frac{\boldsymbol{v}\mathcal{F}(\boldsymbol{\Sigma})}{\sqrt{\boldsymbol{v}\mathcal{F}(\boldsymbol{\Sigma})\boldsymbol{v}^T}} \tag{16}$$

$$\boldsymbol{\Sigma}' = \mathcal{F}^{-1}\left(\left(\frac{r^2}{r^2-1}(1-\alpha^2)\right)\left(\mathcal{F}(\boldsymbol{\Sigma}) - \left(\frac{2(1+r\alpha)}{(r+1)(1+\alpha)}\right)\frac{\mathcal{F}(\boldsymbol{\Sigma})\boldsymbol{v}^T\boldsymbol{v}\mathcal{F}(\boldsymbol{\Sigma})}{\boldsymbol{v}\mathcal{F}(\boldsymbol{\Sigma})\boldsymbol{v}^T}\right)\right) \tag{17}$$

for $-1/r < \alpha \leq 1$, where α is defined as in (6). and r is the rank of $\boldsymbol{\Sigma}$. If $-1 < \alpha \leq -1/r$, then $\Lambda' = \Lambda$, $\boldsymbol{\mu}' = \boldsymbol{\mu}$, and $\boldsymbol{\Sigma}' = \boldsymbol{\Sigma}$, meaning that for inequality hints with α in this range, we do not make progress under the approximation stemming from the ellipsoid method.

Quantitative volume reduction. Using the matrix determinant lemma and properties of rdet and $\boldsymbol{\Sigma}^{\sim}$, we have that

$$\mathsf{rdet}(\boldsymbol{\Sigma}') = \left(\frac{r^2}{r^2-1}(1-\alpha^2)\right)^r \cdot \left(1 - \left(\frac{2(1+r\alpha)}{(r+1)(1+\alpha)}\right)\right) \cdot \mathsf{rdet}(\boldsymbol{\Sigma}).$$

4.2 Combined Hints

We are given two EBDD (resp. DBDD) instances, $(\Lambda_1, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1), (\Lambda_2, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$, with respect to the *same* secret $(c||s)$ (resp. $(e||s)$). Recall that EBDD (resp. DBDD) instances $(\Lambda, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ provide the promise that the secret $(\boldsymbol{c}||\boldsymbol{s}) \in \Lambda$ and $(\boldsymbol{c}||\boldsymbol{s}) \in E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma},)$ (resp. $(\boldsymbol{e}||\boldsymbol{s}) \in \Lambda$ and $(\boldsymbol{e}||\boldsymbol{s}) \in E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma},)$).

Combined hints take the two EBDD (resp. DBDD) instances and combine them into a single instance $(\Lambda', \boldsymbol{\mu}', \boldsymbol{\Sigma}')$ that captures the information from both. Specifically, Λ' will be equal to the intersection of the two lattices Λ_1, Λ_2. Since the intersection of two ellipsoids $E(\boldsymbol{\mu}_1, \mathcal{F}(\boldsymbol{\Sigma}_1)), E(\boldsymbol{\mu}_2, \mathcal{F}(\boldsymbol{\Sigma}_2))$ is not necessarily an ellipsoid, we define $E(\boldsymbol{\mu}', \mathcal{F}(\boldsymbol{\Sigma}'))$ to be an ellipsoid circumscribing their intersection. Exactly computing the minimal volume ellipsoid that circumscribes the intersection of two ellipsoids is computationally difficult. We instead use Theorem 2.10 to find $E(\boldsymbol{\mu}', \mathcal{F}(\boldsymbol{\Sigma}'))$.

$$\Lambda' = \Lambda_1 \cap \Lambda_2 \tag{18}$$

$$\boldsymbol{\mu}' \Pi_{\boldsymbol{X}} = \left(\boldsymbol{\mu}_1 \tilde{\lambda} \mathcal{F}(\boldsymbol{\Sigma}_1)^{\sim} + \boldsymbol{\mu}_2 (1-\tilde{\lambda}) \mathcal{F}(\boldsymbol{\Sigma}_2)^{\sim}\right) \boldsymbol{X}^{\sim} \tag{19}$$

$$\boldsymbol{\Sigma}' = \mathcal{F}^{-1}(k \boldsymbol{X}^{\sim}), \tag{20}$$

where

$$\boldsymbol{X} = \tilde{\lambda} \mathcal{F}(\boldsymbol{\Sigma}_1)^{\sim} + (1-\tilde{\lambda}) \mathcal{F}(\boldsymbol{\Sigma}_2)^{\sim},$$

$$k = 1 - \tilde{\lambda}(1-\tilde{\lambda})(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1) \mathcal{F}(\boldsymbol{\Sigma}_2)^{\sim} \boldsymbol{X}^{\sim} \mathcal{F}(\boldsymbol{\Sigma}_1)^{\sim} (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^T$$

and $\tilde{\lambda}$ is the unique value between $[0, 1]$ that minimizes the volume of $E(\boldsymbol{\mu}', \boldsymbol{\Sigma}')$. Theorem 2.12 provides a computationally efficient way to find $\tilde{\lambda}$. Given $\boldsymbol{\mu}' \Pi_{\boldsymbol{X}}$, the mean $\boldsymbol{\mu}'$ can be recovered from the known linear constraints on the system.

When does fusion yield a volume reduction? If $\tilde{\lambda} = 0$, then $\boldsymbol{\Sigma}' = \boldsymbol{\Sigma}_2$ and if $\tilde{\lambda} = 1$, then $\boldsymbol{\Sigma}' = \boldsymbol{\Sigma}_1$. Therefore, ellipsoid fusion does not always yield a reduction in volume. It is not hard to see that if $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2$, and if $\boldsymbol{\mu}_1 \neq \boldsymbol{\mu_2}$ are in the span of both $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_2$, then the volume of $E^{(\mathsf{Rank})}(\boldsymbol{\mu}', \boldsymbol{\Sigma}')$ is strictly smaller than both the volume of $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and of $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$. In the following, we show that fusion can still sometimes lead to a volume reduction, even in case that the volume of $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ is strictly smaller than the volume of $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$.

Theorem 4.1. *Let $\boldsymbol{c} \in \mathbb{R}^d$ denote the d-dimensional vector that has $c \in \mathbb{R}$ in each position. Let $\sigma_1^2, \sigma_2^2 \in \mathbb{R}$ be such that $\sigma_2^2 < \sigma_1^2$. Consider the rank-scaled ellipsoids $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) = E^{(\mathsf{Rank})}(\boldsymbol{0}, \sigma_1^2 \boldsymbol{I}_d)$ and $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2) = E^{(\mathsf{Rank})}(\boldsymbol{c}, \sigma_2^2 \boldsymbol{I}_d)$. Then the volume of $E^{(\mathsf{Rank})}(\boldsymbol{\mu}', \boldsymbol{\Sigma}')$ is lower than both the volume of $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ if and only if $c^2 > \sigma_1^2 - \sigma_2^2$.*

We defer the proof of Theorem 4.1 to Appendix B.

Remark 5. Consider the setting of Theorem 4.1 and let c be such that $(\sigma_1 - \sigma_2)^2 < c^2 < \sigma_1^2 - \sigma_2^2$. Note that $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2) \not\subseteq E^{(\mathsf{Rank})}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$. This can be seen using the alternate definition of a rank-scaled ellipsoid as a linear transformation and shift of the ball of radius $\sqrt{r}$, where r is the rank. Specificially, since $\|\mathbf{1}\| = \sqrt{d}$ and since

$$\|\mathbf{1} \cdot \sqrt{\boldsymbol{\Sigma}_2} + \boldsymbol{\mu}_2\|^2 = d \cdot (\sigma_2 + c)^2 > d\sigma_1^2,$$

we have that the point $\mathbf{1} \cdot \sqrt{\boldsymbol{\Sigma}_2} + \boldsymbol{\mu}_2$ is contained in $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ but not in $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$. On the other hand, the intersection of the two ellipsoids is not empty, since $\boldsymbol{\mu}_2$ is contained in both ellipsoids. Clearly $\boldsymbol{\mu}_2$ is contained in $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$. We can see that it is contained in $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ since

$$\boldsymbol{\mu}_2 \boldsymbol{\Sigma}_1^{-1} \boldsymbol{\mu}_2^T = d \cdot c^2 \cdot \frac{1}{\sigma_1^2} < d \cdot \frac{\sigma_1^2 - \sigma_2^2}{\sigma_1^2} < d.$$

However, since $c^2 < \sigma_1^2 - \sigma_2^2$, we have by Theorem 4.1 that the volume of $E^{(\mathsf{Rank})}(\boldsymbol{\mu}', \boldsymbol{\Sigma}')$ does not decrease.

Importantly, this means that the ellipsoid fusion technique does not guarantee that we obtain a lower volume ellipsoid, even in the case that $E(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1), E(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ are such that $E(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) \cap E(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2) \neq \emptyset$, $E(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) \not\subseteq E(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$, and $E(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2) \not\subseteq E(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$. This contradicts Theorem 3 of [43].

Remark 6. If $\boldsymbol{x}$ has ellipsoid norm $0 \leq a \leq 1$ with respect to $E(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and ellipsoid norm $0 \leq b \leq 1$ with respect to $E(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$, then its ellipsoid norm with respect to the fused ellipsoid is

$$0 \leq \frac{\tilde{\lambda}a + (1 - \tilde{\lambda})b + k - 1}{k} \leq 1.$$

For diagonal ellipsoids for which $0 \leq k \leq 1$, the above implies that the ellipsoid norm of $\boldsymbol{x}$ with respect to the fused ellipsoid is at most $\tilde{\lambda}a + (1-\tilde{\lambda})b \leq \max(a, b)$.

4.3 Perfect Hints, Revisited

A perfect hint on the secret $(\boldsymbol{c}||\boldsymbol{s})$ is the knowledge of $\boldsymbol{v} \in \mathbb{Z}^d$ and $\gamma \in \mathbb{Z}$, such that $\langle (\boldsymbol{c}||\boldsymbol{s}), \boldsymbol{v} \rangle = \gamma$. We assume that $\boldsymbol{v} \in \mathsf{Span}(\boldsymbol{\Sigma})$.

In our previous work, the resulting instance after incorporating a perfect hint was based on the conditional distribution of a multi-variate gaussian. Instead, using the EBDD problem, we can make use of ellipsoid-hyperplane intersection.

Recall the definition of a hyperplane in 2.8. Here, a perfect hint can represent the knowledge that the secret $(\boldsymbol{c}||\boldsymbol{s})$ is located on a hyperplane defined by $(\boldsymbol{v}, \gamma)$. Thus, we can intersect both the ellipsoid and the lattice with $H(\boldsymbol{v}, \gamma)$. An advantage of this approach presents itself when dealing with non-homogeneous instances. If the hyperplane does not cut through the center of the ellipsoid, the volume of the intersection can be lower than before.

Homogeneous instances. For homogeneous instances, the process for integrating perfect hints is fairly straightforward. It relies on the ellipsoid-hyperplane intersection procedure of the ellipsoid method (4,7). Given a DBDD instance $\mathsf{DBDD}_{\Lambda,\boldsymbol{\mu},\boldsymbol{\Sigma}}$, a homogeneous perfect hint produces a new instance $\mathsf{DBDD}_{\Lambda',\boldsymbol{\mu}',\boldsymbol{\Sigma}'}$,

$$\Lambda' = \Lambda \cap H(\boldsymbol{v}, 0) \tag{21}$$

$$\boldsymbol{\mu}' = \boldsymbol{\mu} - \alpha \frac{\boldsymbol{v}\mathcal{F}(\boldsymbol{\Sigma})}{\sqrt{\boldsymbol{v}\mathcal{F}(\boldsymbol{\Sigma})\boldsymbol{v}^T}} \tag{22}$$

$$\boldsymbol{\Sigma}' = \mathcal{F}^{-1}\left((1-\alpha^2)\left(\mathcal{F}(\boldsymbol{\Sigma}) - \frac{\mathcal{F}(\boldsymbol{\Sigma})\boldsymbol{v}^T\boldsymbol{v}\mathcal{F}(\boldsymbol{\Sigma})}{\boldsymbol{v}\mathcal{F}(\boldsymbol{\Sigma})\boldsymbol{v}^T}\right)\right) \tag{23}$$

for $-1 < \alpha \leq 1$, where α is defined as in (6).

Quantitative volume and rank reduction. Note that the rank of $\boldsymbol{\Sigma}'$ is $r-1$, where r is the rank of $\boldsymbol{\Sigma}$. Using (a generalization of) the matrix determinant lemma and properties of rdet and $\boldsymbol{\Sigma}^{\sim}$, we have that

$$\mathsf{rdet}(\boldsymbol{\Sigma}') = \left(\frac{r(1-\alpha^2)}{(r-1)}\right)^{r-1} \cdot \frac{\boldsymbol{v}\boldsymbol{v}^T}{\boldsymbol{v}\boldsymbol{\Sigma}\boldsymbol{v}^T} \cdot \mathsf{rdet}(\boldsymbol{\Sigma}).$$

Non-homogeneous instances. Note that the above formulation is not complete for non-homogeneous instances. Intersecting the lattice with a hyperplane where $\gamma \neq 0$ results in a shifted lattice *coset*. This severs the connection between the lattice points and the ellipsoid if the exact shift applied is unknown. There are simple geometric relationships that we can use to solve this problem easily, but they quickly become infeasible when applying multiple perfect hints.

Our solution is to find a point $\boldsymbol{y}$ in the lattice coset and shift the entire instance by $\boldsymbol{y}$. This means that the lattice coset is now the zero coset (i.e. it is again a lattice), and the hyperplane contains the origin, while the center of the ellipsoid is now shifted by $\boldsymbol{y}$. This allows us to achieve the smaller intersected volume due to non-homogenized instances, mentioned above. Finally, note that the solution that is obtained from this EBDD instance will now also be shifted by $\boldsymbol{y}$, and so to recover the original solution we must shift back by $\boldsymbol{y}$. Thus, we propose the following procedure.

First, observe that each perfect hint imposes a linear constraint on the space of feasible solutions in Λ. Therefore, we can combine all perfect hints into a linear system. Let $\boldsymbol{V}, \boldsymbol{\gamma}$ denote such a system, where $\boldsymbol{V} \in \mathbb{Z}^{d \times t}, \boldsymbol{\gamma} \in \mathbb{Z}^t$, and t is the number of perfect hints to be integrated. Further, let $\boldsymbol{y} \in \Lambda$ be a solution to the above system (i.e. $\boldsymbol{y}\boldsymbol{V}^T = \boldsymbol{\gamma}$). For non-homogeneous instances, $\boldsymbol{y}$ will not correspond to the origin. We then shift the ellipsoid (and thus the secret) by $\boldsymbol{y}$ so that $((\boldsymbol{c}||\boldsymbol{s}) - \boldsymbol{y})\boldsymbol{V}^T = 0$. For non-homogeneous instances, given a EBDD instance $\mathsf{EBDD}_{\Lambda,\boldsymbol{\mu},\boldsymbol{\Sigma}}$, we obtain a new instance
$\mathsf{EBDD}_{\Lambda'',\boldsymbol{\mu}'',\boldsymbol{\Sigma}''}$, where

$$\Lambda' = [\Lambda \cap H(\boldsymbol{V}, \boldsymbol{\gamma})] + \boldsymbol{y} \tag{24}$$

$$E^{(\mathsf{Rank})}(\boldsymbol{\mu}'', \boldsymbol{\Sigma}'') = E^{(\mathsf{Rank})}(\boldsymbol{\mu}' - \boldsymbol{y}, \boldsymbol{\Sigma}') \tag{25}$$

and $E^{(\mathsf{Rank})}(\boldsymbol{\mu}', \boldsymbol{\Sigma}')$ is obtained by applying (22) and (23) for all perfect hints described by $\boldsymbol{V}, \boldsymbol{\gamma}$. Note also that (with a slight abuse of notation $H(\boldsymbol{V}, \boldsymbol{\gamma})$ refers to the intersection of hyperplanes that meet the constraints of the $\boldsymbol{V}$, $\boldsymbol{\gamma}$ linear system. We note that for our proposed Kannan Ellipsoid embedding, we can solve a system of linear diophantine equations to obtain $\boldsymbol{y}$, as the lattice is $\mathbb{Z}^d$ when perfect hints are integrated first.

Such a system can be solved efficiently using the LLL algorithm or through computing the Hermite Normal Form. We would like to make sure our offset $\boldsymbol{y}$ does not become too large, as this induces numerical errors. Solving the combined system $(\boldsymbol{c}||\boldsymbol{s})\boldsymbol{V} = \boldsymbol{\gamma}$ AND $[(\boldsymbol{c}||\boldsymbol{s})\boldsymbol{B} + (\boldsymbol{b}||\boldsymbol{0})](\boldsymbol{I}_m||\boldsymbol{A})^T - q\boldsymbol{c} = \boldsymbol{b}$ can be done over the integers, (the above uses $\boldsymbol{B}$ defined in (14)) but the solutions become extremely large. Instead, as long as there are at most n perfect hints to integrate, we are left with enough free parameters that we can jointly solve the diophantine system and the LWE equations modulo q. More specifically, we first solve the diophantine system $[(\boldsymbol{e}||\boldsymbol{s}) - (\boldsymbol{b}||\boldsymbol{0})]\boldsymbol{B}^{-1})\boldsymbol{V} = \boldsymbol{\gamma}$ to get a small integer solution $\bar{\boldsymbol{y}}$. In fact, we obtain a solution family $\{\bar{\boldsymbol{y}} + \boldsymbol{x}\boldsymbol{U} \,|\, \boldsymbol{x} \in \mathbb{Z}^{d-t}\}$, where $\boldsymbol{U}$ is the unimodular transformation matrix of the Hermite Normal Form of $\boldsymbol{V}$ of dimension $(d-t) \times d$. Then, we solve the system, $(\bar{\boldsymbol{y}} + \boldsymbol{x}\boldsymbol{U})(\boldsymbol{I}_m||\boldsymbol{A})^T) = \boldsymbol{b}$ mod q, with m equations and $d - t \geq m$ variables. Thus, our final $\boldsymbol{y} = \bar{\boldsymbol{y}} + \boldsymbol{x}\boldsymbol{U}$ is a solution to both systems. This then bounds the value of each coordinate of $\boldsymbol{y}$ by at most q (given the solutions of the first system are sufficiently small). Finally, $\boldsymbol{y}$ must be converted into the $(\boldsymbol{c}||\boldsymbol{s})$ solution space via the relation in Section 3.3.

4.4 Short Vector Hints, Revisited

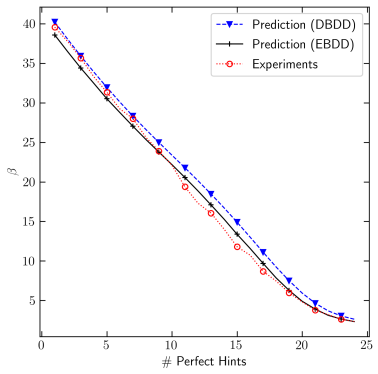
A short vector hint on the lattice Λ is the knowledge of a short vector $\boldsymbol{v}$ such that $\boldsymbol{v} \in \Lambda$.

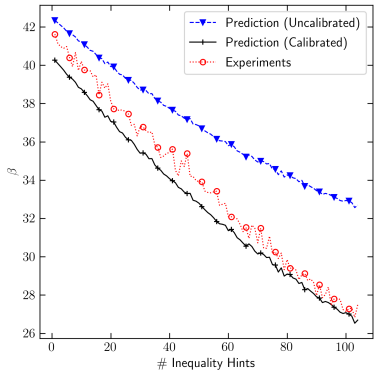
Originally, in [17], these short vector hints were features of the LWE lattice, or the design of a specific cryptographic scheme. The main intuition is that if one knows a "good enough" lattice vector $\boldsymbol{v}$ that is not the secret, then that vector can be treated as a fixed basis vector. Projecting orthogonally to $\boldsymbol{v}$, then results in a tradeoff between dimension and lattice volume that can result in an easier instance to solve. The lost dimensions can be recovered through solving a system of linear equations over the rationals.

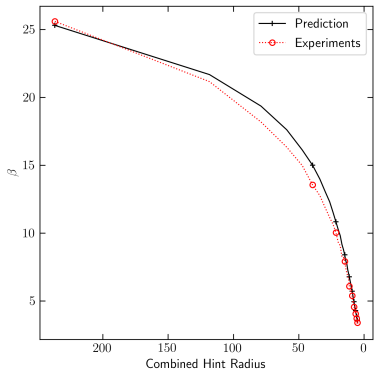
Note that this intuition no longer holds when discussing the ellipsoidal embedding of LWE into DBDD. The idea of a good basis vector for the lattice makes little sense when the lattice is $\mathbb{Z}^d$. Indeed, a short vector in the original Kannan embedding lattice can be transformed into a point in $\mathbb{Z}^d$, but it is not clear what projecting against this point means algebraically.

As such, it is more sensible to (partially) revert back to the Kannan embedding lattice to integrate short vector hints. To accomplish this, we perform the following coordinate space transformation:

$$\Lambda' = \Lambda\boldsymbol{B}$$
$$E(\boldsymbol{\mu}', \boldsymbol{\Sigma}') = E(\boldsymbol{\mu}\boldsymbol{B}, \boldsymbol{B}^T\boldsymbol{\Sigma}\boldsymbol{B})$$







LWE Parameters $n = m = 70,\ q = 3301,\ \sigma^2 = 20$	
Hint Type	Parameter
Perfect hints	–
Inequality hints	± 2
Combined hints	$[20, (20/49)]$

The hint vectors $\boldsymbol{v}$ for inequality and perfect hints were chosen as random ternary vectors of weight 5.

Fig. 1: Experimental verification of the bikz predictions for each type of hint. Each data point was averaged over 256 samples. Inequality and perfect hint validation were conducted by integrating successively larger numbers of hints. Combined hint validation was conducted by integrating instances with decreasing ellipsoid volume (see (26)).

We refer to this transformation as "partial isotropization," as Λ' would be the result of isotropization in an instance where no hints were integrated. Note that if the instance was not homogeneous, then the resulting secret is $(\boldsymbol{e}+\boldsymbol{b}||\boldsymbol{s})$. From here, short vector hints can applied as in the prior work [17]. Following this, we perform homogenization and isotropization as normal.

As with the prior work, it is crucial to integrate these hints last. Thus, this coordinate space transformation will not need to be applied to other hints.

5 Experimental Validation and Applications

5.1 Experimental Validation

For (1) perfect hints, (2) inequality hints, (3) combined hints, we compare the bikz predicted by our tool with the bikz actually needed to launch the attack and recover the LWE secret/error. For (1) and (2), we choose the same set of LWE parameters for the initial instances as in [17], and integrate an increasing number of hints of each type.

For (1), the curve labeled "Prediction (DBDD)" uses DBDD instances obtained by integrating perfect hints via the approach of [17], while "Prediction (EBDD)" uses our new approach. We display a single "Experiments" curve since the EBDD and DBDD instances differ only by a scaling factor, which does not impact the bikz (as verified experimentally).

For (2), we create inequality hints by simulating a known (small) absolute error. Given a hint vector $\boldsymbol{v}$, we create the hint $\langle \boldsymbol{v}, (\boldsymbol{e}||\boldsymbol{s}) \rangle \geq \gamma - 2$, where γ is the inner product of $\boldsymbol{v}$ with the correct secret. Our predicted bikz–the "calibrated" estimate–take into account the length of the shortest vector in our final lattice as in (11) as it deviates from the expected value assumed in (13). When integrating large numbers of inequality hints the ellipsoid norm of the secret w.r.t. the EBDD instance is significantly lower than the rank, while (13) holds under the assumption that the ellipsoid norm is approximately equal to the rank. This leads to overestimation of the hardness when applying (13)–the "uncalibrated" estimate. As such, we also use this calibration when examining the hardness loss resulting from decryption failures in Section 5.2.

For (3) we use the same set of LWE parameters to construct an initial EBDD (see Section 3.3) instance. We then perform combined hints with the initial EBDD instance and each of the EBDD instances corresponding to the ellipsoids

$$E^{(\mathsf{Rank})}((\boldsymbol{c}||\boldsymbol{s}) + \mathcal{E}, (20/i) \cdot \boldsymbol{I}_{m+n}) \tag{26}$$

where $\mathcal{E} \sim \mathcal{N}(\boldsymbol{0}, (20/i) \cdot \boldsymbol{I}_{m+n})$ for $i \in [1, 49]$. See Figure 1 for details.

5.2 Decryption Failures, Revisited

Decryption failures exactly correspond to inequality hints from Section 4.1. Thus, the naive approach to running a decryption failure attack is to iteratively integrate each decryption failure as an inequality hint, obtaining a series of ellipsoids with volumes that are strictly decreasing. We experimented with applying our inequality hint approach to the recent decryption failure attack of Fahr et al. [23]. In their work, the public key of FrodoKEM [6] (NIST level 1 frodo-640) was altered by injecting faults via the Rowhammer exploit to significantly increase the decryption failure rate (by effectively lowering the decryption failure threshold). This enables an attacker to search for failing (honestly generated) ciphertexts in a reasonable amount of time and thus this scenario is more amenable for the experiments in this section.

When instantiating our naive approach in the Fahr et al. [23] setting, we find that while the first batch of hints reduce the volume as expected (e.g. for the first hint if a vector $\boldsymbol{w}$ causes decryption to fail with probability p over choice of secret key, then we see a reduction of volume by nearly a factor of p), hints quickly lose their efficacy, until almost no progress is made in terms of volume reduction as new hints are integrated. In fact, we found that the center of the successive ellipsoids obtained by integrating a sequence of inequality hints converges very quickly to a feasible solution (i.e. a solution that satisfies all the linear constraints). This is due to the one-sided nature of the linear inequalities corresponding to decryption failures. The intersection of a finite number of halfspaces pertaining to these inequalities is an unbounded region of space. Thus, a feasible solution sufficiently inside this region will not be affected by any new constraints of the same form.

After ≈ 200 hints for simulated failures on Frodo-640 [6] the center, $\boldsymbol{\mu}$ itself satisfied *all prior and future* inequality hints, which corresponds to a terminating condition in the ellipsoid method. The full key recovery attack of Fahr et al. [23] required $\approx 100,000$ hints, so reaching a feasible solution after 200 hints is quite surprising. Unfortunately, the Euclidean distance between $\boldsymbol{\mu}$ and the true LWE secret/error remained quite large, so $\boldsymbol{\mu}$ itself was not a good candidate solution. Nevertheless, we found that $\boldsymbol{\mu}$ contains a lot of information about $\boldsymbol{s}$: We argue next that if $\boldsymbol{\mu}$ satisfies all hints, then $\langle \boldsymbol{\mu}, \boldsymbol{s} \rangle \geq \langle \boldsymbol{s}, \boldsymbol{s} \rangle \approx \sigma_s^2 \cdot n$.

As observed in [17], the distribution of hint vectors $\boldsymbol{w}$ decomposes as $\boldsymbol{w} = \alpha \cdot \boldsymbol{s}/||\boldsymbol{s}|| + \boldsymbol{w}'$, where α is a random variable with expectation $\approx t/||\boldsymbol{s}||$ (where t is the decryption failure threshold) and $\boldsymbol{w}'$ is a zero-centered random variable orthogonal to $\boldsymbol{s}$. So for a fixed center $\boldsymbol{\mu}$,

$$\mathbb{E}[\langle \boldsymbol{\mu}, \boldsymbol{w} \rangle] = \mathbb{E}[\alpha] \cdot \langle \boldsymbol{w}, \boldsymbol{s} \rangle / ||\boldsymbol{s}|| \approx t \cdot \langle \boldsymbol{w}, \boldsymbol{s} \rangle / ||\boldsymbol{s}||^2 = t \cdot \langle \boldsymbol{w}, \boldsymbol{s} \rangle / \langle \boldsymbol{s}, \boldsymbol{s} \rangle.$$

If we find empirically, for a sufficiently large hint database, that $\mathbb{E}[\langle \boldsymbol{\mu}, \boldsymbol{w} \rangle] \geq t$– which occurs if $\boldsymbol{\mu}$ satisfies all previous and future hint inequalities–it implies that $\langle \boldsymbol{w}, \boldsymbol{s} \rangle \geq \langle \boldsymbol{s}, \boldsymbol{s} \rangle$. Beyond this, we observed "overfitting behavior," in that after the integration of 1000 hints, the numerical precision required to compute subsequent ellipsoids precluded further progress.

Inequality Hints with Regeneration. To solve both the issue of stalled progress and the overfitting issue we developed the following regeneration approach: When the center $\boldsymbol{\mu}$ of the successive ellipsoids becomes such that $\langle \boldsymbol{\mu}, \boldsymbol{s} \rangle \geq \sigma_s^2 \cdot d$, we simply use $\boldsymbol{\mu}$ itself to perform an inequality hint on a fresh EBDD or DBDD instance. Specifically, we regenerate the initial ellipsoid according to our embedding and integrate the hint $\langle \boldsymbol{\mu}, \boldsymbol{s} \rangle \geq \sigma_s^2 \cdot d$. Once this is done, we find that we can again make progress for some time by integrating more decryption failures. When progress stalls again, we simply regenerate again.

An attacker cannot directly check the condition for regeneration ($\langle \boldsymbol{\mu}, \boldsymbol{s} \rangle \geq \sigma_s^2 \cdot d$). Instead, the attacker can do one of the following: (1) when progress stalls, assume it is due to the fact that $\langle \boldsymbol{\mu}, \boldsymbol{s} \rangle \geq \sigma_s^2 \cdot d$ and regenerate based on this hint (2) use the empirical value of $\mathbb{E}[\langle \boldsymbol{w}, \boldsymbol{\mu} \rangle]$, calculated using all $\boldsymbol{w}$ corresponding to

failing ciphertexts in the attacker's database. In the case that $\langle \boldsymbol{\mu}, \boldsymbol{s} \rangle \geq \sigma_s^2 \cdot d$, we expect $\mathbb{E}[\langle \boldsymbol{w}, \boldsymbol{\mu} \rangle] \geq (1+\epsilon) \cdot t$ where ϵ is a safety margin due to the uncertainty in the empirical expected value. The second approach can allow the attacker to regenerate earlier. E.g. the attacker can already choose to regenerate when $\langle \boldsymbol{\mu}, \boldsymbol{s} \rangle \geq 1/2 \cdot \sigma_s^2 \cdot d$ by checking whether $\mathbb{E}[\langle \boldsymbol{w}, \boldsymbol{\mu} \rangle] \geq 1/2 \cdot (1+\epsilon) \cdot t$.

To evaluate the effectiveness of regeneration compared to the full dimensional approximate hint-based hardness estimates in [17], we continued to use the scenario of Fahr et al. [23]. For our experiment, we generated several simulated public keys (i.e. we directly modified honestly generated public keys to reproduce the result of the fault injection). Then, we searched for 4000 failing ciphertexts for each key. We integrated these 4000 hints as full-dimensional approximate hints and inequality hints with regeneration on two separate DBDD instances. We set the decryption failure threshold $t = 1024$ corresponding to the effect of the fault injection attack. The results can be seen in Figure 2.

	Full-Dimen. Approx. Hints	Inequality Hints				
		Key 1	Key 2	Key 3	Key 4	Key 5
Ellipsoid Norm	–	322.61	213.55	590.57	546.58	485.14
Ciphertexts	**4000**	1224	1106	959	986	965
Final BKZ–β	**307.85**	295.68	279.78	284.47	283.67	284.70

Fig. 2: Comparison of BKZ blocksize β estimates for a fault injection assisted decryption failure attack using 4000 failing ciphertexts for 5 different (simulated) poisoned Frodo-640 public keys. The final calibrated estimates result from shrinking the final ellipsoid by a factor of $\mathsf{Rank}(\boldsymbol{\Sigma})/\|\boldsymbol{s}\|_{\boldsymbol{\Sigma}}$.

When the LWE secret follows a multivariate Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, the expected value of its rank-scaled ellipsoid norm with respect to $E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the rank. This is assumed to be the case when estimating the BKZ–β via equation (13). In our case, the obtained $E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ no longer represents a multivariate Gaussian distribution and so the above no longer holds: Observe that in Figure 2, the ellipsoid norm of the secret is less than half of the rank (the rank is 1279 when considering a single fault injection per column of a Frodo-640 public key). To account for this, we calibrate the estimated BKZ–β using equation (11). This is equivalent to scaling the $E^{(\mathsf{Rank})}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ until the ellipsoid norm of the secret is equal to the rank, and then applying equation (13).

We further note that while it is possible to obtain a β estimate using the full-dimensional approximate hint approach from the prior work by utilizing the ultra-lightweight version of the framework [17], it is not possible to compute the uSVP lattice basis itself required to run BKZ due to high computational overhead. In contrast, our new inequality hint-based method is more efficient and so allows us to compute the final ellipsoid covariance matrix necessary for the reduction from EBDD to uSVP, and consequently would allow one to run a

full key recovery attack given a sufficient number of hints.

A geometric approach to failure boosting. The volume reduction incurred when integrating an inequality hint is almost entirely determined by the geometric parameter α defined in (6). The larger α is, the more we can reduce the set of secrets that are consistent with an inequality hint. Conversely, the α value for a candidate hint, given the current information encapsulated by the ellipsoid, can be used as a proxy for the probability that the query will lead to decryption failure: The smaller α is, the higher the probability of decryption failure. In order for a ciphertext to cause a failure, it either needs to have an abnormally high norm (increasing the denominator of α), or be highly correlated with the secret (decreasing the numerator of α). Thus, computing this α value with respect to the current ellipsoid, *before* submitting a decryption query provides a geometric analogue to the failure boosting approach of D'anvers et al. [20].

Using our regeneration approach, we were able to achieve improved β levels compared to the full dimensional approximate hints approach of [17] by integrating only 959-1224 number of hints in total, as opposed to 4000 hints. To achieve this, we used a greedy algorithm that at each stage chose the hint with the largest α value to integrate. Using this approach, we can profile the range of α values for the i-th hint and obtain a range $[\alpha^i_{\mathsf{low}}, \alpha^i_{\mathsf{high}}]$. We can then run the following online algorithm when making decryption queries to find the i-th hint to integrate into the ellipsoid: Let S be the set of all failing decryption queries made up to this moment. First, search S to try to find a query with α value in the range $[\alpha^i_{\mathsf{low}}, \alpha^i_{\mathsf{high}}]$ with respect to the current ellipsoid. If such a query is found, integrate it into the ellipsoid. Otherwise, generate a set S' of candidate hints of some calibrated size s'. For each $\boldsymbol{w} \in S'$, compute its α value. If $\alpha \notin [\alpha^i_{\mathsf{low}}, \alpha^i_{\mathsf{high}}]$ then remove $\boldsymbol{w}$ from S'. Sort the entire set S' from smallest to largest α value. Make decryption queries in this order until a failing ciphertext is found. Once found, add $\boldsymbol{w}$ to S and integrate $\boldsymbol{w}$ into the current ellipsoid.

We ran the above experiment on a small scale again using the scenario presented by Fahr et al. [23]. Here, instead of generating a database of 4000 failing ciphertexts, we generated a database of $100k$ candidate ciphertexts (note that in the Fahr et al. [23] there is a way to filter candidate ciphertexts that have a relatively high chance of causing a decryption failure). Of these, only 34 actually caused decryption failures (this is consistent with the decryption failure rate (DFR) for filtered ciphertexts reported by Fahr et al. [23]). We integrated these 34 failing ciphertexts as inequality hints in order of decreasing α, each time calculating the histogram of α values for the remaining ciphertext database. Figure 3 shows the evolution of the histogram as more hints are integrated.

Next, we looked to quantify the number of decryption queries required to find all 34 failures, compared to naively querying the database by a linear scan. All 34 failures had α values in the $[0.07, 0.12]$ range, so we only submitted decryption queries for ciphertexts with corresponding α values at each step sorted in ascending order. To obtain all 34 decryption failures in the database, we found that it took 39785 queries versus 94894 for a linear scan.

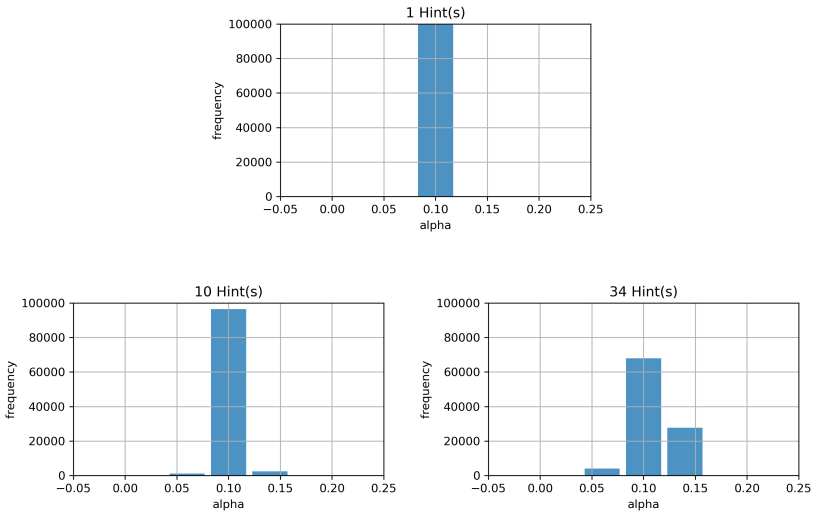


Fig. 3: Histograms of α values for a database of $100k$ ciphertexts after integrating 1 (top), 10 (bottom-left), and 34 (bottom-right) inequality hints based on the failing ciphertexts in the database.

5.3 Combining Decryption Failure and SCA

We illustrate our "Combined Hints" approach from Section 4.2 by combining information on a single $(\boldsymbol{e}||\boldsymbol{s})$ pair from a decryption failure and a side-channel attack. In a recent work, Fahr et al. [23] showed that, for FrodoKEM, obtaining m' number of vectors corresponding to random decryption failures, scaling them by a constant that depends on the parameters of the cryptosystem, and taking their coordinate-wise mean, approximates a draw from the distribution $\mathcal{D}' := (\boldsymbol{e}||\boldsymbol{s}) + \mathcal{W}''_{(m)}$, where the error $\mathcal{W}''_{(m)}$ is a d-dimensional Gaussian with mean $\mathbf{0}$ and covariance matrix $\sigma_{df}^2 \cdot \boldsymbol{I}_d$, where $\sigma_{df}^2 \leq d^2\sigma_1^6/(t^2m')$, σ_1^2 is the error of the original distribution, d is the dimension of the LWE secret/error, and t is the decryption failure threshold. Rearranging terms, given a draw $\boldsymbol{\mu}_{df} \sim \mathcal{D}'$, the secret is equal to $\boldsymbol{s} = \boldsymbol{\mu}_{df} + \mathcal{W}''_{(m)}$. This means that the secret is contained in the rank-scaled ellipsoid $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_{df}, \boldsymbol{\Sigma}_{df})$, where $\boldsymbol{\Sigma}_{df} = \sigma_{df}^2 \cdot \boldsymbol{I}_d$. Note that we could have used the results of Section 5.2 to obtain a DBDD instance with better parameters than the one corresponding to $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_{df}, \boldsymbol{\Sigma}_{df})$. Further, this instance would no longer correspond to a non-centered Gaussian distribution over the secret, and in fact the PDF of the secret would be unknown. Obtaining such a DBDD instance with the target σ_{df}^2 value needed for our experiment would be very computationally intensive. Therefore, for our illustration of the combined hints technique, we use the rank-scaled ellipsoid $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_{df}, \boldsymbol{\Sigma}_{df})$ described above to capture the DBDD instance obtained from the decryption failure information.

Bos et al. [13] studied the feasibility of single-trace power analysis of the Frodo Key Encapsulation Mechanism (FrodoKEM). Subsequently, Dachman-Soled et

al. [17] used this information to conduct a side-channel attack on FrodoKEM on various parameter sets (CCS1, CCS2, CCS3, CCS4, NIST1, NIST2). Dachman-Soled et al. [17] used the score tables constructed from Bos et al. [13] to form an a posteriori distribution incorporating the side-channel information and used the information from the distribution tables to "guess" a large subset of coordinates when the confidence in the guess (where the confidence was calculated using the aforementioned score tables) was sufficiently high.

5.3.1 The Baseline Approach The prior work [17] incorporated the side-channel information, represented by DBDD instance with $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_{sc}, \boldsymbol{\Sigma}_{sc})$, using approximate *a posteriori hints*. In this method, the mean and covariance matrix of the a posteriori distribution (say on the $\boldsymbol{s}$ variables only) is calculated and then fully replaces the part of the covariance matrix in the DBDD instance that corresponds to the $\boldsymbol{s}$ variables. This method was suggested by [17] as an alternative to "conditioning" approximate hints. In our case, both $\boldsymbol{\Sigma}_{df}$ and $\boldsymbol{\Sigma}_{sc}$ are diagonal matrices. Therefore, the *a posteriori hints* approach, which we refer to as the *Baseline approach*, yields the following rank-scaled ellipsoid, $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_{ba}, \boldsymbol{\Sigma}_{ba})$: For each $i \in [d]$, if $\boldsymbol{\Sigma}_{sc}[i][i] \leq \boldsymbol{\Sigma}_{df}[i][i]$, set $\boldsymbol{\Sigma}_{ba}[i][i] = \boldsymbol{\Sigma}_{sc}[i][i]$ and $\boldsymbol{\mu}_{ba}[i] = \boldsymbol{\mu}_{sc}[i]$. Otherwise, set $\boldsymbol{\Sigma}_{ba}[i][i] = \boldsymbol{\Sigma}_{df}[i][i]$ and $\boldsymbol{\mu}_{ba}[i] = \boldsymbol{\mu}_{df}[i]$.

5.3.2 Ellipsoid/Ellipsoid Intersection. For an ellipsoid $E = (\mu, \Sigma)$ (resp. rank-scaled ellipsoid $E^{\mathsf{rank}} = (\mu, \Sigma)$), we denote by $E_S = (\mu_S, \Sigma_S)$ (resp. $E_S^{\mathsf{rank}} = (\mu_S, \Sigma_S)$) the ellipsoid (resp. rank-scaled ellipsoid) resulting from the restriction of the center and shape matrix of E (resp. E^{rank}) to a set of coordinates S. For an ellipsoid E, we denote by n_E the ellipsoid norm of the correct solution with respect to E. Let $E_{DF}^{\mathsf{rank}} = E^{(\mathsf{Rank})}(\boldsymbol{\mu}_{df}, \boldsymbol{\Sigma}_{df})$ and $E_B^{\mathsf{rank}} = E^{(\mathsf{Rank})}(\boldsymbol{\mu}_{ba}, \boldsymbol{\Sigma}_{ba})$. Restricting to the set S of secret (no error coordinates), let $E_{int,S}^{\mathsf{rank}} = E^{(\mathsf{Rank})}(\boldsymbol{\mu}_{int,S}, \boldsymbol{\Sigma}_{int,S})$ be the ellipsoid circumscribing the intersection of $E_{DF,S}^{\mathsf{rank}} = E^{(\mathsf{Rank})}(\boldsymbol{\mu}_{df,S}, \boldsymbol{\Sigma}_{df,S})$ and $E_{B,S}^{\mathsf{rank}} = E^{(\mathsf{Rank})}(\boldsymbol{\mu}_{ba,S}, \boldsymbol{\Sigma}_{ba,S})$. Let the diagonal of $\boldsymbol{\Sigma}_{ba,S}$ be denoted by $(\sigma_{2,1}^2, \ldots, \sigma_{2,n}^2)$. Let $\boldsymbol{c} = \boldsymbol{\mu}_{ba,S} - \boldsymbol{\mu}_{df,S}$. We simplify (19) and (20) as follows:

$$\mathcal{F}(\boldsymbol{\Sigma}_{int,S}) = k\boldsymbol{X}^{-1}; \qquad \boldsymbol{X} = \tilde{\lambda}\mathcal{F}(\boldsymbol{\Sigma}_{df,S})^{-1} + (1-\tilde{\lambda})\mathcal{F}(\boldsymbol{\Sigma}_{ba,S})^{-1}$$

$$\boldsymbol{\mu}_{int} = (\boldsymbol{\mu}_{df,S}\tilde{\lambda}\mathcal{F}(\boldsymbol{\Sigma}_{df,S})^{-1} + \boldsymbol{\mu}_{ba,S}(1-\tilde{\lambda})\mathcal{F}(\boldsymbol{\Sigma}_{df,S})^{-1})\boldsymbol{X}^{-1}$$

$$k = 1 - \tilde{\lambda}(1-\tilde{\lambda}) \cdot \frac{1}{n} \sum_{i \in [n]} \frac{c_i^2}{\tilde{\lambda}\sigma_{2,i}^2 + (1-\tilde{\lambda})\sigma_{df}^2}$$

and $\tilde{\lambda} \in [0,1]$ is the value that minimizes the determinant of $\boldsymbol{\Sigma}_{int,S}$. Specifically,

$$\det(\boldsymbol{\Sigma}_{int,S}) = k^n \cdot \prod_{i \in [n]} \left(\frac{\tilde{\lambda}}{\sigma_{df}^2} + \frac{1-\tilde{\lambda}}{\sigma_{s_2,i}^2} \right)^{-1}. \tag{27}$$

The terms in the product on the right side of equation (27) correspond to weighted harmonic means of σ_{df}^2 and $\sigma_{2,i}^2$, for each i. While the harmonic mean

	Baseline Approach	Combined Hints	
		Condition 1 unknown/known	Condition 2 unkown/known
Original BKZ–β	**268.83**	–	–
DF BKZ–β	**203.02**	–	–
SC BKZ–β before guess	**114.22**	–	–
SC BKZ–β after guess	**68.65**	–	–
$\ln(V_{int}/V_{base})$	–	-26.49/-30.47	-23.46/-32.24
BKZ–β before guesses	**97.86**	95.91/94.83	96.22/94.66
Number of guesses	**190**	190/190	190/190
Guess Success %	**0.76**	0.76/0.76	0.76/0.76
Final BKZ–β	52.20	50.19/ 49.18	50.50/ 49.00

Fig. 4: **Comparison of bikz estimates for FrodoKEM with CCS1 parameters.** Results are the average of 150 randomly generated instances. Starting from the top row, we report the original bikz, the bikz for only the decryption failure attack, and the bikz for only the side channel attack, before and after guesses (throughout we condition on all guesses being correct). We compare the baseline approach (Section 5.3.1) with two combined hints approaches using Condition 1 or 2 (Section 5.3.3) to select the set of coordinates for intersection. For each, we consider the known and unknown cases (Section 5.3.4). We next report the ln of the ratio of the volumes of the intersected and baseline ellipsoids (for the unknown case, these are reported after calibration (Section 5.3.4)). For each, we report the bikz without guesses, the number of guesses, the probability that all guesses are correct and the final bikz after guesses.

tends towards the smaller element, it is at least as large as the minimum of the two values. This is then compensated by multiplication by k, which is always at most 1. However, due to the negative influence of the harmonic mean on the final determinant, we experiment with intersecting only on coordinates i for which the gap between σ^2_{df} and $\sigma^2_{2,i}$ is not too large.

5.3.3 Conditions 1 and 2. We consider two candidate methods of performing intersection: In the first method, referred to as **Condition 1**, we restrict the intersection to the dimension $n-g$ ellipsoids (where g is the number of guesses) corresponding to the coordinates of the LWE secret (but not the error) that are not guessed. This is essentially equivalent to performing the intersection after guesses are made on the remaining coordinates of the LWE secret. For the remaining coordinates, we follow the baseline approach. In the second method, referred to as **Condition 2**, we restrict the intersection to the dimension n' ellipsoids corresponding to the coordinates i of the LWE secret (but not the error), for which $\sigma^2_{2,i}$ is in the range $[\frac{\sigma^2_{df}}{5}, \sigma^2_{df}]$. For the remaining coordinates, we again follow the baseline approach.

5.3.4 The known and unknown cases Let $E_{DF,S} = E(\boldsymbol{\mu}_{df,S}, \mathcal{F}(\boldsymbol{\Sigma}_{df,S}))$ and $E_{B,S} = E(\boldsymbol{\mu}_{ba,S}, \mathcal{F}(\boldsymbol{\Sigma}_{ba,S}))$. We restrict ellipsoids $E_{DF,S}$ and $E_{B,S}$ to a set

of coordinates $P \subseteq S$ corresponding to Condition 1 or 2, yielding $E_{DF,P}$ and $E_{B,P}$. These are then intersected to yield $E_{int,P}$, and $E_{int,P}$ is substituted for the set of P coordinates in $E_{B,S}$ yielding $E_{int,S}$. To maintain consistency of hardness estimates, we would like to keep $\mathsf{n}_{E_{int,P}} = \mathsf{n}_{\mathsf{E_{B,P}}}$. Further, ellipsoid/ellipsoid intersection performs best when intersecting two ellipsoids $E_{DF,P}$ and $E_{B,P}$ such that $\mathsf{n}_{E_{DF,P}} = \mathsf{n}_{E_{B,P}} = 1$, since points on the surface of both $E_{DF,P}$ and $E_{B,P}$ also lie on the surface of $E_{int,P}$.

Assuming that $\mathsf{n}_{E_{DF,P}}$ and $\mathsf{n}_{E_{B,P}}$ are known, we scale $E_{DF,P}$ by $\mathsf{n}_{E_{DF,P}}$ and $E_{B,P}$ by $\mathsf{n}_{E_{B,P}}$, so that the correct solution lies on the surface of both scaled ellipsoids, and hence on the surface of $E_{int,P}$. We then scale $E_{int,P}$ by $1/\mathsf{n}_{E_{B,P}}$, to ensure that $\mathsf{n}_{E_{int,P}} = \mathsf{n}_{E_{B,P}}$. This yields the optimal volume reduction while maintaining the norm constraint but requires knowledge of $\mathsf{n}_{E_{DF,P}}$ and $\mathsf{n}_{E_{B,P}}$. We refer to this case as the **known** case.

While $\mathsf{n}_{E_{DF,P}}$ is fairly stable (since the decryption failure ellipsoid is a multivariate Gaussian in our experiments), $\mathsf{n}_{E_{B,P}}$ can fluctuate. We therefore also explore the case in which the adversary is not presumed to know $\mathsf{n}_{E_{DF,P}}$ and $\mathsf{n}_{E_{B,P}}$. We refer to this case as the **unknown** case, and we next describe the algorithm for this case. We find experimentally that with probability at least $1/2$, $\mathsf{n}_{E_{DF,P}} \leq 0.9 \cdot \mathsf{n}_{E_{B,P}}$. We scale $E_{DF,P}$ by 0.9 before intersection. In the case that indeed $\mathsf{n}_{E_{DF,P}} \leq 0.9\mathsf{n}_{E_{B,P}}$, we have by Remark 6, that $\mathsf{n}_{E_{int,P}} \leq \mathsf{n}_{E_{B,P}}$. In the case that $\mathsf{n}_{E_{DF,P}} > 0.9\mathsf{n}_{E_{B,P}}$, it may be the case that $\mathsf{n}_{E_{int,P}} > \mathsf{n}_{E_{B,P}}$[7] To take into account the fact that $\mathsf{n}_{E_{int,P}}$ can now be smaller or larger than $\mathsf{n}_{E_{B,P}}$, we use Equation (11) to calibrate the predicted β value with respect to the entire instance (including error coordinates).

We present our experimental results with decryption failure information modeled as described above, with $\sigma_{df}^2 = 0.25$, and with side-channel data obtained from the single trace attack of Bos et al. [13] on FrodoKEM. As in [17], we incorporate guesses when the side-channel distribution for a secret coordinate allows for a high confidence guess. Figure 4 displays the predicted hardness (in bikz) of the original and baseline DBDD instances, the intersected instances obtained using Condition 1 and 2, in both the known and unknown cases, both with and without guesses, for the CCS1 parameter set. [8] Our approach lowers the required number of bikz as compared to the baseline approach by 2-3 bikz.

References

1. Alagic, G., Apon, D., Cooper, D., Dang, Q., Dang, T., Kelsey, J., Lichtinger, J., Miller, C., Moody, D., Peralta, R., Perlner, R., Robinson, A., Smith-Tone, D., Liu, Y.K.: Status report on the third round of the nist post-quantum cryptography standardization process. Technical Report : NIST Internal Report (NISTIR) 8413, U.S. Department of Commerce, Washington, D.C. (2022)

[7] Note that in our experiments it was always the case that $1/0.9\mathsf{n}_{E_{DF,P}} \leq 1$ so the intersection is always non-empty.

[8] The number of bikz reported in our table for the SCA-only attack differs slightly from the bikz reported in [17], as we use the updated code found here: https://github.com/lducas/leaky-LWE-Estimator/tree/fix_extreme_hints2.

2. Albrecht, M., Cid, C., Faugère, J.C., Fitzpatrick, R., Perret, L.: On the complexity of the Arora-Ge Algorithm against LWE. In: SCC 2012 – Third international conference on Symbolic Computation and Cryptography, Castro Urdiales, Spain (Jul 2012) 93–99
3. Albrecht, M.R., Bai, S., Li, J., Rowell, J.: Lattice reduction with approximate enumeration oracles - practical algorithms and concrete performance. In Malkin, T., Peikert, C., eds.: Advances in Cryptology – CRYPTO 2021, Part II. Volume 12826 of Lecture Notes in Computer Science., Virtual Event, Springer, Heidelberg, Germany (August 16–20, 2021) 732–759
4. Albrecht, M.R., Cid, C., Faugère, J.C., Fitzpatrick, R., Perret, L.: On the complexity of the BKW algorithm on LWE. Cryptology ePrint Archive, Report 2012/636 (2012) https://eprint.iacr.org/2012/636.
5. Albrecht, M.R., Göpfert, F., Virdia, F., Wunderer, T.: Revisiting the expected cost of solving uSVP and applications to LWE. In Takagi, T., Peyrin, T., eds.: Advances in Cryptology – ASIACRYPT 2017, Part I. Volume 10624 of Lecture Notes in Computer Science., Hong Kong, China, Springer, Heidelberg, Germany (December 3–7, 2017) 297–322
6. Alkim, E., Bos, J.W., Ducas, L., Longa, P., Mironov, I., Naehrig, M., Nikolaenko, V., Peikert, C., Raghunathan, A., Stebila, D., Easterbrook, K., Brian, L.: Frodokem: Practical quantum-secure key encapsulation from generic lattices (Apr 2022)
7. Alkim, E., Ducas, L., Pöppelmann, T., Schwabe, P.: Post-quantum key exchange - A new hope. In Holz, T., Savage, S., eds.: USENIX Security 2016: 25th USENIX Security Symposium, Austin, TX, USA, USENIX Association (August 10–12, 2016) 327–343
8. Bai, S., Stehlé, D., Wen, W.: Measuring, simulating and exploiting the head concavity phenomenon in BKZ. In Peyrin, T., Galbraith, S., eds.: Advances in Cryptology – ASIACRYPT 2018, Part I. Volume 11272 of Lecture Notes in Computer Science., Brisbane, Queensland, Australia, Springer, Heidelberg, Germany (December 2–6, 2018) 369–404
9. Bauer, A., Gilbert, H., Renault, G., Rossi, M.: Assessment of the key-reuse resilience of NewHope. In Matsui, M., ed.: Topics in Cryptology – CT-RSA 2019. Volume 11405 of Lecture Notes in Computer Science., San Francisco, CA, USA, Springer, Heidelberg, Germany (March 4–8, 2019) 272–292
10. Becker, A., Ducas, L., Gama, N., Laarhoven, T.: New directions in nearest neighbor searching with applications to lattice sieving. In Krauthgamer, R., ed.: 27th Annual ACM-SIAM Symposium on Discrete Algorithms, Arlington, VA, USA, ACM-SIAM (January 10–12, 2016) 10–24
11. Bindel, N., Schanck, J.M.: Decryption failure is more likely after success. In Ding, J., Tillich, J.P., eds.: Post-Quantum Cryptography - 11th International Conference, PQCrypto 2020, Paris, France, Springer, Heidelberg, Germany (April 15–17 2020) 206–225
12. Bland, R.G., Goldfarb, D., Todd, M.J.: The ellipsoid method: A survey. Operations research **29**(6) (1981) 1039–1091
13. Bos, J.W., Friedberger, S., Martinoli, M., Oswald, E., Stam, M.: Assessing the feasibility of single trace power analysis of Frodo. In Cid, C., Jacobson Jr:, M.J., eds.: SAC 2018: 25th Annual International Workshop on Selected Areas in Cryptography. Volume 11349 of Lecture Notes in Computer Science., Calgary, AB, Canada, Springer, Heidelberg, Germany (August 15–17, 2019) 216–234

14. Bruna, J., Regev, O., Song, M.J., Tang, Y.: Continuous LWE. In: STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021. (2021) 694–707
15. Chari, S., Rao, J.R., Rohatgi, P.: Template attacks. In Kaliski Jr., B.S., Koç, Çetin Kaya., Paar, C., eds.: Cryptographic Hardware and Embedded Systems – CHES 2002. Volume 2523 of Lecture Notes in Computer Science., Redwood Shores, CA, USA, Springer, Heidelberg, Germany (August 13–15, 2003) 13–28
16. Chen, Y., Nguyen, P.Q.: BKZ 2.0: Better lattice security estimates. In Lee, D.H., Wang, X., eds.: Advances in Cryptology – ASIACRYPT 2011. Volume 7073 of Lecture Notes in Computer Science., Seoul, South Korea, Springer, Heidelberg, Germany (December 4–8, 2011) 1–20
17. Dachman-Soled, D., Ducas, L., Gong, H., Rossi, M.: LWE with side information: Attacks and concrete security estimation. In Micciancio, D., Ristenpart, T., eds.: Advances in Cryptology – CRYPTO 2020, Part II. Volume 12171 of Lecture Notes in Computer Science., Santa Barbara, CA, USA, Springer, Heidelberg, Germany (August 17–21, 2020) 329–358
18. D'Anvers, J.P., Guo, Q., Johansson, T., Nilsson, A., Vercauteren, F., Verbauwhede, I.: Decryption failure attacks on IND-CCA secure lattice-based schemes. In Lin, D., Sako, K., eds.: PKC 2019: 22nd International Conference on Theory and Practice of Public Key Cryptography, Part II. Volume 11443 of Lecture Notes in Computer Science., Beijing, China, Springer, Heidelberg, Germany (April 14–17, 2019) 565–598
19. D'Anvers, J.P., Rossi, M., Virdia, F.: (One) failure is not an option: Bootstrapping the search for failures in lattice-based encryption schemes. In Canteaut, A., Ishai, Y., eds.: Advances in Cryptology – EUROCRYPT 2020, Part III. Volume 12107 of Lecture Notes in Computer Science., Zagreb, Croatia, Springer, Heidelberg, Germany (May 10–14, 2020) 3–33
20. D'Anvers, J.P., Vercauteren, F., Verbauwhede, I.: On the impact of decryption failures on the security of LWE/LWR based schemes. Cryptology ePrint Archive, Report 2018/1089 (2018) https://eprint.iacr.org/2018/1089.
21. Ding, J., Alsayigh, S., RV, S., Fluhrer, S., Lin, X.: Leakage of signal function with reused keys in RLWE key exchange. Cryptology ePrint Archive, Report 2016/1176 (2016) https://eprint.iacr.org/2016/1176.
22. Ding, J., Fluhrer, S.R., RV, S.: Complete attack on RLWE key exchange with reused keys, without signal leakage. In Susilo, W., Yang, G., eds.: ACISP 18: 23rd Australasian Conference on Information Security and Privacy. Volume 10946 of Lecture Notes in Computer Science., Wollongong, NSW, Australia, Springer, Heidelberg, Germany (July 11–13, 2018) 467–486
23. Fahr Jr, M., Kippen, H., Kwong, A., Dang, T., Lichtinger, J., Dachman-Soled, D., Genkin, D., Nelson, A., Perlner, R., Yerukhimovich, A., et al.: When frodo flips: End-to-end key recovery on frodokem via rowhammer. Cryptology ePrint Archive (2022)
24. Fluhrer, S.: Cryptanalysis of ring-LWE based key exchange with key share reuse. Cryptology ePrint Archive, Report 2016/085 (2016) https://eprint.iacr.org/2016/085.
25. Grötschel, M., Lovász, L., Schrijver, A. In: The Ellipsoid Method. Springer Berlin Heidelberg, Berlin, Heidelberg (1988) 64–101
26. Guo, Q., Johansson, T., Nilsson, A.: A generic attack on lattice-based schemes using decryption errors with application to ss-ntru-pke. Cryptology ePrint Archive, Report 2019/043 (2019) https://eprint.iacr.org/2019/043.

27. Gupte, A., Vafa, N., Vaikuntanathan, V.: Continuous LWE is as hard as LWE & applications to learning gaussian mixtures. Cryptology ePrint Archive, Report 2022/437 (2022) https://eprint.iacr.org/2022/437.
28. Herold, G., Kirshanova, E., Laarhoven, T.: Speed-ups and time-memory trade-offs for tuple lattice sieving. In Abdalla, M., Dahab, R., eds.: PKC 2018: 21st International Conference on Theory and Practice of Public Key Cryptography, Part I. Volume 10769 of Lecture Notes in Computer Science., Rio de Janeiro, Brazil, Springer, Heidelberg, Germany (March 25–29, 2018) 407–436
29. Islam, S., Mus, K., Singh, R., Schaumont, P., Sunar, B.: Signature correction attack on dilithium signature scheme (2022)
30. Jr., H.W.L.: Integer programming with a fixed number of variables. Math. Oper. Res. **8**(4) (1983) 538–548
31. Khachiyan, L.G.: A polynomial algorithm in linear programming. In: Doklady Akademii Nauk. Volume 244., Russian Academy of Sciences (1979) 1093–1096
32. Kirkwood, D., Lackey, B.C., McVey, J., Motley, M., Solinas, J.A., Tuller, D.: Failure is not an option: Standardization issues for post-quantum key agreement (2015) https://csrc.nist.gov/csrc/media/events/workshop-on-cybersecurity-in-a-post-quantum-world/documents/presentations/session7-motley-mark.pdf.
33. Kocher, P., Horn, J., Fogh, A., Genkin, D., Gruss, D., Haas, W., Hamburg, M., Lipp, M., Mangard, S., Prescher, T., Schwarz, M., Yarom, Y.: Spectre attacks: exploiting speculative execution. Commun. ACM **63**(7) (2020) 93–101
34. Kocher, P.C.: Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In Koblitz, N., ed.: Advances in Cryptology – CRYPTO'96. Volume 1109 of Lecture Notes in Computer Science., Santa Barbara, CA, USA, Springer, Heidelberg, Germany (August 18–22, 1996) 104–113
35. Kocher, P.C., Jaffe, J., Jun, B.: Differential power analysis. In Wiener, M.J., ed.: Advances in Cryptology – CRYPTO'99. Volume 1666 of Lecture Notes in Computer Science., Santa Barbara, CA, USA, Springer, Heidelberg, Germany (August 15–19, 1999) 388–397
36. Laarhoven, T.: Search problems in cryptography: from fingerprinting to lattice sieving. PhD thesis (2015)
37. Lipp, M., Schwarz, M., Gruss, D., Prescher, T., Haas, W., Horn, J., Mangard, S., Kocher, P., Genkin, D., Yarom, Y., Hamburg, M., Strackx, R.: Meltdown: reading kernel memory from user space. Commun. ACM **63**(6) (2020) 46–56
38. McCann, D., Oswald, E., Whitnall, C.: Towards practical tools for side channel aware software engineering: 'grey box' modelling for instruction leakages. In Kirda, E., Ristenpart, T., eds.: USENIX Security 2017: 26th USENIX Security Symposium, Vancouver, BC, Canada, USENIX Association (August 16–18, 2017) 199–216
39. Mus, K., Islam, S., Sunar, B.: QuantumHammer: A practical hybrid attack on the LUOV signature scheme. In Ligatti, J., Ou, X., Katz, J., Vigna, G., eds.: ACM CCS 2020: 27th Conference on Computer and Communications Security, Virtual Event, USA, ACM Press (November 9–13, 2020) 1071–1084
40. Qin, Y., Cheng, C., Zhang, X., Pan, Y., Hu, L., Ding, J.: A systematic approach and analysis of key mismatch attacks on lattice-based NIST candidate KEMs. Cryptology ePrint Archive, Report 2021/123 (2021) https://eprint.iacr.org/2021/123.
41. Ravi, P., Jhanwar, M.P., Howe, J., Chattopadhyay, A., Bhasin, S.: Side-channel assisted existential forgery attack on Dilithium - A NIST PQC candidate. Cryptology ePrint Archive, Report 2018/821 (2018) https://eprint.iacr.org/2018/821.

42. Ravi, P., Jhanwar, M.P., Howe, J., Chattopadhyay, A., Bhasin, S.: Exploiting determinism in lattice-based signatures: Practical fault attacks on pqm4 implementations of NIST candidates. In Galbraith, S.D., Russello, G., Susilo, W., Gollmann, D., Kirda, E., Liang, Z., eds.: ASIACCS 19: 14th ACM Symposium on Information, Computer and Communications Security, Auckland, New Zealand, ACM Press (July 9–12, 2019) 427–440
43. Ros, L., Sabater i Pruna, A., Thomas, F.: An ellipsoid calculus based on propagation and fusion. IEEE Transactions on Systems Man and Cybernetics Part B (Cybernetics) **32** (08 2002) 430–443
44. Schnorr, C., Euchner, M.: Lattice basis reduction: Improved practical algorithms and solving subset sum problems. Math. Program. **66** (1994) 181–199
45. Sepulveda, J., Zankl, A., Mischke, O.: Cache attacks and countermeasures for ntruencrypt on mpsocs: Post-quantum resistance for the iot. In: 2017 30th IEEE International System-on-Chip Conference (SOCC). (2017) 120–125
46. TSUNOO, Y.: Crypt-analysis of block ciphers implemented on computers with cache. Proc. ISITA2002, Oct. (2002)
47. Villanueva-Polanco, R.: Cold boot attacks on Bliss. In Schwabe, P., Thériault, N., eds.: Progress in Cryptology - LATINCRYPT 2019: 6th International Conference on Cryptology and Information Security in Latin America. Volume 11774 of Lecture Notes in Computer Science., Santiago, Chile, Springer, Heidelberg, Germany (October 2–4, 2019) 40–61
48. Wang, Z., Shen, X., Zhu, Y.: On equivalence of major relaxation methods for minimum ellipsoid covering intersection of ellipsoids. Automatica **103** (2019) 337–345

A Overview of the Ellipsoid Method

A linear program is an optimization problem of a linear objective function subject to linear equality and linear inequality constraints. The set of feasible solutions (if any exist) correspond to a region contained within a convex body $\mathcal{K}$. For any convex body $\mathcal{K}$, the optimal (i.e. minimum volume) circumscribing ellipsoid is known as the Löwner-John Ellipsoid. In general, these ellipsoids are hard to compute, but special cases, including intersections between ellipsoids and hyperplanes, halfspaces, or spaces between two parallel hyperplanes, have closed form expressions. If the solution of the linear program is initially known to be contained in some ellipsoid, the linear program's constraints can be used to obtain successively smaller volume ellipsoids by computing these Löwner-John ellipsoids in an iterative fashion. This procedure can then be used to determine feasibility: (1) In each iteration, check whether the center of the current ellipsoid satisfies the linear constraints. (2) If the center satisfies all constraints, then the center is a solution. (3) Otherwise, there is some constraint that is not satisfied by the center. Set the new ellipsoid to be the Löwner-John ellipsoid circumscribing the intersection of the current ellipsoid and the halfspace of the unsatisfied constraint. Continue to the next iteration. (4) If at some point the volume of the ellipsoid becomes sufficiently small, conclude that the linear program is infeasible. The fact that the ellipsoid method is polynomial time is implied by the fact that the volumes of the successive Löwner-John ellipsoids become sufficiently small in a polynomial number of steps.

B Proof of Theorem 4.1

We restate Theorem 4.1, followed by the proof.

Theorem 4.1. *Let $\boldsymbol{c} \in \mathbb{R}^d$ denote the d-dimensional vector that has $c \in \mathbb{R}$ in each position. Let $\sigma_1^2, \sigma_2^2 \in \mathbb{R}$ be such that $\sigma_2^2 < \sigma_1^2$. Consider the rank-scaled ellipsoids $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) = E^{(\mathsf{Rank})}(\mathbf{0}, \sigma_1^2 \boldsymbol{I}_d)$ and $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2) = E^{(\mathsf{Rank})}(\boldsymbol{c}, \sigma_2^2 \boldsymbol{I}_d)$. Then the volume of $E^{(\mathsf{Rank})}(\boldsymbol{\mu}', \boldsymbol{\Sigma}')$ is lower than both the volume of $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$ and $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ if and only if $c^2 > \sigma_1^2 - \sigma_2^2$.*

Proof of Theorem 4.1. Recall that $E(\boldsymbol{\mu}', \mathcal{F}(\boldsymbol{\Sigma}')) = E^{(\mathsf{Rank})}(\boldsymbol{\mu}', \boldsymbol{\Sigma}')$ is defined as follows:

$$\mathcal{F}(\boldsymbol{\Sigma}') = k\boldsymbol{X}^{-1},$$

$$\boldsymbol{X} = \lambda \mathcal{F}(\boldsymbol{\Sigma}_1)^{-1} + (1-\lambda)\mathcal{F}(\boldsymbol{\Sigma}_2)^{-1}$$

$$\boldsymbol{\mu}' = \frac{(1-\lambda)}{d\sigma_2^2}\boldsymbol{c}\boldsymbol{X}^{-1}$$

$$k = 1 - \lambda(1-\lambda)\frac{c^2}{\lambda\sigma_2^2 + (1-\lambda)\sigma_1^2},$$

for some $\lambda \in [0,1]$. The determinant of $\boldsymbol{\Sigma}'$ is

$$k^d \cdot (\frac{\sigma_1^2\sigma_2^2}{\lambda\sigma_2^2 + (1-\lambda)\sigma_1^2})^d \tag{28}$$

Thus, the volume of $E^{(\mathsf{Rank})}(\boldsymbol{\mu}', \boldsymbol{\Sigma}')$ decreases if and only if there is a setting of $\lambda \in [0,1]$ for which (28) is less than $(\sigma_2^2)^d$, which is the determinant of $\boldsymbol{\Sigma}_2$.

This constraint is satisfied if and only if

$$k \cdot \frac{\sigma_1^2\sigma_2^2}{\lambda\sigma_2^2 + (1-\lambda)\sigma_1^2} < \sigma_2^2.$$

Substituting k from above we get the requirement that:

$$\left(1 - \lambda(1-\lambda)\frac{c^2}{\lambda\sigma_2^2 + (1-\lambda)\sigma_1^2}\right) \cdot \frac{\sigma_1^2\sigma_2^2}{\lambda\sigma_2^2 + (1-\lambda)\sigma_1^2} < \sigma_2^2.$$

Which is true if and only if there exists a $\lambda \in [0,1]$ such that:

$$f(\lambda) = (\lambda\cdot(\sigma_1^2\sigma_2^2 - \sigma_1^4 - \sigma_1^2c^2) + \sigma_1^2\lambda^2c^2 + \sigma_1^4) < (\lambda^2\cdot(\sigma_2^2 - \sigma_1^2)^2 + 2\lambda\cdot(\sigma_1^2\sigma_2^2 - \sigma_1^4) + \sigma_1^4) = g(\lambda),$$

or equivalently, there exists a $\lambda \in [0,1]$ such that:

$$g(\lambda) - f(\lambda) = (g-f)(\lambda) > 0. \tag{29}$$

Note that when $\lambda = 0$, $(g-f)(\lambda) = 0$, and that when $\lambda = 1$, $(g-f)(\lambda) < 0$. Thus, since $(g-f)(\lambda)$ is a degree-2 function of λ, the condition from equation (29) is true if and only if the derivative of $(g-f)(\lambda)$ is positive at $\lambda = 0$.

We have that:

$$(g-f)'(0) = \sigma_1^2\sigma_2^2 - \sigma_1^4 + \sigma_1^2c^2.$$

Given the above, $(g-f)'(0) > 0$ if and only if $c^2 > \sigma_1^2 - \sigma_2^2$.

Thus, the volume of $E(\boldsymbol{\mu}', \mathcal{F}(\boldsymbol{\Sigma}'))$ will be smaller than the volume of $E(\boldsymbol{\mu}_2, \mathcal{F}(\boldsymbol{\Sigma}_2))$ if and only if $c^2 > \sigma_1^2 - \sigma_2^2$. This implies that the volume of $E^{(\mathsf{Rank})}(\boldsymbol{\mu}', \boldsymbol{\Sigma}')$ will be strictly less than the volume of $E^{(\mathsf{Rank})}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$ if and only if $c^2 > \sigma_1^2 - \sigma_2^2$. This concludes the proof of Theorem 4.1. □