Multitarget Decryption Failure Attacks and their Application to Saber and Kyber

Jan-Pieter D'Anvers and Senne Batsleer

imec-COSIC, KU Leuven
Kasteelpark Arenberg 10, Bus 2452, B-3001 Leuven-Heverlee, Belgium
janpieter.danvers@esat.kuleuven.be

Abstract. Many lattice-based encryption schemes are subject to a very small probability of decryption failures. It has been shown that an adversary can efficiently recover the secret key using a number of ciphertexts that cause such a decryption failure. In PKC 2019, D'Anvers et al. introduced 'failure boosting', a technique to speed up the search for decryption failures. In this work we first improve the state-of-the-art multitarget failure boosting attacks. We then improve the cost calculation of failure boosting and extend the applicability of these calculations to permit cost calculations of real-world schemes. Using our newly developed methodologies we determine the multitarget decryption failure attack cost for all parameter sets of Saber and Kyber, showing among others that the quantum security of Saber can theoretically be reduced from 172 bits to 145 bits in specific circumstances. We then discuss the applicability of decryption failure attacks in real-world scenarios, showing that an attack might not be practical to execute.

Keywords: Post-Quantum Cryptography, Lattice-based cryptography, Decryption failure attacks, Failure boosting

1 Introduction

Lattice-based cryptography is known for its versatility, bringing forth among others encryption schemes [23,4], digital signatures [26,24] and fully homomorphic encryption [16], identity based encryption [17] and attribute based encryption [29]. Moreover, lattice-based cryptographic schemes are among the most promising candidates for post-quantum cryptography, i.e. cryptography that is secure even in the presence of quantum computers.

In 2016, the United States National Institute of Standards and Technology (NIST) announced a standardization process with the goal of standardizing one or more post-quantum encryption and digital signature schemes [1]. July 2020 saw the start of the third round of this process, with 3 out of 4 finalists for public key encryption being lattice-based (and 2 out of the 5 alternate 'backup' schemes).

To improve efficiency, many lattice-based encryption schemes are not perfectly correct, which means that even after a correct execution of the protocol, it is possible that the decryption fails to retrieve the correct message or key. Such an event is called a decryption failure, and the ciphertext that caused the failure is

referred to as a failing ciphertext. Three of the lattice-based NIST candidates are subject to such decryption failures: Saber [10], Kyber [8] and FrodoKEM [25].

While the probabilities of these decryption failures are chosen sufficiently small to avoid any impact on performance, they have been used to stage attacks against these schemes. Decryption failure attacks can be roughly divided into two categories: chosen-ciphertext attacks and valid-ciphertext attacks. The first type was introduced by Jaulmes and Joux [21] and can efficiently recover the secret key if it is reused, by crafting specific ciphertexts that fail based on properties of the secret key. However, this attack type can be prevented by using a chosen-ciphertext transformation such as the Fujisaki-Okamoto transformation.

The second type of decryption failure attacks remains a threat even in the presence of chosen-ciphertext security measures. The idea behind this type of attack is to input a large number of correctly generated ciphertexts in search for failing ciphertexts. The authors of Kyber [8] noted that it is possible to do a Grover search for ciphertexts with higher than average failure probability. D'Anvers et al. [12] showed how to retrieve the secret key based on correctly generated but failing ciphertexts, and introduced 'failure boosting', a framework to speed up the search for failing ciphertexts. This was later extended in [11] to 'directional failure boosting', which introduced a method that further speeds up the failing ciphertext search when one or more failing ciphertext have already been found. The latter work studied a simplified lattice-based scheme and focussed on attacking a single target showing that the cost of a decryption failure attack is dominated by the cost of finding the first failure. Moreover, they introduced a simple multitarget attack specifically designed for scenarios where a maximum number of decapsulations can be performed per target. Around the same time, Guo et al. proposed specific decryption failure attacks on ss-ntru-pke [18] and LAC [19].

As opposed to attacks focusing on decryption failures, Bindel and Schanck [7] showed that correctly generated ciphertexts also provide a small amount of information about the secret. While the errors in individual message bits were assumed to happen independently in many NIST submission documents, D'Anvers et al. [13] showed that these errors are in fact correlated, showing an underestimation of the decryption failure probability for schemes that use error correction and thus an overestimation of the security of these schemes. Dachman-Soled et al. [9] developed a tool to include 'hints' into a LWE hard problem and showed that it can be used to retrieve the secret key using failing ciphertexts.

Our contributions: We first improve the state-of-the-art multitarget decryption failure attack using a levelled approach in Section 4, leading to a more efficient attack especially for schemes with low failure probability. Secondly, we enhance the techniques to estimate the cost of decryption failure attacks, and extend them to include practical schemes such as Saber and Kyber: Section 5 points out three inaccuracies in the directional failure boosting calculation for the simplified scheme of [11], which are discussed and remedied. Section 6 shows that this traditional approach of calculating the directional failure boosting cost is not directly applicable to practical schemes such as Kyber and Saber due to compression of the ciphertexts and introduces new methods that adapt the traditional directional failure boosting

approach to these real-world schemes. Thirdly, Section 7 introduces two additional constraints an attacker might face when mounting a decryption failure attack, which have not been taken into account in previous failure boosting attacks. As a result, in Section 8 we discuss the impact of decryption failure attacks on Kyber and Saber.

2 Preliminaries

2.1 Notation

Denote with $\mathbb{Z}_q$ the ring of integers modulo q, represented in $(-q/2,q/2]$. Let R_q be the ring $\mathbb{Z}_q[X]/(X^N+1)$, with N a power of two, and let $R^{l_1 \times l_2}$ be the ring of $l_1 \times l_2$ matrices over R_q. We denote matrices with bold upper case (e.g. $\mathbf{A}$) and vectors and polynomials with bold lower case (e.g. $\mathbf{b}$).

Denote with $\lfloor \cdot \rfloor$ flooring to the nearest lower integer, with $\lfloor \cdot \rceil$ rounding to the nearest integer where ties are rounded upwards, and with $\lfloor \cdot \rceil_{q \to p}$ dividing by p/q followed by rounding, i.e. $\lfloor x \rceil_{q \to p} = \lfloor p/q \cdot x \rceil$. Let $|\cdot|$ denote taking the absolute value. These notations are extended to vectors, matrices and polynomials element wise. The l_2 norm of a polynomial or vector of integers $\mathbf{x}$ is defined as $||\mathbf{x}||_2 = \sqrt{\sum_i \mathbf{x}_i^2}$ and for a vector of polynomials $\mathbf{y}$ as $||\mathbf{y}||_2 = \sqrt{\sum_i ||\mathbf{y}_i||_2^2}$.

Let $x \leftarrow \chi$ mean sampling x according to a probability distribution χ, and let $\mathbf{X} \leftarrow \chi(R^{l_1 \times l_2})$ denote sampling $\mathbf{X} \in R^{l_1 \times l_2}$ with polynomial coefficients according to the distribution χ. When the values are sampled pseudorandomly based on a seed r, this is denoted as $\mathbf{X} \leftarrow \chi(R^{l_1 \times l_2}; r)$. The uniform distribution is denoted $\mathcal{U}$.

We write $P[E]$ to denote the probability of an event E. To simplify notation we denote with $P[a]$ the probability of sampling an element a from a certain distribution χ when this distribution is clear from the context, i.e. $P[x = a \mid x \leftarrow \chi]$. Analogous, we denote with $\mathbb{E}[a]$ the expected value of an element a as sampled from its distribution χ when this distribution is clear from the context.

2.2 Cryptographic definitions

We define a Public Key Encryption scheme (PKE) as a triplet of functions (`KeyGen`,`Encrypt`,`Decrypt`), where the key generation `KeyGen` generates a public key pk and secret key sk, where the encryption `Encrypt` take a public key pk and a message m from the message space $\mathcal{M}$ to generate a ciphertext ct, and where the decryption `Decrypt` retrieves the message m with high probability from the ciphertext ct using the secret key sk. A PKE is δ-correct if:

$$\mathbb{E}[P[\texttt{Decrypt}(\texttt{Encrypt}(m,pk),sk) \neq m]] \leq \delta.$$

Similarly, we define a Key Encapsulation Mechanism (KEM) as the functions (`KeyGen`,`Encaps`,`Decaps`), where `KeyGen` generates a public key pk and secret key sk, where `Encaps` generates a key k from keyspace $\mathcal{K}$ and a ciphertext ct given a public key pk, and where `Decaps` outputs a key k' or $\perp$ when given a ciphertext ct and corresponding secret key sk. We say that a KEM is δ-correct if:

$$\mathbb{E}[P[\texttt{Decaps}(ct,sk) \neq k : (ct,k) \leftarrow \texttt{Encaps}(pk)]] \leq \delta.$$

The Module Learning with Errors (Mod-LWE) is a hard mathematical problem introduced by Langlois and Stehlé [22], as a generalization of the Learning with Errors (LWE) [26] and Ring Learning with Errors (Ring-LWE) [24] problems. Given integers N, q and l, the ring $R_q = \mathbb{Z}[X]/(X^N+1)$, a distribution with limited variance χ and a secret element $\mathbf{s} \in R_q^l$, samples from the Mod-LWE distribution $\mathcal{L}_{R,N,q,l,\chi,\mathbf{s}}$ are generated as:

$$(\mathbf{a}, \mathbf{b} := \mathbf{a}^T\mathbf{s} + \mathbf{e}) \tag{1}$$

$$\text{where: } \mathbf{a} \leftarrow \mathcal{U}(R_q^l); \mathbf{e} \leftarrow \chi(R_q) \tag{2}$$

We will specifically focus on the case where N is a power of two. The decision Mod-LWE problem is then, given k samples, to determine whether they were generated as Mod-LWE samples from $\mathcal{L}_{R,N,q,l,\chi,\mathbf{s}}$ or from the uniform distribution $\mathcal{U}(R_q^l \times R_q)$. The search Mod-LWE problem consists of recovering the secret $\mathbf{s}$ from k Mod-LWE samples. LWE is a specific instance where $R_q = \mathbb{Z}_q$ and Ring-LWE the specific instance where $l = 1$.

Learning with Rounding (LWR), as introduced by Banerjee et al. [5], is a similar problem where the error $\mathbf{e}$ is replaced with a deterministic error obtained by rounding. Analogous to the LWE problem, variants of LWR include Ring-LWR and Mod-LWR. Given two moduli q and p, where $q > p$, sampling from the Mod-LWR distribution can be described as:

$$(\mathbf{a}, \mathbf{b} := \lfloor \mathbf{a}^T\mathbf{s} \rceil_{q \to p}) \tag{3}$$

$$\text{where: } \mathbf{a} \leftarrow \mathcal{U}(R_q^l) \tag{4}$$

In this paper we will specifically consider the case where $p|q$. The Mod-LWR decisional and search problem are defined similar to their respective Mod-LWE versions, where in the decisional problem an adversary has to distinguish between sampling from a Mod-LWR or uniform distribution, and where in the search problem an adversary is tasked to retrieve the secret $\mathbf{s}$ from k Mod-LWR samples.

2.3 Lattice-based Encryption

A generic PKE based on the Mod-LWE or Mod-LWR assumption is given in Algorithm 1 to 3, where q, p_1, p_2 and t are scheme dependent integers, where χ_s and χ_e are scheme specific probability distributions with small variance, where $r \in \mathcal{R} = \{0,1\}^{256}$ and where the message space $\mathcal{M}$ consists of polynomials in R_q with coefficients $\{0,1\}$.

This generic protocol can be used to describe Saber, Kyber and the scheme studied in [11], which was designed to simplify the study of failure boosting and will be referred to as Katana. The parameters of these schemes are given in Table 1. For Saber and Kyber we consider the round 3 submissions as described in [6] and [27] respectively, which are the most recent versions at the time of writing.

For Kyber, the distributions χ_s and χ_e are centered binomial distributions with limited variance. There is no public key compression (i.e. $q = p_1$) but there is ciphertext compression (i.e. $q > p_2 > t$). Saber[1] similarly uses a centered binomial

[1] Saber has slightly different rounding methods, but this does not impact our study as the failure condition remains the same.

Algorithm 1: PKE.KeyGen()

$\mathbf{A} \leftarrow \mathcal{U}(R_q^{l \times l})$
$\mathbf{s},\mathbf{e} \leftarrow \chi_s(R_q^{l \times 1}) \times \chi_e(R_q^{l \times 1})$
$\mathbf{b} := \lfloor \mathbf{As}+\mathbf{e} \rceil_{q \to p_1}$
return $(pk=(\mathbf{b},\mathbf{A}), sk=\mathbf{s})$

Algorithm 2: PKE.Enc($pk=(\mathbf{b},\mathbf{A})$, $\mathbf{m} \in \mathcal{M}; r$)

$\mathbf{s}',\mathbf{e}' \leftarrow \chi_s(R_q^{l \times 1}; r) \times \chi_e(R_q^{l \times 1}; r)$
$\mathbf{e}'' \leftarrow \chi_e(R_q; r)$
$\mathbf{b}' := \lfloor \mathbf{A}^T\mathbf{s}'+\mathbf{e}' \rceil_{q \to p_2}$
$\mathbf{b}_q := \lfloor \mathbf{b} \rceil_{p_1 \to q}$
$\mathbf{v}' := \lfloor \mathbf{b}_q^T\mathbf{s}'+\mathbf{e}''+\lfloor q/2 \rfloor \cdot \mathbf{m} \rceil_{q \to t}$
return $ct=(\mathbf{v}',\mathbf{b}')$

Algorithm 3: PKE.Dec($sk=\mathbf{s}, ct=(\mathbf{v}',\mathbf{b}')$)

$\mathbf{b}'_q := \lfloor \mathbf{b}' \rceil_{p_2 \to q}$
$\mathbf{v}'_q := \lfloor \mathbf{v}' \rceil_{t \to q}$
$\mathbf{m}' := \lfloor \lfloor 2/q \rfloor (\mathbf{v}'_q - \mathbf{b}'^T_q \mathbf{s}) \rceil$
return $\mathbf{m}'$

	l	N	q	$\sigma(\mathbf{s}_i)$	$\sigma(\mathbf{e}_i+\mathbf{u}_i)$	$P[F]$	Classical	Quantum
Katana [11]	3	256	8192	2.00	2.00	2^{-119}	2^{195}	2^{177}
Saber [6]	3	256	8192	1.41	2.29	2^{-136}	2^{189}	2^{172}
Kyber768 [27]	3	256	3329	1.00	1.00/1.38†	2^{-164}	2^{181}	2^{164}

† Standard deviation of the error term in the public key and ciphertext respectively

Table 1. Parameters of Katana, Saber and Kyber. The security is based on the estimates of Albrecht et al. [3,2]

distribution for χ_s, but its distribution χ_e always returns zero. Saber does both public key and ciphertext compression (e.g. $q > p_1 = p_2 > t$). Katana is an idealized scheme with Gaussian distributions for χ_s and χ_e and without compression of the public key or ciphertext (i.e. $q = p_1 = p_2 = t$).

2.4 Chosen-ciphertext security

To protect against chosen-ciphertext attacks, designers typically convert their passively secure PKE to an actively secure KEM using a generic transformation such as a post-quantum variant [28,20] of the Fujisaki-Okamoto [15,14] transformation. The obtained KEM then has a similar key generation, while the encapsulation and decapsulation are constructed as described in Algorithms 4 and 5 respectively. The idea behind this transformation is that the input ciphertext is checked using a re-encryption of the message, and the ciphertext is rejected if the input ciphertext is not valid. As a result of this procedure, an adversary does not learn anything from inputting invalid ciphertexts. However, in case of a valid ciphertext that leads to a decryption failure, the re-encryption still fails and we will assume that an attacker is able to recognize such event.

Algorithm 4: $\texttt{KEM.Encaps}(pk)$

$m \leftarrow \mathcal{U}(\{0,1\}^{256})$
$(\overline{K},r) := \mathcal{G}(pk,m)$
$ct := \texttt{PKE.Enc}(pk,m,r)$
$K := \mathcal{H}(\overline{K},r)$
return (ct,K)

Algorithm 5: $\texttt{KEM.Decaps}(sk,pk,ct,K)$

$m' := \texttt{PKE.Dec}(sk,ct)$
$(\overline{K},r') := \mathcal{G}(pk,m')$
$ct' := \texttt{PKE.Enc}(pk,m';r')$
if $ct = ct'$ **then**
  return $K := \mathcal{H}(\overline{K},r')$
else
  return $K := \perp$

2.5 Decryption failures

A decryption failure is an event where one fails to recover message or key, which can even happen after following the correct protocol. The occurrence of decryption failures depends on the secret terms $\mathbf{s},\mathbf{s}',\mathbf{e},\mathbf{e}',\mathbf{e}''$ in combination with the rounding errors $\mathbf{u},\mathbf{u}',\mathbf{u}''$, which are defined as:

$$\mathbf{u} := \mathbf{b}_q - (\mathbf{A}\mathbf{s}+\mathbf{e}) \tag{5}$$
$$\mathbf{u}' := \mathbf{b}'_q - (\mathbf{A}^T\mathbf{s}'+\mathbf{e}') \tag{6}$$
$$\mathbf{u}'' := \mathbf{v}'_q - (\mathbf{b}_q^T\mathbf{s}'+\mathbf{e}''+\mathbf{m}) \tag{7}$$

Expanding the value of the received message m', we get:

$$\mathbf{m}' = \lfloor \lfloor 2/q \rfloor (\mathbf{v}'_q - \mathbf{b}_q'^T\mathbf{s}) \rceil \tag{8}$$
$$= \mathbf{m} + \lfloor \lfloor 2/q \rfloor ((\mathbf{e}+\mathbf{u})^T\mathbf{s}' - \mathbf{s}^T(\mathbf{e}'+\mathbf{u}') + (\mathbf{e}''+\mathbf{u}'')) \rceil \tag{9}$$

and a decryption failure occurs if any coefficient of this error term exceeds the threshold $q_t = q/4$, which can be formalized as follows:

$$||(\mathbf{e}+\mathbf{u})^T\mathbf{s}' - \mathbf{s}^T(\mathbf{e}'+\mathbf{u}') + (\mathbf{e}''+\mathbf{u}'')||_\infty > q_t$$

Failure vectors: Following [12] we define the failure vectors $\mathcal{S}, \mathcal{C}, \mathcal{G}$ as:

$$\mathcal{S} = \begin{pmatrix} -\mathbf{s} \\ \mathbf{e}+\mathbf{u} \end{pmatrix} \quad \mathcal{C} = \begin{pmatrix} \mathbf{e}'+\mathbf{u}' \\ \mathbf{s}' \end{pmatrix} \quad \mathcal{G} = \mathbf{e}''+\mathbf{u}'' \tag{10}$$

which simplifies the failure condition to:

$$||\mathcal{S}^T\mathcal{C} + \mathcal{G}||_\infty > q_t$$

Geometric notation: To streamline notation, we will use the geometric notation as introduced in [11]. The vector $\overline{\mathcal{S}} \in \mathbb{Z}_q^{lN\times 1}$ is an integer vector representation of $\mathcal{S}$, obtained by arranging all coefficients of the polynomials of $\mathcal{S}$ in a vector. Additionally, the rotation of a vector of polynomials $\mathcal{C}$ is defined as:

$$\mathcal{C}^{(r)} := X^r \cdot \mathcal{C}(X^{-1}) \mod X^N + 1. \tag{11}$$

Using this notation, the i^{th} coefficient of $\mathcal{S}^T\mathcal{C}$ can be calculated as $\overline{\mathcal{S}}^T\overline{\mathcal{C}^{(i)}}$. An illustration of these concepts is given in Example 1. For more information about this representation we refer to [11].

Example 1. [11] For a secret $\mathcal{S}$ and a ciphertext $\mathcal{C}$ in $\mathbb{Z}_q^{2\times 1}[X]/(X^3+1)$:

$$\mathcal{S} = \begin{bmatrix} \mathcal{S}_{0,0}+\mathcal{S}_{0,1}X+\mathcal{S}_{0,2}X^2 \\ \mathcal{S}_{1,0}+\mathcal{S}_{1,1}X+\mathcal{S}_{1,2}X^2 \end{bmatrix}, \quad \mathcal{C} = \begin{bmatrix} \mathcal{C}_{0,0}+\mathcal{C}_{0,1}X+\mathcal{C}_{0,2}X^2 \\ \mathcal{C}_{1,0}+\mathcal{C}_{1,1}X+\mathcal{C}_{1,2}X^2 \end{bmatrix}$$

we get the following vectors:

$$\overline{\mathcal{S}} = \begin{bmatrix} \mathcal{S}_{0,0} \\ \mathcal{S}_{0,1} \\ \mathcal{S}_{0,2} \\ \mathcal{S}_{1,0} \\ \mathcal{S}_{1,1} \\ \mathcal{S}_{1,2} \end{bmatrix}, \quad \overline{\mathcal{C}^{(0)}} = \begin{bmatrix} \mathcal{C}_{0,0} \\ -\mathcal{C}_{0,2} \\ -\mathcal{C}_{0,1} \\ \mathcal{C}_{1,0} \\ -\mathcal{C}_{1,2} \\ -\mathcal{C}_{1,1} \end{bmatrix} \quad \overline{\mathcal{C}^{(1)}} = \begin{bmatrix} \mathcal{C}_{0,1} \\ \mathcal{C}_{0,0} \\ -\mathcal{C}_{0,2} \\ \mathcal{C}_{1,1} \\ \mathcal{C}_{1,0} \\ -\mathcal{C}_{1,2} \end{bmatrix} \quad \overline{\mathcal{C}^{(3)}} = \begin{bmatrix} -\mathcal{C}_{0,0} \\ \mathcal{C}_{0,2} \\ \mathcal{C}_{0,1} \\ -\mathcal{C}_{1,0} \\ \mathcal{C}_{1,2} \\ \mathcal{C}_{1,1} \end{bmatrix} ...$$

Definitions: We will denote with F a decryption failure, and with S a successful decryption. F_i will denote an error at the i^{th} coefficient of $\mathcal{S}^T\mathcal{C}+\mathcal{G}$, which happens when the absolute value of this coefficient is bigger than q_t. Similarly S_i will denote a successful decryption of the i^{th} coefficient. Using the geometric notation we can say that an error F_i occurs if:

$$\left|\overline{\mathcal{S}}^T\overline{\mathcal{C}^{(i)}}+\mathcal{G}_i\right| > q_t$$

We will use the shorthand $P_F[ct]$ to denote the failure probability $P[F|ct]$ for a certain ciphertext ct, which can be formalized as:

$$P_F[ct] = \sum_{\forall \mathcal{S}} P[\mathcal{S}] \cdot P[F|ct,\mathcal{S}]$$

Sometimes, we will group ciphertexts in classes, where a class cl bundles all ciphertexts with certain properties, e.g. $cl = \{\forall ct : ||\mathcal{C}||_2 = c, \mathcal{G} = g\}$. In this case $P_F[cl]$ denotes the weighted average of the failure probabilities of all ciphertexts in the class cl, which can be formalized as:

$$P_F[cl] = \sum_{\forall ct : ct \in cl} P[ct] \cdot P[F|ct]$$

3 Failure boosting attacks

By exploiting decryption failures, an attacker can mount an attack that retrieves the secret key. The crux of such an attack is that failing ciphertexts give information that can be used to reconstruct the secret key as described in [12], [11] and [9]. In this paper we will focus on the process to obtain these failing ciphertexts as efficiently as possible.

We specifically target schemes that are IND-CCA secured, which implies that non-valid ciphertexts are rejected by the decapsulation regardless of the occurrence of a decryption failure and thus that they can not give any information. As such

the attack surface is limited to submitting valid ciphertexts and observing whether a failure occurs.

Failure boosting [12] is a technique to increase the failure probability of valid ciphertexts submitted for decapsulation. It is a two step process consisting of a precomputation step and a query step. We will discuss the cost of a failure boosting attack using two metrics: work $\mathcal{W}$ and queries $\mathcal{Q}$. Work describes the cost of precomputation, where $1\mathcal{W}$ is defined as the cost of generating one ciphertext, while $\mathcal{Q}$ describes the total number of decapsulation queries performed.

Precomputation: During precomputation, the adversary performs an offline search for weak ciphertexts, i.e. valid ciphertexts with a high failure probability. This is accomplished by randomly generating ciphertexts until a ciphertext with failure probability above a certain threshold f_t is found. The probability of finding such a ciphertext can be expressed as follows:

$$\alpha(f_t) = \sum_{\forall ct : P_F[ct] > f_t} P[ct]. \tag{12}$$

Finding a weak failure will take on average $\alpha(f_t)^{-1}$ work, but this can be sped up quadratically using a quantum computer to $\sqrt{\alpha(f_t)^{-1}}$ work.

Querying: Once a weak ciphertext is found, it is submitted for decapsulation and the adversary observes whether it triggers a decryption failure. A failure happens with probability $\beta(f_t)$ for a given threshold f_t, which can be calculated as follows:

$$\beta(f_t) = \frac{\sum_{\forall ct : P_F[ct] > f_t} P[ct] \cdot P_F[ct]}{\sum_{\forall ct : P_F[ct] > f_t} P[ct]} = \frac{\sum_{\forall ct : P_F[ct] > f_t} P[ct] \cdot P_F[ct]}{\alpha(f_t)}. \tag{13}$$

The query step can not be sped up using quantum computers as an adversary has typically no quantum access to the decapsulation oracle. An adversary needs on average $\beta(f_t)^{-1}$ queries to obtain one decryption failure.

Attack cost: For a given threshold f_t, finding a decryption failure costs on average $\alpha(f_t)^{-1}\beta(f_t)^{-1}$ work and $\beta(f_t)^{-1}$ queries, which can be reduced to $\sqrt{\alpha(f_t)^{-1}}\beta(f_t)^{-1}$ work when Grover search is used during precomputation.

3.1 Directional failure boosting

Directional failure boosting [11] improves failure boosting and can be used when at least one other failure has been found. It specifically uses information of previously found failing ciphertexts to improve the search for new failures. In [11], this is done by calculating $\mathcal{E}$, an estimate of the direction of the secret $\overline{\mathcal{S}}$, and taking this into account in the failure estimation $P_F[ct,\mathcal{E}]$.

Directional failure boosting dramatically reduces the cost of finding additional failures after the first failure has been found. As a result, in a single target attack the work and number of queries is dominated by finding the first failure and thus the cost of a single target attack can be approximated as the cost of finding the first failure. An in depth discussion of directional failure boosting can be found in [11].

3.2 Estimation of efficiency

The cost of (directional) failure boosting is described by Equation 12 and 13, which requires to sum over all possible ciphertexts. This is clearly infeasible, but can be simplified by making an approximate failure model and grouping ciphertexts with similar failure probability. Two such models were presented in the literature: Gaussian approximation and geometric approximation.

Gaussian approximation [12]: The Gaussian approximation considers the coefficients of $\mathcal{S}^T\mathcal{C}$ to follow a Gaussian distribution with zero mean and variance depending on $\mathcal{C}$. This assumption can be used to accurately estimate failure boosting efficiency, but does not work for directional failure boosting estimations. The calculation method as presented in [12] takes both $\mathcal{C}$ and $\mathcal{G}$ into account in the weak ciphertext selection. For more information about the exact calculation methodology we refer the reader to [12].

Geometric approximation [11]: The geometric approximation assumes that the angle ϕ between $\overline{\mathcal{S}}^T\overline{\mathcal{C}^{(i)}}$ behaves as a uniformly random angle in dimension $2Nl$. This approximation corresponds to the assumption that χ_s and χ_e are continuous Gaussian distributions with zero mean. Using the geometric approximation, the condition on an error at the i^{th} coefficient can be rewritten from:

$$\left|\overline{\mathcal{S}}^T\overline{\mathcal{C}^{(i)}}+\mathcal{G}_i\right| > q_t \tag{14}$$

to:

$$|\ ||\mathcal{S}||_2\cdot||\mathcal{C}||_2\cdot\cos(\phi)+\mathcal{G}_i| > q_t \tag{15}$$

In directional failure boosting, the vectors $\overline{\mathcal{S}}$ and $\overline{\mathcal{C}^{(i)}}$ are first expanded in a part parallel and a part orthogonal to the estimate of the secret $\mathcal{E}$:

$$\left|\overline{\mathcal{S}}_\perp^T\overline{\mathcal{C}^{(i)}}_\perp+\overline{\mathcal{S}}_\parallel^T\overline{\mathcal{C}^{(i)}}_\parallel+\mathcal{G}_i\right| > q_t \tag{16}$$

which can be further expanded to:

$$\left|\begin{array}{l}||\mathcal{S}||_2\cdot||\mathcal{C}||_2\cdot\cos(\theta_{SE})\cdot\cos(\theta_{C^iE})+\\ ||\mathcal{S}||_2\cdot||\mathcal{C}||_2\cdot\sin(\theta_{SE})\cdot\sin(\theta_{C^iE})\cdot\cos(\psi)+\mathcal{G}_i\end{array}\right| > q_t \tag{17}$$

with ψ a uniformly random angle in dimension $2Nl-1$. In D'Anvers et al. [11], the $\mathcal{G}$ term was neglected in the calculations. A more detailed explanation of this technique can be found in [11].

Attack cost estimation: Using the above approximations, one can bundle ciphertexts with similar failure probability in classes cl to reduce the cost of calculating $\alpha(f_t)$ and $\beta(f_t)$. The values of $\alpha(f_t)$ and $\beta(f_t)$ can be calculated using the formulas below, with the difference that $P[cl]$ is the probability of a randomly generated

ciphertext belongs to the specific class cl, and $P_F[cl]$ the failure probability of ciphertexts in that class.

$$\alpha(f_t) \approx \sum_{\forall cl: P_F[cl] > f_t} P[cl] \tag{18}$$

$$\beta(f_t) \approx \frac{\sum_{\forall cl: P_F[cl] > f_t} P[cl] \cdot P_F[cl]}{\alpha(f_t)} \tag{19}$$

For example, under the geometric approximation, one bundles all ciphertexts with similar $||\mathcal{C}||_2$ for failure boosting. Directional failure boosting in the geometric approximation defines classes based on $||\mathcal{C}||_2$ and the closest angle $\text{maxcos}_\text{i}(\theta_{\text{C}^{(\text{i})}\text{E}})$ between the rotations of the ciphertext and the estimate of the secret $\mathcal{E}$.

4 Multitarget attacks

One of the main constraints in a practical attack is the number of queries that can be performed. For example, NIST [1] set a maximum of $q_{limit} = 2^{64}$ decapsulation queries per target that can be performed during an attack. One possibility to circumvent such limitation is to consider multiple targets, with the goal of breaking one of these targets.

Such a multitarget attack queries a certain number of targets $T^{(0)}$, where each target has an individual query limit. The goal is to retrieve the secret key for at least one of these targets. We assume that multitarget protection is in place, so that ciphertexts are only valid for one given public key and thus target. Such multitarget protection is easily obtained by incorporating (a hash of) the public key in the ciphertext generation, which is the case for Saber and Kyber.

4.1 Naive multitarget

A naive variant of the multitarget attack was introduced in [11], which proceeded as follows: First, find the first failure by performing at most $q_{limit}/2$ per target, which in total implies a maximum of $T^{(0)} \cdot q_{limit}/2$ queries. Then, focus on the target that caused the failure and continue with a single target attack on this target with query limit $q_{limit}/2$.

First note that due to multitarget protection, each generated weak ciphertext is linked to a specific public key and can only be used for that target. Moreover, one can assume that given only the public key the adversary has no efficient way to retrieve information about the secret key $\mathcal{S}$ without solving the Mod-LWE/LWR problem. This implies that he has no efficient way to distinguish between targets with higher or lower failure probability and thus that generating a weak ciphertext and querying it has exactly the same failure probability at each target.

Assuming that successful queries do not contribute any information about the targets, the failure probability at each target stays the same until a decryption failure has been found. Therefore, we can say that finding one failure at $T^{(0)}$ targets

with a maximum of $q_{limit}/2$ queries per target has the same cost as finding one failure at one target with a maximum of $T^{(0)} \cdot q_{limit}/2$ queries, so that the cost of finding the first failure in the naive multitarget attack can be described with:

$$\sqrt{\alpha_0^{-1}}\beta_0^{-1} \text{ work, and } \beta_0^{-1} \text{ queries,} \tag{20}$$

under the condition that:

$$\beta_0^{-1} < T^{(0)} \cdot q_{limit}/2, \tag{21}$$

where α_i and β_i denote the optimal values for $\alpha(f_t)$ and $\beta(f_t)$ for the i^{th} failure, which can be determined by selecting the value of f_t that optimally reduces the work while fulfilling the query limit constraint.

To estimate the cost of finding the follow-up failures, we can use the approximation from [11], which states that in a single target attack the attack cost is dominated by finding the first failure. In this case, the first failure of the single target attack is the second overall failure so that the cost of finding the follow up failures can be calculated as:

$$\sqrt{\alpha_1^{-1}}\beta_1^{-1} \text{ work, and } \beta_1^{-1} \text{ queries,} \tag{22}$$

under the condition that:

$$\beta_1^{-1} < q_{limit}/2, \tag{23}$$

One can easily see that the total number of queries per target is always under q_{limit} in this scenario.

4.2 Levelled multitarget

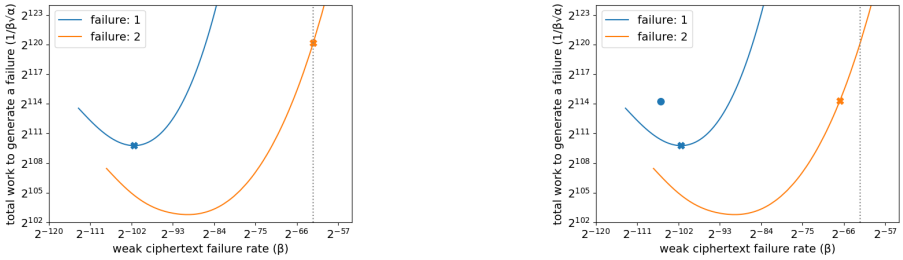
When the cost of finding the second failure is the dominant factor, this naive multitarget attack can be improved using a levelled approach. Notice that the naive multitarget attack essentially reduces the cost of the attack by relaxing the query limit constraint for finding the first failure. To reduce the cost of finding the second failure, we can similarly focus on multiple targets to relax the query constraint. However, this requires the attacker to find multiple failing ciphertexts in the first step of the attack.

More specifically, in the first phase, the attacker aims at obtaining $T^{(1)}$ targets using under $q_{limit}/3$ queries per target (which is a total of $T^{(0)} \cdot q_{limit}/3$ queries). This has a cost of:

$$T^{(1)}\sqrt{\alpha_0^{-1}}\beta_0^{-1} \text{ work and } T^{(1)}\beta_0^{-1} \text{ queries.} \tag{24}$$

Under the condition that:

$$T^{(1)}\beta_0^{-1} < T^{(0)} \cdot q_{limit}/3. \tag{25}$$



(a) Naive multitarget[11]

(b) Levelled multitarget

Fig. 1. Example of multitarget attacks on Katana, with 2^{64} targets and maximum 2^{64} queries. The cost of finding one failure is indicated with x. The cost of finding $T^{(1)}$ failures using failure boosting in the first phase is given by the blue dot, and the corresponding number of queries can be found as β^{-1} where β is the x-axis value of this point. In the naive multitarget attack the cost is dominated by finding the second failure in under 2^{64} queries. In the levelled approach the cost of the two phases is equalized.

	Naive multitarget [11]		Levelled multitarget [ours]	
	work	queries	work	queries
first failure	$\sqrt{\alpha_0^{-1}}\beta_0^{-1}$	$T^{(0)}\cdot q_{limit}/2$	$T^{(1)}\sqrt{\alpha_0^{-1}}\beta_0^{-1}$	$T^{(0)}\cdot q_{limit}/(3T^{(1)})^{\dagger}$
second failure	$\sqrt{\alpha_1^{-1}}\beta_1^{-1}$	$q_{limit}/2$	$\sqrt{\alpha_1^{-1}}\beta_1^{-1}$	$T^{(1)}\cdot q_{limit}/3$
follow up failures	negligible	-	$\sqrt{\alpha_2^{-1}}\beta_2^{-1}$	$q_{limit}/3$

† per failure, total query limit is $T^{(0)}\cdot q_{limit}$

Table 2. Comparison of the naive and levelled multitarget attack. Note that α and β values are not the same between both methods as the difference in query limits leads to a different optimal f_t.

The attacker can then use $T^{(1)}\cdot q_{limit}/3$ queries to find the next failure, which has a cost of:

$$\sqrt{\alpha_1^{-1}}\beta_1^{-1} \text{ work and } \beta_1^{-1} \text{ queries.} \tag{26}$$

Under the condition that:

$$\beta_1^{-1} < T^{(1)}\cdot q_{limit}/3. \tag{27}$$

Once a second failure is found for a given target, the attack continues with a single target attack on that target using at most $q_{limit}/3$ queries. An overview of this levelled multitarget approach is given in Table 2. Note that the query limit per phase is chosen so that the total number of queries at each target over all failures is always under q_{limit}. Figure 1 gives a graphical comparison of the naive and multitarget attack on Katana.

In principle it is possible to extend this approach to more levels: if the third failure would be more expensive than the previous two failures one can target $T^{(2)}$

targets to reduce the cost of finding the third failure. However, we did not find a situation in which this was applicable, as finding the third failure is typically much cheaper than finding previous failures.

5 Better failure boosting estimation

The calculation of the work necessary to perform a multitarget attack is not straightforward. Especially the cost of directional failure boosting is expensive to determine and requires multiple approximations to be able to practically compute. D'Anvers et al. [11] introduced crude approximations to reduce the computational cost of this calculation.

Apart from the geometric approximation, as explained in subsection 3.2, they did not consider $\mathcal{G}$, simplified the distribution of $||\mathcal{S}||_2$ into its average and used a simplified formula for the calculation of θ_{SE}. Additionally, there is a weak key effect in multitarget attacks which has not been addressed before[2].

These simplifications are justifiable in the single target attack, where the cost of the second failure is significantly lower than the cost of the first failure. However, in multitarget attacks, where the second failure cost might be dominant, it is important to have an accurate estimation of the cost to find this failure. We will first detail the weak key effect, then we will improve the estimation of $\cos(\theta_{SE})$ and finally we will consider the distribution of $||\mathcal{S}||_2$ and $\mathcal{G}$. We will clearly compare our improvements with the state-of-the-art. In this section we focus on the case where $\chi_s = \chi_e$ and schemes without rounding, while in Section 6 we will extend the estimation techniques for more general schemes, including the NIST finalists Kyber and Saber.

5.1 Weak keys

Some targets might have secret keys that are more prone to decryption failures, which we will call weak keys. It does not seem possible to efficiently identify targets with weak keys from their public key. However, in a multitarget attack, weak key targets are more prone to produce a failing ciphertext. This means that in the second phase of the attack, when looking for the second failure of a certain target, this target will have higher failure probability compared to a single target attack.

In particular, the norm of the secret $||\mathcal{S}||_2$ determines the failure probability of a given target. We will show that the a posteriori distribution of $||\mathcal{S}||_2$, given a multitarget attack where in the first phase $T^{(0)}$ targets are considered, and with failure boosting threshold f_t can be approximated using:

$$P[||\mathcal{S}||_2] \cdot \frac{T^{(0)} P[F \mid ||\mathcal{S}||_2, f_t]}{P[F \mid ||\mathcal{S}||_2, f_t] + (T^{(0)} - 1) \cdot P[F \mid f_t]} \tag{28}$$

To derive this formula, we first introduce the notation $F(t,q)$ to describe the event where the overall first failure occurs at target t on the q^{th} query. Similarly, we

[2] Guo et al. [18] have used the terminology ('weak keys') in their attack, but this refers to public keys that are vulnerable against specific types of ciphertexts.

define $S(t,q)$ as a success at target t on the q^{th} query. $F(t,\cdot)$ signifies the event where the first failure occurs at target t, regardless of at which query this happens. Without loss of generality we denote the target where the first failure occurs as target $t=0$, which implies that $||\mathcal{S}||_2$ denotes the norm of $\mathcal{S}$ for the 0^{th} target. To simplify the derivation, we will assume that the i^{th} query is performed at all targets at the same time, after which they are all checked for decryption failures. We can then write:

$$P[||\mathcal{S}||_2 \mid F(0,\cdot),f_t] \tag{29}$$

$$= P[||\mathcal{S}||_2 \mid f_t] \cdot \frac{P[F(0,\cdot) \mid ||\mathcal{S}||_2,f_t]}{P[F(0,\cdot) \mid f_t]} \tag{30}$$

$$\approx T^{(0)} \cdot P[||\mathcal{S}||_2] \cdot P[F(0,\cdot) \mid ||\mathcal{S}||_2,f_t] \tag{31}$$

where the latter step uses the fact that a failure occurs with equal probability at all $T^{(0)}$ targets without extra information about the norms $||\mathcal{S}||_2$ of the targets.

The term $P[F(0,\cdot) \mid ||\mathcal{S}||_2,f_t]$ can then be extended by explicitly writing it out as a sum over the probabilities of failures at each query round:

$$P[F(0,\cdot) \mid ||\mathcal{S}||_2,f_t] \tag{32}$$

$$= \sum_{q=0}^{\infty} P\left[\begin{matrix} F(t,q),S(i,j) \\ \forall i \in \{0,\ldots,T^{(0)}-1\}, j \in \{0,\ldots,q\} : (i,j) \neq (t,q) \end{matrix} \;\middle|\; ||\mathcal{S}||_2,f_t \right] \tag{33}$$

$$= \sum_{q=0}^{\infty} P\left[\begin{matrix} F(0,q),S(0,j) \\ \forall j \in \{0\ldots,q-1\} \end{matrix} \;\middle|\; ||\mathcal{S}||_2,f_t \right] \cdot P\left[\begin{matrix} S(1,j) \\ \forall j \in \{0\ldots,q\} \end{matrix} \;\middle|\; f_t \right]^{T^{(0)}-1} \tag{34}$$

The failure probability of a target is reduced slightly when successful ciphertexts are found. However, this effect is small, as the information embedded in successful ciphertexts is limited. We therefore assume that the failure probability of ciphertexts does not change when finding successful ciphertexts. This allows us to simplify the expression as:

$$\approx \sum_{q=0}^{\infty} P[F \mid ||\mathcal{S}||_2,f_t] \cdot P[S \mid ||\mathcal{S}||_2,f_t]^q \cdot P[S \mid f_t]^{(T^{(0)}-1)(q+1)}$$

$$\approx P[F \mid ||\mathcal{S}||_2,f_t] \cdot P[S \mid f_t]^{(T^{(0)}-1)} \sum_{q=0}^{\infty} \left(P[S \mid ||\mathcal{S}||_2,f_t] \cdot P[S \mid f_t]^{(T^{(0)}-1)} \right)^q \tag{35}$$

$$\approx \frac{P[F \mid ||\mathcal{S}||_2,f_t] \cdot P[S \mid f_t]^{(T^{(0)}-1)}}{1 - P[S \mid ||\mathcal{S}||_2,f_t] \cdot P[S \mid f_t]^{(T^{(0)}-1)}}$$

$$\approx \frac{P[F \mid ||\mathcal{S}||_2,f_t]}{P[F \mid ||\mathcal{S}||_2,f_t] + (T^{(0)}-1) \cdot P[F|f_t]} \tag{36}$$

where Equation 35 is an infinite geometric sum, and Equation 36 takes a Taylor approximation where only the highest order terms are kept. We will discuss the effect of weak keys in the next section, after its effects on θ_{SE} have been addressed.

5.2 Calculating θ_{SE}

The angle θ_{SE} can be estimated using the simplified failure equation. Assuming a failure occurred at the i^{th} location we know:

$$\overline{\mathcal{S}}^T\overline{\mathcal{C}^{(i)}} > q_t, \tag{37}$$

which can be rewritten as:

$$\cos(\theta_{SE}) > \frac{q_t}{||\mathcal{S}||_2||\mathcal{C}||_2}. \tag{38}$$

The fact that uniform angles in high dimensions strongly tend to orthogonality can be used to approximate this to:

$$\cos(\theta_{SE}) = \frac{q_t}{||\mathcal{S}||_2||\mathcal{C}||_2}. \tag{39}$$

As such, we can estimate the expected value of $\cos(\theta_{SE})$ by assuming independence between $\mathbb{E}[||\mathcal{S}||_2]$ and $\mathbb{E}[||\mathcal{C}||_2]$ as:

$$\mathbb{E}[\cos(\theta_{SE})] = \frac{q_t}{\mathbb{E}[||\mathcal{S}||_2]\mathbb{E}[||\mathcal{C}||_2]}. \tag{40}$$

In [11], the values of $\mathbb{E}[||\mathcal{S}||_2]$ and $\mathbb{E}[||\mathcal{C}||_2]$ were estimated over the original a priori distribution. However, failure boosting increases the expected norm of $||\mathcal{C}||_2$ and the weak key effect increases the expected norm of $\mathbb{E}[||\mathcal{S}||_2]$. Both effects will decrease $\mathbb{E}[\cos(\theta_{SE})]$ and therefore diminish the efficiency of directional failure boosting.

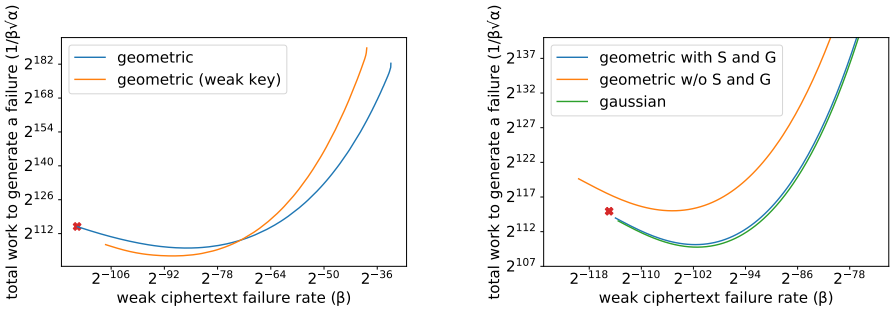
We take these effects into account by considering the a posteriori distributions as follows:

$$\mathbb{E}[||\mathcal{C}||_2] = \sum_{||\mathcal{C}||_2} ||\mathcal{C}||_2 \cdot P[||\mathcal{C}||_2 \mid f_t] \tag{41}$$

$$\mathbb{E}[||\mathcal{S}||_2] = \sum_{||\mathcal{S}||_2} ||\mathcal{S}||_2 \cdot P[||\mathcal{S}||_2 \mid F(0,\cdot), f_t] \tag{42}$$

Note that our expression of $\mathbb{E}[\cos(\theta_{SE})]$ is now significantly better than in previous works, but still not exact for the following reasons: First, $\mathbb{E}[||\mathcal{C}||_2]$ will be slightly higher than calculated above as failures happen with higher probability for higher values of $||\mathcal{C}||_2$. However, this effect is limited as failure boosting pushes $||\mathcal{C}||_2$ to high values where the tails decrease rapidly. Therefore the values of $||\mathcal{C}||_2$ will be strongly focussed around the cut-off value. Secondly, the independence assumption used to obtain Equation 40 is not exact. Nevertheless, the approximation is good enough for our purposes.

Comparison to state-of-the-art: Figure 2a shows the effect of including the weak key effect and improving the $\cos(\theta_{SE})$ estimation. On one hand, one can see that the weak key reduces the failure probability, which is the leftmost point on the curve, from 2^{-115} to 2^{-107}. On the other hand, the increase in $\mathbb{E}[||\mathcal{S}||_2]$ and $\mathbb{E}[||\mathcal{C}||_2]$ and subsequent reduction of $\mathbb{E}[\cos(\theta_{SE})]$ reduces the effectiveness of directional failure boosting, an effect that becomes more pronounced with higher precomputation.



(a) Weak key effect on directional failure boosting cost (second failure) of Katana. The orange curve is estimated using $\beta = 2^{-101}, T^{(0)} = 2^{64}$.

(b) Effect of inclusion of $||\mathcal{S}||_2$ and $\mathcal{G}$ in the failure boosting calculation (first failure) of Katana.

Fig. 2. Effect of inclusion of weak keys and $||\mathcal{S}||_2$ and $\mathcal{G}$ on Katana. The red cross indicates the failure probability of Katana (or equally the cost of finding a failure when random guessing).

5.3 Inclusion of $\mathcal{S}$ and $\mathcal{G}$

In [11], the distributions of $||\mathcal{S}||_2$ and $\mathcal{G}$ were simplified to their mean to speed up calculations. However, the side-effect of this is an underestimation of the failure probability and the attack efficiency. In our calculations, we take into account the distribution of both $||\mathcal{S}||_2$ and $\mathcal{G}$.

Failure boosting: Failure boosting calculations under the geometric approximation can be calculated by making classes based on $||\mathcal{C}||_2$ and using Equations 18 and 19 to determine $\alpha(f_t)$ and $\beta(f_t)$.

Including $\mathcal{S}$ and $\mathcal{G}$ does not change the ciphertext probability $P[cl]$, but does impact the failure probability $P_F[cl]$ needed to calculate $\alpha(f_t)$ and $\beta(f_t)$. A more exact expression of this failure probability that takes into account $||\mathcal{S}||_2$ and $\mathcal{G}$ can be derived as follows:

$$P_F[cl] = P_F[||\mathcal{C}||_2] \tag{43}$$

$$= \sum_{||\mathcal{S}||_2} P[||\mathcal{S}||_2] \cdot P[F \mid ||\mathcal{C}||_2, ||\mathcal{S}||_2] \tag{44}$$

$$= \sum_{||\mathcal{S}||_2} P[||\mathcal{S}||_2] \cdot \left(1 - \prod_{i=0}^{N-1} (1 - P[F_i \mid ||\mathcal{C}||_2, ||\mathcal{S}||_2])\right) \tag{45}$$

$$= \sum_{||\mathcal{S}||_2} P[||\mathcal{S}||_2] \cdot \left(1 - \prod_{i=0}^{N-1} \left(1 - \sum_{\mathcal{G}_i} P[\mathcal{G}_i] \cdot P[F_i \mid ||\mathcal{C}||_2, ||\mathcal{S}||_2, \mathcal{G}_i]\right)\right) \tag{46}$$

where $P[F_i \mid ||\mathcal{C}||_2, ||\mathcal{S}||_2, \mathcal{G}_i]$ can be calculated following the geometric approximation of Equation 15 as:

$$P[F_i \mid ||\mathcal{S}||_2, ||\mathcal{C}||_2, \mathcal{G}_i] = \begin{matrix} P[\cos(\phi) > \frac{q_t - \mathcal{G}_i}{||\mathcal{S}||_2 \cdot ||\mathcal{C}||_2} \mid ||\mathcal{C}||_2, ||\mathcal{S}||_2, \mathcal{G}_i] \\ + P[\cos(\phi) < \frac{-q_t - \mathcal{G}_i}{||\mathcal{S}||_2 \cdot ||\mathcal{C}||_2} \mid ||\mathcal{C}||_2, ||\mathcal{S}||_2, \mathcal{G}_i] \end{matrix}, \tag{47}$$

and where ϕ can be modelled as a uniformly random angle in dimension $2Nl$.

Directional failure boosting: The procedure for directional failure boosting is more complicated, as one should make a list over all values of $||\mathcal{C}||_2$ and $\text{maxcos}_\text{i}(\theta_{\text{C}^{(\text{i})}\text{E}})$. As before, the calculation of $P[cl]$ is the same as in [11], but the calculation of $P_F[cl]$ additionally should take into account $||\mathcal{S}||_2$ and $\mathcal{G}$.

Without loss of generality we will assume that the highest value of $\cos(\theta_{C^{(i)}E})$ occurs at $i=0$, so that $\text{maxcos}_\text{i}(\theta_{\text{C}^{(\text{i})}\text{E}}) = \cos(\theta_{\text{C}^{(0)}\text{E}})$. Similar to the derivation of Equation 46, the failure probability can then be calculated as:

$$P_F[cl] = P_F[||\mathcal{C}||_2, \theta_{C^{(0)}E}] \tag{48}$$

$$= \sum_{||\mathcal{S}||_2} P[||\mathcal{S}||_2] \cdot P[F \mid ||\mathcal{C}||_2, \theta_{C^{(0)}E}, ||\mathcal{S}||_2] \tag{49}$$

$$= \sum_{||\mathcal{S}||_2} P[||\mathcal{S}||_2] \cdot \left(1 - \prod_{i=0}^{N-1} (1 - P[F_i \mid ||\mathcal{C}||_2, \theta_{C^{(0)}E}, ||\mathcal{S}||_2])\right) \tag{50}$$

$$\approx \sum_{||\mathcal{S}||_2} P[||\mathcal{S}||_2] \cdot \left(1 - \left(\begin{matrix} (1 - P[F_0 \mid ||\mathcal{C}||_2, \theta_{C^{(0)}E}, ||\mathcal{S}||_2]) \cdot \\ \prod_{i=1}^{N-1} (1 - P[F_i \mid ||\mathcal{C}||_2, \cos(\theta_{C^{(i)}E}) \leq \cos(\theta_{C^{(0)}E}), ||\mathcal{S}||_2]) \end{matrix}\right)\right) \tag{51}$$

$$\approx \sum_{||\mathcal{S}||_2} P[||\mathcal{S}||_2] \cdot \tag{52}$$

$$\left(1 - \left(\begin{matrix} (1 - \sum_{\mathcal{G}_0} P[\mathcal{G}_0] \cdot P[F_0 \mid ||\mathcal{C}||_2, \theta_{C^{(0)}E}, ||\mathcal{S}||_2, \mathcal{G}_0]) \cdot \\ \prod_{i=1}^{N-1} (1 - \sum_{\mathcal{G}_i} P[\mathcal{G}_i] \cdot P[F_i \mid ||\mathcal{C}||_2, \cos(\theta_{C^{(i)}E}) \leq \cos(\theta_{C^{(0)}E}), ||\mathcal{S}||_2, \mathcal{G}_i]) \end{matrix}\right)\right)$$

$P[F_i \mid ||\mathcal{C}||_2, \theta_{C^{(i)}E}, ||\mathcal{S}||_2, \mathcal{G}_i]$ can be estimated using the geometric assumption and Equation 17 as:

$$\begin{matrix} P[\cos(\psi) > \frac{q_t - \mathcal{G}_i - ||\mathcal{S}||_2 \cdot ||\mathcal{C}||_2 \cdot \cos(\theta_{SE}) \cdot \cos(\theta_{C^{(i)}E})}{||\mathcal{S}||_2 \cdot ||\mathcal{C}||_2 \cdot \sin(\theta_{SE}) \cdot \sin(\theta_{C^{(i)}E})} \mid ||\mathcal{S}||_2, ||\mathcal{C}||_2, \mathcal{G}_i, \cos(\theta_{C^{(i)}E})] \\ + P[\cos(\psi) < \frac{-q_t - \mathcal{G}_i - ||\mathcal{S}||_2 \cdot ||\mathcal{C}||_2 \cdot \cos(\theta_{SE}) \cdot \cos(\theta_{C^{(i)}E})}{||\mathcal{S}||_2 \cdot ||\mathcal{C}||_2 \cdot \sin(\theta_{SE}) \cdot \sin(\theta_{C^{(i)}E})} \mid ||\mathcal{S}||_2, ||\mathcal{C}||_2, \mathcal{G}_i, \cos(\theta_{C^{(i)}E})] \end{matrix}$$

with ψ a uniformly random angle in dimension $2Nl-1$.

The value $P[F_i \mid ||\mathcal{C}||_2, \cos(\theta_{C^{(i)}E}) \leq \cos(\theta_{C^{(0)}E}), ||\mathcal{S}||_2, \mathcal{G}_i]$ can be calculated by taking a weighted average over all $\theta_{C^{(i)}E}$ values for which $\cos(\theta_{C^{(i)}E}) \leq \cos(\theta_{C^{(0)}E})$

as:

$$P[F_i \mid ||\mathcal{C}||_2,\cos(\theta_{C^{(i)}E}) \leq \cos(\theta_{C^{(0)}E}),||\mathcal{S}||_2,\mathcal{G}_i] \tag{53}$$

$$= \sum_{\forall \theta_{C^{(i)}E}:\cos(\theta_{C^{(i)}E}) \leq \cos(\theta_{C^{(0)}E})} P[\theta_{C^{(i)}E}] \cdot P[F_i \mid ||\mathcal{C}||_2,\theta_{C^{(i)}E},||\mathcal{S}||_2,\mathcal{G}_i] \tag{54}$$

Approximate distributions Note that both the failure boosting and directional failure boosting methods require to loop over all possible values of $||\mathcal{C}||_2,\theta_{C^0E},||\mathcal{S}||_2,\mathcal{G}_i$, which is a costly process. To reduce calculation time, these distributions are approximated using a subset of points in the distribution. We use 200 points to approximate $||\mathcal{C}||_2$ and $\theta_{C^{(i)}E}$, 100 points to approximate $||\mathcal{S}||_2$ and a maximum of 40 points to approximate $\mathcal{G}_i$.

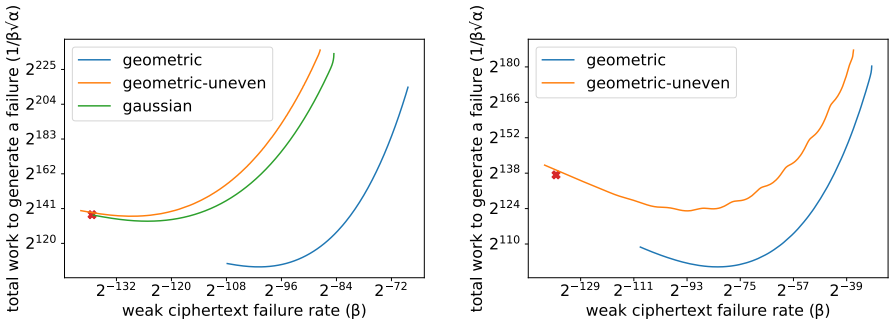
Comparison to state-of-the-art: From Figure 2b, we see that the method that does not take into account $||\mathcal{S}||_2$ and $\mathcal{G}$ does indeed underestimate the failure probability. This effect will become larger for realistic schemes such as Saber and Kyber, who have a larger variance of the distribution of $\mathcal{G}$. Our new methodology that takes $||\mathcal{S}||_2$ and $\mathcal{G}$ into account does match with the reference calculation using the Gaussian approximation, which further confirms our method. Note that this figure presents failure boosting (for the first failure), and that the Gaussian approximation can not be used for directional failure boosting.

6 Dealing with uneven distributions

The cost estimation as described above can not directly be used for calculation of practical schemes that use rounding, such as Kyber or Saber, or more generally schemes that have uneven distributions for the coefficients of $\mathcal{S}$ and $\mathcal{C}$. The main reasons are twofold: first, when the distributions of $\mathbf{s}$ and $\mathbf{e}$ do not have the same variance, values of $||\mathbf{e}'+\mathbf{u}'||_2$ and $||\mathbf{s}'||_2$ have different impact on the overall failure probability. Therefore, using $||\mathcal{C}||_2$ as a predictor of the failure probability, as used in the traditional calculation of direction failure boosting [11], does not give accurate results. Secondly, when rounding occurs, the distributions of $\mathbf{e}$ and $\mathbf{e}'$ are typically not centered and thus the assumption of them following a uniform distribution is not valid.

Note that the Gaussian approximation which is used for the failure boosting (first failure) does not have these problems. Unfortunately it does not seem possible to port the Gaussian assumption to directional failure boosting due to the skew introduced in the distribution of $\mathcal{S}^T\mathcal{C}$ when directional failure boosting is applied.

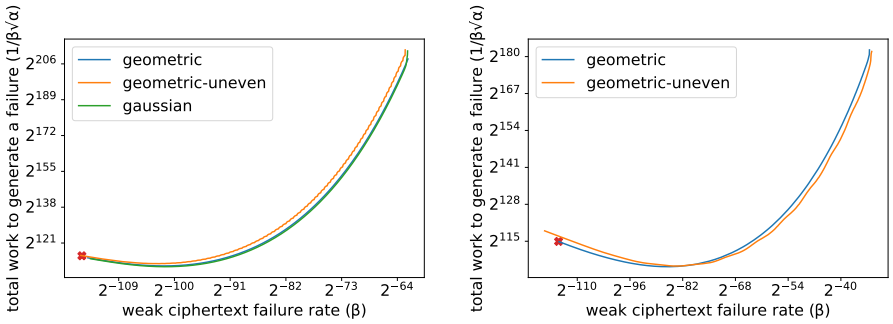
The problems described above have a significant effect on the accuracy of the failure boosting estimation (blue) as can be seen from Figure 3. First, one can see that performing no precomputation (i.e. the leftmost point on the curve, which corresponds to the failure probability before failure boosting) does not correspond to the actual failure probability by a large margin. As an additional check we plotted the Gaussian estimation (green) for finding the first failure, which clearly further shows the discrepancy between both estimations. Looking ahead, we also plotted the geometric-uneven estimate (orange) which will be developed in this section.



(a) Failure boosting (first failure)

(b) Directional failure boosting (second failure)

Fig. 3. Comparison of estimated cost of (directional) failure boosting for Saber. Geometric refers to the method of Section 5, while geometric-uneven indicates the improved method of Section 6 Red cross indicates failure probability (when no precomputation is performed). Gaussian estimation is given for failure boosting as a reference.



(a) Failure boosting (first failure)

(b) Directional failure boosting (second failure)

Fig. 4. Comparison of estimated cost of (directional) failure boosting for Katana. Geometric refers to the method of Section 5, while geometric-uneven indicates the improved method of Section 6 Red cross indicates failure probability (when no precomputation is performed). Gaussian estimation is given for failure boosting as a reference.

6.1 Uneven distributions

When the variance of the coefficients of $\mathbf{s}$ and $\mathbf{e}+\mathbf{u}$ differs, the impact of $||\mathbf{e}'+\mathbf{u}'||_2$ and $||\mathbf{s}'||_2$ varies and they should be considered separately instead of combined in the term $||\mathcal{C}||_2$. For sake of brevity, we will use the following abbreviations:

$$\begin{array}{ll} \mathcal{C}_0 = \mathbf{e}' + \mathbf{u}' & \mathcal{S}_0 = -\mathbf{s} \\ \mathcal{C}_1 = \mathbf{s}' & \mathcal{S}_1 = \mathbf{e} + \mathbf{u} \end{array} \tag{55}$$

Uneven failure boosting: Instead of grouping ciphertexts based on $||\mathcal{C}||_2$, ciphertexts will be grouped in classes based on $||\mathcal{C}_0||_2$ and $||\mathcal{C}_1||_2$. The probability of a class $P[cl]$ can be easily calculated as $P[||\mathcal{C}_0||_2] \cdot P[||\mathcal{C}_1||_2]$, where the distribution of the norms can be calculated exhaustively. The failure probability $P_F[cl]$ becomes more involved to calculate.

Similar to the approach of subsection 5.3, we first include the effect of $\mathcal{S}$ and $\mathcal{G}$, with the difference that we split $||\mathcal{S}||_2$ into $||\mathcal{S}_0||_2$ and $||\mathcal{S}_1||_2$ which leads to:

$$P_F[cl] = P_F[||\mathcal{C}_0||_2, ||\mathcal{C}_1||_2] = \tag{56}$$
$$\sum_{||\mathcal{S}_0||_2} \sum_{||\mathcal{S}_1||_2} \left(\begin{array}{l} P[||\mathcal{S}_0||_2] \cdot P[||\mathcal{S}_1||_2] \cdot \\ \left(1 - \left(1 - \sum_{\mathcal{G}_i} P[\mathcal{G}_i] \cdot P[F_i \mid ||\mathcal{C}_0||_2, ||\mathcal{C}_1||_2, ||\mathcal{S}_0||_2, ||\mathcal{S}_1||_2, \mathcal{G}_i]\right)^N\right) \end{array} \right)$$

To find an expression for $P[F_i \mid ||\mathcal{C}_0||_2, ||\mathcal{C}_1||_2, ||\mathcal{S}_0||_2, ||\mathcal{S}_1||_2, \mathcal{G}_i]$ we go back to the failure term which we rewrite as:

$$\mathcal{S}^T \mathcal{C} + \mathcal{G}_i \tag{57}$$
$$= \mathcal{S}_0^T \cdot \mathcal{C}_0 + \mathcal{S}_1^T \cdot \mathcal{C}_1 + \mathcal{G}_i \tag{58}$$
$$= ||\mathcal{S}_0||_2 \cdot ||\mathcal{C}_0||_2 \cdot \cos(\phi_0) + ||\mathcal{S}_1||_2 \cdot ||\mathcal{C}_1||_2 \cdot \cos(\phi_1) + \mathcal{G}_i \tag{59}$$

Under the geometric assumption, the distribution of ϕ_0 and ϕ_1 can be approximated as angles from the uniform angle distribution in dimension lN. This allows us to calculate the error probability at the i^{th} location for given values of $cond_1 := (||\mathcal{S}_0||_2, ||\mathcal{S}_1||_2, ||\mathcal{C}_0||_2, ||\mathcal{C}_1||_2, \mathcal{G}_i)$ as:

$$P[F \mid cond_1] \tag{60}$$
$$= P[|||\mathcal{S}_0||_2 \cdot ||\mathcal{C}_0||_2 \cdot \cos(\phi_0) + ||\mathcal{S}_1||_2 \cdot ||\mathcal{C}_1||_2 \cdot \cos(\phi_1) + \mathcal{G}_i| > q_t \mid cond_1] \tag{61}$$
$$= \sum_{\phi_0} P[\phi_0] \left(\begin{array}{l} P[\cos(\phi_1) > \frac{q_t - \mathcal{G}_i - ||\mathcal{S}_0||_2 \cdot ||\mathcal{C}_0||_2 \cdot \cos(\phi_0)}{||\mathcal{S}_1||_2 \cdot ||\mathcal{C}_1||_2} \mid cond_1] + \\ P[\cos(\phi_1) < \frac{-q_t - \mathcal{G}_i - ||\mathcal{S}_0||_2 \cdot ||\mathcal{C}_0||_2 \cdot \cos(\phi_0)}{||\mathcal{S}_1||_2 \cdot ||\mathcal{C}_1||_2} \mid cond_1] \end{array} \right) \tag{62}$$

Uneven directional failure boosting: Directional failure boosting not only considers $||\mathcal{C}_0||_2$ and $||\mathcal{C}_1||_2$, but also the angle between the ciphertext and the estimate $\mathcal{E}$. Similar to splitting $||\mathcal{C}||_2$ these angles and the estimate $\mathcal{E}$ also should be split. We will denote with $\mathcal{E}_0$ the estimation of the direction of the secret $\mathcal{S}_0$ and with $\mathcal{E}_1$ the estimation of the direction of the secret $\mathcal{S}_1$. The angles $\theta_{C_0^i E_0}$ and $\theta_{C_0^i E_0}$ denote the angle between $\overline{\mathcal{C}^{(i)}}_0$ and $\mathcal{E}_0$ and between $\overline{\mathcal{C}^{(i)}}_1$ and $\mathcal{E}_1$ respectively.

Ciphertext are then combined in classes based on both the norms and the angles. Ideally one would take the maximal angle out of the lN available angles similar to [11]:

$$cl := \left(||\mathcal{C}_0||_2, ||\mathcal{C}_1||_2, \max_i \cos(\theta_{C_0^i E_0}), \max_i \cos(\theta_{C_1^i E_1}) \right).$$

However, for computational efficiency we only consider failures F_0 at the zero$^{\text{th}}$ coefficient, so that the classes are defined by:

$$cl := \Big(||\mathcal{C}_0||_2, ||\mathcal{C}_1||_2, \theta_{C_0^0 E_0}, \theta_{C_1^0 E_1} \Big).$$

The failure probability is under the same approximation equal to:

$$P_F[cl] \approx P[F_0|cl]$$

For the calculation of $\alpha(f_t)$ and $\beta(f_t)$, the class probability $P[cl]$ can be simplified using independence between the class properties as: $P[||\mathcal{C}_0||_2] \cdot P[||\mathcal{C}_1||_2] \cdot P[\theta_{C_0^0 E_0}] \cdot P[\theta_{C_1^0 E_1}]$. For the failure probability $P_F[cl]$ we first include the influence of $||\mathcal{S}_0||_2$, $||\mathcal{S}_1||_2$ and $\mathcal{G}_0$ as:

$$\begin{aligned} &P_F[cl] \approx P[F_0 \mid cl] \qquad (63)\\ &= \sum_{||\mathcal{S}_0||_2} \sum_{||\mathcal{S}_1||_2} \sum_{\mathcal{G}_0} \begin{pmatrix} P[||\mathcal{S}_0||_2] \cdot P[||\mathcal{S}_1||_2] \cdot P[\mathcal{G}_0] \cdot \\ P[F_0 \mid ||\mathcal{C}_0||_2, ||\mathcal{C}_1||_2, ||\mathcal{S}_0||_2, ||\mathcal{S}_1||_2, \mathcal{G}_0, \theta_{C_0^0 E_0}, \theta_{C_1^0 E_1}] \end{pmatrix}, \end{aligned}$$

and further denoting $cond_2 := \Big(||\mathcal{C}_0||_2, ||\mathcal{C}_1||_2, ||\mathcal{S}_0||_2, ||\mathcal{S}_1||_2, \mathcal{G}_0, \theta_{C_0^0 E_0}, \theta_{C_1^0 E_1} \Big)$, this becomes:

$$= \sum_{||\mathcal{S}_0||_2} \sum_{||\mathcal{S}_1||_2} \sum_{\mathcal{G}_0} P[||\mathcal{S}_0||_2] \cdot P[||\mathcal{S}_1||_2] \cdot P[\mathcal{G}_0] \cdot P[F_0 \mid cond_2]. \qquad (64)$$

To find an expression for the error probability $P[F_0 \mid cond_2]$, we rewrite the failure term as follows:

$$\begin{aligned} &\overline{\mathcal{S}}^T \overline{\mathcal{C}^{(0)}} + \mathcal{G}_0 \qquad (65)\\ &= \overline{\mathcal{S}}_0^T \overline{\mathcal{C}^{(0)}}_0 + \overline{\mathcal{S}}_1^T \overline{\mathcal{C}^{(0)}}_1 + \mathcal{G}_0 \qquad (66)\\ &= \overline{\mathcal{S}}_{0,\parallel}^T \overline{\mathcal{C}^{(0)}}_{0,\parallel} + \overline{\mathcal{S}}_{0,\perp}^T \overline{\mathcal{C}^{(0)}}_{0,\perp} + \overline{\mathcal{S}}_{1,\parallel}^T \overline{\mathcal{C}^{(0)}}_{1,\parallel} + \overline{\mathcal{S}}_{1,\perp}^T \overline{\mathcal{C}^{(0)}}_{1,\perp} + \mathcal{G}_0 \qquad (67)\\ &= ||\mathcal{S}_0||_2 ||\mathcal{C}_0||_2 \cos(\theta_{S_0 E_0}) \cos(\theta_{C_0^0 E_0}) + ||\mathcal{S}_0||_2 ||\mathcal{C}_0||_2 \sin(\theta_{S_0 E_0}) \sin(\theta_{C_0^0 E_0}) \cos(\psi_0)\\ &+ ||\mathcal{S}_1||_2 ||\mathcal{C}_1||_2 \cos(\theta_{S_1 E_1}) \cos(\theta_{C_1^0 E_1}) + ||\mathcal{S}_1||_2 ||\mathcal{C}_1||_2 \sin(\theta_{S_1 E_1}) \sin(\theta_{C_1^0 E_1}) \cos(\psi_1)\\ &+ \mathcal{G}_0, \qquad (68) \end{aligned}$$

with $\theta_{S_0 E_0}$ and $\theta_{S_1 E_1}$ the angles between $\mathcal{S}_0$ and $\mathcal{E}_0$, and $\mathcal{S}_1$ and $\mathcal{E}_1$ respectively. Following the geometric approximation, ψ_0 and ψ_1 are uniformly random angles in dimension $Nl-1$. The failure probability can then be calculated as:

$$\begin{aligned} &P[F_0 \mid cond_2] = \qquad (69)\\ &\sum_{\psi_0} P[\psi_0] \begin{pmatrix} P[\cos(\psi_1) > \frac{q_t - \mathcal{G}_0 - w}{||\mathcal{S}_1||_2 ||\mathcal{C}_1||_2 \sin(\theta_{S_1 E_1}) \sin(\theta_{C_1^0 E_1})}] \mid cond_2, \psi_0] \\ + P[\cos(\psi_1) < \frac{-q_t - \mathcal{G}_0 - w}{||\mathcal{S}_1||_2 ||\mathcal{C}_1||_2 \sin(\theta_{S_1 E_1}) \sin(\theta_{C_1^0 E_1})}] \mid cond_2, \psi_0] \end{pmatrix}, \end{aligned}$$

where:

$$w = \begin{pmatrix} ||\mathcal{S}_1||_2 ||\mathcal{C}_1||_2 \cos(\theta_{S_1 E_1}) \cos(\theta_{C_1^0 E_1}) \\ + ||\mathcal{S}_0||_2 ||\mathcal{C}_0||_2 \cos(\theta_{S_0 E_0}) \cos(\theta_{C_0^0 E_0}) \\ + ||\mathcal{S}_0||_2 ||\mathcal{C}_0||_2 \sin(\theta_{S_0 E_0}) \sin(\theta_{C_0^0 E_0}) \cos(\psi_0) \end{pmatrix}. \qquad (70)$$

6.2 Meet-in-the-middle speedup

While the uneven directional failure boosting method is much more precise for schemes with uneven distributions than the original method of [11], it is computationally very demanding. The prescribed calculation in subsection 6.1 sums over the distributions of $\mathcal{C}_0$, $\mathcal{C}_1$, $\mathcal{S}_0$, $\mathcal{S}_1$, $\mathcal{G}_0$, $\theta_{C_0^0E_0}$, $\theta_{C_1^0E_1}$ and ψ_0. Even when these distributions are approximated, the trade-off between computational cost and accuracy remains unsatisfactory. In this section we will introduce a meet-in-the-middle approach to reduce the computational cost of this method.

From Equation 68, we can see that the failure equation can be written as:

$$x_0\cos(\psi_0)+x_1\cos(\psi_1)+z+\mathcal{G}_0 \tag{71}$$

where:

$$x_0 = ||\mathcal{C}_0||_2 \cdot ||\mathcal{S}_0||_2 \cdot \sin(\theta_{C_0^0E_0}) \cdot \sin(\theta_{SE_0}) \tag{72}$$

$$x_1 = ||\mathcal{C}_1||_2 \cdot ||\mathcal{S}_1||_2 \cdot \sin(\theta_{C_1^0E_1}) \cdot \sin(\theta_{SE_1}) \tag{73}$$

$$z = \begin{pmatrix} ||\mathcal{C}_0||_2 \cdot ||\mathcal{S}_0||_2 \cdot \cos(\theta_{C_0^0E_0}) \cdot \cos(\theta_{SE_0}) + \\ ||\mathcal{C}_1||_2 \cdot ||\mathcal{S}_1||_2 \cdot \cos(\theta_{C_1^0E_1}) \cdot \cos(\theta_{SE_1}) \end{pmatrix} \tag{74}$$

The work can then be split into a precomputation, where the failure probability given x_0, x_1 and z is calculated (i.e. $P_F[x_0,x_1,z]$), and the directional failure boosting calculation itself, which can now use the precomputed values of $P_F[x_0,x_1,z]$ to reduce calculations. During precomputation $P_F[x_0,x_1,z]$ is calculated for a wide range of x_0,x_1 and z values as:

$$P_F[x_0,x_1,z] \approx P[F_0 \mid x_0,x_1,z] \tag{75}$$

$$= P[|x_0\cos(\phi_0)+x_1\cos(\phi_1)+z+\mathcal{G}_0| > q_t \mid x_0,x_1,z] \tag{76}$$

$$= \sum_{\mathcal{G}_0}\sum_{\phi_0} P[\mathcal{G}_0]\cdot P[\phi_0]\cdot P[|x_0\cos(\phi_0)+x_1\cos(\phi_1)+z+\mathcal{G}_0| > q_t \mid x_0,x_1,z] \tag{77}$$

$$= \sum_{\mathcal{G}_0}\sum_{\phi_0} P[\mathcal{G}_0]\cdot P[\phi_0]\cdot \begin{pmatrix} P[\cos(\phi_1) > \frac{q_t - z - \mathcal{G}_0 - x_0\cos(\phi_0)}{x_1} \mid x_0,x_1,z] + \\ P[\cos(\phi_1) < \frac{-q_t - z - \mathcal{G}_0 - x_0\cos(\phi_0)}{x_1} \mid x_0,x_1,z] \end{pmatrix} \tag{78}$$

Using the precomputation, the directional failure boosting calculation of $P_F[ct]$ can then be simplified as:

$$P_F[ct] \approx P[F_0|ct] \tag{79}$$

$$= \sum_{||\mathcal{S}_0||_2}\sum_{||\mathcal{S}_1||_2} P[||\mathcal{S}_0||_2]\cdot P[||\mathcal{S}_1||_2]\cdot P[F_0|ct,||\mathcal{S}_0||_2,||\mathcal{S}_1||_2] \tag{80}$$

$$= \sum_{||\mathcal{S}_0||_2}\sum_{||\mathcal{S}_1||_2} \begin{pmatrix} P[||\mathcal{S}_0||_2]\cdot P[||\mathcal{S}_1||_2]\cdot \\ P\left[F_0 \middle| \begin{array}{l} x_0 = ||\mathcal{C}_0||_2 \cdot ||\mathcal{S}_0||_2 \cdot \sin(\theta_{C_0^0E_0}) \cdot \sin(\theta_{SE_0}), \\ x_1 = ||\mathcal{C}_1||_2 \cdot ||\mathcal{S}_1||_2 \cdot \sin(\theta_{C_1^0E_1}) \cdot \sin(\theta_{SE_1}), \\ z = \begin{pmatrix} ||\mathcal{C}_0||_2 \cdot ||\mathcal{S}_0||_2 \cdot \cos(\theta_{C_0^0E_0}) \cdot \cos(\theta_{SE_0}) + \\ ||\mathcal{C}_1||_2 \cdot ||\mathcal{S}_1||_2 \cdot \cos(\theta_{C_1^0E_1}) \cdot \cos(\theta_{SE_1}) \end{pmatrix} \end{array} \right] \end{pmatrix} \tag{81}$$

with the values of $P[F_0 \mid x_0,x_1,z]$ as calculated in the precomputation.

The precomputation loops over a grid of (x_0,x_1,z) values, and for each gridpoint sums over the distribution of $\mathcal{G}_0$ and ϕ_0. In total, the precomputation thus only loops 5 distributions. The (x_0,x_1,z) grid is calculated over 100 values for each of the elements, and intermediate values of $P[F_0 \mid x_0,x_1,z]$ are linearly interpolated.

The directional failure boosting loops over the distributions of $\mathcal{C}_0$, $\mathcal{C}_1$, $\mathcal{S}_0$, $\mathcal{S}_1$, $\theta_{C_0^0 E_0}$, $\theta_{C_1^0 E_1}$, which is a total of 6 distributions. This can be compared to the loop over 8 distributions in the direct method that does not use meet-in-the-middle calculations. As a result, our meet-in-the-middle approach makes it possible to practically calculate the cost of directional failure boosting for practical schemes such as Saber and Kyber.

6.3 Removing the bias

One of the assumptions that is explicitly used for the geometric estimation of (directional) failure boosting is that the angles ψ_0 and ψ_1 are distributed uniformly random. This corresponds to the idealized scenario where the secret is drawn from a continuous Gaussian distribution, but it is well approximated by binomial distribution, which is typically used in practical designs. In case of rounding, there is typically a bias in the distribution due to a non-zero mean, as a result of which there will be a 'sense of direction' in $\mathcal{C}_0$ and $\mathcal{S}_1$.

To remove this 'sense of direction' we subtract the mean of the distribution of the coefficients of $\mathcal{C}_0$ and $\mathcal{S}_1$:

$$\mathcal{C}_0' = \mathcal{C}_0 - \mu_{\chi_e+\chi_s} \tag{82}$$
$$\mathcal{S}_1' = \mathcal{S}_1 - \mu_{\chi_e+\chi_s}, \tag{83}$$

This subtraction needs to be compensated to keep a correct failure equation, which can be done as follows:

$$\mathcal{S}_0^T\mathcal{C}_0 + \mathcal{S}_1^T\mathcal{C}_1 + \mathcal{G} \tag{84}$$
$$= \mathcal{S}_0^T\mathcal{C}_0' + \mathcal{S}_1'^T\mathcal{C}_1 + (\mathcal{G} + \mu_{\chi_e+\chi_s}\cdot\mathcal{S}_0 + \mu_{\chi_e+\chi_s}\mathcal{C}_1) \tag{85}$$

And thus by selecting:

$$\mathcal{G}' = \mathcal{G} + \mu_{\chi_e+\chi_s}\cdot\mathcal{S}_0 + \mu_{\chi_e+\chi_s} + \mathcal{C}_1, \tag{86}$$

we can use the failure term $\mathcal{S}_0^T\mathcal{C}_0' + \mathcal{S}_1'^T\mathcal{C}_1 + \mathcal{G}'$, which has exactly the same failure probability. However, this term will give slightly lower efficiency of failure boosting, as an adversary only considers $\mathcal{C}_0$ and $\mathcal{C}_1$, and not $\mathcal{G}$, to determine the weakness of ciphertexts. To apply this adjustment to previous techniques one just has to use the $\mathcal{C}_0'$, $\mathcal{S}_1'$ and $\mathcal{G}'$ instead of $\mathcal{C}_0$, $\mathcal{S}_1$ and $\mathcal{G}$.

6.4 Discussion

Figure 3 and Figure 4 give an indication of the accuracy of our newly developed geometric-uneven methods. First, one can see that both in the case of Saber and

Katana, the attack cost when performing no precomputation (the leftmost point on the curves) is approximately the failure probability. This is expected behaviour, but it is not the case for Saber in the geometric calculations following Section 5. This is a first indication that the geometric-uneven method is more accurate than the standard geometric method in this case.

Secondly, one can see that the geometric-uneven curve is relatively close to the Gaussian curve in the failure boosting (first failure) case. For Saber the geometric-uneven approximation gives a significantly more accurate result compared to the geometric approximation. Overall, the geometric-uneven estimation gives an overestimation of the attack cost, which is logical in view of the assumptions and approximations made in its derivation (e.g. only considering F_0 and making the distributions symmetric). On the other hand, for Katana the geometric approach is more accurate than the geometric-uneven approach, which makes sense as the scheme has $\chi_s = \chi_e$ and does not perform rounding.

One can therefore conclude that the geometric approach is best suited for symmetric non-rounding schemes like Katana, while the geometric-uneven approach is considerably better than the geometric approach for practical schemes such as Saber and Kyber.

7 Attack constraints

In previous derivations, as in literature [12,11], it is assumed that there is an unlimited number of possible ciphertexts. However, for schemes that use the FO transformation, ciphertexts are generated deterministicaly from a message $m \in \mathcal{M}$, and as such there are only $|\mathcal{M}|$ ciphertexts for each public key. When an attacker performs strong failure boosting, this maximum number of ciphertexts $|\mathcal{M}|$ might be a limit to the number of weak ciphertexts an adversary can generate, which in turn could limit or even obstruct an attack.

In a failure boosting attack an adversary first searches for weak ciphertexts, which occur with a probability $\alpha(f_t)$. This means that there are on average $|\mathcal{M}| \cdot \alpha(f_t)$ weak ciphertexts that can be found at each target, and thus $|\mathcal{M}| \cdot \alpha(f_t) \cdot T_1$ in total for T_1 targets. It is expected that an attacker needs $\beta(f_t)^{-1}$ of these weak ciphertexts to find one decryption failure and thus an adversary that wants to collect T_2 failures would need $\beta(f_t)^{-1} \cdot T_2$ weak ciphertexts. In short, there are on average $|\mathcal{M}| \cdot \alpha(f_t) \cdot T_1$ weak ciphertexts available, and an adversary would need on average $\beta(f_t)^{-1} \cdot T_2$ of them to proceed to the next phase of the attack.

From the above we can conclude that if $\beta(f_t)^{-1} \cdot T_2 > |\mathcal{M}| \cdot \alpha(f_t) \cdot T_1$, it is probable that the attacker will not find sufficient unique ciphertexts to obtain T_2 decryption failures. Even in the case where $\beta(f_t)^{-1} \cdot T_2 \approx |\mathcal{M}| \cdot \alpha(f_t) \cdot T_1$, the attack will become less efficient as the adversary will with high probability generate non-unique weak ciphertexts, which requires him to restart the precomputation. For $\beta(f_t)^{-1} \cdot T_2 < |\mathcal{M}| \cdot \alpha(f_t) \cdot T_1$, these effects can be expected to be negligible, as there will be enough weak ciphertexts to avoid duplication. To take this observation into account one can add an additional constraint in the attack calculations using the following restriction on f_t: $\beta(f_t)^{-1} \cdot \alpha(f_t)^{-1} < |\mathcal{M}| \cdot T_1 / T_2$.

Another possible obstacle for an attacker is the maximum depth D_{max} of the quantum computer used for the precomputation. Such depth limit reduces the Grover search success probability if $\sqrt{\alpha(f_t)^{-1}} \gg D_{max}$. This can be compensated for by splitting the search space in p partitions and performing a Grover search of depth D_{max} in each partition. Asymptotically one would need $\alpha(f_t)^{-1}/D_{max}^2$ partitions to find a weak ciphertext with probability close to 1.

Thus, when $\sqrt{\alpha(f_t)^{-1}} \leq D_{max}$, the maximum depth does not restrict the Grover search and the cost to find a weak ciphertext is $\sqrt{\alpha(f_t)^{-1}}$, but when $\sqrt{\alpha(f_t)^{-1}} > D_{max}$, the cost is $D_{max} \cdot \alpha(f_t)^{-1}/D_{max}^2 = \alpha(f_t)^{-1}/D_{max}$.

8 Results

We calculated the multitarget attack cost using the geometric-uneven approach for all parameter sets of Saber, Kyber and uSaber with a query limit of 2^{64} per target. In Table 3, we first give the attack cost for 2^{40} and 2^{64} targets following the procedure described until Section 6, where $|\mathcal{M}| = \infty$ and $D_{max} = \infty$.

We then recalculate the results for 2^{64} targets with the following restrictions: in a first instance $|\mathcal{M}| = 2^{256}$, which is the case for the current designs of these schemes, and a second instance $|\mathcal{M}|$ is taken equal to the equivalent AES strength, i.e. 2^{128} schemes that are in NIST category 1, 2^{192} for schemes in NIST category 3 and 2^{256} for schemes in NIST category 5. The maximum depth is in both cases set to $D_{max} = 2^{96}$, which is the worst case scenario put forward by NIST [1]. A graphical overview of the attack for all parameter sets of Saber and Kyber is given in Appendix A, where the full line represents $D_{max} = \infty$ and where the dotted line represents $D_{max} = 2^{96}$.

An interested reader can generate their own numbers and figures for specific constraints using the python source code, which is made available at https://github.com/KULeuven-COSIC/PQCRYPTO-decryption-failures.

8.1 Impact on Saber and Kyber

Before discussing the security impact of our attack on the targeted schemes, we want to go into some considerations considering the attack model. The failure boosting attack cost is expressed in terms of precomputational work $\mathcal{W}$ and queries $\mathcal{Q}$: $1\mathcal{W}$ refers to the cost of 1 offline encapsulation and the quantum speedup is assumed to be quadratic, ignoring subexponential costs; $1\mathcal{Q}$ describes the cost of 1 decapsulation, which is performed as classical computations.

In a real-life scenario, one needs to take into account the fact that $1\mathcal{Q}$ involves performing a decapsulation query online on the targets hardware, which might be a critical constraint in mounting a practical attack. For Saber, in an ideal scenario our attack requires at least 2^{98} queries and thus encapsulations performed on the attacked hardware for an attack that costs $2^{168}\mathcal{W}$. For the attack reported in Table 3, the query cost is 2^{126} queries.

Moreover, in the offline precomputation step one has to take into account the cost of performing the encapsulation ($1\mathcal{W}$). The Grover search is additionally constraint when considering a depth d for executing one encapsulation, leading

	Quantum Sec.	$P[F]$	Single Target	$T^{(0)}=2^{40}$ $(\mathcal{W}/\mathcal{Q})$ $\lvert\mathcal{M}\rvert=\infty$, $D_{max}=\infty$		$T^{(0)}=2^{64}$ $(\mathcal{W}/\mathcal{Q})$ $\lvert\mathcal{M}\rvert=\infty$, $D_{max}=\infty$		$T^{(0)}=2^{64}$ $(\mathcal{W}/\mathcal{Q})$ $\lvert\mathcal{M}\rvert=2^{256}$, $D_{max}=2^{96}$		$T^{(0)}=2^{64}$ $(\mathcal{W}/\mathcal{Q})$ $\lvert\mathcal{M}\rvert=\text{AES}^{\dagger}$, $D_{max}=2^{96}$	
				Naive	Levelled	Naive	Levelled	Naive	Levelled	Naive	Levelled
LightSaber	107	2^{-120}	–	117 / 102	117 / 102	116 / 108	116 / 108	116 / 108	116 / 108	–	124 / 115
Saber [6]	172	2^{-136}	–	157 / 102	**157** / 102	148 / 125	**141** / 126	**148**/126	**141** / 126	–	**144** / 126
FireSaber	236	2^{-165}	–	–	–	–	**216** / 126	–	–	–	–
Kyber512	107	2^{-139}	–	138 / 102	138 / 102	131 / 118	131 / 118	131/118	131/118	–	–
Kyber768 [27]	165	2^{-164}	–	208 / 102	208 / 102	187 / 112	175 / 126	–	186/126	–	–
Kyber1024	232	2^{-174}	–	–	–	–	**228** / 126	–	–	–	–
uLightSaber	101	2^{-184}	–	–	–	–	–	–	–	–	–
uSaber [6]	165	2^{-167}	–	–	–	–	–	–	–	–	–
uFireSaber	232	2^{-154}	–	–	–	–	–	–	–	–	–
Saber - $2t$	172	2^{-156}	–	213 / 102	213 / 102	180 / 126	**170** / 126	–	174 / 126	–	–
Saber - $4t$	172	2^{-165}	–	247 / 102	247 / 102	196 / 126	187 / 126	–	215 / 126	–	–

† $\lvert\mathcal{M}\rvert$ taken equal to the number of messages in the corresponding AES instance
i.e. 2^{128} schemes that are in NIST category 1, 2^{192} for schemes in NIST category 3 and 2^{256} for schemes in NIST category 5.

Table 3. Cost of failure boosting attacks on various schemes. Security values are given $\log_2$. Empty cells indicate an attack is not possible in under $2^{256}\mathcal{W}$.

to a cost of $\alpha(f_t)^{-1} \cdot d/D_{max}\mathcal{W}$ when $\sqrt{\alpha(f_t)^{-1}} > D_{max}/d$ where the cost of one encapsulation is still counted as $1\mathcal{W}$.

Our analysis shows that the category 3 instance of Saber is theoretically vulnerable for a decryption failure attack. A decryption failure attack on Saber would cost $2^{145}\mathcal{W}$ and $2^{126}\mathcal{Q}$ in the specific setting where $q_{limit} = 2^{64}$ and $T^{(0)} = 2^{64}$, which can be compared to the claimed 2^{172} coreSVP security. However, practical execution of the attack would not be straightforward due to the constraints outlined above. The other parameter sets of Saber and Kyber are not vulnerable to the decryption failure attack we developed, in case of Kyber1024 and FireSaber this is due to the constraints on the number of ciphertexts due to $|\mathcal{M}|$. The uSaber parameter sets are not vulnerable to the decryption failure attacks we developed, even without additional constraints.

8.2 Increasing the attack cost

One option to increase the attack cost could be to reduce $|\mathcal{M}|$. Such a design change does not incur an efficiency cost but is limited by the security of the overall scheme as a too low value for $|\mathcal{M}|$ could impact the security under traditional attacks. The effect of a reduction of $|\mathcal{M}|$ to 2^{128} and 2^{192} for schemes of category 1 and 3 respectively is detailed in the last column of Table 3. Note that this change will especially restrain the efficiency of finding follow up failures, as the term $|\mathcal{M}| \cdot T_1/T_2$ is typically much higher for finding the first failure due to a high value of the number of targets T_1. Therefore, a reduction in $|\mathcal{M}|$ is also a good precaution for future advances in decryption failure attacks as will be discussed in subsection 8.3.

Looking at the error term $(\mathbf{e}+\mathbf{u})^T\mathbf{s}' - \mathbf{s}^T(\mathbf{e}'+\mathbf{u}') + (\mathbf{e}''+\mathbf{u}'')$, the compression error $\mathbf{u}''$ can be a significant factor in decryption failures in schemes with strong compression of $\mathbf{v}'$ (i.e. large q/t). In this case the attack cost can be increased by increasing t. This comes at a modest cost in ciphertext size, but generally has no impact on the security of the scheme under non-decryption failure attacks. For Saber, increasing t to $2t$ would make the attack more expensive than solving the Mod-LWR problem while increasing the ciphertext size with only 256 bits. The impact of such a change for Saber is given in the last rows of Table 3.

If increasing t is not sufficient, one needs to adapt the distributions of χ_s and χ_e, which would impact both security as design and thus would require a more in-depth analysis.

8.3 Possible future advances

In this subsection we go into detail on possible future advances in failure boosting and its cost estimations.

Failure boosting The cost calculation of failure boosting takes into account both $\mathcal{C}$ and $\mathcal{G}$ and makes two assumptions. The first being that errors at different coefficients of the message are independent, which has been shown by D'Anvers et al. [13] to be a valid assumption for schemes without error correction. The second being the Gaussian assumption as discussed in subsection 3.2. As a result, the attack cost calculation of failure boosting is nearly optimal in the failure boosting framework.

Directional failure boosting The directional failure boosting calculation uses more assumptions and approximations that make the estimate less accurate. Specifically, the attack relies on two assumptions: The geometric-uneven assumption states that the distributions of $\mathcal{S}_0$, $\mathcal{C}_0$, $\mathcal{S}_1$ and $\mathcal{C}_1$ are multivariate Gaussian distributed with zero mean and equal variance for each coefficient. This is a fairly good approximation for binomial distributions with large variance, but is less accurate for small variance binomial distributions or uniform distributions as is the case in Kyber and Saber. The second assumption is the independency assumption that is also used in the failure boosting calculation and is valid for schemes without error correction.

Furthermore, the directional failure boosting calculation in this work considers a slightly suboptimal attack as some terms are not taken into account in the weak ciphertext selection criterion: First, the attack does not take into account $\mathcal{G}$ in the weak key selection (but it does for the failure probability calculation). Secondly, it removes the bias of $\mathcal{S}_0$ and $\mathcal{C}_1$ due to rounding, and adds it to the term $\mathcal{G}$ as explained in subsection 6.3. Therefore, the above approximations correspond to executable attacks, but the attack is slightly suboptimal as a better weak ciphertext selection criterion (e.g. taking $\mathcal{G}$ into account) would lead to a more efficient attack.

Finally, the directional failure boosting calculation makes two significant approximations: First, in the geometric-uneven directional failure boosting approach, only the error probability of the first bit of the message is considered. This would lead to an underestimation of the failure probability and thus an overestimation of the attack cost. Secondly, the distributions of the different variables are approximated using a limited number of points.

The previous assumptions and approximations are necessary to allow efficient calculation of the attack cost. However, they could result in a less optimal attack and a less accurate cost estimation for directional failure boosting. During the development of our cost estimation methods in Sections 5 and 6 we showed that our calculation methods are still reasonably accurate using three checks:

First we checked the failure probability when no precalculation is performed, which should correspond to the failure probability of the scheme. As shown in the paper, this is always approximately the case for our cost estimation methods (but not in case of Saber or Kyber in the geometric case, which led us to argue that this method is not appropriate for Saber or Kyber).

Secondly, we checked our geometric and geometric-uneven methods in the failure boosting case using the more accurate Gaussian approximation, where we could see that our newly developed methods give approximately the same result. Note that this comparison is not possible in the directional failure boosting case.

Thirdly, we verified the geometric-uneven method using the geometric method in case of Katana. As the latter method makes less approximations and as its assumptions are valid for Katana, this comparison can be used to verify some of the new assumptions (i.e. removing the bias in subsection 6.3 and only considering errors at the first coefficient in subsection 6.1) made in the geometric-uneven method compared to the geometric method.

Conclusion For schemes where the attack cost is dominated by finding the first failure, the calculated cost will be close to the optimal decryption failure attack cost

	Full attack		First failure
	$\mathcal{W}_0/\mathcal{Q}_0$	$\mathcal{W}_1/\mathcal{Q}_1$	$\mathcal{W}_0/\mathcal{Q}_0$
LightSaber	**116 / 108**	104 / 62	116 / 108
Saber	140 / 126	**140 / 68**	133 / 125
FireSaber	215 / 126	**215 / 68**	188 / 128
Kyber512	**131 / 118**	129 / 62	131 / 118
Kyber768	174 / 126	**175 / 69**	161 / 128
Kyber1024	228 / 126	**219 / 71**	191 / 128

Table 4. Cost ($\log_2$) of obtaining the first and second failure in our multitarget attack and cost of obtaining only the first failure if the second failure would be free. $q_{limit} = 2^{64}$ and $T^{(0)} = 2^{64}$. Text is made bold for dominating factor in the attack cost. When performing a levelled multitarget attack where $T^{(1)} \neq 1$, the search for the second failure is considered dominant.

(unless a radical new attack is discovered that outperforms failure boosting). For schemes with an attack cost dominated by directional failure boosting, the estimation will be less accurate. In a worst case attack scenario (from the designers perspective) one could assume the directional failure boosting cost to be reduced even more, leading to an attack that is essentially dominated by finding the first failure. Note that this is a very conservative approach and does not correspond to an existing attack scenario. An overview of the dominant attack costs can be found in Table 4.

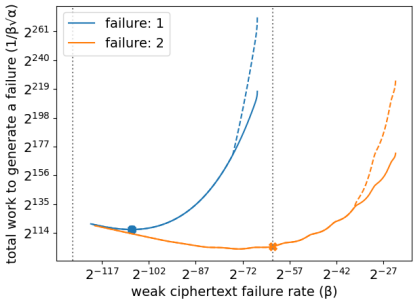
References

1. Submission requirements and evaluation criteria for the post-quantum cryptography standardization process, 2016. https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf.
2. M. R. Albrecht, B. R. Curtis, A. Deo, A. Davidson, R. Player, E. W. Postlethwaite, F. Virdia, and T. Wunderer. Estimate all the LWE, NTRU schemes! In *SCN 18*, LNCS, 2018.
3. M. R. Albrecht, R. Player, and S. Scott. On the concrete hardness of learning with errors. *Journal of Mathematical Cryptology*, 9(3):169–203, 2015.
4. S. Bai and S. D. Galbraith. An improved compression technique for signatures based on learning with errors. In *CT-RSA*. Springer, 2014.
5. A. Banerjee, C. Peikert, and A. Rosen. Pseudorandom functions and lattices. In *EUROCRYPT 2012*, LNCS, 2012.
6. A. Basso, J. M. B. Mera, J.-P. D'Anvers, A. Karmakar, S. S. Roy, M. V. Beirendonck, and F. Vercauteren. SABER. Technical report, National Institute of Standards and Technology, 2020. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions.
7. N. Bindel and J. M. Schanck. Decryption failure is more likely after success. In *Post-Quantum Cryptography - 11th International Conference, PQCrypto 2020*, 2020.
8. J. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. M. Schanck, P. Schwabe, and D. Stehlé. CRYSTALS – Kyber: a CCA-secure module-lattice-based KEM. IACR ePrint 2017/634.

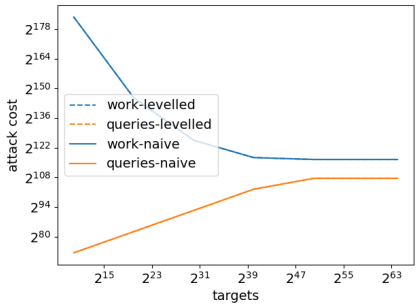
9. D. Dachman-Soled, L. Ducas, H. Gong, and M. Rossi. LWE with side information: Attacks and concrete security estimation. In *CRYPTO 2020, Part II*, LNCS, 2020.
10. J.-P. D'Anvers, A. Karmakar, S. S. Roy, and F. Vercauteren. Saber: Module-LWR based key exchange, CPA-secure encryption and CCA-secure KEM. In *AFRICACRYPT 18*, LNCS, 2018.
11. J.-P. D'Anvers, M. Rossi, and F. Virdia. (One) failure is not an option: Bootstrapping the search for failures in lattice-based encryption schemes. In *EUROCRYPT 2020, Part III*, LNCS, 2020.
12. J.-P. D'Anvers, F. Vercauteren, and I. Verbauwhede. 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.
13. J.-P. D'Anvers, F. Vercauteren, and I. Verbauwhede. The impact of error dependencies on ring/mod-LWE/LWR based schemes. In *Post-Quantum Cryptography - 10th International Conference, PQCrypto 2019*, 2019.
14. A. W. Dent. A designer's guide to KEMs. In *9th IMA International Conference on Cryptography and Coding*, LNCS, 2003.
15. E. Fujisaki and T. Okamoto. Secure integration of asymmetric and symmetric encryption schemes. *Journal of Cryptology*, (1), 2013.
16. C. Gentry and D. Boneh. *A fully homomorphic encryption scheme*. Stanford University Stanford, 2009.
17. C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *STOC 08*. ACM, 2008.
18. Q. Guo, T. Johansson, and A. Nilsson. 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.
19. Q. Guo, T. Johansson, and J. Yang. A novel CCA attack using decryption errors against LAC. In *ASIACRYPT 2019, Part I*, LNCS, 2019.
20. D. Hofheinz, K. Hövelmanns, and E. Kiltz. A modular analysis of the Fujisaki-Okamoto transformation. In *TCC 2017, Part I*, LNCS, 2017.
21. É. Jaulmes and A. Joux. A Chosen-Ciphertext Attack against NTRU. In *CRYPTO 2000*, 2000.
22. A. Langlois and D. Stehlé. Worst-case to average-case reductions for module lattices. *Designs, Codes and Cryptography*, 2015.
23. V. Lyubashevsky. Fiat-shamir with aborts: Applications to lattice and factoring-based signatures. In *ASIACRYPT*, 2009.
24. V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. In *EUROCRYPT 2010*, LNCS, 2010.
25. M. Naehrig, E. Alkim, J. Bos, L. Ducas, K. Easterbrook, B. LaMacchia, P. Longa, I. Mironov, V. Nikolaenko, C. Peikert, A. Raghunathan, and D. Stebila. FrodoKEM. Technical report, NIST, 2017.
26. O. Regev. On Lattices, Learning with Errors, Random Linear Codes, and Cryptography. In *STOC*. ACM, 2005.
27. P. Schwabe, R. Avanzi, J. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. M. Schanck, G. Seiler, and D. Stehlé. CRYSTALS-KYBER. Technical report, National Institute of Standards and Technology, 2020. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions.
28. E. E. Targhi and D. Unruh. Post-quantum security of the Fujisaki-Okamoto and OAEP transforms. In *TCC 2016-B, Part II*, LNCS, 2016.
29. R. Tsabary. Fully secure attribute-based encryption for t-cnf from lwe. In *CRYPTO 2019*. Springer, 2019.

Auxiliary Supporting Material

A Detailed attack costs

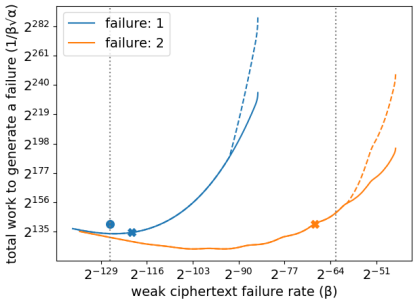


(a) Cost to find a failure vs weak ciphertext failure rate for $q_{limit} = 2^{64}$ and $T^{(0)} = 2^{64}$.

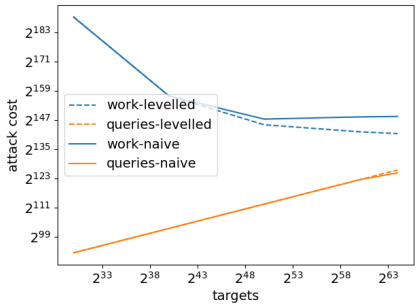


(b) Attack cost in function of targets $T^{(0)}$ for $q_{limit} = 2^{64}$.

Fig. 5. LightSaber

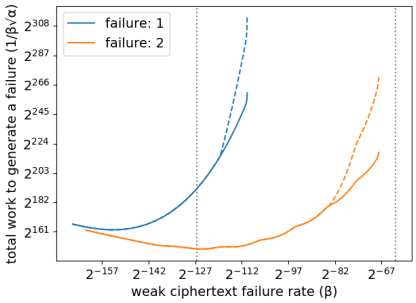


(a) Cost to find a failure vs weak ciphertext failure rate for $q_{limit} = 2^{64}$ and $T^{(0)} = 2^{64}$.

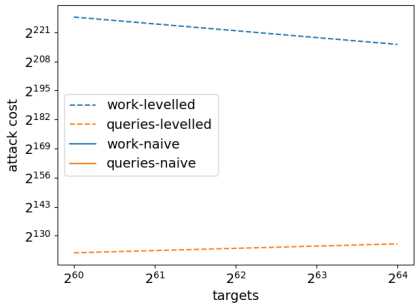


(b) Attack cost in function of targets $T^{(0)}$ for $q_{limit} = 2^{64}$.

Fig. 6. Saber

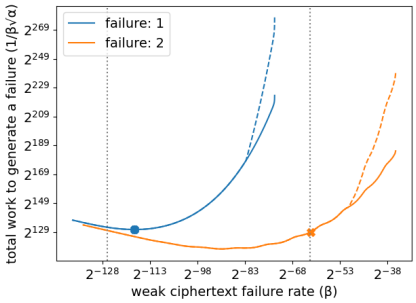


(a) Cost to find a failure vs weak ciphertext failure rate for $q_{limit} = 2^{64}$ and $T^{(0)} = 2^{64}$.

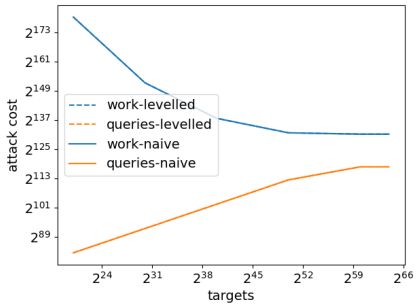


(b) Attack cost in function of targets $T^{(0)}$ for $q_{limit} = 2^{64}$.

Fig. 7. FireSaber

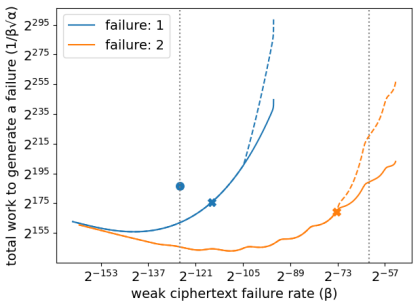


(a) Cost to find a failure vs weak ciphertext failure rate for $q_{limit} = 2^{64}$ and $T^{(0)} = 2^{64}$.

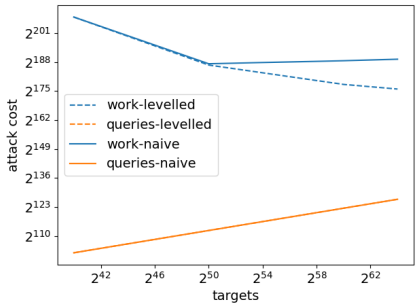


(b) Attack cost in function of targets $T^{(0)}$ for $q_{limit} = 2^{64}$.

Fig. 8. Kyber512

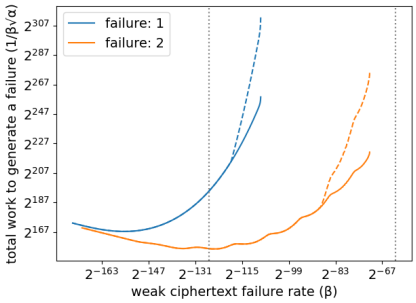


(a) Cost to find a failure vs weak ciphertext failure rate for $q_{limit} = 2^{64}$ and $T^{(0)} = 2^{64}$.

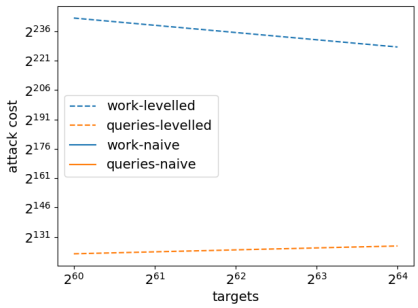


(b) Attack cost in function of targets $T^{(0)}$ for $q_{limit} = 2^{64}$.

Fig. 9. Kyber768



(a) Cost to find a failure vs weak ciphertext failure rate for $q_{limit} = 2^{64}$ and $T^{(0)} = 2^{64}$.



(b) Attack cost in function of targets $T^{(0)}$ for $q_{limit} = 2^{64}$.

Fig. 10. Kyber1024