Does the Dual-Sieve Attack on LWE even Work?

Léo Ducas[1,2], **Ludo N. Pulles**[1]

22 August 2023

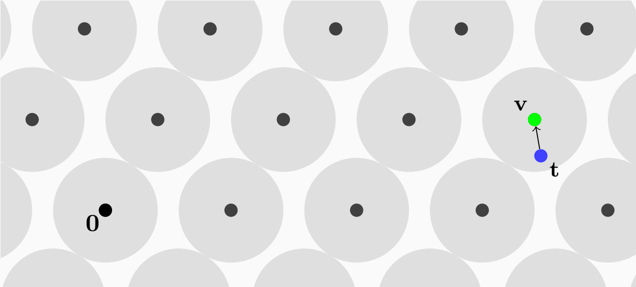
[1]Cryptology Group, CWI, Amsterdam, [2]Mathematical Institute, Leiden University

Hard problem in lattice-based crypto

The security of lattice-based cryptoschemes, like KYBER and DILITHIUM, depends on the hardness of the *Bounded Distance Decoding* (BDD) problem.

Hard problem in lattice-based crypto

The security of lattice-based cryptoschemes, like KYBER and DILITHIUM, depends on the hardness of the *Bounded Distance Decoding* (BDD) problem.



BDD: Given a "noisy" lattice vector, recover the lattice vector.

Lattice attacks against BDD

There are two types of lattice attacks against BDD:

Primal attack

I. Embed Λ and $\mathbf{t}$ into a lattice, where the shortest vector is shorter than expected.
II. Solve unique-SVP instance by lattice reduction.

Dual attack

I. Construct a function that distinguishes between BDD targets and uniform targets,
II. Using this distinguisher, guess and determine part of the secret.

Lattice attacks against BDD

There are two types of lattice attacks against BDD:

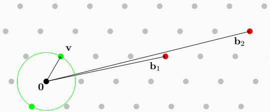
Primal attack

I. Embed Λ and $\mathbf{t}$ into a lattice, where the shortest vector is shorter than expected.
II. Solve unique-SVP instance by lattice reduction.

Dual attack

I. Construct a function that distinguishes between BDD targets and uniform targets,
II. Using this distinguisher, guess and determine part of the secret.

Shortest vector problem (SVP)



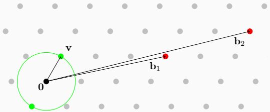
Lattice attacks against BDD

There are two types of lattice attacks against BDD:

Primal attack

I. Embed Λ and $\mathbf{t}$ into a lattice, where the shortest vector is shorter than expected.
II. Solve unique-SVP instance by lattice reduction.

Shortest vector problem (SVP)



Dual attack

I. Construct a function that distinguishes between BDD targets and uniform targets,
II. Using this distinguisher, guess and determine part of the secret.

Lattice attacks against BDD

There are two types of lattice attacks against BDD:

Primal attack

I. Embed Λ and $\mathbf{t}$ into a lattice, where the shortest vector is shorter than expected.
II. Solve unique-SVP instance by lattice reduction.

Dual attack

I. Construct a function that distinguishes between BDD targets and uniform targets,
II. Using this distinguisher, guess and determine part of the secret.

Theoretically and experimentally well-studied.

Lattice attacks against BDD

There are two types of lattice attacks against BDD:

Primal attack

I. Embed Λ and $\mathbf{t}$ into a lattice, where the shortest vector is shorter than expected.
II. Solve unique-SVP instance by lattice reduction.

Theoretically and experimentally well-studied.

Dual attack

I. Construct a function that distinguishes between BDD targets and uniform targets,
II. Using this distinguisher, guess and determine part of the secret.

Received little experimental attention so far.

Recent improvements to the dual attack

Beginning of the dual attack

[AR'05][1]: use short dual vectors for distinguishing.

Recent developments

- [ADPS'16][2]: A lattice sieve yields many short dual vectors.*
- [GJ'21][3]: Speed up evaluating distinguisher with a Fast Fourier Transform (FFT).*
- [MAT'22][4]: Improves dual attack with modulus switching technique.*

*: specific to LWE problem.

[1] Aharonov & Regev. "Lattice problems in NP $\cap$ coNP". JACM '05.
[2] Alkim, Ducas, Pöppelmann & Schwabe. "Post-quantum Key Exchange – A New Hope". USENIX '16.
[3] Guo & Johansson. "Faster Dual Lattice Attacks for Solving LWE with Applications to CRYSTALS". AC'21.
[4] MATZOV. "Report on the Security of LWE: Improved Dual Lattice Attack". Zenodo #6493704

Recent improvements to the dual attack

Beginning of the dual attack

[AR'05][1]: use short dual vectors for distinguishing.

Recent developments

- [ADPS'16][2]: A lattice sieve yields many short dual vectors.*
- [GJ'21][3]: Speed up evaluating distinguisher with a Fast Fourier Transform (FFT).*
- [MAT'22][4]: Improves dual attack with modulus switching technique.*

*: specific to LWE problem.

[1] Aharonov & Regev. "Lattice problems in NP $\cap$ coNP". JACM '05.
[2] Alkim, Ducas, Pöppelmann & Schwabe. "Post-quantum Key Exchange – A New Hope". USENIX '16.
[3] Guo & Johansson. "Faster Dual Lattice Attacks for Solving LWE with Applications to CRYSTALS". AC'21.
[4] MATZOV. "Report on the Security of LWE: Improved Dual Lattice Attack". Zenodo #6493704

Recent improvements to the dual attack

Beginning of the dual attack

[AR'05][1]: use short dual vectors for distinguishing.

Recent developments

- [ADPS'16][2]: A lattice sieve yields many short dual vectors.*
- [GJ'21][3]: Speed up evaluating distinguisher with a Fast Fourier Transform (FFT).*
- [MAT'22][4]: Improves dual attack with modulus switching technique.*

*: specific to LWE problem.

[1] Aharonov & Regev. "Lattice problems in NP $\cap$ coNP". JACM '05.
[2] Alkim, Ducas, Pöppelmann & Schwabe. "Post-quantum Key Exchange – A New Hope". USENIX '16.
[3] Guo & Johansson. "Faster Dual Lattice Attacks for Solving LWE with Applications to CRYSTALS". AC'21.
[4] MATZOV. "Report on the Security of LWE: Improved Dual Lattice Attack". Zenodo #6493704

Recent improvements to the dual attack

Beginning of the dual attack

[AR'05][1]: use short dual vectors for distinguishing.

Recent developments

- [ADPS'16][2]: A lattice sieve yields many short dual vectors.*
- [GJ'21][3]: Speed up evaluating distinguisher with a Fast Fourier Transform (FFT).*
- [MAT'22][4]: Improves dual attack with modulus switching technique.*

*: specific to LWE problem.

[1] Aharonov & Regev. "Lattice problems in NP ∩ coNP". JACM '05.
[2] Alkim, Ducas, Pöppelmann & Schwabe. "Post-quantum Key Exchange – A New Hope". USENIX '16.
[3] Guo & Johansson. "Faster Dual Lattice Attacks for Solving LWE with Applications to CRYSTALS". AC'21.
[4] MATZOV. "Report on the Security of LWE: Improved Dual Lattice Attack". Zenodo #6493704

Recent improvements to the dual attack

Beginning of the dual attack

[AR'05][1]: use short dual vectors for distinguishing.

Recent developments

- [ADPS'16][2]: A lattice sieve yields many short dual vectors.*
- [GJ'21][3]: Speed up evaluating distinguisher with a Fast Fourier Transform (FFT).*
- [MAT'22][4]: Improves dual attack with modulus switching technique.*

*: specific to LWE problem.

[1] Aharonov & Regev. "Lattice problems in NP $\cap$ coNP". JACM '05.
[2] Alkim, Ducas, Pöppelmann & Schwabe. "Post-quantum Key Exchange – A New Hope". USENIX '16.
[3] Guo & Johansson. "Faster Dual Lattice Attacks for Solving LWE with Applications to CRYSTALS". AC'21.
[4] MATZOV. "Report on the Security of LWE: Improved Dual Lattice Attack". Zenodo #6493704

Our contributions

Generalization of FFT trick to BDD

- Provides geometric insight!
- Allows further improvements.

A heuristic used in earlier works leads to two contradictions

The distinguisher does not work as well as predicted.

Experimental confirmation

Derived cryptanalysis overestimates the success probability of attacks.

Our contributions

Generalization of FFT trick to BDD

- Provides geometric insight!
- Allows further improvements.

A heuristic used in earlier works leads to two contradictions

The distinguisher does not work as well as predicted.

Experimental confirmation

Derived cryptanalysis overestimates the success probability of attacks.

Our contributions

Generalization of FFT trick to BDD

- Provides geometric insight!
- Allows further improvements.

A heuristic used in earlier works leads to two contradictions

The distinguisher does not work as well as predicted.

Experimental confirmation

Derived cryptanalysis overestimates the success probability of attacks.

Generalization of FFT trick to BDD

Generalization of FFT trick to BDD

α-BDD search problem

Given: lattice Λ and target $\mathbf{t} \in \mathbb{R}^n$, such that $\mathbf{t} = \mathbf{v} + \mathbf{e}$ with $\mathbf{v} \in \Lambda$ and $\|\mathbf{e}\| \approx \alpha\lambda_1$,

Problem: recover $\mathbf{v}$.

(λ_1 is length of shortest vector)

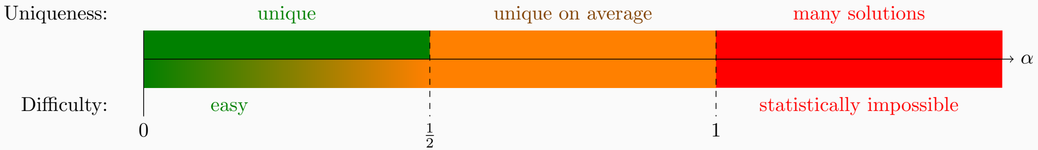
Generalization of FFT trick to BDD

α-BDD search problem

Given: lattice Λ and target $\mathbf{t} \in \mathbb{R}^n$, such that $\mathbf{t} = \mathbf{v} + \mathbf{e}$ with $\mathbf{v} \in \Lambda$ and $\|\mathbf{e}\| \approx \alpha\lambda_1$,

Problem: recover $\mathbf{v}$.

(λ_1 is length of shortest vector)



Dual lattice

Dual lattice

The *dual lattice* $\Lambda^\vee$ consists of all points $\mathbf{w}$ such that $\langle \mathbf{w}, \Lambda \rangle \subseteq \mathbb{Z}$.

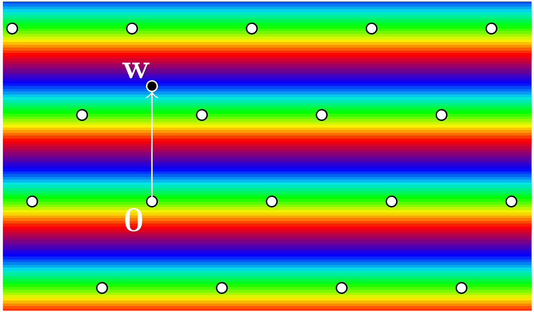
A dual vector $\mathbf{w}$ corresponds to the *character* $\chi_{\mathbf{w}}$:

Dual lattice

Dual lattice

The *dual lattice* $\Lambda^\vee$ consists of all points $\mathbf{w}$ such that $\langle \mathbf{w}, \Lambda \rangle \subseteq \mathbb{Z}$.

A dual vector $\mathbf{w}$ corresponds to the *character* $\chi_{\mathbf{w}}$:



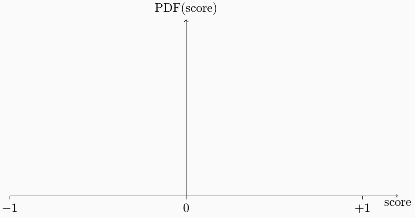
Distinguish based on score

Consider the score function:

$$f_{\mathbf{w}}(\mathbf{t}) = \cos\left(2\pi \langle \mathbf{w}, \mathbf{t} \rangle\right),$$

- $\mathbf{t} \in \Lambda \implies$ score $= 1$,
- $\mathbf{t}$ *close* to Λ and $\mathbf{w}$ short $\implies$ score ≈ 1,
- $\mathbf{t}$ uniform from torus $\mathbb{R}^n/\Lambda$
  $\implies$ expected score is 0.

⚠ If score ≈ 1, $\mathbf{t}$ can be uniform!



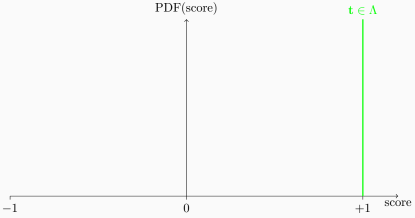
Distinguish based on score

Consider the score function:

$$f_{\mathbf{w}}(\mathbf{t}) = \cos(2\pi \langle \mathbf{w}, \mathbf{t} \rangle),$$

- $\mathbf{t} \in \Lambda \implies \text{score} = 1$,
- $\mathbf{t}$ *close* to Λ and $\mathbf{w}$ short $\implies \text{score} \approx 1$,
- $\mathbf{t}$ uniform from torus $\mathbb{R}^n/\Lambda$
  $\implies$ expected score is 0.

⚠ If score ≈ 1, $\mathbf{t}$ can be uniform!



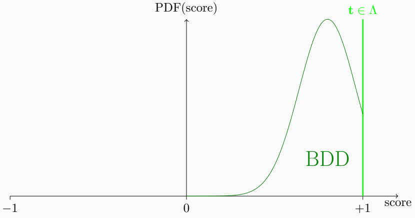
Distinguish based on score

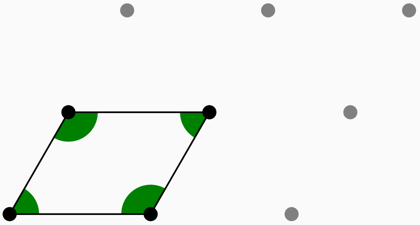
Consider the score function:

$$f_{\mathbf{w}}(\mathbf{t}) = \cos(2\pi \langle \mathbf{w}, \mathbf{t} \rangle),$$

- $\mathbf{t} \in \Lambda \implies$ score $= 1$,
- $\mathbf{t}$ *close* to Λ and $\mathbf{w}$ short $\implies$ score ≈ 1,
- $\mathbf{t}$ uniform from torus $\mathbb{R}^n/\Lambda$
  $\implies$ expected score is 0.

⚠ If score ≈ 1, $\mathbf{t}$ can be uniform!





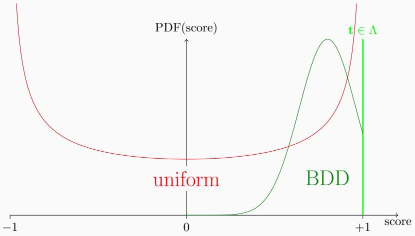
Distinguish based on score

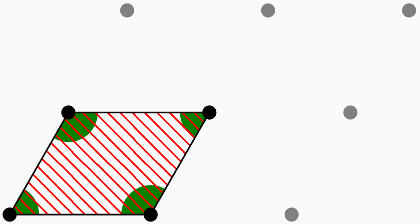
Consider the score function:

$$f_{\mathbf{w}}(\mathbf{t}) = \cos\left(2\pi \langle \mathbf{w}, \mathbf{t} \rangle\right),$$

- $\mathbf{t} \in \Lambda \implies$ score $= 1$,
- $\mathbf{t}$ *close* to Λ and $\mathbf{w}$ short $\implies$ score ≈ 1,
- $\mathbf{t}$ uniform from torus $\mathbb{R}^n/\Lambda$
  $\implies$ expected score is 0.

⚠ If score ≈ 1, $\mathbf{t}$ can be uniform!





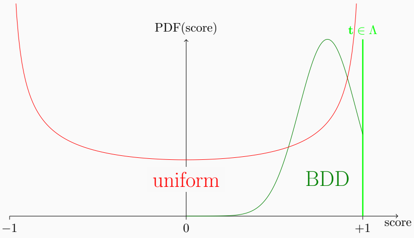
Distinguish based on score

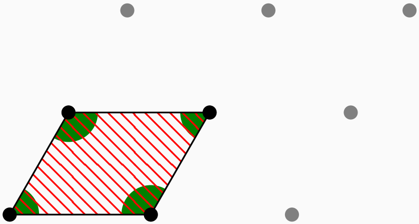
Consider the score function:

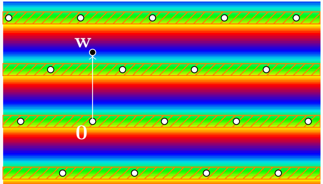
$$f_{\mathbf{w}}(\mathbf{t}) = \cos\left(2\pi \langle \mathbf{w}, \mathbf{t} \rangle\right),$$

- $\mathbf{t} \in \Lambda \implies$ score $= 1$,
- $\mathbf{t}$ *close* to Λ and $\mathbf{w}$ short $\implies$ score ≈ 1,
- $\mathbf{t}$ uniform from torus $\mathbb{R}^n/\Lambda$
  $\implies$ expected score is 0.

⚠ If score ≈ 1, $\mathbf{t}$ can be uniform!







Dual-Sieve distinguisher

To improve the distinguisher, we use all $(4/3)^{n/2}$ short dual vectors from a lattice sieve:

$$f_{\mathcal{W}}(\mathbf{t}) = \sum_{\mathbf{w} \in \mathcal{W}} f_{\mathbf{w}}(\mathbf{t}) = \sum_{\mathbf{w} \in \mathcal{W}} \cos\Big(2\pi \langle \mathbf{w}, \mathbf{t} \rangle \Big).$$

Independence Heuristic used in [GJ'21], [MAT'22] and more

Given a set of dual vectors $\mathcal{W}$ from a sieve,
the scores $\cos(2\pi \langle \mathbf{w}, \mathbf{t} \rangle)$ are mutually independent.

Dual-Sieve distinguisher

To improve the distinguisher, we use all $(4/3)^{n/2}$ short dual vectors from a lattice sieve:

$$f_{\mathcal{W}}(\mathbf{t}) = \sum_{\mathbf{w}\in\mathcal{W}} f_{\mathbf{w}}(\mathbf{t}) = \sum_{\mathbf{w}\in\mathcal{W}} \cos\Big(2\pi \langle \mathbf{w}, \mathbf{t} \rangle\Big).$$

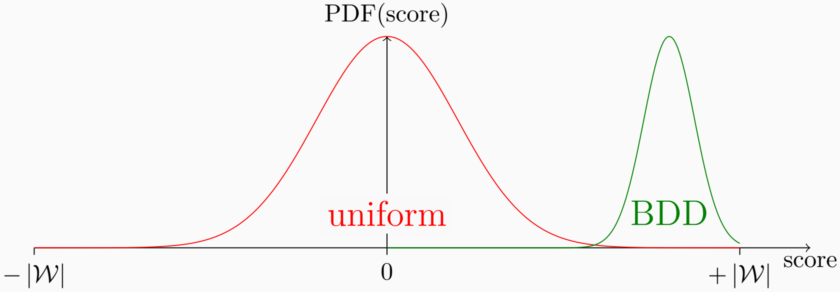
Independence Heuristic used in [GJ'21], [MAT'22] and more

Given a set of dual vectors $\mathcal{W}$ from a sieve,
the scores $\cos(2\pi \langle \mathbf{w}, \mathbf{t} \rangle)$ are mutually independent.

Dual-Sieve distinguisher

Independence Heuristic used in [GJ'21], [MAT'22] and more

Given a set of dual vectors $\mathcal{W}$ from a sieve,
the scores $\cos(2\pi \langle \mathbf{w}, \mathbf{t} \rangle)$ are mutually independent.

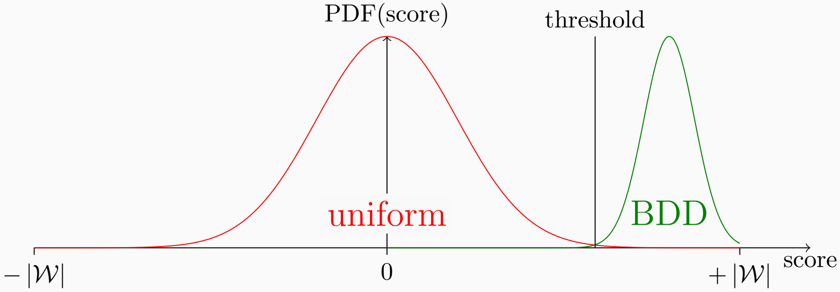


$\mathbf{t}$ uniform mod Λ $\xrightarrow{\text{w.h.p.}}$ $f_{\mathcal{W}}(\mathbf{t}) \approx 0$, $\mathbf{t}$ BDD target $\xrightarrow{\text{w.h.p.}}$ $f_{\mathcal{W}}(\mathbf{t})$ large.

Dual-Sieve distinguisher

Independence Heuristic used in [GJ'21], [MAT'22] and more

Given a set of dual vectors $\mathcal{W}$ from a sieve,
the scores $\cos(2\pi \langle \mathbf{w}, \mathbf{t} \rangle)$ are mutually independent.



$\mathbf{t}$ uniform mod Λ $\xrightarrow{\text{w.h.p.}}$ $f_{\mathcal{W}}(\mathbf{t}) \approx 0$, $\mathbf{t}$ BDD target $\xrightarrow{\text{w.h.p.}}$ $f_{\mathcal{W}}(\mathbf{t})$ large.

Search-BDD $\Longrightarrow$ Decision-BDD

- Take a sparsified sublattice $\Lambda' \subset \Lambda$,
- Use the distinguisher $f_{\mathcal{W}}$ for Λ',
- For $\mathbf{t} = \mathbf{v} + \mathbf{e}$ and a guess $\mathbf{g} \in \Lambda$,

$$\begin{aligned} \mathbf{v} \in \mathbf{g} + \Lambda' \quad &\Longleftrightarrow \quad \mathbf{t} \text{ close to } \mathbf{g} + \Lambda' \\ &\Longleftrightarrow \quad \mathbf{t} - \mathbf{g} \text{ close to } \Lambda' \\ &\overset{\text{w.h.p}}{\Longleftrightarrow} \quad \text{distinguisher marks } \mathbf{t} - \mathbf{g} \text{ as BDD.} \end{aligned}$$

Search-BDD $\implies$ Decision-BDD

- Take a sparsified sublattice $\Lambda' \subset \Lambda$,
- Use the distinguisher $f_{\mathcal{W}}$ for Λ',
- For $\mathbf{t} = \mathbf{v} + \mathbf{e}$ and a guess $\mathbf{g} \in \Lambda$,

$$\begin{aligned} \mathbf{v} \in \mathbf{g} + \Lambda' \quad &\iff \quad \mathbf{t} \text{ close to } \mathbf{g} + \Lambda' \\ &\iff \quad \mathbf{t} - \mathbf{g} \text{ close to } \Lambda' \\ &\xLeftrightarrow{\text{w.h.p}} \quad \text{distinguisher marks } \mathbf{t} - \mathbf{g} \text{ as BDD.} \end{aligned}$$

Search-BDD $\implies$ Decision-BDD

- Take a sparsified sublattice $\Lambda' \subset \Lambda$,
- Use the distinguisher $f_{\mathcal{W}}$ for Λ',
- For $\mathbf{t} = \mathbf{v} + \mathbf{e}$ and a guess $\mathbf{g} \in \Lambda$,

$$\begin{aligned} \mathbf{v} \in \mathbf{g} + \Lambda' \quad &\iff \quad \mathbf{t} \text{ close to } \mathbf{g} + \Lambda' \\ &\iff \quad \mathbf{t} - \mathbf{g} \text{ close to } \Lambda' \\ &\overset{\text{w.h.p}}{\iff} \quad \text{distinguisher marks } \mathbf{t} - \mathbf{g} \text{ as BDD.} \end{aligned}$$

Dual-Sieve attack

DualAttack($\Lambda, \mathbf{t}$):

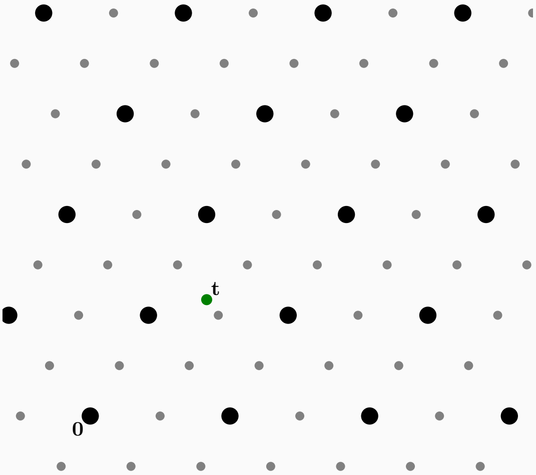
1. Pick a sublattice $\Lambda' \subset \Lambda$,
2. Run a lattice sieve on $(\Lambda')^\vee$ to acquire dual vectors $\mathcal{W}$,
3. Write Λ as union of Λ'-cosets:

$$\Lambda = \bigcup_{\mathbf{g}} (\Lambda' + \mathbf{g}) \quad (\mathbf{g} \in \Lambda),$$

4. Pick $\Lambda' + \mathbf{g}$ that maximizes $f_{\mathcal{W}}(\mathbf{t} - \mathbf{g})$.

– We recovered part of the secret: $\mathbf{g}$.

– The new BDD instance is easier.



Dual-Sieve attack

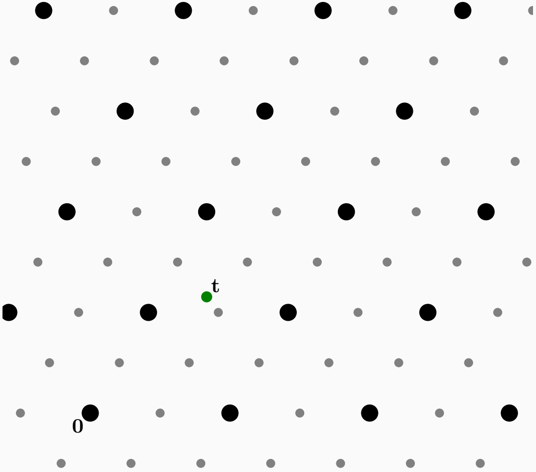
DualAttack($\Lambda, \mathbf{t}$):

1. Pick a sublattice $\Lambda' \subset \Lambda$,
2. Run a lattice sieve on $(\Lambda')^\vee$ to acquire dual vectors $\mathcal{W}$,
3. Write Λ as union of Λ'-cosets:

$$\Lambda = \bigcup_{\mathbf{g}} (\Lambda' + \mathbf{g}) \quad (\mathbf{g} \in \Lambda),$$

4. Pick $\Lambda' + \mathbf{g}$ that maximizes $f_{\mathcal{W}}(\mathbf{t} - \mathbf{g})$.

- We recovered part of the secret: $\mathbf{g}$.
- The new BDD instance is easier.



Dual-Sieve attack

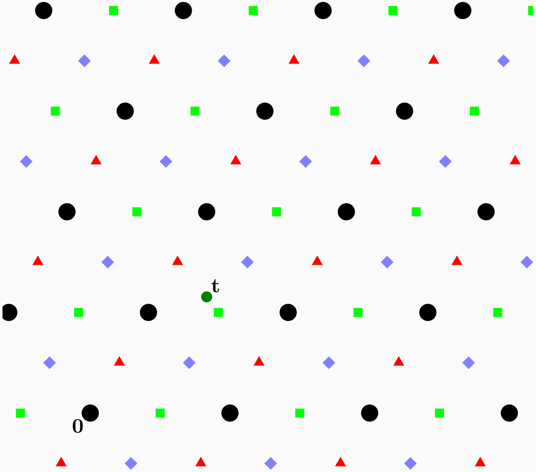
DualAttack($\Lambda, \mathbf{t}$):

1. Pick a sublattice $\Lambda' \subset \Lambda$,
2. Run a lattice sieve on $(\Lambda')^\vee$ to acquire dual vectors $\mathcal{W}$,
3. Write Λ as union of Λ'-cosets:

$$\Lambda = \bigcup_{\mathbf{g}} (\Lambda' + \mathbf{g}) \quad (\mathbf{g} \in \Lambda),$$

4. Pick $\Lambda' + \mathbf{g}$ that maximizes $f_{\mathcal{W}}(\mathbf{t} - \mathbf{g})$.

- We recovered part of the secret: $\mathbf{g}$.
- The new BDD instance is easier.



Dual-Sieve attack

DualAttack($\Lambda, \mathbf{t}$):

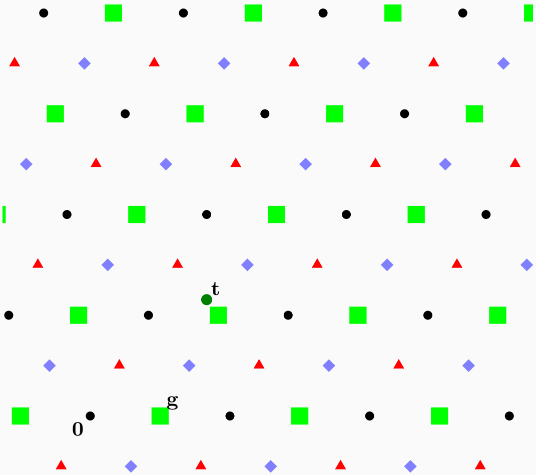
1. Pick a sublattice $\Lambda' \subset \Lambda$,
2. Run a lattice sieve on $(\Lambda')^\vee$ to acquire dual vectors $\mathcal{W}$,
3. Write Λ as union of Λ'-cosets:

$$\Lambda = \bigcup_{\mathbf{g}} (\Lambda' + \mathbf{g}) \quad (\mathbf{g} \in \Lambda),$$

4. Pick $\Lambda' + \mathbf{g}$ that maximizes $f_{\mathcal{W}}(\mathbf{t} - \mathbf{g})$.

– We recovered part of the secret: $\mathbf{g}$.

– The new BDD instance is easier.



Dual-Sieve attack

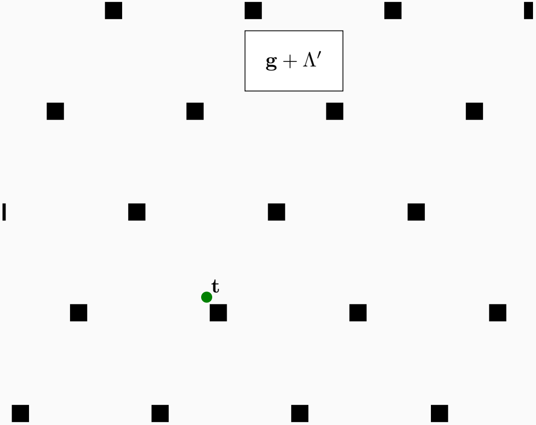
DualAttack($\Lambda, \mathbf{t}$):

1. Pick a sublattice $\Lambda' \subset \Lambda$,
2. Run a lattice sieve on $(\Lambda')^\vee$ to acquire dual vectors $\mathcal{W}$,
3. Write Λ as union of Λ'-cosets:

$$\Lambda = \bigcup_{\mathbf{g}} (\Lambda' + \mathbf{g}) \quad (\mathbf{g} \in \Lambda),$$

4. Pick $\Lambda' + \mathbf{g}$ that maximizes $f_{\mathcal{W}}(\mathbf{t} - \mathbf{g})$.

- We recovered part of the secret: $\mathbf{g}$.
- The new BDD instance is easier.



General Dual-Sieve-FFT attack

Naïvely, computing $f_{\mathcal{W}}(\mathbf{t} - \mathbf{g})$, takes time $|\mathcal{W}|$ per guess.

Fast Fourier Transform

Computes scores for T many guesses in amortized time $\log_2(T)$ per guess!

Benefits of geometric insights

- Attack works for any lattice Λ and sparsification Λ', not only q-ary lattices.
- Flexibility in sparsification $\implies$ better attack.

General Dual-Sieve-FFT attack

Naïvely, computing $f_{\mathcal{W}}(\mathbf{t} - \mathbf{g})$, takes time $|\mathcal{W}|$ per guess.

Fast Fourier Transform

Computes scores for T many guesses in amortized time $\log_2(T)$ per guess!

Benefits of geometric insights

- Attack works for any lattice Λ and sparsification Λ', not only q-ary lattices.
- Flexibility in sparsification $\implies$ better attack.

General Dual-Sieve-FFT attack

Naïvely, computing $f_{\mathcal{W}}(\mathbf{t}-\mathbf{g})$, takes time $|\mathcal{W}|$ per guess.

Fast Fourier Transform

Computes scores for T many guesses in amortized time $\log_2(T)$ per guess!

Benefits of geometric insights

- Attack works for any lattice Λ and sparsification Λ', not only q-ary lattices.
- Flexibility in sparsification $\Longrightarrow$ better attack.

General Dual-Sieve-FFT attack

Naïvely, computing $f_{\mathcal{W}}(\mathbf{t}-\mathbf{g})$, takes time $|\mathcal{W}|$ per guess.

Fast Fourier Transform

Computes scores for T many guesses in amortized time $\log_2(T)$ per guess!

Benefits of geometric insights

- Attack works for any lattice Λ and sparsification Λ', not only q-ary lattices.
- Flexibility in sparsification $\implies$ better attack.

Independence Heuristic leads to two contradictions

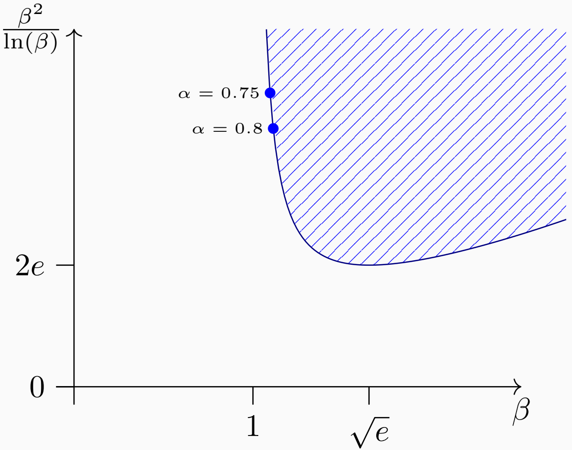
#1: Distinguishing the indistinguishable (1/2)

[LW'21][5]: Distinguishing a single target under Independence Heuristic

For *any* $\alpha > 0$, take $\beta > 1$ satisfying

$$\frac{\beta^2}{\ln(\beta)} = \frac{e^2}{\alpha^2}.$$

Given the shortest β^n dual vectors, $f_{\mathcal{W}}(\mathbf{t})$ distinguishes between a uniform and a α-BDD target[6] with success probability 99%.



[5] Laarhoven and Walter. "Dual lattice attacks for closest vector problems (with preprocessing)". CT-RSA 2021.

[6] Recall: $\mathbf{t} = \mathbf{v} + \mathbf{e}$ such that $\mathbf{v} \in \Lambda$ and $\|\mathbf{e}\| \approx \alpha\lambda_1$.

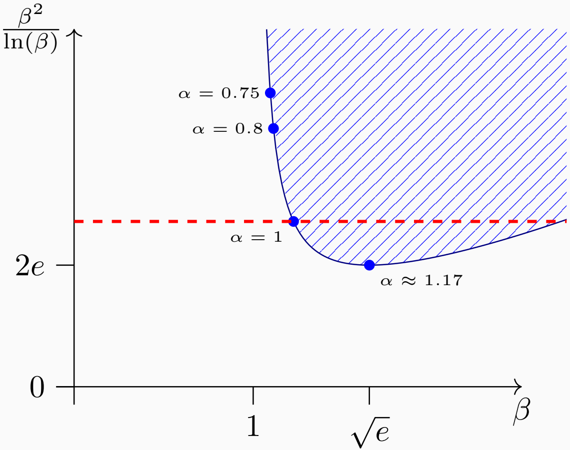
#1: Distinguishing the indistinguishable (1/2)

[LW'21][5]: Distinguishing a single target under Independence Heuristic

For *any* $\alpha > 0$, take $\beta > 1$ satisfying

$$\frac{\beta^2}{\ln(\beta)} = \frac{e^2}{\alpha^2}.$$

Given the shortest β^n dual vectors, $f_{\mathcal{W}}(\mathbf{t})$ distinguishes between a uniform and a α-BDD target[6] with success probability 99%.



[5] Laarhoven and Walter. "Dual lattice attacks for closest vector problems (with preprocessing)". CT-RSA 2021.

[6] Recall: $\mathbf{t} = \mathbf{v} + \mathbf{e}$ such that $\mathbf{v} \in \Lambda$ and $\|\mathbf{e}\| \approx \alpha\lambda_1$.

#1: Distinguishing the indistinguishable (1/2)

[LW'21][5]: Distinguishing a single target under Independence Heuristic

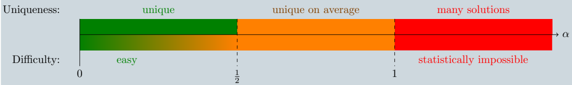
For *any* $\alpha > 0$, take $\beta > 1$ satisfying

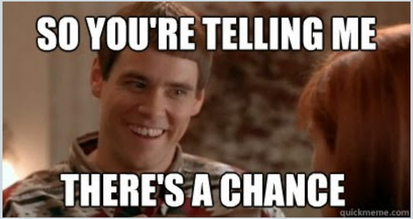
$$\frac{\beta^2}{\ln(\beta)} = \frac{e^2}{\alpha^2}.$$

Given the shortest β^n dual vectors, $f_{\mathcal{W}}(\mathbf{t})$ distinguishes between a uniform and a α-BDD target[6] with success probability 99%.

Can we still distinguish when $\alpha > 1$?

Hardness of Search-BDD:





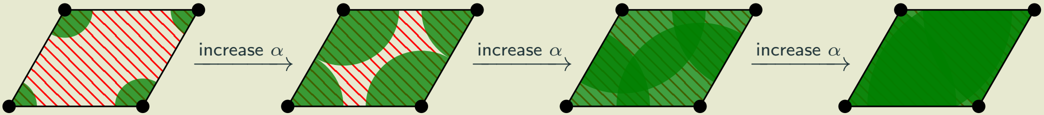
[5] Laarhoven and Walter. "Dual lattice attacks for closest vector problems (with preprocessing)". CT-RSA 2021.

[6] Recall: $\mathbf{t} = \mathbf{v} + \mathbf{e}$ such that $\mathbf{v} \in \Lambda$ and $\|\mathbf{e}\| \approx \alpha\lambda_1$.

#1: Distinguishing the indistinguishable (2/2)

Indistinguishability Theorem ("Smoothing bound")

[DDRT'22][7]: In a random lattice, errors uniform from the ball of radius $\alpha\lambda_1$ become *statistically indistinguishable* from uniform errors in $\mathbb{R}^n/\Lambda$ when $\alpha > 1$.



In particular, no adversary (having unbounded runtime) can ever succeed distinguishing with probability more than $\frac{1}{2} + \alpha^{-n/2}$.

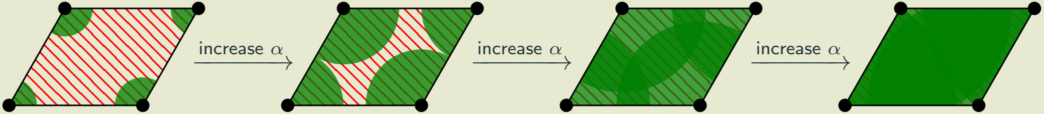
$\lightning$

[7] Debris-Alazard, Ducas, Resch & Tillich. "Smoothing codes and lattices: Systematic Study and New Bounds".

#1: Distinguishing the indistinguishable (2/2)

Indistinguishability Theorem ("Smoothing bound")

[DDRT'22][7]: In a random lattice, errors uniform from the ball of radius $\alpha\lambda_1$ become *statistically indistinguishable* from uniform errors in $\mathbb{R}^n/\Lambda$ when $\alpha > 1$.



In particular, no adversary (having unbounded runtime) can ever succeed distinguishing with probability more than $\frac{1}{2} + \alpha^{-n/2}$.

↯

[7] Debris-Alazard, Ducas, Resch & Tillich. "Smoothing codes and lattices: Systematic Study and New Bounds".

#2: Candidates Closer than the Solution (1/3)

Distinguishing α-BDD among many uniforms

Given: T random uniform targets and a single α-BDD target, shuffled.

Return: the BDD target.

Recall from Dual-Sieve attack ([GJ'21], [MAT'22] & more):

4. *Pick $\Lambda' + \mathbf{g}$ that maximizes $f_{\mathcal{W}}(\mathbf{t} - \mathbf{g})$.*

Limit on T

Question: What is biggest T for which Dual-Sieve attack works with 99% probability?

#2: Candidates Closer than the Solution (1/3)

Distinguishing α-BDD among many uniforms

Given: T random uniform targets and a single α-BDD target, shuffled.

Return: the BDD target.

By Dimitris Vetsikas @Pixabay

Recall from Dual-Sieve attack ([GJ'21], [MAT'22] & more):

4. *Pick* $\Lambda' + \mathbf{g}$ *that maximizes* $f_{\mathcal{W}}(\mathbf{t} - \mathbf{g})$.

Limit on T

Question: What is biggest T for which Dual-Sieve attack works with 99% probability?

#2: Candidates Closer than the Solution (1/3)

Distinguishing α-BDD among many uniforms

Given: T random uniform targets and a single α-BDD target, shuffled.

Return: the BDD target.

By Dimitris Vetsikas @Pixabay

Recall from Dual-Sieve attack ([GJ'21], [MAT'22] & more):

4. *Pick* $\Lambda' + \mathbf{g}$ *that maximizes* $f_{\mathcal{W}}(\mathbf{t} - \mathbf{g})$.

Limit on T

Question: What is biggest T for which Dual-Sieve attack works with 99% probability?

#2: Candidates Closer than the Solution (2/3)

Distinguishing failures

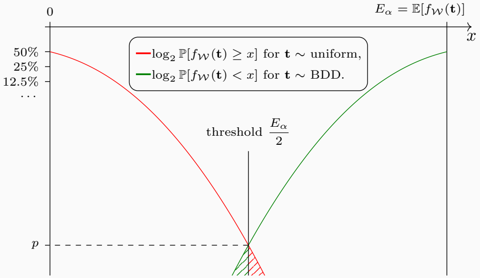
Failure $\Longrightarrow$ a) α-BDD target has low score, **or**
b) *any* of the T uniform targets has high score.

Claim [GJ'21], [MAT'22] under Independence Heuristic:

Classic tail bound: $p \le e^{-E_\alpha^2/|\mathcal{W}|}$.

For $\alpha < 0.89$: $E_\alpha^2 / |\mathcal{W}| \sim e^{Cn}$, as $n \to \infty$.

$\Longrightarrow$ Dual-Sieve attack works for $T = \frac{1}{p} = e^{e^{Cn}}$?!



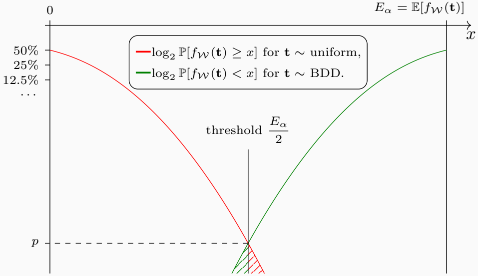
#2: Candidates Closer than the Solution (2/3)

Distinguishing failures

Failure $\implies$ a) α-BDD target has low score, **or** b) *any* of the T uniform targets has high score.

Claim [GJ'21], [MAT'22] under Independence Heuristic:

Classic tail bound: $p \leq e^{-E_\alpha^2/|\mathcal{W}|}$.



For $\alpha < 0.89$: $E_\alpha^2 / |\mathcal{W}| \sim e^{Cn}$, as $n \to \infty$.

$\implies$ Dual-Sieve attack works for $T = \frac{1}{p} = e^{e^{Cn}}$?!

#2: Candidates Closer than the Solution (2/3)

Distinguishing failures

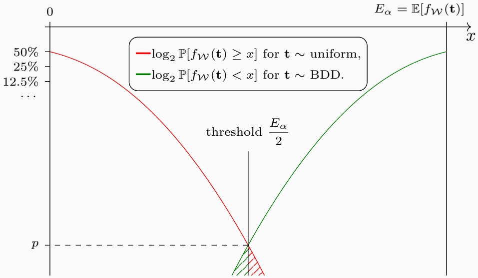
Failure $\implies$ a) α-BDD target has low score, **or** b) *any* of the T uniform targets has high score.

Claim [GJ'21], [MAT'22] under Independence Heuristic:

Classic tail bound: $p \leq e^{-E_\alpha^2/|\mathcal{W}|}$.

For $\alpha < 0.89$: $\quad E_\alpha^2/\,|\mathcal{W}| \sim e^{Cn}$, as $n \to \infty$.

$\implies$ Dual-Sieve attack works for $T = \frac{1}{p} = e^{e^{Cn}}$?!



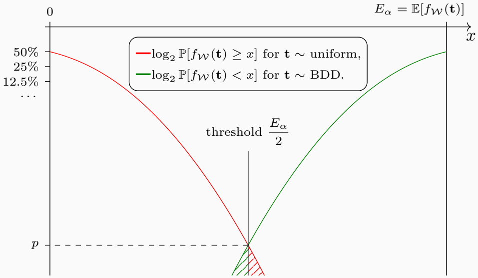
#2: Candidates Closer than the Solution (2/3)

Distinguishing failures

Failure $\implies$ a) α-BDD target has low score, **or**
b) *any* of the T uniform targets has high score.

Claim [GJ'21], [MAT'22] under Independence Heuristic:

Classic tail bound: $p \leq e^{-E_\alpha^2/|\mathcal{W}|}$.



For $\alpha < 0.89$: $E_\alpha^2 / |\mathcal{W}| \sim e^{Cn}$, as $n \to \infty$.

$\implies$ Dual-Sieve attack works for $T = \frac{1}{p} = e^{e^{Cn}}$?!

Distinguishing failures

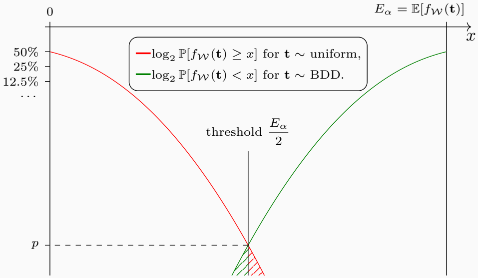
Failure $\Longrightarrow$ a) α-BDD target has low score, **or** b) *any* of the T uniform targets has high score.

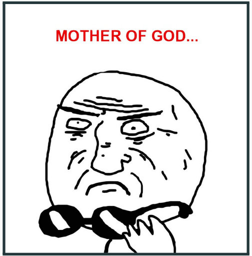
Claim [GJ'21], [MAT'22] under Independence Heuristic:

Classic tail bound: $p \leq e^{-E_\alpha^2/|\mathcal{W}|}$.

For $\alpha < 0.89$: $E_\alpha^2 / |\mathcal{W}| \sim e^{Cn}$, as $n \to \infty$.

$\Longrightarrow$ Dual-Sieve attack works for $T = \frac{1}{p} = e^{e^{Cn}}$?!





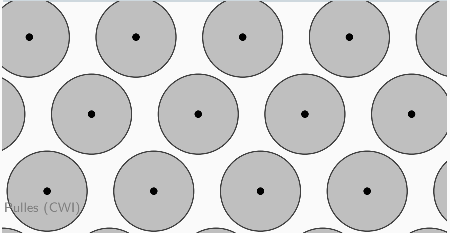
#2: Candidates Closer than the Solution (3/3)

Closeness Lemma

Given a random lattice Λ *and* $r < \frac{1}{2}$*, a uniform target*

$$\mathbf{t} \xleftarrow{\$} \mathbb{R}^n/\Lambda,$$

is at most $r\lambda_1$ *away from a lattice point with probability* r^n*.*



Geometric contradiction

- Given $T \gg \alpha^{-n}$ uniform targets, there is one of them *closer to* Λ than the α-BDD target.
- This target has a *higher score* than the α-BDD target!

↯

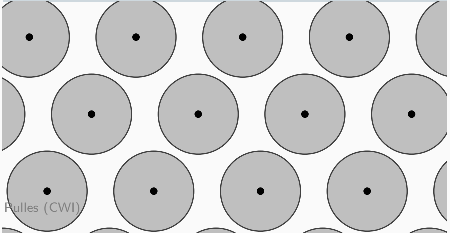
#2: Candidates Closer than the Solution (3/3)

Closeness Lemma

Given a random lattice Λ and $r < \frac{1}{2}$, a uniform target

$$\mathbf{t} \xleftarrow{\$} \mathbb{R}^n/\Lambda,$$

is at most $r\lambda_1$ away from a lattice point with probability r^n.



Geometric contradiction

- Given $T \gg \alpha^{-n}$ uniform targets, there is one of them *closer to* Λ than the α-BDD target.
- This target has a *higher score* than the α-BDD target!

↯

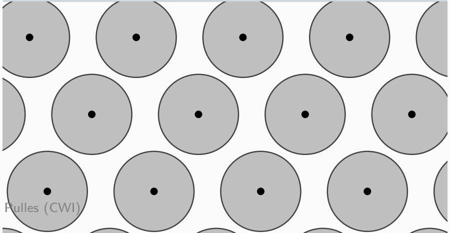
#2: Candidates Closer than the Solution (3/3)

Closeness Lemma

Given a random lattice Λ and $r < \frac{1}{2}$, a uniform target

$$\mathbf{t} \xleftarrow{\$} \mathbb{R}^n/\Lambda,$$

is at most $r\lambda_1$ away from a lattice point with probability r^n.



Geometric contradiction

- Given $T \gg \alpha^{-n}$ uniform targets, there is one of them *closer to* Λ than the α-BDD target.
- This target has a *higher score* than the α-BDD target!

↯

What could be the cause?

Independence Heuristic:

"The scores $(\cos(2\pi \langle \mathbf{w}, \mathbf{t} \rangle))_{\mathbf{w} \in \mathcal{W}}$ **are independent."**

What could be the cause?

Independence Heuristic:

"The scores $\left(\cos(2\pi \langle \mathbf{w}, \mathbf{t} \rangle)\right)_{\mathbf{w} \in \mathcal{W}}$ are independent."

What could be the cause?

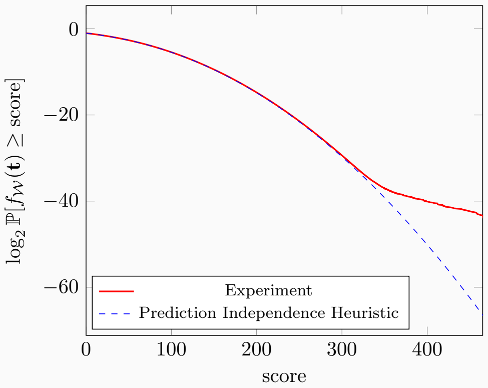
Independence Heuristic:

"The scores $(\cos(2\pi\langle \mathbf{w}, \mathbf{t} \rangle))_{\mathbf{w}\in\mathcal{W}}$ **are independent."**

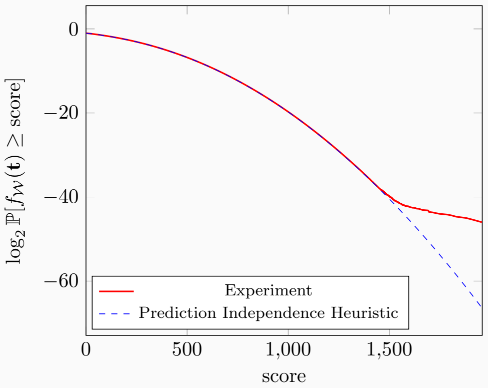
Experimental confirmation

Scores from uniform targets

Score distribution of uniform targets in dimension 60

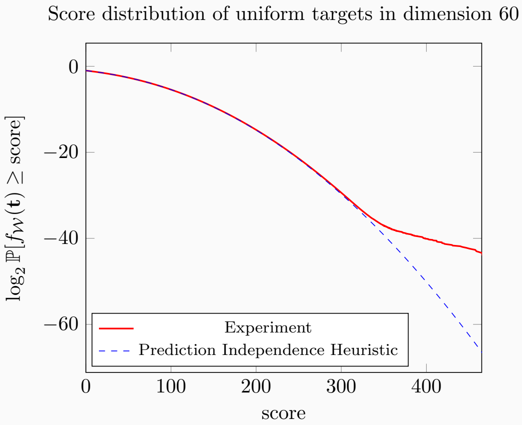


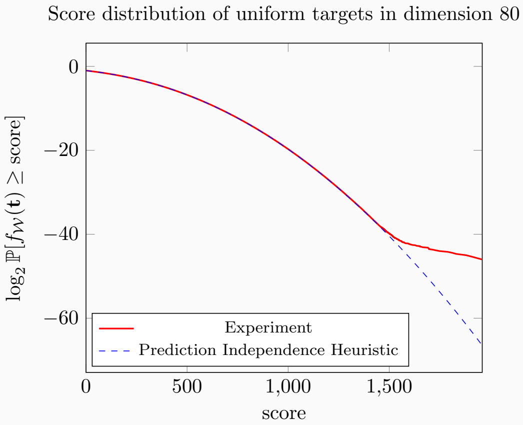
Score distribution of uniform targets in dimension 80



Independence Heuristic overestimates success probability of attack.

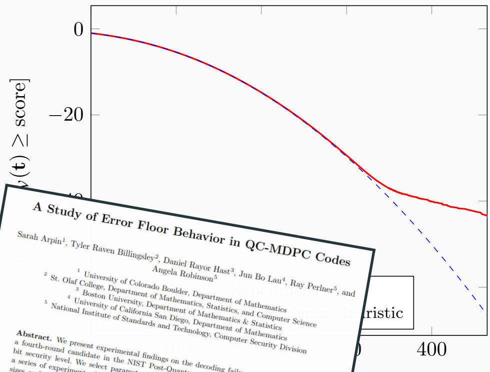
Scores from uniform targets

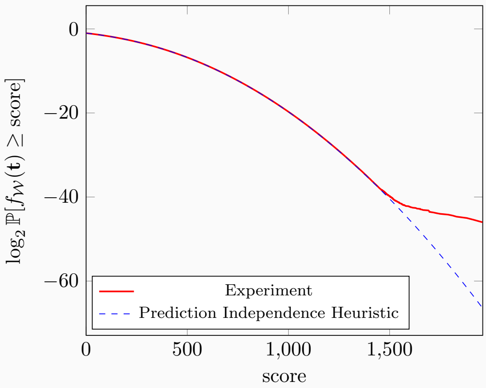


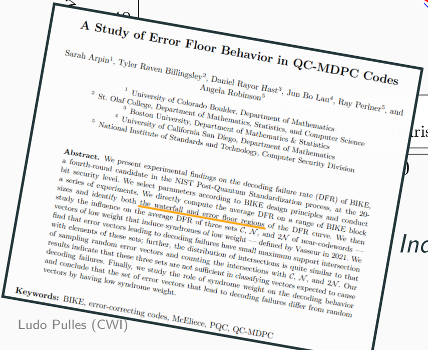


Independence Heuristic overestimates
success probability of attack.

Scores from uniform targets







A Study of Error Floor Behavior in QC-MDPC Codes

Sarah Arpin[1], Tyler Raven Billingsley[2], Daniel Rayor Hast[3], Jun Bo Lau[4], Ray Perlner[5], and Angela Robinson[5]

[1] University of Colorado Boulder, Department of Mathematics
[2] St. Olaf College, Department of Mathematics, Statistics, and Computer Science
[3] Boston University, Department of Mathematics & Statistics
[4] University of California San Diego, Department of Mathematics
[5] National Institute of Standards and Technology, Computer Security Division

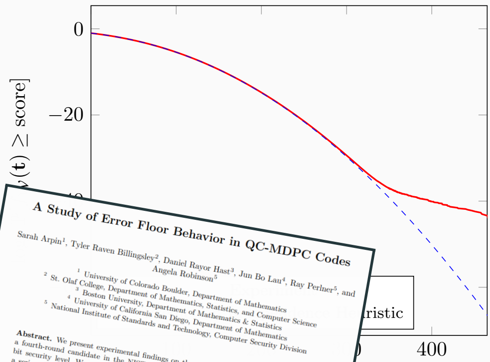
Abstract. We present experimental findings on the decoding failure rate (DFR) of BIKE, a fourth-round candidate in the NIST Post-Quantum Standardization process, at the 20-bit security level. We select parameters according to BIKE design principles and conduct a series of experiments. We directly compute the average DFR on a range of BIKE block sizes and identify both the waterfall and error floor regions of the DFR curve. We then study the influence on the average DFR of three sets $\mathcal{C}$, $\mathcal{N}$, and $2\mathcal{N}$ of near-codewords — vectors of low weight that induce syndromes of low weight — defined by Vasseur in 2021. We find that error vectors leading to decoding failures have small maximum support intersection with elements of these sets; further, the distribution of intersections is quite similar to that of sampling random error vectors and counting the intersections with $\mathcal{C}$, $\mathcal{N}$, and $2\mathcal{N}$. Our results indicate that these three sets are not sufficient in classifying vectors expected to cause decoding failures. Finally, we study the role of syndrome weight on the decoding behavior and conclude that the set of error vectors that lead to decoding failures differ from random vectors by having low syndrome weight.

Keywords: BIKE, error-correcting codes, McEliece, PQC, QC-MDPC

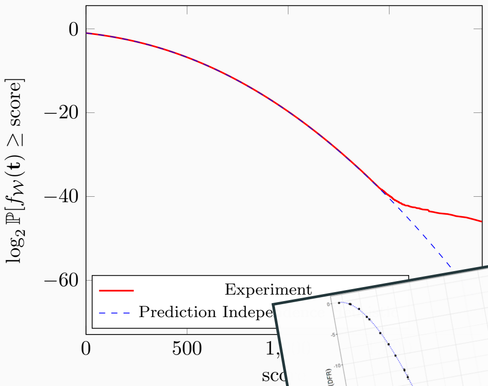
Independence Heuristic overestimates success probability of attack.

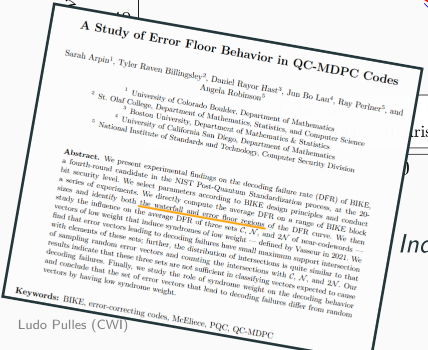
Scores from uniform targets

Score distribution of uniform targets in dimension 60



Score distribution of uniform targets in dimension 80





A Study of Error Floor Behavior in QC-MDPC Codes

Sarah Arpin[1], Tyler Raven Billingsley[2], Daniel Rayor Hast[3], Jun Bo Lau[4], Ray Perlner[5], and Angela Robinson[5]

[1] University of Colorado Boulder, Department of Mathematics
[2] St. Olaf College, Department of Mathematics, Statistics, and Computer Science
[3] Boston University, Department of Mathematics & Statistics
[4] University of California San Diego, Department of Mathematics
[5] National Institute of Standards and Technology, Computer Security Division

Abstract. We present experimental findings on the decoding failure rate (DFR) of BIKE, a fourth-round candidate in the NIST Post-Quantum Standardization process, at the 20-bit security level. We select parameters according to BIKE design principles and conduct a series of experiments. We directly compute the average DFR on a range of BIKE block sizes and identify both the waterfall and error floor regions of the DFR curve. We then study the influence on the average DFR of three sets $\mathcal{C}$, $\mathcal{N}$, and $2\mathcal{N}$ of near-codewords — vectors of low weight that induce syndromes of low weight — defined by Vasseur in 2021. We find that error vectors leading to decoding failures have small maximum support intersection with elements of these sets; further, the distribution of intersections is quite similar to that of sampling random error vectors and counting the intersections with $\mathcal{C}$, $\mathcal{N}$, and $2\mathcal{N}$. Our results indicate that these three sets are not sufficient in classifying vectors expected to cause decoding failures. Finally, we study the role of syndrome weight on the decoding behavior and conclude that the set of error vectors that lead to decoding failures differ from random vectors by having low syndrome weight.

Keywords: BIKE, error-correcting codes, McEliece, PQC, QC-MDPC

Independence Heuristic overestimates

success probability of attack.

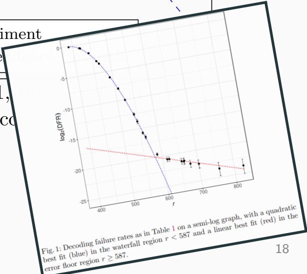
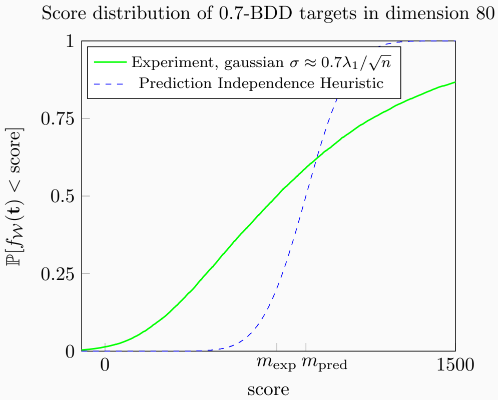


Fig. 1: Decoding failure rates as in Table 1 on a semi-log graph, with a quadratic best fit (blue) in the waterfall region $r < 587$ and a linear best fit (red) in the error floor region $r \geq 587$.

Scores from BDD targets

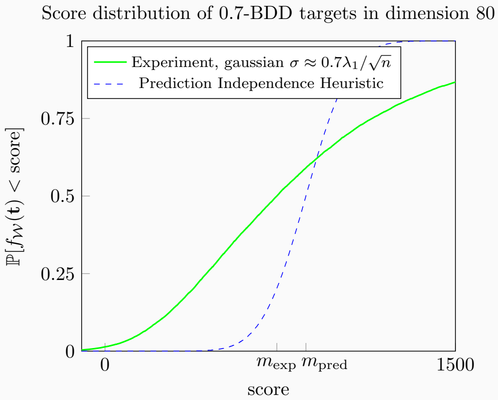


Even prediction of BDD scores is off

- Variance is much higher than predicted.
- Median is lower than predicted.

Again, *Independence Heuristic* overestimates success probability of attack.

Scores from BDD targets



Even prediction of BDD scores is off

- Variance is much higher than predicted.
- Median is lower than predicted.

Again, *Independence Heuristic* overestimates success probability of attack.

Aftermath

What is the impact?

Dual-Sieve analyses are invalidated

- Success probability of the Dual-Sieve attack is significantly overestimated.
- Hardness of BDD with respect to the Dual-Sieve attack is currently unknown.

What is the impact?

Dual-Sieve analyses are invalidated

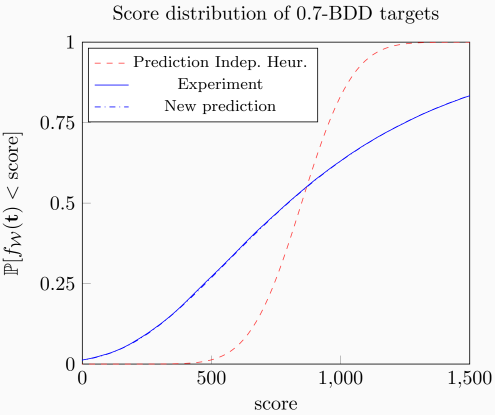
- Success probability of the Dual-Sieve attack is significantly overestimated.
- Hardness of BDD with respect to the Dual-Sieve attack is currently unknown.

What is next?

Ongoing research

- Describing the score distribution of BDD targets using Bessel functions.
- New prediction for uniform targets that predicts "waterfall-floor phenomenon".

A heuristic has to be *stress-tested* on small instances before being used in cryptographic attacks!



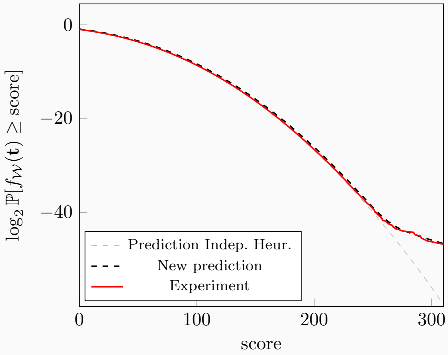
What is next?

Ongoing research

- Describing the score distribution of BDD targets using Bessel functions.
- New prediction for uniform targets that predicts "waterfall-floor phenomenon".

A heuristic has to be *stress-tested* on small instances before being used in cryptographic attacks!

Score distribution of targets drawn uniformly from $\mathbb{R}^n/\Lambda$

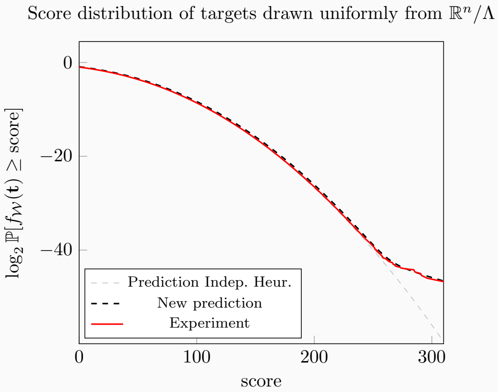


What is next?

Ongoing research

- Describing the score distribution of BDD targets using Bessel functions.
- New prediction for uniform targets that predicts “waterfall-floor phenomenon”.

A heuristic has to be *stress-tested* on small instances before being used in cryptographic attacks!



Thank you!

Questions?

ePrint: https://ia.cr/2023/302

code & data: https://github.com/ludopulles/DoesDualSieveWork