Design and analysis of dual attacks in code- and lattice-based cryptography

PhD Defense, Inria Paris, September 30, 2025

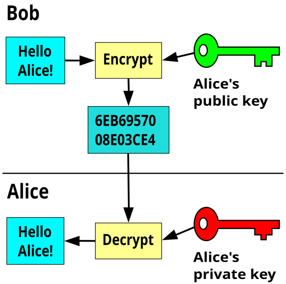
Charles Meyer-Hilfiger, Irisa & Univ. Rennes

Prepared at Inria Paris, COSMIQ

Under the supervision of Nicolas Sendrier and Jean-Pierre Tillich

Public-Key cryptography

Used for safe communication over insecure channel without pre-shared secret.



RSA, DH

$\downarrow$

Hard computational problem

$\uparrow$

Easily solved by quantum computer

Post-Quantum (Public-Key) cryptography

Lattice, Code, Multivariate, Isogenies, ...

	Code-based	Lattice-based
Encryption	**HQC (NIST)**, McEliece, Bike, ...	**Kyber (NIST)**,...
Signature	SDiTH,...	**Dilithium (NIST)**,...
Security	**Decoding problem**	**Learning with Errors**

$\rightarrow$ **Hard** problem even for quantum computer

Complexity of best algorithms used to parametrize schemes.

Post-Quantum (Public-Key) cryptography

Lattice, Code, Multivariate, Isogenies, ...

	Code-based	Lattice-based
Encryption	**HQC (NIST)**, McEliece, Bike, ...	**Kyber (NIST)**,...
Signature	SDiTH,...	**Dilithium (NIST)**,...
Security	**Decoding problem**	**Learning with Errors**

$\rightarrow$ **Hard** problem even for quantum computer

Complexity of best algorithms used to parametrize schemes.

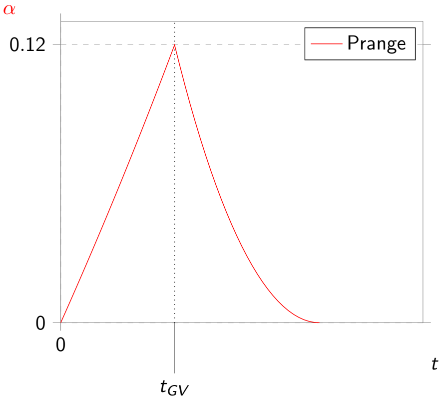
Binary Decoding Problem

$$\text{Binary Linear code} \quad \rightarrow \quad \mathscr{C} = \{\ \mathbf{mG} \ : \ \mathbf{m} \in \mathbb{F}_2^n \}$$

Decoding at a small distance t:

- **Input:** $(\mathbf{G}, \mathbf{y} = \mathbf{c} + \mathbf{e}) \in \mathbb{F}_2^{k \times n} \times \mathbb{F}_2^n$ where $\mathbf{c} \in \mathscr{C}$ and $|\mathbf{e}| = t$
- **Output:** $\mathbf{e}$ such that $|\mathbf{e}| = t$ and $\mathbf{y} - \mathbf{e} \in \mathscr{C}$

Hardness of the decoding problem as a function of the distance



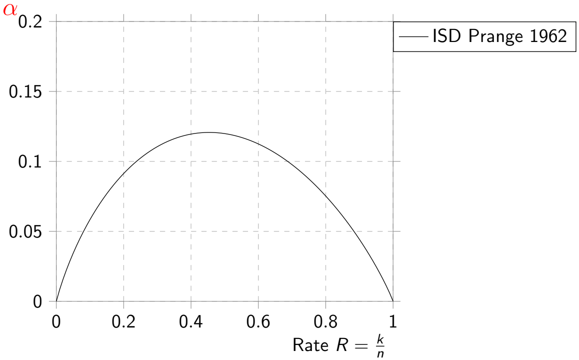
Complexity : $2^{\alpha n}$

Gilbert-Varshamov distance t_{GV} is where the problem is hardest

Complexity of some decoders

Complexity is $2^{\alpha n}$

$k \stackrel{\triangle}{=} \mathrm{Dimension}\,(\mathscr{C})$

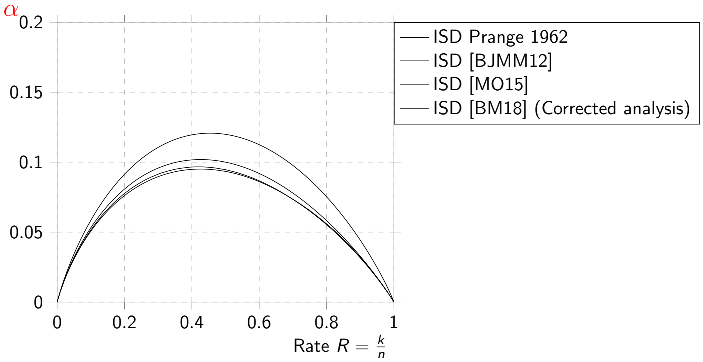


Complexity of some decoders

Main family of algorithms for 60 years : Information Set Decoders (ISD)

Complexity is $2^{\alpha n}$

$k \triangleq \mathrm{Dimension}\,(\mathscr{C})$



Complexity of some decoders

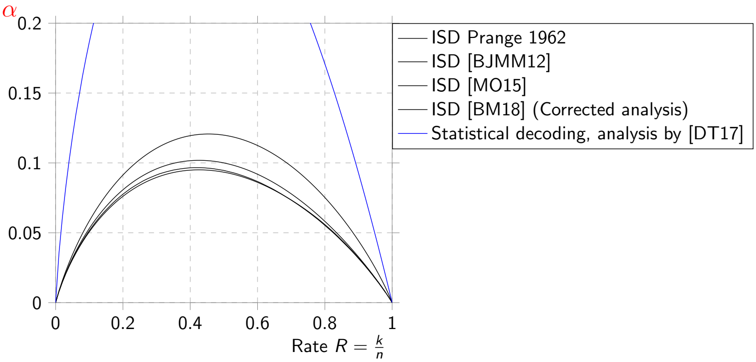
Main family of algorithms for 60 years : Information Set Decoders (ISD)

An outlier, a **Dual attack** : **Statistical decoding** by Al-Jabri 2001

$\rightarrow$ Debris-Alazard & Tillich 2017 shows that it is asymptotically not competitive.

Complexity is $2^{\alpha n}$

$k \stackrel{\triangle}{=} \text{Dimension}\left(\mathscr{C}\right)$



Code-Based Contribution of this thesis (1)

New dual attacks:

State-of-art : Code-based dual attacks are not competitive

Our work:

- **Significant improvement** of **statistical decoding** by generalizing it.
- Our best attack **outperforms Information Set Decoders** for a significant regime.

Analyzing dual attacks:

State-of-art : Analyze of dual attacks require the use of key **Independence assumption**

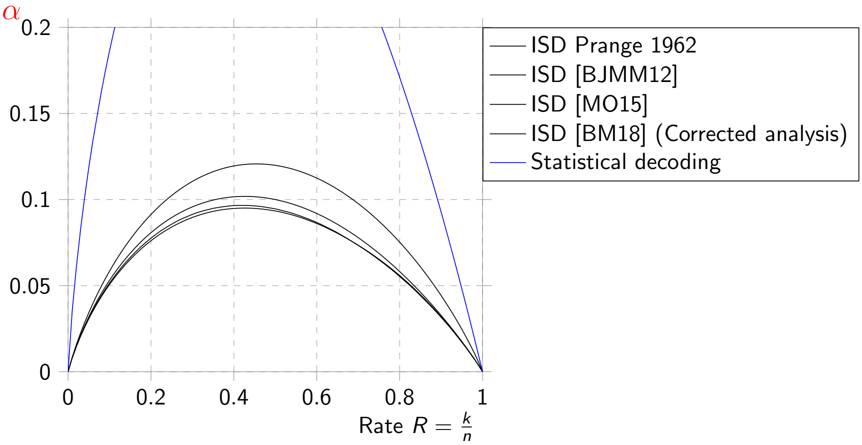
Our work:

- Show experimentally that these **Independence assumptions** do not always hold.
- Replace these **Independence assumptions** by a new **Poisson Model**.
- Eventually find a way to analyze these attacks **without any assumptions**.

Complexity of our best attack

Complexity is
$2^{\alpha n}$

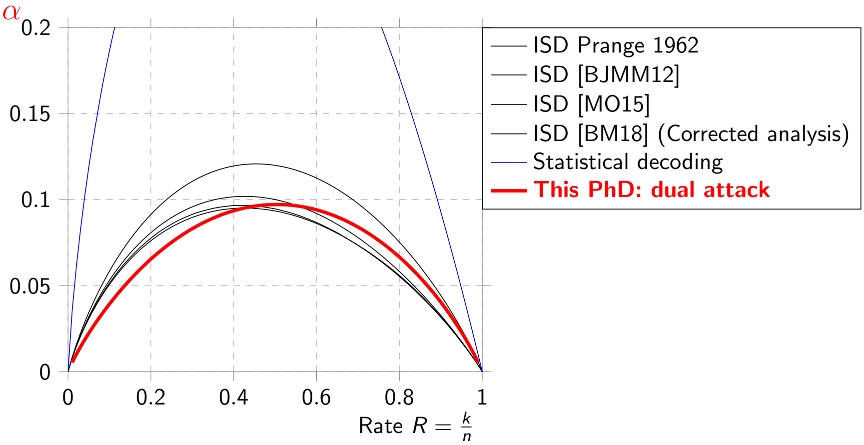
$k \stackrel{\triangle}{=} \mathrm{Dimension}\,(\mathscr{C})$



Complexity of our best attack

Complexity is
$2^{\alpha n}$

$k \stackrel{\triangle}{=} \mathrm{Dimension}(\mathscr{C})$



State-of-the-art

Learning With Errors : Primal Attacks vs **Dual attacks**

$\downarrow$

Recently became competitive :

Guo & Johansson 2021 and Matzov 2022 attack on **Kyber**

Analysis relies on **standard independence assumption**

Controversy:

Ducas & Pulles 2023 $\rightarrow$ **Independence assumption** is flawed

"Does the Dual-Sieve Attack on Learning with Errors even Work?"

Lattice-based contribution

Our work:

Settling the controversy : A competitive **dual attack** can work as expected.

$\uparrow$

- Devise a **slightly improved variant** of Matzov dual attack
- Analyze : **No Independence assumption** but a new **Model**
- **Dents the security of Kyber**

Publications

Most of these results come from the following publications:

- [$CDMT$22] : K. Carrier, T. Debris-Alazard, J-P. Tillich. Asiacrypt 2022.
- [MT23] : J-P. Tillich. TCC 2023.
- [$CDMT$24] : K. Carrier, T. Debris-Alazard, J-P. Tillich. Eurocrypt 2024.
- [$CMST$25] : K. Carrier, Y. Shen, J-P. Tillich. Crypto 2025.

Setting for Dual Attacks

Dual code:

$$\mathscr{C}^{\perp} = \{\mathbf{h} \in \mathbb{F}_q^n \ : \ \langle \mathbf{h}, \mathbf{c} \rangle = 0 \quad \forall \mathbf{c} \in \mathscr{C}\} \qquad \text{with} \qquad \langle \mathbf{x}, \mathbf{y} \rangle = \sum x_i \, y_i \pmod{q}$$

Compute dual vector $\mathbf{h} \in \mathscr{C}^{\perp}$

Observation:

Given $\mathbf{y} = \mathbf{c} + \mathbf{e}$ $\quad \rightarrow \langle \mathbf{y}, \mathbf{h} \rangle = \langle \mathbf{c} + \mathbf{e}, \mathbf{h} \rangle = \langle \mathbf{e}, \mathbf{h} \rangle$

Key fact:

More biased toward 0 as $|\mathbf{e}|$, $|\mathbf{h}|$ smaller.

First dual attack: Statistical Decoding (Al-Jabri 2001)

Compute $\mathbf{h} \in \mathscr{C}^{\perp}$ of low weight $|\mathbf{h}| = w$ such that $\mathbf{h}_1 = 1$:

$$\langle \mathbf{y}, \mathbf{h} \rangle = \langle \mathbf{e}, \mathbf{h} \rangle = \sum \mathbf{e}_i \mathbf{h}_i = \mathbf{e}_1 + \sum \mathbf{e}_i \mathbf{h}_i \sim \begin{cases} \text{Bernouilli}\left(\frac{1-\delta}{2}\right) & \text{if } \mathbf{e}_1 = 0 \\ \text{Bernouilli}\left(\frac{1+\delta}{2}\right) & \text{if } \mathbf{e}_1 = 1 \end{cases}$$

Compute N such dual vectors $\rightarrow$ Decide with majority voting

How big must N be to make good decision?

Condition for statistical decoding to succeed

$$\langle \mathbf{y}, \mathbf{h} \rangle = \langle \mathbf{e}, \mathbf{h} \rangle = \sum \mathbf{e}_i \mathbf{h}_i = \mathbf{e}_1 + \sum \mathbf{e}_i \mathbf{h}_i \sim \begin{cases} \mathrm{Bernouilli}\left(\frac{1-\delta}{2}\right) & \text{if } \mathbf{e}_1 = 0 \\ \mathrm{Bernouilli}\left(\frac{1+\delta}{2}\right) & \text{if } \mathbf{e}_1 = 1 \end{cases}$$

Supposing $\mathbf{h}$ is taken uniformly in $\mathscr{C}^\perp$ of weight w such that $\mathbf{h}_1 = 1$:

$$\mathrm{Bias}\left(\langle \mathbf{e}, \mathbf{h} \rangle\right) \quad \triangleq \quad \mathbb{P}\left(\langle \mathbf{e}, \mathbf{h} \rangle = 0\right) - \mathbb{P}\left(\langle \mathbf{e}, \mathbf{h} \rangle = 1\right) \quad = \quad \pm\delta\left(w\right)$$

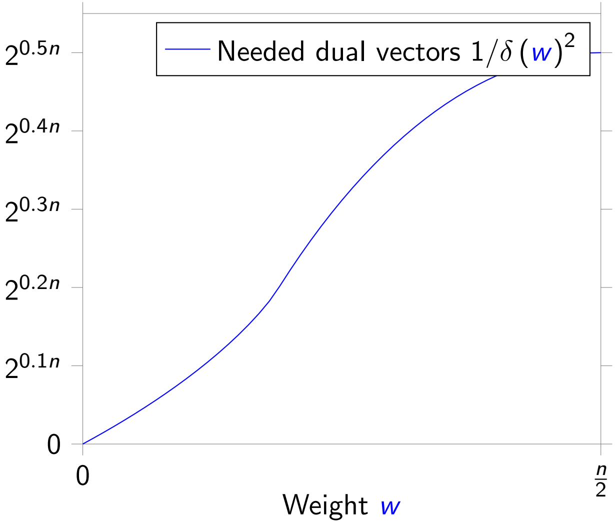
To make right decision, under assumption that the $\langle \mathbf{y}, \mathbf{h} \rangle$'s are **independent**, N required to be

$$N > \frac{1}{\mathrm{Bias}\left(\langle \mathbf{e}, \mathbf{h} \rangle\right)^2}$$

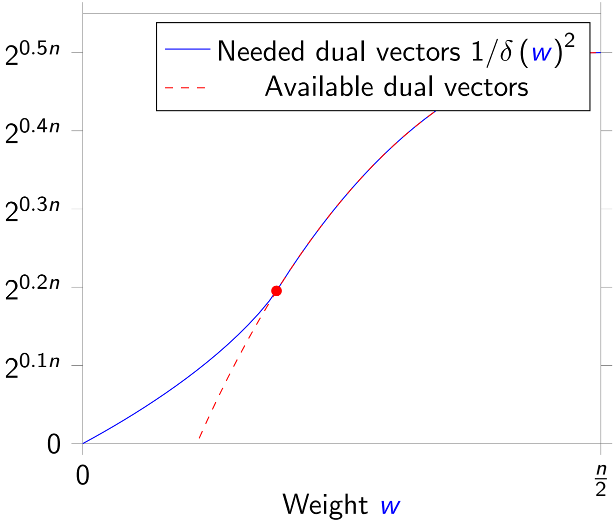
Condition

$$N > \frac{1}{\delta\left(w\right)^2}$$

Limiting factor in statistical decoding



Limiting factor in statistical decoding



A path toward improvement as an open question

Suggestion of Debris-Alazard & Tillich 2017 :

$\rightarrow$ Compute dual vectors of low weight only on a subpart of the support ?

- Split support in complementary part $\mathscr{P}$ and $\mathscr{N}$ $\rightarrow$ Recover $\mathbf{e}_{\mathscr{P}}$?
- Compute dual vector $\mathbf{h} = \underbrace{\boxed{}}_{\mathscr{P}}\underbrace{\boxed{\quad w \text{ (small)} \quad}}_{\mathscr{N}}$

$$\rightarrow \langle \mathbf{y}, \mathbf{h} \rangle = \langle \mathbf{e}, \mathbf{h} \rangle = \langle \underbrace{\mathbf{e}_{\mathscr{P}}}_{\text{secret}}, \mathbf{h}_{\mathscr{P}} \rangle + \underbrace{\langle \mathbf{e}_{\mathscr{N}}, \mathbf{h}_{\mathscr{N}} \rangle}_{\text{noise: biased to 0}}$$

Intuition : improve this limiting factor by decreasing the noise.

Why is this so advantageous?

This strategy is highly beneficial (1)

- Compute dual vector $\mathbf{h} = \underbrace{\boxed{}}_{\mathscr{P}}\underbrace{\boxed{\quad w \text{ (small)} \quad}}_{\mathscr{N}}$

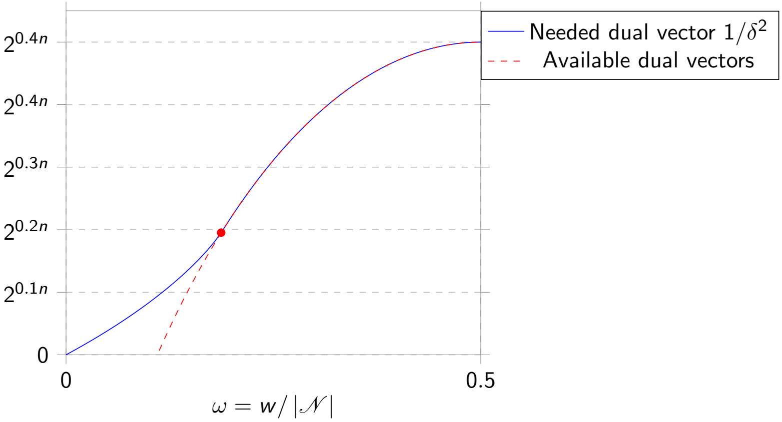
$$\rightarrow \langle \mathbf{y}, \mathbf{h} \rangle = \langle \mathbf{e}, \mathbf{h} \rangle = \langle \underbrace{\mathbf{e}_{\mathscr{P}}}_{\text{secret}}, \mathbf{h}_{\mathscr{P}} \rangle + \underbrace{\langle \mathbf{e}_{\mathscr{N}}, \mathbf{h}_{\mathscr{N}} \rangle}_{\text{noise: biased to 0}}$$

Supposing **Independence assumption**

Number of dual vectors $N \geq \dfrac{1}{\mathrm{bias}\left(\langle \mathbf{e}_{\mathscr{N}}, \mathbf{h}_{\mathscr{N}} \rangle\right)^2}$ $\rightarrow$ Can recover secret $\mathbf{e}_{\mathscr{P}}$

This strategy is highly beneficial (2)

Statistical Decoding
$\downarrow$
$|\mathscr{P}| = 1$

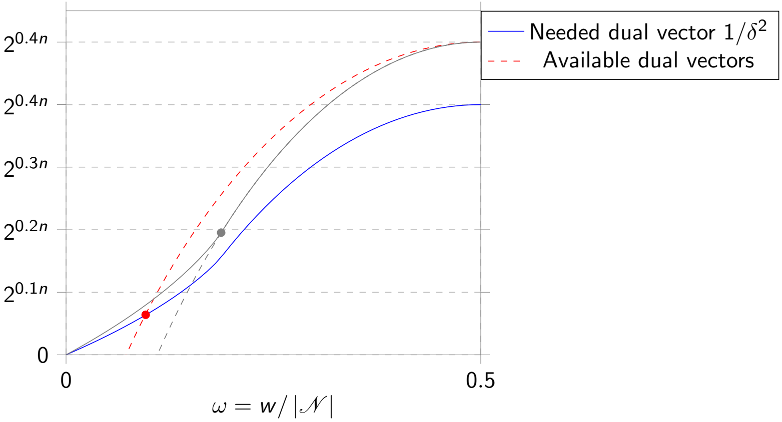


This strategy is highly beneficial (2)

Our attacks

$\downarrow$

$|\mathscr{P}|$ $>$ 1



Can we leverage it?

Reducing Decoding to LPN

- Compute dual vector $\mathbf{h} = \underbrace{\boxed{}}_{\mathscr{P}}\underbrace{\boxed{\;w \text{ (small)}\;}}_{\mathscr{N}}$

$$\to \langle \mathbf{y}, \mathbf{h} \rangle = \langle \mathbf{e}, \mathbf{h} \rangle = \langle \underbrace{\mathbf{e}_{\mathscr{P}}}_{\text{secret}}, \mathbf{h}_{\mathscr{P}} \rangle + \underbrace{\langle \mathbf{e}_{\mathscr{N}}, \mathbf{h}_{\mathscr{N}} \rangle}_{\text{noise: biased to } 0}$$

LPN Problem

- **Input:** Many samples $(\mathbf{a}, \langle \mathbf{a}, \mathbf{s} \rangle + e)$
  - $\mathbf{s} \in \mathbb{F}_2^s$ fixed secret
  - $\mathbf{a}$ taken at random in $\mathbb{F}_2^s$
  - $e \sim \mathrm{Ber}\,(p)$
- **Output:** $\mathbf{s}$

N dual vectors $\rightarrow$ N LPN samples

$$(\mathbf{a}, \langle \mathbf{s}, \mathbf{a} \rangle + e) \text{ w.t } \begin{cases} \mathbf{a} = \mathbf{h}_{\mathscr{P}} & \in \mathbb{F}_2^{|\mathscr{P}|} \\ \mathbf{s} = \mathbf{e}_{\mathscr{P}} \\ e = \langle \mathbf{e}_{\mathscr{N}}, \mathbf{h}_{\mathscr{N}} \rangle \end{cases}$$

Recovering $\mathbf{e}_{\mathscr{P}}$ is solving an LPN problem

Score function

$$\text{LPN sample } \langle \mathbf{y}, \mathbf{h} \rangle = \langle \mathbf{e}_{\mathscr{P}}, \mathbf{h}_{\mathscr{P}} \rangle + \langle \mathbf{h}_{\mathscr{N}}, \mathbf{e}_{\mathscr{N}} \rangle$$

Score function

For $\mathbf{x} \in \mathbb{F}_2^{|\mathscr{P}|}$ score function

$$\mathbf{F}(\mathbf{x}) \triangleq \sum_{\mathbf{h} \in \mathscr{H}} (-1)^{\langle \mathbf{y}, \mathbf{h} \rangle - \langle \mathbf{x}, \mathbf{h}_{\mathscr{P}} \rangle}$$

where $\mathscr{H}$ is set of N computed low weight dual vectors.

$\langle \mathbf{y}, \mathbf{h} \rangle - \langle \mathbf{e}_{\mathscr{P}}, \mathbf{h}_{\mathscr{P}} \rangle = \langle \mathbf{h}_{\mathscr{N}}, \mathbf{e}_{\mathscr{N}} \rangle$ is biased toward $0 \to \mathbf{F}(\mathbf{e}_{\mathscr{P}})$ **Big**.

Goal of LPN solver

LPN Solver

Return set of candidates for the solution

$$\mathcal{S} \triangleq \{ \mathbf{x} \in \mathbb{F}_2^{|\mathscr{P}|} \ : \ \mathbf{F}(\mathbf{x}) > T \}$$

$$\text{where } T \triangleq \frac{1}{2}\mathbb{E}\left(\mathbf{F}\left(\mathbf{e}_{\mathscr{P}}\right)\right)$$

An FFT based LPN solver

We have computed N dual vectors $\mathbf{h}$. Compute for each $\mathbf{x} \in \mathbb{F}_2^{|\mathscr{P}|}$

$$\mathbf{F}(\mathbf{x}) \triangleq \sum_{\mathbf{h}} (-1)^{\langle \mathbf{y},\mathbf{h}\rangle - \langle \mathbf{x},\mathbf{h}_{\mathscr{P}}\rangle}$$

Naive search

$$2^{|\mathscr{P}|} \times N$$

Levieil & Fouque 2006

Use a Fast Fourier Transform

$$|\mathscr{P}|\, 2^{|\mathscr{P}|} + N$$

$\rightarrow$ Exponential speed-up

Returns set of candidates $\mathcal{S} \triangleq \{\, \mathbf{x} \in \mathbb{F}_2^{|\mathscr{P}|} \; : \; \mathbf{F}(\mathbf{x}) > T \}$

Algorithm

Decode(n, k, t)
Input : $\mathscr{C}$, $\mathbf{y} = \mathbf{c} + \mathbf{e}$
Output : $\mathbf{e}$
Choose $\mathscr{P}$ and $\mathscr{N}$ at random
$\mathscr{H} \leftarrow$ Compute N dual vectors of $\mathscr{C}$ such that $|\mathbf{h}_{\mathscr{N}}| = w$ $\triangleright$ *Using technique from ISD*
$\mathcal{S} \leftarrow \mathrm{LPNSolver}\left(\quad ((\mathbf{h}_{\mathscr{P}}, \langle \mathbf{y}, \mathbf{h} \rangle))_{\mathbf{h} \in \mathscr{H}} \quad \right)$ $\triangleright$ *Small set of candidates for the secret* $\mathbf{e}_{\mathscr{P}}$
for $\mathbf{x} \in \mathcal{S}$ **do**
  $\textsc{Decode}(n - |\mathscr{P}|\,,\; k - |\mathscr{P}|\,,\; t')$ $\triangleright$ *Check if* $\mathbf{x} = \mathbf{e}_{\mathscr{P}}$ *by solving a smaller decoding problem. If* $\mathbf{x} = \mathbf{e}_{\mathscr{P}}$ *this decoding succeed and returns* $\mathbf{e}$.

Complexity:

$$T_{\text{Compute Vectors}} + T_{\text{LPN Solver}} + T_{\text{Decode}} \times |\mathcal{S}|$$

Complexity

Complexity:

$$T_{\text{Compute Vectors}} + T_{\text{LPN Solver}} + T_{\text{Decode}} \times |\mathcal{S}|$$

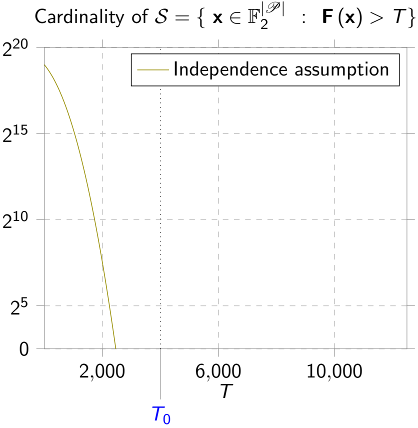
Goal : Prove that the last part **is negligible** for **reasonable parameters.**

Key study:

Tight bound of cardinalilty of $\mathcal{S} \stackrel{\triangle}{=} \{ \mathbf{x} \in \mathbb{F}_2^{|\mathscr{P}|} \ : \ \mathbf{F}(\mathbf{x}) > T \}$

Difficulty $|\mathscr{P}| = \Theta(n) \rightarrow$ Needs to understand the exponential tail behavior of $\mathbf{F}(\mathbf{x})$.

Number of false candidates in a perfect world



Independence Assumption:

The terms in
$\mathbf{F}(\mathbf{x}) = \sum_{\mathbf{h} \in \mathscr{C}^{\perp}} (-1)^{\langle \mathbf{y}, \mathbf{h} \rangle - \langle \mathbf{x}, \mathbf{h}_{\mathscr{P}} \rangle} 1_{|\mathbf{h}_{\mathscr{N}}| = w}$ are independent variables.

Under independence assumption if

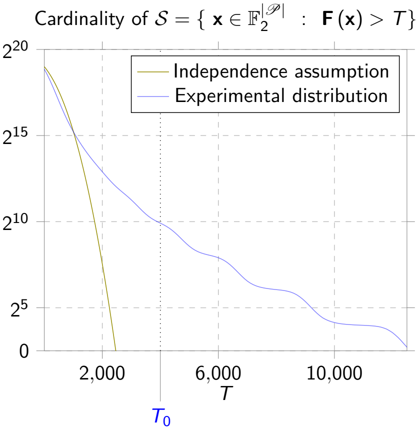
$$N > \frac{n}{\delta^2}$$

then taking $T_0 \triangleq \frac{1}{2} \mathbb{E}(\mathbf{F}(\mathbf{e}_{\mathscr{P}}))$

$$\{ \mathbf{x} \in \mathbb{F}_2^{|\mathscr{P}|} \; : \; \mathbf{F}(\mathbf{x}) > T_0 \} = \{ \mathbf{e}_{\mathscr{P}} \}$$

Can distinguish $\mathbf{e}_{\mathscr{P}}$, no false candidate.

Number of false candidates in a perfect world



~~**Independence Assumption**~~

Independence Assumption:

The terms in $\mathbf{F}(\mathbf{x}) = \sum_{\mathbf{h} \in \mathscr{C}^\perp} (-1)^{\langle \mathbf{y}, \mathbf{h} \rangle - \langle \mathbf{x}, \mathbf{h}_{\mathscr{P}} \rangle} 1_{|\mathbf{h}_{\mathscr{N}}| = w}$ are independent variables.

Under independence assumption if

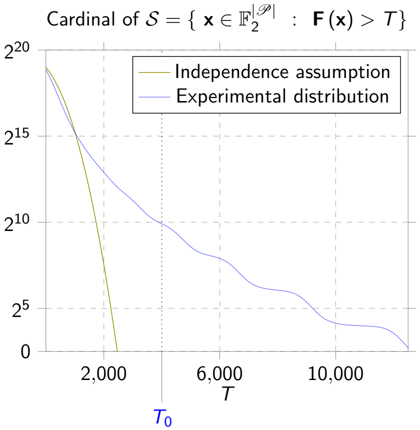
$$N > \frac{n}{\delta^2}$$

then taking $T_0 \stackrel{\triangle}{=} \frac{1}{2} \mathbb{E}(\mathbf{F}(\mathbf{e}_{\mathscr{P}}))$

$$\{ \mathbf{x} \in \mathbb{F}_2^{|\mathscr{P}|} \ : \ \mathbf{F}(\mathbf{x}) > T_0 \} = \{ \mathbf{e}_{\mathscr{P}} \}$$

Can distinguish $\mathbf{e}_{\mathscr{P}}$, no false candidate.

Prediction of the number of false candidates



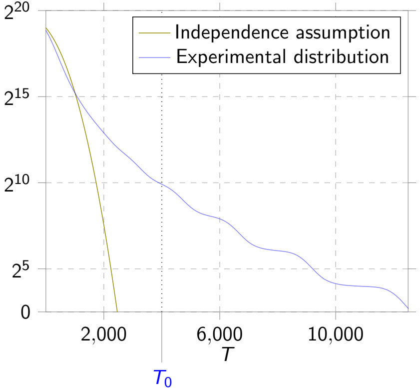
Theorem : Dual formula

$$F(\mathbf{x}) = \sum_{\mathbf{i} \in \mathbb{N}} N_{\mathbf{i}} \left(\mathscr{C}^{\mathscr{N}} + g(\mathbf{x}) \right) K_w(\mathbf{i})$$

- $\mathscr{C}^{\mathscr{N}} \stackrel{\triangle}{=} \{ \mathbf{c}_{\mathscr{N}} \ : \ \mathbf{c} \in \mathscr{C} \text{ s.t } \mathbf{c}_{\mathscr{P}} = 0 \}$
- $N_{\mathbf{i}}(\mathscr{D})$ number word of weight $\mathbf{i}$ of $\mathscr{D}$
- K_w Krawtchouk polynomial
- $g(\mathbf{x})$ affine function

Proof: Poisson formula + $\widehat{1_w} = K_w$

Prediction of the number of false candidates



Theorem : Dual formula

$$F(\mathbf{x}) = \sum_{\mathbf{i} \in \mathbb{N}} N_{\mathbf{i}} \left(\mathscr{C}^{\mathscr{N}} + g(\mathbf{x}) \right) K_w(\mathbf{i})$$

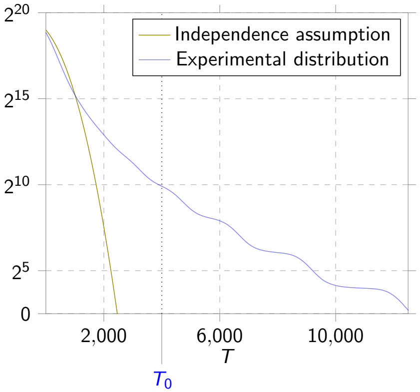
- $\mathscr{C}^{\mathscr{N}} \stackrel{\triangle}{=} \{ \mathbf{c}_{\mathscr{N}} \ : \ \mathbf{c} \in \mathscr{C} \text{ s.t } \mathbf{c}_{\mathscr{P}} = 0 \}$
- $N_{\mathbf{i}}(\mathscr{D})$ number word of weight $\mathbf{i}$ of $\mathscr{D}$
- K_w Krawtchouk polynomial
- $g(\mathbf{x})$ affine function

Proof: Poisson formula + $\widehat{1_w} = K_w$

Tight estimation of number of candidates $\Leftarrow \mathbb{P}\left(N_{\mathbf{i}} - \mathbb{E}(N_{\mathbf{i}}) > \mathrm{poly}(n) \sqrt{\mathbf{Var} N_{\mathbf{i}}} \right) = 2^{-\Theta(n)}$

Prediction of the number of false candidates

Cardinal of $\mathcal{S} = \{ \mathbf{x} \in \mathbb{F}_2^{|\mathscr{P}|} \ : \ \mathbf{F}(\mathbf{x}) > T \}$



Theorem : Dual formula

$$F(\mathbf{x}) = \sum_{\mathbf{i} \in \mathbb{N}} N_{\mathbf{i}} \left(\mathscr{C}^{\mathscr{N}} + g(\mathbf{x}) \right) K_w(\mathbf{i})$$

- $\mathscr{C}^{\mathscr{N}} \stackrel{\triangle}{=} \{ \mathbf{c}_{\mathscr{N}} \ : \ \mathbf{c} \in \mathscr{C} \text{ s.t } \mathbf{c}_{\mathscr{P}} = 0 \ \}$
- $N_{\mathbf{i}}(\mathscr{D})$ number word of weight $\mathbf{i}$ of $\mathscr{D}$
- K_w Krawtchouk polynomial
- $g(\mathbf{x})$ affine function

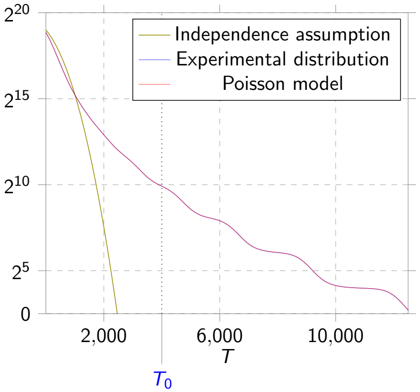
Proof: Poisson formula $+ \widehat{1_w} = K_w$

Tight estimation of number of candidates $\Leftarrow \mathbb{P}\left(N_{\mathbf{i}} - \mathbb{E}(N_{\mathbf{i}}) > \mathrm{poly}(n) \sqrt{\mathbf{Var} N_{\mathbf{i}}} \right) = 2^{-\Theta(n)}$

Model: $N_{\mathbf{i}}(\mathscr{D}) \sim \mathrm{Poisson}$ variable of right expected value

Prediction of the number of false candidates

Cardinal of $\mathcal{S} = \{ \mathbf{x} \in \mathbb{F}_2^{|\mathscr{P}|} \ : \ \mathbf{F}(\mathbf{x}) > T \}$



Theorem : Dual formula

$$F(\mathbf{x}) = \sum_{\mathbf{i} \in \mathbb{N}} N_{\mathbf{i}} \left(\mathscr{C}^{\mathscr{N}} + g(\mathbf{x}) \right) K_w(\mathbf{i})$$

- $\mathscr{C}^{\mathscr{N}} \stackrel{\triangle}{=} \{ \mathbf{c}_{\mathscr{N}} \ : \ \mathbf{c} \in \mathscr{C} \text{ s.t } \mathbf{c}_{\mathscr{P}} = 0 \ \}$
- $N_{\mathbf{i}}(\mathscr{D})$ number word of weight $\mathbf{i}$ of $\mathscr{D}$
- K_w Krawtchouk polynomial
- $g(\mathbf{x})$ affine function

Proof: Poisson formula + $\widehat{1_w} = K_w$

Tight estimation of number of candidates $\Leftarrow \mathbb{P}\left(N_{\mathbf{i}} - \mathbb{E}(N_{\mathbf{i}}) > \mathrm{poly}(n)\sqrt{\mathbf{Var} N_{\mathbf{i}}}\right) = 2^{-\Theta(n)}$

Model: $N_{\mathbf{i}}(\mathscr{D}) \sim$ Poisson variable of right expected value

Number of false candidates

Theorem:

Under the Poisson Model when

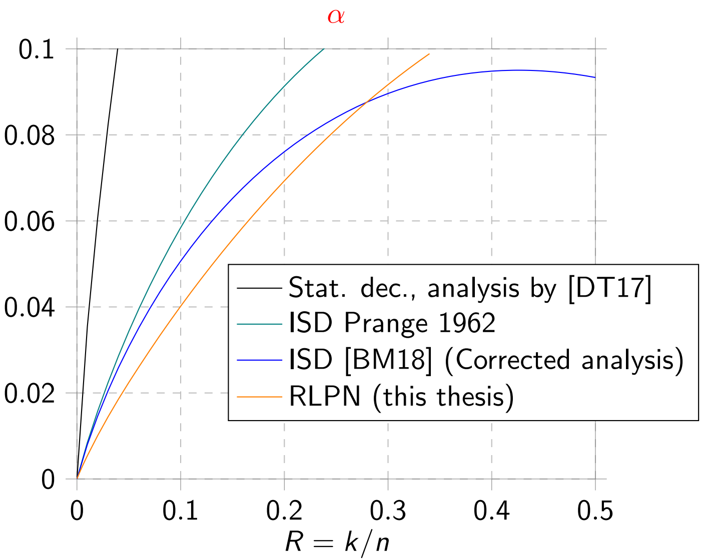
$$N > \frac{n^8}{\delta^2}$$

there are $\mathrm{poly}(n)$ **false candidates.**

$\rightarrow$ **Overall cost of dealing with false candidates is negligible.**

Results

Complexity : $2^{\alpha n}$



RLPN is not optimal

$$\langle \mathbf{y}, \mathbf{h} \rangle = \langle \mathbf{e}_{\mathscr{P}}, \mathbf{h}_{\mathscr{P}} \rangle + \langle \mathbf{e}_{\mathscr{N}}, \mathbf{h}_{\mathscr{N}} \rangle$$

N dual vectors $\rightarrow$ N LPN samples

$$(\mathbf{a}, \langle \mathbf{s}, \mathbf{a} \rangle + e) \text{ w.t } \begin{cases} \mathbf{a} = \mathbf{h}_{\mathscr{P}} \quad \in \mathbb{F}_2^{|\mathscr{P}|} \\ \mathbf{s} = \mathbf{e}_{\mathscr{P}} \\ e = \langle \mathbf{e}_{\mathscr{N}}, \mathbf{h}_{\mathscr{N}} \rangle \end{cases}$$

Secret $\mathbf{e}_{\mathscr{P}}$ is sparse and yet FFT computes $F(\mathbf{x})$ for all $\mathbf{x} \in \mathbb{F}_2^{|\mathscr{P}|}$

Reducing sparse LPN to plain LPN (1)

General approach : dimension reduction

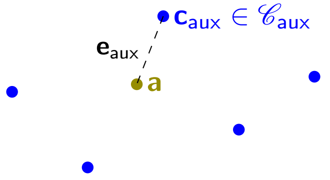
$$\text{LPN sample :} \quad \left(\underset{\mathbb{F}_2^{|\mathscr{P}|}}{\overset{}{\underset{\cap}{\mathbf{a}}}} \;,\; \langle \overset{\text{sparse}}{\overset{\uparrow}{\mathbf{s}}} \;,\; \mathbf{a} \rangle + e \right) \xrightarrow[\text{Increase Noise}]{\text{Lower Dimension}} \left(\underset{\mathbb{F}_2^{\leq |\mathscr{P}|}}{\underset{\cap}{\mathbf{a}'}} \;,\; \langle \overset{\text{uniform}}{\overset{\uparrow}{\mathbf{s}'}} \;,\; \mathbf{a}' \rangle + e' \right)$$

Reduction from sparse to plain LPN

$\rightarrow$ Technique by Guo, Johansson, Löndahl (2014)

Linear code $\mathscr{C}_{\text{aux}} \subset \mathbb{F}_2^{|\mathscr{P}|}$

$$\mathscr{C}_{\text{aux}} = \{\mathbf{m}_{\text{aux}}\mathbf{G}_{\text{aux}} : \mathbf{m}_{\text{aux}} \in \mathbb{F}_2^{\dim(\mathscr{C}_{\text{aux}})}\}$$



$$\mathbf{a} = \mathbf{c}_{\text{aux}} + \underbrace{\mathbf{e}_{\text{aux}}}_{\text{short}}$$

$$\langle \mathbf{s}, \mathbf{a} \rangle + e = \langle \mathbf{s}, \mathbf{c}_{\text{aux}} \rangle + \underbrace{\langle \mathbf{s}, \mathbf{e}_{\text{aux}} \rangle + e}_{e' \text{ new noise}}$$

$$\langle \mathbf{s}, \mathbf{c}_{\text{aux}} \rangle = \langle \mathbf{s}, \mathbf{m}_{\text{aux}}\mathbf{G}_{\text{aux}} \rangle = \langle \mathbf{s}\mathbf{G}_{\text{aux}}^{\top}, \mathbf{m}_{\text{aux}} \rangle$$

Sample space $\mathbb{F}_2^{|\mathscr{P}|} \rightarrow \mathbb{F}_2^{\dim(\mathscr{C}_{\text{aux}})}$ is smaller!

The complete algorithm

DoubleRLPN

Same as RLPN but replace FFT LPN solver by Reduction + FFT

Number of false candidates in doubleRLPN

Theorem

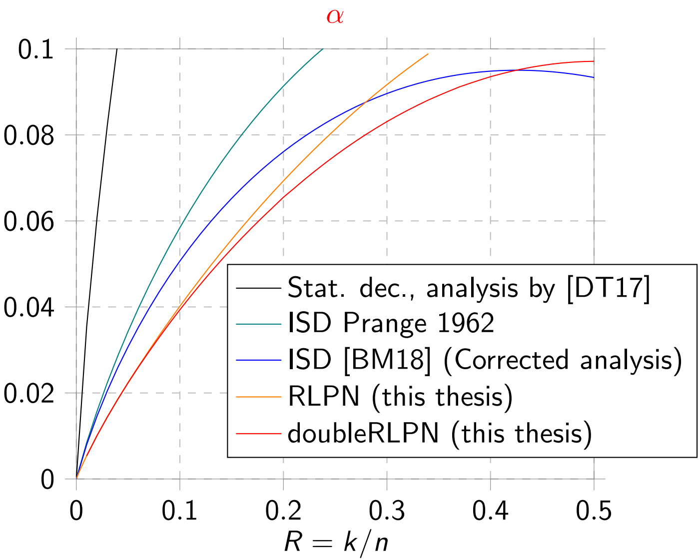
Under the Poisson Model when

$$N > \frac{n^8}{\delta^2}$$

there are $2^{\beta n}$ **false candidates.** (instead of $\mathrm{poly}(n)$ in RLPN)

$\rightarrow$ **Overall cost of dealing with false candidates is still negligible.**

Results



Complexity : $2^{\alpha n}$

Outperforms state-of-the-art for $R < 0.42$

What was intractable before

RLPN

Tight estimates $\qquad \mathbb{E}\left(\left|\{\mathbf{x} \in \mathbb{F}_2^{|\mathscr{P}|} \;:\; \mathbf{F}(\mathbf{x}) > T\}\right|\right)$

$$\uparrow$$

As $\left|\mathbb{F}_2^{|\mathscr{P}|}\right| = 2^{\Theta(n)}$ needs **exponential tail behavior** of $\mathbf{F}(\mathbf{x})$

$$\uparrow$$

Poisson model

However, we can prove:

Proposition [CDMT22]

If $\quad N > \dfrac{n^2}{\delta^2} \quad$ **then** $\quad \mathbf{F}(\mathbf{e}_{\mathscr{P}}) = N\delta(1 + o(1/n)) \quad$ **with probability** $\quad 1 - o(1/n)$

Goal

Theorem

There exists an algorithm that has the **same performance***, up to polynomial factors, as (double)***RLPN** *and that we can* **fully prove***.*

$\downarrow$

Make a new algorithm whose proof relies only on this proposition.

Approach

Approach :

Compute $\mathrm{poly}(n)$ score functions to recover $\mathbf{e}_{\mathscr{P}}$ and $\mathbf{e}_{\mathscr{N}}$

Making a guess:

- For each $\mathbf{x} \in \mathbb{F}_2^{\mathscr{P}}$ compute $\mathbf{g}(\mathbf{x})$, a **guess** for the value of $\mathbf{e}_{\mathscr{N}}$.
  - Property: when $\mathbf{x} = \mathbf{e}_{\mathscr{P}}$ then $\mathbf{g}(\mathbf{x}) = \mathbf{e}_{\mathscr{N}}$

$$+$$

Testing a guess:

For any $\mathbf{x}$ we can test if $\mathbf{x} = \mathbf{e}_{\mathscr{P}}$ and $\mathbf{g}(\mathbf{x}) = \mathbf{e}_{\mathscr{N}}$ in **polynomial time**.

Observation

$$\mathbf{y}^{(i)} \triangleq \begin{cases} \mathbf{y}^{(i)}_{\mathscr{P}} & = \mathbf{y}_{\mathscr{P}} \\ \mathbf{y}^{(i)}_{\mathscr{N}} & = \mathbf{y}_{\mathscr{N}} + \delta_i = \mathbf{c}_{\mathscr{N}} + \underbrace{(\mathbf{e}_{\mathscr{N}} + \delta_i)}_{\text{New Error}} \end{cases}$$

Noise of LPN sample $\langle \mathbf{y}^{(i)}, \mathbf{h} \rangle = \langle \mathbf{e}_{\mathscr{P}}, \mathbf{h}_{\mathscr{P}} \rangle + \langle \mathbf{e}_{\mathscr{N}} + \delta_i, \mathbf{h}_{\mathscr{N}} \rangle$ smaller if $\mathbf{e}_{\mathscr{N}} = 1$

$$\mathbf{F}_i(\mathbf{x}) = \sum_{\mathbf{h}} (-1)^{\langle \mathbf{y}^{(i)}, \mathbf{h} \rangle - \langle \mathbf{x}, \mathbf{h}_{\mathscr{P}} \rangle}$$

$\mathbf{F}_i$ is score when we flipped i'th bit of $\mathbf{y}_{\mathscr{N}}$

Main observation

If $(\mathbf{e}_{\mathscr{N}})_i = 1$ we expect $\mathbf{F}_i(\mathbf{e}_{\mathscr{P}}) > \mathbf{F}(\mathbf{e}_{\mathscr{P}})$

Fully provable variant of RLPN

- Computing the score functions
  - Choose $\mathscr{P}$ and $\mathscr{N}$ at random
  - Compute N dual vectors of $\mathscr{C}$ such that $|\mathbf{h}_{\mathscr{N}}| = w$
  - Compute the score functions $\mathbf{F}, \mathbf{F}_1, \mathbf{F}_2, \cdots, \mathbf{F}_{|\mathscr{N}|}$ with an FFT

- For each $\mathbf{x} \in \mathbb{F}_2^{|\mathscr{P}|}$ make a guess $\mathbf{g}(\mathbf{x}) \in \mathbb{F}_2^{|\mathscr{N}|}$ for the value of $\mathbf{e}_{\mathscr{N}}$
  - **For** $i = 1, ..., |\mathscr{N}|$:
    - $\mathbf{g}(\mathbf{x})_i \leftarrow \begin{cases} 1 & \textbf{If} \quad \mathbf{F}_i(\mathbf{x}) > \mathbf{F}(\mathbf{x}) \\ 0 & \textbf{Else} \end{cases}$

- For each $\mathbf{x} \in \mathbb{F}_2^{|\mathscr{P}|}$ test the guess $\mathbf{g}(\mathbf{x})$ and reconstruct $\mathbf{e}$
  - $\mathbf{e}_{\mathscr{P}} \leftarrow \mathbf{x}$ and $\mathbf{e}_{\mathscr{N}} \leftarrow \mathbf{g}(\mathbf{x})$
  - **If** $|\mathbf{e}| = t$ and $\mathbf{y} - \mathbf{e} \in \mathscr{C}$ **Then Return** $\mathbf{e}$

Complexity : same up to polynomial factor as RLPN

Analysis

Proposition:

If $N > \frac{\mathbf{poly}(n)}{\delta^2}$ then when $\mathbf{x} = \mathbf{e}_{\mathscr{P}}$ our guess on $\mathbf{e}_{\mathscr{N}}$ is good

Proof :

$$\mathrm{bias}\left(\langle \mathbf{e}_{\mathscr{N}} + \delta_i, \mathbf{h}_{\mathscr{N}}\rangle\right) - \underbrace{\mathrm{bias}\left(\langle \mathbf{e}_{\mathscr{N}}, \mathbf{h}_{\mathscr{N}}\rangle\right)}_{\delta} = \mathrm{poly}\,(n)\,\underbrace{\mathrm{bias}\left(\langle \mathbf{e}_{\mathscr{N}}, \mathbf{h}_{\mathscr{N}}\rangle\right)}_{\delta}$$

$$+$$

If $N > \frac{n^2}{\delta^2}$ **then** $\mathbf{F}\left(\mathbf{e}_{\mathscr{P}}\right) = N\delta(1 + o(1/n))$ **with probability** $1 - o(1/n)$

LWE problem

LWE problem

- **Input:** $(\mathbf{G}, \mathbf{y} = \mathbf{c} + \mathbf{e}) \in \mathbb{Z}_q^{k \times n} \times \mathbb{Z}_q^n$ where $\mathbf{c} \in \mathscr{C}$ and $\mathbf{e} \sim \chi^n$
- **Output:** $\mathbf{e}$

Binary Decoding (Code)	Learning with Errors (Lattice)
$\mathbb{F}_2$	$\mathbb{Z}_q$
Small Hamming weight	Small Euclidean norm

Dual attacks in **lattice-based** cryptography

Compute **small** (Euclidean norm) dual vectors of $\mathbf{h} \in \mathscr{C}^\perp$:

$\rightarrow$ By sampling short vectors in Euclidean lattice $\Lambda = \mathscr{C}^\perp + q\mathbb{Z}^n$

Key observation

$$\langle \mathbf{y}, \mathbf{h} \rangle = \langle \mathbf{c} + \mathbf{e}, \mathbf{h} \rangle = \langle \mathbf{e}, \mathbf{h} \rangle$$

is more biased toward small values of $\mathbb{Z}_q$ as $\mathbf{e}$ and $\mathbf{h}$ small

Newer lattice-based dual attacks

Matzov 2022 uses same splitting strategy:

$$\langle \mathbf{y}, \mathbf{h} \rangle = \langle \mathbf{e}_{\mathscr{P}}, \mathbf{h}_{\mathscr{P}} \rangle + \langle \mathbf{e}_{\mathscr{N}}, \mathbf{h}_{\mathscr{N}} \rangle$$

Score function

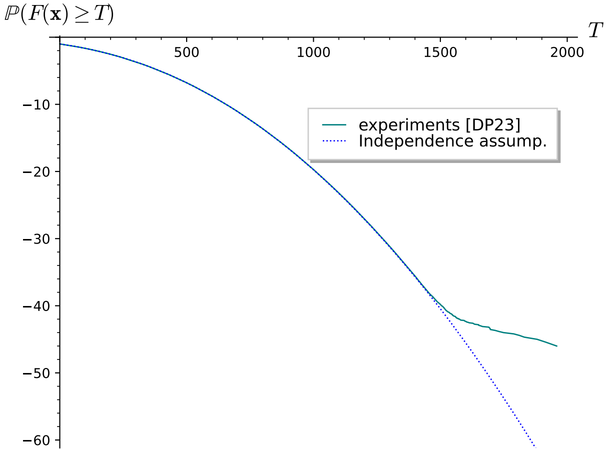
$$\mathbf{F}(\mathbf{x}) = \sum_{\mathbf{h} \in \mathscr{H}} \exp\left(\frac{2i\pi}{q} \left(\langle \mathbf{y}, \mathbf{h} \rangle - \langle \mathbf{x}, \mathbf{h}_{\mathscr{P}} \rangle \right) \right)$$

Matzov 2022 uses Modulus Switching ($\mathbb{Z}_q \to \mathbb{Z}_p$) and then an FFT as a solver.

Attack of Guo & Johansson 2021 and Matzov 2022 on **Kyber** use standard **Independence assumption** in their analysis.

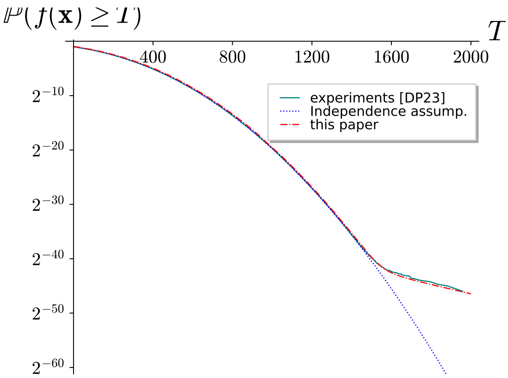
Flawed independence assumption

Ducas & Pulles 2023 $\rightarrow$ Show independence assumption are invalid



Ducas & Pulles 2023 "Does the Dual-Sieve Attack on Learning with Errors even Work?"

Accurate score prediction [CDMT24]



Dual formula

If we could apply Poisson summation:

$$F(\mathbf{x}) \approx \sum_i N_i(\Lambda) \left(\frac{w}{i}\right)^{n/2} J_{\frac{n}{2}}(2\pi\, w\, i)$$

- $N_i(\Lambda)$ number of lattice points of length i
- J_n Bessel function, related to $\widehat{1_{\leq w}}$

Model: $F(\mathbf{x}) \sim$ First term of the sum $+$ Normal

$\rightarrow$ Concurrent work with Ducas & Pulles 2023.

Dual attack of [CMST25] : Variant of Matzov 2022

Same framework as our code-based dual attacks doubleRLPN.

LPN solver

Decoding technique on $\mathbb{Z}_q$ instantiated with **Polar codes** + FFT

$\downarrow$

Using new model we show that it dents the security of **Kyber**

Results

Lead to attack against **Kyber** (using the same complexity model as Matzov)

Scheme	Required security by NIST (bits)	Matzov 2022	Our attack (bits)
Kyber-512	143	139.2	139.5
Kyber-768	207	196.1	195.1
Kyber-1024	272	262.4	259.7

Conclusion

In this thesis

- Code:
  - Significantly develop dual attacks
  - Best dual attacks improve all previous decoders for codes of rate $R < 0.42$ at GV
  - New tools (Poisson Model) and tweaks to analyze dual attacks
- Lattice:
  - New tools to analyze dual attacks
  - New attack whose analysis is backed up by experimental evidences
  - Dents the security of Kyber

Futur work:

- Asymptotic complexity exponent when using more involved way of computing dual vectors
- Non-asymptotic complexity of the attack?
- Adapt these dual attacks against scheme like CROSS?
- Can we prove exponential bound for the weight enumerator of random linear code?