Boolean Modeling and Analysis of Learning With Rounding

Jules Baudrin, Rachelle Heim Boissier, François-Xavier Standaert

Dagstuhl Seminar

Feb. 2026

Outline

Hard learning problems

Learning With Error (LWE), Learning With Rounding (LWR), Learning Parity with Noise (LPN)

and their ring/module variants.

Central importance in post-quantum cryptography

- Encryption, Key encapsulation mechanisms: CRYSTALS-Kyber, Saber
- Signatures: CRYSTALS-Dilithium, BLISS

and in symmetric cryptography:

- Essentially to build (key homomorphic) PRFs for a variety of applications.
- E.g. distributed PRFs, proxy re-encryption, updatable encryption (Boneh et al., 2013)

Learning With Errors

In a nutshell: solving a noisy linear system over a ring.

Search Learning With Errors (Regev 05)

Parameters: $q \in \mathbb{N}$, $n \in \mathbb{N}^*$, small (Gaussian) distribution χ over $\mathbb{Z}_q$, secret $\boldsymbol{x} \xleftarrow{\$} \mathbb{Z}_q^n$

Given samples from the distribution

$$\mathscr{D}^{\mathsf{LWE}} = \{ \ (\boldsymbol{a}, \langle \boldsymbol{a}, \boldsymbol{x} \rangle + e), \ \boldsymbol{a} \xleftarrow{\$} \mathbb{Z}_q^n, \ e \leftarrow \chi \ \}$$

Find $\boldsymbol{x}$.

Learning With Errors

In a nutshell: solving a noisy linear system over a ring.

Search Learning With Errors (Regev 05)

Parameters: $q \in \mathbb{N}$, $n \in \mathbb{N}^*$, small (Gaussian) distribution χ over $\mathbb{Z}_q$, secret $\boldsymbol{x} \xleftarrow{\$} \mathbb{Z}_q^n$

Given samples from the distribution

$$\mathscr{D}^{\mathsf{LWE}} = \{ \ (\boldsymbol{a}, \langle \boldsymbol{a}, \boldsymbol{x} \rangle + e), \ \boldsymbol{a} \xleftarrow{\$} \mathbb{Z}_q^n, \ e \leftarrow \chi \ \}$$

Find $\boldsymbol{x}$.

Decision LWE: distinguish from $\mathscr{D}_0 = \{(\boldsymbol{a}, r) \mid \boldsymbol{a} \xleftarrow{\$} \mathbb{Z}_q^n, r \xleftarrow{\$} \mathbb{Z}_q\}$

Learning With Errors

In a nutshell: solving a noisy linear system over a ring.

Search Learning With Errors (Regev 05)

Parameters: $q \in \mathbb{N}$, $n \in \mathbb{N}^*$, small (Gaussian) distribution χ over $\mathbb{Z}_q$, secret $\boldsymbol{x} \xleftarrow{\$} \mathbb{Z}_q^n$

Given samples from the distribution

$$\mathscr{D}^{\mathsf{LWE}} = \{ \ (\boldsymbol{a}, \langle \boldsymbol{a}, \boldsymbol{x} \rangle + e), \ \boldsymbol{a} \xleftarrow{\$} \mathbb{Z}_q^n, \ e \leftarrow \chi \ \}$$

Find $\boldsymbol{x}$.

Decision LWE: distinguish from $\mathscr{D}_0 = \{(\boldsymbol{a}, r) \mid \boldsymbol{a} \xleftarrow{\$} \mathbb{Z}_q^n, r \xleftarrow{\$} \mathbb{Z}_q\}$

- Security level is determined by n, q, and standard deviation σ of χ.
- Drawback: LWE cannot be used to build deterministic primitives such as PRFs.

Learning with Rounding

'A way of partially 'derandomizing' the LWE problem, i.e. generating errors efficiently and deterministically'.

Banerjee, Peikert, Rosen, EC' 2012.

Search Learning With Rounding

Parameters: $q \in \mathbb{N}$, $p, n \in \mathbb{N}^*$, $p < q$, rounding function $\lfloor \cdot \rceil_p : \mathbb{Z}_q \to \mathbb{Z}_p$, secret $\boldsymbol{x} \xleftarrow{\$} \mathbb{Z}_q^n$

Given samples from the distribution

$$\mathscr{D}^{\mathsf{LWR}} = \{ \ (\boldsymbol{a}, \lfloor \langle \boldsymbol{a}, \boldsymbol{x} \rangle \rceil_p), \ \boldsymbol{a} \xleftarrow{\$} \mathbb{Z}_q^n \ \}$$

Find $\boldsymbol{x}$.

Power-of-two LWR

Search Learning With Rounding

Parameters: $q \in \mathbb{N}$, $p, n \in \mathbb{N}^*$, $p < q$, rounding function $\lfloor \cdot \rceil_p : \mathbb{Z}_q \to \mathbb{Z}_p$, secret $\boldsymbol{x} \xleftarrow{\$} \mathbb{Z}_q^n$

Given samples from the distribution

$$\mathscr{D}^{\mathsf{LWR}} = \{ \ (\boldsymbol{a}, s_{\boldsymbol{a}} = \lfloor \langle \boldsymbol{a}, \boldsymbol{x} \rangle \rceil_p), \ \boldsymbol{a} \xleftarrow{\$} \mathbb{Z}_q^n \ \}$$

Find $\boldsymbol{x}$.

Power-of-two LWR

Search Learning With Rounding

Parameters: $q \in \mathbb{N}$, $p, n \in \mathbb{N}^*$, $p < q$, rounding function $\lfloor \cdot \rceil_{2^p} : \mathbb{Z}_{2^q} \to \mathbb{Z}_{2^p}$, secret $\boldsymbol{x} \xleftarrow{\$} \mathbb{Z}_{2^q}^n$

Given samples from the distribution

$$\mathscr{D}^{\mathsf{LWR}} = \{ \ (\boldsymbol{a}, s_{\boldsymbol{a}} = \lfloor \langle \boldsymbol{a}, \boldsymbol{x} \rangle \rceil_p), \ \boldsymbol{a} \xleftarrow{\$} \mathbb{Z}_{2^q}^n \ \}$$

Find $\boldsymbol{x}$.

Power-of-two LWR

Search Learning With Rounding

Parameters: $q \in \mathbb{N}$, $p, n \in \mathbb{N}^*$, $p < q$, rounding function $\lfloor \cdot \rceil_{2^p} : \mathbb{Z}_{2^q} \to \mathbb{Z}_{2^p}$, secret $\boldsymbol{x} \xleftarrow{\$} \mathbb{Z}_{2^q}^n$

Given samples from the distribution

$$\mathscr{D}^{\mathrm{LWR}} = \{ \ (\boldsymbol{a}, s_{\boldsymbol{a}} = \lfloor \langle \boldsymbol{a}, \boldsymbol{x} \rangle \rceil_p), \ \boldsymbol{a} \xleftarrow{\$} \mathbb{Z}_{2^q}^n \ \}$$

Find $\boldsymbol{x}$.

In this case:

- rounding function $\lfloor \cdot \rceil_p : \mathbb{Z}_{2^q} \to \mathbb{Z}_{2^p}$ removes the $q - p$ LSBs.
- Security level is determined by n, q and $q - p$: noise $\sim \mathit{Uniform}[-2^{q-p}, 0)$

  e.g. LightSaber: $n = 512, q - p = 3$, dPRF LaKey $n = 256, q - p = 4$.

Power-of-two LWR

Search Learning With Rounding

Parameters: $q \in \mathbb{N}$, $p, n \in \mathbb{N}^*$, $p < q$, rounding function $\lfloor \cdot \rceil_{2^p} : \mathbb{Z}_{2^q} \to \mathbb{Z}_{2^p}$, secret $\boldsymbol{x} \xleftarrow{\$} \mathbb{Z}_{2^q}^n$

Given samples from the distribution

$$\mathscr{D}^{\mathrm{LWR}} = \{ \ (\boldsymbol{a}, s_{\boldsymbol{a}} = \lfloor \langle \boldsymbol{a}, \boldsymbol{x} \rangle \rceil_p), \ \boldsymbol{a} \xleftarrow{\$} \mathbb{Z}_{2^q}^n \ \}$$

Find $\boldsymbol{x}$.

In this case:

- rounding function $\lfloor \cdot \rceil_p : \mathbb{Z}_{2^q} \to \mathbb{Z}_{2^p}$ removes the $q - p$ LSBs.
- Security level is determined by n, q and $q - p$: noise $\sim \mathit{Uniform}[-2^{q-p}, 0)$

  e.g. LightSaber: $n = 512, q - p = 3$, dPRF LaKey $n = 256, q - p = 4$.

Hardness

Theory

- LWE: Solid theoretical foundations (e.g. Brakerski et al. 13).
- LWR is as hard as LWE (asymptotic reduction, underlying assumptions).

Practice

- Parameter selection driven by best known attacks (Lattice estimator, Albrecht et al.)

'The hardness of (ring or module) LWR can be analyzed as an LWE problem, since there is no known attacks that make use of the additional structure offered by these variants'.

SABER specifications

Open question: what does a deterministic error do to (practical) security?

Linearisation attack by Arora & Ge (2011)

Parameters: $n \in \mathbb{N}^*$, Noise in set E.

Any sample $(\boldsymbol{a}, s_{\boldsymbol{a}})$, yields the following equation over $\mathbb{Z}_{2^q}$ in the unknowns $\boldsymbol{x} = (x_0, \ldots, x_{n-1})$

$$\prod_{e \in E} \left(\sum_{i=0}^{n-1} \boldsymbol{a}_i \times \boldsymbol{x}_i - e - s_{\boldsymbol{a}} \right) = 0 \,.$$

Linearisation: $\binom{n+|E|}{|E|}$ in data, $\binom{n+|E|}{|E|}^{\omega}$ in time, ω linear algebra constant.

- **LWE**: Gaussian distribution: bounded noise for a well-chosen number of samples.
- **LWR**: $|E| = 2^{q-p}$.

Linearisation attack by Arora & Ge (2011)

Parameters: $n \in \mathbb{N}^*$, Noise in set E.

Any sample (a, s_a), yields the following equation over $\mathbb{Z}_{2^q}$ in the unknowns $x = (x_0, \ldots, x_{n-1})$

$$\prod_{e \in E} \left(\sum_{i=0}^{n-1} a_i \times x_i - e - s_a \right) = 0 \,.$$

Linearisation: $\binom{n+|E|}{|E|}$ in data, $\binom{n+|E|}{|E|}^{\omega}$ in time, ω linear algebra constant.

- **LWE**: Gaussian distribution: bounded noise for a well-chosen number of samples.
- **LWR**: $|E| = 2^{q-p}$.

Our observation: inapplicable for some parameter regimes independently of the nr of samples.

Our main result: in the case of LWR, one can do an attack that 1) works 2) better.

Outline

A symmetric point of view

$$\mathbb{Z}_{2^q} \quad \simeq \quad \{0, 1, 2, \ldots, 2^q - 1\} \quad \simeq \quad \mathbb{F}_2^q$$

A symmetric point of view

$$\mathbb{Z}_{2^q} \quad \simeq \quad \{0, 1, 2, \ldots, 2^q - 1\} \quad \simeq \quad \mathbb{F}_2^q$$

- n known values $a_i \in \mathbb{Z}_{2^q}$ $\quad \simeq \quad$ nq known bits $(a_{i,j})_{\substack{0 \le i \le n-1 \\ 0 \le j \le q-1}} \in \mathbb{F}_2^{nq}$.
- n unknowns x_i in $\mathbb{Z}_{2^q}$ $\quad \simeq \quad$ nq binary unknowns $(x_{i,j})_{\substack{0 \le i \le n-1 \\ 0 \le j \le q-1}} \in \mathbb{F}_2^{nq}$.

A symmetric point of view

$$\mathbb{Z}_{2^q} \quad \simeq \quad \{0, 1, 2, \ldots, 2^q - 1\} \quad \simeq \quad \mathbb{F}_2^q$$

- n known values $\boldsymbol{a}_i \in \mathbb{Z}_{2^q}$ $\quad \simeq \quad$ nq known bits $(\boldsymbol{a}_{i,j})_{\substack{0 \leq i \leq n-1 \\ 0 \leq j \leq q-1}} \in \mathbb{F}_2^{nq}$.
- n unknowns $\boldsymbol{x}_i$ in $\mathbb{Z}_{2^q}$ $\quad \simeq \quad$ nq binary unknowns $(\boldsymbol{x}_{i,j})_{\substack{0 \leq i \leq n-1 \\ 0 \leq j \leq q-1}} \in \mathbb{F}_2^{nq}$.

The LWR function is a $(2nq, p)$-vectorial Boolean function:

$$F : \mathbb{F}_2^{nq} \times \mathbb{F}_2^{nq} \to \mathbb{F}_2^p \qquad (\boldsymbol{a}, \boldsymbol{x}) \mapsto \left\lfloor \sum_{i=0}^{n-1} \boldsymbol{a}_i \times \boldsymbol{x}_i \right\rfloor_{2^p}$$

The **LWR problem** can be studied in a "symmetric" manner ($\simeq$ weak-PRF).

Exponential notation

If $u_0, \ldots, u_{n-1}, v_0, \ldots, v_{n-1}$ are q-bit integers:

$$\boldsymbol{a}^{\boldsymbol{u}}\boldsymbol{x}^{\boldsymbol{v}} := \prod_i \prod_j a_{i,j}{}^{u_{i,j}} x_{i,j}{}^{v_{i,j}}$$

Consider $\quad M = a_{0,0}a_{0,1}a_{1,0}a_{1,1}a_{1,2}x_{0,0}x_{1,0}x_{1,1}$.

We denote it by $M = a_0^{0\mathtt{b}011}a_1^{0\mathtt{b}111}x_0^{0\mathtt{b}001}x_1^{0\mathtt{b}011} = \boldsymbol{a}^{(3,7)}\boldsymbol{x}^{(1,3)}$.

Algebraic Normal Form

Algebraic Normal Form (ANF). Any Boolean function $f : \mathbb{F}_2^s \to \mathbb{F}_2$ admits a unique multivariate polynomial form:

$$\forall x \in \mathbb{F}_2^s, \quad f(x) = \sum_{u \in \mathbb{F}_2^s} \alpha_u(f) x^u \,.$$

Product of coordinates. For any $F : \mathbb{Z}_{2^q} \to \mathbb{Z}_{2^q}$, and any $m = \sum_i m_i 2^i$, recall:

$$x \mapsto (F(x))^m := \prod_i F_i(x)^{m_i}.$$

In the following, we study (products of) coordinates of the inner product:

$$F^{m,n} : (\boldsymbol{a}, \boldsymbol{x}) \mapsto (\langle \boldsymbol{a}, \boldsymbol{x} \rangle)^m \,.$$

$\mathbb{F}_2[\boldsymbol{a}, \boldsymbol{x}]$ or $\mathbb{F}_2[\boldsymbol{a}][\boldsymbol{x}]$?

$$F^{m,n} : (\mathbb{Z}_{2^q})^n \times (\mathbb{Z}_{2^q})^n \longrightarrow \mathbb{F}_2$$
$$(\boldsymbol{a}, \boldsymbol{x}) \longmapsto (\langle \boldsymbol{a}, \boldsymbol{x} \rangle)^m .$$

If $m = 2^{q-p}$: coordinate of index $q - p$ (LSB of the sample).

$\mathbb{F}_2[a, x]$ or $\mathbb{F}_2[a][x]$?

$$F^{m,n} : (\mathbb{Z}_{2^q})^n \times (\mathbb{Z}_{2^q})^n \longrightarrow \mathbb{F}_2$$
$$(a, x) \longmapsto (\langle a, x \rangle)^m .$$

If $m = 2^{q-p}$: coordinate of index $q - p$ (LSB of the sample).

$\mathbb{F}_2[a, x]$.

$$F^{m,n} = \sum_{u,v} \alpha_{u,v} a^u x^v, \qquad \alpha_{u,v} \in \mathbb{F}_2.$$

$\mathbb{F}_2[a, x]$ or $\mathbb{F}_2[a][x]$?

$$F^{m,n} : (\mathbb{Z}_{2^q})^n \times (\mathbb{Z}_{2^q})^n \longrightarrow \mathbb{F}_2$$
$$(a, x) \longmapsto (\langle a, x \rangle)^m .$$

If $m = 2^{q-p}$: coordinate of index $q - p$ (LSB of the sample).

$\mathbb{F}_2[a, x]$.

$$F^{m,n} = \sum_{u,v} \alpha_{u,v} a^u x^v, \qquad \alpha_{u,v} \in \mathbb{F}_2.$$

$\mathbb{F}_2[a][x]$.

$$F^{m,n} = \sum_{v} \underbrace{\left(\sum_{u} \alpha_{u,v} a^u \right)}_{\alpha_v} x^v, \qquad \alpha_v \in \mathbb{F}_2[a].$$

- Cost of linearisation $\leq \#\text{monomials}^\omega = |\mathrm{Exp}_x(F^{m,n})|^\omega$ with $\mathrm{Exp}_x(F^{m,n}) = \{v \mid \alpha_v \neq 0\}$.
- Linearisation is possible only if the ANF of each α_v is known (We'll get back to it...)

Outline

Set of exponents of $F^{m,n}$

Ordered integer partitions. Let $S_k^n = \{v \in \mathbb{N}^n : \sum_{i=0}^{n-1} v_i = k\}$ be the set of n-long vectors that sum to k.

Set of exponents of $F^{m,n}$

Ordered integer partitions. Let $S_k^n = \{v \in \mathbb{N}^n : \sum_{i=0}^{n-1} v_i = k\}$ be the set of n-long vectors that sum to k.

Theorem (Exponents of $F^{m,n}$).

$$\mathrm{Exp}_{\boldsymbol{x}}(F^{m,n}) \subset \bigcup_{k=1}^{m} S_k^n .$$

i.e. if $\alpha_{\boldsymbol{v}} \neq 0$ then $\boldsymbol{v}$ must satifsy $\sum_{i=0}^{n-1} v_i \leq m$.

Set of exponents of $F^{m,n}$

Ordered integer partitions. Let $S_k^n = \{v \in \mathbb{N}^n : \sum_{i=0}^{n-1} v_i = k\}$ be the set of n-long vectors that sum to k.

Theorem (Exponents of $F^{m,n}$).

$$\mathrm{Exp}_{\boldsymbol{x}}(F^{m,n}) \subset \bigcup_{k=1}^{m} S_k^n .$$

i.e. if $\alpha_{\boldsymbol{v}} \neq 0$ then $\boldsymbol{v}$ must satifsy $\sum_{i=0}^{n-1} v_i \leq m$.

- **Proven using [Braeken & Semaev FSE05].**
  If $G(\boldsymbol{x}) = x_0 + x_1 + \cdots + x_{n-1} \in \mathbb{Z}_{2^q}$, then $\mathrm{Exp}_{\boldsymbol{x}}(G^m) = S_m^n$
- Related to the properties of **S-functions**. [MouCanIndPre09]

An attack that works and costs the same (in fact, a little less)

- **Number of monomials.** $|\mathrm{Exp}_{x}(F^{m,n})| \leq |\bigcup_{k=1}^{m} S_k^n| = \binom{n+m}{m}$.
  Open question: we conjecture equality when m is a power of two.
- **Degree.** $\deg(F^{m,n}) \leq m$. Equality holds when $m \leq n$.

An attack that works and costs the same (in fact, a little less)

- **Number of monomials.** $|\mathrm{Exp}_{\boldsymbol{x}}(F^{m,n})| \leq |\bigcup_{k=1}^{m} S_k^n| = \binom{n+m}{m}$.
  Open question: we conjecture equality when m is a power of two.
- **Degree.** $\deg(F^{m,n}) \leq m$. Equality holds when $m \leq n$.

- Recall over $\mathbb{Z}_{2^q}$ (when applicable): $\binom{n+2^{q-p}}{2^{q-p}}$ monomials. [Arora & Ge]

An attack that works and costs the same (in fact, a little less)

- **Number of monomials.** $|\mathrm{Exp}_{\boldsymbol{x}}(F^{m,n})| \leq |\bigcup_{k=1}^{m} S_k^n| = \binom{n+m}{m}$.
  Open question: we conjecture equality when m is a power of two.
- **Degree.** $\deg(F^{m,n}) \leq m$. Equality holds when $m \leq n$.

- Recall over $\mathbb{Z}_{2^q}$ (when applicable): $\binom{n+2^{q-p}}{2^{q-p}}$ monomials. [Arora & Ge]
- If we use the LSB of the sample ($m = 2^{q-p}$), same number of monomials (Surprising!).

An attack that works and costs the same (in fact, a little less)

- **Number of monomials.** $|\mathrm{Exp}_{\boldsymbol{x}}(F^{m,n})| \leq |\bigcup_{k=1}^{m} S_k^n| = \binom{n+m}{m}$.
  Open question: we conjecture equality when m is a power of two.
- **Degree.** $\deg(F^{m,n}) \leq m$. Equality holds when $m \leq n$.

- Recall over $\mathbb{Z}_{2^q}$ (when applicable): $\binom{n+2^{q-p}}{2^{q-p}}$ monomials. [Arora & Ge]
- If we use the LSB of the sample ($m = 2^{q-p}$), same number of monomials (Surprising!).
- Working over $\mathbb{F}_2$ rather than $\mathbb{Z}_{2^q}$: we gain at least 2^{q-p}.
  1. it's a field.
  2. operations are cheaper

The general strategy

1. With $\binom{n+2^{q-p}}{2^{q-p}}$ samples, recover the $q-p$ LSBs of each x_i by linearisation.
2. Observe that each bit of $\langle \boldsymbol{a}, \boldsymbol{x} \rangle$ satisfies $b_j = \sum_{i=0}^{n-1} \boldsymbol{a}_{i,j} \boldsymbol{x}_{i,j} + c_j$ where c_j is a carry.
3. Compute c_j from the known $a_{i,k}$'s and recovered $x_{i,k}$'s, $k < j$.
4. Solve a linear system in $\boldsymbol{x}_{i,j}$ only.
5. Repeat for increasing j.

The cost of steps 2 to 5 is negligible before the cost of step 1.

Effective computation of the ANF

We improve Arora & Ge generically...

Effective computation of the ANF

We improve Arora & Ge generically. . . as long as we can compute the ANF.

Effective computation of the ANF

We improve Arora & Ge generically... as long as we can compute the ANF.

This is *a priori* hard:

- Möbius transform is out of reach (the LUT cannot be fully computed/stored).
- Direct computations using recursive formulas seem hopeless.

Our results

The ANF can be 'understood' for arbitrary large n and for m up to 16 by:

- storing the ANF for $n = m$.
- computing properties for arbitrary large n.

Additional improvement of the attack

Default linearisation: one auxiliary variable per monomial x^{v} in the ANF of $F^{m,n}$.

Example.

$$F(a, x) = \alpha_{v} x^{v} + \alpha_{v'} x^{v'} + \alpha_{v''} x^{v''} + \cdots \qquad \alpha_{v}, \alpha_{v'}, \alpha_{v''} \in \mathbb{F}_2[a]$$

- If $\alpha_{v} = \alpha_{v'} = \alpha_{v''}$, then x^{v}, $x^{v'}$, $x^{v''}$ appear/vanish together for any value of a.
- It makes sense to introduce $y = x^{v} + x^{v'} + x^{v''}$.

Additional improvement of the attack

Default linearisation: one auxiliary variable per monomial $\boldsymbol{x}^{\boldsymbol{v}}$ in the ANF of $F^{m,n}$.

Example.

$$F(\boldsymbol{a}, \boldsymbol{x}) = \alpha_{\boldsymbol{v}} \boldsymbol{x}^{\boldsymbol{v}} + \alpha_{\boldsymbol{v}'} \boldsymbol{x}^{\boldsymbol{v}'} + \alpha_{\boldsymbol{v}''} \boldsymbol{x}^{\boldsymbol{v}''} + \cdots \qquad \alpha_{\boldsymbol{v}}, \alpha_{\boldsymbol{v}'}, \alpha_{\boldsymbol{v}''} \in \mathbb{F}_2[\boldsymbol{a}]$$

- If $\alpha_{\boldsymbol{v}} = \alpha_{\boldsymbol{v}'} = \alpha_{\boldsymbol{v}''}$, then $\boldsymbol{x}^{\boldsymbol{v}}$, $\boldsymbol{x}^{\boldsymbol{v}'}$, $\boldsymbol{x}^{\boldsymbol{v}''}$ appear/vanish together for any value of $\boldsymbol{a}$.
- It makes sense to introduce $y = \boldsymbol{x}^{\boldsymbol{v}} + \boldsymbol{x}^{\boldsymbol{v}'} + \boldsymbol{x}^{\boldsymbol{v}''}$.

- Less auxiliary variables: improved linearisation (data and time).
- Ideally: compute the rank and a basis for $\{\alpha_{\boldsymbol{v}}, \boldsymbol{v} \in \mathrm{Exp}(F^{m,n})\}$. Work in progress

Additional improvement of the attack

Default linearisation: one auxiliary variable per monomial $\boldsymbol{x}^{\boldsymbol{v}}$ in the ANF of $F^{m,n}$.

Example.

$$F(\boldsymbol{a}, \boldsymbol{x}) = \alpha_{\boldsymbol{v}} \boldsymbol{x}^{\boldsymbol{v}} + \alpha_{\boldsymbol{v}'} \boldsymbol{x}^{\boldsymbol{v}'} + \alpha_{\boldsymbol{v}''} \boldsymbol{x}^{\boldsymbol{v}''} + \cdots \qquad \alpha_{\boldsymbol{v}}, \alpha_{\boldsymbol{v}'}, \alpha_{\boldsymbol{v}''} \in \mathbb{F}_2[\boldsymbol{a}]$$

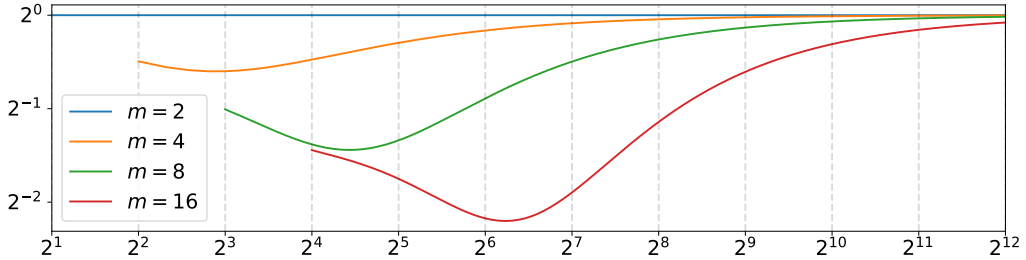
- If $\alpha_{\boldsymbol{v}} = \alpha_{\boldsymbol{v}'} = \alpha_{\boldsymbol{v}''}$, then $\boldsymbol{x}^{\boldsymbol{v}}$, $\boldsymbol{x}^{\boldsymbol{v}'}$, $\boldsymbol{x}^{\boldsymbol{v}''}$ appear/vanish together for any value of $\boldsymbol{a}$.
- It makes sense to introduce $y = \boldsymbol{x}^{\boldsymbol{v}} + \boldsymbol{x}^{\boldsymbol{v}'} + \boldsymbol{x}^{\boldsymbol{v}''}$.

- Less auxiliary variables: improved linearisation (data and time).
- Ideally: compute the rank and a basis for $\{\alpha_{\boldsymbol{v}}, \boldsymbol{v} \in \mathrm{Exp}(F^{m,n})\}$. Work in progress

- In practice, we compute the generating family $Q^{m,n}$ made of distinct $\alpha_{\boldsymbol{v}}$'s. $m \leq 16$.
- The average sparsity for both families (monomials, $Q^{m,n}$). $m \leq 8$.

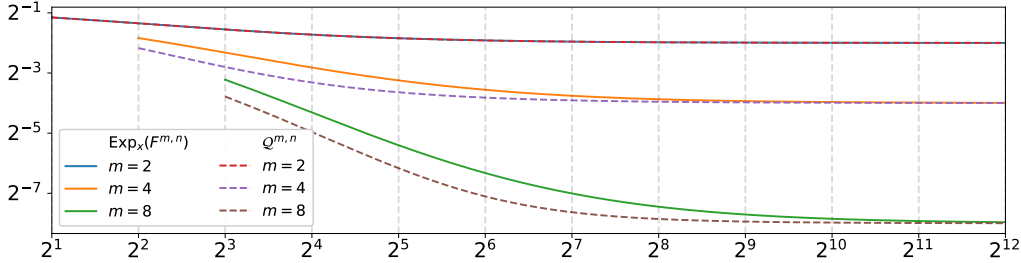
Our results: Upper bound on the rank

Ratio $|Q^{m,n}| / \binom{n+m}{m} - 1$ as a function of n, for $n \in [\![m, 4096]\!]$.



Our results: Sparsity

Average fraction of terms in a random equation $F_{\boldsymbol{a}}^{m,n}(\boldsymbol{x})$ as a function of n.



Improvements over Arora-Ge

Comparison to the linearisation attack by Arora & Ge using $\omega = 3$.
and cost of modular addition/multiplication $m = 2^{q-p}$.

$m = 2^{q-p}$	n	Arora-Ge	Our work ($\leq$ rank only)	Our work ($\leq$ rank and sparsity)
8	64	$2^{103.4}$	$2^{97.8}$	$2^{87.2}$
	128	$2^{126.3}$	$2^{121.8}$	$2^{110.8}$
	256	$2^{149.7}$	$2^{145.9}$	$2^{134.7}$
16	64	$2^{167.7}$	$2^{157.2}$	Non-available
	128	$2^{211.7}$	$2^{202.00}$	Non-available
	256	$2^{257.5}$	$2^{250.1}$	Non-available

Outline

Overview

Our result

The ANF (and additionnal properties) can be understood for arbitrary large n and for m up to 16.

NB: these m's are used in practice e.g. $m = 8$ for SABER, $m = 16$ for LaKEY.

We take advantage of symmetries due to the commutativity of modular addition.

Observe that:

- $\langle (a_0, a_1), (x_0, x_1) \rangle = a_0 x_0 + a_1 x_1 = a_1 x_1 + a_0 x_0 = \langle (a_1, a_0), (x_1, x_0) \rangle$.

Overview

Our result

The ANF (and additionnal properties) can be understood for arbitrary large n and for m up to 16.

NB: these m's are used in practice e.g. $m = 8$ for SABER, $m = 16$ for LaKEY.

We take advantage of symmetries due to the commutativity of modular addition.

Observe that:

- $\langle (a_0, a_1), (x_0, x_1) \rangle = a_0 x_0 + a_1 x_1 = a_1 x_1 + a_0 x_0 = \langle (a_1, a_0), (x_1, x_0) \rangle$.
- More generally for any n and any permutation $\sigma \in \mathfrak{S}_n$, $\langle \boldsymbol{a}, \boldsymbol{x} \rangle = \langle \sigma \cdot \boldsymbol{a}, \sigma \cdot \boldsymbol{x} \rangle$.

Overview

Our result

The ANF (and additionnal properties) can be understood for arbitrary large n and for m up to 16.

NB: these m's are used in practice e.g. $m = 8$ for SABER, $m = 16$ for LaKEY.

We take advantage of symmetries due to the commutativity of modular addition.

Observe that:

- $\langle (a_0, a_1), (x_0, x_1) \rangle = a_0 x_0 + a_1 x_1 = a_1 x_1 + a_0 x_0 = \langle (a_1, a_0), (x_1, x_0) \rangle$.
- More generally for any n and any permutation $\sigma \in \mathfrak{S}_n$, $\langle \boldsymbol{a}, \boldsymbol{x} \rangle = \langle \sigma \cdot \boldsymbol{a}, \sigma \cdot \boldsymbol{x} \rangle$.

Observe that:

- $\langle (a_0, a_1, 0), (x_0, x_1, 0) \rangle = a_0 x_0 + a_1 x_1 = \langle (a_0, a_1), (x_0, x_1) \rangle$.

Overview

Our result

The ANF (and additionnal properties) can be understood for arbitrary large n and for m up to 16.

NB: these m's are used in practice e.g. $m = 8$ for SABER, $m = 16$ for LaKEY.

We take advantage of symmetries due to the commutativity of modular addition.

Observe that:

- $\langle (a_0, a_1), (x_0, x_1) \rangle = a_0 x_0 + a_1 x_1 = a_1 x_1 + a_0 x_0 = \langle (a_1, a_0), (x_1, x_0) \rangle$.
- More generally for any n and any permutation $\sigma \in \mathfrak{S}_n$, $\langle \boldsymbol{a}, \boldsymbol{x} \rangle = \langle \sigma \cdot \boldsymbol{a}, \sigma \cdot \boldsymbol{x} \rangle$.

Observe that:

- $\langle (a_0, a_1, 0), (x_0, x_1, 0) \rangle = a_0 x_0 + a_1 x_1 = \langle (a_0, a_1), (x_0, x_1) \rangle$.
- More generally for any $n \leq n'$, $\langle \boldsymbol{a} || 0^{n'-n}, \boldsymbol{x} || 0^{n'-n} \rangle_{n'} = \langle \boldsymbol{a}, \boldsymbol{x} \rangle_n$.

Reduction to a system of representatives

- The group $\mathfrak{S}_n$ acts on vectors of length n: $\sigma \cdot \boldsymbol{u} := (u_{\sigma^{-1}(0)}, \ldots, u_{\sigma^{-1}(n-1)})$.
- $F(\boldsymbol{a}, \boldsymbol{x})$ is $\mathfrak{S}_n$-invariant if $F(\boldsymbol{a}, \boldsymbol{x}) = F(\sigma \cdot \boldsymbol{a}, \sigma \cdot \boldsymbol{x})$ for all $\sigma \in \mathfrak{S}_n$.

Reduction to a system of representatives

- The group $\mathfrak{S}_n$ acts on vectors of length n: $\sigma \cdot \boldsymbol{u} \coloneqq (u_{\sigma^{-1}(0)}, \ldots, u_{\sigma^{-1}(n-1)})$.
- $F(\boldsymbol{a}, \boldsymbol{x})$ is $\mathfrak{S}_n$-invariant if $F(\boldsymbol{a}, \boldsymbol{x}) = F(\sigma \cdot \boldsymbol{a}, \sigma \cdot \boldsymbol{x})$ for all $\sigma \in \mathfrak{S}_n$.

Example. The function $F : (\boldsymbol{a}, \boldsymbol{x}) \mapsto \boldsymbol{a}^{(1,0)}\boldsymbol{x}^{(3,0)} + \boldsymbol{a}^{(0,1)}\boldsymbol{x}^{(0,3)}$ is $\mathfrak{S}_2$-invariant (not symmetric!)

Indeed, $\mathfrak{S}_2 = \{\mathrm{id}, (0\ 1)\}$ and $(0\ 1) \cdot F = F((0\ 1) \cdot \boldsymbol{a}, (0\ 1) \cdot \boldsymbol{x}) = F(\boldsymbol{a}, \boldsymbol{x})$.

Reduction to a system of representatives

- The group $\mathfrak{S}_n$ acts on vectors of length n: $\sigma \cdot \boldsymbol{u} := (u_{\sigma^{-1}(0)}, \ldots, u_{\sigma^{-1}(n-1)})$.
- $F(\boldsymbol{a}, \boldsymbol{x})$ is $\mathfrak{S}_n$-invariant if $F(\boldsymbol{a}, \boldsymbol{x}) = F(\sigma \cdot \boldsymbol{a}, \sigma \cdot \boldsymbol{x})$ for all $\sigma \in \mathfrak{S}_n$.

Example. The function $F : (\boldsymbol{a}, \boldsymbol{x}) \mapsto \boldsymbol{a}^{(1,0)}\boldsymbol{x}^{(3,0)} + \boldsymbol{a}^{(0,1)}\boldsymbol{x}^{(0,3)}$ is $\mathfrak{S}_2$-invariant (not symmetric!)

Indeed, $\mathfrak{S}_2 = \{\mathrm{id}, (0\ 1)\}$ and $(0\ 1) \cdot F = F((0\ 1) \cdot \boldsymbol{a}, (0\ 1) \cdot \boldsymbol{x}) = F(\boldsymbol{a}, \boldsymbol{x})$.

- $\alpha_{(1,0),(3,0)} = \alpha_{(0,1),(0,3)}$.
- $\alpha_{(3,0)} = \boldsymbol{a}^{(1,0)} \neq \boldsymbol{a}^{(0,1)} = \alpha_{(0,3)}$.

Reduction to a system of representatives

- The group $\mathfrak{S}_n$ acts on vectors of length n: $\sigma \cdot \boldsymbol{u} := (u_{\sigma^{-1}(0)}, \ldots, u_{\sigma^{-1}(n-1)})$.
- $F(\boldsymbol{a}, \boldsymbol{x})$ is $\mathfrak{S}_n$-invariant if $F(\boldsymbol{a}, \boldsymbol{x}) = F(\sigma \cdot \boldsymbol{a}, \sigma \cdot \boldsymbol{x})$ for all $\sigma \in \mathfrak{S}_n$.

Example. The function $F : (\boldsymbol{a}, \boldsymbol{x}) \mapsto \boldsymbol{a}^{(1,0)}\boldsymbol{x}^{(3,0)} + \boldsymbol{a}^{(0,1)}\boldsymbol{x}^{(0,3)}$ is $\mathfrak{S}_2$-invariant (not symmetric!)

Indeed, $\mathfrak{S}_2 = \{\mathrm{id}, (0\ 1)\}$ and $(0\ 1) \cdot F = F((0\ 1) \cdot \boldsymbol{a}, (0\ 1) \cdot \boldsymbol{x}) = F(\boldsymbol{a}, \boldsymbol{x})$.

- $\alpha_{(1,0),(3,0)} = \alpha_{(0,1),(0,3)}$.
- $\alpha_{(3,0)} = \boldsymbol{a}^{(1,0)} \neq \boldsymbol{a}^{(0,1)} = \alpha_{(0,3)}$.

For any m, n, $F^{m,n}$ is $\mathfrak{S}_n$-invariant. $\mathfrak{S}_n$-invariance is equivalent to:

- $\forall \boldsymbol{u}, \boldsymbol{v}, \sigma, \quad \alpha_{(\sigma\cdot\boldsymbol{u}, \sigma\cdot\boldsymbol{v})} = \alpha_{(\boldsymbol{u},\boldsymbol{v})}$ $\qquad \alpha_{\boldsymbol{u},\boldsymbol{v}} \in \mathbb{F}_2$
- and to $\forall \boldsymbol{u}, \sigma, \quad \sigma \cdot \alpha_{\boldsymbol{v}} = \alpha_{\sigma^{-1}\cdot\boldsymbol{v}}$ $\qquad \alpha_{\boldsymbol{v}} \in \mathbb{F}_2[\boldsymbol{a}]$

Allows to represent the ANF in a compact way.

Scaling

How do we get results for arbitrary n?

Example. Let's look at the $\mathfrak{S}_2$-invariant function

$$F^{(2)}((a_0, a_1), (x_0, x_1)) = \boldsymbol{a}^{(1,0)}\boldsymbol{x}^{(3,0)} + \boldsymbol{a}^{(0,1)}\boldsymbol{x}^{(0,3)}.$$

then there exists a $\approx$ unique $\mathfrak{S}_3$-invariant function $F^{(3)}$ such that

$$((a_0, a_1), (x_0, x_1)) \mapsto F^{(3)}((a_0, a_1, 0), (x_0, x_1, 0))$$

is equal to $F^{(2)}$:

$$F^{(3)}((a_0, a_1, a_2), (x_0, x_1, x_2)) \mapsto \boldsymbol{a}^{(1,0,0)}\boldsymbol{x}^{(3,0,0)} + \boldsymbol{a}^{(0,1,0)}\boldsymbol{x}^{(0,3,0)} + \boldsymbol{a}^{(0,0,1)}\boldsymbol{x}^{(0,0,3)}.$$

(Properties of) $F^{m,n}$ can be derived from $F^{m,m}$.

Outline

Group actions

Let $\mathbb{G}$ be a group acting on a set E. Let $x \in E$.

- **Orbit.** $\mathrm{Orb}(x) \coloneqq \{\sigma \cdot x,\ \sigma \in \mathbb{G}\} \subset E$.
- **Stabilizer.** $\mathrm{Stab}(x) \coloneqq \{\sigma,\ \sigma \cdot x = x\} < \mathbb{G}$.

Example. Let $\boldsymbol{v} = (3, 1, 1)$. Then $\mathrm{Orb}(\boldsymbol{v}) = \{(3, 1, 1), (1, 3, 1), (1, 1, 3)\}$ and $\mathrm{Stab}(\boldsymbol{v}) = \{\mathrm{id}, (1\ 2)\}$.

Important properties.

- The set of orbits $\{\mathrm{Orb}(x),\ x \in E\}$ is a partition of E.
- It induces an equivalence relation: $x \sim x'$ if and only if $x' \in \mathrm{Orb}(x)$.
- For any $x \in E$,

$$|\mathrm{Orb}(x)| \cdot |\mathrm{Stab}(x)| = |\mathbb{G}| \ .$$

Effective computation of the ANF

Recall: S_m^n ordered partitions of length n of m.

$$F^{m,n}(\boldsymbol{a},\boldsymbol{x}) = \langle \boldsymbol{a},\boldsymbol{x}\rangle^m = \Big(\sum_i a_i \times x_i\Big)^m \underset{[BraSem05]}{=} \sum_{\boldsymbol{c}\in S_m^n} \prod_i (a_i \times x_i)^{c_i}\,.$$

Let $\mathscr{C}_m^n$ be a system of representatives (unordered partitions):

$$F^{m,n} = \ldots = \sum_{\boldsymbol{c}\in\mathscr{C}_m^n} \underbrace{\sum_{\boldsymbol{c}'\in \mathrm{Orb}(\boldsymbol{c})} \overbrace{\prod_i (a_i \times x_i)^{c_i'}}^{H_{\boldsymbol{c}'}}}_{G_{\boldsymbol{c}}}$$

We observe:

- $G_{\boldsymbol{c}}$ is $\mathfrak{S}_n$-invariant.
- $H_{\sigma\cdot\boldsymbol{c}} = \sigma^{-1}\cdot H_{\boldsymbol{c}}\,.$

Effective computation of the ANF

Theorem. Let $\boldsymbol{c} \in \mathbb{Z}_{2^q}^n$, $(\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{Z}_{2^q}^n \times \mathbb{Z}_{2^q}^n$. Define $E_{\boldsymbol{c}}(\boldsymbol{u}, \boldsymbol{v}) \coloneqq \mathrm{Exp}(H_{\boldsymbol{c}}) \cap \mathrm{Orb}(\boldsymbol{u}, \boldsymbol{v})$.

$$\alpha_{(\boldsymbol{u},\boldsymbol{v})}(G_{\boldsymbol{c}}) \quad = \quad \frac{|E_{\boldsymbol{c}}(\boldsymbol{u}, \boldsymbol{v})|\, n!}{|\mathrm{Stab}(\boldsymbol{c})|\, |\mathrm{Orb}(\boldsymbol{u}, \boldsymbol{v})|} \mod 2\,.$$

Effective computation of the ANF

Theorem. Let $\boldsymbol{c} \in \mathbb{Z}_{2^q}^n$, $(\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{Z}_{2^q}^n \times \mathbb{Z}_{2^q}^n$. Define $E_{\boldsymbol{c}}(\boldsymbol{u}, \boldsymbol{v}) \coloneqq \mathrm{Exp}(H_{\boldsymbol{c}}) \cap \mathrm{Orb}(\boldsymbol{u}, \boldsymbol{v})$.

$$\alpha_{(\boldsymbol{u},\boldsymbol{v})}(G_{\boldsymbol{c}}) \quad = \quad \frac{|E_{\boldsymbol{c}}(\boldsymbol{u}, \boldsymbol{v})|\, n!}{|\mathrm{Stab}(\boldsymbol{c})|\, |\mathrm{Orb}(\boldsymbol{u}, \boldsymbol{v})|} \mod 2\,.$$

We provide algorithms (and implementations) to:

(i) compute the ANF of $H_{\boldsymbol{c}^\star}$ for $\boldsymbol{c}^\star \in \mathscr{C}_m^n$.

(ii) compute a SoR of the ANF of $G_{\boldsymbol{c}}$ (Theorem). $\{(\boldsymbol{u}, \boldsymbol{v})^\star, \alpha_{(\boldsymbol{u},\boldsymbol{v})^\star}(G_{\boldsymbol{c}}) = 1\}$.

Effective computation of the ANF

Theorem. Let $\boldsymbol{c} \in \mathbb{Z}_{2^q}^n$, $(\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{Z}_{2^q}^n \times \mathbb{Z}_{2^q}^n$. Define $E_{\boldsymbol{c}}(\boldsymbol{u}, \boldsymbol{v}) \coloneqq \mathrm{Exp}(H_{\boldsymbol{c}}) \cap \mathrm{Orb}(\boldsymbol{u}, \boldsymbol{v})$.

$$\alpha_{(\boldsymbol{u},\boldsymbol{v})}(G_{\boldsymbol{c}}) \quad = \quad \frac{|E_{\boldsymbol{c}}(\boldsymbol{u}, \boldsymbol{v})|\, n!}{|\mathrm{Stab}(\boldsymbol{c})|\, |\mathrm{Orb}(\boldsymbol{u}, \boldsymbol{v})|} \mod 2\,.$$

We provide algorithms (and implementations) to:

(i) compute the ANF of $H_{\boldsymbol{c}^\star}$ for $\boldsymbol{c}^\star \in \mathscr{C}_m^n$.

(ii) compute a SoR of the ANF of $G_{\boldsymbol{c}}$ (Theorem). $\{(\boldsymbol{u}, \boldsymbol{v})^\star, \alpha_{(\boldsymbol{u},\boldsymbol{v})^\star}(G_{\boldsymbol{c}}) = 1\}$.

(iii) compute a SoR of the ANF of $F^{m,n}$ from the $G_{\boldsymbol{c}}$'s. $\{(\boldsymbol{u}, \boldsymbol{v})^\star, \alpha_{(\boldsymbol{u},\boldsymbol{v})^\star}(F^{m,n}) = 1\}$.

Effective computation of the ANF

Theorem. Let $\boldsymbol{c} \in \mathbb{Z}_{2^q}^n$, $(\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{Z}_{2^q}^n \times \mathbb{Z}_{2^q}^n$. Define $E_{\boldsymbol{c}}(\boldsymbol{u}, \boldsymbol{v}) \coloneqq \mathrm{Exp}(H_{\boldsymbol{c}}) \cap \mathrm{Orb}(\boldsymbol{u}, \boldsymbol{v})$.

$$\alpha_{(\boldsymbol{u},\boldsymbol{v})}(G_{\boldsymbol{c}}) \quad = \quad \frac{|E_{\boldsymbol{c}}(\boldsymbol{u}, \boldsymbol{v})|\, n!}{|\mathrm{Stab}(\boldsymbol{c})|\, |\mathrm{Orb}(\boldsymbol{u}, \boldsymbol{v})|} \mod 2\,.$$

We provide algorithms (and implementations) to:

(i) compute the ANF of $H_{\boldsymbol{c}^\star}$ for $\boldsymbol{c}^\star \in \mathscr{C}_m^n$.

(ii) compute a SoR of the ANF of $G_{\boldsymbol{c}}$ (Theorem). $\{(\boldsymbol{u}, \boldsymbol{v})^\star, \alpha_{(\boldsymbol{u},\boldsymbol{v})^\star}(G_{\boldsymbol{c}}) = 1\}$.

(iii) compute a SoR of the ANF of $F^{m,n}$ from the $G_{\boldsymbol{c}}$'s. $\{(\boldsymbol{u}, \boldsymbol{v})^\star, \alpha_{(\boldsymbol{u},\boldsymbol{v})^\star}(F^{m,n}) = 1\}$.

(iv) compute $\{(\boldsymbol{v}^\star, \alpha_{\boldsymbol{v}^\star}(F^{m,n}) \neq 0)\}$. (non-trivial from (iii)).

Effective computation of the ANF

Theorem. Let $\boldsymbol{c} \in \mathbb{Z}_{2^q}^n$, $(\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{Z}_{2^q}^n \times \mathbb{Z}_{2^q}^n$. Define $E_{\boldsymbol{c}}(\boldsymbol{u}, \boldsymbol{v}) \coloneqq \mathrm{Exp}(H_{\boldsymbol{c}}) \cap \mathrm{Orb}(\boldsymbol{u}, \boldsymbol{v})$.

$$\alpha_{(\boldsymbol{u},\boldsymbol{v})}(G_{\boldsymbol{c}}) \quad = \quad \frac{|E_{\boldsymbol{c}}(\boldsymbol{u}, \boldsymbol{v})|\, n!}{|\mathrm{Stab}(\boldsymbol{c})|\, |\mathrm{Orb}(\boldsymbol{u}, \boldsymbol{v})|} \mod 2\,.$$

We provide algorithms (and implementations) to:

(i) compute the ANF of $H_{\boldsymbol{c}^\star}$ for $\boldsymbol{c}^\star \in \mathscr{C}_m^n$.

(ii) compute a SoR of the ANF of $G_{\boldsymbol{c}}$ (Theorem). $\{(\boldsymbol{u}, \boldsymbol{v})^\star, \alpha_{(\boldsymbol{u},\boldsymbol{v})^\star}(G_{\boldsymbol{c}}) = 1\}$.

(iii) compute a SoR of the ANF of $F^{m,n}$ from the $G_{\boldsymbol{c}}$'s. $\{(\boldsymbol{u}, \boldsymbol{v})^\star, \alpha_{(\boldsymbol{u},\boldsymbol{v})^\star}(F^{m,n}) = 1\}$.

(iv) compute $\{(\boldsymbol{v}^\star, \alpha_{\boldsymbol{v}^\star}(F^{m,n}) \neq 0)\}$. (non-trivial from (iii)).

(v) compute $\{\alpha \in \mathrm{Exp}_{\boldsymbol{x}}(F^{m,n})\}/\sim \subset \mathbb{F}_2[\boldsymbol{a}]$. (even less trivial).

Ex. of scaling: number of monomials

Recall. For each random $\boldsymbol{a}$, if we can compute the ANF of $F_{\boldsymbol{a}}^{m,n}$, we obtain an equation in the secret $\boldsymbol{x}$:

$$F^{m,n}(\boldsymbol{a}, \boldsymbol{x}) = \sum_{\boldsymbol{v} \in (\mathbb{F}_2^q)^n} \alpha_{\boldsymbol{v}}(F^{m,n})(\boldsymbol{a})\ \boldsymbol{x}^{\boldsymbol{v}} = \text{LSB of the sample}\,.$$

Ex. of scaling: number of monomials

Recall. For each random $\boldsymbol{a}$, if we can compute the ANF of $F_{\boldsymbol{a}}^{m,n}$, we obtain an equation in the secret $\boldsymbol{x}$:

$$F^{m,n}(\boldsymbol{a}, \boldsymbol{x}) = \sum_{\boldsymbol{v} \in (\mathbb{F}_2^q)^n} \alpha_{\boldsymbol{v}}(F^{m,n})(\boldsymbol{a})\, \boldsymbol{x}^{\boldsymbol{v}} = \text{LSB of the sample}\,.$$

- Our algorithm (iv) returns $\{(\boldsymbol{v}^\star, \alpha_{\boldsymbol{v}^\star} \neq 0)\}$.

$$\text{Nr of monomials } = |\{\boldsymbol{v} \ : \ \alpha_{\boldsymbol{v}} \neq 0\}| = \textstyle\sum_{\boldsymbol{v}^\star,\ \alpha_{\boldsymbol{v}^\star} \neq 0} \mathrm{Orb}(\alpha_{\boldsymbol{v}^\star}) \binom{n}{|\mathrm{Supp}(\boldsymbol{v}^\star)|}$$

Ex. of scaling: number of monomials

Recall. For each random a, if we can compute the ANF of $F_a^{m,n}$, we obtain an equation in the secret x:

$$F^{m,n}(a,x) = \sum_{v \in (\mathbb{F}_2^q)^n} \alpha_v(F^{m,n})(a)\, x^v = \text{LSB of the sample}.$$

- Our algorithm (iv) returns $\{(v^\star, \alpha_{v^\star} \neq 0)\}$.

$$\text{Nr of monomials } = |\{v \ : \ \alpha_v \neq 0\}| = \sum_{v^\star,\ \alpha_{v^\star} \neq 0} \mathrm{Orb}(\alpha_{v^\star}) \binom{n}{|\mathrm{Supp}(v^\star)|}$$

In practice, when m is a power of two, this is always equal to the upper bound $\binom{n+m}{m} - 1$.

- In general, scaling is not the hard part (even for other properties).

Conclusion

Results

- Deterministic noise impacts attacks.
- Generic improvement (and "correction") of Arora & Ge:
  - same number of monomials + working over $\mathbb{F}_2$.
  - computation of additional parameters.

Many open questions

- Understanding the ANF even better (e.g. rank).
- Help us solve this system (with less data)!
  - E.g. super structured ANF guides guess-and-solve strategies.
- More applications of group actions.