Correlated Pseudorandomness from the Hardness of Quasi-Abelian Decoding

Maxime Bombar[1] Geoffroy Couteau [2] Alain Couvreur[1]
Clément Ducros[3]

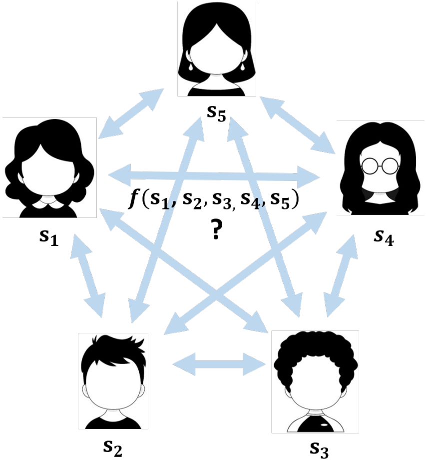
[1]INRIA, Institut Polytechnique de Paris

[2]CNRS, IRIF, Université de Paris

[3]Université de Paris, IRIF, INRIA

24 August, 2023

MPC



Correlated Randomness.

Random Correlations

A trusted dealer gives additional correlations to the players. Some examples, for α the input of Alice and β the input of Bob.

- Oblivious Transfer $\alpha = (a_0, a_1), \beta = (b, a_b)$

Correlated Randomness.

Random Correlations

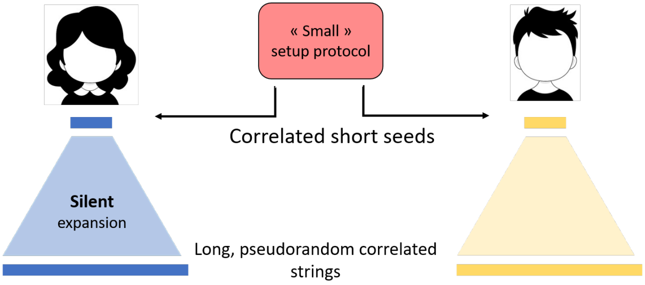
A trusted dealer gives additional correlations to the players. Some examples, for α the input of Alice and β the input of Bob.

- Oblivious Transfer $\alpha = (a_0, a_1), \beta = (b, a_b)$
- Oblivious Linear Evaluation $\alpha = (u, v), \beta = (\Delta, w = \Delta \cdot u + v)$.
  Can be rewritten as: $\alpha = (u, [\![\Delta \cdot u]\!]_0), \beta = (\Delta, [\![\Delta \cdot u]\!]_1)$

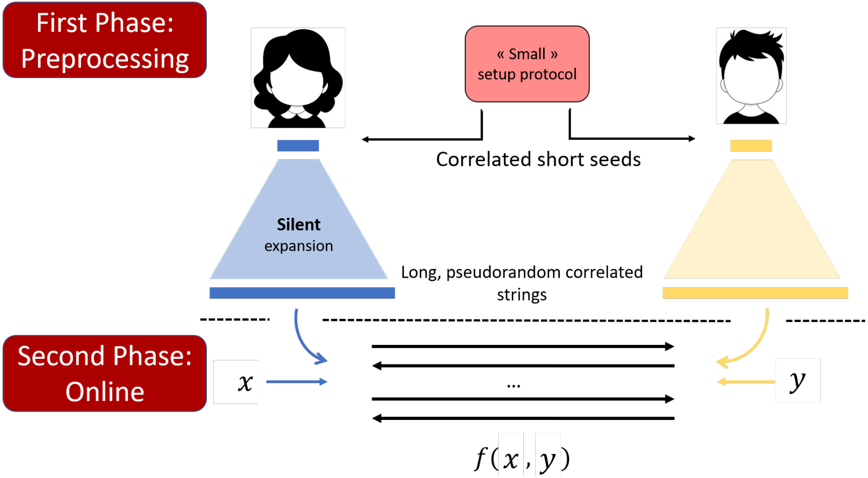
Pseudorandom Correlation Generator

Pseudorandom Correlation Generator

A PCG is a functionality that shares short correlated seeds with the parties, and that the parties can locally extend into long strings of the target correlation.



MPC with Silent Preprocessing



State of the art on silent PCG

Underlying assumption	Correlation	Programmability	Correlations per second	Field size?
Syndrome Decoding for Expand and Accumulate Code [BCG+22] Expand and Convolute Codes[RRT23]	OT	x	10^7	q $= 2$
Syndrome Decoding for Silver Codes [CRR21] (broken by [RRT23])	OT	x	10^7	q $= 2$
Ring Syndrome Decoding [BCG+20]	OLE	o	10^5	q very large
Quasi Abelian Syndrome Decoding	OLE	o	estimated 10^5	every ≥ 3

Table: State of the art on silent PCG, for the OT and OLE correlations

Programmability [BCG+19]

A PCG is said to be programmable when you can fix a part of the correlation produced by different seeds.
It is a crucial property to obtain MPC from 2PC, to obtain malicious security from semi-honest security.

Alice "programs"

- an instance of OLE with Bob
  $\alpha = (u, [\![\Delta_B \cdot u]\!])\ \ \beta = (\Delta_B, [\![\Delta_B \cdot u]\!])$
- and another with Charlie :
  $\alpha = (u, [\![\Delta_C \cdot u]\!])\ \ \beta = (\Delta_C, [\![\Delta_C \cdot u]\!])$

Note that it is the same u.

State of the art on silent PCG

Underlying assumption	Correlation	Programmability	Correlations per second	Field size?
Syndrome Decoding for Expand and Accumulate Code [BCG+22] Expand and Convolute Codes[RRT23]	OT	x	10^7	q $= 2$
Syndrome Decoding for Silver Codes [CRR21] (broken by [RRT23])	OT	x	10^7	q $= 2$
Ring Syndrome Decoding [BCG+20]	**OLE**	**o**	$\mathbf{10^5}$	**q very large**
Quasi Abelian Syndrome Decoding	OLE	o	estimated 10^5	every ≥ 3

Table: State of the art on silent PCG, for the OT and OLE correlations

Programmability [BCG+19]

A PCG is said to be programmable when you can fix a part of the correlation produced by different seeds.
It is a crucial property to obtain MPC from 2PC, to obtain malicious security from semi-honest security.

Alice "programs"

- an instance of OLE with Bob
  $\alpha = (u, [\![\Delta_B \cdot u]\!])\ \ \beta = (\Delta_B, [\![\Delta_B \cdot u]\!])$
- and another with Charlie :
  $\alpha = (u, [\![\Delta_C \cdot u]\!])\ \ \beta = (\Delta_C, [\![\Delta_C \cdot u]\!])$

Note that it is the same u.

A first solution [BCG+20]

Solution for producing n instances of OLE [BCG+20]

- Choose a polynomial P that splits into $n = \deg(P)$ linear factors
- Build a PCG for a single OLE over $\mathcal{R} = \mathbb{F}_q[X]/(P(X))$
- Use the Chinese Remainder Theorem to convert this unique OLE, into n OLE correlation over $\mathbb{F}_q$.
- Security relies on the ring Ring Syndrome Decoding assumption.

Some limitations of the construction:

- If we want to produce n correlations, we should have $|\mathbb{F}_q| > n$. Hence the construction works only over large fields.
- Conditions on P? The choice of P matters for security: how to choose it?

Our Contribution

Introduction of Quasi-Abelian Syndrome Decoding.

- Broad family of possible instantiations
- Rich structure that allows stronger security foundations

We identify some group algebras $\mathcal{R}$ such that:

- They support fast operations.
- They are isomorphic to a product of n copies of $\mathbb{F}_q$ for $q > 2$.
- They have a canonical notion of sparsity.

Group Algebras and Quasi-Abelian Codes

We define a Group Algebra, for a finite abelian group G of formal sums $\mathbb{F}_q[G] := \left\{ \sum_{g \in G} a_g g \mid a_g \in \mathbb{F}_q \right\}$.

Some examples:

- Let $G = \{1\}$ be the trivial group with one element. Then the group algebra $\mathbb{F}_q[G]$ is isomorphic to the finite field $\mathbb{F}_q$
- Let $G = \mathbb{Z}/n\mathbb{Z}$ be the cyclic group with n elements. When q is coprime to n, $\mathbb{F}_q[G] \simeq \mathbb{F}_q[X]/(X^n - 1)$. This can be generalize :

$$\mathbb{F}_q[\mathbb{Z}/d_1\mathbb{Z} \times \cdots \times \mathbb{Z}/d_r\mathbb{Z}] \simeq \mathbb{F}_q[X_1, \cdots, X_r]/(X_1^{d_1} - 1, \cdots, X_r^{d_r} - 1).$$

Group Algebras and Quasi-Abelian Codes

Given a matrix

$$\mathbf{\Gamma} = \begin{pmatrix} \gamma_{1,1} & \cdots & \gamma_{1,\ell} \\ \vdots & \ddots & \vdots \\ \gamma_{k,1} & \cdots & \gamma_{k,\ell} \end{pmatrix} \in (\mathbb{F}_q[G])^{k\times\ell},$$

a *Quasi-Abelian*-G group code defined by $\mathbf{\Gamma}$ is

$$C = \{\mathbf{m}\mathbf{\Gamma} \mid \mathbf{m} = (m_1, \ldots, m_k) \in (\mathbb{F}_q[G])^k\},$$

Quasi-Abelian Codes examples

Some examples

- if $G = \{1\}$ then any linear code is a quasi-G code.
- if $G = \mathbb{Z}/n\mathbb{Z}$, and q is coprime to n. If we assume that $k = 1$ and $l = 2$ then a quasi-$\mathbb{Z}/n\mathbb{Z}$ code of index 2 is defined over $\mathbb{F}_q$ by a double-circulant generator matrix:

$$\left(\begin{array}{cccc|cccc} a_0 & a_1 & \dots & a_{n-1} & b_0 & b_1 & \dots & b_{n-1} \\ a_{n-1} & a_0 & \dots & a_{n-2} & b_{n-1} & b_0 & \dots & b_{n-2} \\ \vdots & & & \vdots & \vdots & & & \vdots \\ a_1 & a_{n-1} & \dots & a_0 & b_1 & b_{n-1} & \dots & b_0 \end{array}\right).$$

This exactly a standard quasi-cyclic code with block length n.

The QA-SD assumption

Definition ((Decisional) QA-SD problem)

Given a target weight t, the goal of this decisional QA-SD problem is to distinguish, with a non-negligible advantage, between the distributions

$$\begin{array}{lll} \mathcal{D}_0 : & (\mathbf{a}, \mathbf{s}) & \text{where } \mathbf{a}, \mathbf{s} \leftarrow_r \mathbb{F}_q[G] \\ \mathcal{D}_1 : & (\mathbf{a}, \mathbf{a}\mathbf{e}_1 + \mathbf{e}_2) & \text{where } \mathbf{a} \leftarrow_r \mathbb{F}_q[G] \text{ and } \mathbf{e}_i \leftarrow_r \Delta_t(\mathbb{F}_q[G]). \end{array}$$

where $\Delta_t(\mathbb{F}_q[G])$ denotes a distribution over $\mathbb{F}_q[G]$ such that $\mathbb{E}[wt(e)] = t$ when $e \leftarrow_r \Delta_t$.

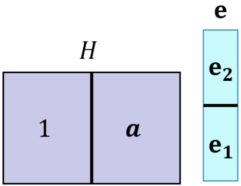
The QA-SD assumption

Definition ((Decisional) QA-SD problem)

Given a target weight t, the goal of this decisional QA-SD problem is to distinguish, with a non-negligible advantage, between the distributions

$$\begin{array}{lcl} \mathcal{D}_0 : & (\mathbf{a}, \mathbf{s}) & \text{where } \mathbf{a}, \mathbf{s} \leftarrow_r \mathbb{F}_q[G] \\ \mathcal{D}_1 : & (\mathbf{a}, \mathbf{a}\mathbf{e}_1 + \mathbf{e}_2) & \text{where } \mathbf{a} \leftarrow_r \mathbb{F}_q[G] \text{ and } \mathbf{e}_i \leftarrow_r \Delta_t(\mathbb{F}_q[G]). \end{array}$$

where $\Delta_t(\mathbb{F}_q[G])$ denotes a distribution over $\mathbb{F}_q[G]$ such that $\mathbb{E}[wt(e)] = t$ when $e \leftarrow_r \Delta_t$.



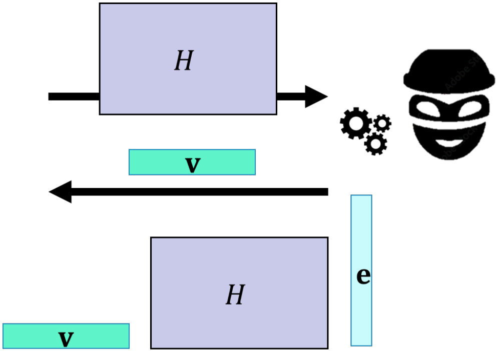
Linear attacks paradigm [BCG+20]

Bias of a distribution

Given a distribution $\mathcal{D}$ over $\mathbb{F}_2^n$, a vector $\mathbf{v} \in \mathbb{F}_2^n$:

$$\mathrm{bias}_{\mathbf{v}}(\mathcal{D}) = \left| \frac{1}{2} - \Pr_{\mathbf{u} \xleftarrow{\$} \mathcal{D}} [\mathbf{v}^\top \cdot \mathbf{u} = 1] \right|$$

The bias of $\mathcal{D}$, denoted $\mathrm{bias}(\mathcal{D})$, is the maximum bias of $\mathcal{D}$ with respect to any nonzero vector $\mathbf{v}$.



- Send H to the adversary
- The adversary returns a **test vector** $\mathbf{v}$ computed from H with unbounded time.
- Is $\mathbf{v}^\top \cdot \mathbf{u} = \mathbf{v}^\top \cdot H \cdot e$ biased ?

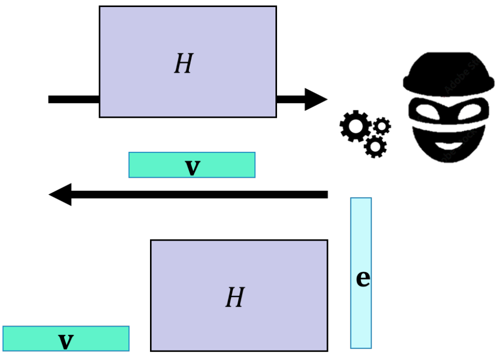
Linear attacks paradigm [BCG+20]

Bias of a distribution

Given a distribution $\mathcal{D}$ over $\mathbb{F}_2^n$, a vector $\mathbf{v} \in \mathbb{F}_2^n$:

$$\mathrm{bias}_{\mathbf{v}}(\mathcal{D}) = \left| \frac{1}{2} - \Pr_{\mathbf{u} \xleftarrow{\$} \mathcal{D}} \left[\mathbf{v}^\top \cdot \mathbf{u} = 1 \right] \right|$$

The bias of $\mathcal{D}$, denoted bias$(\mathcal{D})$, is the maximum bias of $\mathcal{D}$ with respect to any nonzero vector $\mathbf{v}$.



- Send H to the adversary
- The adversary returns a **test vector** $\mathbf{v}$ computed from H with unbounded time.
- Is $\mathbf{v}^\top \cdot \mathbf{u} = \mathbf{v}^\top \cdot H \cdot e$ biased ?

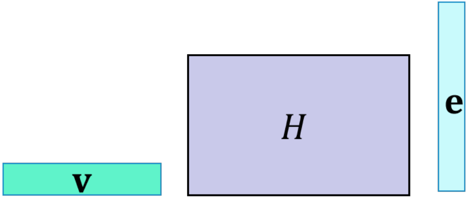
Resistance against linear attacks

Attacks	Linear?
Gaussian elimination	✓
Statistical decoding	✓
Information set decoding	✓
BKW	✓
Algebraic attack	✗

Table: Linearity of classical attacks

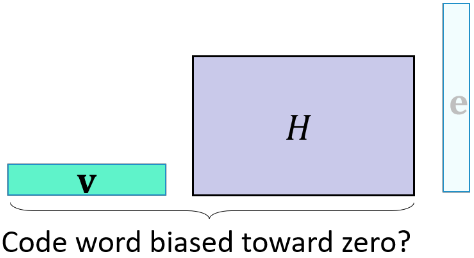
Security analysis of the QA-SD assumption

Analysis of the bias.



Security analysis of the QA-SD assumption

Analysis of the bias.



- Resistance against linear attacks can be shown by analyzing the minimum distance of the code generated by the rows of H.

Security analysis of the QA-SD assumption

$$H = \left(\begin{array}{ccc|c|ccc} a_{0,0} & \cdots & a_{0,n-1} & & b_{\ell-1,0} & \cdots & b_{\ell-1,n-1} \\ a_{0,n-1} & \cdots & a_{0,n-2} & & b_{\ell-1,n-1} & \cdots & b_{\ell-1,n-2} \\ \vdots & & \vdots & \cdots & \vdots & & \vdots \\ a_{0,1} & \cdots & a_{0,0} & & b_{\ell-1,1} & \cdots & b_{\ell-1,0} \end{array} \right).$$

Theorem (Fan and Lin,2015)

Let G be a finite abelian group, and let $(C_\ell)_\ell$ be a sequence of random quasi-G codes of length $\ell \in \mathbb{N}$ and rate $r \in (0,1)$. Let $\delta \in (0, 1 - \frac{1}{q})$. Then,

$$\lim_{\ell \to \infty} \Pr\left(\frac{d_{min}(C_\ell)}{|G|} > \delta\ell \right) = \begin{cases} 1 & \textit{if } r < 1 - h_q(\delta); \\ 0 & \textit{if } r > 1 - h_q(\delta); \end{cases}$$

and the convergence is exponentially fast.

Security Analysis of the QA-SD assumption

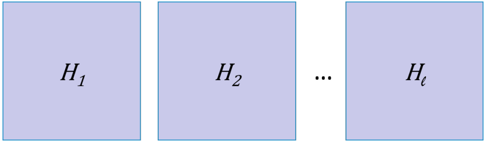


Figure: Case of Fan and Lin

Security Analysis of the QA-SD assumption

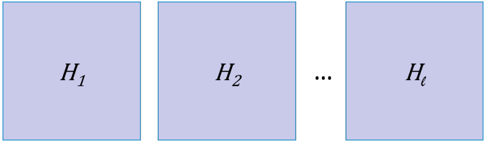


Figure: Case of Fan and Lin

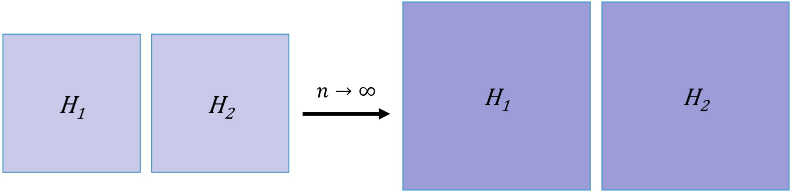


Figure: What we would like

- Open problem: Can we prove the same result whem we fix the number of blocks but their size grows?

Concrete Instance

Group algebra using $G = \prod_{i=1}^{n} \mathbb{Z}/(q-1)\mathbb{Z}$, $q \geq 3$.

$$\mathbb{F}_q[G] \simeq \mathbb{F}_q[X_1, \cdots, X_n]/(X_1^{q-1} - 1, \cdots, X_n^{q-1} - 1) \simeq \prod_{i=1}^{T} \mathbb{F}_q.$$

- Let $\mathbf{e_0^0}, \mathbf{e_0^1}, \mathbf{e_1^0}, \mathbf{e_1^1}$ be sparse elements of $\mathbb{F}_q[G]$ and $\mathbf{a} \in \mathbb{F}_q[G]$. Alice and Bob compute locally $\mathbf{u}$ and $\boldsymbol{\Delta}$:

$$\mathbf{u} = \mathbf{a} \cdot \mathbf{e_0^0} + \mathbf{e_0^1}, \quad ; \quad \boldsymbol{\Delta} = \mathbf{a} \cdot \mathbf{e_1^0} + \mathbf{e_1^1}$$

Because of the QA-SD assumption $\mathbf{u}, \boldsymbol{\Delta}$ appears to be random.

Concrete Instance

Group algebra using $G = \prod_{i=1}^{n} \mathbb{Z}/(q-1)\mathbb{Z}$, $q \geq 3$.

$$\mathbb{F}_q[G] \simeq \mathbb{F}_q[X_1, \cdots, X_n]/(X_1^{q-1} - 1, \cdots, X_n^{q-1} - 1) \simeq \prod_{i=1}^{T} \mathbb{F}_q.$$

- Let $\mathbf{e_0^0}, \mathbf{e_0^1}, \mathbf{e_1^0}, \mathbf{e_1^1}$ be sparse elements of $\mathbb{F}_q[G]$ and $\mathbf{a} \in \mathbb{F}_q[G]$. Alice and Bob compute locally $\mathbf{u}$ and $\mathbf{\Delta}$:

$$\mathbf{u} = \mathbf{a} \cdot \mathbf{e_0^0} + \mathbf{e_0^1}, \quad ; \quad \mathbf{\Delta} = \mathbf{a} \cdot \mathbf{e_1^0} + \mathbf{e_1^1}$$

  Because of the QA-SD assumption $\mathbf{u}, \mathbf{\Delta}$ appears to be random.

- Then
  $\mathbf{u} \cdot \mathbf{\Delta} = \mathbf{a}^2 \cdot \mathbf{e_0^0} \cdot \mathbf{e_1^0} + \mathbf{a} \cdot (\mathbf{e_0^0} \cdot \mathbf{e_1^1} + \mathbf{e_0^1} \cdot \mathbf{e_1^0}) + \mathbf{e_0^1} \cdot \mathbf{e_1^1}$.

Concrete Instance

Group algebra using $G = \prod_{i=1}^{n} \mathbb{Z}/(q-1)\mathbb{Z}$, $q \geq 3$.

$$\mathbb{F}_q[G] \simeq \mathbb{F}_q[X_1, \cdots, X_n]/(X_1^{q-1} - 1, \cdots, X_n^{q-1} - 1) \simeq \prod_{i=1}^{T} \mathbb{F}_q.$$

- Let $\mathbf{e_0^0}, \mathbf{e_0^1}, \mathbf{e_1^0}, \mathbf{e_1^1}$ be sparse elements of $\mathbb{F}_q[G]$ and $\mathbf{a} \in \mathbb{F}_q[G]$. Alice and Bob compute locally $\mathbf{u}$ and $\boldsymbol{\Delta}$:

$$\mathbf{u} = \mathbf{a} \cdot \mathbf{e_0^0} + \mathbf{e_0^1}, \quad ; \quad \boldsymbol{\Delta} = \mathbf{a} \cdot \mathbf{e_1^0} + \mathbf{e_1^1}$$

  Because of the QA-SD assumption $\mathbf{u}, \boldsymbol{\Delta}$ appears to be random.

- Then $\mathbf{u} \cdot \boldsymbol{\Delta} = \mathbf{a}^2 \cdot \mathbf{e_0^0} \cdot \mathbf{e_1^0} + \mathbf{a} \cdot (\mathbf{e_0^0} \cdot \mathbf{e_1^1} + \mathbf{e_0^1} \cdot \mathbf{e_1^0}) + \mathbf{e_0^1} \cdot \mathbf{e_1^1}$.

Concrete Instance

Group algebra using $G = \prod_{i=1}^{n} \mathbb{Z}/(q-1)\mathbb{Z}$, $q \geq 3$.

$$\mathbb{F}_q[G] \simeq \mathbb{F}_q[X_1, \cdots, X_n]/(X_1^{q-1} - 1, \cdots, X_n^{q-1} - 1) \simeq \prod_{i=1}^{T} \mathbb{F}_q.$$

- Let $\mathbf{e_0^0}, \mathbf{e_0^1}, \mathbf{e_1^0}, \mathbf{e_1^1}$ be sparse elements of $\mathbb{F}_q[G]$ and $\mathbf{a} \in \mathbb{F}_q[G]$. Alice and Bob compute locally $\mathbf{u}$ and $\mathbf{\Delta}$:

$$\mathbf{u} = \mathbf{a} \cdot \mathbf{e_0^0} + \mathbf{e_0^1}, \quad ; \quad \mathbf{\Delta} = \mathbf{a} \cdot \mathbf{e_1^0} + \mathbf{e_1^1}$$

  Because of the QA-SD assumption $\mathbf{u}, \mathbf{\Delta}$ appears to be random.

- Then
  $\mathbf{u} \cdot \mathbf{\Delta} = \mathbf{a}^2 \cdot \mathbf{e_0^0} \cdot \mathbf{e_1^0} + \mathbf{a} \cdot (\mathbf{e_0^0} \cdot \mathbf{e_1^1} + \mathbf{e_0^1} \cdot \mathbf{e_1^0}) + \mathbf{e_0^1} \cdot \mathbf{e_1^1}$.
- The product of two sparse elements remains sparse $\rightarrow$ Can be succinctly distributed using FSS.

Function Secret Sharing (FSS)[BGI15]

For functions that are mainly zero, one can succinctly share the function f into

$$f = f_1 + f_2$$

Enables to split sparse multiplication of the form $e_0 \cdot e_1$.

Final results

General remarks

- Operations over the group algebra can be accelerated using generalized FFT.
- Our construction works for any $q \geq 3$. When $q = 2$, $\mathbb{F}_2^n = \mathbb{F}_2 \times \cdots \times \mathbb{F}_2$ has only one invertible element, and is therefore a group algebra only in the case $n = 1$.
- *Main applications in MPC*
    - We achieve the first efficient N-party silent secure computation protocols for computing general arithmetic circuit over $\mathbb{F}_q$ for any $q > 2$.
    - Secure N-party computation of a batch of T arithmetic circuits over $\mathbb{F}_q$, $q > 2$.
    - It extends also to authenticated correlated randomness.

Final results

General remarks

- Operations over the group algebra can be accelerated using generalized FFT.
- Our construction works for any $q \geq 3$. When $q = 2$, $\mathbb{F}_2^n = \mathbb{F}_2 \times \cdots \times \mathbb{F}_2$ has only one invertible element, and is therefore a group algebra only in the case $n = 1$.
- *Main applications in MPC*
  - We achieve the first efficient N-party silent secure computation protocols for computing general arithmetic circuit over $\mathbb{F}_q$ for any $q > 2$.
  - Secure N-party computation of a batch of T arithmetic circuits over $\mathbb{F}_q$, $q > 2$.
  - It extends also to authenticated correlated randomness.

Open problems and perspectives

- Optimize the generalized FFT.
- Extend Fan and Lin to a fixed number of blocks.
- Find a solution for $q = 2$.

Thank you!