Cryptography from Lattices

This talk

homomorphic signatures

NIZK

<table>
<thead>
<tr>
<th>Year</th>
<th>Concept</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>1996</td>
<td>OWF</td>
<td>[Ajt96]</td>
</tr>
<tr>
<td>2004</td>
<td>PKE</td>
<td>[Reg04]</td>
</tr>
<tr>
<td>2008</td>
<td>IBE signatures</td>
<td>[PVW08]</td>
</tr>
<tr>
<td></td>
<td>OT</td>
<td></td>
</tr>
<tr>
<td>2009</td>
<td>low-depth PRFs</td>
<td>[Gen09]</td>
</tr>
<tr>
<td>2012</td>
<td>FHE</td>
<td></td>
</tr>
<tr>
<td>2013+</td>
<td>ABE</td>
<td>[BPR12], [GVW13], [Gen09a], [BLMR13], [BGG14], [GVW15a], [GKW18], [KW18], [PS19], [BV15]</td>
</tr>
</tbody>
</table>
Computing on Encrypted Data

confidentiality for computations

\[ ct_f \leftarrow \text{Eval}(f, ct) \]

\[
\begin{align*}
\text{pk} & \quad \text{ct} \leftarrow \text{Encrypt(pk, } x) \\
\text{ct} & \quad \text{ct}_f \\
\text{sk} & \quad \text{Decrypt(sk, } ct_f) \\
& \quad f(x)
\end{align*}
\]

fully homomorphic encryption
Computing on Encrypted Data

Confidentiality for computations

\[ ct_f \leftarrow \text{Eval}(f, ct) \]

Security: \( ct \) hides \( x \)
Compactness: \( |ct_f| \) depends on \( |f(x)| \), not \( |x| \) or \( |f| \)
Computing on Signed Data

integrity for computations

\[
y \leftarrow f(x) \\
\sigma_f \leftarrow \text{Eval}(f, \sigma)
\]

\[
\sigma \leftarrow \text{Sign}(sk, x)
\]

\[
(f, y, \sigma_f) \rightarrow \text{Verify}(vk, f, y, \sigma_f)
\]

fully homomorphic signatures
Computing on Signed Data

integrity for computations

$\sigma \leftarrow \text{Sign}(sk, x)$

$f, y, \sigma_f \leftarrow \text{Eval}(f, \sigma)$

Security: if $y = f(x)$, cannot convince verifier of $y' \neq f(x)$

Compactness: $|\sigma_f|$ depends on $|f(x)|$, not $|x|$ or $|f|$
The GSW FHE Scheme

Recall the GSW encryption scheme:

\[ \text{pk: } A \in \mathbb{Z}_q^{n \times m} \]
\[ \text{sk: } s \in \mathbb{Z}_q^n \]

Ciphertext for \( x \in \{0,1\} \):
\[ C = AR + xG \]
where \( R \) is random short matrix

Public key is an LWE matrix (columns are LWE samples)
\[ s^T A = e^T \approx 0^T \]
recall the GSW encryption scheme:

\[
pk: A \in \mathbb{Z}_{q}^{n \times m} \\
sk: s \in \mathbb{Z}_{q}^{n}
\]

ciphertext for \( x \in \{0,1\} \):

\[
C = AR + xG \quad \text{where } R \text{ is random short matrix}
\]

\[
G \text{ is the “gadget” matrix:}
\]

\[
G = (1,2,4, \ldots, 2^\ell) \otimes I_n \in \mathbb{Z}_{q}^{n \times n\ell}
\]

\[
G^{-1}: \mathbb{Z}_{q}^{n \times k} \rightarrow \{0,1\}^{n\ell \times k} \text{ is “binary decomposition”}
\]

\[
GG^{-1}(A) = A
\]
recall the GSW encryption scheme:

\[
\begin{align*}
\text{pk: } & \quad A \in \mathbb{Z}_q^{n \times m} \\
\text{sk: } & \quad s \in \mathbb{Z}_q^n \\
\end{align*}
\]

public key is an \textbf{LWE matrix} (columns are LWE samples)

\[ s^T A = e^T \approx 0^T \]

ciphertext for \( x \in \{0,1\} \):

\[ C = A R + x G \quad \text{where } R \text{ is random short matrix} \]

decryption:

\[ s^T C = s^T A R + x \cdot s^T G \approx x \cdot s^T G \]
Homomorphic Operations in GSW

$$C_1 = AR_1 + x_1 G \quad C_2 = AR_2 + x_2 G$$

$$C_+ = C_1 + C_2 = A(R_1 + R_2) + (x_1 + x_2)G$$
Homomorphic Operations in GSW

\[ C_1 = AR_1 + x_1 G \quad \quad C_2 = AR_2 + x_2 G \]

\[ C_+ = C_1 + C_2 = A(R_1 + R_2) + (x_1 + x_2)G \]

\[ = AR_+ + (x_1 + x_2)G \]

\[ C_x = C_1 G^{-1}(C_2) = AR_1 G^{-1}(C_2) + x_1 C_2 \]

\[ = A(R_1 G^{-1}(C_2) + x_1 R_2) + x_1 x_2 G \]

\[ R_x \]
Homomorphic Operations in GSW

\[ C_1 = AR_1 + x_1 G \quad C_2 = AR_2 + x_2 G \]

\[ C_+ = C_1 + C_2 = A(R_1 + R_2) + (x_1 + x_2)G \]
\[ = AR_+ + (x_1 + x_2)G \]

\[ C_\times = C_1 G^{-1}(C_2) = AR_1 G^{-1}(C_2) + x_1 C_2 \]
\[ = A(R_1 G^{-1}(C_2) + x_1 R_2) + x_1 x_2 G \]
\[ = AR_\times + x_1 x_2 G \]

Correctness: \( R_1, R_2, x_1 \) short \( \Rightarrow R_+, R_\times \) also short
Homomorphic Operations in GSW

\[ C_1 = AR_1 + x_1 G \]
\[ C_2 = AR_2 + x_2 G \]
\[ \vdots \]
\[ C_n = AR_n + x_n G \]

\[ C_f = AR_{f,x} + f(x)G \]

“input-independent” evaluation

\( C_f \) is a function of \( C_1, \ldots, C_n, f \)
(and independent of \( x \))
Homomorphic Operations in GSW

\[ C_1 = AR_1 + x_1 G \quad \quad C_2 = AR_2 + x_2 G \]

\[ C_+ = C_1 + C_2 = A(R_1 + R_2) + (x_1 + x_2)G \]
\[ = AR_+ + (x_1 + x_2)G \]

\[ C_x = C_1 G^{-1}(C_2) = A(R_1 G^{-1}(C_2) + x_1 R_2) + x_1 x_2 G \]
\[ = AR_x + x_1 x_2 G \]
Homomorphic Operations in GSW

\[ C_1 = AR_1 + x_1 G \quad \quad \quad C_2 = AR_2 + x_2 G \]

\[ C_+ = C_1 + C_2 = A(R_1 + R_2) + (x_1 + x_2)G \]
\[ = AR_+ + (x_1 + x_2)G \]

\[ C_\times = C_1 G^{-1}(C_2) = A(R_1 G^{-1}(C_2) + x_1 R_2) + x_1 x_2 G \]
\[ = AR_\times + x_1 x_2 G \]

**observation:** \( R_+ \) and \( R_\times \) is a short linear combination of \( R_1 \) and \( R_2 \)
The BGG$^+$ Homomorphisms

\[ C_1 = AR_1 + x_1 G \quad \cdots \quad C_n = AR_n + x_n G \]

\[ C_f = AR_{f,x} + f(x) G \quad \text{where} \quad R_{f,x} = [R_1 | \cdots | R_n] H_{f,x} \]

and \( H_{f,x} \) is short

equivalently:

\[ [AR_1 | \cdots | AR_n] H_{f,x} = AR_{f,x} \]

\[ [C_1 - x_1 G | \cdots | C_n - x_n G] H_{f,x} = C_f - f(x) G \]

[BGGHNSV14]
The BGG\(^+\) Homomorphisms

“input-independent” evaluation (given \(C_1, \ldots, C_n, f\)):
\[ C_1, \ldots, C_n \mapsto C_f \]

“input-dependent” evaluation (given \(C_1, \ldots, C_n, f, x\)):
\[ [C_1 - x_1 G | \ldots | C_n - x_n G] H_{f,x} = C_f - f(x) G \]

applications:

<table>
<thead>
<tr>
<th></th>
<th>input-independent evaluation ((A_f))</th>
<th>input-dependent evaluation ((H_{f,x}))</th>
</tr>
</thead>
<tbody>
<tr>
<td>attribute-based encryption [BGGHNSVV14]</td>
<td>key-generation</td>
<td>decryption</td>
</tr>
<tr>
<td>homomorphomic signatures [GVW15]</td>
<td>verification</td>
<td>signing</td>
</tr>
<tr>
<td>constrained PRFs [BV15]</td>
<td>normal evaluation</td>
<td>constrained evaluation</td>
</tr>
</tbody>
</table>
public parameters $A \in \mathbb{Z}_q^{n \times m}$ (LWE matrix)

$$C = AR + xG$$

commitment

opening (check $R$ short)

message

encryption of $x$ with randomness $R$

commitment to $x$ with opening $R$
public parameters $A \in \mathbb{Z}_q^{n \times m}$ (LWE matrix)

$$C = AR + xG$$

statistically binding: correctness of GSW (in fact, extractable)

computationally hiding: security of GSW (under LWE)
GSW as a Homomorphic Commitment

computing on committed values:

\[ C_1 = AR_1 + x_1 G \]
\[ C_2 = AR_2 + x_2 G \]
\[ \vdots \]
\[ C_n = AR_n + x_n G \]

goal: open the committed value to \( y = f(x) \)

syntax: Open(pp, c, (f, y), r)

pp: public parameters
\( c \): commitment
\( (f, y) \): value
\( r \): opening

binding:

adversary cannot open \( c \) to \( (f, y) \neq (f, y') \)

Openings are with respect to a value \( y \) and a function \( f \)
GSW as a Homomorphic Commitment

computing on committed values:

\[ C_1 = AR_1 + x_1 G \]
\[ C_2 = AR_2 + x_2 G \]
\[ \vdots \]
\[ C_n = AR_n + x_n G \]

\textbf{goal:} open the committed value to \( y = f(x) \)

\textbf{syntax:} Open(pp, c, (f, y), r)

pp: public parameters \hspace{1cm} (f, y): value
\hspace{1cm} c: commitment \hspace{1cm} r: opening

\textbf{binding:}

adversary cannot open \( c \) to \( (f, y) \neq (f, y') \)

\textbf{Application:}
preprocessing NIZKs
GSW as a Homomorphic Commitment

computing on committed values:

\[ C_1 = AR_1 + x_1 G \]
\[ C_2 = AR_2 + x_2 G \]
\[ \vdots \]
\[ C_n = AR_n + x_n G \]

commitment:

\[ C_f = AR_{f,x} + f(x)G \]

\( C_f \) is a commitment to \( f(x) \) with opening \( R_{f,x} \)
GSW as a Homomorphic Commitment

[GVW14]

computing on committed values:

\[ C_1 = AR_1 + x_1 G \]
\[ C_2 = AR_2 + x_2 G \]
\[ \vdots \]
\[ C_n = AR_n + x_n G \]

commitment:

\[ C_f = AR_{f,x} + f(x)G \]

opening:

\[ R_{f,x} = [R_1 | \cdots | R_n]H_{f,x} \]

check opening by computing \( C_f \) from \( C_1, \ldots, C_n \) (does not need to know \( x \)) and verifying that \( R_{f,x} \) is small and \( C_f = AR_{f,x} + f(x)G \)
**GSW as a Homomorphic Commitment**

[GVW14]

computing on committed values:

\[ C_1 = AR_1 + x_1 G \]
\[ C_2 = AR_2 + x_2 G \]
\[ \vdots \]
\[ C_n = AR_n + x_n G \]

“input-independent” evaluation (given \( C_1, \ldots, C_n, f \)):

\[ C_1, \ldots, C_n \mapsto C_f \]

“input-dependent” evaluation (given \( C_1, \ldots, C_n, f, x \)):

\[ [C_1 - x_1 G | \cdots | C_n - x_n G]H_{f,x} = C_f - f(x)G \]

commitment:

\[ C_f = AR_{f,x} + f(x)G \]

opening:

\[ R_{f,x} = [R_1 | \cdots | R_n]H_{f,x} \]

verification

evaluation
From Commitments to Proofs

homomorphic commitments can be used to prove relations on secret values

Goal: prove that a (secret) statement $x$ satisfies some relation $\mathcal{R}$
From Commitments to NIZKs (Dream Version)

\( \mathcal{R}(x, w) \): NP relation

\[ C_w \leftarrow \text{Commit}(pp, w) \]

opening for \( C_{\mathcal{R}_x, w} \)

\( R_x(w) := R(x, w) \)

function that depends only on the statement \( x \)

verifier checks \( C_{\mathcal{R}_x, w} \) opens to 1
From Commitments to NIZKs (Dream Version)

$R(x, w)$: NP relation

$C_w \leftarrow \text{Commit}(pp, w)$
opening for $C_{R_x,w}$

**Zero-Knowledge** (“proof hides $w$”):
- $C_w$ hides $w$ (commitment is hiding)
- $C_{R_x,w}$ is a public function of $C_w$
- opening to $C_{R_x,w}$ might leak information about $w$ (can be fixed)
\( \mathcal{R}(x, w) \): NP relation

\[ C_w \leftarrow \text{Commit}(pp, w) \]

opening for \( C_{\mathcal{R}_x,w} \)

**Soundness** (for \( x \) where \( \mathcal{R}_x(w) = 0 \) for all \( w \)):

- if \( C_{w^*} \) is an honestly-generated commitment to some value \( w^* \), then \( C_{\mathcal{R}_x,w^*} \) is a commitment to \( \mathcal{R}_x(w^*) = 0 \) by correctness
- statistical soundness follows by statistical binding
From Commitments to NIZKs (Dream Version)

Open Problem: NIZK proof of well-formedness of GSW ciphertext $C \in \mathbb{Z}_q^{n \times m}$

$\exists x \in \{0,1\}$, short $R \in \mathbb{Z}_q^{m \times m}$ : $C = AR + xG$

Would yield direct construction of NIZK for NP (lattice "analog" of [GOS06])

- Construction makes black-box use of cryptography (in contrast to Fiat-Shamir approach [CCHLRRW19, PS19])

Soundness (for $x$ where $\mathcal{R}_x(w) = 0$ for all $w$):

- if $C_{w^*}$ is an honestly-generated commitment to some value $w^*$, then $C_{\mathcal{R}_x,w^*}$ is a commitment to $\mathcal{R}_x(w^*) = 0$ by correctness
- statistical soundness follows by statistical binding
Can we still use this approach to obtain some type of NIZK?

Yes! But in a weaker “preprocessing” or “correlated randomness” model
NIZKs in the Preprocessing Model

(trusted) setup algorithm generates both proving key $k_P$ and a verification key $k_V$ (statement-independent)

$\pi = \text{Prove}(k_P, x, w)$

prover algorithm takes proving key $k_P$, NP statement $x$, and NP witness $w$

Verify($k_V, x, \pi$)
NIZKs in the Preprocessing Model

simpler than CRS model:
• soundness holds assuming $k_V$ is hidden
• zero-knowledge holds assuming $k_P$ is hidden

main requirement: reusability suffices for many applications of NIZKs

CRS model: $k_P$ and $k_V$ are both public
From Commitments to Preprocessing NIZKs

\[ k_P = (C_w, R_w) \]

openings

\[ k_V = C_w \]

\[ C_w \leftarrow \text{Commit}(pp, w) \]

opening for \( C_{Rx,w} \)

**challenge:** proving that \( C_w \) is a valid commitment

**solution:** have a trusted party generate it!
From Commitments to Preprocessing NIZKs

\[ k_P = (C_w, R_w) \]

problem: preprocessing is witness-dependent

solution: add a layer of indirection

\[ k_V = C_w \]
From Commitments to Preprocessing NIZKs

\[(k, C_k, R_k)\]  prover is given commitment and opening to an encryption key \(k\)

**solution:** add a layer of indirection
From Commitments to Preprocessing NIZKs

\[(k, C_k, R_k)\] verifier given commitment to \(k\)

\[C_k\]

**solution:** add a layer of indirection
From Commitments to Preprocessing NIZKs

\[(k, C_k, R_k)\]

\[f_{x,ct}(k) = R(x, \text{Decrypt}(k, ct))\]

[Checks that ct encrypts a valid witness]

opening for \(C_{f_{x,ct},k}\)

\[ct \leftarrow \text{Encrypt}(k, w)\]

[ct is an encryption of the witness w]

**solution:** add a layer of indirection
From Commitments to Preprocessing NIZKs

\[(k, C_k, R_k)\]

\[f_{x,ct}(k) = R(x, \text{Decrypt}(k, ct))\]
[Checks that ct encrypts a valid witness]

\[\text{ct} \leftarrow \text{Encrypt}(k, w)\]
[ct is an encryption of the witness w]

verifier computes \(C_{f_{x,ct},k}\) from \((x, ct, C_k)\) and checks that it opens to 1
From Commitments to Preprocessing NIZKs

\[(k, C_k, R_k)\]

\[f_{x,ct}(k) = \mathcal{R}(x, \text{Decrypt}(k, ct))\]
[Checks that ct encrypts a valid witness]

\[ct \leftarrow \text{Encrypt}(k, w)\]
[ct is an encryption of the witness w]

**Soundness:** \(C_{f_{x,ct},k}\) is a commitment on \(f_{x,ct}(k) = 0\) for all \(k\) and a false \(x\); soundness follows by statistical binding of commitment scheme
From Commitments to Preprocessing NIZKs

\[ (k, C_k, R_k) \]

\[ f_{x,ct}(k) = \mathcal{R}(x, \text{Decrypt}(k, ct)) \]
[Checks that \( ct \) encrypts a valid witness]

\[ ct \leftarrow \text{Encrypt}(k, w) \]
[\( ct \) is an encryption of the witness \( w \)]

Zero-Knowledge: commitment + opening hide \( k \) and encryption scheme hides \( w \)
From Commitments to Preprocessing NIZKs

\[ k_P = (k, C_k, R_k) \quad \text{Prove}(k_P, x, w) \quad \pi = \text{Prove}(k_P, x, w) \]

\[ k_V = C_k \quad \text{Verify}(k_V, x, \pi) \]

designated-prover NIZK from homomorphic commitments (under LWE)
From Commitments to Preprocessing NIZKs

$$k_P = (k, C_k, R_k)$$

$$k_V = C_k$$

- Using homomorphic commitments to construct correlation-intractable hash functions $\Rightarrow$ full NIZKs for NP from LWE [PS19]!

**designated-prover** NIZK from homomorphic commitments (under LWE)
computing on committed values:

\[ C_1 = AR_1 + x_1 G \]
\[ C_2 = AR_2 + x_2 G \]
\[ \vdots \]
\[ C_n = AR_n + x_n G \]

commitment:

\[ C_f = AR_{f,x} + f(x)G \]

opening:

\[ R_{f,x} = [R_1 | \cdots | R_n]H_{f,x} \]

Requirement (for ZK): openings hides \( x \) up to what is revealed by \( f(x) \) ("context-hiding")

not true as written since \( R_{f,x} \) leaks information about \( R_1, \ldots, R_n \)
computing on committed values:

\[ C_1 = AR_1 + x_1 G \]
\[ C_2 = AR_2 + x_2 G \]
\[ \vdots \]
\[ C_n = AR_n + x_n G \]

commitment:

\[ C_f = AR_{f,x} + f(x)G \]

opening:

\[ R_{f,x} = [R_1 | \cdots | R_n]H_{f,x} \]

Requirement (for ZK): openings hides \( x \) up to what is revealed by \( f(x) \) ("context-hiding")

**Context-Hiding:** public parameters \( A \), commitments \( C_1, \ldots, C_n \) and opening \( R_{f,x} \) can be simulated given only \((f, f(x))\)
Another Ingredient: Lattice Trapdoors

gadget trapdoors [MP12]

random matrix $A$

short matrix (trapdoor) $R$

gadget matrix $G$

[Ajt99, GPV08, AP09, CHKP10, MP12, LW15]
Another Ingredient: Lattice Trapdoors

[gadget trapdoors [MP12]]

short $R$ such that $AR = G$

enables preimage sampling for SIS:

- let $f_A(x) := Ax$
- given $u = f_A(x)$ and $R$, can sample short $x'$ where $f_A(x') = u$

and $x'$ is Gaussian-distributed
Another Ingredient: Lattice Trapdoors

\[ A = [A_1 | A_2] \]

two possible trapdoors:

- if \( R_1 \) is trapdoor for \( A_1 \), then \( A_1 R_1 = G \) and
  \[
  [A_1|A_2] \cdot \begin{bmatrix} R_1 \\ 0 \end{bmatrix} = G
  \]

- if \( A_2 = A_1 R_2 \pm G \) for short \( R_2 \), then
  \[
  [A_1|A_2] \cdot \begin{bmatrix} R_2 \\ I \end{bmatrix} = G
  \]

two statistically-indistinguishable ways to sample \( f_A^{-1}(u) \)
computing on committed values:
\[
C_1 = AR_1 + x_1G \\
C_2 = AR_2 + x_2G \\
\vdots \\
C_n = AR_n + x_nG
\]

commitment:
\[
C_f = AR_{f,x} + f(x)G
\]

opening:
\[
R_{f,x} = [R_1 | \cdots | R_n]H_{f,x}
\]

**Context-Hiding:** public parameters \( A \), commitments \( C_1, \ldots, C_n \) and opening \( R_{f,x} \) can be simulated given only \( (f, f(x)) \)
Context-Hiding for Commitments

[GVW14]

for simplicity: only support openings to \( f(x) = 1 \)

commitment:

\[
C_f = AR_{f,x} + f(x)G
\]

opening:

\[
R_{f,x} = [R_1 \mid \cdots \mid R_n]H_{f,x}
\]

Context-Hiding: public parameters \( A \), commitments \( C_1, \ldots, C_n \) and opening \( R_{f,x} \) can be simulated given only \((f, f(x))\)
for simplicity: only support openings to $f(x) = 1$

opening can be used to obtain trapdoor for

$$[A \mid C_f] = [A \mid AR_{f,x} + G]$$

if simulator chooses $A$, can choose $A$ with trapdoor

if commitments are well-formed, committer also has trapdoor

commitment:

$$C_f = AR_{f,x} + f(x)G$$

opening:

$$R_{f,x} = [R_1 \mid \cdots \mid R_n]H_{f,x}$$
Context-Hiding for Commitments

for simplicity: only support openings to \( f(x) = 1 \)

opening can be used to obtain trapdoor for
\[
[A \mid C_f] = [A \mid AR_f,x + G]
\]

idea: include random target vector \( u \) in public parameters

opening: short vector \( v \) such that
\[
[A \mid C_f]v = u
\]

commitment:
\[
C_f = AR_f,x + f(x)G
\]

opening:
\[
R_{f,x} = [R_1 \mid \cdots \mid R_n]H_{f,x}
\]

Context-Hiding: public parameters \( A \), commitments \( C_1, \ldots, C_n \) and opening \( R_{f,x} \) can be simulated given only \((f, f(x))\)
## Context-Hiding for Commitments

**real scheme:**

public parameters:
- LWE matrix $\mathbf{A}$
- sample random $\mathbf{u}$

commitments:
- $C_i \leftarrow \mathbf{A}R_i + x_i \mathbf{G}$

opening:
- compute $C_f$ from $C_1, \ldots, C_n$
- sample short $\mathbf{v}$ such that

\[
[A \mid C_f] \mathbf{v} = \mathbf{u}
\]

using $R_{f, x} \leftarrow [R_1 \mid \cdots \mid R_n]H_{f, x}$

**to simulate:**

public parameters:
- sample $\mathbf{A}$ with trapdoor $\mathbf{R}$
- sample random $\mathbf{u}$

commitments:
- sample random matrices $C_i$

opening:
- compute $C_f$ from $C_1, \ldots, C_n$
- sample short $\mathbf{v}$ such that

\[
[A \mid C_f] \mathbf{v} = \mathbf{u}
\]

using $\mathbf{R}$

**Context-Hiding:** public parameters $\mathbf{A}$, commitments $C_1, \ldots, C_n$ and opening $R_{f, x}$ can be simulated given only $(f, f(x))$
Dual-Mode Homomorphic Commitments

public parameters $A \in \mathbb{Z}_q^{n \times m}$ (LWE matrix)

$$C = AR + xG$$

commitment

opening (check $R$ short)

message

statistically binding: correctness of GSW (in fact, extractable)

computationally hiding: security of GSW (under LWE)
Dual-Mode Homomorphic Commitments

Public parameters $A \in \mathbb{Z}_q^{n \times m}$ (uniformly random)

$$C = AR + xG$$

- **Commitment**
- **Opening** (check $R$ short)
- **Message**

**Statistically hiding**: leftover hash lemma (in fact, equivocable)

**Computational binding**: switch $A$ to LWE matrix
Homomorphic Signatures

public parameters $A \in \mathbb{Z}_q^{n \times m}$ (uniformly random)

\[ C = AR + xG \]

- public parameters
- signature (check $R$ short)
- message

equivocation $\Rightarrow$ signature
Homomorphic Signatures

\[ C = AR + xG \]

- **Public parameters**\( A \in \mathbb{Z}_{q}^{n \times m} \) (uniformly random)
- **Signature**\( C_1, \ldots, C_n \)
- **Verification key**\( A \)
- **Signing key** trapdoor for \( A \)
Homomorphic Signatures

vk: $A, C_1, ..., C_n \in \mathbb{Z}_q^{n \times m}$

sk: trapdoor for $A$

signature on $x \in \{0,1\}^n$:

short $R_1, ..., R_n \in \mathbb{Z}_q^{n \times m}$

where $C_i = AR_i + x_i G$

compute $f$ on signatures:

$$R_{f,x} = [R_1 \mid \cdots \mid R_n]H_{f,x}$$

verify signature $R$ on $(f, f(x))$

$$C_1, ..., C_n, f \mapsto C_f$$

check $AR + f(x)G = C_f$

unforgeability follows from binding property of the commitment scheme

[GVW14]
GSW ciphertexts:

\[ C_i = AR_i + x_i G \]

“input-independent” evaluation (given \( C_1, \ldots, C_n, f \)):

\( C_1, \ldots, C_n \mapsto C_f \)

“input-dependent” evaluation (given \( C_1, \ldots, C_n, f, x \)):

\[ [C_1 - x_1 G | \ldots | C_n - x_n G] H_{f,x} = C_f - f(x)G \]

\( A \) is LWE matrix \( \Rightarrow \) extractable commitments

\( A \) is uniform \( \Rightarrow \) equivocable commitments (homomorphic signatures)

homomorphic commitments/signatures \( \Rightarrow \) designated-prover NIZKs
Open Questions

NIZK proof of well-formedness of GSW ciphertexts?

Fully homomorphic commitments/signatures from lattices?

\[
R_{f,x} = [R_1 | \cdots | R_n]H_{f,x}
\]

\[\|H_{f,x}\|\] scales with exponentially in the depth \(d\) of the function \(f\), so modulus \(q > 2^{O(d)}\)
Open Questions

NIZK proof of well-formedness of GSW ciphertexts?

Fully homomorphic commitments/signatures from lattices?

$$R_{f,x} = [R_1 | \cdots | R_n]H_{f,x}$$

Short public parameters without random oracles?

Thank you!