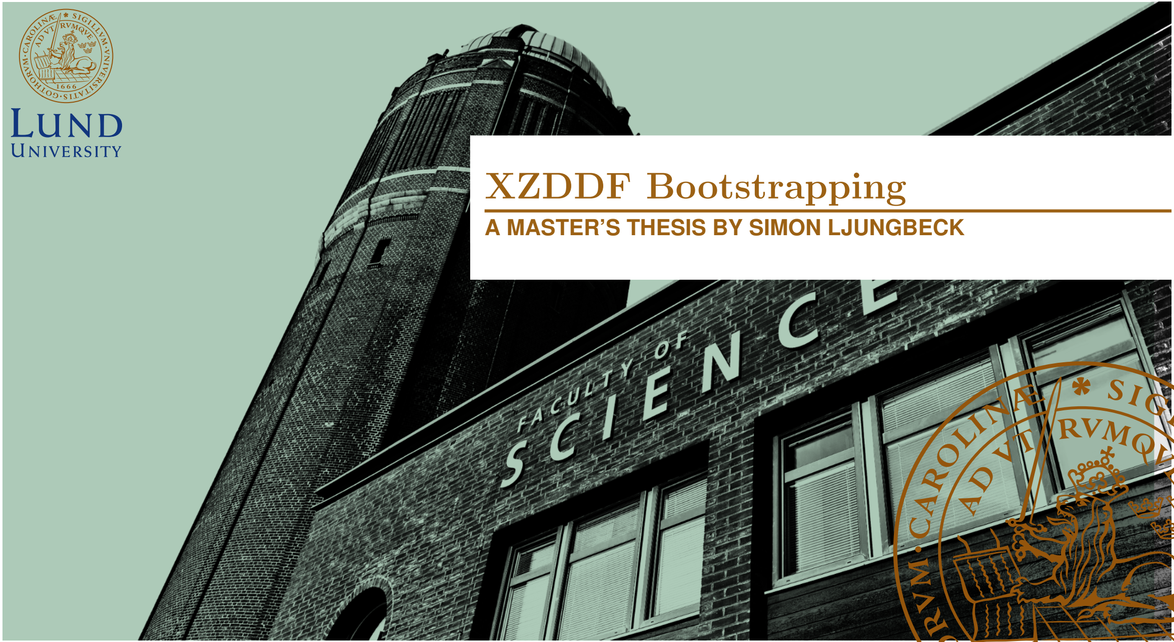
LUND UNIVERSITY



XZDDF Bootstrapping

A MASTER'S THESIS BY SIMON LJUNGBECK

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

Project Description

- Study Fully Homomorphic Encryption (FHE)
  - How does it work?
  - What are the main problems?
- Investigate XZDDF[1] bootstrapping
- Implement XZDDF[1] bootstrapping

[1] `https://eprint.iacr.org/2023/1564`

Outline

Introduction

Fully Homomorphic Encryption

Fully Homomorphic Encryption

XZDDF Bootstrapping

Modification of XZDDF

Benchmark Tests

Conclusions

- Introduction to FHE
- XZDDF bootstrapping
- Modification of XZDDF bootstrapping
- Benchmark tests of XZDDF implementation

Introduction

Fully Homomorphic Encryption

Fully Homomorphic Encryption

XZDDF Bootstrapping

Modification of XZDDF

Benchmark Tests

Conclusions

Fully Homomorphic Encryption

What is Fully Homomorphic Encryption (FHE)?

Introduction

Fully Homomorphic Encryption

Fully Homomorphic Encryption

XZDDF Bootstrapping

Modification of XZDDF

Benchmark Tests

Conclusions

- Let $c_1 = \mathrm{Enc}(m_1)$ and $c_2 = \mathrm{Enc}(m_2)$ be two ciphertexts
- Assume we want to compute $c_3 = \mathrm{Enc}(m_1 + m_2)$
  - Normally: $c_3 = \mathrm{Enc}(\mathrm{Dec}(c_1) + \mathrm{Dec}(c_2))$
  - FHE: $c_3 = c_1 + c_2$
- FHE: $\mathrm{Enc}(f(m_1, \ldots, m_t)) = f(\mathrm{Enc}(m_1), \ldots, \mathrm{Enc}(m_t))$

Why FHE?

- Keep privacy when third parties do computations on data
  - Cloud services
  - Fog computing
- Ex: training an ML model with sensitive data
- Today's problem: FHE too inefficient

Noise-based FHE

Introduction

Fully Homomorphic Encryption

Fully Homomorphic Encryption

XZDDF Bootstrapping

Modification of XZDDF

Benchmark Tests

Conclusions

- FHE ciphertexts usually contain some noise
- Learning With Errors (LWE):
  - Enc : $\mathbb{Z}_q \ni m \mapsto \mathrm{LWE}_q(m) := (\mathbf{a}, b = \langle \mathbf{a}, \mathbf{s} \rangle + m + e \mod q) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$
  - Dec : $c = (\mathbf{a}, b) \mapsto b - \langle \mathbf{a}, \mathbf{s} \rangle = m + e \approx m$

The noise grows...

Introduction

Fully Homomorphic Encryption

Fully Homomorphic Encryption

XZDDF Bootstrapping

Modification of XZDDF

Benchmark Tests

Conclusions

- Homomorphic property of LWE:

$$
\begin{aligned}
c_1 + c_2 &= (\mathbf{a}_1, b_1) + (\mathbf{a}_2, b_2) \\
&= (\mathbf{a}_1 + \mathbf{a}_2, \langle \mathbf{a}_1, \mathbf{s} \rangle + \langle \mathbf{a}_2, \mathbf{s} \rangle + m_1 + m_2 + e_1 + e_2) \\
&= (\mathbf{a}_1 + \mathbf{a}_2, \langle \mathbf{a}_1 + \mathbf{a}_2, \mathbf{s} \rangle + (m_1 + m_2) + (e_1 + e_2))
\end{aligned}
$$

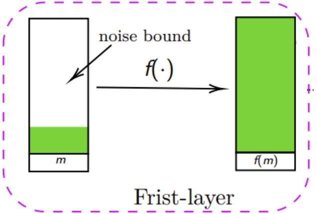


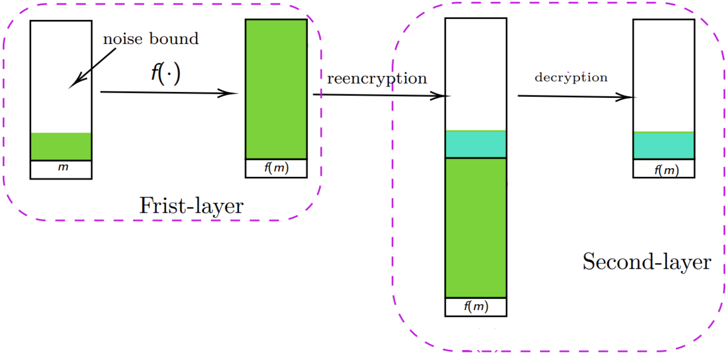
Figure: From Xiang et al.[2]

[2] https://iacr.org/cryptodb//data/paper.php?pubkey=33119

Why is FHE slow?

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

- Bootstrapping:



- LWE : $\mathrm{Dec}(c) = b - \langle \mathbf{a}, \mathbf{s} \rangle \mod q$

Introduction

Fully Homomorphic Encryption

Fully Homomorphic Encryption

XZDDF Bootstrapping

Modification of XZDDF

Benchmark Tests

Conclusions

XZDDF Bootstrapping

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

XZDDF Bootstrapping

- Assume first-layer: $(\mathbf{a}, b = \sum_{i=0}^{n-1} a_i s_i - \text{noised}(m)) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$
  - $\implies$ applicable with Regev, BGV, CKKS
  - $\implies \text{noised}(m) = \sum_{i=0}^{n-1} a_i s_i - b \mod q$
- $\mathcal{R}_Q := \mathbb{Z}_Q[X]/(X^N + 1)$ where $N = 2^k \qquad \implies \quad X^{2N} \equiv 1$
- Assume $q = 2N \implies X^{\text{noised}(m)} = X^{\sum_{i=0}^{n-1} a_i s_i - b \mod q} = X^{\sum_{i=0}^{n-1} a_i s_i - b}$
  - Let $r(X) = \sum_{i=0}^{q-1} iX^{-i} \implies \text{noised}(m) = \text{coeff}_0 \left(r(X) \cdot X^{\sum_{i=0}^{n-1} a_i s_i - b} \right)$
- If $q | 2N$ instead:
  - $\left(X^{\frac{2N}{q}} \right)^q \equiv 1$
  - $\text{noised}(m) = \text{coeff}_0 \left(r(X^{\frac{2N}{q}}) \cdot X^{-\frac{2N}{q} b} X^{\frac{2N}{q} \sum_{i=0}^{n-1} a_i s_i} \right)$

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

LUND UNIVERSITY

More XZDDF Bootstrapping

XZDDF Bootstrapping

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

- Assume $c_i(X)$ encrypts X^{s_i} under $f(X)$
- Automorphism: $c_i(X^{a_i})$ encrypts $X^{a_i s_i}$ under $f(X^{a_i})$
- Problem 1: might have $2 | a_i \implies a_i$ and $2N$ not coprime
  - Solution: $q|N$ instead of $q|2N$

$$\implies X^{\frac{2N}{q}a_i s_i} = X^{(\frac{2N}{q}a_i+1)s_i - s_i} = X^{w_i s_i} X^{-s_i}, \text{ where } w_i \text{ is odd}$$

- Problem 2: want key $f(X)$, not $f(X^{a_i})$
  - Solution: use NTRU encryption...

NTRU Encryption

Introduction

Fully Homomorphic Encryption

Fully Homomorphic Encryption

XZDDF Bootstrapping

Modification of XZDDF

Benchmark Tests

Conclusions

- Define

$$(\tau, \Delta) := \begin{cases} \left(1, \left\lfloor \frac{Q}{t} \right\rceil\right), & \text{if noised}(m) = e + \left\lfloor \frac{q}{t} \right\rceil \cdot m \\ (t, 1), & \text{if noised}(m) = t \cdot e + m \\ (1, 1), & \text{if noised}(m) = e + m. \end{cases}$$

- Scalar NTRU encryption: $\mathrm{NTRU}_{Q,f,\tau,\Delta}(u) := \tau \cdot g/f + \Delta \cdot u/f \in \mathcal{R}_Q$
- Vector NTRU encryption:
  $\mathrm{NTRU}'_{Q,f,\tau}(v) := (\tau \cdot g_0/f + B^0 \cdot v, \ldots, \tau \cdot g_{d-1}/f + B^{d-1} \cdot v) \in \mathcal{R}_Q^d$

LUND UNIVERSITY

Homomorphic Multiplication for NTRU

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

- $\mathrm{BitDecom}_B(a) := (a_0, \ldots, a_{d-1}) \in \mathcal{R}_B^d$, where $a = \sum_{i=0}^{d-1} a_i \cdot B^i$
- $c \odot \mathbf{c}' := \langle \mathrm{BitDecom}_B(c), \mathbf{c}' \rangle = \sum_{i=0}^{d-1} c_i c_i' = \tau \cdot \sum_{i=0}^{d-1} c_i g_i / f + cv \in \mathcal{R}_Q$
- **Lemma 4.1** (Homomorphic multiplication)**.** Assume that $c = \mathrm{NTRU}_{Q,f,\tau,\Delta}(u)$ and $\mathbf{c}' = \mathrm{NTRU}'_{Q,f,\tau}(v)$. Then $\hat{c} = c \odot \mathbf{c}'$ is a scalar NTRU ciphertext of uv.

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

NTRU Key Switching

- $\mathbf{ksk}_{f_1,f_2} := (\tau \cdot g_0/f_2 + B^0 \cdot f_1/f_2, \ldots, \tau \cdot g_{d-1}/f_2 + B^{d-1} \cdot f_1/f_2)$
- **Lemma 4.2** (NTRU key switching)**.** The product $c \odot \mathbf{ksk}_{f_1,f_2}$ is a scalar NTRU encryption of the same message as c but under the new private key $f_2 \in \mathcal{R}_Q$.
  $\implies$ Problem 2 solved

Generating the Blind Rotation Key

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

Algorithm 10 XZDDF.BRKGen

Require:
- $q, n \in \mathbb{N}^*$ // first-layer parameters
- $\mathbf{s} \in \mathbb{Z}_q^n$ // first-layer private key
- $Q, N, \tau, \Delta \in \mathbb{N}^*$ // second-layer parameters
- $f \in \mathcal{R}_Q$ // second-layer private key

Ensure: $\mathbf{EVK}_{\tau,\Delta}$ // blind rotation evaluation keys

$\mathbf{evk}_0 \leftarrow \mathrm{NTRU}'_{Q,f,\tau}(X^{s_0}/f)$
for $i = 1 \mathrel{..} (n-1)$ **do**
  $\mathbf{evk}_i \leftarrow \mathrm{NTRU}'_{Q,f,\tau}(X^{s_i})$
end for
$\mathbf{evk}_n \leftarrow \mathrm{NTRU}'_{Q,f,\tau}(X^{-\sum_{i=0}^{n-1} s_i})$
$S \leftarrow \left\{\frac{2N}{q} i + 1\right\}_{i=1}^{q-1}$ // all elements $j \in S$ are odd
for $j \in S$ **do**
  $\mathbf{ksk}_j \leftarrow \mathsf{NTRU.AutoKGen}(j, f)$
end for
$\mathbf{EVK}_{\tau,\Delta} \leftarrow (\mathbf{evk}_0, \ldots, \mathbf{evk}_n, \{\mathbf{ksk}_j\}_{j \in S})$

Performing the Blind Rotation

Introduction

Fully Homomorphic Encryption

Fully Homomorphic Encryption

XZDDF Bootstrapping

Modification of XZDDF

Benchmark Tests

Conclusions

Algorithm 11 XZDDF.BREval

Require:

$(\mathbf{a}, b) = \mathsf{LWE}_{\mathbf{s},q}(m) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$

$r(X) \in \mathcal{R}_Q$ // rotation polynomial

$\mathbf{EVK}_{\tau,\Delta} = (\mathbf{evk}_0, \ldots, \mathbf{evk}_n, \{\mathbf{ksk}_j\}_{j\in S})$

Ensure: $\mathsf{ACC} = \mathrm{NTRU}_{Q,f,\tau,\Delta}\left(r(X^{\frac{2N}{q}}) \cdot X^{\frac{2N}{q}(-b+\sum_{i=0}^{n-1} a_i s_i)}\right)$

for $i = 1 \mathrel{..} (n-1)$ **do**

  $w_i \leftarrow \frac{2N}{q} a_i + 1$

  $w_i' \leftarrow w_i^{-1} \mod 2N$

end for

$w_n' \leftarrow 1$

$\mathsf{ACC} \leftarrow \Delta \cdot r(X^{\frac{2N}{q} w_0'}) \cdot X^{-\frac{2N}{q} b w_0'}$

for $i = 1 \mathrel{..} (n-1)$ **do**

  $\mathsf{ACC} \leftarrow \mathsf{ACC} \odot \mathbf{evk}_i$

  if $w_i w_{i+1}' \neq 1$ **then**

    $\mathsf{ACC} \leftarrow \mathsf{NTRU.EvalAuto}(\mathsf{ACC}, \mathbf{ksk}_{w_i w_{i+1}'})$

  endif

end for

$\mathsf{ACC} \leftarrow \mathsf{ACC} \odot \mathbf{evk}_n$

Extraction

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

- After the blind rotation, we get a ciphertext

$$c = \mathrm{NTRU}_{Q,f,\tau,\Delta}\left(r(X^{\frac{2N}{q}}) \cdot X^{\frac{2N}{q}(-b+\sum_{i=0}^{n-1} a_i s_i)}\right)$$

- Define

$$\mathbf{f} = (f_0, \ldots, f_{N-1}) \in \mathbb{Z}_Q^N$$
$$\hat{\mathbf{c}} = (c_0, -c_{N-1}, \ldots, -c_1) \in \mathbb{Z}_Q^N$$

- Then

$$(\hat{\mathbf{c}}, 0) \in \mathbb{Z}_Q^N \times \mathbb{Z}_Q = \mathsf{LWE}_{Q,\mathbf{f}}(m)$$

Introduction

Fully Homomorphic Encryption

Fully Homomorphic Encryption

XZDDF Bootstrapping

Modification of XZDDF

Benchmark Tests

Conclusions

Modification of XZDDF

The Problem...

Introduction

Fully Homomorphic Encryption

Fully Homomorphic Encryption

XZDDF Bootstrapping

Modification of XZDDF

Benchmark Tests

Conclusions

- Want to compute

$$\text{noised}(m) = \text{coeff}_0 \left(r(X^{\frac{2N}{q}}) \cdot X^{\frac{2N}{q}(-b+\sum_{i=0}^{n-1} a_i s_i)} \right),$$

where $r(X^{\frac{2N}{q}}) = \sum_{i=0}^{q-1} iX^{-\frac{2N}{q}\cdot i}$

- But in $\mathcal{R}_Q = \mathbb{Z}_Q[X]/(X^N + 1)$ we have that

$$X^{-i} = \begin{cases} 1, & \text{if } i = 0 \\ -X^{N-i}, & \text{if } 1 \leq i \leq N \\ X^{2N-i}, & \text{if } N+1 \leq i \leq 2N-1. \end{cases}$$

The Problem...

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

- If for example $q = 2N$, we get

$$
\begin{aligned}
r(X) &= \sum_{i=0}^{q-1} iX^{-i} \\
&= -1 \cdot X^{N-1} - 2 \cdot X^{N-2} - \cdots - N + \\
&\quad + (N+1) \cdot X^{N-1} + (N+2) \cdot X^{N-2} + \cdots + (2N-1) \cdot X \\
&= -N + N \cdot X + N \cdot X^2 + \cdots + N \cdot X^{N-1}.
\end{aligned}
$$

- Same problem for any $q|N$

A Suggestion of Solution

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

- Assume $\begin{cases} \text{Boolean operations} \\ \text{Binary messages} \\ \text{Regev-like first-layer encryption} \end{cases}$
- $\mathrm{LWE}^{\mathrm{Regev}}_{q,\mathbf{s}}(m) = \left(\mathbf{a}, b = \langle \mathbf{a}, \mathbf{s} \rangle + m \cdot \frac{q}{t} + e\right) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$
  - We will use $t = 4$

A Suggestion of Solution

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

- Let $\diamond$ denote a binary operation
- Let $c_1 = (\mathbf{a}_1, b_1)$ and $c_2 = (\mathbf{a}_2, b_2)$
- Start by computing $c = c_1 + c_2 = (\mathbf{a}_1 + \mathbf{a}_2, b_1 + b_2) =: (\mathbf{a}, b)$

$$\implies \mathrm{Dec}(c) = \begin{cases} 0, & \text{if } (m_1, m_2) = (0,0) \\ 1, & \text{if } (m_1, m_2) = (0,1) \text{ or } (1,0) \\ 2, & \text{if } (m_1, m_2) = (1,1) \end{cases}$$

LUND UNIVERSITY

A Suggestion of Solution

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

- Define t intervals $I_i = \left[i \cdot \frac{q}{t} - \frac{q}{2t}, i \cdot \frac{q}{t} + \frac{q}{2t}\right) \subset \mathbb{Z}_q$ for $i = 0, 1, 2, t-1$

$$
\begin{aligned}
I_0 &= \left[-\frac{q}{8} = \frac{7q}{8}, \frac{q}{8}\right), \\
I_1 &= \left[\frac{q}{8}, \frac{3q}{8}\right), \\
I_2 &= \left[\frac{3q}{8}, \frac{5q}{8}\right), \\
I_3 &= \left[\frac{5q}{8}, \frac{7q}{8}\right).
\end{aligned}
$$

A Suggestion of Solution

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

- If for example $\diamond = \text{OR}$, we now want a function f_{OR} that maps

$$f_{\text{OR}}: \quad \left(X^{\frac{2N}{q}}\right)^{\text{noised}(m)} \mapsto \begin{cases} 0, & \text{if noised}(m) \in I_0 \\ 1, & \text{if noised}(m) \in I_1 \\ 1, & \text{if noised}(m) \in I_2 \\ 0, & \text{if noised}(m) \in I_3. \end{cases}$$

LUND UNIVERSITY

A Suggestion of Solution

Introduction

Fully Homomorphic Encryption

Fully Homomorphic Encryption

XZDDF Bootstrapping

Modification of XZDDF

Benchmark Tests

Conclusions

- If for example $\diamond = \text{AND}$, we now want a function f_{AND} that maps

$$f_{\text{AND}} : \quad \left(X^{\frac{2N}{q}}\right)^{\text{noised}(m)} \mapsto \begin{cases} 0, & \text{if noised}(m) \in I_0 \\ 0, & \text{if noised}(m) \in I_1 \\ 1, & \text{if noised}(m) \in I_2 \\ 1, & \text{if noised}(m) \in I_3. \end{cases}$$

A Suggestion of Solution

- It turns out (see `OpenFHE`) that $[0, q)$ can always be split into two intervals

$$
\begin{aligned}
I^0 &= I_k \cup I_{(k+1 \bmod 4)} \\
I^1 &= I_{(k+2 \bmod 4)} \cup I_{(k+3 \bmod 4)}
\end{aligned}
$$

- Ex: $\begin{cases} \text{OR} \implies k = 3 \\ \text{AND} \implies k = 0 \end{cases}$

- Let $I^0 = [q_0, q_1)$ and $I^1 = [q_1, q_0)$

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

A Suggestion of Solution

- Negacyclical property of $\mathcal{R}_Q$: $aX^i \equiv -aX^{i+N} \mod (X^N + 1)$
- Use

$$r(X^{\frac{2N}{q}}) = -1 \cdot \left(X^{-\frac{2N}{q}}\right)^0 - 1 \cdot \left(X^{-\frac{2N}{q}}\right)^1 - \cdots - 1 \cdot \left(X^{-\frac{2N}{q}}\right)^{\frac{q}{4}-1} + \\ + 1 \cdot \left(X^{-\frac{2N}{q}}\right)^{\frac{q}{4}} + \cdots + 1 \cdot \left(X^{-\frac{2N}{q}}\right)^{\frac{q}{2}-1}.$$

$$\implies m' := \mathrm{coeff}_0\left(r(X^{\frac{2N}{q}}) \cdot \left(X^{\frac{2N}{q}(\mathrm{noised}(m)+(\frac{q}{4}-q_1))}\right)\right) = \\ = \begin{cases} -1, & \text{if noised}(m) \in [q_0, q_1) = I^0 \\ 1, & \text{if noised}(m) \in [q_1, q_0) = I^1. \end{cases}$$

A Suggestion of Solution

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

- Now, we just want to map

$$m' \mapsto \begin{cases} 0, & \text{if } m' = -1 \\ 1, & \text{if } m' = 1. \end{cases}$$

- $c' = \mathrm{LWE}_{Q,\mathbf{f}}(m') = (\mathbf{a}, b' = \langle \mathbf{a}, \mathbf{s} \rangle + \Delta \cdot m' + e)$
  - Choose $\Delta = \frac{Q}{4} \cdot \frac{1}{2} = \frac{Q}{8}$

$$\implies c' = \mathrm{LWE}_{Q,\mathbf{f}}(m') = \begin{cases} (\mathbf{a}, b' = \langle \mathbf{a}, \mathbf{s} \rangle - \frac{Q}{8} + e), & \text{if } m' = -1 \\ (\mathbf{a}, b' = \langle \mathbf{a}, \mathbf{s} \rangle + \frac{Q}{8} + e), & \text{if } m' = 1. \end{cases}$$

LUND UNIVERSITY

A Suggestion of Solution

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

- Finally, add $Q/8$ to b'

$$c = (\mathbf{a}, b) = \left(\mathbf{a}, b' + \frac{Q}{8}\right)$$

$$\implies c = \mathrm{LWE}_{Q,\mathbf{f}}(m') = \begin{cases} (\mathbf{a}, b = \langle \mathbf{a}, \mathbf{s} \rangle + e), & \text{if } m' = -1 \\ (\mathbf{a}, b = \langle \mathbf{a}, \mathbf{s} \rangle + \frac{Q}{4} + e), & \text{if } m' = 1, \end{cases}$$

$$\implies m = \mathrm{Dec}(c) = \begin{cases} 0, & \text{if } m' = -1 \\ 1, & \text{if } m' = 1, \end{cases}$$

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

LUND UNIVERSITY

Benchmark Tests

Implementation

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

- Implemented XZDDF in `OpenFHE`
- See `https://github.com/SL2000s/masters_thesis_xzddf`

Tests

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

Test	Description
S1:	Generating a bootstrapping key.
S2:	Performing a single bootstrapping.
S3:	Performing an OR operation on two ciphertexts c_0 and c_1.
S4:	Performing an AND operation on two ciphertexts c_0 and c_1.

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

Results

Algorithm	Param.	S1 (ms)	S2 (ms)	S3 (ms)	S4 (ms)
AP	STD128	10541	182	175	175
GINX	STD128	2583	153	145	145
LMKCDEY	STD128L	2121	120	132	134
XZDDF	STD128	2438	174	184	185
XZDDF	P128T	6386	214	216	216
XZDDF	P128G	5820	194	195	195
AP	STD192	38489	651	662	645
GINX	STD192	8546	467	467	468
LMKCDEY	STD192	8833	493	512	435
XZDDF	STD192	8391	626	622	626
XZDDF	P192T	11808	700	699	699
XZDDF	P192G	9989	592	592	592

Conclusions

Introduction

Fully Homomorphic Encryption

Fully Homomorphic Encryption

XZDDF Bootstrapping

Modification of XZDDF

Benchmark Tests

Conclusions

- XZDDF implementation performs quite well at key generation
- XZDDF implementation not as fast as it theoretically should
  - LMKCDEY still seems to be faster
- Bootstrapping is the main bottleneck in all FHE algorithms
- New rotation polynomial works, but only for a special case

Visions and Future Work

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

- Find a rotation polynomial $r(X)$ for the general case?
- More efficient XZDDF implementation?
- FHE still needs to become more efficient
- Bootstrapping 2010: 30 minutes
- Today: 100 ms

LUND UNIVERSITY

Introduction
Fully Homomorphic Encryption
Fully Homomorphic Encryption
XZDDF Bootstrapping
Modification of XZDDF
Benchmark Tests
Conclusions

Thank you for listening!

Questions?

Acknowledgments:

- Supervisor: Qian Guo
- Examiner: Thomas Johansson