Flattening NTRU for Evaluation Key Free Homomorphic Encryption

Yarkın Doröz

Worcester Polytechnic Institute

August 19, 2018

MathCrypt 2018

Homomorphic Encryption

- First FHE Implementation
  - Completed in 2011
  - Lacks performance, e.g. single AND takes 30 seconds
- Brakerski, Gentry and Vaikuntanathan (BGV) [3]
  - Based on LWE
  - Batching to compute parallel messages
  - Noise coping mechanism: modulus switching
- López-Alt, Tromer, Vaikuntanathan (LTV) [6]
  - Based on NTRU
  - Key switching and relinearization
- Bos et. al. (YASHE) [2]
  - Tensor product for better noise management
  - No Decisional Small Polynomial Ratio (DSPR) assumption
- Gentry, Sahai and Waters (GSW) [4]
  - Flattenning operation which decomposes the ciphertext
  - Eliminates relinearization, modulus switching, bootstrapping

Problems

- **Performance**
  - Relinearization and bootstrapping techniques takes long time to compute
  - Relinearization and bootstrapping operations uses evaluation keys – Large memory requirements
- **Security**
  - Recent attacks on the security of the NTRU based schemes
  - Significantly reduce the security

The Subfield Lattice Attack

- Introduced by Albrecht, Bai and Ducas [1]
- Exploits subfield structure (NTRU problem)
- Decisional Small Polynomial Ratio (DSPR) assumption
  – Poor choice of parameters reduce security levels
  – Example: FHE schemes LTV and YASHE'
- More cautious choice of parameters is required:
  – Increasing the lattice dimension
  – Eliminating subfield structure: disables batching
  – Performance problems
- Kirchner and Fouque [5]: A variant of the subfield attack
- Recover secret keys:
  – NTRU-based FHE implementations (YASHE' and LTV-based FHE)
  – Hermite factors of 1.0058, i.e. 80-bit security.
- Security parameter:
  – Stehlé and Steinfeld's NTRU variant [7], i.e. $\sigma = \sqrt{\mathcal{O}(q)}$

Motivation

- Flattenning noise management technique to NTRU based FHE
- Better noise management:
  – Ciphertext multiplication: linear noise increase
- Security only relies on lattice reductions
  – Larger noise distribution $\sigma = \sqrt{2n \log(8nq)} \cdot q^{1/2+\epsilon}$
  – No DSPR assumption
  – Immunity to Subfield Lattice Attacks
- Avoid expensive noise reduction techniques
  – Relinearization
- Smaller Ciphertext sizes
  – YASHE
- No Evaluation Keys
- Achieve noise asymmetry property
  – Compute fast homomorphic multiplications, e.g. level 30 multiplication takes 76 msec.

Stehlé and Steinfeld's NTRU

- Parameters
  - $R = \mathbb{Z}[x]/\langle x+1 \rangle$
  - Message space $\mathbb{Z}_p$
  - Gaussian Distribution χ
  - $f', g \in \chi$
- Secret Key
  - $f = pf' + 1$
- Public Key
  - $h = pf^{-1}g$
- $\text{Enc}(\mu) = c = hs + pe + \mu$
  - $\{s, e\} \in \chi$
- $\text{Dec}(c) = \mu = c \cdot f \pmod p$

Our proposal: F-NTRU Scheme (Preliminaries)

- **Bit-Decomposition:**
  – Convert chipertext to binary polynomial vector

$$\vec{c}(x) = \mathsf{BitDecomp}(c(x)) = [c_{\ell-1}(x)c_{\ell-2}(x)\ldots c_1(x)c_0(x)]$$

- **$\mathsf{BitDecomp}^{-1}$(Ciphertext reconstruction):**
  – Reconstruct ciphertext from vector of polynomials
  – Vector elements might be non-binary polynomials

$$c(x) = \sum_{i=0}^{\ell-1} 2^i \cdot c_i(x)$$

- **Flatten:**

$$\mathsf{Flatten}(\vec{c}(x)) = \mathsf{BitDecomp}(\mathsf{BitDecomp}^{-1}(\vec{c}(x)))$$

Our proposal: F-NTRU Scheme

- **KeyGen:**
  - Choose security parameter λ
  - Choose $q = q(\lambda)$ and $n = n(\lambda)$ and n is power of 2
  - Create two Gaussian Distribution χ_{err} and χ_{key}
  - Secret Key:
    – Sample $g, f' \in \chi_{\text{key}}$

$$f = 2f' + 1$$

  - Public Key:

$$h = 2gf^{-1}$$

Our proposal: F-NTRU Scheme

- **Encrypt(μ):**
  - Sample $s, e \in \chi_{\text{err}}$
  - $\mathsf{Enc}(\mu) = hs + 2e + \mu$
    - Create ciphertext vector
$$\vec{c}(x) = [\mathsf{Enc}(2^{\ell-1}\mu), \mathsf{Enc}(2^{\ell-2}\mu), \ldots, \mathsf{Enc}(2^{0}\mu)]$$
    - Create binary polynomial matrix
$$C = \mathsf{BitDecomp}(\vec{c}^{T}) = \begin{bmatrix} c_{(\ell-1,\ell-1)} & \cdots & c_{(\ell-1,1)} & c_{(\ell-1,0)} \\ c_{(\ell-2,\ell-1)} & \cdots & c_{(\ell-2,1)} & c_{(\ell-2,0)} \\ \cdots & \cdots & \cdots & \cdots \\ c_{(0,\ell-1)} & \cdots & c_{(0,1)} & c_{(0,0)} \end{bmatrix}$$
- **Decrypt(C):**
  - Compute $\mathsf{BitDecomp}^{-1}(C) = [\vec{c}[\ell-1], \ldots, \vec{c}[1], \vec{c}[0]]$
  - Choose first element on the ciphertext array $\vec{c}[0]$
  - Compute $\lfloor \vec{c}[0] \cdot f \rceil \bmod 2 = \mu$
- **Homomorphic Eval.**
$$C' = \mathsf{Flatten}(C + \tilde{C}) \qquad C' = \mathsf{Flatten}(C \cdot \tilde{C})$$

Our proposal: F-NTRU Scheme

- **Homomorphic AND.**
  – $C' = \mathsf{Flatten}(C \cdot \tilde{C}) = \left[\vec{c'}_3, \vec{c'}_2, \vec{c'}_1, \vec{c'}_0\right]^\top$

$$\begin{bmatrix} c_{(3,3)}+\mu & c_{(3,2)} & c_{(3,1)} & c_{(3,0)} \\ c_{(2,3)} & c_{(2,2)}+\mu & c_{(2,1)} & c_{(2,0)} \\ c_{(1,3)} & c_{(1,2)} & c_{(1,1)}+\mu & c_{(1,0)} \\ c_{(0,3)} & c_{(0,2)} & c_{(0,1)} & c_{(0,0)}+\mu \end{bmatrix} \cdot \begin{bmatrix} \tilde{c}_{(3,3)}+\tilde{\mu} & \tilde{c}_{(3,2)} & \tilde{c}_{(3,1)} & \tilde{c}_{(3,0)} \\ \tilde{c}_{(2,3)} & \tilde{c}_{(2,2)}+\tilde{\mu} & \tilde{c}_{(2,1)} & \tilde{c}_{(2,0)} \\ \tilde{c}_{(1,3)} & \tilde{c}_{(1,2)} & \tilde{c}_{(1,1)}+\tilde{\mu} & \tilde{c}_{(1,0)} \\ \tilde{c}_{(0,3)} & \tilde{c}_{(0,2)} & \tilde{c}_{(0,1)} & \tilde{c}_{(0,0)}+\tilde{\mu} \end{bmatrix}$$

$$\begin{array}{l|llll|l}
\vec{c'}_{(0,3)} & c_{(0,3)}\cdot(\tilde{c}_{(3,3)}+\tilde{\mu}) & +c_{(0,2)}\cdot\tilde{c}_{(2,3)} & +c_{(0,1)}\cdot\tilde{c}_{(1,3)} & +(c_{(0,0)}+\mu)\cdot\tilde{c}_{(0,3)} & 8 \\
\vec{c'}_{(0,2)} & c_{(0,3)}\cdot\tilde{c}_{(3,2)} & +c_{(0,2)}\cdot(\tilde{c}_{(2,2)}+\tilde{\mu}) & +c_{(0,1)}\cdot\tilde{c}_{(1,2)} & +(c_{(0,0)}+\mu)\cdot\tilde{c}_{(0,2)} & 4 \\
\vec{c'}_{(0,1)} & c_{(0,3)}\cdot\tilde{c}_{(3,1)} & +c_{(0,2)}\cdot\tilde{c}_{(2,1)} & +c_{(0,1)}\cdot(\tilde{c}_{(1,1)}+\tilde{\mu}) & +(c_{(0,0)}+\mu)\cdot\tilde{c}_{(0,1)} & 2 \\
\vec{c'}_{(0,0)} & c_{(0,3)}\cdot\tilde{c}_{(3,0)} & +c_{(0,2)}\cdot\tilde{c}_{(2,0)} & +c_{(0,1)}\cdot\tilde{c}_{(1,0)} & +(c_{(0,0)}+\mu)\cdot(\tilde{c}_{(0,0)}+\tilde{\mu}) & 1 \\
\hline
c'_0 & \multicolumn{5}{l}{\underbrace{c_{(0,3)}\cdot\tilde{c}_3+c_{(0,2)}\cdot\tilde{c}_2+c_{(0,1)}\cdot\tilde{c}_1+c_{(0,0)}\cdot\tilde{c}_0+c_0\cdot\tilde{\mu}+\tilde{c}_0\cdot\mu}_{\bar{c}_0}+\mu\cdot\tilde{\mu}}
\end{array}$$

$$c'_i = \underbrace{\sum_{j=0}^{\ell-1} c_{(i,j)}\cdot\tilde{c}_j + c_i\cdot\tilde{\mu} + \tilde{c}_i\cdot\mu}_{\bar{c}_i} + 2^i(\mu\cdot\tilde{\mu}).$$

Our proposal: F-NTRU Scheme

- **Homomorphic XOR**
  - $C' = \mathsf{Flatten}(C + \tilde{C}) = \left[\vec{c'}_3, \vec{c'}_2, \vec{c'}_1, \vec{c'}_0\right]^\top$

$$\begin{bmatrix} c_{(3,3)} + \tilde{c}_{(3,3)} + \mu + \tilde{\mu} & c_{(3,2)} + \tilde{c}_{(3,2)} & c_{(3,1)} + \tilde{c}_{(3,1)} & c_{(3,0)} + \tilde{c}_{(3,0)} \\ c_{(2,3)} + \tilde{c}_{(2,3)} & c_{(2,2)} + \tilde{c}_{(2,2)} + \mu + \tilde{\mu} & c_{(2,1)} + \tilde{c}_{(2,1)} & c_{(2,0)} + \tilde{c}_{(2,0)} \\ c_{(1,3)} + \tilde{c}_{(1,3)} & c_{(1,2)} + \tilde{c}_{(1,2)} & c_{(1,1)} + \tilde{c}_{(1,1)} + \mu + \tilde{\mu} & c_{(1,0)} + \tilde{c}_{(1,0)} \\ c_{(0,3)} + \tilde{c}_{(0,3)} & c_{(0,2)} + \tilde{c}_{(0,2)} & c_{(0,1)} + \tilde{c}_{(0,1)} & c_{(0,0)} + \tilde{c}_{(0,0)} + \mu + \tilde{\mu} \end{bmatrix}$$

$$\begin{array}{c|l|c} \vec{c'}_{(0,3)} & c_{(0,3)} + \tilde{c}_{(0,3)} & 8 \\ \vec{c'}_{(0,2)} & c_{(0,2)} + \tilde{c}_{(0,2)} & 4 \\ \vec{c'}_{(0,1)} & c_{(0,1)} + \tilde{c}_{(0,1)} & 2 \\ \vec{c'}_{(0,0)} & c_{(0,0)} + \tilde{c}_{(0,0)} + \mu + \tilde{\mu} & 1 \\ \hline c'_0 & \underbrace{c_0 + \tilde{c}_0}_{\bar{c}_0} + \mu \cdot \tilde{\mu} & \end{array}$$

- Form of ciphertext elements

$$[(c_3 + \tilde{c}_3) + 8(\mu + \tilde{\mu}), (c_2 + \tilde{c}_2) + 4(\mu + \tilde{\mu}),$$
$$(c_1 + \tilde{c}_1) + 2(\mu + \tilde{\mu}), (c_0 + \tilde{c}_0) + 1(\mu + \tilde{\mu})]$$

Optimizations

- Large matrix size: $\ell^2 = \mathcal{O}((\log q)^2)$
  - Using a higher radix system, i.e. 2^ω
  - Ciphertext size reduction by ω^2
- Long matrix multiplication time:
  - Naive method: $\mathcal{O}(\ell^3)$
  - Coppersmith-Winograd algorithm: $\mathcal{O}(\ell^{2.374})$
  - Leveraging the special structure of ciphertext
    – Matrix-Vector multiplication $\mathcal{O}(\ell^2)$
    – Recall $\mathsf{BitDecomp}^{-1}(C) = [c_3, c_2, c_1, c_0]$
    – Matrix-Matrix to Matrix-Vector multiplication

    Matrix-Matrix:

$$\tilde{C} = C \cdot C'$$

    Matrix-Vector:

$$\mathsf{BitDecomp}^{-1}(\tilde{C}) = C \cdot \mathsf{BitDecomp}^{-1}(C')$$

Security Analysis

- Adoption of parameters form Sthelé and Steinfeld's NTRU
- Indistinguishability under chosen-plaintext attack (IND-CPA)

$$\sigma_{\text{key}} > 2n\sqrt{\log 8nq} \cdot q^{1/2+\epsilon}$$

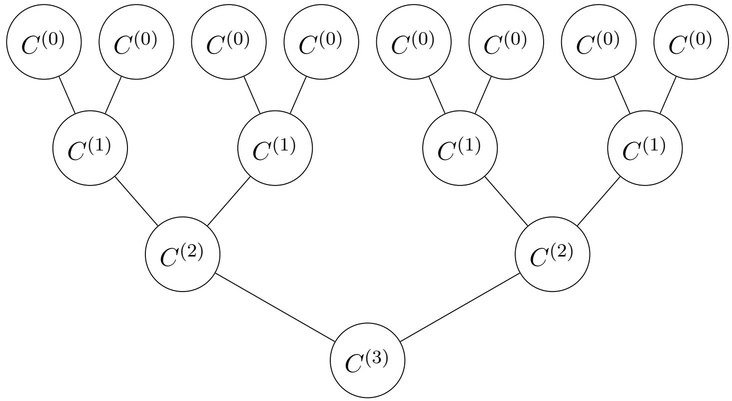
- Statistical distance Δ between uniformly random and Gaussian distributed selected polynomials: $\Delta \leq 2^{3n} q^{-\lfloor \epsilon n \rfloor}$
- R-LWE security distribution $\sigma_{\text{err}} > \sqrt{n \log n}$
- Hermite Factor
  - Work by van de Pol and Smart
  - Fixed Hermite factor for all the lattice dimensions is not true

$$\log(q) \leq \min_{n \leq m} \frac{m^2 \log(\delta(m)) + m \log(\sigma/\alpha)}{m-n}$$

$$\alpha = \sqrt{-\log(\epsilon)/\pi}$$

Noise Analysis

- **Multiplication: Binary Tree**
  – $C^{(i)} = C^{(i-1)} \cdot C^{(i-1)}$



- Each ciphertext element on the $\mathsf{BitDecomp}^{-1}(C^{(i)})$

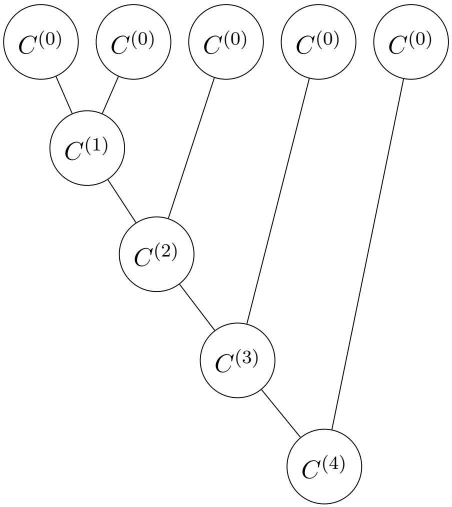
$$c_j^{(i)} = \sum_{k=0}^{\ell-1} c_{(j,k)} \cdot \tilde{c}_k^{(i-1)} + c_j^{(i-1)} \cdot \tilde{\mu} + \tilde{c}_j^{(i-1)} \cdot \mu + 2^j(\mu \cdot \tilde{\mu}).$$

- Worst-case analysis:

$$B_i \leq \ell n(2^\omega - 1)B_{(i-1)} + 2n^{2^i} B_{(i-1)} + n^{2^{i+1}}(2B_{\text{key}} + 1)$$

Noise

- **Multiplication: Left-to-Right**
  – Take advantage of the noise asymmetry



- Worst-case analysis:

$$\begin{aligned} B_i = ||fy_i||_\infty &\leq [2n^2 B_{\text{err}}(3B_{\text{key}}+1)(2^w-1)\ell] \\ &+ [+2{\color{red}n^{i+2}}B_{\text{err}}(3B_{\text{key}}+1)] \\ &+ [nB_{i-1}] + [{\color{red}n^{i+2}}(2B_{\text{key}}+1)] \end{aligned}$$

Noise

- **Single Bit Encryption**
  – Improves scalability

$$\begin{aligned} B_i &\leq [2n^2 B_{\text{key}} B_{\text{err}} (2^w - 1)\ell \\ &+ 2n^2 B_{\text{err}} (2B_{\text{key}} + 1)(2^w - 1)\ell] \\ &+ [2nB_{\text{err}} B_{\text{key}} + 2nB_{\text{err}} (2B_{\text{key}} + 1)] \\ &+ [B_{i-1}] + [(2B_{\text{key}} + 1)] \end{aligned}$$

- Rewrite the equation

$$B_i \leq B_{i-1} + B_{\text{constant}}$$

- Noise complexity for level L (2^L multiplication)

$$\underbrace{\mathcal{O}(n^{2^L})}_{\text{Binary Tree}} \rightarrow \underbrace{\mathcal{O}(n^L)}_{\text{Left-to-Right}} \rightarrow \underbrace{\mathcal{O}(2^L n^2)}_{\text{Single Bit}}$$

Circuit Evaluation

- Single bit encryption should be preserved!
- Homomorphic evaluation
  – Ciphertexts still need to hold 0 or 1
- Restriction to circuit computation
  – NAND (universal) gates
- Gates:
  - NOT: $C = I_N - A$
  - AND: $C = A \cdot B$
  - NAND: $C = I_N - A \cdot B$
  - XOR: $C = (I_N - A) \cdot B + A \cdot (I_N - B) = A + B - 2A \cdot B$
  - OR: $C = I_N - ((I_N - A) \cdot (I_N - B)) = A + B - A \cdot B$

Comparison and Results

- Complexity
  - $\ell = \log q / \omega$

	F-NTRU	YASHE
Eval. Key Size	-	$\mathcal{O}(\ell^3 n \log q)$
Ciphertext Size	$\mathcal{O}(\ell n \log q)$	$\mathcal{O}(n \log q)$
Final Ciphertext Size	$\mathcal{O}(n \log q)$	$\mathcal{O}(n \log q)$
AND Eval.	$\mathcal{O}(\ell^2)$	$\mathcal{O}(\ell^2)$
One Sided AND Eval.	$\mathcal{O}(\ell^2)$	$\mathcal{O}(\ell)$
Key-Switching	-	$\mathcal{O}(\ell^3)$

- Parameters
  - Security level λ
  - Required $(\log n, \log q)$ pairs for multiplicative level L

	F-NTRU		YASHE
L	$\lambda \geq 80$	$\lambda \geq 128$	$\lambda \geq 128$
5	(12,136)	(12,136)	(11,359)
10	(12,147)	(13,152)	(13,840)
20	(12,169)	(13,173)	(14,1705)
30	(13,195)	(13,195)	(14,2538)

Comparison and Results

- Key and Ciphertext sizes

	Evaluation Key	Ciphertext		
	YASHE	F-NTRU	F-NTRU	YASHE
	$\lambda \geq 80$	$\lambda \geq 80$	$\lambda \geq 128$	$\lambda \geq 80$
L	$\omega = 2$	$\omega = 16$	$\omega = 16$	$\omega = 2$
5	3.86 TB	578 KB	578 KB	87 KB
10	478 TB	675 KB	1444 KB	820 KB
20	n/a	892 KB	1870 KB	3.3 MB
30	n/a	2376 KB	2376 KB	4.9 MB

- Timings (in msec)
  - Intel Xeon E5-2637v2 64-bit (3.5 Ghz)
  - 125 GBs of RAM
  - C as thread number

	F-NTRU $\lambda \geq 80$			F-NTRU $\lambda \geq 128$		
L	C=1	C=4	C=8	C=1	C=4	C=8
5	43.5	25.1	24.4	43.5	25.1	24.4
10	53.3	29.8	30.8	110.7	74.2	60.7
20	60.0	32.0	31.2	133.4	68.1	72.5
30	145.9	92.5	76.0	145.9	92.5	76.0

Conclusion

- Presented a new FHE scheme F-NTRU
- Adopt a new noise management technique from GSW scheme
  – Flattenning
- Eliminate
  – Evaluation keys
  – Key Switching
  – Relinearization
- Analyzed security and noise performance
- Resilient against the Subfield Attacks
- NO DSPR assumption
- Supports deep homomorphic evaluation
  – Ciphertext sizes: ~ 2 MB for 30 multiplicative levels
- Competitive speeds:
  – Multiplication takes 24.4 msec at depth 5 and 76 msec at depth 30

Questions?

Thank You!

[1] Martin Albrecht, Shi Bai, and Léo Ducas, *A subfield lattice attack on overstretched ntru assumptions* (Matthew Robshaw and Jonathan Katz, eds.), Springer Berlin Heidelberg, Berlin, Heidelberg, 2016.

[2] Joppe W. Bos, Kristin Lauter, Jake Loftus, and Michael Naehrig, *Improved security for a ring-based fully homomorphic encryption scheme* (Martijn Stam, ed.), Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[3] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan, *(leveled) fully homomorphic encryption without bootstrapping*, Proceedings of the 3rd innovations in theoretical computer science conference, 2012, pp. 309–325.

[4] Craig Gentry, Amit Sahai, and Brent Waters, *Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based* (Ran Canetti and Juan A. Garay, eds.), Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[5] Paul Kirchner and Pierre-Alain Fouque, *Revisiting lattice attacks on overstretched ntru parameters*, Advances in cryptology – eurocrypt 2017, 2017, pp. 3–26.

[6] Adriana López-Alt, Eran Tromer, and Vinod Vaikuntanathan, *On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption*, Proceedings of the forty-fourth annual acm symposium on theory of computing, 2012, pp. 1219–1234.

[7] Damien Stehlé and Ron Steinfeld, *Making ntru as secure as worst-case problems over ideal lattices* (Kenneth G. Paterson, ed.), Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.