# Limits of Polynomial Packings for $\mathbb{Z}_{p^k}$ and $\mathbb{F}_{p^k}$


Jung Hee Cheon

(Seoul National University & Crypto Lab Inc.)

**Keewoo Lee**

(Seoul National University)

# Sketch

- **Formal & Unified Study of "Polynomial Packing"**
  - ... which appears in various contexts:
  - HE & SHE-based MPC (HE Packing), IT-MPC (RMFE), Correlation Extractor, ZK...
- **Upper Bounds & Impossibility Results**
  - Packing Density, Level-Consistency, & Surjectivity
- **Implications**
  - SHE-based MPC over $\mathbb{Z}_{2^k}$, HE Packing, RMFE

# Definition

# Definition

## Polynomial Packing

$$R^n \;\; (R = \mathbb{Z}_{p^k}, \mathbb{F}_{p^k}) \quad \underset{\text{Unpack}}{\overset{\text{Pack}}{\rightleftarrows}} \quad \mathcal{R} = \mathbb{Z}_{p^t}[x]/f(x)$$

* Packing Density = $\log(|R|^n) \,/\, \log(|\mathcal{R}|)$

---

## Degree-$D$ Packing

$$\begin{tikzcd} R^n \arrow[r, "\text{Pack}"] \arrow[d, "P(\cdot)"'] \arrow[dr, phantom, "\circlearrowleft"] & \mathcal{R} \arrow[d, "P(\cdot)"] \\ R^n & \mathcal{R} \arrow[l, "\text{Unpack}"] \end{tikzcd}$$

$P(\cdot)$: (Multivariate) Polynomial of Degree $\leq D$

# Definition

## Remark: Unpack may differ for each multiplicative level.

**Definition 3.2 (Degree-$D$ Packing).** *Let $\mathcal{P} = (\mathsf{Pack}_i, \mathsf{Unpack}_i)_{i=1}^{D}$ be a collection of packing methods for $R^n$ into $\mathcal{R}$. We call $\mathcal{P}$ a degree-D packing method, if it satisfies the following for all $1 \le i \le D$:*

- *If $a(x), b(x)$ satisfy $\mathsf{Unpack}_i(a(x)) = \boldsymbol{a}$, $\mathsf{Unpack}_i(b(x)) = \boldsymbol{b}$ for $\boldsymbol{a}, \boldsymbol{b} \in R^n$, then $\mathsf{Unpack}_i(a(x) \pm b(x)) = \boldsymbol{a} \pm \boldsymbol{b}$ holds;*
- *If $a(x), b(x)$ satisfy $\mathsf{Unpack}_s(a(x)) = \boldsymbol{a}$, $\mathsf{Unpack}_t(b(x)) = \boldsymbol{b}$ for $\boldsymbol{a}, \boldsymbol{b} \in R^n$ and $s, t \in \mathbb{Z}^+$ such that $s + t = i$, then $\mathsf{Unpack}_i(a(x) \cdot b(x)) = \boldsymbol{a} \odot \boldsymbol{b}$ holds.*

# Contexts & Examples

# Homomorphic Encryption

- HE supports computation on encrypted data.

$$
\begin{tikzcd}
m \arrow[r, "\mathrm{Enc}"] \arrow[d, "P(\cdot)"'] \arrow[dr, phantom, "\circlearrowleft"] & Enc(m) \arrow[d, "\mathrm{Eval}_P"] \\
P(m) & Enc(P(m)) \arrow[l, "\mathrm{Dec}"]
\end{tikzcd}
$$

# Homomorphic Encryption

- HE supports computation on encrypted data.
- Concurrent HE schemes are often based on RLWE for efficiency.
  - e.g. BGV, FV

$$
\begin{tikzcd}
m \arrow[r, "\mathsf{Enc}"] \arrow[d, "P(\cdot)"'] \arrow[dr, phantom, "\circlearrowleft"] & Enc(m) \arrow[d, "\mathsf{Eval}_P"] \\
P(m) & Enc(P(m)) \arrow[l, "\mathsf{Dec}"]
\end{tikzcd}
$$

# Homomorphic Encryption

- HE supports computation on encrypted data.
- Concurrent HE schemes are often based on RLWE for efficiency.
  - ➢ e.g. BGV, FV
  - ➢ Practical Usability?

$$\begin{tikzcd}
\mathbb{Z}_t[x]/\Phi_M(x) & \\
m(x) \arrow[u, phantom, sloped, "\in"] \arrow[r, "\mathrm{Enc}"] \arrow[d, "P(\cdot)"'] \arrow[dr, phantom, "\circlearrowleft"] & Enc(m(x)) \arrow[d, "\mathrm{Eval}_P"] \\
P(m(x)) & Enc(P(m(x))) \arrow[l, "\mathrm{Dec}"]
\end{tikzcd}$$

# HE Packing [Smart-Vercauteren;PKC10]

$$
\begin{tikzcd}
R^n & \mathbb{Z}_t[x]/\Phi_M(x) & \\
\rotatebox{90}{$\in$} & \rotatebox{90}{$\in$} & \\
\vec{m} \arrow[r, "\text{Pack}"] \arrow[d, "P(\cdot)"'] \arrow[dr, phantom, "\circlearrowright"] & m(x) \arrow[r, "\text{Enc}"] \arrow[d, "P(\cdot)"'] \arrow[dr, phantom, "\circlearrowright"] & Enc(m(x)) \arrow[d, "\text{Eval}_P"] \\
P(\vec{m}) & P(m(x)) \arrow[l, "\text{Unpack}"] & Enc(P(m(x))) \arrow[l, "\text{Dec}"]
\end{tikzcd}
$$

# HE Packing: Examples

- $\Phi_M(x) = \prod_{i=1}^{r} F_i(x) \bmod p$ & $\deg F_i = d$

- **Traditional Packing Method** [Smart-Vercauteren;PKC10]
  - $(\mathbb{F}_{p^d})^r \overset{\cong}{\to} \mathbb{Z}_p[x]/\Phi_M(x)$ : **Degree-∞**, Density$= 1$
  - $(\mathbb{Z}_p)^{\varphi(M)} \overset{\cong}{\to} \mathbb{Z}_p[x]/\Phi_M(x)$, if $\Phi_M(x)$ fully splits mod $p$ : **Degree-∞**, Density$= 1$
- **HELib Packing for $\mathbb{Z}_{p^k}$-messages** [Gentry-Halevi-Smart;PKC12], [Halevi-Shoup;Eurocrypt15]
  - $(\mathbb{Z}_{p^k})^r \to \mathbb{Z}_{p^k}[x]/\Phi_M(x)$ : **Degree-∞**, Density$= 1/d$
- **Recent Developments in SHE-based MPC over $\mathbb{Z}_{2^k}$ (SPDZ-family)**
  - Overdrive2k [Orsini-Smart-Vercauteren;CT-RSA20] : **Degree-2**, Density$\approx 1/5$
  - MHz2k [**Cheon**-Kim-**Lee**;Crypto21] : **Degree-2**, **Density$\approx 1/2$**

# RMFE [Cascudo-Cramer-Xing-Yuan;Crypto18]

Using "Large Field" is often required due to:

1. **Mathematical Structures**
   - **Shamir Secret Sharing** : We can interpolate at most $q$ points over $\mathbb{F}_q$
2. **Security**
   - **Linear MAC** : $MAC_\alpha(x) \coloneqq \alpha \cdot x$ over $\mathbb{F}_q$ has soundness error $1/q$

# RMFE [Cascudo-Cramer-Xing-Yuan;Crypto18]

- **Reverse Multiplication-Friendly Embedding (RMFE)**
  - Embed algebraic structure of copies of small field (e.g. $\mathbb{F}_2^n$) into a larger field (e.g. $\mathbb{F}_{2^d}$).
  - Essentially, RMFEs are **Degree-2** packings from $\mathbb{F}_q^n$ into $\mathbb{F}_{q^d} \cong \mathbb{F}_q[x]/f(x)$.
  - Now a Standard Tool in IT-MPC (e.g. [DLN;Crypto19], [DLSV;Euro20], [PS;Euro21], ...)
  - Also used in ZK (e.g. [BMRS;Crypto21], [CG;FC22])

# Theorems & Implications

# Packing Density

- **Theorem**
  - Roughly speaking, density of degree-$D$ packing method $\lesssim 1/D$
  - For $d =$ [deg. of irreducible quotient poly.],

$$[\text{packing density}] \leq \frac{1}{D} + \frac{1}{d}\left(1 - \frac{1}{D}\right)$$

- **Implications**
  1. MHz2k [CKL;Crypto21] achieves near-optimal density (as a degree-2 packing for $\mathbb{Z}_{2^k}$)
  2. ($\mathbb{F}_{p^k}$ Version) New and more general proof for upper bound on rate of RMFE
  3. First upper bound on rate of RMFE over Galois rings [Cramer-Rambaud-Xing;Crypto21]

# Level-Consistency

- **Motivation**
  - FHE, Homomorphic computation between different mult. levels (e.g. Reshare Protocol)
- **Theorem**
  - If level-consistency holds,

$$n \leq [\ \#\text{ of distinct } \textbf{irred.} \text{ factors of quotient poly. mod } p\ ]$$

- **Implications**
  1. Optimality of HELib packing with respect to packing density and level-consistency
  2. Impossibility of Efficient Level-Consistent HE Packing for $\mathbb{Z}_{2^k}$
  3. Importance of "Constant Packing Trick" of MHz2k for Level-dependent packings

# Surjectivity

- **Motivation**
  - Malicious “Packer” might leverage invalid packings in protocols.
- **Theorem**
  - If surjectivity holds,

$$n \leq [\ \#\ \text{of distinct } \textbf{linear} \text{ factors of quotient poly. mod } p^k\ ]$$

- **Implications**

1. Impossibility of Surjective HE Packing for $\mathbb{Z}_{2^k}$
2. Necessity of ZKPoMK in HE-based MPC over $\mathbb{Z}_{2^k}$ (First conceptualized in MHz2k)

# Summary

- Formal & Unified Study of Polynomial Packing
  - which appears in various contexts:
  - HE & SHE-based MPC (HE Packing), IT-MPC (RMFE), Correlation Extractor, ZK…
- Upper Bounds & Impossibility Results
  - Packing Density, Level-consistency, and Surjectivity

# Summary

- Implications on SHE-based MPC over $\mathbb{Z}_{2^k}$ (c.f. MHz2k [CKL;Crypto21])
  1. MHz2k achieves near-optimal packing density
  2. Importance of "Constant Packing Trick" of MHz2k for Level-dependent packings
  3. Necessity of ZKPoMK in HE-based MPC over $\mathbb{Z}_{2^k}$ (First conceptualized in MHz2k)
- Implication on HE Packing
  1. Optimality of HELib packing with respect to packing density and level-consistency
- Implications on RMFE
  1. New and more general proof for upper bound on rate of RMFE
  2. First upper bound on rate of RMFE over Galois rings (c.f. [CRX;Crypto21])

# Conclusion

1. Packing is not a question asked before secure computation.
   - Messages are “static” (e.g. PKE): No need to worry about structure of messages.
2. Packing is a question shared by secure computation primitives.
   - Messages are “dynamic” (HE, MPC, ZK): Algebraic structure of messages matters.
3. There might be more questions of like this!
   - Especially when we try to apply secure computation to real-life problems.

# Thank You!

* ePrint: ia.cr/2021/1033
* E-mail: activecondor@snu.ac.kr
* Webpage: keewoolee.github.io