Re: Bootstrapping Fonctionnel

Algorithme pour évaluer en aveugle $f : \mathbb{Z}_p \to \mathbb{Z}_p$ et réduire le bruit

- Étant donné une LUT $L = (f(0), f(1), \ldots, f(p-1))$ et un chiffré $[x]$ de vecteur $x \in \mathbb{Z}_p^w$

1. $T(X) \leftarrow Hermite(L)$ // Interpolation “nettoyante” en un polynôme en $X = e^{2\pi i x}$
2. $P(X) \leftarrow Chebyshev(x \mapsto e^{2\pi i x})$ // Interpolation “précise” de l’exponentielle complexe
3. $[e^{2\pi i x}] \leftarrow Evaluate(P(X), [x])$
4. $\uparrow Evaluate(T(X), [e^{2\pi i x}])$

Évaluation avec l’algorithme de Paterson-Stockmeyer

Conclusion

Résultats et conséquences de [AKP25]

- Pour évaluer une LUT de 8 bits vers 8 bits
  - [AKP25] single-threaded i7 avec 64GB RAM: $< 1ms$ amorti sur 65k valeurs ($\approx 50s$ total)
  - [ZamaPBS] TFHE-rs multi-threaded 96-core 740GB RAM: $\approx 200ms$
- Avec CKKS (et un bon choix de paramètre) on peut maintenant
  - Évaluer de manière exacte des fonctions arbitraires
  - Enchaîner autant de calcul qu'on le désire en gardant le bruit sous contrôle
- Passage à l'échelle via la méthode Tree-based de [GBA21]