Hitchhiker's Guide to a Practical Automated TFHE Parameter Setup for Custom Applications

Jakub Klemsa
jakub.klemsa@eurecom.fr

EURECOM
Sophia Antipolis

Motivation

TFHE **cipher** is instantiated with **8 parameters**:

- determine **security level** and plaintext/evaluation **error rate**
  ⇒ parameters are **application-specific**,
- vast impact on **performance** ⇒ optimization problem,

⇒ **need for a tool** for TFHE parameter setup.

Given application specifications:

+ **fully automated** TFHE parameter setup,
+ optimized for **best performance**,
+ unified approach ⇒ allows to **compare** different parameters:
  - same target security & error rates, aim for best parameters,
  - e.g., different homomorphization in digit-based arithmetics [1].

Application Specifications

Specify application needs (security & error rates) by **three parameters**:

A. **bit-security level** denoted λ;
B. requested **cleartext space bit-precision** denoted π; and
C. bound on the number of homomorphic additions before the sample gets bootstrapped, denoted $2^{2\Delta}$, referred **quadratic weights**;

⇒ input parameters for our TFHE parameter setup tool.

A. Bit-Security Level λ

Observation 1. At fixed security λ, the logarithm of stddev of LWE **noise** (den. α), is roughly linear in the LWE **dimension** n (with factor den. s_λ):

$$-\log_2(\alpha) \approx s_\lambda \cdot n; \tag{1}$$

cf. Figure 1. Due to the **collision attack**, the relation is limited to $n \geq 2\lambda$, also the behavior changes for $-\log_2(\alpha) > \tau$ with τ the torus precision.

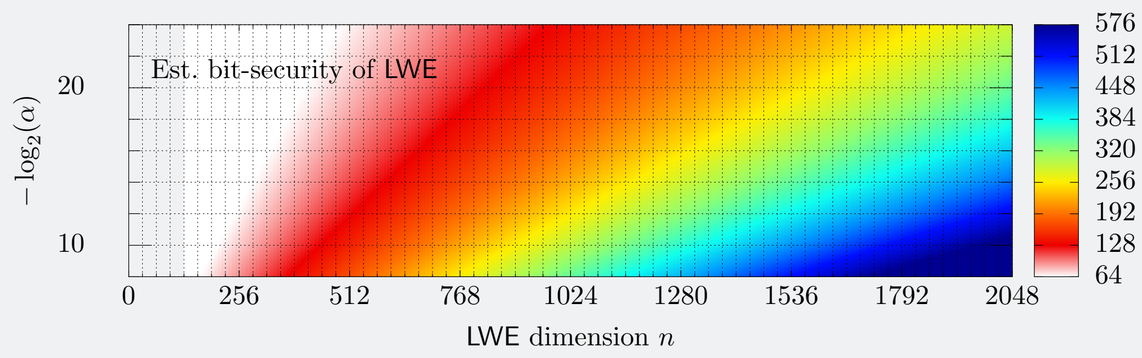

Figure 1: LWE bit-security level λ as estimated by the *LWE Estimator* by Albrecht et al. [2]. For $\lambda = 128$ bits, $s_\lambda \approx 0.0235$.

B. Cleartext Space Bit-Precision π

Before selecting an appropriate π, bare in mind:

i. complexity of TFHE bootstrapping is roughly **exponential** in π,
   - practical times for up to $\pi \approx 6$ bits (cf. Figure 2),
ii. bootstrapping Look-Up Table (LUT) is inherently **negacyclic**:

$$LUT(2^{\pi-1} + m) = -LUT(m), \quad m \in [\![0, 2^{\pi-1}[\![. \tag{2}$$

C. Quadratic weights $2^{2\Delta}$

Observation 2. LWE **noises accumulate** with each homomorphic addition: for indep. TLWE samples $\mathbf{c}_i$ with equal noise variance denoted V_0:

$$\mathsf{Var}\Big(\mathsf{Err}\Big(\sum w_i \cdot \mathbf{c}_i\Big)\Big) = \underbrace{\sum w_i^2}_{2^{2\Delta}} \cdot \underbrace{\mathsf{Var}(\mathsf{Err}(\mathbf{c}_i))}_{V_0}, \quad w_i \in \mathbb{Z}. \tag{3}$$

To ensure **correct LUT evaluation** during bootstrapping:

- **bound** on $2^{2\Delta}$ before a sample gets bootstrapped (refresh noise),
- $2^{2\Delta}$ ... sum of squared weights; Δ ... bits of stddev of addit'l noise.

Acknowledgements

This work was supported by the MESRI-BMBF project UPCARE (ANR-20-CYAL-0003-01).

Parameter Restrictions

Goal: **bound the noise** of a fresh(ly bootstrapped) sample, s.t.

i. **limited number of additions** can be performed (cf. $2^{2\Delta}$); and
ii. the noise can be **refreshed correctly** during bootstrapping.

2^π cleartext values ⇒ max error $\overset{!}{<} 1/2^{\pi+1}$ ... by 3σ-rule:

$$V_{\max} \leq 2^{2\Delta} V_0 + V_{round} \overset{!}{\leq} \frac{1}{3^2 \cdot 2^{2\pi+2}}, \qquad \text{where } V_{round} = \frac{n+1}{48N^2}. \tag{4}$$

V_0 depends on implementation, for plain TFHE [3]:

$$V_0 \leq \underbrace{2nlN2^{2(\gamma-1)}V_{\mathsf{BK}}(N)}_{(\heartsuit)} + \underbrace{n(1+N)2^{-2(\gamma l+1)}}_{(\diamondsuit)} + \underbrace{\mathsf{Var}\big(\mathsf{Err}(u,v)\big)}_{=0} + \\ + \underbrace{tN2^{2(\kappa-1)}V_{\mathsf{KS}}(n)}_{(\clubsuit)} + \underbrace{2^{-2(\kappa t+1)}N}_{(\spadesuit)}. \tag{5}$$

To derive **good** TFHE parameters, we need to:

i. **satisfy the bound** (4) (error budget), using (5); and
ii. check their quality in terms of **bootstrapping time**.

Experimental Results

	$\mathbb{Z}/16\mathbb{Z}$ Demo: $\pi = 6$, $2^{2\Delta} = 2$		Repo with exp. code
	Orig. param's [4]	New param's	
N, n ; γ, l	2 048, 750 ; 7, 3	2 048, 766 ; 21, 1	
κ, t ; $\log(\alpha_{\mathsf{BK},\mathsf{KS}})$	2, 7 ; −52, −18	3, 5 ; −48, −18	
λ ; t_{BS}	128.2 ; 199.6 ms	131.2 ; 124.6 ms	
η_C, η_m[%]	86.0, 85.5	91.2, 90.5	

Table 1: Original and newly identified TFHE parameters for the $\mathbb{Z}/16\mathbb{Z}$ demo [4] with $\pi = 6$ bits. 500 runs with Concrete [5] on Intel Core i7-7800X. η_C and η_m stand resp. for the usage of the $3\sigma_{\max}$ error budget as calculated by Concrete and as measured after decryption. Experimental code at `https://gitlab.eurecom.fr/fakub/tfhe-param-testing`.

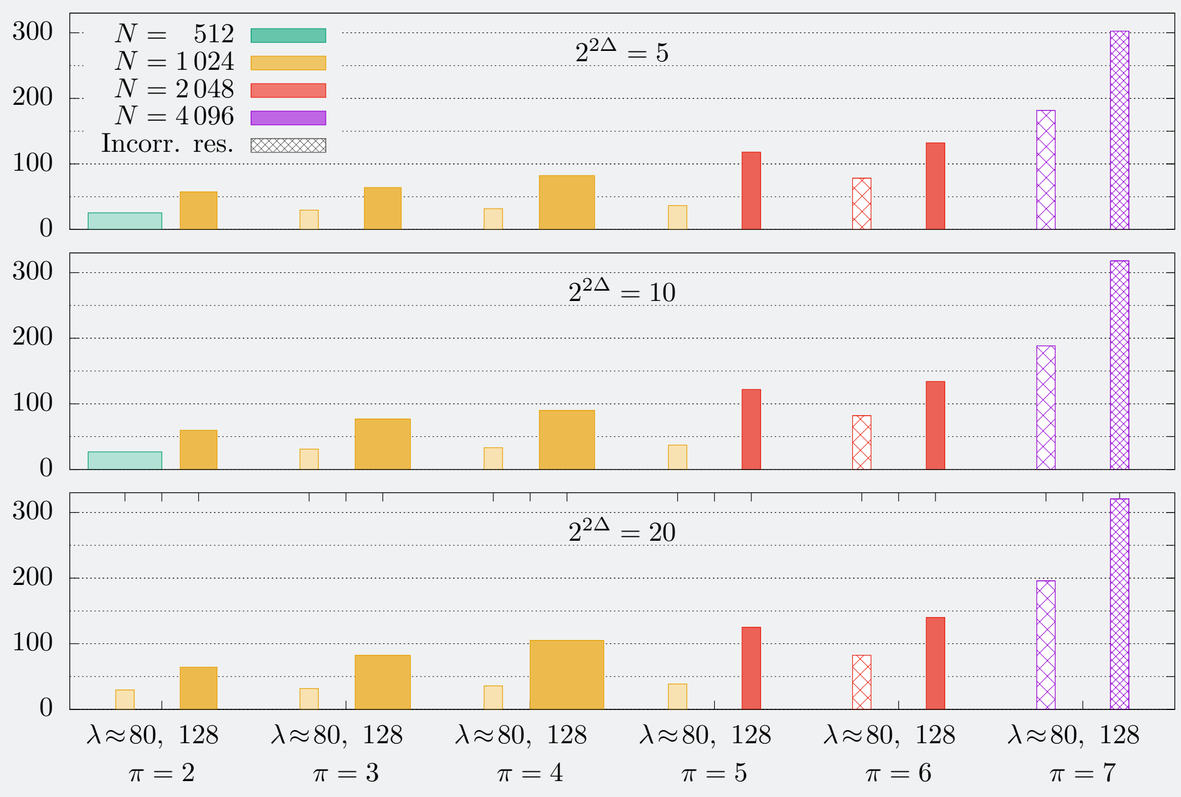

Figure 2: Bootstrapping times of best TFHE parameters with $\eta_C < 100\%$ for various scenarios, chosen automatically. The width of the bars represents $l \in [\![1, 4]\!]$. Hatched bars represent incorrect results, presumably due to $\log(\alpha) < -64$ being out of Concrete's `v0.1.11` supported range.

References

[1] J. Klemsa and M. Önen, "Parallel Operations over TFHE-Encrypted Multi-Digit Integers," ser. CODASPY '22, 2022.

[2] M. R. Albrecht, B. R. Curtis, A. Deo, *et al.*, *LWE Estimator*, `https://bitbucket.org/malb/lwe-estimator`, 2018.

[3] I. Chillotti, N. Gama, M. Georgieva, and M. Izabachène, "TFHE: fast fully homomorphic encryption over the torus," *Journal of Cryptology*, vol. 33, no. 1, pp. 34–91, 2020.

[4] Zama, *Demo Z/8Z*, `https://github.com/zama-ai/demo_z8z`, 2021.

[5] Zama, *CONCRETE*, `https://concrete.zama.ai`, 2021.