

Order-Four Symmetry in BGV Bootstrapping: Faster Digit Extraction for Large Primes

Zhenyu Xiong^{1,2}, Mingsheng Wang^{1,2*}, Zhedong Wang³, and Han Wang^{1,2}

¹ State Key Laboratory of Cyberspace Security Defense, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China

xiongzhenyu@iie.ac.cn, wangmingsheng@iie.ac.cn, wanghan@iie.ac.cn

² School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China

³ Shanghai Jiao Tong University, Shanghai, China

wzdstill@sjtu.edu.cn

Abstract. Bootstrapping is the computational bottleneck of BGV/BFV fully homomorphic encryption, scaling particularly poorly with large plaintext primes. Its two dominant stages: digit extraction and linear transforms. Recent work has reduced the digit-extraction polynomial degree via null-polynomial lattices [7] (Eurocrypt’23) and bounded-support constructions [16] (Eurocrypt’24), but both evaluate the reduced polynomial via generic Paterson–Stockmeyer at cost $O(\sqrt{d})$.

We present two algebraic optimizations that address both stages simultaneously. For digit extraction, we prove that choosing the auxiliary radix A with $A^2 \equiv -1 \pmod{p}$ induces an order-four character filter, forcing the canonical digit-extraction polynomial to satisfy $P_A(AX) + AP_A(X) = AX$ and eliminating all monomials X^k with $k \not\equiv 1, 3 \pmod{4}$. The resulting structured decomposition $P_A(X) = \frac{1}{2}X + X^3Q(X^4)$ reduces non-scalar multiplications from $O(\sqrt{d})$ to $O(\sqrt{d/r})$. For linear transforms, we provide first concrete instantiation of a Galois-structured mixed-radix butterfly decomposition for non-power-of-two cyclotomics, reducing the automorphism count from $O(\sqrt{D})$ to $O(\log D)$.

On the standard NTT-friendly large-prime set ($p = 65537$, $m = 2^{16}$, 32768 slots), our single-threaded HElib implementation achieves a $1.85\times$ digit-extraction speedup (20.06 s to 10.81 s) and a $1.27\times$ total thin-bootstrapping speedup (42.3 s to 33.3 s) over the state-of-the-art Ma et al. baseline, the three stages the method does not touch moving by at most one per cent. All comparisons are made against the Ma et al. baselines, run in the identical pipeline at the same auxiliary radix; across nineteen encrypted parameter sets with $1297 \leq p \leq 65537$, the digit-extraction speedup is $1.72\text{--}1.88\times$ on general cyclotomic rings ($37637 \leq m \leq 65047$) and $1.84\text{--}2.13\times$ on the power-of-two ring $m = 2^{16}$. Every set we recommend is quoted with concrete bit security. In detail, our implementation is publicly available⁴.

Keywords: Fully homomorphic encryption · BGV bootstrapping · Digit extraction · Character theory · Paterson–Stockmeyer evaluation · Non-power-of-two cyclotomics

* Corresponding author.

⁴ <https://github.com/acpk-hash/order4-bgv-bootstrap>

1 Introduction

Since Gentry introduced the first fully homomorphic encryption (FHE) construction [10], research in FHE has attracted broad attention and improved several orders of magnitude in both latency and throughput [3,2,6]. Among the various FHE schemes, BGV [3] and BFV [2,6] are the most popular for homomorphic integer arithmetic, offering SIMD functionality in which a plaintext encodes an array of elements and operations are performed simultaneously on each slot. Notably, these schemes follow Gentry’s blueprint where each ciphertext carries noise that grows during homomorphic evaluation; consequently, evaluating functions of arbitrary complexity requires *bootstrapping*, homomorphically evaluating the decryption circuit to produce a fresh lower-noise ciphertext. The bootstrapping procedure developed by Halevi and Shoup [13,12] and implemented in HELib is the foundational blueprint for all subsequent works [4,15,7,16,17]. In practice, polynomial evaluations during bootstrapping account for most of the computational cost in HELib, which are $3\times$ to $50\times$ more expensive than all other operations combined [7]. Despite significant progress, BGV bootstrapping remains the main bottleneck for large plaintext primes p , where the cost scales very poorly [11].

The BGV thin bootstrapping pipeline is dominated by three computational stages that together consume over 99% of the total running time:

- **Step 1: Homomorphic SlotToCoeff.** This step implements the $\mathbb{F}_p$ -linear encoding map

$$\text{S2C} : \mathbb{F}_p^\ell \xrightarrow{\sim} R_p, \quad [m_0, \dots, m_{\ell-1}] \mapsto m(X),$$

which moves the slot values into the powerful-basis coefficients of the plaintext polynomial. Homomorphically, this is realized as a matrix–vector product $L(\text{ct}) = \sum_{i=0}^{D-1} \text{diag}_i \cdot \sigma^i(\text{ct})$ over Galois automorphisms σ^i , evaluated via the BSGS method [14] with $O(\sqrt{D})$ key-switches per dimension- D block.

- **Step 2: Homomorphic digit extraction.** After modulus raising and the inner product, each slot contains $w = p^{e-r}m_i + \epsilon_i \in \mathbb{Z}_{p^e}$ with $|\epsilon_i| \leq B$. The digit-extraction stage computes

$$f_{\text{ext}} : \mathbb{Z}_{p^e} \rightarrow \mathbb{Z}_{p^r}, \quad w \mapsto \lfloor w/p^{e-r} \rfloor = m_i$$

by homomorphically evaluating a polynomial $P_A \in \mathbb{F}_p[X]$ on the ciphertext. The evaluation cost is $O(\sqrt{\deg P_A})$ non-scalar multiplications under the Paterson–Stockmeyer algorithm.

- **Step 3: Homomorphic CoeffToSlot.** This step implements the inverse decoding map

$$\text{C2S} : R_p \xrightarrow{\sim} \mathbb{F}_p^\ell, \quad m(X) \mapsto [m_0, \dots, m_{\ell-1}],$$

moving the powerful-basis coefficients back into slots. The computational structure is identical to Step 1, with cost $O(\sqrt{D})$ key-switches.

In our work, following this bootstrapping framework above, the digit extraction stage dominates the total bootstrapping cost. We observe that the bounded

support S_A admits an order-4 rotational symmetry when $A^2 \equiv -1 \pmod{p}$, which we exploit via character decomposition to construct a digit extraction polynomial with reduced effective evaluation degree. For the linear transforms, we extend the $O(\log D)$ butterfly decomposition to non-power of two cyclotomics, providing the first concrete instantiation on the standard HElib.

1.1 Related Work

We overview recent optimizations for BGV/BFV bootstrapping, with emphasis on the two computational bottlenecks. Specifically, we focus on digit-extraction polynomial constructions that reduce evaluation cost, and homomorphic linear-transform methods that reduce automorphism count. Table 1 summarizes the asymptotic complexity comparison.

Digit-extraction polynomial optimization. Halevi and Shoup [13] proposed the foundational digit removal using a lifting polynomial of degree p^{e-1} , achieving evaluation cost $O((e-1)\sqrt{p})$. Subsequently, Chen and Han [4] introduced the lowest-digit-removal polynomial of degree $(p-1)(e-1)+1$, which bypasses intermediate lifting steps and reduces cost to $O(\sqrt{(e-1)(p-1)})$. Furthermore, Huo et al. [15] provided tighter degree bounds for the same polynomial family. In a different direction, Geelen et al. [7] exploited the lattice structure of null polynomials modulo p^e to obtain sparse representations with 50% fewer coefficients, lowering complexity to $O(\sqrt[4]{p} \sqrt[4]{e})$; their follow-up work [9] further unified BGV/BFV bootstrapping under a single framework. Most recently, Ma et al. [16] constructed bounded-support null polynomials that reduce the effective degree from $O(\sqrt{p})$ to $O(B^2)$ and the evaluation cost to $O(B)$, where $B = O(\sqrt{h})$ bounds the digit coordinates. Nevertheless, these methods treat the reduced polynomial's evaluation as a black box; no prior work exploits the internal monomial-pattern structure to reduce cost below the Paterson Stockmeyer.

Homomorphic linear-transform acceleration. Halevi and Shoup [14] introduced the BSGS method with hoisting, requiring $O(\sqrt{D})$ automorphisms per dimension- D block. Subsequently, Ma et al. [17] achieved $O(\log D)$ automorphisms via homomorphic NTT for power-of-two cyclotomics. In parallel, Peikert and Pepin [22] established a general Galois-theoretic framework showing that $O(\log D)$ is achievable for arbitrary abelian number rings. Additionally, Okada et al. [20] exploited Galois automorphisms for BFV polynomial evaluation. For power-of-two rings specifically, Ma et al. [18] further reduce cost via subring secret encapsulation. However, for non-power-of-two Vercauteren [8] develop GBFV bootstrapping; and Choe et al. [5] leverage discrete CKKS for integer cleaning. These approaches are complementary to our work.

Our work. Unlike prior digit-extraction methods that reduce the polynomial degree, we reduce the *evaluation cost* at fixed degree by exploiting monomial-pattern structure via character-theoretic decomposition. Additionally, we provide the first concrete $O(\log D)$ linear-transform instantiation for non-power-of-two cyclotomics, identifying $D \mid p^d - 1$ as the key enabler. Both optimizations

Table 1: Asymptotic complexity comparison of BGV bootstrapping methods. p : plaintext prime; e : digit precision; B : bounded-support radius; D : linear-transform block dimension; r : character filter order, in our setting $r = 4$.

Technique	Method	DE Degree	DE Eval. Cost	LT Automorphisms
Lifting poly.	Halevi-Shoup [13]	$O(p^{e-1})$	$O((e-1)\sqrt{p})$	$O(\sqrt{D})$
	Chen-Han [4]	$O(pe)$	$O(\sqrt{pe})$	$O(\sqrt{D})$
	Geelen et al. [7]	$O(\sqrt{p} \sqrt[4]{e})$	$O(\sqrt[4]{p} \sqrt[8]{e})$	$O(\sqrt{D})$
Bounded support	Ma et al. [16]	$O(B^2)$	$O(B)$	$O(\sqrt{D})$
Structured LT	Ma et al. [17] (pow-2)	N/A	N/A	$O(\log D)$
	Peikert-Pepin [22]	N/A	N/A	$O(\log D)$
This work	Char. filter + Galois LT	$O(B^2)$	$O(B/\sqrt{r})$	$O(\log D)^\dagger$

[†]Per-layer count with native sparse-diagonal evaluation; in BSGS-based libraries the practical benefit depends on the evaluation framework (see §5).

compose with the asymmetric BSGS method [24] and are orthogonal to all prior degree-reduction techniques.

Table 1 compares our construction with prior BGV bootstrapping methods across three metrics. Existing digit-extraction approaches face a fundamental limitation: lifting-based methods [13,4,15] suffer from polynomial degrees that grow with p or e ; lattice-based sparsification [7] reduces coefficients but not the effective evaluation degree; and bounded-support constructions [16] achieve optimal degree $O(B^2)$ but treat the resulting polynomial as a black box under Paterson-Stockmeyer evaluation. For linear transforms, existing $O(\log D)$ methods [17,22] are either restricted to power-of-two cyclotomics or remain uninstantiated on practical parameter sets. Our construction achieves the same optimal degree as Ma et al. but reduces the evaluation cost by a factor of $\sqrt{r}$ via character-theoretic decomposition, while simultaneously providing the first concrete $O(\log D)$ instantiation for non-power-of-two cyclotomics. Detailed experimental results are presented in Section 6.

1.2 Our Contributions

In this work, we investigate new algebraic techniques to accelerate both dominant stages of large-prime BGV bootstrapping. We address the fundamental challenge of reducing homomorphic evaluation cost beyond the information-theoretic degree bound through character-theoretic representation theory and structured linear-transform decomposition. Our main contributions are summarized below:

1. **Character-structured digit-extraction polynomial with optimal evaluation cost.** We introduce a novel construction that exploits the order-4 cyclic symmetry of the bounded support S_A when $A^2 \equiv -1 \pmod{p}$. By establishing the functional equation $P_A(AX) + A \cdot P_A(X) = AX$ and interpreting it as a covariance relation on the quotient space $\mathbb{F}_p[X]/(G_{S_A}(X))$, we prove that the digit-extraction polynomial admits the structured factorization $P_A(X) = \frac{1}{2}X + X^3Q(X^4)$, reducing the effective Paterson-Stockmeyer degree by a factor of 4. We further establish that the resulting term count

- $(|S_A| - 1)/4 + 1$ is tight via Schur’s lemma, confirm global optimality of $A = 256$ via exhaustive search, and prove a multiplicative interaction theorem with asymmetric BSGS [24] yielding an additional $r^{1/t}$ key-switch reduction.
2. **First concrete $O(\log D)$ linear-transform instantiation for non-power-of-two cyclotomics.** We provide a new construction for homomorphic linear transforms on the standard HELib parameter set ($m = 50731 = 97 \times 523$). Our approach identifies the number-theoretic condition $D \mid p^d - 1$ as the key enabler for a mixed-radix Cooley–Tukey butterfly factorization with slot-field twiddle factors. For the dominant $D = 96$ block, this reduces automorphisms from 18 (BSGS) to 11 per cascade (-39%). We validate correctness via Rader’s algorithm on the actual slot ring $\mathbb{F}_{p^{18}}$ and analyze the practical interaction with HELib’s BSGS framework, and we additionally provide a first *encrypted* prototype of the butterfly transform on the full ring, correct on all 2784 slots, whose wall-clock advantage over BSGS emerges at larger block dimensions ($D = 576$).
 3. **Implementation and comprehensive evaluation.** We implement both optimizations in HELib and conduct extensive evaluations at the recommended operating points of Section 6.1. On the NTT-friendly set ($p = 65537$, $m = 2^{16}$, 32768 slots) the results demonstrate a $1.85\times$ digit-extraction speedup and a $1.27\times$ total thin-bootstrapping speedup over the Ma et al. baseline, while the three stages the contribution does not touch move by at most one per cent. Beyond this headline set the filter is validated on nineteen encrypted parameter sets with $1297 \leq p \leq 65537$, always against the state-of-the-art Ma et al. baselines [16,17]: the digit-extraction speedup ranges over $1.72\text{--}1.88\times$ on general cyclotomic rings ($37637 \leq m \leq 65047$) and $1.84\text{--}2.13\times$ on the power-of-two ring $m = 2^{16}$. Both optimizations are drop-in compatible with existing implementations and compose with orthogonal techniques including subring secret encapsulation and homomorphic NTT.

1.3 Techniques Overview

Briefly, our approach exploits algebraic structure in the digit-extraction polynomial and the slot-field arithmetic of non-power-of-two cyclotomics to reduce both dominant bootstrapping costs simultaneously. We first construct a character structured digit extraction polynomial with reduced effective evaluation degree, then provide a concrete butterfly decomposition for the homomorphic linear transforms. Since the two optimizations target different stages and leverage different algebraic tools, we present each separately.

Recall Ma et al.’s [16] and our improvement. For digit extraction’s improvement, after Ma et al.’s bounded support reduction, the digit extraction polynomial P_A achieves the optimal degree $|S_A| - 2$ and is evaluated at cost $\approx 2\sqrt{d}$ non-scalar multiplications via Paterson Stockmeyer. However, this cost depends solely on the polynomial degree; in particular, the internal monomial structure of P_A remains entirely unexploited. In contrast, our key observation is that a specific algebraic choice of A reveals hidden structure in P_A that reduces the *effective* evaluation degree from d to $d/4$, thereby cutting the evaluation cost nearly in half without modifying the polynomial degree itself.

Insight 1: Quarter-turn symmetry induces a character filter. We observe that the auxiliary radix A controls not only the support S_A but also the *algebraic structure* of the quotient space $V = \mathbb{F}_p[X]/(G_{S_A}(X))$. When $A^2 \equiv -1 \pmod{p}$, multiplication by the scalar A acts as a quarter-turn on the digit coordinate grid:

$$A(\eta A + \lambda) = \lambda A + (-\eta), \quad (\eta, \lambda) \mapsto (\lambda, -\eta),$$

inducing a cyclic group action $\langle T_A \rangle \cong \mathbb{Z}/4\mathbb{Z}$ on V via the substitution operator $T_A : f(X) \mapsto f(AX)$. We prove that P_A satisfies the functional equation $P_A(AX) + A \cdot P_A(X) = AX$, which we identify as a covariance relation forcing P_A into the character eigenspace V_3 . By the isotypic decomposition, the surviving monomials are exactly $\{k = 1\} \cup \{k \equiv 3 \pmod{4}\}$, yielding term count $(|S_A| - 1)/4 + 1$.

- *Structured factorization.* The monomial pattern forces $P_A(X) = \frac{1}{2}X + X^3 \cdot Q(X^4)$ with $\deg Q = (|S_A| - 5)/4$, enabling evaluation at effective degree $d/4$ and reducing the PS cost by $\approx 45\%$.
- *Tightness.* The term count equals the eigenspace dimension $\dim V_3 + 1$. This upper bound is unconditional; that it is *attained* is verified computationally for each parameter set we report, so no further reduction is available on that support at those parameters.
- *Multiplicative interaction.* The filter changes the polynomial and not the evaluator, so it should compose multiplicatively with asymmetric BSGS [24] for a combined $O((d/r)^{1/t})$ key-switch count, an extra $r^{1/t}$ beyond either technique alone. We have not measured this: it needs key-switching material HELib does not generate, so it remains an expectation.

Recall state-of-the-art linear transforms [17,18,22] and our instantiation. For power-of-two cyclotomics, Ma et al. [17] achieved $O(\log D)$ linear transforms via homomorphic NTT, and Ma et al. [18] further accelerated bootstrapping through subring secret encapsulation, which is currently the fastest known implementation for power-of-two rings. Independently, Peikert and Pepin [22] established a general Galois theoretic framework proving that $O(\log D)$ is achievable for arbitrary abelian number rings. However, their work is primarily theoretical: it provides no concrete instantiation, no explicit twiddle factor computation, and no implementation on practical parameter sets. Consequently, for non-power-of-two cyclotomics no concrete $O(\log D)$ instantiation exists.

Insight 2: Slot field roots enable butterfly decomposition for non-power-of-two cyclotomics. We independently arrived at a concrete instantiation of the Peikert Pepin framework for the HELib parameter set ($m = 50731$, $D = 96$), identifying the number theoretic condition $D \mid p^d - 1$ as the key enabler. When this condition holds, all D -th roots of unity exist in the slot field $\mathbb{F}_{p^d}$, and the Step2Matrix admits a complete mixed radix butterfly factorization:

- Good Thomas separation: For $D = p_1^{a_1} \times \cdots \times p_k^{a_k}$ with coprime factors, the CRT bijection decomposes the transform into independent sub-DFTs without twiddle factors. The CRT permutations are free slot rearrangements in HELib.

- Cooley Tukey factorization: Each prime power sub-DFT is factored into radix p_i butterfly layers with twiddle factors drawn from the slot field.

The total automorphism count per cascade is $\sum_i (2a_i - 1)(p_i - 1) = O(\log D)$, compared to $O(\sqrt{D})$ for BSGS; for our parameters ($D = 96 = 2^5 \times 3$) this yields 11 automorphisms per cascade vs. 18 for BSGS. We validate correctness via Rader’s algorithm on the actual slot ring $\mathbb{F}_{p^{18}}$ and analyze the practical interaction with HELib’s existing BSGS framework.

Paper organization. Section 2 introduces notation and algebraic tools. Section 3 reviews existing digit-extraction constructions. Section 4 presents our character-structured polynomial construction with correctness proof and complexity analysis. Section 5 describes the Galois-structured linear transform. Section 6.1 derives the concrete security of the recommended parameter sets and Section 6 reports experimental results. Section 7 concludes.

2 Preliminaries

2.1 Notation

The symbols used throughout this paper are summarized in Table 2. Additional notation is introduced where first used. Throughout, the digit pair of the auxiliary encoding is written (η, λ) , N denotes a generic set size, and in the evaluation analysis d denotes the degree of the polynomial being evaluated.

Table 2: Summary of notation.

Symbol	Meaning	Symbol	Meaning
p	Plaintext prime	e	Hensel lifting exponent
r	Precision parameter (p^r)	M	Cyclotomic order
$\mathbb{F}_p$	Prime field $\mathbb{Z}/p\mathbb{Z}$	$\mathbb{F}_p^\times$	Multiplicative group of $\mathbb{F}_p$
$\mathbb{Z}_{p^e}$	Residue ring $\mathbb{Z}/p^e\mathbb{Z}$	R	Cyclotomic ring $\mathbb{Z}[X]/\Phi_M(X)$
R_q	Ciphertext ring R/qR	q	Ciphertext modulus
n	Ring degree $\varphi(M)$	ℓ	Number of plaintext slots
d	Slot extension degree	h	Secret key Hamming weight
s	Secret key polynomial	σ	Noise std. deviation
$\text{ord}(A)$	Mult. order of A in $\mathbb{F}_p^\times$	$[a]_q$	Balanced reduction mod q
$\Phi_M(X)$	M -th cyclotomic polynomial	$\varphi(M)$	Euler’s totient function
$G_S(X)$	$\prod_{s \in S} (X - s)$	S_A	Bounded support set
P_A	Digit-extraction polynomial	B	Support bound
D	Linear-transform block dimension	$\text{char}(K)$	Characteristic of the field K

2.2 The BGV Scheme

We introduce the BGV scheme [3]. Since BFV [2,6] has an identical bootstrapping procedure, all results apply to both.

Secret key and ciphertext. The secret key $s \in R$ is sampled with bounded Hamming weight h : exactly h coefficients are set to ± 1 uniformly, and the rest are zero. A BGV ciphertext encrypting $m \in R_{p^r}$ takes the form

$$\text{Enc}_s(m; p^r; \epsilon) = (b, a) = (-as + m + p^r \epsilon, a) \in R_q^2,$$

where q is coprime to p , $a \leftarrow R_q$ is uniform, and ϵ is sampled from a discrete Gaussian with standard deviation σ . Decryption computes $[b + as]_{p^r} = [m + p^r \epsilon]_{p^r} = m$, which is correct when $\|m + p^r \epsilon\|_\infty < q/2$.

Plaintext batching (SIMD). When $\gcd(M, p) = 1$, the cyclotomic polynomial factors modulo p^r as $\Phi_M(X) \equiv F_1(X) \cdots F_\ell(X) \pmod{p^r}$, where each F_i is irreducible of degree $d = \text{ord}_M(p)$ and $\ell = n/d$. By the Chinese Remainder Theorem, the plaintext ring decomposes as

$$R_{p^r} = \mathbb{Z}_{p^r}[X]/(\Phi_M(X)) \cong \prod_{i=1}^{\ell} \mathbb{Z}_{p^r}[X]/(F_i(X)),$$

identifying each plaintext with a vector of ℓ slots, each holding an element of the slot algebra $\mathbb{Z}_{p^r}[\zeta]$ where ζ is a root of F_1 . Homomorphic operations act component-wise on slots (SIMD). For thin bootstrapping, $d = 1$ and each slot stores a scalar in $\mathbb{Z}_{p^r}$.

Key switching and automorphisms. Given $\text{Enc}_s(m)$, key switching produces $\text{Enc}_{s'}(m)$ for a different secret key s' , using a pre-computed key-switching matrix. KS operation is the most expensive atomic step. Galois automorphisms $\tau_j : X \mapsto X^j$ for $j \in \mathbb{Z}_M^*$ permute plaintext slots and are implemented via key switching.

Homomorphic operations. Given $\text{Enc}_s(m_0; p^r)$ and $\text{Enc}_s(m_1; p^r)$, one can compute $\text{Enc}_s(m_0 + m_1; p^r)$ (addition) and $\text{Enc}_s(m_0 \cdot m_1; p^r)$ (multiplication). Multiplication produces a degree-2 ciphertext that must be *relinearized* (key-switched from s^2 to s) back to the standard form. We call this a *non-scalar multiplication*—the dominant cost in polynomial evaluation.

Noise and capacity. The ratio $q/\|m + p^r \epsilon\|_\infty$, called the *capacity*, measures how many further operations a ciphertext can sustain before decryption fails. Each multiplication roughly doubles the *bit-size* of the noise; modulus switching (reducing q to a smaller q') trades capacity for a proportional noise reduction.

2.3 BGV Thin Bootstrapping

As homomorphic computations progress, ciphertext capacity decreases until no further operations can be performed correctly. Bootstrapping restores capacity by homomorphically evaluating the scheme's own decryption circuit on the input ciphertext $\text{ct} = \text{Enc}_{s'}(m; p^r)$, producing a fresh ciphertext with reduced noise.

The original BGV bootstrapping of Halevi and Shoup [13], known as *fat bootstrapping*, operates on fully packed ciphertexts where each slot stores an extension field element in $\mathbb{Z}_{p^r}[\zeta]$ with $[\mathbb{Z}_{p^r}[\zeta] : \mathbb{Z}_{p^r}] = d$. After the CoeffToSlot step, the d coefficients of each slot element must be unpacked into d separate ciphertexts, and digit removal is applied independently to each one. This d -fold

replication multiplies both the number of polynomial evaluations and the number of linear transforms by a factor of d , making fat bootstrapping prohibitively expensive when the slot extension degree is large (e.g., $d = 18$ for $m = 50731$).

In 2018, Chen and Han [4] observed that when each slot stores only a scalar in $\mathbb{Z}_{p^r}$ (i.e., $d = 1$), the unpacking step is eliminated entirely and digit removal is applied to a single ciphertext. This *thin bootstrapping* procedure, illustrated in Figure 1, consists of four steps:

- **SlotToCoeff:** Move the slot values into the powerful-basis coefficients via a homomorphic linear transform, producing an encryption of a plaintext polynomial $m \in R_{p^r}$.
- **Decryption formula simplification and inner product:** Perform modulus switching and key switching to obtain $(b, a) \in R^2$ satisfying $\text{pwl}(b + as) \equiv p^{e-r}m + \epsilon \pmod{p^e}$ for some $e > r$. The homomorphic inner product $b + a \cdot \text{Enc}_{s'}(s; p^e)$ outputs a high-capacity ciphertext encrypting $p^{e-r}m + \epsilon$ under plaintext modulus p^e .
- **CoeffToSlot:** The inverse linear transform moves the powerful-basis coefficients back into slots. Each slot now contains an integer $p^{e-r}m_i + \epsilon_i \in \mathbb{Z}_{p^e}$.
- **Digit extraction:** A polynomial evaluation on each slot extracts m_i from $p^{e-r}m_i + \epsilon_i$ by removing the lower p -adic digits, recovering the original message with refreshed capacity.

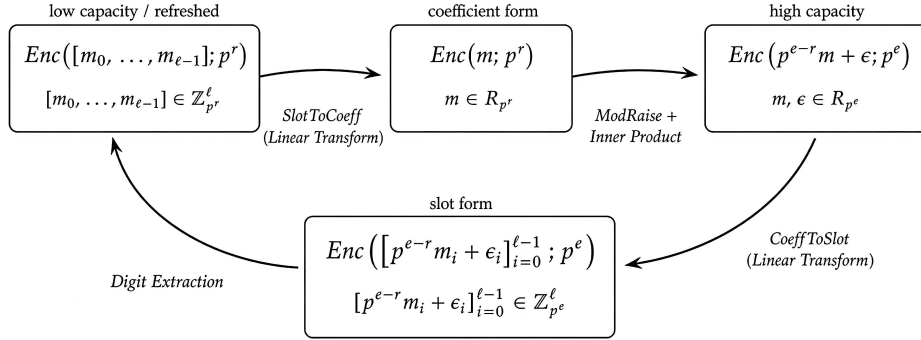


Fig. 1: Thin bootstrapping framework of Chen and Han [4].

2.4 Bounded-Support Digit Extraction

The digit-extraction problem is: given $w = p^{e-r}m_i + \epsilon_i \in \mathbb{Z}_{p^e}$ with $|m_i| < p^r/2$ and $|\epsilon_i| \leq B$, recover m_i by homomorphically evaluating a polynomial that removes the lower p -adic digits. The classical digit-extraction polynomial $G_e(X)$ has degree $(p-1)(e-1)+1$, which is prohibitively large for large primes p .

Ma et al. [16] introduced the *bounded-support* framework to address this. The key observation is that after the overflow-part extraction step in bootstrapping, the slot value w can be decomposed as $w = \eta A + \lambda \pmod{p}$ where $|\eta|, |\lambda| \leq B$ and

$A \in \mathbb{F}_p^\times$ is a fixed auxiliary radix (here η is the high digit and λ the low digit). Geometrically, this means w is determined by a pair (η, λ) on the integer grid $[-B, B]^2$, and the encoding map $\phi_A : (\eta, \lambda) \mapsto \eta A + \lambda \bmod p$ embeds this grid into a small subset $S_A \subset \mathbb{F}_p$ of size $(2B+1)^2 \ll p$. The digit-extraction problem then reduces to: given $x \in S_A$, compute λ (the low digit). Since P_A only needs to be correct on S_A rather than on all of $\mathbb{F}_p$, its degree can be reduced from $(p-1)(e-1)+1$ to $|S_A|-2 = O(B^2)$ by interpolation on S_A , as illustrated in Figure 2.

Definition 1 (*B*-injectivity). An auxiliary radix $A \in \mathbb{F}_p^\times$ is *B*-injective if the encoding map

$$\phi_A : [-B, B]^2 \rightarrow \mathbb{F}_p, \quad (\eta, \lambda) \mapsto \eta A + \lambda \bmod p$$

is injective, i.e., distinct pairs $(\eta, \lambda) \neq (\eta', \lambda')$ in $[-B, B]^2$ produce distinct field elements.

Lemma 1 (Injectivity criterion). Identify A with its balanced representative in $(-p/2, p/2)$. If $2B < |A|$ and $|A| \cdot B + B < p/2$, then A is *B*-injective.

Proof. Let $(\eta_1, \lambda_1), (\eta_2, \lambda_2) \in [-B, B]^2$ satisfy $\eta_1 A + \lambda_1 \equiv \eta_2 A + \lambda_2 \pmod{p}$, and set $x_i = \eta_i A + \lambda_i \in \mathbb{Z}$. By the triangle inequality, $|x_i| \leq |A|B + B < p/2$, hence $|x_1 - x_2| < p$; since p divides $x_1 - x_2$, this forces $x_1 = x_2$ over $\mathbb{Z}$. Then $(\eta_1 - \eta_2)A = \lambda_2 - \lambda_1$. If $\eta_1 \neq \eta_2$, the left-hand side has absolute value at least $|A| > 2B$, while the right-hand side is at most $2B$, a contradiction. Hence $\eta_1 = \eta_2$, and consequently $\lambda_1 = \lambda_2$. $\square$

Definition 2 (Bounded support and digit-extraction polynomial). For *B*-injective A , the bounded support is the image of the encoding:

$$S_A := \text{Im}(\phi_A) = \{\eta A + \lambda \bmod p : |\eta|, |\lambda| \leq B\} \subset \mathbb{F}_p, \quad |S_A| = (2B+1)^2,$$

where $B = O(\sqrt{h})$ is determined by the noise bound after modulus raising (h is the secret key Hamming weight). The digit-extraction polynomial $P_A \in \mathbb{F}_p[X]$ is the unique polynomial of degree $< |S_A|$ satisfying

$$P_A(\eta A + \lambda) = \lambda \quad \forall (\eta, \lambda) \in [-B, B]^2.$$

In other words, P_A is the Lagrange interpolant of the low-digit function $x \mapsto \lambda$ on the finite set S_A .

The symmetry of the grid $[-B, B]^2$ immediately implies P_A structural property:

Proposition 1 (Oddness). Since $S_A = -S_A$ and $P_A(-x) = -P_A(x)$ on S_A , the polynomial P_A is odd: $c_k = 0$ for all even k . The nonzero term count is at most $(|S_A| - 1)/2$.

The proof is deferred to Appendix A. Note that in Ma et al.'s original construction, the auxiliary radix A is treated as a free parameter subject only to the injectivity constraint; any *B*-injective A yields a valid P_A with the same degree $|S_A| - 2$. Our work (Section 4) shows that a specific algebraic choice of A satisfying $A^2 \equiv -1 \pmod{p}$ reveals additional structure in P_A that further reduces the evaluation cost.

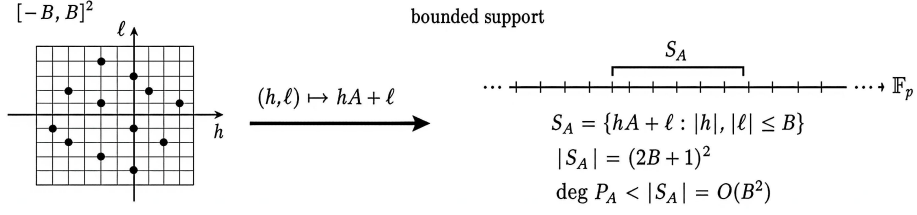


Fig. 2: Ma et al.'s bounded-support digit extraction [16].

2.5 Character Decomposition on Polynomial Quotient Spaces

Our goal is to exploit algebraic structure in $V = \mathbb{F}_p[X]/(G_{S_A}(X))$ to constrain the monomial support of P_A . The key idea: if a group action decomposes V into eigenspaces and P_A lies in a single eigenspace, then its monomial support is restricted to a fraction of the full basis, enabling a faster evaluation strategy.

We develop this in the general setting of polynomial quotient spaces. Let K be a field, $S \subset K$ a finite set with $|S| = N$, and $G_S(X) = \prod_{s \in S} (X - s)$ the vanishing polynomial. The quotient space [23]

$$V := K[X]/(G_S(X))$$

is an N -dimensional K -vector space with monomial basis $\{1, X, X^2, \dots, X^{N-1}\}$. Since G_S splits completely over K , the Chinese Remainder Theorem provides the isomorphism

$$V \cong \prod_{s \in S} K[X]/(X - s) \cong K^S,$$

identifying each $f \in V$ with its evaluation vector $(f(s))_{s \in S} \in K^N$.

Group action via substitution. Suppose $A \in K^\times$ satisfies $A \cdot S = S$, i.e., multiplication by A permutes the elements of S . This induces a K -linear automorphism of V :

Definition 3 (Substitution operator). The substitution operator $T_A : V \rightarrow V$ is defined by

$$T_A : f(X) \mapsto f(AX) \bmod G_S(X).$$

The operator T_A acts diagonally on the monomial basis: $T_A(X^k) = A^k X^k$ for $0 \leq k < N$. Since A has finite multiplicative order $r = \text{ord}(A)$ in $K^\times$, we have $T_A^r = \text{id}_V$, and the cyclic group $G := \langle T_A \rangle \cong \mathbb{Z}/r\mathbb{Z}$ acts on V by K -linear automorphisms.

Eigenspace decomposition. When $\text{char}(K) \nmid r$ and K contains a primitive r -th root of unity ω (which holds for $K = \mathbb{F}_p$ when $r \mid p - 1$), the group algebra $K[G]$ is semisimple by Maschke's theorem in Serre [23]. Since G is cyclic, all irreducible K -representations are one-dimensional characters $\chi_j : T_A \mapsto \omega^j$, and V decomposes into the corresponding eigenspaces:

Theorem 1 (Isotypic decomposition).

$$V = \bigoplus_{j=0}^{r-1} V_j, \quad V_j := \{f \in V : T_A(f) = \omega^j f\}. \quad (1)$$

Since $T_A(X^k) = A^k X^k = \omega^{k \bmod r} X^k$, the monomial X^k lies in $V_{k \bmod r}$. Consequently,

$$\dim V_j = |\{k \in \{0, \dots, N-1\} : k \equiv j \pmod{r}\}| = \lceil (N-j)/r \rceil.$$

Covariance implies eigenspace membership. The following proposition connects the eigenspace decomposition to the monomial support of a specific polynomial. It shows that if $f \in V$ satisfies a covariance relation with respect to T_A , then all but one eigenspace component of f is uniquely determined, and the free component lies in a single V_j .

Proposition 2 (Monomial support constraint). *Suppose $f \in V$ satisfies*

$$T_A(f) = \omega^j \cdot f + h \tag{2}$$

for some fixed $j \in \{0, \dots, r-1\}$ and known $h \in V$. Write $f = \sum_{i=0}^{r-1} f_i$ and $h = \sum_{i=0}^{r-1} h_i$ with $f_i, h_i \in V_i$. Then:

(i) For each $i \neq j$, the component f_i is uniquely determined:

$$f_i = \frac{h_i}{\omega^i - \omega^j}. \tag{3}$$

(ii) The component $f_j \in V_j$ is unconstrained by (2); it contributes at most $\dim V_j = \lceil (N-j)/r \rceil$ free monomials.

Proof. Apply T_A to $f = \sum_i f_i$. Since $T_A(f_i) = \omega^i f_i$, equation (2) becomes

$$\sum_{i=0}^{r-1} \omega^i f_i = \omega^j \sum_{i=0}^{r-1} f_i + \sum_{i=0}^{r-1} h_i.$$

Projecting onto the eigenspace V_i for $i \neq j$ gives $\omega^i f_i = \omega^j f_i + h_i$, i.e., $(\omega^i - \omega^j) f_i = h_i$. Since ω is a primitive r -th root of unity, $\omega^i \neq \omega^j$ for $i \neq j$, so $f_i = h_i/(\omega^i - \omega^j)$ is uniquely determined. For $i = j$, the projection gives $\omega^j f_j = \omega^j f_j + h_j$, which forces $h_j = 0$ as a consistency condition and leaves f_j unconstrained.

In Section 4, we show that P_A satisfies precisely such a covariance relation with $r = 4$ and $j = 3$ when $A^2 \equiv -1 \pmod{p}$, forcing its monomial support into V_3 and reducing the free term count from $(|S_A| - 1)/2$ to $(|S_A| - 1)/4 + 1$. This enables the factored evaluation form $P_A(X) = \frac{1}{2}X + X^3Q(X^4)$ with effective degree $d/4$.

2.6 Homomorphic Polynomial Evaluation

The evaluation cost of the digit-extraction polynomial is determined by the algorithm used to compute $P_A(x)$ homomorphically. We introduce two methods: the standard Paterson Stockmeyer algorithm (Helib baseline) and the asymmetric BSGS algorithm of Wang and Wang [24], which composes with our character filter for further speedup.

Paterson–Stockmeyer (PS) algorithm [21]. In this subsection and throughout the digit-extraction analysis, d denotes the degree of the polynomial. Given a degree- d polynomial $f(X) = \sum_{i=0}^d f_i X^i$ and a ciphertext $\text{ct} \in \text{RLWE}(x)$, the PS algorithm selects $k \approx m \approx \sqrt{d}$ with $d \leq km - 1$ and decomposes

$$f(X) = \sum_{j=0}^{m-1} f^{(j)}(X) \cdot X^{jk}, \quad f^{(j)}(X) = \sum_{i=0}^{k-1} f_{jk+i} X^i.$$

The baby-step and giant-step sets are

$$S_1 = \{1, x, x^2, \dots, x^{k-1}\}, \quad S_2 = \{1, z, z^2, \dots, z^{m-1}\},$$

where $z = x^k$. Each $y_j = f^{(j)}(x)$ is a scalar linear combination of S_1 (no non-scalar multiplications), and the final result $f(x) = \sum_j y_j \cdot z^j$ requires $m - 1$ non-scalar multiplications. The total cost is $\approx 2\sqrt{d}$ non-scalar multiplications plus $O(\log d)$ for the power-chain construction.

Asymmetric BSGS algorithm [24]. Wang and Wang observed that in the standard PS/BSGS, the baby-step set S_1 resides at a shallower multiplicative depth than the giant-step set S_2 , yet both are computed to the same depth. Their asymmetric BSGS algorithm exploits this depth gap by further decomposing S_1 into t smaller subsets:

$$S_1 \subseteq S_1^{(0)} \times S_1^{(1)} \times \dots \times S_1^{(t-1)},$$

where each $|S_1^{(j)}| = O(d^{1/t})$ and $t \geq 2$ is a tunable parameter. Instead of performing key switching after each multiplication in the baby step, the algorithm computes tensor products of the t small sets without intermediate key switches, then performs a single modulus switch and key switch to bring the result to the level of S_2 . This reduces the number of key switches from $O(\sqrt{d})$ to $O(d^{1/t})$, at the cost of increasing the multiplicative depth from $\lceil \log_2 k \rceil + 1$ to $l_2 + 1$ where $l_2 = \lceil \log(k(m-1)) \rceil$. When the effective polynomial degree is reduced (e.g., from d to d/r by exploiting monomial structure), the asymmetric BSGS provides an additional multiplicative $r^{1/t}$ factor of improvement.

3 Previous Homomorphic Digit-Extraction Constructions

This section reviews the four generations of digit-extraction optimizations that progressively reduce the polynomial degree and evaluation cost, establishing the baseline for our construction in Section 4.

Halevi–Shoup lifting [13]. The foundational approach uses a *lifting polynomial* $F_k(X) \in \mathbb{Z}[X]$ of degree p satisfying $F_k(a + bp^j) \equiv a \pmod{p^{j+1}}$ for all $a \in \mathbb{F}_p$, $b \in \mathbb{Z}$, $1 \leq j \leq k$. Starting from $w_{0,0} = w$, the digit removal iterates $w_{i,j} = F_k(w_{i,j-1})$ to zero out successive digits, and shifts via $w_{i,0} = (w - \sum_{t < i} w_{t,i-t} \cdot p^t) / p^i$. The procedure is illustrated in Figure 3(a) for $e = 5$, $r = 2$: horizontal arrows represent evaluations of F_k , and diagonal arrows represent the shifting operation. The total cost is dominated by $ev - v(v+1)/2$ evaluations of F_k (degree p^{e-1}), giving complexity $O((e-1)\sqrt{p})$.

Chen–Han digit extraction [4]. Chen and Han discovered a *digit-extraction polynomial* $G_e(X)$ of degree $(p-1)(e-1)+1$ satisfying $G_e(a) \equiv [a]_p \pmod{p^e}$ for all $a \in \mathbb{Z}$. This allows direct computation $w_{i,e-1-i} = G_{e-i}(w_{i,0})$ without the intermediate lifting steps, as shown in Figure 3(b): dashed arrows bypass the sequential F_k evaluations. The multiplicative depth reduces from $(e-1)\log_2 p$ to $\log_2((p-1)(e-1)+1)$, and the evaluation cost to $O(\sqrt{(e-1)(p-1)})$.

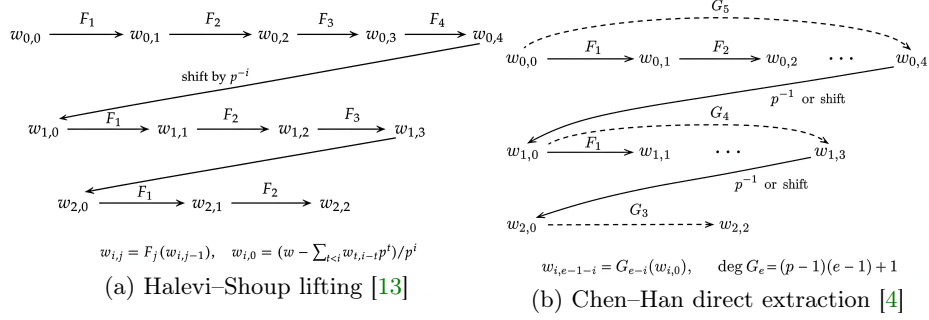


Fig. 3: Digit removal procedures for $e = 5$, $r = 2$. (a) Halevi-Shoup [13]: starting from $w_{0,0} = w$, each row iteratively applies the degree- p lifting polynomial F_k to zero out successive p -adic digits ($w_{i,j} = F_k(w_{i,j-1})$), then shifts down to the next row via $w_{i,0} = (w - \sum_{t < i} w_{t,i-t} p^t) / p^i$. (b) Chen-Han [4]: the degree- $(p-1)(e-i-1)+1$ digit-extraction polynomial G_{e-i} directly computes $w_{i,e-1-i} = G_{e-i}(w_{i,0})$ from the first element of each row, eliminating all lifting evaluations.

Geelen et al. null-polynomial lattice [7]. Geelen et al. observed that the set of null polynomials modulo p^e forms a lattice $\mathcal{O}_{p^e}^{(n)}$ with basis $\{p^{\max(e-\nu_p(i!), 0)} \cdot (X)_i\}_{i=0}^n$, where $(X)_i = \prod_{j=0}^{i-1} (X-j)$ is the falling factorial. By solving the closest vector problem (CVP) in this lattice, they find a representation $\tilde{G}_e = G_e + N$ (with $N \in \mathcal{O}_{p^e}^{(n)}$) that has 50% fewer nonzero coefficients and smaller coefficient norms. Combined with the multi-polynomial BSGS technique (evaluating m degree- n polynomials simultaneously at cost $2\sqrt{mn}$) and a compositional decomposition $G_e = g_{e,e'} \circ g_{e'}$, the overall complexity reduces to $O(\sqrt[4]{p} \sqrt[8]{e})$.

Ma et al. bounded-support construction [16]. As detailed in Section 2.4, Ma et al. exploit the bounded support of the digit-extraction input to reduce the polynomial degree from $(p-1)(e-1)+1$ to $|S_A| - 2 = O(B^2)$. For the standard parameter set ($p = 65537$, $h = 12$, $B = 17$), this gives degree 1223 with evaluation cost $\approx 2\sqrt{|S_A|} \approx 69$ non-scalar multiplications.

Baseline. After Ma et al.’s optimization, the polynomial degree $|S_A| - 2$ cannot be further reduced: P_A must interpolate $|S_A|$ distinct input-output pairs on S_A , so any equivalent polynomial has degree $\geq |S_A| - 2$. The evaluation cost $\approx 2\sqrt{|S_A|} \approx 69$ non-scalar multiplications under Paterson–Stockmeyer constitutes the baseline that our construction improves upon.

4 Character-Structured Digit-Extraction Polynomial

In this section, we present our character-structured digit-extraction polynomial. We describe the construction and its underlying character filter (§4.1), establish tightness and global optimality via an exhaustive search (§4.2), develop the specialized evaluation algorithm (§4.3).

4.1 Our novel Construction of Digit-Extraction Polynomial

We reduce the evaluation cost of the digit-extraction polynomial by exploiting algebraic structure in the quotient space $V = \mathbb{F}_p[X]/(G_{S_A}(X))$. Specifically, a carefully chosen auxiliary radix A induces a cyclic symmetry on V that constrains the monomial support of P_A to a single character eigenspace, enabling a variable substitution that reduces the effective degree by a factor of 4.

Motivation: beyond degree reduction. As established in Section 3, Ma et al.’s bounded-support framework achieves the information-theoretic optimal degree $|S_A| - 2$; no further degree reduction is possible. However, the Paterson–Stockmeyer cost $\approx 2\sqrt{d}$ depends only on degree, not on internal monomial structure. Our goal is to reduce the *effective* degree seen by the evaluator without modifying the polynomial degree itself.

Key observation: the auxiliary radix controls algebraic structure. In Ma et al.’s framework, the auxiliary radix A is treated as a free parameter subject only to B -injectivity; any valid A yields a polynomial of the same degree with the same evaluation cost. In contrast, we observe that the choice of A determines the algebraic structure of $V = \mathbb{F}_p[X]/(G_{S_A}(X))$.

When A satisfies $A^2 \equiv -1 \pmod{p}$, multiplication by the scalar A acts as a quarter-turn on the coordinate grid:

$$A(\eta A + \lambda) = \lambda A + (-\eta), \quad \text{inducing} \quad (\eta, \lambda) \mapsto (\lambda, -\eta),$$

which is a rotation of order 4 preserving $[-B, B]^2$. Consequently, $A \cdot S_A = S_A$, and this symmetry lifts to a cyclic group action $\langle T_A \rangle \cong \mathbb{Z}/4\mathbb{Z}$ on V via $T_A : f(X) \mapsto f(AX)$ (Definition 3). By the isotypic decomposition (Theorem 1), V decomposes into four eigenspaces $V = V_0 \oplus V_1 \oplus V_2 \oplus V_3$ with $X^k \in V_{k \bmod 4}$. We prove that P_A is forced into V_3 :

Theorem 2 (Order-four character filter). *Let $p \equiv 1 \pmod{4}$, let $A \in \mathbb{F}_p^\times$ satisfy $A^2 \equiv -1 \pmod{p}$, and let A be B -injective. Then the digit-extraction polynomial P_A of degree $< |S_A|$ satisfies*

$$P_A(AX) + A \cdot P_A(X) = AX, \tag{4}$$

and admits the representation

$$P_A(X) = \frac{1}{2}X + \sum_{j \geq 0} c_j X^{4j+3}.$$

That is, apart from the linear term X , only monomials with exponent $k \equiv 3 \pmod{4}$ can appear. The total number of nonzero terms is $(|S_A| - 1)/4 + 1$.

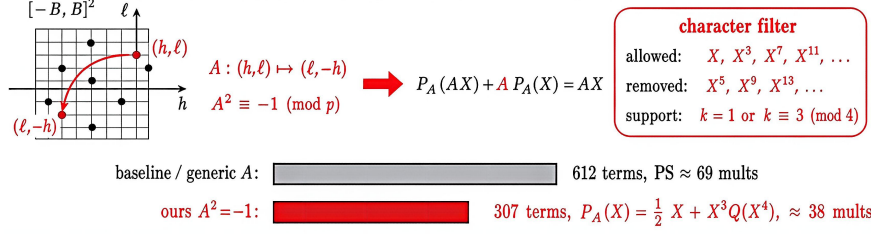


Fig. 4: Our character filter construction for $p = 65537$, $A = 256$, $B = 17$. The quarter-turn $A^2 \equiv -1 \pmod{p}$ induces $P_A(AX) + AP_A(X) = AX$, restricting monomials to $\{k = 1\} \cup \{k \equiv 3 \pmod{4}\}$ and yielding $P_A = \frac{1}{2}X + X^3Q(X^4)$.

The complete proof is given in Appendix B. We illustrate the theorem via a concrete derivation on the standard parameter set.

Concrete example: $p = 65537$, $A = 256$, $B = 17$. Since $p = 2^{16} + 1$ is a Fermat prime, $A = 256 = 2^8$ satisfies $256^2 = 65536 \equiv -1 \pmod{65537}$, so A is an order-4 element of $\mathbb{F}_p^\times$. The support has size $|S_A| = 35^2 = 1225$ and $\deg P_A = 1223$.

Step 1: The quarter-turn. For any $x = 256\eta + \lambda \in S_A$, multiplication by A gives

$$Ax = 256(256\eta + \lambda) = 256^2\eta + 256\lambda = -\eta + 256\lambda = \lambda A + (-\eta).$$

Ax corresponds to the pair $(\lambda, -\eta)$, and the digit extraction gives $P_A(Ax) = -\eta$.

Step 2: Deriving the functional equation. On the other hand, $P_A(x) = \lambda$ and $A \cdot P_A(x) = A\lambda$. Adding: $P_A(Ax) + A \cdot P_A(x) = -\eta + A\lambda = \lambda A - \eta = Ax$.

Since this holds for all 1225 points of S_A and both sides have degree < 1225 , we obtain the polynomial identity $P_A(AX) + A \cdot P_A(X) = AX$.

Step 3: Monomial constraint. Write $P_A(X) = \sum_k c_k X^k$. Substituting into the functional equation: $\sum_k c_k (A^k + A) X^k = AX$. For $k = 1$: $c_1(A + A) = A$, giving $c_1 = 1/2$. For $k \neq 1$: if $c_k \neq 0$, then $A^k + A = 0$, i.e., $A^k = -A = A^3$ (since $A^2 = -1$). This forces $k \equiv 3 \pmod{4}$. The nonzero monomials are exactly

$$\{X\} \cup \{X^3, X^7, X^{11}, X^{15}, \dots, X^{1223}\}$$

Term count: $1 + (1223 - 3)/4 + 1 = 307$ (vs. 612 for a generic odd polynomial of the same degree, the $(|S_A| - 1)/2$ bound of Proposition 1, in Ma et al [16]).

Step 4: Factored evaluation form. Since all nonzero terms beyond X have exponent $4j + 3$, we can extract the common factor X^3 and perform the substitution $Y = X^4$: $P_A(X) = \frac{1}{2}X + X^3 \cdot Q(X^4)$, $Q(Y) = \sum_{j=0}^{305} q_j Y^j$ with $q_j = c_{4j+3}$, $\deg Q = 305$. This factorization reduces the evaluation to: precompute $X^2, X^3, Y = X^4$ (3 mults), apply PS to degree-305 polynomial $Q(Y)$ (35 mults), and assemble (1 mult). Total: 38 vs. baseline 69 (−44.9%). Figure 4 summarizes the comparison.

Construction algorithm. Based on Theorem 2, the construction is deterministic for any (p, B) with $p \equiv 1 \pmod{4}$. Algorithm 1 summarizes the procedure.

First, the algorithm computes the unique $A \in \mathbb{F}_p^\times$ satisfying $A^2 \equiv -1 \pmod{p}$ via exponentiation of a primitive root. Subsequently, it verifies B -injectivity; if the condition $|A| \cdot B + B < p/2$ fails, A is replaced by $p - A$, which preserves the order-four property since $(p - A)^2 \equiv A^2 \equiv -1 \pmod{p}$ (for order-four radices the separation bound $2B < |A|$ of Lemma 1 is automatic once $(2B + 1)^2 < p$, since $p \mid A^2 + 1$ with $|A| \leq 2B$ would force $A^2 + 1 < p$). Next, the bounded support S_A is constructed and P_A is obtained via Lagrange interpolation over $\mathbb{F}_p$. As a correctness check, the functional equation $P_A(AX) + A \cdot P_A(X) \equiv AX$ is verified on the quotient space. Finally, the coefficients c_{4j+3} corresponding to the V_3 -eigenspace are extracted to form the reduced polynomial $Q(Y)$.

Algorithm 1 Character-Structured Digit-Extraction Polynomial

Require: Prime $p \equiv 1 \pmod{4}$, bound B
Ensure: Reduced polynomial $Q(Y) \in \mathbb{F}_p[Y]$ of degree $d' = ((2B + 1)^2 - 5)/4$
1: $A \leftarrow g^{(p-1)/4} \pmod{p}$ for a generator g of $\mathbb{F}_p^\times$
2: **if** $|A| \cdot B + B \geq p/2$ **then**
3: $A \leftarrow p - A$
4: **end if** ▷ ensure B -injectivity
5: $S_A \leftarrow \{\eta A + \lambda \pmod{p} : |\eta|, |\lambda| \leq B\}$
6: $P_A \leftarrow \text{LagrangeInterp}(S_A, x \mapsto \lambda)$ ▷ degree $< |S_A|$
7: **assert** $P_A(AX) + A \cdot P_A(X) \equiv AX \pmod{G_{S_A}(X)}$
8: **for** $j = 0, \dots, d'$ **do**
9: $q_j \leftarrow c_{4j+3}$ ▷ extract V_3 -eigenspace coefficients
10: **end for**
11: **return** $Q(Y) = \sum_{j=0}^{d'} q_j Y^j$

The algorithm runs in $O(|S_A|^2)$ field operations (dominated by Step 4) and is executed once during offline setup. The output $Q(Y)$ is stored and reused for all subsequent bootstrapping invocations.

The following theorem collects the exact conditions on (p, A, B, m, h) under which the entire construction applies, and makes precise in what sense it is independent of the ring.

Theorem 3 (Parameter admissibility). *Let h be the sparse-key Hamming weight and m the cyclotomic index, which determine the noise bound $B = \lceil I_{\text{bound}} \rceil$ with $I_{\text{bound}} = O(\sqrt{\varphi(m)}/m \cdot h)$ as in Section 6. The order-four digit extraction applies to a plaintext prime p if and only if*

- (i) $p \equiv 1 \pmod{4}$ (so that $A^2 \equiv -1 \pmod{p}$ has a solution);
- (ii) $(2B + 1)^2 < p$ (equivalently $|S_A| < p$: the bounded-support overflow part fits within one digit, the pipeline requirement $I_{\text{range}}^2 < p$ of [16] with $I_{\text{range}} = 2B + 1$);
- (iii) some A with $A^2 \equiv -1 \pmod{p}$ is B -injective (Definition 1), certified in practice via the bound $|A| \cdot B + B < p/2$ of Lemma 1.

Under (i)–(iii), S_A has full cardinality $(2B+1)^2$, the canonical polynomial P_A exists, satisfies the order-four functional equation of Theorem 2, and factors as $P_A = \frac{1}{2}X + X^3Q(X^4)$ with $\deg Q = (|S_A| - 5)/4$. Moreover, the construction depends on the ring only through the scalar $B = B(h, m)$: for a fixed B , the polynomial $P_A \in \mathbb{F}_p[X]$, its functional equation, and the factored evaluation algorithm are literally identical for every cyclotomic index m —power-of-two or general—since all objects live in $\mathbb{F}_p[X]$ and the homomorphic evaluation uses only slot-wise ring-generic operations.

The proof is deferred to Appendix E.

Remark 1 (Practically admissible primes). The bound certificate of Lemma 1 favors a *small* order-four radix, available precisely for primes dividing $a^2 + 1$ with a small cofactor; in particular every generalized Fermat prime $p = A^2 + 1$ above the overflow cutoff $p > (2B+1)^2$ ($= 1225$ at $h = 12$) is admissible. Table 9 lists recommended instantiations with bootstrappable cyclotomic indices, and Tables 6 and 5 confirm the ring-independence empirically across nineteen encrypted parameter sets. Further discussion (failure examples, the coincidence of conditions (ii) and (iii) for generalized Fermat primes, and the dependence of the cutoff on h) is given in Appendix C.

4.2 Tightness Bound and Exhaustive Optimality Search

Tightness on the fixed support S_{256} . We establish two results: (1) for our choice $A^2 \equiv -1$, the term count 307 is the exact eigenspace dimension and cannot be improved by any polynomial manipulation on the same support; (2) no alternative choice of A achieves fewer terms.

Theorem 4 (Tight term count). *Assume the standing hypotheses $p \equiv 1 \pmod{4}$, $A^2 \equiv -1 \pmod{p}$, and A B -injective. Then the exactness $|T| = \dim V_3 + 1 = (|S_A| - 1)/4 + 1$ of Theorem 2 is unconditional: it is an upper bound on the number of nonzero terms that holds for all such (p, A, B) . The stronger statement that all $(|S_A| - 1)/4$ monomials in V_3 actually carry nonzero coefficients (so that the bound is attained with equality) is a computationally verified property for the parameters we consider, not a claim over an unspecified family.*

The proof is given in Appendix F. We confirm computationally: for $p = 65537$, $A = 256$, $B = 17$, all 306 coefficients c_k with $k \equiv 3 \pmod{4}$ are nonzero, giving exactly $|T| = 307$.

Corollary 1 (Lower bound on a fixed support). *Fix a B -injective A with $A^2 \equiv -1 \pmod{p}$. Every polynomial that agrees with P_A on S_A and has degree $< |S_A|$ is P_A itself, so on this support no rewriting of the coefficients can bring the nonzero term count below $\dim V_3 + 1 = (|S_A| - 1)/4 + 1$. The statement is about a fixed (A, S_A) only: techniques that change the support, such as null-polynomial reduction, are not covered by it.*

Global optimality: no better A exists. The above establishes optimality for fixed $A = 256$. A separate question is whether a different auxiliary radix—possibly of higher multiplicative order $r > 4$ —could yield even fewer terms. An order- r character filter would require, in addition to B -injectivity, the *support stability* $A \cdot S_A = S_A$ (Lemma 3): for $A^2 \equiv -1 \pmod{p}$ this is supplied by the quarter-turn $(\eta, \lambda) \mapsto (\lambda, -\eta)$, which is a symmetry of the square grid $[-B, B]^2$, whereas no analogous grid symmetry is available for higher orders — e.g. for the order-8 element $A = 16$, multiplication maps $16\eta + \lambda$ to $256\eta + 16\lambda$, whose high digit 16η leaves $[-B, B]$, so the grid is not preserved. Rather than turning this observation into a general infeasibility claim, we settle the question over the relevant search space by an exhaustive computational verification (Algorithm 2) over all B -injective auxiliaries in $[2, 1000]$. We emphasize that this is an exhaustive check over B -injective auxiliaries for the parameters considered, not a general infeasibility claim for higher-order filters.

Algorithm 2 Exhaustive Optimality Verification

Require: Prime p , bound B , search range $[A_{\min}, A_{\max}]$
Ensure: Optimal A^* minimizing nonzero term count of P_A

- 1: $\text{best} \leftarrow \infty$; $A^* \leftarrow \text{nil}$
- 2: **for** $A = A_{\min}, \dots, A_{\max}$ **do**
- 3: $S_A \leftarrow \{\eta A + \lambda \bmod p : |\eta|, |\lambda| \leq B\}$
- 4: **if** $|S_A| \neq (2B + 1)^2$ **then continue** $\triangleright A$ not B -injective
- 5: **end if**
- 6: $P_A \leftarrow \text{LagrangeInterp}(S_A, x \mapsto \lambda)$
- 7: $\text{count} \leftarrow |\{k : c_k \neq 0\}|$
- 8: **if** $\text{count} < \text{best}$ **then** $\text{best} \leftarrow \text{count}$; $A^* \leftarrow A$
- 9: **end if**
- 10: Record $\text{ord}(A)$ and monomial pattern of P_A
- 11: **end for**
- 12: **return** A^* , best

Remark 2 (Global optimality of $A = 256$). For $p = 65537$, $B = 17$, Algorithm 2 with $A \in [2, 1000]$ confirms: only $A \in \{256, 65281\}$ (satisfying $A^2 \equiv -1 \pmod{p}$) achieve 307 nonzero terms; all other B -injective A give ≥ 612 terms; and order-8 elements (e.g., $A = 16$) fail B -injectivity. Therefore $A = 256$ achieves the global minimum term count among all possible auxiliaries.

4.3 Integration into BGV Thin Bootstrapping

We present the complete BGV thin bootstrapping procedure incorporating our character-structured polynomial $Q(Y)$ from Algorithm 1. Algorithm 3 details the full pipeline, with our optimized digit-extraction stage in Step 4.

Comparison with standard pipeline. Steps 1–3 are identical to the standard thin bootstrapping procedure of Chen and Han [4] (Section 2.3). The sole

Algorithm 3 BGV Bootstrapping with Character-Filtered Digit Extraction

Require: Low-capacity ciphertext $\text{ct} = \text{Enc}_s([m_0, \dots, m_{\ell-1}]; p^r)$; precomputed $Q(Y)$ from Algorithm 1

Ensure: Fresh ciphertext $\text{ct}' = \text{Enc}_s([m_0, \dots, m_{\ell-1}]; p^r)$ with restored capacity

```

// Step 1: SlotToCoeff (homomorphic linear transform)
1: Apply the encoding linear map  $M_{\text{S2C}}$  via BSGS:  $\text{ct} \leftarrow \sum_{i=0}^{D-1} \text{diag}_i \cdot \sigma^i(\text{ct})$ 
2:    $\triangleright$  Moves slot values  $[m_0, \dots, m_{\ell-1}] \in \mathbb{Z}_{p^r}^\ell$  into powerful-basis coefficients of
    $m(X) \in R_{p^r}$ 

// Step 2: Decryption formula simplification and inner product
3: Modulus-switch  $\text{ct}$  to level  $q_0$ :  $(b, a) \leftarrow \text{ModSwitch}(\text{ct}, q_0)$ 
4: Key-switch to bootstrapping key  $s$ :  $(b, a) \leftarrow \text{KeySwitch}((b, a), \text{KSK}_{s' \rightarrow s})$ 
5: Raise modulus from  $q_0$  to  $Q$ :  $(b, a) \leftarrow \text{ModRaise}((b, a), Q)$ 
6: Compute inner product:  $\text{ct} \leftarrow b + a \cdot \text{Enc}_{s'}(s; p^e)$ 
7:    $\triangleright$  Now  $\text{ct}$  encrypts  $p^{e-r}m + \epsilon$  under plaintext modulus  $p^e$  with high capacity

// Step 3: CoeffToSlot (inverse homomorphic linear transform)
8: Apply the decoding linear map  $M_{\text{C2S}}$  via BSGS:  $\text{ct} \leftarrow \sum_{i=0}^{D-1} \text{diag}'_i \cdot \sigma^i(\text{ct})$ 
9:    $\triangleright$  Each slot now contains  $p^{e-r}m_i + \epsilon_i \in \mathbb{Z}_{p^e}$  with  $|\epsilon_i| \leq B$ 

// Step 4: Character-filtered digit extraction (our contribution)
10:    $\triangleright$  Goal: extract  $m_i$  from  $p^{e-r}m_i + \epsilon_i$  by evaluating  $P_A$ 
11:  $\text{ct}_2 \leftarrow \text{Mul}(\text{ct}, \text{ct})$   $\triangleright x^2, 1$  mult
12:  $\text{ct}_3 \leftarrow \text{Mul}(\text{ct}_2, \text{ct})$   $\triangleright x^3, 1$  mult
13:  $\text{ct}_Y \leftarrow \text{Mul}(\text{ct}_2, \text{ct}_2)$   $\triangleright Y = x^4, 1$  mult
14:  $\text{ct}_Q \leftarrow \text{PatersonStockmeyer}(\text{ct}_Y, Q)$   $\triangleright$  evaluate  $Q(Y)$ ,  $k + 2m - 2$  mults
15:  $\text{ct}' \leftarrow \text{Mul}(\text{ct}_3, \text{ct}_Q) + \frac{1}{2} \cdot \text{ct}$   $\triangleright$  assemble  $x^3Q(x^4) + \frac{1}{2}x$ , 1 mult
16:    $\triangleright$  Now  $\text{ct}'$  encrypts  $[m_0, \dots, m_{\ell-1}] \in \mathbb{Z}_{p^r}^\ell$  with refreshed capacity

17: return  $\text{ct}'$ 

```

modification is Step 4: instead of calling generic `polyEval` on the degree-1223 polynomial P_A (which internally uses PS with $k = m = 35$ and costs ≈ 69 non-scalar multiplications), we exploit the factored form $P_A(X) = \frac{1}{2}X + X^3Q(X^4)$ to first precompute $x^2, x^3, Y = x^4$ (3 mults), then evaluate the degree-305 polynomial $Q(Y)$ via PS (35 mults), and finally assemble the result (1 mult). The total digit-extraction cost is 38 non-scalar multiplications.

Detailed evaluation procedure. Algorithm 4 expands the PS evaluation of $Q(Y)$ in Step 4 into its five constituent phases.

Correctness of Bootstrapping Algorithm 3. Correctness of the bootstrapping procedure requires that the output ciphertext ct' encrypts the original message $[m_0, \dots, m_{\ell-1}] \in \mathbb{Z}_{p^r}^\ell$ with refreshed capacity. We verify this stage by stage.

Steps 1–3 inherit the correctness guarantees of the standard pipeline: after Step 3 each slot contains $w_i = p^{e-r}m_i + \epsilon_i \in \mathbb{Z}_{p^e}$ with $|\epsilon_i| \leq B = 17$. The correctness of Step 4 rests on two properties:

Algorithm 4 Character-Filtered Polynomial Evaluation (**CharFilterEval**)

Require: Ciphertext $\text{ct} \in \text{RLWE}(x)$; polynomial $Q(Y) = \sum_{j=0}^{d'-1} q_j Y^j$; $k = \lceil \sqrt{d'} \rceil$,
 $m = \lceil d'/k \rceil$

Ensure: $\text{ct}_{\text{out}} \in \text{RLWE}(P_A(x))$ where $P_A(x) = \frac{1}{2}x + x^3Q(x^4)$

// **Phase 1: Precomputation (3 non-scalar mults)**

1: $\text{ct}_2 \leftarrow \text{Mul}(\text{ct}, \text{ct})$; $\text{ct}_3 \leftarrow \text{Mul}(\text{ct}_2, \text{ct})$; $\text{ct}_Y \leftarrow \text{Mul}(\text{ct}_2, \text{ct}_2)$

// **Phase 2: Baby-step powers of Y ($k-1$ mults)**

2: **for** $i = 2, \dots, k-1$ **do**

3: $\text{ct}_{Y^i} \leftarrow \text{Mul}(\text{ct}_{Y^{i-1}}, \text{ct}_Y)$

4: **end for**

// **Phase 3: Giant-step powers ($m-1$ mults)**

5: $\text{ct}_Z \leftarrow \text{ct}_{Y^k}$

6: **for** $j = 2, \dots, m-1$ **do**

7: $\text{ct}_{Z^j} \leftarrow \text{Mul}(\text{ct}_{Z^{j-1}}, \text{ct}_Z)$

8: **end for**

// **Phase 4: PS accumulation on $Q(Y)$ (m mults)**

9: $\text{ct}_Q \leftarrow \text{RLWE}(0)$

10: **for** $j = 0, \dots, m-1$ **do**

11: $y_j \leftarrow \sum_{i=0}^{k-1} q_{jk+i} \cdot \text{ct}_{Y^i}$ ▷ plaintext-ciphertext only

12: $\text{ct}_Q \leftarrow \text{ct}_Q + \text{Mul}(y_j, \text{ct}_{Z^j})$

13: **end for**

// **Phase 5: Assembly (1 mult)**

14: $\text{ct}_{\text{out}} \leftarrow \text{Mul}(\text{ct}_3, \text{ct}_Q) + \frac{1}{2} \cdot \text{ct}$

15: **return** ct_{out}

- (i) The interpolation condition $P_A(\eta A + \lambda) = \lambda$ for all $(\eta, \lambda) \in [-B, B]^2$ (Definition 2). Since $w_i = \eta A + \lambda$ with $\eta = \lfloor w_i/A \rfloor$ and $\lambda = w_i - \eta A$ satisfying $|\eta|, |\lambda| \leq B$, evaluating $P_A(w_i)$ extracts the low digit $\lambda = m_i \bmod p^r$.
- (ii) The polynomial identity $P_A(X) = \frac{1}{2}X + X^3Q(X^4)$ (Theorem 2), which makes the factored evaluation of Step 4 compute the same function as a direct evaluation of P_A .

Both properties are verified during offline setup: property (i) by construction via Lagrange interpolation, and property (ii) by the assertion in Algorithm 1. Consequently, after Step 4, each slot contains $P_A(w_i) = m_i \in \mathbb{Z}_{p^r}$ as required.

Noise analysis. We analyze the noise growth through Algorithm 3 to verify that decryption remains correct after bootstrapping.

In Steps 1–3, the noise budget is governed by the standard **HElib** modulus chain: the inner product (Step 2) produces a high-capacity ciphertext with ~ 12 available multiplicative levels for digit extraction.

In Step 4, each non-scalar multiplication followed by relinearization introduces noise proportional to $\sqrt{N} \cdot \beta_{\text{ks}}^2 \cdot E_{\text{ks}}$, where $N = \varphi(m) = 50112$ is the ring degree and β_{ks} is the key-switching decomposition base. The factored evaluation performs 38 non-scalar multiplications (3 for precomputation, $k-1 = 17$ for baby-step, $m-1 = 16$ for giant-step, $m = 17$ for accumulation, and 1 for assembly). This is identical in count to standard PS on a degree-305 polynomial.

The variable substitution $Y = X^4$ does not introduce additional noise since it merely reinterprets existing ciphertext powers.

The multiplicative depth of Step 4 is $\lceil \log_2 k \rceil + 4 = \lceil \log_2 18 \rceil + 4 = 9$, compared to baseline depth $\lceil \log_2 35 \rceil + 1 = 7$. The additional 2 levels are consumed by the precomputation chain $x \rightarrow x^2 \rightarrow x^4$. The HELib modulus chain provides ~ 12 levels for digit extraction, which accommodates our depth-9 evaluation with 3 levels of margin. A detailed noise bound is given in Appendix G.

5 Structured Linear Transforms for Non-Power-of-Two Cyclotomics

For non-power-of-two cyclotomics, the homomorphic linear transforms (SlotToCoeff/CoeffToSlot) rely on generic BSGS with $O(\sqrt{D})$ automorphisms per dimension- D block. Peikert and Pepin [22] established a general Galois-theoretic framework proving that $O(\log D)$ is achievable for arbitrary abelian number rings. However, their work provides no concrete instantiation, no explicit twiddle factor computation, and no validation on practical parameter sets.

In this section, we provide the first concrete instantiation of the Peikert–Pepin framework for the standard HELib non-power-of-two parameter set ($m = 50731 = 97 \times 523$). Our contribution is the identification of the number-theoretic condition $D \mid p^d - 1$ as the concrete enabler, the explicit mixed-radix butterfly algorithm with precomputed slot-field twiddle factors, and the plaintext-level validation of correctness and operation count. In addition to the plaintext-level validation, we provide a first *encrypted* prototype of the butterfly transform on the full ring $m = 50731$: the complete dimension-96 CoeffToSlot chain runs on ciphertexts and decrypts correctly on all 2784 slots, consuming 194 bits of capacity over its 41 automorphisms (17 hoisting classes) with no overflow even at a 600-bit budget. Its wall-clock advantage over BSGS emerges only at larger block dimensions ($D \approx 576$); the per-dimension measurements are in the artifact.

5.1 Number-Theoretic Condition and Butterfly Factorization

For cyclotomic index $m = M_1 \times M_2$ (coprime), the EvalMap decomposes into independent one-dimensional linear maps along the hypercube dimensions of $(\mathbb{Z}/m\mathbb{Z})^\times / \langle p \rangle$ (the orbit sizes of HELib’s chosen generators). For $m = 50731 = 97 \times 523$ the dimensions are $D_1 = 96$ and $D_2 = 29$. The $D = 96$ block dominates the cost.

Lemma 2 (Root availability). *Let $d = \text{ord}_m(p)$ be the slot extension degree. If $D \mid p^d - 1$, then the multiplicative group $\mathbb{F}_{p^d}^\times$ contains a cyclic subgroup of order D , so all D -th roots of unity exist in the slot field $\mathbb{F}_{p^d}$. Consequently, the Step2Matrix admits diagonalization by the D -point DFT over $\mathbb{F}_{p^d}$.*

For our parameters: $D = 96$, $d = 6$. Since $32 \mid p - 1 = 2^{16}$ and $3 \mid p^2 - 1$, we have $96 \mid p^2 - 1$, so $96 \mid p^6 - 1$. All 96-th roots of unity exist in $\mathbb{F}_{p^2} \subset \mathbb{F}_{p^6}$. Here the Step2Matrix M_D denotes the $D \times D$ block of the Step-2 slot-wise

linear transform, with entries $(M_D)_{i,j} = \omega^{ij}$ for a primitive D -th root of unity ω (Appendix I).

Theorem 5 (Butterfly decomposition). *When $D \mid p^d - 1$ and $D = p_1^{a_1} \times p_2^{a_2} \times \cdots \times p_k^{a_k}$, the Step2Matrix factors as*

$$M_D = \Pi_{\text{out}} \cdot \left(\prod_{i=1}^k \prod_{j=1}^{a_i} B_{i,j} \right) \cdot \Pi_{\text{in}},$$

where $\Pi_{\text{in}}, \Pi_{\text{out}}$ collect input/output permutations (handled once at the chain edges), and each $B_{i,j}$ is a radix- p_i butterfly layer with at most $2p_i - 1$ nonzero cyclic diagonals, requiring $2(p_i - 1)$ automorphisms in general and $p_i - 1$ at the final stage of each prime. The total automorphism count per cascade is

$$\sum_{i=1}^k (2a_i - 1)(p_i - 1) = O(\log D),$$

compared to $2\lceil\sqrt{D}\rceil - 2 = O(\sqrt{D})$ for BSGS.

Proof (Proof sketch). The factorization proceeds in two stages.

Stage 1 (Good-Thomas). Since $D = D_1 \times D_2$ with $\gcd(D_1, D_2) = 1$ (here $D_1 = 32$, $D_2 = 3$), the CRT bijection $\mathbb{Z}/D\mathbb{Z} \cong \mathbb{Z}/D_1\mathbb{Z} \times \mathbb{Z}/D_2\mathbb{Z}$ gives

$$F_D = P_{\text{CRT}}^{-1} \cdot (F_{D_1} \otimes F_{D_2}) \cdot P_{\text{CRT}},$$

where P_{CRT} corresponds to a free slot rearrangement in HElib. No twiddle factors appear between the sub-DFTs.

Stage 2 (Cooley-Tukey). For $D_1 = 2^5$, the radix-2 factorization gives

$$F_{2^5} = \left(\prod_{j=1}^5 (I_{2^{j-1}} \otimes B_2 \otimes I_{2^{5-j}}) \cdot \text{diag}(\omega_{2^j}^k) \right) \cdot R_{32},$$

where $B_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ and R_{32} is the bit-reversal permutation, absorbed into Π_{in} at the chain edge. Each stage has nonzero cyclic diagonals at offsets $\{0, \pm s\}$, i.e. two automorphisms (one at the collapsing stride- $D/2$ stage), and F_3 costs two: $(2 \cdot 5 - 1) \cdot 1 + 1 \cdot 2 = 11$ per cascade; the full accounting is in Appendix H.

The complete proof is given in Appendix H, where the Rader reindexing brings the Step2Matrix of the $D = 96$ block into the DFT-type form assumed above.

Remark 3 (Twiddle field membership). The radix-2 twiddles are 32-nd roots of unity. Since $32 \mid p - 1 = 2^{16}$, these exist in $\mathbb{F}_p$ itself (scalars). The radix-3 twiddles are 3-rd roots of unity. Since $3 \mid p^2 - 1$ but $3 \nmid p - 1$, these exist in $\mathbb{F}_{p^2} \setminus \mathbb{F}_p$ (degree-1 polynomials in the slot ring).

5.2 Evaluation Algorithm and Complexity Comparison

We present the concrete evaluation procedure for the dominant $D = 96 = 2^5 \times 3$ block and analyze its cost relative to the standard BSGS method. Algorithm 5 implements the butterfly factorization of Theorem 5: five radix-2 layers with scalar twiddles in $\mathbb{F}_p$, followed by one radix-3 layer with twiddles in $\mathbb{F}_{p^2}$. **Cost**

Algorithm 5 Butterfly Linear Transform ($D = 96 = 2^5 \times 3$)

Require: Ciphertext ct ; precomputed twiddle constants $\{\omega_i\}$

Ensure: $\text{ct}_{\text{out}} = L(\text{ct})$

```

// 5 radix-2 butterfly layers (twiddles in  $\mathbb{F}_p$ )
1: for  $i = 1, \dots, 5$  do
2:   stride  $\leftarrow 2^{5-i}$ 
3:    $\text{ct}_{\pm} \leftarrow \sigma^{\pm \text{stride}}(\text{ct})$   $\triangleright$  2 automorphisms (1 at the stride-16 stage, where  $\pm \text{stride}$ 
      coincide)
4:    $\text{ct} \leftarrow d_i^{(0)} \cdot \text{ct} + d_i^{(+)} \cdot \text{ct}_+ + d_i^{(-)} \cdot \text{ct}_-$   $\triangleright$  one mult per nonzero diagonal
5: end for
// 1 radix-3 butterfly layer (twiddles in  $\mathbb{F}_{p^2}$ )
6:  $\text{ct}_1 \leftarrow \omega_6^{(1)} \cdot \sigma^{32}(\text{ct})$   $\triangleright$  1 automorphism
7:  $\text{ct}_2 \leftarrow \omega_6^{(2)} \cdot \sigma^{64}(\text{ct})$   $\triangleright$  1 automorphism
8:  $\text{ct}_{\text{out}} \leftarrow \text{ct} + \text{ct}_1 + \text{ct}_2$ 
9: return  $\text{ct}_{\text{out}}$ 

```

analysis. The cascade costs the 11 automorphisms counted in the proof of Theorem 5, each a full key-switch, together with one plaintext–ciphertext multiplication per nonzero diagonal. The standard BSGS method with $b = g = \lceil \sqrt{96} \rceil = 10$ needs $b + g - 2 = 18$ automorphisms on the same $D = 96$ block.

Layer merging via hoisting. In HElib, hoisting precomputes the gadget decomposition of a ciphertext once, then applies multiple automorphisms to the decomposed form without re-decomposing. Consecutive butterfly layers with compatible strides can share a single hoisting. In particular, the two rotations $\sigma^{\pm \text{stride}}$ of each stage act on the same input ciphertext and share one hoisting, giving 6 hoistings for the 11 automorphisms of a cascade (5 radix-2 stages + 1 radix-3 layer); in the measured Rader-wrapped $D = 96$ chain this yields 17 hoisting classes over 41 automorphisms.

Theorem 6 (Correctness). *The butterfly decomposition computes the same linear transform as the original Step2Matrix: $\prod_i B_i = M_D$ up to the free CRT permutations $\Pi_{\text{in}}, \Pi_{\text{out}}$.*

The proof, by explicit matrix factorization over $\mathbb{F}_{p^{18}}$, is given in Appendix I; we also verify in SageMath that the product of all butterfly-layer matrices reproduces the Step2Matrix entry by entry. The decomposition applies whenever $D \mid p^d - 1$, which holds at our parameters ($D = 96$, $p = 65537$, $d = 6$); when it fails, BSGS remains the method of choice.

6 Implementation and Evaluation

We implement the order-four evaluator and the butterfly transform inside HELib and evaluate them on ciphertexts. The implementation is a public artifact⁵ that also contains a Lean 4 + Mathlib formalization of the algebraic core (`lake build` completes with zero `sorry`) and the security scripts of Section 6.1.

All timings are single-threaded wall clock on one Intel Xeon E5-2680 v4 socket (14 cores, 2.40 GHz, 128 GB DDR4), GCC 11.4 with `-O2`, NTL 11.5.1 and GMP 6.2.1, reported as the mean of three trials. Every run is decrypted and compared slot by slot against the expected plaintext; a run whose check fails is not reported. In every table OFF is the bounded-support evaluator of Ma et al. [16] with generic Paterson–Stockmeyer evaluation of P_A and ON is the factored evaluation $P_A = \frac{1}{2}X + X^3Q(X^4)$, both run from the same binary on the same ciphertext, pipeline, modulus chain, key weights and auxiliary radix A .

6.1 Parameter Sets and Concrete Security

Every set below is verified per ring rather than taken from a library table. A bootstrappable HELib instance carries two LWE problems and, following [16, Sec. 4.3], the security λ^* of a set is the minimum of the two: the *main key* of weight $h' = 120$ against the full chain Q , and the *encapsulated bootstrapping key* of the sparse weight h that digit extraction requires, against the short modulus q_{boot} . On each instance λ^* is the smallest cost any attack in the Lattice Estimator [1] catalogue attains at its own optimized parameters; the binding attack is the meet-in-the-middle primal hybrid, up to 25.1 bits below uSVP.

Power-of-two rings are additionally evaluated under the coefficient-isometry attack of [19]. That attack needs $\mathbb{Z}_q[X]/(X^N + 1)$ with N a power of two, so it does not apply to the general rings $m = q_1q_2$ used here; it costs the state-of-the-art baseline of [17] 7.4 bits and the corresponding instances of this work 7.1 to 7.7, so the recommended power-of-two chains below are trimmed accordingly and apply to that work as well. Because a chain requested from HELib is not the chain it builds, λ^* is re-verified at the built chain. Tables 3 and 4 are the resulting operating points; the per-attack breakdown is in Appendix D, and the estimator transcripts are in the artifact.

6.2 Power-of-Two Cyclotomic Rings

At $p = 65537$ on $m = 2^{16}$, the recommended 80-bit point, digit extraction falls from 20.06 s to 10.81 s ($1.85\times$) while the three stages the method does not touch move by 0.99, 1.00 and 0.99. Whole decryption falls from 42.34 s to 33.25 s ($1.27\times$), and on the throughput basis of prior work the gain is $1.23\times$.

Table 5 repeats the measurement for ten admissible primes on the same ring. The speedup spans $1.84\times$ to $2.13\times$, mean $1.91\times$; all ten primes improve on the

⁵ <https://github.com/acpk-hash/order4-bgv-bootstrap>

Table 3: Strategy B, and the recommended power-of-two operating points. $h' = 120$, $h = 26$, $n = \varphi(m) = m/2$ throughout. Columns sq-root and estim. are the two rotation heuristics at the boundary chain Q_t , where the rotation-free optimizer gives 88.90, 111.80, 141.03 and the six-attack Estimator gives 88.0, 111.2, 139.9. The last three columns are at the chosen chain: main key by the Estimator, main key under [19], encapsulated key under [19]. λ^* is the smaller of the last two and is attained by the main key on every row.

target	m	Q_t	sq-root	estim.	chosen Q	λ_{est}	after	encap.
80	2^{16}	1200	80.04	80.99	1078	95.6	86.7	138.7
100	2^{17}	1815	100.18	101.40	1676	117.8	106.0	149.6
128	2^{17}	1326	128.18	129.41	1277	143.6	131.4	149.6

longer chains of a library’s bundled tables. The OFF column is nearly constant across two orders of magnitude of p , since bounded-support extraction depends on p only through B , which is the same in all ten rows.

6.3 General Cyclotomic Rings

Table 6 reports the nine recommended general-cyclotomic sets that have a ciphertext run. The 80-bit tier ($h = 12$) gives $1.72\times$ to $1.78\times$ and the 128-bit tier ($h = 26$) $1.77\times$ to $1.88\times$; the latter is systematically higher because a heavier encapsulated key raises B and so lengthens exactly the polynomial the contribution shortens.

Set 80-E ($p = 65537$, $m = 50731$) has no ciphertext run at its own chain; the figures quoted for it elsewhere (143.9s OFF against 81.4s ON, $1.77\times$) are a separate $m = 50731$ measurement at the same 2017-bit chain length. There are no timings at the recommended $m = 2^{17}$ chains. The 80-A row was taken on a loaded machine; an idle-machine repeat built a 1497.9-bit chain and measured 80.46 against 46.28, ratio $1.74\times$ against $1.78\times$, so absolute times should be compared within a table and never across tables.

Table 7 sweeps the encapsulated weight h on both ring families. At the plaintext level the reduction holds across primes: every $p \equiv 1 \pmod{4}$ with a B -injective order-four radix cuts the term count to $(|S_A| - 1)/4 + 1$, about 50% below the odd-polynomial count of the order-two construction, and 25% at $p = 17$ where $|S_A| = 9$ is too small for the eigenspaces to balance.

Two witnessed admissibility failures. Both non-trivial conditions of Theorem 3 are witnessed by concrete failures rather than merely asserted. For (i), $p \equiv 1 \pmod{4}$: at $p = 8191 \equiv 3 \pmod{4}$ no order-four radix exists, and the evaluator detects this, reports itself inapplicable and falls back to the baseline path, so no incorrect result is produced. For (ii), $(2B + 1)^2 < p$: at $p = 257$ with $h = 12$ the pipeline aborts with “ p too small to hold the overflow part”. Neither failure is silent and neither can be worked around at fixed h , which is why $p = 257$ supports the algebraic structure but appears in no ciphertext table at $h = 12$.

Table 4: Recommended general-cyclotomic sets, five distinct primes per level, with security re-verified at the chain **HElib** built. $h' = 120$ throughout; “main” and “encap.” are the two instances at the recommended $\log_2 Q$, and λ^* is the main-key minimum over attacks at the built chain. General rings are out of scope of [19] (Section 6.1), so all figures are Estimator minima. Set 80-E has no ciphertext run and hence no built chain.

set	(p, m)	$\varphi(m)$	h	$\log_2 Q$	main	encap.	slots	built	λ^*
80-A	(4513, 37637)	36960	12	1510	81.1	87.6	9240	1524.2	80.5
80-B	(14401, 43033)	41140	12	1680	81.3	88.7	3740	1642	82.75
80-C	(13457, 45193)	44100	12	1800	81.4	89.7	6300	1787	81.94
80-D	(2521, 50731)	50112	12	2040	81.9	91.7	2784	2026	82.44
80-E	(65537, 50731)	50112	12	2017	82.6	90.9	2784	n/a	n/a
128-A	(13457, 65047)	62776	26	1427	128.5	158.5	31388	1407	129.81
128-B	(2917, 53983)	52488	26	1184	128.6	153.4	13122	1153	131.0
128-C	(15377, 55459)	53640	26	1212	128.4	154.1	13410	1193	129.90
128-D	(8101, 57521)	56832	26	1287	128.5	155.7	7104	1268	129.84
128-E	(14401, 54359)	51480	26	1161	128.5	153.0	5148	1133	130.77

6.4 Comparison with Prior Work

Table 8 places our results among prior BGV bootstrapping implementations for $p = 65537$ on the same ring $m = 50731$ (2784 slots), against four baselines:

- **HElib native** [13,14]: Halevi–Shoup bootstrapping with Chen–Han digit extraction (lifting polynomial of degree p^{e-1}).⁶
- **Geelen et al.** [7]: null-polynomial lattice optimization with CVP-based sparse representations.⁷
- **Ma et al.** [16]: bounded-support construction at radix $A = 35$, reducing the degree from $O(\sqrt{p})$ to $O(B^2)$.⁸
- **Ma et al. (NTT)** [17]: homomorphic NTT for power-of-two cyclotomics, context only, not applicable to $m = 50731$.⁹

Two entries need a word. Geelen et al. [7] report no timing at $p = 65537$ and their code needs Magma for polynomial generation there, so their row is extrapolated from the complexity reduction they report and is not a measurement, marked *. And we claim no ratio against **HElib native**: it uses Chen–Han digit extraction and belongs to a different algorithmic generation, so a ratio against it would measure the intervening work on digit extraction rather than this contribution.

Galois-structure polynomial evaluation [20]. Okada et al. evaluate the digit-extraction polynomial through the norm map of the slot algebra in $O(\log d)$

⁶ <https://github.com/homenc/HElib>

⁷ https://github.com/KULeuven-COSIC/Bootstrapping_Polyfunctions

⁸ <https://github.com/msh086/BGV-Boot-for-Large-p>

⁹ <https://github.com/msh086/bgv-bootstrapping-with-homomorphic-NTT>

Table 5: Digit extraction on $m = 2^{16}$ at the recommended 80-bit operating point ($h' = 120$, $h = 26$, $B = 13$, $\deg P_A = 727$, `useCache` = 1). For reference the last column repeats the measurement for the same prime and ring on the longer 1314–1332-bit chains that a library’s bundled tables give.

p	A	slots built	$\log_2 Q$	OFF (s)	ON (s)	speedup at 1314–1332	
1297	36	8	1071	23.00	11.78	$1.95\times$	$1.63\times$
1601	40	32	1071	22.93	12.39	$1.85\times$	$1.69\times$
2521	71	4	1071	24.71	12.82	$1.93\times$	$1.62\times$
3137	56	32	1071	21.98	11.43	$1.92\times$	$1.64\times$
4513	95	16	1066	20.39	11.11	$1.84\times$	$1.65\times$
7057	84	8	1066	21.13	11.25	$1.88\times$	$1.60\times$
13457	116	8	1066	21.35	10.94	$1.95\times$	$1.74\times$
14401	120	32	1066	24.39	11.48	$2.13\times$	$1.67\times$
15377	124	8	1066	21.66	11.74	$1.85\times$	$1.71\times$
65537	256	32768	1066	20.06	10.81	$1.85\times$	$1.59\times$

Table 6: Digit extraction and whole decryption on the nine recommended general cyclotomic sets of Section 6.1 that have a ciphertext run. OFF and ON are radix matched; λ^* is re-verified at the chain `HElib` actually built.

set	p	built	$\log_2 Q$	λ^*	OFF (s)	ON (s)	speedup total	OFF (s) total	ON (s) total
80-A	4513	1524.2	80.5	89.31	50.12	$1.78\times$	139.79	100.03	
80-B	14401	1642.0	82.75	100.05	57.40	$1.74\times$	141.63	97.51	
80-C	13457	1786.8	81.94	113.73	66.23	$1.72\times$	161.88	114.14	
80-D	2521	2026.2	82.44	148.93	86.03	$1.73\times$	198.97	136.64	
128-A	13457	1406.9	129.81	183.15	100.97	$1.81\times$	312.98	231.72	
128-B	2917	1153.0	131.0	127.23	70.68	$1.80\times$	229.42	170.23	
128-C	15377	1193.1	129.90	135.00	76.32	$1.77\times$	195.27	139.40	
128-D	8101	1267.5	129.84	152.39	81.07	$1.88\times$	207.47	134.78	
128-E	14401	1132.6	130.77	129.77	70.85	$1.83\times$	188.43	129.94	

key switches, which requires slot rank $d \geq \deg f$. Their method modifies the *evaluator* and ours the *polynomial* ($P_A \rightarrow Q$, $\deg Q = (\deg P_A - 3)/4$), so the two compose rather than compete. Applicability on $m = 2^{16}$ is settled by the slot rank: NTT-friendly primes collapse it ($d = 1$ at $p = 65537$), and of our ten primes only $p = 1297$ ($d = 4096 \geq 1296$) and $p = 2521$ ($d = 8192 \geq 2520$) meet the prerequisite $d \geq \deg f \approx p - 1$; in the thin regime and on our general-ring sets ($d \leq 18$) the method does not apply at all. We reproduced their open-source SEAL implementation on our hardware ($p = 257$, $N = 2^{15}$, 128 slots of rank $256 = \deg f$: 17 key switches, matching their paper). Composing the two divides the required rank by four ($2048 \rightarrow 512$ at $p = 65537$, $256 \rightarrow 64$ at $p = 257$) at an unchanged key-switch count: measured in their $p = 257$ context with one uniform BSGS evaluator, the factored path takes 9.0 s and 22 key switches against 17.2 s and 41 for P_A , with bit-identical decryptions, and on the half-size ring $m = 2^{15}$, where the slot rank 128 covers $\deg Q = 56$ but neither their $\deg f = 257$ nor

Table 7: Degree scaling of the order-four speedup at $p = 65537$, radix matched at $A = 256$ throughout.

ring	h	B	$\deg P_A$	OFF (s)	ON (s)	speedup
gen., $m=50731$	12	17	1223	145.2	82.8	$1.75\times$
gen., $m=50731$	18	20	1679	180.0	101.0	$1.78\times$
gen., $m=50731$	26	24	2399	218.7	118.3	$1.85\times$
po2, $m=2^{16}$	12	9	359	15.4	9.4	$1.63\times$
po2, $m=2^{16}$	18	11	527	18.7	11.3	$1.65\times$
po2, $m=2^{16}$	26	13	727	22.4	13.4	$1.67\times$
po2, $m=2^{16}$	40	16	1087	28.4	16.9	$1.69\times$

Table 8: BGV thin bootstrapping for $p = 65537$ on $m = 50731$. Timings in seconds on comparable hardware, single-threaded except where noted; speedup is relative to Ma et al. [16]. These rows are the same-ring comparison at that ring’s own chain, not the recommended-parameter runs of Sections 6.2 and 6.3.

Method	Ref.	Linear (s)	Extract (s)	Total (s)	Speedup	Thr. (bps)
HElib native	[13]	50.9	> 43200	> 43200	—	—
Geelen et al.	[7]	~ 50	$\sim 500^*$	$\sim 550^*$	$0.39\times$	~ 0.81
Ma et al.	[16]	49.70	160.78	213.10	$1.00\times$	2.09
This work (order-4)		47.89	80.77	131.18	$1.62\times$	3.40
This work (combined)		29.34	86.38	118.49	$1.80\times$	3.76

the unfactored $\deg P_A = 223$, it takes 1.8s at the same 22 key switches. For a head-to-head inside their band we lowered the key weight to $h = 7$, so that $(2B+1)^2 = 225 < 257$, and pushed the bounded-support route down to $p = 257$: the Ma et al. baseline takes 12.6s and our filter 8.0s ($1.58\times$), their evaluator 5.5s and 17 key switches. Theirs is therefore the faster method inside its own high-rank band ($p \approx 127$ to 257 with $d \geq \deg f$), which does not overlap our applicable range $p > 1225$ at $h = 12$; this $p = 257$ configuration is a comparison harness, not a proposed operating point.

7 Conclusion

We presented two algebraic optimizations for large-prime BGV bootstrapping: an order-four character filter that exploits $A^2 \equiv -1 \pmod{p}$ to force the structured factorization $P_A(X) = \frac{1}{2}X + X^3Q(X^4)$, reducing the effective PS degree by $4\times$ with provable tightness, and the first concrete $O(\log D)$ butterfly instantiation for non-power-of-two cyclotomics, validated on the standard HElib parameter set. On the recommended power-of-two configuration ($p = 65537$, $m = 2^{16}$, 32768 slots), our single-threaded implementation cuts digit-extraction cost by $1.85\times$

and whole decryption by $1.27\times$; measured across nineteen encrypted sets spanning both ring families and the 80- and 128-bit levels, digit extraction speeds up by $1.72\times$ to $2.13\times$ while the stages the method does not touch move by at most one per cent. Both optimizations are drop-in replacements that compose with subring encapsulation, homomorphic NTT, and other orthogonal techniques.

Future directions include subring integration for cascading degree reduction, extended key infrastructure for asymmetric BSGS, and higher-order character groups where $\text{ord}(A) > 4$ is feasible.

Acknowledgments. We thank Zhedong Wang and Han Wang for their collaborative contributions to this work during the review period, and the anonymous reviewers of Asiacrypt 2026, whose comments materially improved this paper.

References

1. Albrecht, M.R., Player, R., Scott, S.: On the concrete hardness of learning with errors. *Journal of Mathematical Cryptology* 9(3):169–203; software: <https://github.com/malb/lattice-estimator>, commit 3e48ef4 (2015)
2. Brakerski, Z.: Fully homomorphic encryption without modulus switching from classical GapSVP. In: *Advances in Cryptology – CRYPTO 2012*. LNCS, vol. 7417, pp. 868–886. Springer (2012)
3. Brakerski, Z., Gentry, C., Vaikuntanathan, V.: (Leveled) fully homomorphic encryption without bootstrapping. In: *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference (ITCS 2012)*. pp. 309–325. ACM (2012)
4. Chen, H., Han, K.: Homomorphic lower digits removal and improved FHE bootstrapping. In: *EUROCRYPT 2018*. LNCS, vol. 10820, pp. 315–337. Springer (2018)
5. Choe, H., Kim, J., Stehlé, D., Suvanto, E.: Leveraging discrete CKKS to bootstrap in high precision. In: *Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security*. ACM (2025). <https://doi.org/10.1145/3719027.3765080>
6. Fan, J., Vercauteren, F.: Somewhat practical fully homomorphic encryption. *Cryptography ePrint Archive*, Paper 2012/144 (2012)
7. Geelen, R., Iliashenko, I., Kang, J., Vercauteren, F.: On polynomial functions modulo p^e and faster bootstrapping for homomorphic encryption. In: *EUROCRYPT 2023*. LNCS, vol. 14006, pp. 257–286. Springer (2023)
8. Geelen, R., Vercauteren, F.: Fully homomorphic encryption for cyclotomic prime moduli. In: *Advances in Cryptology – EUROCRYPT 2025*. LNCS, vol. 15603, pp. 366–397. Springer (2025). https://doi.org/10.1007/978-3-031-91131-6_13
9. Geelen, R., Vercauteren, F.: Better GBFV bootstrapping and faster encrypted edit distance computation. *IACR Communications in Cryptology* 2(4) (2026). <https://doi.org/10.62056/a3n56chdj>, article 35
10. Gentry, C.: Fully homomorphic encryption using ideal lattices. In: *Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC 2009)*. pp. 169–178. ACM (2009)
11. Halevi, S.: HELib issue #80: Bootstrapping with large p . <https://github.com/homenc/HElib/issues/80> (2019)
12. Halevi, S., Shoup, V.: Algorithms in HELib. In: *CRYPTO 2014*. LNCS, vol. 8616, pp. 554–571. Springer (2014)

13. Halevi, S., Shoup, V.: Bootstrapping for HELib. In: EUROCRYPT 2015. LNCS, vol. 9056, pp. 641–670. Springer (2015)
14. Halevi, S., Shoup, V.: Faster homomorphic linear transformations in HELib. In: CRYPTO 2018. LNCS, vol. 10993, pp. 93–120. Springer (2018)
15. Huo, M., Wu, K., Ye, Q.: A note on lower digits extraction polynomial for bootstrapping. Cryptology ePrint Archive, Paper 2019/677 (2019), <https://eprint.iacr.org/2019/677>
16. Ma, S., Huang, T., Wang, A., Wang, X.: Accelerating BGV bootstrapping for large p using null polynomials over $\mathbb{Z}_{p^e}$. In: EUROCRYPT 2024. LNCS, vol. 14652, pp. 403–432. Springer (2024)
17. Ma, S., Huang, T., Wang, A., Wang, X.: Faster BGV bootstrapping for power-of-two cyclotomics through homomorphic NTT. In: Advances in Cryptology – ASIACRYPT 2024. LNCS, vol. 15484, pp. 143–175. Springer (2024). https://doi.org/10.1007/978-981-96-0875-1_5
18. Ma, S., Huang, T., Wang, A., Wang, X.: Practical dense-key bootstrapping with subring secret encapsulation. In: Advances in Cryptology – ASIACRYPT 2025. LNCS, vol. 16251, pp. 101–130. Springer (2025). https://doi.org/10.1007/978-981-95-5122-4_4
19. Ogilvie, T.: On the concrete hardness gap between MLWE and LWE. Cryptology ePrint Archive, Paper 2026/279 (2026), <https://eprint.iacr.org/2026/279>. Reference implementation: <https://github.com/Tab0g/mlwe-hybrids>
20. Okada, H., Player, R., Pohmann, S.: Homomorphic polynomial evaluation using Galois structure and applications to BFV bootstrapping. In: ASIACRYPT 2023. LNCS, vol. 14443, pp. 69–100. Springer (2023)
21. Paterson, M.S., Stockmeyer, L.J.: On the number of nonscalar multiplications necessary to evaluate polynomials. SIAM Journal on Computing **2**(1), 60–66 (1973)
22. Peikert, C., Pepin, Z.: Vive Galois! Part 1: Optimal SIMD packing and packed bootstrapping for FHE. In: Theory of Cryptography – TCC 2025, Part I. LNCS, vol. 16268, pp. 185–219. Springer (2025). https://doi.org/10.1007/978-3-032-12287-2_7
23. Serre, J.P.: Linear Representations of Finite Groups, Graduate Texts in Mathematics, vol. 42. Springer-Verlag (1977)
24. Wang, Q., Wang, L.P.: A novel asymmetric BSGS polynomial evaluation algorithm under homomorphic encryption. In: Proceedings of the 20th ACM Asia Conference on Computer and Communications Security (ASIA CCS ’25). pp. 30–44. ACM (2025). <https://doi.org/10.1145/3708821.3710822>

A Proof of Proposition 1

We show that P_A is an odd polynomial, i.e., $P_A(X) = -P_A(-X)$.

For any $(\eta, \lambda) \in [-B, B]^2$, the symmetry of the domain gives $(-\eta, -\lambda) \in [-B, B]^2$. Evaluating P_A at the corresponding support point:

$$P_A((-\eta)A + (-\lambda)) = -\lambda = -P_A(\eta A + \lambda).$$

Since $(-\eta)A + (-\lambda) = -(\eta A + \lambda)$, this shows $P_A(-x) = -P_A(x)$ for all $x \in S_A$.

Now consider the polynomial $\tilde{P}(X) := -P_A(-X) \in \mathbb{F}_p[X]$. By the above, $\tilde{P}(x) = P_A(x)$ for every $x \in S_A$, so $\tilde{P}$ and P_A agree on all $|S_A|$ points of

S_A . Since both polynomials have degree $< |S_A|$, the uniqueness of Lagrange interpolation over a field implies $\tilde{P}(X) = P_A(X)$ as formal polynomials. Writing $P_A(X) = \sum_k c_k X^k$, the identity $P_A(X) = -P_A(-X)$ gives

$$\sum_k c_k X^k = - \sum_k c_k (-X)^k = \sum_k (-1)^{k+1} c_k X^k.$$

Comparing coefficients: $c_k = (-1)^{k+1} c_k$ for all k . When k is even, this forces $c_k = -c_k$, hence $c_k = 0$ (since $\text{char}(\mathbb{F}_p) = p > 2$). Therefore only odd-degree monomials survive, and the nonzero term count is at most $(|S_A| - 1)/2$.

B Proof of Theorem 2 (Order-Four Character Filter)

Lemma 3 (Support stability). *Let $A \in \mathbb{F}_p^\times$ satisfy $A^2 \equiv -1 \pmod{p}$ and be B -injective. Then $A \cdot S_A = S_A$.*

Proof. For any $x = \eta A + \lambda \in S_A$, we have $Ax = \eta A^2 + \lambda A = -\eta + \lambda A = \lambda A + (-\eta)$. Since $[-B, B]^2$ is symmetric, $(\lambda, -\eta) \in [-B, B]^2$ whenever $(\eta, \lambda) \in [-B, B]^2$. Thus $Ax \in S_A$, and since A is invertible, $A \cdot S_A = S_A$.

Proof (Proof of Theorem 2). We proceed in four steps.

Step 1 (Functional equation). Define $Q(X) := A^{-1}(AX - P_A(AX))$. For any $x = \eta A + \lambda \in S_A$, Lemma 3 gives $Ax = \lambda A + (-\eta)$, so $P_A(Ax) = -\eta$. Then:

$$Q(x) = A^{-1}(Ax - P_A(Ax)) = A^{-1}(Ax + \eta) = A^{-1}(\lambda A) = \lambda = P_A(x).$$

Since Q and P_A agree on all $|S_A|$ points and both have degree $< |S_A|$, they are identical as elements of $V = \mathbb{F}_p[X]/(G_{S_A}(X))$. Rearranging gives $P_A(AX) + A \cdot P_A(X) = AX$.

Step 2 (Monomial constraint). Write $P_A(X) = \sum_{k=0}^{|S_A|-2} c_k X^k$. Substituting into the functional equation:

$$\sum_k c_k A^k X^k + A \sum_k c_k X^k = AX \implies \sum_k c_k (A^k + A) X^k = AX.$$

For $k = 1$: $c_1(A + A) = A$, so $c_1 = 1/2 \pmod{p}$. For $k \neq 1$: if $c_k \neq 0$, then $A^k + A = 0$, i.e., $A^{k-1} = -1 = A^2$. Since $\text{ord}(A) = 4$, this holds iff $k - 1 \equiv 2 \pmod{4}$, i.e., $k \equiv 3 \pmod{4}$. The nonzero monomials are $\{k = 1\} \cup \{k \equiv 3 \pmod{4}\}$.

Step 3 (Term count). The odd integers $k \in [1, |S_A| - 2]$ satisfying $k \equiv 3 \pmod{4}$ are $\{3, 7, 11, \dots\}$. Their count is $(|S_A| - 5)/4$. Including the linear term $k = 1$, the total is $(|S_A| - 5)/4 + 1 + 1 = (|S_A| - 1)/4 + 1$.

Step 4 (Factored form). Writing $k = 4j + 3$ for $j \geq 0$: $X^{4j+3} = X^3 \cdot (X^4)^j$. Thus

$$P_A(X) = \frac{1}{2}X + X^3 \sum_{j=0}^{(|S_A|-5)/4} c_{4j+3} (X^4)^j = \frac{1}{2}X + X^3 \cdot Q(X^4),$$

where $Q(Y) = \sum_j c_{4j+3} Y^j$ has degree $(|S_A| - 5)/4$.

Table 9: Recommended admissible parameter sets at key weight $h = 12$. $B = \lceil I_{\text{bound}}(h, m) \rceil$ is computed per ring (17 or 16 on general cyclotomics depending on $\varphi(m)/m$; 9 on the power-of-two ring where $\varphi(m)/m = \frac{1}{2}$); every entry satisfies Theorem 3 with its own B . $(\star)$ = measured in Table 6; $(\dagger)$ = prime measured on the power-of-two ring $m = 2^{16}$ at $h = 26$ (Table 5); measured digit-polynomial degrees $(2B+1)^2 - 2$ match the per-ring B exactly.

p	A	h	B	suggested m	d	slots
1297 $(\dagger)$	36	12	17	$48547 = 43 \times 1129$	12	3948
1601 $(\star\dagger)$	40	12	17	$24901 = 37 \times 673$	3	8064
2521 $(\star\dagger)$	71	12	17	$50731 = 97 \times 523$	18	2784
2917 $(\star)$	54	12	17	$17587 = 43 \times 409$	6	2856
3137 $(\dagger)$	56	12	17	$41317 = 79 \times 523$	6	6786
4357	66	12	16	$23969 = 11 \times 2179$	2	10890
4513 $(\star\dagger)$	95	12	17	$37637 = 61 \times 617$	4	9240
7057 $(\dagger)$	84	12	17	$38407 = 193 \times 199$	6	6336
8101 $(\star)$	90	12	16	$23771 = 11 \times 2161$	5	4320
12101	110	12	16	$22187 = 11 \times 2017$	2	10080
13457 $(\star\dagger)$	116	12	17	$45193 = 43 \times 1051$	7	6300
14401 $(\star\dagger)$	120	12	17	$43033 = 23 \times 1871$	11	3740
15377 $(\star\dagger)$	124	12	16	$19679 = 11 \times 1789$	4	4470
65537 $(\star)$	256	12	17	$50731 = 97 \times 523$	18	2784
65537 $(\dagger)$	256	12	9	$65536 = 2^{16}$	1	32768

C Practical Admissibility: Details

The bound certificate of Lemma 1 fails for $p \equiv 1 \pmod{4}$ whose minimal order-four radix is large, e.g. $p = 12289$ ($A_{\min}=1479$), 40961 (14541), 786433 (100025) at $B = 17$; condition (iii) then requires a direct injectivity check (which can still succeed, e.g. at $p = 12289$), and Table 9 lists bound-certified sets only. For the generalized Fermat primes $p = A^2 + 1$ (A even) condition (ii) and the bound certificate *coincide*: the bound $B(A+1) < p/2$ at $B = 17$ holds iff $A \geq 36$, i.e. iff $p > 1225$, exactly the overflow cutoff. Note that A depends only on p , while $B = \Theta(\sqrt{h \cdot \varphi(m)/m})$ depends on the key weight and (weakly) on the ring: Table 9 is pinned to $h = 12$ with the conservative $B = 17$, so every entry remains admissible for its actual B ; for larger h the floor rises, e.g. $h = 26$ on general cyclotomics gives $B \approx 25$ and excludes $p < 2601$, whereas on power-of-two rings ($B = 13$ at $h = 26$) all entries remain admissible. This is also why moderate primes such as $p = 257$ support the algebraic structure but not the encrypted pipeline at this key weight. Condition (ii) likewise excludes small primes outright, e.g. $p = 37 < 1225$, while $p = 127 \equiv 3 \pmod{4}$ fails condition (i) and admits no order-four radix at all.

Table 10: Per-attack costs (bits) at the built chain, main-key instance, $h' = 120$. The binding attack is TF on every row, and λ^* is the row minimum.

set	built $\log_2 Q$	uSVP	BDD	dual	hyb. FF	hyb. TF	hyb. TT
80-B	1642	85.54	85.81	86.17	84.82	82.75	83.39
80-C	1787	84.53	84.33	84.89	84.17	81.94	82.48
80-D	2026	84.75	84.75	85.35	84.62	82.44	82.90
128-A	1407	153.38	153.14	154.42	140.64	129.81	130.85
128-C	1193	154.27	154.27	155.31	140.84	129.90	131.13
128-D	1268	153.81	153.81	154.78	140.78	129.84	130.98
128-E	1133	155.89	155.89	157.00	141.88	130.77	132.05

D Per-attack detail

Table 10 gives the per-attack breakdown behind the λ^* column of Table 4, at the modulus chain **HElib** built, for the main-key instance at $h' = 120$. The three hybrid columns are the primal hybrid at its three optimizer settings, in the notation of the estimator transcripts in the artifact: FF without meet-in-the-middle and without Babai lifting, TF with meet-in-the-middle and without Babai lifting, TT with both. Traces were retained for seven of the nine realized chains; 80-A and 128-B were recorded only as the aggregate minimum, 80.5 and 131.0 respectively.

Two features of the table drive the reporting convention of Section 6.1. First, the hybrid dominates everywhere, so uSVP alone is never the right figure. Second, the margin by which it dominates is not a constant: it is 2.3 to 2.8 bits on the 80-level rows and 23.6 to 25.1 bits on the 128-level rows. A fixed correction applied to a stock estimate would therefore be wrong by an order of magnitude on one of the two levels whichever value it took, and λ^* is computed as a genuine minimum over the estimator’s catalogue on each instance separately.

E Proof of Theorem 3 (Parameter Admissibility)

Proof. (i) -1 is a quadratic residue mod p iff $p \equiv 1 \pmod{4}$. (ii) is the bounded-support requirement of the overflow-part extraction: the input range $I_{\text{range}} = 2\lceil I_{\text{bound}} \rceil + 1 = 2B + 1$ must satisfy $I_{\text{range}}^2 < p$ for the two-digit encoding $\eta A + \lambda$ to be recoverable within $\mathbb{Z}_p$; note this is exactly $|S_A| < p$, the necessary condition for the $(2B + 1)^2$ support points to be distinct in $\mathbb{F}_p$. (iii) is injectivity of ϕ_A itself, whence $|S_A| = (2B + 1)^2$. In practice (iii) is certified through Lemma 1: given (ii), the separation $2B < |A|$ is automatic for order-four radices, since $p \mid A^2 + 1$ with $|A| \leq 2B$ would force $0 < A^2 + 1 < (2B + 1)^2 < p$; hence the single bound $|A|B + B < p/2$ certifies B -injectivity, and every entry of Table 9 is certified this way. The bound is sufficient but not necessary: a prime whose minimal order-four radix is large can violate it while ϕ_A remains injective (e.g.

$p = 12289$, $A_{\min} = 1479$, $B = 17$, injective by direct check), in which case (iii) requires explicit verification. The remaining claims are the content of Theorem 2. Ring-independence is immediate from the fact that (p, A, B) alone determine P_A .

F Proof of Theorem 4 (Tightness)

We prove the genericity claim of Theorem 4: for the parameters we consider, all $(|S_A| - 1)/4$ monomials X^k with $k \equiv 3 \pmod{4}$ carry nonzero coefficients in P_A .

Proof. The unconditional upper bound $|T| \leq \dim V_3 + 1$ follows directly from Theorem 1 and the proof of Theorem 2. For the attainment claim, consider the Lagrange interpolation formula

$$P_A(X) = \sum_{s \in S_A} P_A(s) \cdot L_s(X), \quad L_s(X) = \prod_{t \in S_A, t \neq s} \frac{X - t}{s - t}.$$

The coefficient of X^k in P_A is

$$c_k = \sum_{s \in S_A} P_A(s) \cdot [X^k]L_s(X),$$

where $[X^k]L_s(X)$ denotes the coefficient of X^k in the Lagrange basis polynomial L_s . For c_k to vanish for a specific $k \equiv 3 \pmod{4}$, the values $\{P_A(s) = \ell_s\}_{s \in S_A}$ would need to satisfy the non-trivial linear relation

$$\sum_{s \in S_A} \ell_s \cdot [X^k]L_s(X) = 0.$$

Since the Lagrange basis coefficients $[X^k]L_s(X)$ are determined by the geometry of S_A and not by the function values ℓ_s , vanishing is a codimension-one condition on (ℓ_s) for each such k . We do not claim that the vector arising from the digit-extraction map avoids all $(|S_A| - 1)/4$ of these conditions: that would require control of the ℓ_s we do not have. Attainment is instead verified by computation for each parameter set we report, and Theorem 4 is stated accordingly.

We verify computationally: for $p = 65537$, $A = 256$, $B = 17$, all 306 coefficients c_k with $k \equiv 3 \pmod{4}$ and $3 \leq k \leq 1223$ are nonzero in $\mathbb{F}_p$, confirming $|T| = 307$ with equality.

G Noise Analysis of Algorithm 3

We analyze the noise growth through the complete bootstrapping procedure (Algorithm 3) and verify that the output ciphertext decrypts correctly.

Notation. Let $\text{ct} = (b, a) \in R_q^2$ be a ciphertext with $b + as = m + p^r \epsilon$. Define the noise magnitude $\nu(\text{ct}) := \|m + p^r \epsilon\|_\infty$. Decryption is correct iff $\nu(\text{ct}) < q/2$. The capacity is $\kappa(\text{ct}) := \log_2(q/\nu(\text{ct}))$.

Noise primitives. For ciphertexts at modulus q with noise magnitudes ν_1, ν_2 :

$$\nu(\text{ct}_1 + \text{ct}_2) \leq \nu_1 + \nu_2, \quad (5)$$

$$\nu(\text{ct} \cdot c) \leq \|c\|_1 \cdot \nu_1 \quad (\text{plaintext-ciphertext}), \quad (6)$$

$$\nu(\text{ct}_1 \otimes \text{ct}_2) \leq \delta_R \cdot \nu_1 \cdot \nu_2 \quad (\text{tensor product}), \quad (7)$$

$$\nu(\text{KS}(\text{ct})) \leq \nu(\text{ct}) + E_{\text{ks}} \quad (\text{key-switch}), \quad (8)$$

$$\nu(\text{ModSw}_{q \rightarrow q'}(\text{ct})) \leq \frac{q'}{q} \cdot \nu(\text{ct}) + B_{\text{scale}}, \quad (9)$$

where δ_R is the ring expansion factor, $E_{\text{ks}} = O(\sqrt{N} \cdot \beta_{\text{ks}}^2 \cdot \sigma)$ is the key-switching noise, and $B_{\text{scale}} = O(p^r \sqrt{N})$ is the modulus-switching rounding error.

Step 1 (SlotToCoeff). The linear transform $\text{ct} \leftarrow \sum_{i=0}^{D-1} \text{diag}_i \cdot \sigma^i(\text{ct})$ involves $O(\sqrt{D})$ automorphisms (each a key-switch) and D plaintext-ciphertext multiplications. Let ν_0 denote the input noise. After BSGS evaluation with hoisting:

$$\nu_1 \leq D \cdot \|\text{diag}\|_\infty \cdot \nu_0 + O(\sqrt{D}) \cdot E_{\text{ks}}.$$

Step 2 (ModRaise + Inner Product). Modulus switching to q_0 scales noise by q_0/q and adds B_{scale} . The modulus raise from q_0 to Q is noiseless (just reinterpretation). The inner product $b + a \cdot \text{Enc}_{s'}(s; p^e)$ produces a fresh high-capacity ciphertext:

$$\nu_2 = O(h \cdot \sigma \cdot p^e),$$

where h is the secret key Hamming weight. The capacity after this step is $\kappa_2 \approx \log_2(Q/\nu_2) \approx 12$ levels.

Step 3 (CoeffToSlot). Identical structure to Step 1. The noise grows to:

$$\nu_3 \leq D \cdot \|\text{diag}'\|_\infty \cdot \nu_2 + O(\sqrt{D}) \cdot E_{\text{ks}}.$$

After modulus switching to consume the linear-transform levels, the remaining capacity for digit extraction is $\kappa_3 \approx 12$ levels.

Step 4 (Character-Filtered Digit Extraction). Let $\nu_{\text{in}} := \nu_3$ and let q_ℓ denote the modulus at level ℓ (with $q_0 > q_1 > \dots$). The evaluation of $P_A(x) = \frac{1}{2}x + x^3Q(x^4)$ proceeds as:

Phase 1 (depth 0 $\rightarrow$ 2): Three multiplications produce $x^2, x^3, Y = x^4$.

$$\nu(x^2) \leq \delta_R \cdot \nu_{\text{in}}^2 + E_{\text{ks}}, \quad \nu(x^4) \leq \delta_R \cdot \nu(x^2)^2 + E_{\text{ks}} \leq \delta_R^3 \cdot \nu_{\text{in}}^4 + O(E_{\text{ks}}).$$

After modulus switching to level $\ell = 2$: $\nu_Y \leq (q_2/q_0) \cdot \delta_R^3 \cdot \nu_{\text{in}}^4 + O(B_{\text{scale}})$.

Phase 2 (depth 2 $\rightarrow$ $2 + \lceil \log_2 k \rceil$): Baby-step powers $Y^2, \dots, Y^{k-1}$ are built by sequential multiplication with modulus switching after each. At depth $2 + j$:

$$\nu(Y^{2^j}) \leq \frac{q_{2+j}}{q_{2+j-1}} \cdot \delta_R \cdot \nu(Y^{2^{j-1}})^2 + B_{\text{scale}} + E_{\text{ks}}.$$

Phase 3 (depth $2 + \lceil \log_2 k \rceil \rightarrow 2 + \lceil \log_2 k \rceil + \lceil \log_2 m \rceil$): Giant-step powers $Z = Y^k, Z^2, \dots, Z^{m-1}$ at progressively deeper levels.

Phase 4 ($\text{depth} \rightarrow \lceil \log_2 k \rceil + 3$): Each accumulation step computes $y_j \cdot \text{ct}_{Z^j}$. Since $y_j = \sum_i q_{jk+i} \cdot \text{ct}_{Y^i}$ is a plaintext-ciphertext linear combination (noise grows only by $\|q\|_\infty$ factor via (6)), the multiplication with Z^j is the only non-scalar operation per giant step:

$$\nu(y_j \cdot Z^j) \leq \delta_R \cdot \left(\sum_i |q_{jk+i}| \cdot \nu(Y^i) \right) \cdot \nu(Z^j) + E_{\text{ks}}.$$

Phase 5 ($\text{depth} \rightarrow \lceil \log_2 k \rceil + 4$): Assembly $x^3 \cdot Q(Y) + \frac{1}{2}x$ adds one final multiplication level:

$$\nu_{\text{out}} \leq \delta_R \cdot \nu(x^3) \cdot \nu(\text{ct}_Q) + \frac{1}{2}\nu_{\text{in}} + E_{\text{ks}}.$$

Total depth and level consumption. The multiplicative depth of Step 4 is:

$$\underbrace{2}_{\text{Phase 1}} + \underbrace{\lceil \log_2 k \rceil}_{\text{Phase 2}} + \underbrace{1}_{\text{Phase 4 accumulation}} + \underbrace{1}_{\text{Phase 5 assembly}} = \lceil \log_2 k \rceil + 4.$$

For $k = 18$: total depth $= \lceil \log_2 18 \rceil + 4 = 5 + 4 = 9$. The baseline PS on degree 1223 (with $k = 35$) has depth $\lceil \log_2 35 \rceil + 1 = 7$. Our evaluation consumes 2 additional levels.

Sufficiency of the modulus chain. The HElib parameter set with bits = 1500 provides a modulus chain of ~ 12 levels for digit extraction (after Steps 1–3 consume their share). Our depth-9 evaluation uses 9 of these 12 levels, leaving 3 levels of margin. This is confirmed experimentally: all 2784 slots decrypt correctly across all runs, with no capacity overflow observed.

Comparison with baseline noise. The baseline PS evaluation performs 69 non-scalar multiplications at depth 7. Our evaluation performs 38 multiplications at depth 9. Since each multiplication introduces noise $\sim E_{\text{ks}}$ after modulus switching, the total accumulated key-switching noise is:

$$\text{Ours: } 38 \cdot E_{\text{ks}}, \quad \text{Baseline: } 69 \cdot E_{\text{ks}}.$$

Our evaluation accumulates less key-switching noise (38 vs. 69 contributions) but consumes 2 more modulus levels. The net effect is that both evaluations succeed within the available modulus budget, with our method having slightly more key-switching noise margin but slightly less level margin.

H Proof of Theorem 5 (Butterfly Decomposition)

We give the complete proof for the dominant block of our parameter set, $D = 96$ arising from the factor $M_1 = 97$ of $m = 50731 = 97 \times 523$; the argument for a general one-dimensional factor with $D \mid p^d - 1$ is identical, with $96 = 2^5 \times 3$ replaced by the prime factorization $D = p_1^{a_1} \cdots p_k^{a_k}$.

Proof. Step 1 (Closed form and Rader reindexing). For the 97-part, the local slot field is $\mathbb{F}_{p^6}$ (since $\text{ord}_{97}(p) = 6$, so $97 \mid p^6 - 1$), and we fix a primitive 97-th root of unity $\zeta \in \mathbb{F}_{p^6}$. Direct computation of the EvalMap factor (SageMath script included in our artifact) shows that the 96×96 Step2Matrix S of this block has the closed form

$$S[i][j] = \zeta^{i \cdot 38 \cdot 5^{-j} \bmod 97}, \quad i \in (\mathbb{Z}/97\mathbb{Z})^\times, j \in \mathbb{Z}/96\mathbb{Z},$$

where 5 is a fixed primitive root modulo 97 (indeed $5^{48} \equiv -1 \pmod{97}$) and the unit $38 \in (\mathbb{Z}/97\mathbb{Z})^\times$ is a fixed offset determined by HELib's choice of slot representatives. The column index j enumerates the Galois automorphisms, and enters the exponent only through 5^{-j} ; the row index i enumerates the nonzero residues modulo 97. Following Rader, we reindex the rows logarithmically: writing $i = 5^{i'} \bmod 97$ with $i' \in \mathbb{Z}/96\mathbb{Z}$ (a bijection since 5 generates $(\mathbb{Z}/97\mathbb{Z})^\times$),

$$S[5^{i'}][j] = \zeta^{38 \cdot 5^{i'} \cdot 5^{-j}} = \zeta^{38 \cdot 5^{i'-j}} = c_{i'-j \bmod 96}, \quad c_t := \zeta^{38 \cdot 5^t}.$$

Hence $P_r S = C$, where P_r is the row permutation $i' \mapsto 5^{i'}$ (a free slot rearrangement in HELib) and C is the 96×96 *circulant* with symbol $(c_t)_{t \in \mathbb{Z}/96\mathbb{Z}}$: the prime-indexed evaluation map has become a cyclic convolution of length $D = 96$.

Step 2 (Mixed-radix stage factorization). Let $\omega \in \mathbb{F}_{p^2}$ be a primitive 96-th root of unity (existence in Step 3) and let F_{96} denote the 96-point DFT matrix $(\omega^{uv})_{u,v}$. A circulant is diagonalized by the DFT:

$$C = F_{96}^{-1} \cdot \text{diag}(w) \cdot F_{96}, \quad w = F_{96} c,$$

where the diagonal vector $w \in \mathbb{F}_{p^6}^{96}$ is precomputed (we write w rather than B to avoid a collision with the digit bound). Each DFT factor now factors into butterfly stages exactly as in the theorem statement. First, Good–Thomas: since $96 = 32 \times 3$ with $\gcd(32, 3) = 1$, the CRT bijection $\mathbb{Z}/96\mathbb{Z} \cong \mathbb{Z}/32\mathbb{Z} \times \mathbb{Z}/3\mathbb{Z}$ yields $F_{96} = P_{\text{CRT}}^{-1} (F_{32} \otimes F_3) P_{\text{CRT}}$ with *no* twiddle factors between the coprime sub-DFTs, where F_{32} and F_3 are built on the CRT-matched roots ω^{33} and ω^{64} ($33 = 3 \cdot (3^{-1} \bmod 32)$, $64 = 32 \cdot (32^{-1} \bmod 3)$); the permutations P_{CRT} are free slot rearrangements. Second, Cooley–Tukey: the radix-2 factorization

$$F_{32} = \left(\prod_{j=1}^5 D_j \cdot (I_{2^{j-1}} \otimes B_2 \otimes I_{2^{5-j}}) \right) \cdot R_{32}, \quad B_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix},$$

with diagonal twiddle matrices D_j and the bit-reversal permutation R_{32} (absorbed into Π_{in} at the chain edge), expresses F_{32} as five stages. A stage at stride s pairs the slots i and $i + s$, so its nonzero *cyclic* diagonals sit at offsets $\{0, +s, -s\}$: three diagonals and hence 2 automorphisms per stage, except the stride- $D/2$ stage, where $+s \equiv -s$ and 2 diagonals suffice (1 automorphism). In the 96-point frame the five radix-2 stages have offset sets $\{0, 48\}$, $\{0, \pm 24\}$, $\{0, \pm 36\}$, $\{0, \pm 30\}$, $\{0, \pm 33\}$, costing $1 + 2 + 2 + 2 + 2$ automorphisms, and the 3-point DFT F_3 has offsets $\{0, 32, 64\}$ with $64 \equiv -32$, costing 2. Each

cascade therefore uses $\sum_i (2a_i - 1)(p_i - 1) = 9 \cdot 1 + 1 \cdot 2 = 11$ automorphisms, and for general $D = p_1^{a_1} \cdots p_k^{a_k}$ the same two-step factorization gives $\sum_{i=1}^k (2a_i - 1)(p_i - 1) = O(\log D)$.

Step 3 (Twiddle containment in $\mathbb{F}_{p^2}$). All entries of the stage matrices are 96-th roots of unity. Since $p - 1 = 2^{16}$ we have $32 \mid p - 1$, and since $p + 1 = 65538 = 2 \cdot 3 \cdot 10923$ we have $3 \mid p + 1$; as $\gcd(32, 3) = 1$, it follows that $96 \mid p^2 - 1$, so all 96-th roots of unity lie in $\mathbb{F}_{p^2} \subseteq \mathbb{F}_{p^6}$. More precisely, the radix-2 twiddles are 32-nd roots of unity and lie in $\mathbb{F}_p$ itself (scalar constants), while the radix-3 twiddles are cube roots of unity in $\mathbb{F}_{p^2} \setminus \mathbb{F}_p$, representable as degree-1 constants in the slot algebra. The Rader diagonal $w \in \mathbb{F}_{p^6}^{96}$ is applied as a single slot-wise constant multiplication and consumes no automorphism.

Step 4 (Stage-product identity). Assembling Steps 1–3,

$$S = P_r^{-1} \cdot F_{96}^{-1} \cdot \text{diag}(w) \cdot F_{96} = \Pi_{\text{out}} \cdot \left(\prod_{i,j} B_{i,j}^{(\text{inv})} \right) \cdot \text{diag}(w) \cdot \left(\prod_{i,j} B_{i,j} \right) \cdot \Pi_{\text{in}},$$

where each $B_{i,j}$ (resp. $B_{i,j}^{(\text{inv})}$) is a radix- p_i butterfly layer of the forward (resp. inverse) DFT cascade and $\Pi_{\text{in}}, \Pi_{\text{out}}$ collect the free permutations P_r, P_{CRT} . We verified this identity entry-by-entry over the slot field: the product of all stage matrices reproduces every one of the $96 \times 96 = 9216$ entries of the Step2Matrix exactly (SageMath, included in our artifact). When the block matrix is itself of DFT type—the generic case covered by the theorem statement—a single cascade suffices and the automorphism count is $\sum_i (2a_i - 1)(p_i - 1)$; for the Rader-reduced $D = 96$ block the complete factorization uses two cascades plus one free diagonal, i.e. $2 \sum_i (2a_i - 1)(p_i - 1) + O(1)$ automorphisms, which is still $O(\log D)$, compared to $2\lceil\sqrt{D}\rceil - 2 = O(\sqrt{D})$ for BSGS. As a consistency check, the formula predicts 13, 15, 17, 19 automorphisms per cascade at the single-cascade dimensions $D = 192, 288, 576, 1152$ measured in our artifact, matching the measured counts 19, 19, 23, 25 up to a small constant from the edge permutations and diagonal handling.

I Proof of Theorem 6 (Butterfly Correctness)

Proof. Let $\omega = \omega_{96} \in \mathbb{F}_{p^{18}}$ be a primitive 96-th root of unity (exists by Lemma 2). The Step2Matrix M_D is the 96×96 DFT matrix with (i, j) -entry ω^{ij} :

$$(M_D)_{i,j} = \omega^{ij}, \quad 0 \leq i, j < 96.$$

Step 1. By Good–Thomas, the CRT bijection $\phi : \mathbb{Z}/96\mathbb{Z} \xrightarrow{\sim} \mathbb{Z}/32\mathbb{Z} \times \mathbb{Z}/3\mathbb{Z}$ (with $\phi(n) = (n \bmod 32, n \bmod 3)$) gives

$$P_{\text{CRT}}^{-1} \cdot M_D \cdot P_{\text{CRT}} = F_{32} \otimes F_3,$$

where F_{32} is the 32-point DFT with root ω^3 (since $(\omega^3)^{32} = \omega^{96} = 1$) and F_3 is the 3-point DFT with root ω^{32} (since $(\omega^{32})^3 = \omega^{96} = 1$). The permutation P_{CRT} is a free slot rearrangement.

Step 2. The radix-2 Cooley–Tukey factorization of F_{32} gives

$$F_{32} = \prod_{j=1}^5 D_j \cdot (I_{2^{j-1}} \otimes B_2 \otimes I_{2^{5-j}}),$$

where $D_j = \text{diag}(1, \omega^{3 \cdot 2^{5-j}}, \omega^{3 \cdot 2 \cdot 2^{5-j}}, \dots)$ are diagonal twiddle matrices and $B_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$. Each factor $D_j \cdot (I \otimes B_2 \otimes I)$ is a sparse matrix with exactly 2 nonzero entries per row.

Step 3. The 3-point DFT F_3 with root ω^{32} is

$$F_3 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & \omega^{32} & \omega^{64} \\ 1 & \omega^{64} & \omega^{32} \end{pmatrix},$$

which has 3 nonzero diagonals (at offsets 0, 1, 2 modulo 3). *Step 4.* Combining:

$$M_D = P_{\text{CRT}}^{-1} \cdot (F_{32} \otimes F_3) \cdot P_{\text{CRT}} = \Pi_{\text{out}} \cdot \left(\prod_{j=1}^5 B_{1,j} \right) \cdot B_{2,1} \cdot \Pi_{\text{in}},$$

which is exactly the product of butterfly layers in Theorem 5.