

Jambon: Asynchronous Universally Composable Threshold FHE with Low Communication

Jung Hee Cheon¹, Kevin Choi¹, Changmin Lee², JeongHwan Lee², Junho Lim¹, and Duc Anh Luong²

¹ Seoul National University, Seoul, Republic of Korea

`jhcheon@snu.ac.kr`, `kc2853@snu.ac.kr`, `noviajun@snu.ac.kr`

² Korea University, Seoul, Republic of Korea

`changminlee@korea.ac.kr`, `hwani0814@korea.ac.kr`, `adluong11@korea.ac.kr`

Abstract. Threshold fully homomorphic encryption distributes decryption authority among multiple parties, but achieving strong security can significantly increase the required ciphertext modulus and the size of each decryption response. Existing parameter analyses typically focus on threshold decryption alone. In particular, the recent work of Kim et al. (KLL⁺26) analyzes correctness and strong simulation security of partial decryption without covering distributed key generation (DKG). When the parties generate the keys themselves, the DKG and partial-decryption protocols must satisfy their respective correctness and security requirements.

We revisit correctness and simulation-security requirements for threshold decryption when it is composed with DKG. We then introduce Jambon, which applies deterministic rounding to partial decryptions to further reduce the modulus used for decryption responses. The composed security bound accounts separately for setup and decryption errors. We prove exact recovery for BFV and extend the approach to CKKS under an additional condition.

Under the stated temporary-HE assumptions and resource hybrids, we establish universally composable security with strong threshold privacy against static malicious corruption in the random-oracle model. We prove security for the DKG and its composition with verifiable partial decryption. Under the stated algebraic and statistical parameters, Jambon’s numerical decryption responses are at least 60% smaller than the native responses of the updated KLL⁺26 strong-security construction at every threshold considered for 256, 512, and 1024 parties. This comparison uses the specified sufficient bounds and excludes proof and setup traffic.

1 Introduction

Fully homomorphic encryption (FHE) lets a server compute on encrypted data without learning the inputs [4, 13, 9]. Threshold FHE (ThFHE) distributes the decryption key among N parties so that responses from any t parties recover the result [2]. The server can thus compute without authority to release the output. Distributed key generation (DKG) lets the parties generate the secret key and

its shares without a trusted dealer [15]. For FHE, setup must also produce public and evaluation keys.

Lattice-based ThFHE communication depends largely on ciphertext and response moduli. For ring degree n , one element modulo q requires about $n \log_2 q$ bits. Correctness bounds q from below through the accumulated evaluation noise. Privacy can require additional noise to hide the secret shares: conventional noise flooding increases the reconstruction margin and hence the required modulus.

KLL⁺26 [17] sharpens modulus bounds for correctness and simulation security of threshold FHE. To realize such parameters without a trusted dealer, a DKG must securely generate the corresponding keys and shares. Its setup transcript must be simulated jointly with the shares and public/evaluation keys. This raises the question: *how compact can threshold decryption remain when composed with distributed setup?*

Even when the ciphertext modulus satisfies these bounds, decryption responses need not use that modulus. Can the response modulus be reduced further without sacrificing correctness or privacy? Rounding compresses ciphertexts in Kyber [1], distributed-PRF partial evaluations [10], and threshold-decryption responses in other settings [26,23,24]. For ThFHE, separately rounded responses must remain correct after interpolation and jointly simulatable from only the permitted information.

Beyond parameter analysis, setup and decryption must remain secure when composed, including before authorized plaintext release. Under strong threshold privacy, the simulator must answer partial-decryption requests without the plaintext before release [12]. Existing works address parts of this picture: MHP⁺26 [20] gives an efficient semi-honest DKG for CKKS, while BFM⁺25 [5] establishes malicious universally composable (UC) security for BGV and describes a CKKS extension. Other works reduce decryption communication under different assumptions on servers, randomness, or the participating set [26,23,24].

The challenge is to obtain compact threshold decryption with composable security for distributed setup and decryption.

1.1 Our contributions

We present Jambon, a ThFHE construction with distributed key generation and compact asynchronous threshold decryption, retaining the underlying FHE encryption and evaluation algorithms. Here, *asynchronous* means parties compute numerical responses without prior responder-set agreement (i.e. agreement on the exact set of parties partaking in threshold decryption). Our contributions are twofold.

Compact asynchronous decryption with distributed setup We revisit correctness and simulation-security requirements for threshold decryption when it is composed with DKG. Furthermore, Jambon deterministically rounds partial decryptions, representing responses over a smaller modulus than the ciphertexts. The composed security bound includes setup and decryption distinguishing errors.

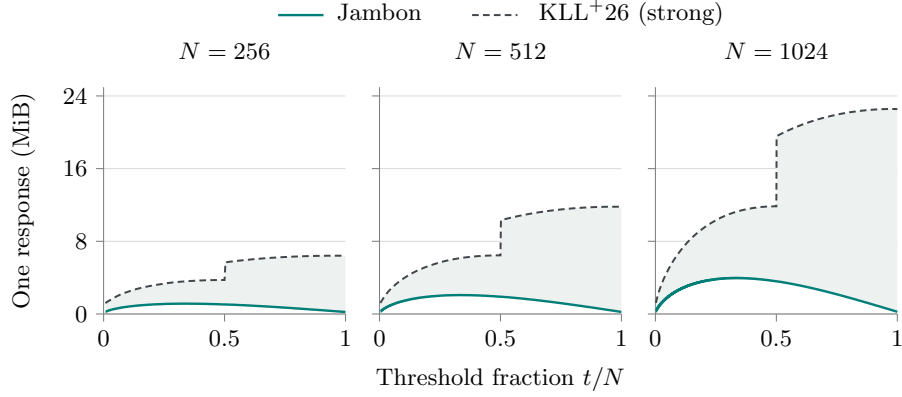


Fig. 1. Numerical response sizes at $n = 2^{16}$ for every integer $2 \leq t \leq N$. Jambon is solid; the strong-security baseline from the updated KLL⁺26 ePrint is dashed. Lines join certified samples without smoothing; panels share a linear scale. The KLL jump above $t/N = 1/2$ reflects its changed interpolation geometry (Section 8.1). Ciphertext, proof, authentication, and setup traffic is excluded.

Using the reusable key/zero-share masking mechanism³, each party sends one deterministically rounded ring element without fresh private decryption noise or prior responder-set agreement.

We prove exact BFV recovery accounting for homomorphic, interpolation, and rounding errors, and give CKKS profiles under additional conditions. Under the stated algebraic and statistical parameters, Jambon’s BFV responses are at least 60% smaller than native responses of the strong-security construction in the updated KLL⁺26 ePrint for every threshold $2 \leq t \leq N$ with $N \in \{256, 512, 1024\}$; representative profiles save 64.8–87.2%. These numerical response sizes exclude ciphertext, verification, and setup traffic. Section 8 gives the parameter comparison; Figure 1 plots the results.

Universally composable security for distributed setup and decryption We prove UC security with strong threshold privacy against static malicious corruption in the random-oracle model, in the stated resource hybrids, for DKG and its composition with verifiable partial decryption.

The partial-decryption simulator answers without the plaintext before release and remains consistent afterwards [12]; every accepted response is linked to DKG shares. We give a six-wave static-malicious DKG using local commit-and-prove, authenticated channels, and notifications of round completion. We simulate setup disclosures jointly with the keys and shares required by the threshold-decryption reduction, establishing composable security for setup and asynchronous decryption (Sections 5 to 7).

³ The updated KLL⁺26 ePrint [17] provides strong security.

Table 1. Selected ThFHE profiles. Here q is the evaluated-ciphertext modulus before server preprocessing, $r < q$ is a reduced response modulus, and ℓ_i counts $\{0, 1\}$ -LSS scalar components. Domains omit proof and authentication traffic; ring degrees and moduli are source-specific.

Work (variant)	DKG	Async.	Response	Security
BGG ⁺ 18 [2] ($\{0, 1\}$ -LSS)	No	Yes	$\mathbb{Z}_q^{\ell_i}$	SIM, hon.
BFM ⁺ 25 [5] (Protocol 1)	Yes	Yes	$\mathcal{R}_q$	UC, mal.
KLL ⁺ 26 [17] (strong)	No	Yes	$\mathcal{R}_q$	SIM, strong, hon.
PS24 [26] (postproc.)	No	—	$\mathbb{Z}_r; r < q$	SIM, hon.
OT25 [23] (module)	No	No	$\mathcal{R}_r; r \mid q$	SIM, hon.
OT26 [24] ($\{0, 1\}$ -LSS)	No	Yes	$\mathbb{Z}_r^{\ell_i}; r \mid q$	SIM, hon.
OT26 (Shamir)	No	No	$\mathbb{Z}_r; r \mid q$	SIM, hon.
Jambon	Yes	Yes	$\mathcal{R}_r; r \mid q$	UC, strong, mal.

DKG: Yes denotes a distributed protocol in the stated resource hybrid; No denotes dealer setup for the selected profile. Async. means decryption without prior active-set agreement, combining any qualifying collection as responses arrive; it implies neither robust setup nor guaranteed delivery. For PS24, the selected profile fixes $t = N$. SIM denotes the source’s simulation definition; hon./mal. denote semi-honest/malicious corruption; strong denotes simulation without the plaintext before authorized release [12].

2 Related Work

Table 1 compares key generation, asynchronous decryption, response domains, and security for selected ThFHE profiles.

Rounding in related primitives. Kyber compresses ciphertext coefficients while retaining decryption precision [1]. CLW25 [10] rounds distributed-PRF partial evaluations and uses a second rounding to ensure agreement with centralized evaluation on random inputs; its threshold-PKE construction adds fresh partial-decryption noise. SPM25 [28] and SPM26 [29] provide further rounded distributed-PRF constructions, including adaptive security in the latter. Jambon addresses evaluated FHE ciphertexts and jointly simulates responses before and after plaintext release (Section 3).

Flooded and compressed ThFHE responses. BGG⁺18 [2] constructs ThFHE for arbitrary thresholds from linear secret sharing and noise flooding. BFM⁺25 [5] develops efficient asynchronous decryption with ring sharing and malicious security. KLL⁺26 [17] sharpens interpolation bounds for standard and strong simulation security, retaining native flooded responses at the ciphertext modulus. Jambon uses its sharing geometry and bounds, and proves decoding and simulation after deterministic response rounding. KLL⁺26 also provides implemented numerical experiments on the tightness of its interpolation bounds. Section 8 compares numerical partial-decryption payloads with uncompressed strong-security

KLL⁺26 under the stated parameters, with interpolation points selected separately for each construction. The metric excludes ciphertext, proof, authentication, and setup traffic; the reported savings therefore concern numerical responses rather than total protocol communication.

PS24 [26] uses a trusted server to rerandomize, flood, and Gaussian-round evaluated ciphertexts; further rounding compresses the responses by postprocessing its base scheme. Its proved compact construction uses additive all-party scalar-LWE sharing, with server randomness hidden from corrupted decryptors. OT25 [23] gives server-assisted low-communication ThFHE with padding for a bounded decryption workload and no fresh private partial-decryption noise. Jambon uses the same key and zero shares across ciphertexts and computes each response from a locally validated ciphertext.

The serverless constructions of OT26 [24] offer two different tradeoffs. The binary-sharing variant rounds flooded products and retains multiple shares; the Shamir variant applies the participating set’s weights before Gaussian randomized rounding and uses reusable pairwise-PRF zero masks. Jambon applies interpolation weights at the combiner, making local responses independent of the eventual set, and accounts for the resulting error amplification in its correctness bound. THED [25] compresses structured CKKS responses by selecting coefficients and truncating low bits; comparison with complete ring responses must also fix the useful plaintext slots and precision. Interactive alternatives use secure comparisons, preprocessed double sharings, or MPC-based modulus conversion [31,16,11]; their interaction and preprocessing used once per decryption incur different costs.

Strong simulation and distributed setup. EY24 [12] formalizes strong threshold-FHE simulation, withholding the evaluated plaintext until corrupted and queried parties qualify, and constructs PRF zero masks with fresh smudging noise. KLL⁺26 achieves strong simulation for asynchronous Shamir sharing using two reusable ring shares and a ciphertext-hash mask, under base-FHE security and RLWE in the random-oracle model [17, Fig. 4, Thm. 4.2]. Its optional omission of the small RLWE error retains fresh flooding [17, Rem. 1]. BFM⁺25 proves malicious UC security for a request-gated threshold functionality [5]. Jambon proves strong simulation for deterministic compressed responses, including incomplete sessions and responses exposed before verification. Its malicious protocol validates input proofs and public evaluation records locally; for CKKS, the theorem additionally requires the lift bound and exact-output stability in (Z). Adaptive-corruption threshold encryption, as studied in [30], provides a guarantee outside our static model.

Jambon’s key resharing uses the standard additive-to-Shamir conversion [21] for multiparty RLWE encryption [22], alongside one independent sharing of zero. Following KLL⁺26, each party reuses its key and zero shares across ciphertexts; a ciphertext-dependent hash supplies the mask multiplier. Jambon rounds the resulting masked response. Its temporary-HE approach follows MHP⁺26 [20], whose four-round protocol already accommodates resharing in its final round. BFM⁺25 also proves UC-secure DKG with an affine key-bias interface [5]. Jam-

bon’s functionality instead samples the prescribed joint record and permits abort; selecting which key-generation instances complete can still bias the completed keys. Its contribution is the joint setup simulation needed for the threshold-decryption security proof and the link from DKG shares to accepted responses. Six waves in its resource hybrid do not establish an implementation-level round improvement.

3 Technical Overview

Jambon combines deterministic rounding with KLL⁺26’s hashed zero-sharing mask [17, Fig. 4]; its setup supplies the joint distribution used by the simulation.

3.1 Local compression and reconstruction

Let $\mathcal{R} = \mathbb{Z}[X]/(X^n + 1)$ for a power-of-two degree n , with coefficient norm $\|\cdot\|_\infty$, and write $\mathcal{R}_q = \mathcal{R}/q\mathcal{R}$. For a BFV ciphertext (a, b) under secret s , assume $b - as = (q/p)m + e \pmod{q}$, where p is the plaintext modulus, m the intended result, and $\|e\|_\infty \leq B_e$ bounds the accumulated error. Choose odd moduli $p \mid r \mid q$ and let $L = q/r$. Here r is the response modulus and L the rounding gap.

Setup gives party i the evaluations $f_i = F(\alpha_i)$ and $g_i = G(\alpha_i)$ of independent Ring–Shamir polynomials of degree at most $t - 1$, with constants s and 0 . The public interpolation points use KLL⁺26’s spaced-monomial geometry [17]. For a canonical serialization τ of the key identifier, parameters, and full ciphertext, the public oracle supplies $h_\tau \in \mathcal{R}_q$. Using centered representatives and rounding each coefficient, party i returns

$$p_i = \left\lfloor \left\lceil \frac{[af_i + h_\tau g_i]_q}{L} \right\rceil \right\rfloor_r, \quad Lp_i = af_i + h_\tau g_i + \rho_i \pmod{q},$$

Here $\|\rho_i\|_\infty \leq (L - 1)/2$; key and mask contributions are rounded together.

For a qualified set S , integral interpolation coefficients λ_i^S reconstruct D times the constant, where D is a common power-of-two denominator-clearing factor. The zero mask cancels, so the reconstructed phase is

$$\left[Db - L \sum_{i \in S} \lambda_i^S p_i \right]_q \equiv \frac{q}{p} Dm + De - \sum_{i \in S} \lambda_i^S \rho_i \pmod{q}.$$

Let $M_1(S) = \sum_{i \in S} \|\lambda_i^S\|_1$. The rounding contribution is bounded by $\mathcal{B}(S) = (L - 1)M_1(S)/2$. Exact BFV recovery follows if $DB_e + \mathcal{B}(S) < q/(2p)$ for every qualified set. A proved upper bound $M \geq \max_S M_1(S)$ gives the sufficient condition

$$\frac{r}{p} > \left(1 - \frac{1}{L}\right) M + \frac{2DB_e}{L}.$$

Reconstruction recovers Dm and multiplies by D^{-1} modulo p . Simulation determines a sufficient L ; correctness then determines r . Only the response modulus

r is used for transmission. The CKKS core uses centered scalar division by D and the ciphertext scale, with separate no-wrap and precision bounds. Malicious CKKS additionally needs the exact-output condition (Z), including its lift and stability requirements; precision alone does not imply it (Sections 5 and 7.3).

3.2 Simulation across release

Fix a corrupted set C of size $c < t$. Conditioning on its zero shares leaves $k = t - 1 - c$ independent uniform coefficients in the representation of G by polynomials vanishing at the corrupted parties' points. The proof uses them as RLWE secrets, leaving the FHE secret and its public and evaluation keys intact. The proof adds small errors to these coefficient products before applying RLWE hybrids, making the unrounded-response polynomial uniform subject to its constant as and the corrupted evaluations. As in KLL⁺26 [17, Thm. 4.2], this remaining zero-sharing randomness supports replies before plaintext release.

At the publication reaching threshold, let T contain the $t - 1$ corrupted and previously published identities. For BFV, interpolation preserves their unrounded evaluations while replacing the constant by $b - (q/p)m$. The functionality applies this rule atomically with release of m . The old and new unrounded-response polynomials differ by

$$e \prod_{j \in T} (1 - Y \alpha_j^{-1}).$$

The shift uses only the interpolation coefficient at zero, avoiding the all-index bound in KLL⁺26's analysis. A bounded shift changes a rounded coefficient only near a boundary; uniformity modulo L at the first oracle query bounds this event. The probability that some zero share's coefficients are not collectively coprime to q is charged once: at error insertion if $k > 0$, and at release if $k = 0$. A union bound covers all candidate labels, parties, and coefficients (Sections D.1 and 7).

The malicious construction validates input proofs and replays public evaluation circuits locally. Each response proves the unique rounding relation for shares linked to DKG. Commitment binding fixes corrupted shares; hiding replaces commitments to honest shares by commitments to zero before the setup and decryption hybrids. Public exposure counts toward release, while local receipt with an accepted proof enables reconstruction. Section 5 and theorem 7.3 give the validation rules and joint guarantee.

3.3 Producing the full key and sharing distribution

Following MHP⁺26 [20], temporary HE evaluates the full key-generation circuit on the XOR of the parties' committed private random strings. An independent uniform honest string makes this XOR uniform, yielding the prescribed key distribution. The malicious protocol uses independent uniform masks $\text{fl}_j \in \mathcal{R}_q$ and the specified addition-preserving embedding into the temporary-HE plaintext

space to open the public outputs and $s + \sum_j \text{fl}_j$ modulo q . This algebraic embedding is distinct from label serialization and the CKKS slot embedding. Each dealer shares its private $-\text{fl}_j$ and independently shares zero; adding the public opening yields shares of s . This resharing uses the masked opening directly in $\mathcal{R}_q$; no integer opening is required. The six-wave schedule commits to the sharing polynomials before releasing evaluations and binds the final commitments to the resulting shares.

The proof simulates earlier openings and the adversary's stored sharing messages jointly with the final shares. It samples honest partial openings with the prescribed aggregate and samples the key and zero sharing messages consistently with corrupted outputs and fixed corrupted contributions. After hiding honest commitments, this simulated view does not require the remaining zero-sharing randomness used in the decryption security proof. The argument includes aborts and pending executions; it does not assert an unbiased key distribution conditioned on completion. [Sections 6](#) and [7](#) give the samplers, the temporary-HE requirements, and the composition proof.

4 Preliminaries

4.1 Notation

Let λ and κ denote computational and statistical security parameters, and let $[N] = \{1, \dots, N\}$. There are N parties, threshold t , and a power-of-two ring degree n . Write $\mathcal{R} = \mathbb{Z}[x]/(x^n + 1)$, $\mathcal{R}_q = \mathcal{R}/q\mathcal{R}$, and reserve Y for sharing polynomials. The notation $[v]_q$ denotes coefficient-wise representatives in $(-q/2, q/2]$; all polynomial norms are coefficient norms. We use $\xleftarrow{\$}$ for sampling, uniformly for a finite set, $\approx_c$ for computational indistinguishability, and $\lfloor z \rfloor = \lfloor z + 1/2 \rfloor$ coefficient-wise. For $r \mid q$, write $L = q/r$ and define modulus switching by

$$\lfloor v \rfloor_r := \left[\left\lfloor \frac{\lfloor v \rfloor_q}{L} \right\rfloor \right]_r \in \mathcal{R}_r.$$

By [Lemma 6](#), modulus switching preserves uniformity and, for odd L , its lifted rounding residue has coefficient norm at most $(L - 1)/2$.

We use $\|uv\|_\infty \leq \|u\|_1 \|v\|_\infty$ ([Lemma 2](#)).

4.2 Ring-Shamir sharing

We use Shamir sharing over $\mathcal{R}_q$ at evaluation points whose pairwise differences are invertible [\[27, 17\]](#). Fix $2 \leq t \leq N$ and $\alpha_0 = 0, \alpha_1, \dots, \alpha_N \in \mathcal{R}_q$ with $\alpha_i - \alpha_j$ a unit for $i \neq j$, including $i = 0$.

Definition 1 (Ring-Shamir syntax). To sample $(f_i)_{i \in [N]} \leftarrow \text{Share}(s)$, choose independent uniform $r_1, \dots, r_{t-1} \in \mathcal{R}_q$ and set $f_i = F(\alpha_i)$ for $F(Y) = s + \sum_{j=1}^{t-1} r_j Y^j$. For $|S| = t$, reconstruction is $s = \sum_{i \in S} \ell_i^S f_i$, where

$$\ell_i^S = \prod_{j \in S \setminus \{i\}} \frac{-\alpha_j}{\alpha_i - \alpha_j}.$$

The required invertibility and privacy properties are recalled in [Lemmas 3](#) and [4](#).

We use the equally spaced monomial evaluation points of [[17](#), Cor. 5.1],

$$N' = 2^{\lceil \log_2 N \rceil} \leq 2n, \quad \alpha_i = x^{(i-1)2n/N'}. \quad (1)$$

For odd q , every difference $\alpha_i - \alpha_j$ with $i \neq j$ is a unit in $\mathcal{R}_q$, so the corresponding Lagrange coefficients are well defined. Moreover, by [[17](#), Cor. 5.1], the denominators of the corresponding Lagrange coefficients can be cleared by $D = 2^{\lceil \log_2 t \rceil}$.

Thus, interpreting ℓ_i^S in $\mathcal{R} \otimes \mathbb{Q}$, we have the integral representatives

$$\lambda_i^S = D\ell_i^S \in \mathcal{R}, \quad \sum_{i \in S} \lambda_i^S = D, \quad \sum_{i \in S} \lambda_i^S f_i = Ds \pmod{q}. \quad (2)$$

All reconstruction norms refer to these representatives. The auxiliary algebraic proofs appear in [Section A](#).

4.3 Ring learning with errors (RLWE)

Definition 2 (Decisional RLWE). The assumption $\text{DRLWE}_{n,q,\chi_s,\chi_e}$ asserts that polynomially many samples $(a_j, [a_j s + e_j]_q)_j$ and $(a_j, u_j)_j$ are computationally indistinguishable, where $s \xleftarrow{\$} \chi_s$ is fixed and $a_j, u_j \xleftarrow{\$} \mathcal{R}_q$, $e_j \xleftarrow{\$} \chi_e$ are independent.

Security advantages. Write $\text{Adv}_{\text{RLWE}}(T_{\text{red}}, Q)$ for the maximum absolute difference between the output-one probabilities in the preceding two experiments, over distinguishers of running time at most T_{red} using at most Q samples with one shared secret; see [[19](#), Definitions 3.1 and 3.3]. The threshold-decryption security proof uses $\chi_s = U(\mathcal{R}_q)$ and $\chi_e = \chi$. For commitments, use the hiding and binding games of [[3](#), Section 8.12], under the prescribed public-parameter setup. In the hiding game, the adversary chooses valid equal-length m_0, m_1 and receives the commitment $\text{Com}(m_b; \vartheta)$, where Com is the commitment algorithm and ϑ is fresh commitment randomness. The advantage $\text{Adv}_{\text{Com}}^{\text{hide}}(T_{\text{red}})$ is the maximum absolute difference between its output-one probabilities for $b = 0, 1$. The binding advantage $\text{Adv}_{\text{Com}}^{\text{bind}}(T_{\text{red}})$ is the maximum probability of producing two valid openings of one commitment to distinct messages. All maxima use time bound T_{red} , including reduction overhead.

4.4 Underlying FHE

Let $\text{FHE} = (\text{KeyGen}, \text{Enc}, \text{Eval}, \text{Dec})$ have public key fhepk , including evaluation keys, and two-component ciphertexts $\text{ct} = (a, b) \in \mathcal{R}_q^2$ at the selected decryption level [[13,9](#)].

Definition 3 (Exact noisy-linear FHE (BFV)). For plaintext space $\mathcal{R}_p$, $p \mid q$, and $\Delta = q/p$, a ciphertext is B_e -admissible for $m \in \mathcal{R}_p$ when

$$b - as = \Delta m + e \pmod{q}, \quad e \in \mathcal{R}, \quad \|e\|_\infty \leq B_e. \quad (3)$$

Decryption $\lfloor b - as \rfloor_p$ recovers m when $B_e < q/(2p)$.

The divisibility $p \mid q$ is part of this interface; a native BFV scale $\lfloor q/p \rfloor$ requires an encoding-error term. For CKKS, the phase is $b - as = \hat{m} + e \pmod{q}$, with encoded target $\hat{m}$, public scale Δ_{ct} , and slot embedding σ_{slot} ; the precise bounds are in [Definition 7](#). We specify the encryption and evaluation procedures, including rescaling and key switching, with proved message and error bounds. Admissibility guarantees these bounds; it is not a public test for secret decryption noise.

5 Threshold Fully Homomorphic Encryption

Jambon replaces each unrounded partial decryption by one deterministically rounded ring element. We give its interface, construction, correctness bounds, and verification mechanism; [Section 7](#) states the security guarantees.

5.1 Core interface and correctness

The core syntax uses trusted setup; [Section 6](#) replaces it by distributed setup.

Definition 4 (ThFHE syntax).

A (t, N) -ThFHE scheme consists of ThKeyGen , ThEnc , ThEval , PartDec , and Combine . Key generation returns pk and privately delivers sk_i to party i ; encryption and evaluation use the base FHE. A response $p_i = \text{PartDec}(\text{sk}_i, \text{ct})$ does not depend on the other responders. For $|S| = t$, $\text{Combine}(\text{pk}, \text{ct}, S, (p_i)_{i \in S})$ returns an exact BFV plaintext, a finite-precision CKKS vector, or $\perp$.

Each response is indexed by the canonical label $\tau = \text{Encode}(\text{kid}, \text{pp}, \text{ct})$, an injective serialization of the key-generation identifier kid , public parameters and entire ciphertext. This fixes the mask input and response record for that key and ciphertext: aliases, retries and requester nonces cannot create fresh masks. Here kid identifies one key-generation instance; serialization does not itself prove ciphertext validity. For admitted τ , the private target is $v_\tau = m_\tau \in \mathcal{R}_p$ for BFV or $v_\tau = \hat{m}_\tau \in \mathcal{R}$ for CKKS. The CKKS profile fixes a finite-precision output space $\mathcal{Y}_\tau$, scale $\Delta_{\text{ct}, \tau}$, encoding error $\varepsilon_{\text{enc}, \tau}$, and decryption allowance $\varepsilon_{\text{dec}, \tau}$.

Definition 5 (Correctness). For y of the declared output type, define

$$\text{OutOK}_\tau(v, y) = \begin{cases} [y = v], & \text{BFV,} \\ \left[\left\| y - \frac{\sigma_{\text{slot}}(v)}{\Delta_{\text{ct}, \tau}} \right\|_\infty \leq \varepsilon_{\text{dec}, \tau} \right], & \text{CKKS.} \end{cases} \quad (4)$$

The relation is zero for $\perp$ or an output outside the specified space. The scheme is ε_{cor} -correct if the probability that any admitted τ and any $|S| = t$ fail this relation on correctly generated responses is at most ε_{cor} over the entire execution. This is a failure probability, whereas $\varepsilon_{\text{dec},\tau}$ bounds numerical error. Responder sets may be chosen from the preceding transcript; corruption remains static. For CKKS, accepted outputs approximate the intended slots within $\varepsilon_{\text{enc},\tau} + \varepsilon_{\text{dec},\tau}$; different qualified sets need not return identical vectors.

5.2 Core security model

We require IND-CPA security given any static set of fewer than t key shares, and UC simulation of strong threshold decryption, including incomplete sessions [6,5,31,12]. The core model is static semi-honest, with corruption fixed before setup and oracle queries; Section 5.5 adds malicious verification. The adversary may choose requests, responder sets and message-delivery times from the transcript observed so far; it cannot change the corrupted set. The random oracle H_0 accepts classical queries and is shared by all invocations for one key-generation instance, with separate instances across keys and a domain separate from the base FHE.

The setup resource fixes the full joint distribution of public/evaluation keys, shares, and declared retained setup state. The admission functionality runs the allowed encryption and evaluation algorithms, records the intended decryption value, and guarantees the stated phase bounds; it does not accept arbitrary ciphertexts. These core interfaces are explicit hybrids (Section B); the malicious protocol below replaces ideal admission by input proofs and public evaluation checks.

The functionality counts corrupted identities from the outset and honest identities at their first public response for τ . For each publication, the simulator supplies a bounded deterministic circuit, evaluated on $\perp$ before t identities are counted and on v_τ thereafter. At the threshold-reaching publication, the circuit is fixed before the simulator learns v_τ ; circuit evaluation, response recording and target release are atomic. The circuit has no oracle or secret-share access and is not transmitted. Thus the privacy threshold counts exposure, while a recipient reconstructs after completing setup, associating its request with the ciphertext, and receiving t prescribed responses. Malicious decryption also requires verified responses; delayed verification or delivery cannot postpone target release. Records persist across retries and cancellation, without a decryption agreement step or global abort. Under fair activation and delivery, t willing responders suffice, but corruption tolerance alone does not guarantee that many available parties.

5.3 Construction

The public parameters pp fix the base scheme, the output specification, and odd moduli $r < q$ with $r \mid q$ for the key-generation instance; put $L = q/r$. For BFV, also fix odd $p < r$ with $p \mid r$ and let $\Delta = q/p$. For CKKS, $n \geq 2$ and there is

no plaintext modulus p ; $\Delta_{\text{ct}} > 0$ is the public scale at the selected decryption level. The specified encryption and evaluation procedures output two-component ciphertexts at modulus q ; their scales may differ. Use the points and integral coefficients of (1)–(2), with the corresponding base-FHE interface in Definitions 3 and 7. Use one public mask oracle $H_0 : \{0, 1\}^* \rightarrow \mathcal{R}_q$, domain-separated from the base FHE. Figure 2 specifies the numerical algorithms. The asynchronous commands PDec and FinDec invoke PartDec and Combine, respectively, after checking the local request and response records (Section B). For CKKS, fix a public deterministic numerical decoder Num_τ from the combined phase to $\mathcal{Y}_\tau$, with error bound $\nu_\tau \geq 0$: for every $u \in \mathcal{R}$ with $u = [u]_q$,

$$\left\| \text{Num}_\tau(u) - \frac{\sigma_{\text{slot}}(u)}{D\Delta_{\text{ct}_\tau}} \right\|_\infty \leq \nu_\tau. \quad (5)$$

This public decoder applies the slot embedding and scalar division by $D\Delta_{\text{ct}_\tau}$ to the centered phase, with the specified finite-precision error; it uses neither the secret key nor a modular inverse of D .

ThKeyGen ($1^\lambda, 1^t, 1^N$)
1. Run $(\text{fhepk}, s) \leftarrow \text{FHE.KeyGen}(1^\lambda)$ and assign a key-generation identifier kid. 2. Independently sample $(f_i)_{i \in [N]} \leftarrow \text{Share}(s)$ and $(g_i)_{i \in [N]} \leftarrow \text{Share}(0)$ over $\mathcal{R}_q$. 3. Set $\text{pk} = (\text{fhepk}, \text{pp}, \text{kid})$ and privately deliver $\text{sk}_i = (f_i, g_i)$ to party i.
PartDec (sk_i, ct)
1. Parse $\text{ct} = (a, b)$ and set $\tau = \text{Encode}(\text{kid}, \text{pp}, \text{ct})$. 2. Return $p_i = \lfloor [af_i + H_0(\tau)g_i]_q \rfloor_r$.
Combine ($\text{pk}, \text{ct}, S, (p_i)_{i \in S}$)
1. Reject malformed inputs; require $S \subseteq [N]$ and $S = t$; derive τ and $(\lambda_i^S)_{i \in S}$. 2. Compute $z_S = [Db - L \sum_{i \in S} \lambda_i^S p_i]_q$. 3. For BFV, return $y_{\tau, S} = [D^{-1} \lfloor z_S \rfloor_p]_p$, with D^{-1} taken modulo p. 4. For CKKS, return $y_{\tau, S} = \text{Num}_\tau(z_S)$.
ThEnc and ThEval call the base Enc and Eval using fhepk .

Fig. 2. Jambon with BFV or CKKS decoding. All partial-decryption calls are for admitted ciphertexts.

The reusable key/zero shares and hash mask follow KLL⁺26 [17, Fig. 4]; each party stores two $\mathcal{R}_q$ elements. Jambon sends one deterministically rounded $\mathcal{R}_r$ element, sampling no private noise or per-ciphertext secret material; retries are deterministic. The multiplication Lp_i lifts a response to modulus q : different representatives of $p_i \in \mathcal{R}_r$ give the same result modulo q .

5.4 Correctness

For both profiles, define

$$M_1(S) = \sum_{i \in S} \|\lambda_i^S\|_1, \quad \mathcal{B}(S) = \frac{L-1}{2} M_1(S). \quad (6)$$

Lemma 1 (Aggregation identity). *Fix a ciphertext $\text{ct}_\tau = (a, b)$ and correctly generated Jambon responses. There are integral residues ρ_i with $\|\rho_i\|_\infty \leq (L-1)/2$ such that*

$$z_S \equiv D(b - as) - \sum_{i \in S} \lambda_i^S \rho_i \pmod{q}. \quad (7)$$

In particular,

$$\left\| \sum_{i \in S} \lambda_i^S \rho_i \right\|_\infty \leq \mathcal{B}(S).$$

The identity also holds when the unrounded values are evaluations of any polynomial P_τ of degree at most $t-1$ with $P_\tau(0) = as$.

Theorem 5.1 (Deterministic BFV correctness). *In the BFV profile, any t correctly generated responses to a B_e -admissible ciphertext reconstruct its intended plaintext provided*

$$DB_e + \mathcal{B}(S) < \frac{q}{2p}. \quad (8)$$

If this inequality holds for every $|S| = t$, correctness holds simultaneously for all responder sets and all admitted ciphertexts.

Proof (sketch). For the unrounded-response polynomial $P_\tau = aF + H_0(\tau)G$, the zero sharing gives $P_\tau(0) = as$. Balanced rounding writes each lifted response as $P_\tau(\alpha_i) + \rho_i$ modulo q ; interpolation therefore gives (7). Substituting $b - as = \Delta m + e$ leaves total error at most $DB_e + \mathcal{B}(S)$ around the scaled plaintext $D\Delta m$. Under (8), rounding recovers $Dm \bmod p$, and $D^{-1} \bmod p$ recovers m . The argument is pointwise in every ciphertext and responder set; [Section B.2](#) gives the full proofs of [aggregation](#) and [BFV recovery](#).

Remark 1 (BFV modulus choice). The BFV profile is correct under the sufficient conditions

$$q > 2pDB_e, \quad r > \frac{pM_1^{\max}}{1 - 2pDB_e/q}, \quad (9)$$

where $M_1^{\max} = \max_{|S|=t} M_1(S)$. These follow by substituting $L = q/r$ into $DB_e + (L/2)M_1^{\max} < q/(2p)$ and rearranging.

For CKKS, the same error must satisfy both a no-wrap bound and a slot-precision bound.

Corollary 5.1 (CKKS correctness). *For every admitted CKKS ciphertext, let*

$$b - as = \hat{m}_\tau + e_\tau \pmod{q}, \quad \|\hat{m}_\tau\|_\infty \leq B_{m,\tau}, \quad \|e_\tau\|_\infty \leq B_e.$$

Use $\mathcal{B}(S)$ from (6). Suppose that, for every admitted label and every $|S| = t$,

$$D(B_{m,\tau} + B_e) + \mathcal{B}(S) < q/2, \quad (10)$$

$$\frac{n}{\Delta_{\text{ct},\tau}} \left(B_e + \frac{\mathcal{B}(S)}{D} \right) + \nu_\tau \leq \varepsilon_{\text{dec},\tau}. \quad (11)$$

Then the CKKS profile is deterministically correct in the sense of Definition 5, conditional on the stated pipeline bounds. In particular, all qualified sets simultaneously satisfy the output relation of the trusted CKKS instance in Definition 11.

Proof (sketch). The no-wrap bound makes the aggregated phase an equality in $\mathcal{R}$ after centering. Divide its slot embedding by $D\Delta_{\text{ct},\tau}$, use $\|\sigma_{\text{slot}}(w)\|_\infty \leq n\|w\|_\infty$, and add the numerical error bound ν_τ . This gives exactly (11); no distributional assumption on the rounding residues is used. The full proof is in Section B.2.

These deterministic results give $\varepsilon_{\text{cor}} = 0$ conditional on the stated bounds; any failure of those bounds is charged separately.

5.5 Verifiable decryption against malicious parties

For static malicious C , $|C| < t$, the protocol checks input encryptions and public evaluations, and binds responses to DKG shares. Each sender proves the following relation separately to each other recipient in the specified ideal-ZK hybrid [8, Sect. 7.1], without an admission functionality:

$$\exists \mu, \omega : \text{ValidInput}_{\text{pp}}(\mu, \omega) = 1, \quad \text{ct} = \text{Enc}(\text{pub}, \mu; \omega). \quad (\text{V0})$$

The predicate checks message validity, permitted encryption randomness and coefficient bounds; it does not certify random sampling. Each recipient verifies the input proofs, ciphertext formats and parameters, and recomputes the complete public evaluation circuit; no recipient agreement is required. The semantic bounds must hold for every accepted witness and allowed circuit, including adversarially chosen randomness (Section 7.3).

R6 fixes the commitment table $C_i = \text{Com}(\langle f_i, g_i \rangle; \vartheta_i)$ using (D1), binding later responses to the final DKG shares. For public $p_{i,\tau}$ and $h_\tau = H_0(\tau)$, a responder proves

$$\begin{aligned} C_i &= \text{Com}(\langle f, g \rangle; \vartheta), \\ Lp_{i,\tau} &\equiv a_\tau f + h_\tau g + \rho \pmod{q}, \\ f, g &\in \mathcal{R}_q, \quad \rho \in \mathcal{R}, \quad \|\rho\|_\infty \leq (L-1)/2. \end{aligned} \quad (\text{V1})$$

For odd L , this relation proves that the response is the unique prescribed value. The commitment table, hash, encoding and exact session identifiers are checked outside the NP circuit, which takes h_τ as a constant.

First publication exposes the response and fixes one proof task per other recipient; these tasks survive cancellation. A recipient records a verified response when its message and proof acceptance match, setup has completed, and the local request is associated with the validated ciphertext; earlier arrivals are buffered. BFV combines any t verified responses unchanged. The verifiable CKKS protocol instead returns $Q_\tau(\text{Combine})$ under the exact-output CKKS condition (Z) of Section 7.3; ordinary precision correctness alone does not establish that condition. The full functionality, protocol and simulator are in Section B.3. Input proofs, evaluation records and response proofs add communication beyond the one-ring numerical response.

6 Distributed Key Generation

Jambon setup generates public and evaluation keys and independent Ring-Shamir sharings of the secret and zero. We first convert additive shares to Shamir shares against semi-honest adversaries, then give a separate six-wave protocol against static malicious adversaries, allowing abort. Both use the ring and evaluation points of Section 5.3, where the points and their pairwise differences are invertible.

6.1 Sharing the secret key and zero

The underlying key-generation protocol (the backend) returns public/evaluation keys pk_B and gives party j an additive secret-key share $s_j^{\text{add}} \in \mathcal{R}_q$. These shares satisfy $\sum_{j=1}^N s_j^{\text{add}} = s$. We use j for a dealer and i for a recipient. We apply standard Ring-Shamir resharing [21] to these shares and, independently, to zero. Its security interface specifies the joint key distribution and corrupted parties' stored state and received messages (Definitions 13 and 14). Sharing uses authenticated private channels, key-generation instance identifiers, and fresh randomness independent of the backend and the threshold-decryption random oracle H_0 .

- (1) Run the backend, retaining its public and evaluation keys.
- (2) Each dealer j independently samples $A_{j,\ell}, B_{j,\ell} \xleftarrow{\$} \mathcal{R}_q$ for $1 \leq \ell < t$ and forms

$$\begin{aligned} K_j(Y) &= s_j^{\text{add}} + \sum_{\ell=1}^{t-1} A_{j,\ell} Y^\ell, \\ Z_j(Y) &= \sum_{\ell=1}^{t-1} B_{j,\ell} Y^\ell. \end{aligned} \tag{12}$$

Send $(x_{j,i}, y_{j,i}) = (K_j(\alpha_i), Z_j(\alpha_i))$ privately to each $i \neq j$ and retain the self-contribution.

- (3) After its backend output and all contributions arrive, party i computes and stores

$$f_i = \sum_j x_{j,i}, \quad g_i = \sum_j y_{j,i}, \quad \text{sk}_i = (f_i, g_i). \quad (13)$$

The target distribution is obtained by sampling two independent coefficient families:

$$F(Y) = s + \sum_{\ell=1}^{t-1} a_\ell Y^\ell, \quad G(Y) = \sum_{\ell=1}^{t-1} b_\ell Y^\ell, \quad (14)$$

$$a_\ell, b_\ell \xleftarrow{\$} \mathcal{R}_q \text{ independently.}$$

Theorem 6.1 (Ring–Shamir DKG output distribution). *Assume the semi-honest protocol above, [Definition 13](#), and exact sampling. Conditional on the sampled backend randomness and key outputs, the aggregate polynomials follow the distribution in [\(14\)](#), and every completed party obtains their evaluations at its point. If the backend’s joint base-key distribution is the specified one, the sampled application keys and shares have exactly the output distribution of ThKeyGen. This statement concerns the sampled record, before scheduler-selected completion events; the deterministic correctness results of [Section 5.4](#) apply to every completed key under their stated conditions.*

Proof (sketch). The constant terms sum to s and zero, respectively. Independent uniform coefficients remain independent and uniform under summation, even with the backend randomness and outputs fixed. For fewer than t corrupted recipients, the simulator samples all but one honest dealer’s message vectors uniformly and determines the last from the aggregate shares and corrupted contributions. The point conditions give every consistent table equally many polynomial completions. The [full proof of the output distribution](#) is in [Section D.2](#); [Lemma 12](#) and [theorem D.1](#) give the setup-transcript simulation and composition arguments.

Sharing the secret key and zero sends $2N(N-1)nw(q)$ share bits in one parallel wave, where $w(q) = \lceil \log_2 q \rceil$. Adding the zero component to an existing key re-sharing message adds no wave. Backend costs and channel overhead remain additional; readiness requires every prescribed contribution, so setup need not complete under withholding. [Theorem D.1](#) additionally requires simulation of corrupted parties’ stored backend state.

6.2 Sharing private masks before a public opening

Suppose each dealer j knows a private mask fl_j and the backend publicly opens

$$\mathbf{m} = s + \sum_{j=1}^N \text{fl}_j \quad (15)$$

with exact equality in the sharing ring. Dealers can prepare shares before $\mathbf{m}$ is available, using

$$\tilde{K}_j(Y) = -\mathfrak{f}_j + \sum_{\ell=1}^{t-1} A_{j,\ell} Y^\ell, \quad (16)$$

and, after the opening, compute

$$f_i = \mathbf{m} + \sum_j \tilde{K}_j(\alpha_i). \quad (17)$$

Zero sharing is unchanged. The public opening cancels the masks without division by N ; [Lemma 7](#) proves the same output distribution and privacy of each honest dealer's messages to fewer than t recipients. An exact integer opening also permits reduction to q even if q does not divide the backend modulus, provided the retained public/evaluation keys are compatible. This optional modulus conversion requires an aggregate no-wrap bound; a modular identity alone does not suffice. [Section C.2](#) and [remark 7](#) give the lemma and the additional simulation obligations for early dealing.

6.3 Six-wave malicious setup

The numerical construction realizes $\mathcal{F}_{\text{DKG},\perp}^{t,N,\mathcal{D}}$ under the hypotheses below, against a static malicious coalition C with $|C| < t$. Here $\mathcal{D}$ fixes the full joint distribution of the public key, evaluation key, and secret, and pub is the base public/evaluation-key record in $\text{pk} = (\text{pub}, \text{pp}, \text{kid})$. The functionality in [Section C.5](#) samples once and permits abort after candidate disclosure. The prescribed distribution concerns this sample: adversarially selecting completed key-generation instances can bias the distribution conditioned on non-abort. Publication and readiness are scheduled separately; abort preserves past releases, and omitted messages remain pending.

The hybrid supplies consistent authenticated broadcast, authenticated private channels, separately delivered round completion notifications, and session-aware Verified Commit-and-Prove $\mathcal{F}_{\text{VCP}}$. VCP registers immutable typed values, verifies prescribed relations on those values, and privately transfers registered coordinates to designated recipients for use in later proofs. Its exact resource interface and public-key publication wrapper appear in [Section C.5.1](#). Temporary HE must satisfy IND-CPA with its complete public evaluation key, the stated auxiliary-RLWE bounds, and correctness for the entire fixed key-generation circuit, including enlarged flooding noise, level projections, and embeddings. The six-wave count is in this hybrid; a concrete realization must establish these conditions and include its resource costs and composition losses. Honest setup randomness is independent of H_0 , setup calls use disjoint domains, and corrupted choices may depend on earlier answers. This construction shares uniform private masks directly in $\mathcal{R}_q$; it does not use the optional integer-opening conversion.

[Section C.5.3](#) gives the exact protocol; its six message waves have the following roles.

- R1:** Register temporary secrets, key-generation randomness, source masks, coins, and noise rows; broadcast public-key and relinearization-key contributions with consistency proofs.
- R2:** Open the registered coins, encrypt the key-generation random tapes (private bit strings fixed in R1) with smudging, and privately transfer the registered row coordinates.
- R3:** Certify receipt status and the temporary relinearization key, then locally evaluate $\mathbf{KG}^*(\omega_1 \oplus \dots \oplus \omega_N)$.
- R4:** Open the public key record and $m = s + \sum_i \mathbf{fl}_i$; commit to sharings of the negative private masks and zero, with proofs linking them to the original masks; withhold their evaluations.
- R5:** After locally verifying every required R4 commitment and proof, privately send the committed evaluations and retain self-contributions. Corrupted dealers must therefore fix their polynomials before honest evaluations are disclosed.
- R6:** Certify receipt status and final shares; on receiving its R6 completion notification, each party queues a readiness notice independently. Delivery of that notice enables use of the stored shares.

The final proof links the shares used for decryption to the actual DKG outputs:

$$\begin{aligned}
C_i &= \text{Com}(\langle f_i, g_i \rangle; \vartheta_i), \\
C_i &= \text{Com}\left(\left\langle m + \sum_j x_{j,i}, \sum_j y_{j,i} \right\rangle; \vartheta_i\right). \tag{D1}
\end{aligned}$$

It is checked against the original ordered incoming references, including self-contributions, and adds no wave. The first delivered R6 completion notification fixes success and one commitment per identity; new source proofs, updates, and transfers are then rejected while historical receipts remain verifiable. The commitment assumptions and their additional losses belong to [Theorem 7.3](#).

[Theorem 7.2](#) proves security jointly with the application keys and shares for subsequent decryption. Its proof simulates flooded openings, then constructs the key- and zero-sharing messages from corrupted shares. Fixing R4 rows before R5 exposure accommodates malicious rushing and the release gates. [Section C](#) gives the specifications and variants; [Section D.3](#) proves the auxiliary hybrid comparisons.

Resource instantiation. The commitment/proof resource must preserve a private value when its ownership passes to a recipient. A concrete classical starting point is the multiparty extended committed oblivious transfer of Garay et al. [14], which includes precisely such transfer-and-recommit operations. [Section C.5.2](#) describes a proposed bit-level implementation of our commitment-and-proof relations and its costs. Its session, delivery, and shared-state security remains to be proved; the candidate does not establish six concrete communication rounds or a lattice-only implementation.

7 Security Analysis

We prove threshold-decryption, DKG, and joint malicious security. Corruptions are static; oracle queries are classical and separated by key-generation instance. The numerical core uses ideal admission; the malicious construction replaces it with local input proofs and public evaluation checks. The DKG hybrid chain follows; other proofs are in [Section D](#). Error terms are summarized in [Table 5](#).

7.1 Strong security of threshold decryption

Fix a corrupted set C of size $c < t$ and put $k = t - 1 - c$. For one key-generation instance, $r \mid q$ are fixed odd moduli, $L = q/r$, and q_H bounds all distinct labels queried to H_0 , including discarded candidates and environment queries. Let χ be the error distribution of uniform-secret $\text{DRLWE}_{n,q,U(\mathcal{R}_q),\chi}$ and fix a cutoff B_χ . For the monomial points of (1), with signed-monomial inverses taken in $\mathcal{R}$, define

$$\Gamma_j := \max_{\substack{T \subseteq [N], |T|=j \\ i \in [N] \setminus T}} \left\| \prod_{u \in T} (1 - \alpha_i \alpha_u^{-1}) \right\|_1, \quad \Gamma_0 = 1.$$

The mask and output shifts are bounded by $B_{\text{mask}} := k\Gamma_c B_\chi$ and $B_{\text{out}} := \Gamma_{t-1} B_e$. The error-tail and coefficient-primitive-mask exceptions are

$$\varepsilon_\chi := q_H k \Pr_{E \leftarrow \chi} [\|E\|_\infty > B_\chi], \quad \varepsilon_{\text{prim}} := N \sum_{\substack{\ell \mid q \\ \ell \text{ prime}}} \ell^{-n}.$$

For a shift bound B , the rounding-boundary probability is bounded by

$$\beta_L(B) := \begin{cases} 0, & B = 0, \\ \min\{1, (2\lceil B \rceil + 2)/L\}, & B > 0. \end{cases}$$

A mask is coefficient-primitive when $\gcd(q, g_0, \dots, g_{n-1}) = 1$; multiplication by a fresh uniform hash then makes each unrounded coefficient uniform modulo L . The term $\varepsilon_{\text{prim}}$ bounds the exceptional setup event ([Lemma 10](#)). Only the proof samples errors from χ ; Jambon samples no private decryption noise.

Theorem 7.1 (UC security for strong threshold decryption). *Use the BFV interface of [Definition 3](#), the execution model of [Section 5.2](#), and the deterministic correctness condition (8) for every qualified set. Assume uniform-secret decisional RLWE with error distribution χ . Let T_{red} bound the running times of the reductions in [Section D.1](#), including their simulation overhead. Then the Jambon wrapper realizes $\mathcal{F}_{\text{ThFHE}}^{\text{str}}[\text{ThKeyGen}, N, t, \text{bfv}, \emptyset]$ of [Definition 11](#) with*

$$\begin{aligned} \varepsilon_{\text{uc}} \leq & k \text{Adv}_{\text{RLWE}}(T_{\text{red}}, q_H) + \varepsilon_\chi + \varepsilon_{\text{prim}} \\ & + q_H N n (\beta_L(B_{\text{mask}}) + \beta_L(B_{\text{out}})). \end{aligned} \tag{18}$$

It satisfies [Definition 12](#) when this bound is negligible. The straight-line simulator receives neither s nor honest key shares and obtains the decryption value only when the functionality authorizes its release.

Proof (sketch). Following KLL⁺26’s zero-sharing strategy [17, Thm. 4.2], conditioning on corrupted zero shares leaves k independent uniform ring coefficients; for $k = 0$, the following RLWE steps are empty. Bounded proof-only errors change rounded responses only near a boundary; k RLWE hybrids replace the hashed products by uniform values, preserving the public and evaluation keys. The resulting polynomial has degree at most $t - 1$, fixed evaluations on C , and constant $a_\tau s$. Write $\text{Seen}_\tau = C \cup E_\tau$, where E_τ contains the honest identities whose responses for τ have already been publicly exposed. The first k honest evaluations of this polynomial can be sampled uniformly at identities chosen from the preceding transcript. Immediately before the next, threshold-reaching exposure, set $T = \text{Seen}_\tau$, so $|T| = t - 1$. The simulator commits an affine interpolation circuit before learning the decryption value. The functionality evaluates the circuit and releases that value atomically; thereafter the simulated polynomial differs by

$$P'_\tau(Y) - P_\tau(Y) = e_\tau \prod_{j \in T} (1 - Y \alpha_j^{-1}), \quad |T| = t - 1.$$

The universal boundary event covers every shift of norm at most B_{out} , including the adaptively selected crossing set T . Outside these events all packets agree, and pointwise correctness gives identical outputs for every qualified delivered set. Stored records preserve retries, cancellation, and incomplete executions without fairness or early disclosure. The [full proof](#) and its auxiliary lemmas are in [Section D.1](#).

Ciphertext privacy separately inherits the base FHE’s IND-CPA guarantee with its public evaluation keys ([Proposition D.1](#)). The CKKS extension permits release of the encoded target and requires bounds preventing modular wraparound and ensuring numerical precision ([Corollary D.2](#)); it does not assert approximate-output-only leakage.

7.2 Malicious distributed setup

The six-wave setup of [Section 6.3](#) requires temporary-HE correctness for its complete fixed circuit, including enlarged floods and the declared embeddings. [Section D.3](#) defines its hardness experiments, reductions, and error bounds. In particular, (45) accounts for honest samples outside their declared domains and shifts exceeding their cutoffs, without conditioning on acceptance or completion; output failures belong to ε_{HE} . Write Adv_{tmp} for the temporary-HE IND-CPA bound and Adv_{aux} for distinguishing $(a_j, -a_j u + e_{0,j})_j$ from uniform second components, with shared $u \leftarrow \chi_u$. The distinguishing-advantage bound Adv_{lift} for simulating auxiliary samples from the public key is defined in [Lemma 13](#).

Theorem 7.2 (Malicious UC DKG). *Assume the ordinary public-EVK TempHE IND-CPA and auxiliary-RLWE bounds of [Section D.3](#), HE-Corr, the embedding and sampler conditions, and the local resources of [Section 6.3](#). For every PPT static malicious adversary $\mathcal{A}$ corrupting corr with $|\text{corr}| < t$, there is*

a straight-line PPT simulator $\mathcal{S}$ such that, for every PPT environment $\mathcal{Z}$, the numerical protocol Π_6 realizes $\mathcal{F}_{\text{DKG},\perp}^{t,N,\mathcal{D}}$ with distinguishing advantage at most

$$\begin{aligned}\varepsilon_{\text{crypt}} &:= \text{Adv}_{\text{tmp}} + 2(\text{Adv}_{\text{aux}} + \text{Adv}_{\text{lift}} + \varepsilon_{\text{lift}} + \varepsilon_{\delta} + \varepsilon_{\text{evk}} + \varepsilon_{\text{in}}), \\ \varepsilon_{\text{DKG}} &\leq \varepsilon_{\text{crypt}} + \varepsilon_{\text{out}} + \varepsilon_{\text{samp}} + \varepsilon_{\text{HE}} + \varepsilon_{\text{tail}}.\end{aligned}\tag{19}$$

The six-wave claim includes abort/incomplete prefixes and (40). Comparisons retain (pub, s, F, G) jointly with the public/corrupted transcript and release history, hence allow causal continuation using the honest parties' final DKG shares. The actual simulator receives only specified candidates/releases, with no secret or honest-share query. UC security requires negligible displayed error; robustness/adaptive corruption are excluded. Concrete resources need their own UC/joint-state theorem and losses.

Proof. Auxiliary derivations appear in Section D.3.2; the simulator is specified in Section C.5.

Retain the full application record and abort/pending history from $H_0 = \text{Real}$ to $H_6 = \text{Ideal}_{\mathcal{S}_6}$. Costs are statistical distance, except H3–H4, which bounds every PPT $\mathcal{Z}$'s distinguishing advantage.

Step	Transition and justification	Cost bound
H0–H1	Couple real role samplers; record corrupted submissions without $\mathcal{A}$'s tape.	ε_{pre}
H1–H2	Reparameterize masks below; Lemma 14.	0
H2–H3	Apply HE-Corr to the real execution before suppressing honest domain failures; (51).	$\varepsilon_{\text{HE}} + \varepsilon_{\text{out}}$
H3–H4	Ciphertext replacement (50); available-state reduction in Section D.3.2.	$\varepsilon_{\text{back}}$
H4–H5	Uniform honest tape gives uniform XOR; fresh joint distribution of the key and zero sharings (40).	ε_{KG}
H5–H6	Match real and ideal events using $\mathcal{S}_6$.	$\varepsilon_{\text{post}}$

For H2, let $H = [N] \setminus C$, choose $h^* \in H$, fix s and corrupted masks, and sample

$$\begin{aligned}m &\xleftarrow{\$} \mathcal{R}_q, & \text{fl}_h &\xleftarrow{\$} \mathcal{R}_q \ (h \in H \setminus \{h^*\}), \\ \text{fl}_{h^*} &= m - s - \sum_{i \neq h^*} \text{fl}_i.\end{aligned}$$

This exact change of variables costs zero statistical distance; HE and flooding costs remain. Only the analysis uses s and the residual; with corrupted R4 rows fixed, Lemma 14 preserves the joint output distribution. After H3, visible rows need only $m, f_{\text{corr}}, g_{\text{corr}}$ and corrupted rows; Section D.3.2 lists the data determining the H4 continuation. For H5, R1 fixes corrupted tapes; one independent uniform honest tape gives $\text{KeySamp}_{\mathcal{D}}$. Fresh honest coefficients and corrupted R4 coefficients fixed before honest numerical exposure give (40), preserving the corrupted view and zero-sharing kernel.

For H6, match starts and delivered notifications of round completion to **Started** and **Done**, respectively; match readiness and failure notices to **ReleaseKey** and **ReleaseAbort**. The invariant $B_r \subseteq B_{r-1}$ and guard $H \subseteq B_{r-1}$ give the R4/R5 candidate times. Required-slot failure blocks every unreached candidate/success boundary through its immutable public slot or missing later **ok**. Remaining prefixes, delayed public-key registration, abort, partial initialization, local readiness and joint final share state match without completion conditioning. Session/key tags separate records; $\mathcal{S}_6$ is independent of $\mathcal{Z}$. Sum the table's costs for (19) and across key-generation instances. $\square$

7.3 Malicious threshold decryption

The verifiable wrapper checks ciphertexts locally and binds responses to the final DKG shares. Its ideal decryption value is the exact finite output below; the encoded-target CKKS core is a separate profile.

Semantic profiles. Let the allowed ciphertexts be the closure of (V0)-verified inputs under the checked evaluation circuits of Section B.3. All bounds below are pointwise over the declared key distribution, every accepted witness (including adversarially chosen randomness), and every allowed adaptively chosen evaluation circuit; honest-coin averages or conditioning on acceptance/success do not suffice. These are HE/policy conditions, not secret-noise tests or an admission resource. Write $u_\tau = [b_\tau - a_\tau s]_q$ and $\mathcal{B}(S)$ as in (6). For BFV require standard correctness and, for every allowed ciphertext with intended plaintext m_τ and every $|S| = t$,

$$v_\tau = \text{Dec}(s, \text{ct}_\tau) = m_\tau, \quad \|[u_\tau - (q/p)v_\tau]_q\|_\infty \leq B_e, \quad DB_e + \mathcal{B}(S) < q/(2p).$$

Thus every qualified reconstruction equals v_τ .

For CKKS use the *exact decryption* approach of Li–Micciancio [18, Sect. 6]: exact deterministic computation and decryption at fixed precision. Input and output spaces are finite with polynomial-length canonical encodings. Fix public total polynomial-time maps $Q_\tau : \mathcal{Y}_\tau \rightarrow \mathcal{V}_\tau$ and $\text{Lift}_\tau : \mathcal{V}_\tau \rightarrow \mathcal{R}$, indexed by the public output profile and independent of the mask oracle. With the existing numerical decoding map, define

$$\text{Dec}_\tau^Q(s, \text{ct}_\tau) := Q_\tau(\text{Num}_\tau([Du_\tau]_q)).$$

Require the following exact-output CKKS condition:

$$\begin{aligned} v_\tau &= \text{Dec}_\tau^Q(s, \text{ct}_\tau) = g_\mathcal{V}(\mu), \\ \|u_\tau - \text{Lift}_\tau(v_\tau)\|_\infty &\leq B_{\text{lift}}, \\ D\|u_\tau\|_\infty + \mathcal{B}(S) &< q/2, \\ Q_\tau(\text{Num}_\tau([Du_\tau - R]_q)) &= v_\tau \quad (\|R\|_\infty \leq \mathcal{B}(S)). \end{aligned} \tag{Z}$$

Here $g_\mathcal{V}$ is the declared deterministic clear computation (identity for encryption), independent of keys and encryption/evaluation coins. The last two lines

quantify over every $|S| = t$ and every integral $R \in \mathcal{R}$ in range, including non-response residues. The lift and perturbation bounds are additional threshold requirements, simultaneous with the security bound below; no message lattice or concrete CKKS instantiation is assumed.

Theorem 7.3 (Malicious UC security with strong threshold privacy).

Let $2 \leq t \leq N$. Assume the DKG hypotheses/resources of [Theorem 7.2](#), the algebraic, uniform-secret RLWE and oracle hypotheses of [Theorem 7.1](#), the semantic profile above, and local ideal ZK of [Section 5.5](#) with the [local validation procedure](#). The released decryption value is exact finite v_τ ; CKKS is conditional on (Z). Commitments are computationally hiding/binding, with fixed-length injective encoding of $\mathcal{R}_q^2$ and secure public parameters independent of the key-generation and mask randomness; shared setup requires an explicit joint guarantee.

For one key-generation instance and static malicious C , $c = |C| < t$, the six-wave DKG, R6 certification (D1), and [verifiable online wrapper](#) realize the [strong functionality](#) with

$$\varepsilon_{\text{total}} \leq \varepsilon_{\text{DKG}} + \varepsilon_{\text{online}} + \varepsilon_{\text{hide}} + \varepsilon_{\text{bind}}. \quad (20)$$

Use (19) for ε_{DKG} and the complete right-hand side of (18) for $\varepsilon_{\text{online}}$, with

$$\begin{aligned} B_{\text{out}} &= \Gamma_{t-1} B^*, & B^* &= B_e \text{ (BFV)}, & B^* &= B_{\text{lift}} \text{ (CKKS)}, \\ k &= t - 1 - c, & \varepsilon_\chi &= q_H k \Pr_{E \leftarrow \chi} [\|E\|_\infty > B_\chi], \\ \varepsilon_{\text{prim}} &= N \sum_{\substack{\ell | q \\ \ell \text{ prime}}} \ell^{-n}. \end{aligned}$$

Count all distinct queried candidates, including discarded queries and exposed proof statements, in q_H . For T_{red} bounding all reduction running times, the per-key cost of replacing honest final-share commitments by commitments to zero is $\varepsilon_{\text{hide}} \leq (N - c) \text{Adv}_{\text{Com}}^{\text{hide}}(T_{\text{red}})$. The term $\varepsilon_{\text{bind}}$ bounds the probability of valid openings of one registered commitment to distinct share pairs in the DKG share-binding proofs or proofs of correct partial decryption. Ideal ZK is exact; input certification counts toward its invocation budget. No admission functionality is supplied, and local-validation simulation costs zero in this hybrid. Honest randomness outside the permitted set gives the same $\perp_{\text{input}}$ in both worlds without resampling; honest-encryption completeness must account for this failure. Probabilistic profiles additionally incur their joint failure cost over the compared executions and adaptive records. Add the established realization errors for ZK, VCP, channels, the resource providing notifications of round completion, and commitment parameters, and sum the bounds over key-generation instances. UC realization requires the total to be negligible for the declared execution budget.

Proof (sketch). Every accepted ciphertext is a verified input encryption or an allowed evaluation of such inputs, including inputs encrypted with adversarially chosen randomness. Couple honest inputs and local failures; the simulator

emulates their proof notices without witnesses and forwards only actual corrupted witness submissions. Source binding and (V1) fix corrupted responses to their prescribed values. Hide honest commitments, replace DKG jointly with its outputs, then apply the residual RLWE hybrids; uniformity is conditioned on the permitted algebraic view, not the real transcript. When v_τ is released, the simulator uses $(q/p)v_\tau$ for BFV and $\text{Lift}_\tau(v_\tau)$ for CKKS; the latter changes the output-shift bound to $\Gamma_{t-1}B_{\text{lift}}$. First-query boundary events cover proofs and responders chosen from the transcript, without conditioning on future validation or delivery. Public response exposure counts toward release; recipients record verified responses once all local conditions and matching receipts are satisfied. Coupled packets give the same unquantized aggregate, and BFV correctness or (Z) forces the output to equal v_τ for every qualified set. The full proof in [Section D.4](#) gives the straight-line simulation for pending requests, immutable proof sessions, cancellation, and incomplete prefixes.

8 Evaluation

We compare numerical partial-decryption sizes with KLL⁺26’s strong construction, using our finite-query refinement at matched ring degree, threshold, plaintext modulus, ciphertext-error bound, and statistical budget. This is a parameter study, without implementation, runtime, or concrete lattice-security claims.

8.1 Metric and parameter selection

A Jambon partial decryption contains one element of $\mathcal{R}_r$; the strong KLL⁺26 response contains one flooded element of $\mathcal{R}_{q_K}$ [17, Fig. 4]. Writing $w(u) = \lceil \log_2 u \rceil$ for fixed-width coefficient serialization, we measure

$$C_{\text{part}} = nw(r), \quad C_{\text{qualified}} = tnw(r), \quad C_{\text{ct}} = 2nw(q), \quad (21)$$

counting t response objects and one two-component ciphertext once each. The saved fraction relative to KLL⁺26 is $1 - w(r)/w(q_K)$. Authentication, commitments, proofs, and additional network copies are outside this response-size metric.

We fix the analytical inputs

$$\begin{aligned} n &= 2^{16}, & p &= 65537, & B_e &= 2^{30}, \\ B_\chi &= 64, & q_H &= 2^{32}, & \kappa &= 40. \end{aligned} \quad (22)$$

Here B_e bounds accepted ciphertext errors, q_H bounds all candidate labels in the security theorem, and $2^{-\kappa}$ covers its rounding-boundary or flooding transition. The proof-only χ is centered binomial on $[-64, 64]$; Jambon samples no decryption noise. Deployment additionally requires a workload, validation of base-FHE and mask-RLWE security, and bounds on the remaining terms in (18).

Put $D = 2^{\lceil \log_2 t \rceil}$. For the power-of-two party counts studied here, Jambon uses spaced monomials with $N'_j = N$; KLL⁺26 uses the geometry $N'_K =$

$2^{\lceil \log_2 \max\{N, 2t\} \rceil} \leq 2n$ chosen in [17, Cor. 5.4, Thm. 5.5]. Both N'_K and D double just above $t = N/2$, causing the plotted jump (Section E.3); Section E.4 also checks a common geometry. Write M_{cert} and M_K for the reconstruction bounds, $\widehat{I}_j$ for the bound on the coefficient associated with zero, and G_K for the bound covering all interpolation coefficients (Section E.1, including $t = 2$). We use the following exact choices.

Jambon. Choose the least odd $L > 1$ satisfying

$$\begin{aligned} A &= B_\chi \max_{0 \leq c < t} (t - 1 - c) \widehat{I}_c, \quad B = B_e \widehat{I}_{t-1}, \\ J(x) &= \begin{cases} 0, & x = 0, \\ 2\lceil x \rceil + 2, & x > 0, \end{cases} \quad L \geq 2^\kappa q_H N n [J(A) + J(B)]. \end{aligned} \quad (23)$$

The maximum covers every corruption count $c < t$, including the pre-release mask term. Set $q = rL$, where r is the least odd multiple of p satisfying the BFV correctness condition

$$\frac{r}{p} > \left(1 - \frac{1}{L}\right) M_{\text{cert}} + \frac{2DB_e}{L}. \quad (24)$$

This order of selection separates the statistical gap from the transmitted modulus.

Strong KLL⁺26 baseline. Use Figure 4's small RLWE error and fresh independent flooding, with corruption fixed before queries and one invocation per party per ciphertext over at most q_H ciphertexts. Only responses outside the first $t - 1$ exposed identities (including corruptions) require a flooding shift: $H_t = (N - t + 1)q_H$ replacements. Exact integer-cube overlap refines the source's bound (Lemma 16):

$$\begin{aligned} E_K &= DB_e \widehat{I}_{t-1}(N'_K) + B_\chi [(t - 1)G_K + D], \\ B_{\text{sm}} &= \max \left\{ 1, \left\lceil \frac{2^\kappa H_t n E_K - 1}{2} \right\rceil \right\}, \\ q_K &> 2p [(B_e + p/4)D^2 + M_K(DB_\chi + B_{\text{sm}})], \quad q_K > 2D^2. \end{aligned} \quad (25)$$

Choose the least admissible odd q_K . This retains the full correctness condition, including $p/4$ [17, Thm. 4.1], while bounding ciphertext and RLWE errors separately. This common allocation does not identify the security notions or assert optimality of either modulus. Integer choices use outward-rounded bounds; deployment must also impose workload-specific modulus constraints.

8.2 Partial-decryption sizes

Table 2 reports 64.8–87.2% savings, using three threshold fractions for each $N \in \{256, 512, 1024\}$. At $n = 2^{16}$, width w gives $w/128$ MiB (1 MiB is 2^{20}

Table 2. Analytical BFV response sizes under (22). Widths are bits per coefficient; response sizes are MiB per party, rounded to two decimals. Saved uses the exact widths. The strong KLL⁺26 baseline uses the finite-query refinement; geometries are selected separately.

N	t	Width (bits)		Response (MiB)		Saved (%)
		$w(r)$	$w(q_K)$	KLL ⁺ 26	Jambon	
256	85	145	458	3.58	1.13	68.3
	128	134	479	3.74	1.05	72.0
	170	108	784	6.12	0.84	86.2
512	170	267	786	6.14	2.09	66.0
	256	243	826	6.45	1.90	70.6
	341	188	1436	11.22	1.47	86.9
1024	341	506	1438	11.23	3.95	64.8
	512	460	1517	11.85	3.59	69.7
	682	349	2736	21.38	2.73	87.2

bytes): $(N, t) = (1024, 512)$ reduces 11.85 to 3.59 MiB. The same ratio applies to t responses.

The ciphertext moduli $q_J := q = rL$ and q_K are selected separately under each scheme’s sufficient conditions. At $N = 1024$, their ciphertext coefficient widths are respectively (957, 1024, 960) and (1438, 1517, 2736) for the three displayed profiles. At full threshold they are 627 and 2884 bits (Section E.3). Counting one ciphertext once and t numerical responses gives savings of 64.6%, 69.5% and 87.2% for the three profiles. This excludes verification and setup; network traffic depends on dissemination (Section E.6).

Every-threshold guarantee. We also check every integer $2 \leq t \leq N$. The minimum savings for $N = 256, 512, 1024$ are, respectively, 59/89 at $t = 36$, 318/505 at $t = 52$, and 163/270 at $t = 88$. The minimum across this finite grid is $163/270 \approx 60.37\%$, attained at $(N, t) = (1024, 88)$. Directed-interval enclosures certify the integer bit widths, and exact rational comparisons identify the minima. Section E.4 supplies the verification method, a worked example, extended grids, and statistical-budget sensitivity.

Dependence on the party count. The all-threshold minimum decreases from 66.29% at $N = 256$ to 54.31% at $N = 32768$ (Table 3); the 60% floor is claimed only for the main three party counts. The one-third profile instead approaches 63.17% along admissibly growing ring degrees (Proposition E.2). This limit does not bound all thresholds, and the finite sweep proves no asymptotic value for the minimum.

Sensitivity to reconstruction bounds. Both budgets use upper bounds rather than exact coefficient maxima. Table 4 applies the same hypothetical d -bit tightening to each reconstruction bound, with necessary floor D , and recomputes every

Table 3. Scaling with N , with the inputs of (22) and fixed $n = 65536$. The bit widths are at a minimizing threshold t^* ; the last column shows the one-third profile. Percentages are rounded. Larger party counts are analytical extensions.

N	t^*	$w(r)$	$w(q_K)$	Minimum (%)	At $\lfloor N/3 \rfloor$ (%)
256	36	120	356	66.29	68.34
512	52	187	505	62.97	66.03
1024	88	321	810	60.37	64.81
2048	106	459	1103	58.39	64.02
4096	162	751	1743	56.91	63.61
8192	186	1007	2279	55.81	63.39
16384	220	1353	3004	54.96	63.28
32768	266	1824	3992	54.31	63.22

Table 4. Counterfactual reconstruction-bound sensitivity: minimum savings over all thresholds. $d = 0$ uses the proved bounds; $d > 0$ assumes unproved tighter bounds.

N	$d = 0$	$d = 8$	$d = 16$	$d = 32$
256	66.29	67.71	69.15	72.16
512	62.97	63.92	64.80	66.61
1024	60.37	60.96	61.54	62.67

threshold with simulation bounds fixed. In the dominant-term range, this removes d bits from each response width, preserving their difference and increasing the saved fraction. The tighter bounds remain unproved (Section E.5).

8.3 Communication scope

For s responders, the malicious protocol runs $E = s(N - 1)$ external sender–recipient proof sessions, excluded above; self-receipt needs none. Let b_V count proof-session bytes beyond the numerical response, including repeated public inputs, and b_{off} count offer and authentication bytes. With $b_r = nw(r)/8$, the complete response dissemination and amortized setup budget is

$$C_{\text{mal}} = E(b_r + b_V + b_{\text{off}}) + K_{\text{setup}}/J_{\text{dec}}. \quad (26)$$

Here K_{setup} includes all setup traffic and J_{dec} counts decryptions sharing it. Input proofs, ciphertexts, and evaluation circuits are separate; nonuniform session costs require a sum. Thus this total is below the native *numerical* response transport $Enw(q_K)/8$ precisely when

$$b_V + b_{\text{off}} + \frac{K_{\text{setup}}}{J_{\text{dec}}E} < \frac{n(w(q_K) - w(r))}{8}. \quad (27)$$

For $(N, t) = (1024, 512)$ and $s = t$, the right side is 8456 KiB per external session. Malicious comparisons must include both proof systems, input validation, and

setup at matched workloads and computational security. Reconstruction needs t verified responses; DKG can wait or abort. These results do not rank other compressed-response schemes. [Sections E.6](#) and [E.7](#) detail transport costs and CKKS precision, lift and stability requirements.

References

1. Avanzi, R., Bos, J., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schanck, J.M., Schwabe, P., Seiler, G., Stehlé, D.: CRYSTALS-Kyber: Algorithm specifications and supporting documentation (version 3.02). Submission to round 3 of the NIST Post-Quantum Cryptography Standardization Project (Aug 2021), <https://pq-crystals.org/kyber/data/kyber-specification-round3-20210804.pdf>, specification document dated August 4, 2021
2. Boneh, D., Gennaro, R., Goldfeder, S., Jain, A., Kim, S., Rasmussen, P.M., Sahai, A.: Threshold cryptosystems from threshold fully homomorphic encryption. In: Annual International Cryptology Conference. pp. 565–596. Springer (2018)
3. Boneh, D., Shoup, V.: A graduate course in applied cryptography (2023), <https://toc.cryptobook.us/book.pdf>, version 0.6, January 2023
4. Brakerski, Z.: Fully homomorphic encryption without modulus switching from classical gapsvp. In: Annual cryptology conference. pp. 868–886. Springer (2012)
5. Brakerski, Z., Friedman, O., Marmor, A., Mutzari, D., Spiizer, Y., Trieu, N.: Threshold fhe with efficient asynchronous decryption. Cryptology ePrint Archive (2025)
6. Canetti, R.: Universally composable security: A new paradigm for cryptographic protocols. In: Proceedings 42nd IEEE symposium on foundations of computer science. pp. 136–145. IEEE (2001)
7. Canetti, R., Jain, A., Scafuro, A.: Practical uc security with a global random oracle. In: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security. pp. 597–608 (2014)
8. Canetti, R., Lindell, Y., Ostrovsky, R., Sahai, A.: Universally composable two-party and multi-party secure computation. Cryptology ePrint Archive, Report 2002/140 (2003), <https://eprint.iacr.org/2002/140>, full version, 14 July 2003
9. Cheon, J.H., Kim, A., Kim, M., Song, Y.: Homomorphic encryption for arithmetic of approximate numbers. In: International conference on the theory and application of cryptology and information security. pp. 409–437. Springer (2017)
10. Cini, V., Lai, R.W., Woo, I.K.: Pilvi: Lattice threshold pke with small decryption shares and improved security. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 539–569. Springer (2025)
11. Damgård, I., Kolby, S., Orlandi, C., Pawlak, S.: Efficient mpc-based modulus conversion for threshold fhe decryption. Cryptology ePrint Archive (2026)
12. Ebrahimi, E., Yadav, A.: Strongly secure universal thresholdizer. In: ASIACRYPT (2024)
13. Fan, J., Vercauteren, F.: Somewhat practical fully homomorphic encryption. Cryptology ePrint Archive (2012)
14. Garay, J.A., MacKenzie, P., Yang, K.: Efficient and universally composable committed oblivious transfer and applications. In: Theory of Cryptography Conference. pp. 297–316. Springer (2004)
15. Gennaro, R., Jarecki, S., Krawczyk, H., Rabin, T.: Secure distributed key generation for discrete-log based cryptosystems. In: International Conference on the Theory and Applications of Cryptographic Techniques. pp. 295–310. Springer (1999)
16. Hu, Z., Liang, H., Wang, X., Xie, X., Yang, K., Yu, Y., Zhang, W.: Ajax: Fast threshold fully homomorphic encryption without noise flooding. In: 35th USENIX Security Symposium (USENIX Security 26) (2026)

17. Kim, W., Lee, C., Lee, J., Passelègue, A., Stehlé, D.: Asynchronous lagrange-based threshold fhe with smaller modulus overhead. *Cryptology ePrint Archive* (2026)
18. Li, B., Micciancio, D.: On the security of homomorphic encryption on approximate numbers. In: *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. pp. 648–677. Springer (2021)
19. Lyubashevsky, V., Peikert, C., Regev, O.: On ideal lattices and learning with errors over rings (2012), <https://cims.nyu.edu/~regev/papers/ideal-lwe.pdf>, full version, April 24, 2012
20. Min, S., Hanrot, G., Park, J.H., Passelègue, A., Stehlé, D.: Distributed key generation for efficient threshold-ckks. *Cryptology ePrint Archive* (2025)
21. Mouchet, C., Bertrand, E., Hubaux, J.P.: An efficient threshold access-structure for rlwe-based multiparty homomorphic encryption. *Journal of Cryptology* **36**(2), 10 (2023)
22. Mouchet, C., Troncoso-Pastoriza, J., Bossuat, J.P., Hubaux, J.P.: Multiparty homomorphic encryption from ring-learning-with-errors. *Proceedings on Privacy Enhancing Technologies* **2021**(4), 291–311 (2021)
23. Okada, H., Takagi, T.: Low communication threshold fhe from standard (module-) lwe. In: *International Conference on the Theory and Application of Cryptology and Information Security*. pp. 165–198. Springer (2025)
24. Okada, H., Takagi, T.: Threshold fhe with short decryption shares without a semi-trusted server. In: *Australasian Conference on Information Security and Privacy*. pp. 166–193. Springer (2026)
25. Park, J.H., Passelègue, A., Stehlé, D.: Thed: Threshold dilithium from fhe. *Cryptology ePrint Archive* (2026)
26. Passelègue, A., Stehlé, D.: Low communication threshold fully homomorphic encryption. In: *International conference on the theory and application of cryptology and information security*. pp. 297–329. Springer (2024)
27. Shamir, A.: How to share a secret. *Communications of the ACM* **22**(11), 612–613 (1979)
28. Sinha, S., Patranabis, S., Mukhopadhyay, D.: Efficient quantum-safe distributed prf and applications: Playing dice in a quantum world. In: *International Conference on Applied Cryptography and Network Security*. pp. 47–78. Springer (2024)
29. Sinha, S., Patranabis, S., Mukhopadhyay, D.: Equadise: Efficient quantum-safe adaptive distributed symmetric-key encryption. *Cryptology ePrint Archive* (2026)
30. Zhang, Y., Zhou, Y., Han, S., Liu, S., Huang, X.: Adaptively secure threshold decryption from lwe with polynomial modulus. *Cryptology ePrint Archive* (2026)
31. Zyskind, G., Zarchy, D., Leibovich, M., Peikert, C.: High-throughput universally composable threshold fhe decryption. In: *Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security*. pp. 2339–2353 (2025)

A Auxiliary preliminaries

We record the elementary ring facts used in the proofs and the complete CKKS interface.

Lemma 2 (Coefficient-norm product bound). *For $u, v \in \mathcal{R}$, $\|uv\|_\infty \leq \|u\|_1 \|v\|_\infty$.*

Proof. Each coefficient of the negacyclic product is a sum of signed products $u_j v_k$, with every coefficient of u occurring once. The triangle inequality gives the bound. $\square$

Lemma 3 (Invertible evaluation matrices). *For $A_{i,j} = \alpha_i^j$, $i \in [N]$, $1 \leq j < t$, every $(t-1)$ -row restriction A_T and every t -row restriction $V_S = (\mathbf{1}_N \mid A)_S$ are invertible. Restrictions of A to at most $t-1$ rows are surjective.*

Proof. For indices in increasing order,

$$\det A_T = \left(\prod_{i \in T} \alpha_i \right) \prod_{\substack{i,j \in T \\ i < j}} (\alpha_j - \alpha_i), \quad \det V_S = \prod_{\substack{i,j \in S \\ i < j}} (\alpha_j - \alpha_i).$$

Both determinants are products of units. A smaller row restriction extends to an invertible block and is surjective after coordinate projection. $\square$

Lemma 4 (Correctness and privacy). *Reconstruction is correct, and any $|T| < t$ shares are uniform over $\mathcal{R}_q^{|T|}$ independently of s .*

Proof. Interpolation through V_S yields $\sum_{i \in S} \ell_i^S F(\alpha_i) = F(0) = s$. For privacy, write $f_T = s\mathbf{1}_T + A_T r$ and apply the surjectivity in Lemma 3 to uniform r . $\square$

Lemma 5 (Monomial differences are units). *If q is odd, any two distinct monomials in $\{1, x, \dots, x^{2n-1}\}$ have a unit difference in $\mathcal{R}_q$.*

Proof. For $0 < e < 2n$, set $h = 2n/\gcd(e, 2n)$ and $z = x^e$. Then h is even, $z^{h/2} = -1$, and $(1-z) \sum_{j=0}^{h/2-1} z^j = 2$. Thus $1-z$ is invertible modulo odd q ; multiplication by a monomial gives the claim. $\square$

Lemma 6 (Balanced rounding). *Each output has exactly L^n preimages, so modulus switching preserves uniformity. For odd L , every $v \in \mathcal{R}_q$ satisfies*

$$L[v]_r \equiv v + \rho \pmod{q}, \quad \rho \in \mathcal{R}, \quad \|\rho\|_\infty \leq (L-1)/2.$$

Proof. Translation by L on $\mathbb{Z}_q$ maps the preimages of one rounded value bijectively to those of the next, so every scalar output has L preimages. Taking the product over n coefficients gives L^n . For odd L , set $\rho = L\lfloor [v]_q/L \rfloor - [v]_q$; each coefficient lies in $\{-(L-1)/2, \dots, (L-1)/2\}$. $\square$

A.1 LWR terminology and the CKKS interface

The main security theorem uses RLWE; the following definition only identifies the rounded-product form.

Definition 6 (Learning with rounding). For a fixed uniform secret $s \xleftarrow{\$} \mathcal{R}_q$, a ring-LWR sample is $(a, \lfloor as \rfloor_r)$, where $a \xleftarrow{\$} \mathcal{R}_q$. Decisional ring-LWR asks to distinguish polynomially many such samples from (a, u) with independent $u \xleftarrow{\$} \mathcal{R}_r$. Ordinary LWR replaces the ring product by $\langle \mathbf{a}, \mathbf{s} \rangle$ for $\mathbf{a}, \mathbf{s} \in \mathbb{Z}_q^d$ and has scalar output in $\mathbb{Z}_r$.

The core approximate-decoding profile fixes both an encoded target and its public error budget. The malicious exact-output profile adds the separate semantic conditions of [Section 7.3](#).

Definition 7 (Approximate noisy-linear FHE (CKKS)). Let $\Delta_{\text{ct}} > 0$ be the public scale at the ciphertext's current level, and let $\sigma_{\text{slot}}: \mathcal{R} \rightarrow \mathbb{C}^{n/2}$ be the canonical embedding restricted to one coordinate from each conjugate pair. For public bounds $B_m, B_e, \varepsilon_{\text{enc}}$, a ciphertext is admissible for an intended vector $\mathbf{z} \in \mathbb{C}^{n/2}$ if there exist an encoded polynomial $\hat{m} \in \mathcal{R}$ and an error $e \in \mathcal{R}$ satisfying

$$\begin{aligned} b - as &= \hat{m} + e \pmod{q}, \quad \|\hat{m}\|_\infty \leq B_m, \quad \|e\|_\infty \leq B_e, \\ \left\| \frac{\sigma_{\text{slot}}(\hat{m})}{\Delta_{\text{ct}}} - \mathbf{z} \right\|_\infty &\leq \varepsilon_{\text{enc}}. \end{aligned}$$

If $B_m + B_e < q/2$, then decoding returns

$$\text{Dec}(s, \text{ct}) = \frac{\sigma_{\text{slot}}(\lfloor b - as \rfloor_q)}{\Delta_{\text{ct}}}, \quad \|\text{Dec}(s, \text{ct}) - \mathbf{z}\|_\infty \leq \varepsilon_{\text{enc}} + \frac{nB_e}{\Delta_{\text{ct}}}.$$

Here $\|\sigma_{\text{slot}}(e)\|_\infty \leq n\|e\|_\infty$ converts the coefficient-error bound to a slot-error bound. CKKS has no plaintext modulus p and does not require $\Delta_{\text{ct}} = q/p$; unlike BFV, its decoded output retains the phase error.

B Full threshold-decryption interfaces and proofs

This appendix specifies the execution interfaces and proves correctness.

B.1 Formal security interfaces

We require ciphertext privacy and UC security for strong threshold decryption, including incomplete sessions and recipient-specific message delivery [\[6, 5, 31, 12\]](#). KLL⁺26 counts queried identities separately for each ciphertext [\[17, Def. 2.10\]](#). Plaintext release counts published responses for a canonical ciphertext; output readiness counts responses received by a particular recipient.

Execution model. A static semi-honest adversary $\mathcal{A}$ fixes $C \subseteq [N]$, $|C| < t$, before setup and oracle queries. Corrupted parties follow the prescribed computation when participating; their states are exposed, and the adversary controls the order and delay of network deliveries. Requests, concurrent invocations, and responder sets may depend on the preceding transcript; the corrupted set remains fixed. We use authenticated public channels $\mathcal{F}_{\text{auth}}$ and a random oracle $\mathcal{F}_{\text{RO}}$ for H_0 with classical queries, restricted to one key-generation instance and domain-separated from the base FHE. The key identifier contains the UC session identifier. Commands referencing unknown or wrong-session records fail type checking without changing state. Invocations under one key share label, publication, and response tables. A local identifier ssid names a party's decryption context within UC session sid , without changing the input to H_0 .

Definition 8 (Ciphertext privacy). For every static C with $|C| < t$, IND-CPA security holds when $(\text{sk}_i)_{i \in C}$ is additionally given to the adversary. The two challenge messages have identical public encoding parameters, and $|\Pr[\beta' = \beta] - 1/2|$ is negligible. This experiment has no honest partial-decryption oracle; the following functionality specifies the leakage of such responses.

Key and admission interfaces. The hybrid resources in [Definition 12](#) are specified below; as in [\[31, Sect. 3.1\]](#), the key resource is parameterized by a key-generation algorithm.

Definition 9 (Key interface).

The resource $\mathcal{F}_{\text{KG}}[\text{KGen}, \xi]$ specifies a joint sampler for application keys and stored setup state, with key marginal exactly KGen , including public/evaluation-key and share correlations. The fixed specification ξ determines setup events, public disclosures, successive disclosures ξ_C to the corrupted set, and local readiness conditions. The first setup request samples the record once; only coins and algebraic outputs are presampled, not future scheduling decisions. Repeats reuse this record; keys and shares cannot be altered, replaced, or resampled. Real parties store their shares privately; the ideal service stores the joint record and reveals to $\mathcal{S}$ only the items and events authorized by ξ . For $\xi = \emptyset$, initialization reveals exactly (pk, sk_C) to $\mathcal{S}$ and queues independently delayed key-ready notices. Otherwise each event releases only its enabled items, and readiness requires the specified conditions. There is no phase-specific abort command, and no claim about the sampled distribution conditional on delivery. For $\text{KGen} = \text{ThKeyGen}$ of [Figure 2](#), the application shares are the independent sharings $\text{Share}(s)$ and $\text{Share}(0)$. The trusted setup in this section uses $\xi = \emptyset$. A DKG replacement must realize the specified pair (KGen, ξ) or justify a secure conversion to this interface.

Definition 10 (Admission interface).

The resource $\mathcal{F}_{\text{adm}}$ records the specified Enc/Eval procedure, with public ciphertext bytes, identifiers, input dependencies, and output parameters. Encryption uses fresh base-scheme coins; evaluation uses registered inputs and allowed circuits. An honest input and its coins are not disclosed by registration; the

same resource is used on both sides of the realization statement. There is no arbitrary-ciphertext import, and aliases of identical canonical ciphertexts use one record. An admitted record contains the private target v_τ of Definition 5: the BFV plaintext m_τ or CKKS encoded target $\hat{m}_\tau$ fixed by the computation and encoding. Only the ideal service receives this target before authorized release; it is not adversarially asserted or chosen after seeing the secret phase. The specified procedure guarantees the noisy-linear equation and rejects inconsistent records; this is not a public test of secret-dependent ciphertext noise. Realizing this admission interface is separate from threshold decryption, as is standard for restricted FHE decryption [5,31].

Atomic publication. The simulator submits a deterministic Boolean circuit $M : \{\perp\} \cup \mathcal{V}_\tau \rightarrow \mathcal{R}_r$, where $\mathcal{V}_\tau$ is the finite, publicly encoded target space. Public parameters bound its size polynomially, sufficiently for interpolation and rounding. Before accessing the target, the functionality checks syntax and size; canonical reduction of the output makes evaluation total. The circuit has no oracle gates, share input, or secret-phase input and is not transmitted. At the first publication reaching t counted identities, $\mathcal{S}$ fixes M before learning v_τ ; response exposure, counting, and target release form one transition.

Shared request and state rules. The core and verifiable interfaces use the following rules, with the additional validation checks specified below for the latter. For a key-ready party, $\text{Bind}(\text{ssid}, \tau)$ immutably associates a local context with an admitted ciphertext and returns its bound notice. In an active context, PDec replays an existing response or creates a pending ticket $e = (\tau, i, \text{serial})$ with fresh serial; duplicates coalesce. An unready FinDec returns $\perp_{\text{notready}}$ without closing the context. Cancel stops new work and detaches the context's requests; a ticket is canceled only if no live context needs it. Bindings, published responses, counts, queues, received records, and outputs persist across retries and contexts, including canceled contexts, which may still receive and finalize existing records. Silence remains pending; there is no global decryption abort or timeout.

Definition 11 (Strong asynchronous functionality).

For an admissible (KGen, ξ) , $t \leq N$, and $\Pi \in \{\text{bfv}, \text{ckks}\}$, functionality $\mathcal{F}_{\text{ThFHE}}^{\text{str}}[\text{KGen}, N, t, \Pi, \xi]$ uses the key/admission interfaces and atomic publication rule above. Commands carry (sid, kid) ; invalid requests have no effect and repeats reuse records.

- (1) **Setup and admission.** On the first KeyGen , sample once; release keys, corrupted views, and individual readiness notices according to $\mathcal{F}_{\text{KG}}[\text{KGen}, \xi]$. After public-key release, forward Enc/Eval to admission. For each admitted $(\tau, \text{ct}_\tau, v_\tau)$, initialize $\text{Seen}_\tau = C$ and empty response, receipt, ticket, and output records; disclose only public data. Aliases share these records.
- (2) **Requests.** Apply the shared request rules; notify $\mathcal{S}$ of each pending honest ticket.
- (3) **Publication.** On $\text{CommitPart}(e, T, M)$ from $\mathcal{S}$, require a live unpublished ticket, $T = \text{Seen}_\tau$, and the public circuit checks before target access. Put

$w = \perp$ when $|T \cup \{i\}| < t$, and $w = v_\tau$ otherwise. Atomically expose and store $(\tau, i, M(w))$, complete e , add i to Seen_τ , and queue the immutable packet for every recipient, with immediate self-receipt. At the first crossing also send $(\text{Open}, \tau, v_\tau)$ to $\mathcal{S}$. Corrupted publications store only the prescribed response on the actual sk_i ; their identities are already counted.

- (4) **Delivery and output.** On $\text{ReleasePart}(\tau, i, j)$, deliver an existing queued packet to key-ready, locally bound j and insert i into $D_{\tau, j}$ once. Deliveries are independent. On $\text{FinDec}(\text{ssid}, S)$ for bound τ , require $|S| = t$ and $S \subseteq D_{\tau, j}$; otherwise return $\perp_{\text{notready}}$ without closing the context. Return m_τ for BFV; for CKKS, recompute $y = \text{Combine}(\text{pk}, \text{ct}_\tau, S, (p_{i, \tau})_{i \in S})$ and return y if $\text{OutOK}_\tau(v_\tau, y) = 1$, otherwise $\perp_{\text{badtr}}$. Record the output under (τ, S) for reuse and notify $\mathcal{S}$; no final approval is required.
- (5) **Cancellation.** Apply [the shared state rules](#).

Real wrapper. The wrapper Π_{ThFHE} implements these state and delivery rules. A live request runs PartDec and publishes atomically, without a proposed responder set; the sender retains its packet immediately. Other recipients store authenticated, correctly typed packets, buffering arrivals until key readiness and local binding. Retries reuse bytes; KLL⁺26 instead resamples flooding on each invocation [17, Sect. 4.2]. A ready FinDec runs Combine locally. The private-target CKKS check occurs only in the ideal service; correctness and simulation bound its failure. $\text{Bind}/\text{Cancel}$ require no network agreement, and authentication does not prove correct computation of a response.

Definition 12 (UC security for strong threshold decryption). Fix an instance $(\text{KGen}, N, t, \Pi, \xi)$ and let $\mathcal{G} = (\mathcal{F}_{\text{auth}}, \mathcal{F}_{\text{RO}}, \mathcal{F}_{\text{adm}})$ with $\mathcal{F}_{\text{adm}}$ as in [Definition 10](#). For every PPT adversary $\mathcal{A}$ in the stated model, there exists a PPT simulator $\mathcal{S}$ such that, for every PPT environment $\mathcal{Z}$ and polynomial-length auxiliary input z ,

$$\left| \Pr [\text{EXEC}_{\Pi_{\text{ThFHE}}, \mathcal{A}, \mathcal{Z}}^{\mathcal{F}_{\text{KG}}[\text{KGen}, \xi], \mathcal{G}}(1^\lambda, z) = 1] - \Pr [\text{EXEC}_{\mathcal{F}_{\text{ThFHE}}^{\text{str}}[\text{KGen}, N, t, \Pi, \xi], \mathcal{S}, \mathcal{Z}}^{\mathcal{G}}(1^\lambda, z) = 1] \right| \leq \varepsilon_{\text{uc}}(\lambda).$$

Negligible ε_{uc} gives UC realization in this hybrid [6]. The comparison includes all packets and honest outputs under interleaving, cancellation, alias retries, recipient-specific delivery, and more than t responses.

Remark 2 (Asynchrony and composition). No responder-set agreement or common delivery order is required. Once any t prescribed packets reach a bound recipient, its next final-decryption activation requires no further approval. Output requires fair local activation/delivery, t willing responders, and a recipient request; $|C| < t$ alone does not ensure $N - |C| \geq t$. Malicious security additionally requires the verifiable protocol and proof below. Replacing $\mathcal{F}_{\text{KG}}$ by DKG requires the same joint key distribution, compatible setup delivery/failure semantics, and a composable simulator for its public transcript [6]. The oracle is

not shared with unrelated protocols; quantum-superposition queries lie outside the theorem [7]. In Section 7.1, q_H counts all distinct queried candidate labels, including discarded hash queries; local invocation identifiers do not increase the number of distinct mask inputs.

B.2 Full correctness proofs

B.2.1 Proof of Lemma 1.

Proof. Set $P_\tau(Y) = aF(Y) + H_0(\tau)G(Y)$, where $F(0) = s$ and $G(0) = 0$. By Lemma 6,

$$Lp_i \equiv P_\tau(\alpha_i) + \rho_i \pmod{q}.$$

Changing the lift of p_i by a multiple of r changes Lp_i only by a multiple of q . Interpolation gives $\sum_i \lambda_i^S P_\tau(\alpha_i) = DP_\tau(0) = Das \pmod{q}$, and substitution in z_S yields (7). The same argument applies to any such P_τ ; Lemma 2 gives the norm bound. $\square$

B.2.2 Proof of Theorem 5.1.

Proof. Substitute $b - as = \Delta m + e \pmod{q}$ into (7) to obtain

$$z_S \equiv D\Delta m + e_S \pmod{q}, \quad e_S = De - \sum_{i \in S} \lambda_i^S \rho_i. \quad (28)$$

The norm bound in Lemma 1 implies $\|e_S\|_\infty < q/(2p)$. The p scaled plaintext representatives Δj in $\mathbb{Z}_q$ have circular spacing q/p , so coefficient-wise rounding recovers $Dm \bmod p$. Because D is a power of two and p is odd, multiplication by $D^{-1} \bmod p$ recovers m . $\square$

B.2.3 Proof of Corollary 5.1.

Proof. By (7), $z_S \equiv D(\hat{m}_\tau + e_\tau) - \sum_{i \in S} \lambda_i^S \rho_i \pmod{q}$. The right-hand side has coefficient norm at most $D(B_{m,\tau} + B_e) + \mathcal{B}(S) < q/2$, so this is equality of centered representatives in $\mathcal{R}$. Every embedding root has modulus one, hence $\|\sigma_{\text{slot}}(w)\|_\infty \leq n\|w\|_\infty$. Linearity and (5) give

$$\left\| \text{Num}_\tau(z_S) - \frac{\sigma_{\text{slot}}(\hat{m}_\tau)}{\Delta_{\text{ct}_\tau}} \right\|_\infty \leq \frac{n}{\Delta_{\text{ct}_\tau}} \left(B_e + \frac{\mathcal{B}(S)}{D} \right) + \nu_\tau.$$

Equation (11) gives $\text{OutOK}_\tau(\hat{m}_\tau, y_{\tau,S}) = 1$; approximating $\mathbf{z}_\tau$ by $\sigma_{\text{slot}}(\hat{m}_\tau)/\Delta_{\text{ct}_\tau}$ adds $\varepsilon_{\text{enc},\tau}$. The argument is pointwise in the keys, oracle answers, ciphertext, and responder set, so it holds for every admitted label and qualified set, even when selected using the preceding transcript. $\square$

Remark 3 (CKKS scaling). Multiplication by D requires (10), beyond $B_{m,\tau} + B_e < q/2$. CKKS retains residual error, so (10)–(11) must hold alongside the security bound on $L = q/r$. Deterministic responses and Num_τ give identical outputs for fixed (τ, S) across recipients and retries; different qualified sets may disagree. Correctness assumes no distribution or independence of rounding residues; joint response/output simulation is proved separately.

B.3 Verifiable decryption: functionality, protocol, and simulator

For $2 \leq t \leq N$ and static $|C| < t$, retain the numerical algorithms, labels, and [shared state rules](#). Replace ideal admission by the local validation below, and bind responses to DKG shares in the local-ZK hybrid ([Theorem 7.3](#)). BfV returns **Combine**; CKKS returns $Q_\tau(\text{Combine})$ under [Section 7.3](#)'s additional conditions. First exposure advances the privacy threshold; verified receipt enables reconstruction.

Functionality $\mathcal{F}_{\text{StThFHE}, \perp}^{\text{mal}, \text{com}}$

1. **KeyGen_i**: invoke DKG **Init_i** on (sid, kid) .
2. **R6 evidence**: accept well-typed honest C_i from $\mathcal{S}$ without openings; require corrupt C_i to open to its actual ideal pair. The first successful **Done_{i,6}** requires a complete valid commitment table with one entry per party and makes the table immutable. Deliver **Ready_j** when its local DKG readiness conditions hold.
3. **Accepted DKG PublishKey**: enable **Enc/Eval** and execute the input-proof and local-validation rules below, including their private checks, sampling, failures, and recipient-specific arrivals.
4. **First local admission of (τ, ct_τ)** : internally store $v_\tau = \text{Dec}_\tau^*(s, \text{ct}_\tau)$, set $\text{Seen}_\tau = C$, and initialize missing application state once, preserving previously buffered records. Further admissions and alternate validation records reuse this state.
5. **Exposure (e, T, M) from $\mathcal{S}$** : require a live honest ticket, $T = \text{Seen}_\tau$, no published (τ, i) response, and M satisfying the atomic-publication checks above; otherwise reject without target access or state changes. Use the core functionality's atomic counting and target-release rule to store/expose $(\text{kid}, \tau, i, C_i, M(w))$, complete e , and queue this immutable message for all recipients. Fix one proof task per $j \neq i$, persisting after cancellation.
6. **Delivery and validation events**: apply **Process** below.
7. **FinDec(ssid, S)**: require $|S| = t$ and $S \subseteq D_{\tau, j}^{\text{ver}}$; otherwise return $\perp_{\text{notready}}$ without closing the context. Return v_τ using the output record indexed by (τ, S) , without final adversarial approval.
8. **Cancel**: apply the shared rules, preserving proof tasks and sessions.

Verifiable decryption protocol Π_{vJambon}

Store the final shares and commitment randomness (f_i, g_i, ϑ_i) after the R6 checks; public records may exist earlier.

1. **Enc_i(μ)** after key publication: sample ω with the declared encryption distribution. If $\text{ValidInput}_{\text{pp}}(\mu, \omega) = 0$, return $\perp_{\text{input}}$ without a root or proof task. Otherwise publish $\text{ct} = \text{Enc}(\text{pub}, \mu; \omega)$ and its root record, then invoke (V0) for each $j \neq i$; record the honest sender's self-acceptance.
2. **Eval** or a validation-record arrival: recompute the public evaluation records and update local admission by the validation procedure below. Return the evaluated ciphertext; evaluation does not require plaintexts.
3. Ready live request bound to τ with $\text{Adm}_i(\tau) = 1$ and finalized commitment table: compute $p_{i,\tau} = \lfloor [a_\tau f_i + H_0(\tau)g_i]_q \rfloor_r$. Publish and retain the authenticated response $(\text{kid}, \tau, i, C_i, p_{i,\tau})$, atomically fixing one persistent proof task per $j \neq i$. Each task subsequently invokes local ZK session $(\text{sid}, \text{kid}, \text{part}, \tau, i, j)$ with $x = (\text{sid}, \text{kid}, \tau, i, C_i, p_{i,\tau}, h_\tau)$, $h_\tau = H_0(\tau)$, and witness (f, g, ϑ, ρ) satisfying (V1).
4. Delivery and validation events: apply **Process** below.
5. **FinDec**: apply the functionality's output checks and rules for recording and reusing outputs; return **Combine** for BFV and $Q_\tau(\text{Combine})$ for CKKS. Retries and **Cancel** use the common rules.

Straight-line simulator $\mathcal{S}_{\text{mal}}$

1. Setup: run $\mathcal{S}_6$ at its prescribed events. At R6 supply honest $C_i = \text{Com}(\langle 0, 0 \rangle; \widehat{\vartheta}_i)$ and source notices; check corrupt references/openings against actual ideal pairs.
2. Inputs/evaluation: publish the honest ciphertext records supplied by the functionality and emulate their prescribed ZK tasks without witnesses. Read corrupt (V0) witnesses from actual local-resource submissions; forward them privately for validation. Replay public evaluation computations and schedule each record/proof arrival independently, preserving local admission.
3. Exposure ticket, $T = \text{Seen}_\tau$: if $|T| < t-1$, provisionally sample $u_{i,\tau} \xleftarrow{\$} \mathcal{R}_q$ and propose $M(\cdot) = \lfloor u_{i,\tau} \rfloor_r$. At $|T| = t-1$, interpolate Φ of degree at most $t-1$ by

$$\begin{aligned} \Phi(0; v) &= b_\tau - \text{EncTarget}_\tau(v), \\ \Phi(\alpha_j; v) &= a_\tau f_j + H_0(\tau)g_j & (j \in C), \\ \Phi(\alpha_j; v) &= u_{j,\tau} & (j \in T \setminus C). \end{aligned} \tag{S1}$$

Propose $M(v) = \lfloor \Phi(\alpha_i; v) \rfloor_r$, $M(\perp) = 0$; after accepted exposure retain $P'_\tau = \Phi(Y; v_\tau)$. Later first responses use $M(\cdot) = \lfloor P'_\tau(\alpha_i) \rfloor_r$.

4. Accepted exposure: store the unrounded value or polynomial P'_τ . Rejected/canceled proposal: discard it. Stale proposal: recompute for the live ticket/current T . Retry: replay the published response.
5. Transport: schedule from notified tickets, published responses, resource notices and outputs, using the emulation rules below.

Common execution rules. Each rule below applies to the functionality, protocol, or simulator it names.

Setup and contexts. On the same (sid, kid) , $\text{KeyGen}_i = \text{Init}_i$ uses the [DKG functionality](#)'s single sampler and start, phase, candidate, abort, replay and pending rules; a party need not wait for other parties to become ready. The real protocol uses six-wave DKG, R6 certification ([D1](#)), finalization of the commitment table, and [Section 6.3](#)'s publication wrapper (prescribed DKG callers and public R4 checks). After the first R6 completion notification is delivered, new source proofs, updates, and transfers are rejected; historical receipts remain verifiable. Enabled **PublishKey** permits public operations and local validation even before candidate/readiness conditions hold or after abort; records persist. Honest binding/decryption requires local readiness, validation, and a finalized commitment table. Input proofs add communication, not a DKG wave.

Input proofs and public evaluation. Fix public **Enc/Eval** code, finite message and randomness spaces, an allowed set of circuits and parameters, and polynomial-time **ValidInput_{pp}**. This predicate checks message and randomness formats, canonical encodings, membership in the declared randomness support, and coefficient bounds of the [output and correctness conditions](#). Support membership checks permitted values, not their sampling probabilities. Honest encryption samples its declared distribution without resampling; corrupt encryption may choose any accepted witness. For root $x = (\text{sid}, \text{kid}, \text{input}, i, \ell, \text{pub}, \text{pp}, \text{ct})$, with fresh honest serial ℓ , session $(\text{sid}, \text{kid}, \text{input}, i, \ell, j)$ for $j \neq i$ proves the relation ([V0](#)) (with encryption randomness denoted here by ω). Each evaluation record gives its full polynomial-size public circuit, ordered canonical parent records, output parameters and ciphertext. Check every operation, ciphertext format and parameter, and recompute every node by **Eval**, using checked explicit public randomness for randomized evaluation. Each input edge requires its complete source record, including its verified ancestry; bare ciphertexts or unsupported imports do not suffice.

Local ciphertext-validation procedure. At j , buffer authenticated records and matching ([V0](#)) acceptances in either order, including before readiness. Key publication and every record/proof arrival process all newly enabled nodes topologically. Set the local flag $\text{Adm}_j(\tau) = 1$ iff the complete received evaluation circuit passes public checks, each root has matching local acceptance (or honest self-acceptance), and every statement matches the published key and output specification. Admission at j does not wait for other recipients. The local-ZK rules below allow a proof session to start before its statement is registered (an off-registry session) or its record published; witness submission alone neither publishes nor admits a record. Immutability is per recipient/session;

match corrupt equivocation against complete records without global first-record selection or agreement. Canonical $\tau = \text{Encode}(\text{kid}, \text{pp}, \text{ct})$ excludes record/proof serials and input dependencies: alternate validation records share exposure, response, and output records. Records, acceptances and local flags persist after cancellation/setup abort. No admission functionality is supplied.

Ideal execution of input proofs. Honest Enc_i privately receives μ , performs the same checks/sampling, and publishes only its root before enabling proof sessions; failure creates none. For proof sessions initiated by corrupted parties, $\mathcal{S}$ forwards actual submitted (x, μ, ω) privately; check (V0) and reproduce immutable valid-start/acceptance events, including before record arrival. Run the same public/local validation procedure. Honest inputs/coins are neither requested from nor disclosed to $\mathcal{S}$. DKG and decryption share one internal (pub, s, F, G) with $F(0) = s$, $G(0) = 0$; s persists after publication/abort; no command in this interface returns it to the simulator or any party. Dec_τ^* is BFV decryption or finite CKKS Dec_τ^Q ; its stored target in finite $\mathcal{V}_\tau$ stays private until authorized release.

Local ZK and public checks. Use a separate ideal ZK session $\mathcal{F}_{\text{ZK}}$ for each sender–recipient pair, following the commit-and-prove pattern of [8, Sect. 7.1]. Acceptance is specific to that recipient and session; it is not a transferable NIZK. A private valid $(\text{Prove}, \text{zkid}, x, w)$ stores x , notifies $\mathcal{A}$ of (zkid, i, j, x) , and enables scheduled authenticated acceptance at j without revealing w . The first valid input fixes even an off-registry session; invalid inputs preserve state, conflicts give no acceptance, repeats reuse records, and admission never resets them. Public checks require $\text{Adm}_j(\tau)$, canonical encodings, key/identity/label, $C_i = \text{KeyCom}[i]$, $p_{i,\tau} \in \mathcal{R}_\tau$, $h_\tau = H_0(\tau)$ and matching local acceptance. Check the hash externally; the NP circuit uses constant h_τ without an oracle gate.

Delivery: Process(τ, j). In both worlds, discard wrongly tagged arrivals; buffer well-typed responses and verifier-local acceptances independently, even before validation/readiness/binding. Arrival, local admission, commitment-table finalization, readiness, or binding processes all eligible buffered pairs for affected (τ, j) in the same activation, without another $\mathcal{S}$ command. Finalizing the commitment table triggers processing for all pairs; Ready_j triggers processing for all of j 's contexts. Insert i once in $D_{\tau,j}^{\text{ver}}$ iff matching arrivals pass public checks, the commitment table is finalized, and j is ready/bound to τ with $\text{Adm}_j(\tau) = 1$. Queued records cannot enter the verified set before receipt at j ; honest self-receipt is immediate. In the ideal execution, a response from a corrupted party must also be the prescribed response on its actual ideal pair; other traffic creates no response-table entry.

Retries and cancellation. The [shared rules](#) apply also to ZK: retries reuse proof sessions; cancellation preserves pending proof tasks, sessions, acceptances, and buffered records.

Simulator interface. Run $\mathcal{A}$; emulate source/ZK resources and all corrupt submissions (including off-registry validity, conflicts and repeats), retaining witnesses. Check source-pair consistency only at registered public checks. Honest notices use published responses/persistent tasks, no honest witness, and the sup-

plied unprogrammed oracle. Set $\text{EncTarget}_\tau(v) = (q/p)v$ for BFV and $\text{Lift}_\tau(v)$ for CKKS; the public total polynomial-time lift makes (S1) admissible even if nonlinear. The functionality handles private Bind/Cancel and buffers; $\mathcal{S}$ requests no private context, honest canonical share or early target. Rejected proposals change no stored record. Without local validation, honest Bind/PDec has no effect and FinDec returns $\perp_{\text{notready}}$; other readiness/failure rules persist. A new recipient never resets Seen_τ or creates a second response.

C Distributed Setup: Specifications and Variants

This appendix specifies setup; [Sections D.2](#) and [D.3](#) prove its security.

C.1 Underlying key generation and compatibility

Definition 13 (Key generation with additive shares and DKG syntax).

A backend Π_B generates a public record pk_B , including all evaluation keys, and private additive shares $s_1^{\text{add}}, \dots, s_N^{\text{add}} \in \mathcal{R}_q$ with $\sum_j s_j^{\text{add}} = s$. Its specification fixes their joint distribution, auxiliary state, public transcript, and permitted message-delivery sequences. The sampled backend record consists of its randomness and key outputs, excluding subsequent adversarial delivery decisions. Let ξ_C denote the current backend view of the statically corrupted set C : their available additive shares, stored randomness, and messages received so far. No uniform distribution is assumed for the additive-share vector $s^{\text{add}} = (s_j^{\text{add}})_{j \in [N]}$. The compiler $\text{DKG}[\Pi_B](1^\lambda, 1^t, 1^N)$ runs this backend and privately delivers $\text{sk}_i = (f_i, g_i)$ and the common public record to each party completing setup. Backend state and dealing coins may be retained.

Key-distribution and modulus compatibility. To obtain the trusted key distribution of [Section 5.3](#), the backend’s joint (pk_B, s) distribution must equal that of the specified base-FHE key generator, including evaluation keys and their correlations. The additive reconstruction identity alone does not imply this condition. Shares must lie in the application ring $\mathcal{R}_q$. Reduction from a backend modulus Q_0 requires $q \mid Q_0$ and compatible base keys; arbitrary integer representatives need not preserve reconstruction. The exact-integer-opening variant of [Lemma 7\(iv\)](#) removes divisibility only for sharing; compatibility of the retained public and evaluation keys remains required. In the prime- Q_0 option of [\[20, Sect. 4.1\]](#), taking $q = Q_0$ is incompatible with $1 < r < q$ and $r \mid q$. A compatible composite-modulus backend is not excluded, but its parameter and exactness conditions must be established.

Definition 14 (Setup interface and security). Fix $C \subset [N]$ before setup, with $c = |C| < t$. The backend-adapted setup interface samples one backend record according to its declared distribution and independent polynomials of [\(14\)](#). It sets $f_i = F(\alpha_i)$ and $g_i = G(\alpha_i)$ and supplies the aggregate values f_C, g_C privately to the simulator once this record is sampled. The public record and updates to ξ_C are released at their specified backend events; this does not

disclose future received messages early. Honest key-ready notices obey the actual backend-output and all-contribution receipt gates, without resampling when messages are withheld. Only shares, public data, and readiness are application outputs; retained backend state belongs to the corrupted setup view. Let ThKeyGen_B denote the application-key generator obtained from this joint sampling distribution by outputting the public record and $(f_i, g_i)_{i \in [N]}$. Let ξ_B denote the setup-view profile of Definition 9 with this joint backend record, corrupted-share release at sampling, and the publication, message-delivery, and readiness rules just stated. Thus the setup resource is $\mathcal{F}_{\text{KG}}[\text{ThKeyGen}_B, \xi_B]$.

A secure realization simulates retained coins, visible messages, metadata, delivery order, concurrent sessions, and incomplete executions, respecting each item's release time. Sharing security is in the backend and authenticated private-channel hybrid; full DKG requires a composable realization of that backend interface. To realize the original $\mathcal{F}_{\text{KG}}[\text{ThKeyGen}, \emptyset]$, whose simulator receives only (pk, sk_C) , the application-key distribution must be that of ThKeyGen and an additional simulator must reproduce ξ_C and each delivered part of the backend transcript from those permitted outputs with the same joint distribution. Charge this additional simulation error separately; the output distribution alone does not establish transcript simulation.

For threshold decryption, use $\mathcal{F}_{\text{ThFHE}}^{\text{str}}[\text{ThKeyGen}_B, N, t, \Pi, \xi_B]$ from Definition 11. The trusted instance additionally requires the compatibility and transcript simulation above; corruption remains static.

C.2 Key-generation protocols and mask sharing

Backend instantiation. MHP⁺26 [20] supplies a concrete additive-key architecture for sparse-key threshold CKKS. We retain its keys and its costs for sparse-secret generation, temporary threshold encryption, and public/evaluation-key generation, including temporary noise flooding. Its main public/evaluation-key generation variant has party-dependent noise, so the joint-distribution and modulus conditions above require separate verification. Its static honest-but-curious analysis, including circular security for evaluation keys, does not establish the entire composable interface of Definition 14.

Rounds and availability. The sharing wave of Section 6.1 can include zero shares in existing key-resharing messages. In [20, Sect. 4], additive shares become available after round three and may be re-shared during the fourth and final round, alongside evaluation-key completion. Thus one additional wave bounds the sequential construction, not this integrated schedule; the full DKG is not one-round. Completion requires every prescribed contribution and backend output. Withholding can block setup, and no timeout selects a new dealer subset.

Mask sharing and the key-generation schedule. The architecture of [20] opens the masked value in (15) and forms additive shares $s_j^{\text{add}} = \frac{1}{N}\mathbf{m} - \text{fl}_j$. It already samples masks early [20, Sect. 4.2] and reshares in its final round, so (16)–(17) claim no further round reduction. They rearrange standard resharing [21]: reshare $s_j^{\text{add}} = c_j\mathbf{m} - \text{fl}_j$ for public $\sum_j c_j = 1$ and collect the public offsets at aggregation. Taking $c_1 = 1$ and the others zero avoids division by N .

Lemma 7 (Sharing private masks and adding the public opening).

Assume the semi-honest construction above, with exact independent coefficient sampling, and a backend satisfying (15). Replace steps (2) and (3) by (16) and (17). Then the following hold.

- (i) (Output distribution.) Conditional on the sampled backend record, $f_i = F(\alpha_i)$ for the polynomial F of (14) with $a_\ell = \sum_j A_{j,\ell}$, which is independently uniform in $\mathcal{R}_q$ under the prescribed sampling; the conclusion of Theorem 6.1 is therefore unchanged.
- (ii) (Privacy of a dealing.) For a fixed coalition C with $|C| < t$ and an honest dealer j , the received values $(\tilde{K}_j(\alpha_i))_{i \in C}$ are uniform in $\mathcal{R}_q^{|C|}$ and independent of $\mathfrak{f}_j$.
- (iii) (No inverse of N .) This sharing step never divides by N , so the prime- Q_0 option of [20, Sect. 4.1] is not required by that step.
- (iv) (Changing the sharing modulus.) If (15) holds in $\mathcal{R}$ without wrap-around, reduce it modulo any admissible application modulus q and sample $A_{j,\ell} \xleftarrow{\$} \mathcal{R}_q$. The resulting shares lie in $\mathcal{R}_q$ without requiring $q \mid Q_0$.

Proof. For (i), substituting (16) into (17) and using (15) gives

$$f_i = \mathbf{m} - \sum_{j=1}^N \mathfrak{f}_j + \sum_{\ell=1}^{t-1} \left(\sum_{j=1}^N A_{j,\ell} \right) \alpha_i^\ell = s + \sum_{\ell=1}^{t-1} a_\ell \alpha_i^\ell.$$

A sum containing one uniform independent summand is uniform and independent of the remaining ones, which gives the coefficient distribution. For (ii), every α_i is a monomial unit, and Lemma 5 shows that the pairwise differences of distinct points are units for odd q . For $c = |C| > 0$, the first c columns of the evaluation map $(A_{j,\ell})_\ell \mapsto (\sum_\ell A_{j,\ell} \alpha_i^\ell)_{i \in C}$ have determinant $\prod_{i \in C} \alpha_i \prod_{i < k, i, k \in C} (\alpha_k - \alpha_i)$, which is a unit. The map is therefore surjective for $c \leq t-1$ and sends uniform coefficients to a uniform vector, which additively masks the constant term. For $c = 0$, the claim is vacuous. Claim (iii) is immediate from (17), in which $\mathbf{m}$ enters with coefficient one. For (iv), an identity that holds in $\mathcal{R}$ reduces to every quotient $\mathcal{R}_q$, and the dealing coefficients are drawn in the target ring rather than transported from $\mathcal{R}_{Q_0}$. $\square$

Claim (iv) remains unverified for [20]: it requires consistent integer representatives, an exact opening, and an aggregate no-wrap bound at the opening modulus, including failure probabilities. A scale gap or modular identity does not suffice: lifting the latter may introduce an unknown multiple of the modulus. The six-wave construction below instead cancels uniform masks directly in $\mathcal{R}_q$ using explicit temporary-HE embeddings. Preparing shares before $\mathbf{m}$ requires early access to each private mask; Remark 7 states the additional simulation obligations, beyond the composition theorem's subsequent sharing wave.

Remark 4 (Sampling). For each coefficient, rejection sampling retains blocks of J independent $w(q)$ -bit candidates and uses the first below q , repeating if necessary. A bounded implementation reports setup failure after one unsuccessful

block. Couple their retained coins until this event. Since each rejection has probability below $1/2$, their complete views differ with probability at most

$$\varepsilon_{\text{samp}} \leq 2N(t-1)n2^{-J} + \varepsilon_{\text{KG}},$$

where ε_{KG} is a separately established backend key-sampling discrepancy or failure probability. This bounds the sampled record and all partial transcripts, without conditioning on completion or selecting a different dealer set after failure. For the additive profile below, replace the coefficient count $2N(t-1)n$ by $N(N-1)n$.

C.3 A deterministic CKKS refinement

Remark 5 (Deterministic canonical-norm refinement for CKKS).

Keep the original coefficient-norm no-wrap condition unchanged. Let σ_j range over the canonical embeddings of the cyclotomic field. For normalized reconstruction weights $\ell_i^S = \lambda_i^S/D$, define

$$A_{\text{can}}(S) := \max_j \sum_{i \in S} |\sigma_j(\ell_i^S)| \leq M_1(S)/D.$$

The slot-error bound can be sharpened to

$$\frac{n}{\Delta_{\text{ct}_\tau}} \left(B_e + \frac{L-1}{2} A_{\text{can}}(S) \right) + \nu_\tau. \quad (29)$$

The proof in [Section D.2](#) applies each embedding to the same integral residues as [Lemma 1](#). This deterministic estimate needs no independence premise, but a uniform guarantee requires a proved bound over every qualified set; sampled-set measurements do not justify revised parameters.

C.4 Additive sharing with all parties required

Definition 15 (Additive profile). For the separate access structure requiring all N parties, retain $f_i = s_i^{\text{add}}$ and omit key re-sharing. Each dealer j samples a uniform vector $(y_{j,1}, \dots, y_{j,N}) \in \mathcal{R}_q^N$ with sum zero by choosing $N-1$ coordinates independently and setting the last to minus their sum. Privately send each recipient i its coordinate and set $g_i = \sum_j y_{j,i}$. Then g is a uniform additive zero sharing independent of the backend. Use the same canonical labels and response

$$p_i = \lfloor [af_i + H_0(\tau)g_i]_q \rfloor_r.$$

Combine all N responses by

$$z = \left[b - L \sum_{i=1}^N p_i \right]_q. \quad (30)$$

For BFV output $\lfloor z \rfloor_p$. For CKKS use a public deterministic numerical decoding map $\text{Num}_\tau^{\text{add}}$ with error at most ν_τ from $\sigma_{\text{slot}}(z)/\Delta_{\text{ct}_\tau}$. The reconstruction weights

and scale are 1 independently of the Ring–Shamir definition of D . This is a separate profile, not a substitution $t = N$ in that definition. We write $\text{ThKeyGen}^{\text{add}}$ for the algorithm that samples the selected backend’s joint public/evaluation-key and additive-key-share distribution, independently samples the uniform additive zero sharing above, and outputs the public record and $(f_i, g_i)_{i \in [N]}$. Its application-key instance is $\mathcal{F}_{\text{ThFHE}}^{\text{str}}[\text{ThKeyGen}^{\text{add}}, N, N, \Pi, \emptyset]$ of Definition 11. Let $\xi_{\text{B}}^{\text{add}}$ follow the sampling and release rules of Definition 14 for this additive distribution; readiness requires the backend output and all zero contributions. The retained-state setup is $\mathcal{F}_{\text{KG}}[\text{ThKeyGen}^{\text{add}}, \xi_{\text{B}}^{\text{add}}]$, and its online instance is $\mathcal{F}_{\text{ThFHE}}^{\text{str}}[\text{ThKeyGen}^{\text{add}}, N, N, \Pi, \xi_{\text{B}}^{\text{add}}]$ of Definition 11.

Theorem C.1 (Additive correctness). *For every additive key and zero sharing and every admitted ciphertext, there are integral residues ρ_i with $\|\rho_i\|_{\infty} \leq (L - 1)/2$ such that*

$$z \equiv b - as - \sum_{i=1}^N \rho_i \pmod{q}.$$

BFV correctness holds if $B_e + N(L - 1)/2 < q/(2p)$. For CKKS, sufficient conditions are

$$B_{m,\tau} + B_e + N(L - 1)/2 < q/2, \quad (31)$$

$$\frac{n}{\Delta_{\text{ct}\tau}} (B_e + N(L - 1)/2) + \nu_{\tau} \leq \varepsilon_{\text{dec},\tau}. \quad (32)$$

The strong-security proof for threshold decryption in Theorem D.2 has residual dimension $k = N - |C| - 1$, mask bound kB_{χ} , and output-shift bound B_e . In this precise sense the reconstruction and simulation factors are $D = 1$ and $\Gamma = 1$. The backend fixes the key-share distribution, which need not be uniform, and must separately ensure ciphertext privacy given the corrupted setup view. Zero dealing still costs $N(N - 1)nw(q)$ share bits and one wave unless a specified backend schedule carries it already. Each response remains one $\mathcal{R}_r$ element, or $nw(r)$ bits. Any numerical saving from the smaller factors requires matched security, precision, and modulus conditions.

C.5 Malicious setup functionality and resources

Use the static coalition, base-key distribution, and ring parameters of Section 6.3, and put $c = |C|$. The functionality distinguishes candidate disclosure, abort, public-key publication, and recipient-local key delivery.

Functionality $\mathcal{F}_{\text{DKG},\perp}^{t,N,\mathcal{D}}$

1. **Init_i** ($i \in H$), **first call**. Insert i into $I = B_0$ and notify $\mathcal{S}$ of **Started**(i), even after abort. If $I = H$, active, and unsampled: fix $\widetilde{\text{corr}} \supseteq \text{corr}$, $|\widetilde{\text{corr}}| = t - 1$; sample $(\text{pub}, s) \leftarrow \text{KeySamp}_{\mathcal{D}}(1^{\lambda})$ once and, independently,

$(f_i, g_i) \xleftarrow{\$} \mathcal{R}_q^2$ for $i \in \widetilde{\text{corr}}$. Interpolate the unique degree- $< t$ polynomials

$$F(0) = s, \quad F(\alpha_i) = f_i, \quad G(0) = 0, \quad G(\alpha_i) = g_i \quad (i \in \widetilde{\text{corr}}),$$

and set all remaining pairs to $(F(\alpha_i), G(\alpha_i))$.

2. **Done(i, r) from $\mathcal{S}$.** Require

$$i \in H \setminus B_r, \quad H \subseteq B_{r-1}, \quad 1 \leq r \leq 6, \quad \text{active} \vee (\text{success} \wedge r = 6).$$

Insert i into B_r . The first insertion into B_3 releases **KeyCandidate(pub)** only to $\mathcal{S}$; the first into B_4 releases **ShareCandidate** $((f_i, g_i)_{i \in C})$ only to $\mathcal{S}$, hiding the padding. The first into B_6 fixes success.

3. **PublishKey from $\mathcal{S}$.** After **KeyCandidate**, release **pub** once, even after abort.

4. **ReleaseKey(j) from $\mathcal{S}$, $j \in B_6$.** Independently deliver the queued **Ready_j** and privately provide, for application use, (pub, f_j, g_j) once.

5. **Abort from $\mathcal{S}$.** Require no success and $C \neq \emptyset$; permanently abort, even before $I = H$. Record stage initial if $B_3 = \emptyset$, key if $B_3 \neq \emptyset = B_4$, and share otherwise.

6. **ReleaseAbort(j) from $\mathcal{S}$, $j \in I$.** After abort, deliver $\perp$ once, including to late starters.

C.5.1 Local resources and control The following rules complete the functionality, protocol, and simulator. Only the simulator issues the ideal scheduling commands; these commands do not authenticate real events. All commands carry (sid, kid) ; invalid or disabled inputs do nothing, and repeats reuse records. Initially the key-generation instance is active and unsampled, with $H = [N] \setminus C$, $I = B_0 = \emptyset$, and $B_1 = \dots = B_6 = \emptyset$. Never resample; abort preserves past releases and enabled publication, while missing inputs and notices stay pending. Honest sampling/domain failures are charged to [Theorem 7.2](#). The hybrid provides consistent authenticated broadcast, authenticated private channels, separately delivered round completion notifications, and session-aware $\mathcal{F}_{\text{VCP}}$. A party's first **Init** publicly registers **Started(i)** even after abort and starts R1 in that activation unless locally terminal. A party's round- r completion notification is delivered only after all required messages and receipts verify; for $r < 6$, its delivery immediately starts $R(r+1)$. Delivery to one recipient does not wait for delivery to others. Required message entries are immutable; ignore messages with the wrong session, unauthenticated messages, duplicates, and messages for rounds that have already closed. Only failure of a required current message or check terminates the receiving party, which sends no later **ok**. Silence remains pending. Other parties may continue until locally notified; a late first **Init** still starts R1 if no local failure is known. No party restarts after its own terminal notice. Readiness and failure notices have independent recipient-local delivery. For public-key publication, a wrapper checks one immutable R4 packet per identity

and every exact-statement receipt, bound to this key instance, certified output handles, and the same preceding transcript. It decodes only designated public outputs into canonical `pub` and registers that record once; invalid or conflicting submissions have no effect. Honest parties invoke it automatically at verified R4 reconstruction; corrupted invocations are actions of $\mathcal{A}$. Only DKG identities may submit. This local resource call adds no message wave. $\mathcal{F}_{\text{VCP}}$ supports the following restricted operations. A handle identifies a registered value; its type fixes the domain and encoding. A receipt attests that the prescribed verification succeeded.

1. **Commit** registers a typed, immutable, owner-specific value and returns only fixed public metadata, padded by type.
2. **Prove** accepts only registered stage relations on original owner handles and authorized incoming references. Its receipt binds the exact public statement.
3. **Transfer** privately delivers a registered coordinate to its named recipient and registers that same value for subsequent prescribed relations. Values transferred between honest parties remain private. Coin opening is the prescribed public-opening operation.

There is no external honest-witness query, unrestricted relation, or post-protocol export. A simulator emulating these local subroutines sees corrupted submissions, not the adversary’s private tape, and may defer unexposed honest records. These capabilities are part of the hybrid specification. A concrete commitment/proof implementation must realize this entire interface.

Domains. Use the negacyclic temporary ring $\mathcal{R} = \mathbb{Z}[Z]/(Z^{n_{\text{tmp}}} + 1)$ with power-of-two n_{tmp} , and let $\mathcal{R}_v = \mathcal{R}/v\mathcal{R}$. Temporary ciphertexts lie in $\mathcal{R}_Q$; relinearization-key generation uses modulus $M = PQ$. This coefficient representation satisfies $\|uv\|_\infty \leq \|u\|_1 \|v\|_\infty$ for integer lifts, as used by the error bounds. Final masks and sharing polynomials lie in $\mathcal{R}_q$. Every output identifier o has a fixed output module $\mathcal{R}_o$ and an explicit addition-preserving embedding Emb_o of its plaintext domain. For example, $q \mid Q_o$ permits coefficient padding and multiplication by Q_o/q into a sufficiently large output module. Different rings are not identified, and no inverse of N or uncertified integer opening is used. The application, temporary-ring, and output-module dimensions are n , n_{tmp} , and n_o ; each statistical bound uses the dimension of its sampled noise vector.

Use the direct-RLWE temporary HE of (42): its public key is available at M , and lower-level keys are its projections. At each declared level its encryption distribution is the R2 formula below with $\xi = 0$ and the scheme’s prescribed encryption coins. Its ordinary IND-CPA experiment includes the complete public evaluation key (EVK) of the temporary HE, distinct from the application keys produced by the key-generation circuit. The proof uses the explicitly defined public auxiliary samples of Section D.3.1; Lemma 13 reduces that experiment to IND-CPA with that public evaluation key and auxiliary RLWE. No honest temporary secret or unused witness is disclosed by the resource interface.

The full tape circuit, input/output handles, encodings, witness domains, and error distributions are fixed before execution. Use the temporary-HE hypotheses and error bounds of Section D.3. The correctness bound must cover the com-

plete circuit, level projections, embeddings, and all enlarged relinearization-key, input, and output flooding noise. [Theorem 7.2](#) assumes this bound; no concrete parameter set establishes its feasibility here. Concrete resource realizations add their UC/joint-state composition losses.

C.5.2 A candidate for the commitment/proof resource Garay et al.’s multiparty extended committed oblivious transfer ($\mathcal{F}_{\text{mECOT}}$) [14, full version, Sect. 6.2] supplies immutable bit commitments, Boolean gate relations, and private transfer with a new recipient-owned commitment to the transferred bit. For a candidate implementation, encode each value as a fixed-length bit vector with session-, stage-, owner-, and handle-specific identifiers. Express the prescribed domain and stage predicates as Boolean circuits on original and authorized incoming handles; commit intermediate wires and prove their gates. For a coordinate bit, use its original handle as both ECOT branch handles. Either receiver choice then transfers exactly that bit and registers it under the recipient, supplying the incoming reference needed by later relations, including (D1).

The cited construction uses CRS and broadcast, DDH, and its specified UC-ZK resources; the paper’s UC-ZK compiler additionally uses strong RSA or its stated DSA assumption. We use only its static-corruption restriction, not the adaptive theorem requiring erasures. A realization still needs proofs of exact statement binding, recipient-local delivery, immutable values, concurrent sessions, and joint-state setup security. The temporary-HE correctness condition HE-Corr must be established separately.

If B_{reg} , G , B_{tr} , and B_{op} count registered bits (including intermediate wires), proved gates, transferred bits, and opened bits, a concrete implementation must account for

$$B_{\text{reg}}c_{\text{com}} + Gc_{\text{gate}} + B_{\text{tr}}c_{\text{tr}} + B_{\text{op}}c_{\text{op}} + C_{\text{meta}},$$

where each unit cost includes its verifier fan-out and resource traffic. These implementation-dependent costs and any additional communication waves are excluded from the response-size tables.

C.5.3 Six-wave protocol The public setup samples independent uniform a_{pk}, a_j in $\mathcal{R}_M$. The values $h_{0,i,j}, h_{1,i,j}$ below are intermediate relinearization-key contributions; R3 combines them into the temporary relinearization key $(K_j, A_j)_j$, used for homomorphic multiplication in the key-generation circuit. We display one gadget column with coefficient γ , suppressing column indices, including on independent coins $v_{i,j}$ and $A_j = \sum_i v_{i,j}$. Columns share temporary secrets, with $s^{\text{tmp}} = \sum_i s_i^{\text{tmp}}$ and $u = \sum_i u_i$. Honest secrets follow $\chi_{\text{tmp}}, \chi_u$; other honest randomness is independent except for the specified joint error distributions. Honest e_i^{pk} follows the actual TempHE public-key error distribution, and $e_{0,i,j} \leftarrow \chi_{0,j}$ follows the auxiliary-RLWE error distribution, independently. For every honest i and every column j , the R1 error is $e_{1,i,j} \xleftarrow{\$} [-B_{1,j}, B_{1,j}]^{n_{\text{tmp}}}$, independently. (44) includes these radii and the independent uniform coefficient

noise for row sums and input flooding; (L1) bounds the public-sample simulation error. Honest samples outside the required domains count toward θ_{reg} in (45), including before R3 and when $C = \emptyset$, rather than the conditional HE-Corr bound. Every message and reference is bound to (sid, kid, stage, owner).

Protocol Π_6 : rounds 1–3

R1: fixed state and noise rows. Party i registers temporary secrets s_i^{tmp}, u_i , setup errors, a uniform random tape ω_i (the private bit string used by the key-generation circuit), a uniform source mask $\text{fl}_i \in \mathcal{R}_q$, and a uniform coin $v_i \in \mathcal{R}_M$. For each gadget column, it samples independent $\eta_i \xleftarrow{\$} [-B_\eta, B_\eta]^{n_{\text{tmp}}}$. It broadcasts and proves

$$\begin{aligned} b_i &= -a_{\text{pk}} s_i^{\text{tmp}} + e_i^{\text{pk}}, \\ h_{0,i} &= -a u_i + \gamma s_i^{\text{tmp}} + e_{0,i}, \quad h_{1,i} = a s_i^{\text{tmp}} + e_{1,i}. \end{aligned}$$

For each output handle o , it samples a small row sum $\rho_{i,o}$ and a uniform row $(w_{i \rightarrow j,o})_{j \in [N]} \in \mathcal{A}_o^N$ conditioned on $\sum_j w_{i \rightarrow j,o} = \rho_{i,o}$; it commits the row and proves its sum and domains. All bounds are part of the registered relations. After checking, set $b = \sum_i b_i$, $\bar{h}_0 = \sum_i h_{0,i}$, $\bar{h}_1 = \sum_i h_{1,i}$.

R2: coins, encrypted tapes, and row transfers. Open the registered v_i and set $A = \sum_i v_i$. Broadcast smudged encryptions of the registered tape blocks and prove consistency. With fresh encryption randomness v_{enc} (distinct from v_i), the sign convention is

$$\text{Enc}_{\text{sm}}(m) = (b v_{\text{enc}} + e_0 + \text{Emb}(m) + \xi, a_{\text{pk}} v_{\text{enc}} + e_1).$$

Privately transfer each $w_{i \rightarrow j,o}$ to j with its stored reference.

R3: receipt status and relinearization key. If the R2 receipt checks pass, broadcast and prove, under the original handles and exact aggregates,

$$k_i = (\bar{h}_0 - A) s_i^{\text{tmp}} + \bar{h}_1 u_i + \eta_i.$$

Publish each combined k_i , without separately publishing its two secret products, and set $K = \sum_i k_i$. All required statuses and proofs must pass. Using the relinearization key (K, A) , locally evaluate the fixed circuit

$$\mathcal{C}_{\text{KG}}(\omega_1, \dots, \omega_N) = \text{KG}^*(\omega_1 \oplus \dots \oplus \omega_N).$$

Its outputs are the required secret objects and final public/evaluation key bytes, not the key-generation errors or random tape. Local evaluation adds no message wave.

Protocol Π_6 : rounds 4–6

R4: output opening and commitments to mask and zero shares.

Set $z_{i,o} = \sum_j w_{j \rightarrow i,o}$. On the prescribed evaluated ciphertext handles only, broadcast and prove

$$p_{i,s} = c_{1,s} s_i^{\text{tmp}} + \text{Emb}_s(\text{fl}_i) + z_{i,s}, \quad p_{i,o} = c_{1,o} s_i^{\text{tmp}} + z_{i,o}$$

for the secret and public handles, respectively. For the sharing step, write j for the dealer and i for the recipient. Dealer j independently samples $A_{j,\ell}, B_{j,\ell} \xleftarrow{\$} \mathcal{R}_q$ and forms

$$\tilde{K}_j(Y) = -\text{fl}_j + \sum_{\ell=1}^{t-1} A_{j,\ell} Y^\ell, \quad Z_j(Y) = \sum_{\ell=1}^{t-1} B_{j,\ell} Y^\ell.$$

Commit $x_{j,i} = \tilde{K}_j(\alpha_i)$ and $y_{j,i} = Z_j(\alpha_i)$; prove degree, evaluation, and constant-term consistency, linked to the same source-mask handle. No numerical $x_{j,i}, y_{j,i}$ is transferred in this round. After verification, decode **pub** and $m = s + \sum_i \text{fl}_i \in \mathcal{R}_q$.

R5: private transfer of mask and zero shares. After locally accepting all required R4 commitments and consistency proofs, dealer j privately transfers the already registered $(x_{j,i}, y_{j,i})$ to recipient i . Thus corrupted dealers fix their polynomials before honest evaluations are disclosed. Check the stored openings and retain self-contributions locally.

R6: final status. Broadcast the prescribed receipt/verification status. Party i retains

$$f_i = m + \sum_j x_{j,i}, \quad g_i = \sum_j y_{j,i}.$$

Delivery of its R6 completion notification queues a key-ready notice; delivery of the latter enables use of the stored pair. The first delivered R6 completion notification fixes success, independently of other recipients' deliveries. A required-message or verification failure terminates locally; no dealer is dropped and no sharing polynomial changes. Pending is not success. The final-share proofs below are carried in this same wave.

Simulator $\mathcal{S}_6$

1. **Start and R1–R3.** Match every **Started**(i) registration; unless locally terminal, start R1. Generate honest temporary states, actual coin openings, valid relinearization-key contributions and corrupted-column noise coordinates. At honest tape positions, use zero encryptions and value-independent honest receipts.
2. **KeyCandidate(pub) (R4).** Sample $m \xleftarrow{\$} \mathcal{R}_q$; apply **SimOpen** of (51) to m and public targets. Supply mask- and zero-sharing receipts without

numerical transfers. Check rushing corrupted messages against registered state and mirror failure.

3. **ShareCandidate**($f_{\text{corr}}, g_{\text{corr}}$) **(R5)**. Sample the full table by (52); release entries in actual transfer order. Honest-to-honest transfers expose only metadata; preserve complaints and releases.
4. **Round completion and output delivery**. Before abort, call **Done**(i, r) on each delivered round- r completion notification, before the next wave ($r < 6$) or readiness queue ($r = 6$). Mirror individual readiness delivery by **ReleaseKey**(j); call **ReleaseAbort**(j) only on j 's local failure notice.
5. **Failure and publication**. Own sampling/domain failure leaves the key-generation instance pending. Otherwise, the first verified corrupted-message failure invokes **Abort** at any stage. Match **PublishKey** to its wrap-per event, even after abort.

Simulator conventions. Run $\mathcal{A}$, forward its interaction with $\mathcal{Z}$, and emulate the session-aware resources, recording and checking corrupted submissions. Match metadata, invalidity, omissions and pending round completion notifications; candidates follow actual openings, not mere eligibility. Charge the simulator's own sampling/domain failures to Theorem 7.2; continue permitted partial executions and late starts without post-abort **Done** calls or resampling. The simulator receives no s, F, G, f_H, g_H or canonical honest masks. The full sampled application record exists only in the proof coupling, which supports subsequent use of the stored shares and gives the simulator no additional information. The simulator's own temporary secrets are distinct from the unknown secret in the temporary-HE reduction.

C.5.4 Proofs of final shares in round 6 After a successful R5, party i has its complete ordered incoming references, including its self-contribution. It samples fresh independent ϑ_i and forms the commitment in (D1). The R6 $\mathcal{F}_{\text{VCP}}$ relation certifies the second equality against exactly those original source references, with canonical encodings and the exact session, key, stage, and identity. The new coins are registered only for this source relation. It broadcasts $(\text{ok}, \text{sid}, \text{kid}, i, C_i, \sigma_i)$, where σ_i denotes that relation's authenticated receipt. This message depends only on R5 state and fresh local coins, so it adds no round. An R6 completion notification is delivered only after every prescribed status and source proof verifies, with exactly one C_i per identity. The first such delivery freezes **KeyCom**; new source proofs, updates, and transfers are then rejected, while historical receipts remain verifiable for later recipients. Each party retains (f_i, g_i, ϑ_i) locally. Failure uses the existing share-stage abort gate. No erasure or adaptive-security claim is made. These proofs use the declared commitment setup and hiding/binding assumptions of the malicious threshold-decryption theorem; those losses are not included in (19).

D Security Proofs and Auxiliary Profiles

This appendix gives the complete security proofs for threshold decryption and the auxiliary distributional arguments used by the main-body DKG chain. The encoded-target core and the exact-output malicious extension use distinct ciphertext and output interfaces. The functionality releases the decryption value v_τ ; in the CKKS core this is the encoded target.

Table 5: Error bounds and their roles. Computational advantages, statistical distances, failure probabilities and numerical tolerances are distinct.

Symbol	Meaning
$\varepsilon_{uc}, \varepsilon_{online}$	Total online UC advantage and its upper bound.
$\varepsilon_{DKG}, \varepsilon_{total}$	Total DKG bound for keys, shares and transcript; total bound for DKG composed with malicious ThFHE.
$\text{Adv}_{tmp}, \text{Adv}_{aux}$	Computational bounds: temporary-HE IND-CPA with its full public evaluation key; auxiliary RLWE with the shared secret $u \leftarrow \chi_u$.
Adv_{lift}	Distinguishing bound $\sum_j \text{Adv}_{RLWE,j}^{(1)} \leq J\text{Adv}_{aux}$ for simulating auxiliary samples from the public key; each summand is a single-sample auxiliary-RLWE bound.
$\varepsilon_{hide}, \varepsilon_{bind}$	Computational bounds for honest-commitment hiding and obtaining distinct valid openings of one commitment.
$\varepsilon_\chi, \varepsilon_{prim}$	Failure probabilities: an RLWE error exceeds B_χ ; some zero share has $\gcd(q, g_0, \dots, g_{n-1}) > 1$.
$\varepsilon_{noise}, \varepsilon_{cross}$	Statistical bounds for proof-only noise and the polynomial change at release, including the primitive-mask failure for $k > 0$ and $k = 0$, respectively.
ε_{RLWE}	Computational bound: sum of the k online RLWE advantages.
$\varepsilon_{lift}, \varepsilon_\delta, \varepsilon_{evk}$	Statistical distances for hiding the public-sample correction, R3 correction and evaluation-key errors with independent noise.
$\varepsilon_{in}, \varepsilon_{out}$	Statistical distances for shifting input ciphertext distributions and hiding remaining errors in output openings.
$\varepsilon_{HE}, \varepsilon_{tail}$	Probability bounds for temporary-HE correctness failures, and for honest samples or shifts outside their specified bounds. Output failures are included only in ε_{HE} .
ε_{smp} (malicious DKG)	Sum of the four sampling distances below.
$\varepsilon_{pre}, \varepsilon_{KG}, \varepsilon_{post}, \varepsilon_{red}$	Statistical distances for sampling in the real protocol, key generation, simulator and reductions, counting all uses.
$\varepsilon_{crypt}, \varepsilon_{back}$	Combined computational/statistical DKG and ciphertext-replacement bounds, included in the total.
ε_X	Statistical distance for comparison X ; θ_X bounds the probability that its shift exceeds the stated bound.
$\varepsilon_B, \varepsilon_{priv}$	UC losses of the external additive-key backend and authenticated private channels for the complete setup.

Continued on the next page

Table 5 (continued)

Symbol	Meaning
$\varepsilon_{\text{compat}}$	Additional loss for simulating the backend state and transcript, including incomplete executions, from the specified setup interface.
$\varepsilon_{\text{samp}}, \varepsilon_{\text{KG}}$ (backend com- position)	Share sampling distance plus the backend key-sampling distance or failure bound ε_{KG} ; distinct from malicious-DKG terms.
ε_{add}	UC distinguishing loss for the separate additive N/N online profile.
ε_{cor}	Probability bound for any allowed ciphertext and qualified set violating correctness during the execution.
$\varepsilon_{\text{enc}, \tau}, \varepsilon_{\text{dec}, \tau}$	Numerical errors: encoded target versus intended result; threshold output versus encoded target. Unindexed forms give common bounds.

Bounds use the execution budgets in their theorems; sums already include their component terms. In particular, ε does not always denote a statistical distance. The symbols α_i are reserved for evaluation points.

The remaining δ symbols denote the R3 corrections $\delta_h, \delta_{h,j}$ or additive shifts and their coordinates in the statistical-distance bounds.

D.1 Threshold-decryption security proof

We first prove the sharing and boundary lemmas used in [Theorem 7.1](#). Conditioned zero-sharing masks and release-time interpolation also underlie KLL⁺26’s proof [[17](#), Thm. 4.2]; our boundary couplings handle deterministic compressed packets.

Proposition D.1 (Ciphertext privacy). *If the underlying FHE is IND-CPA secure with its public evaluation keys, then Jambon satisfies [Definition 8](#) with the same advantage bound.*

Proof. For every fixed s , [Lemma 4](#) makes f_C and g_C independent uniform vectors, independently of the randomness generating the public and evaluation keys. A base-FHE IND-CPA reduction therefore samples these corrupted shares uniformly and forwards the challenge unchanged. The mask oracle is independent of the base FHE and is answered consistently by lazy sampling. This experiment has no honest partial-decryption oracle. $\square$

Lemma 8 (Publication and readiness invariants). *For every prefix of an execution of [Definition 11](#), let E_τ be the honest indices with a committed packet for τ and Resp_τ its packet table. Then*

$$\text{Seen}_\tau = C \cup E_\tau, \quad D_{\tau,j} \subseteq \text{dom}(\text{Resp}_\tau) \subseteq \text{Seen}_\tau.$$

Each stored response is immutable and identical at all recipients that receive it. The first release of v_τ to the simulator coincides with installation of the $(t - |C|)$ -th distinct honest packet. No pending ticket, alias, cancellation, or rejected publication changes these statements. In particular, a ready final-decryption request implies that v_τ has been released to the simulator.

Proof. Initially $\text{Seen}_\tau = C$ and both packet and receipt tables are empty. A successful honest publication inserts exactly one previously absent honest index and its packet in one transition; corrupted publications add packets only at indices already in C . A delivery or self-receipt copies a stored packet, and retries never create a second packet for that index. Binding and cancellation do not remove any published entry or reset a canonical record, while a stale or invalid commitment changes no state. Induction on the command sequence gives the inclusions and immutability. Since $|C| < t$, the first transition to size t is an honest publication, and t distinct received indices cannot precede it. $\square$

Lemma 9 (Conditioned sharing kernel). *Let G_C^0 be the interpolant of degree at most c satisfying $G_C^0(0) = 0$ and $G_C^0(\alpha_i) = g_i$ for $i \in C$. Conditioned on g_C , the zero-sharing polynomial has the unique representation*

$$G(Y) = G_C^0(Y) + W_C(Y)U(Y), \quad W_C(Y) := Y \prod_{i \in C} (Y - \alpha_i),$$

where $U(Y) = \sum_{j=0}^{k-1} \beta_j Y^j$ has independent uniform coefficients in $\mathcal{R}_q$. Moreover, for $i \notin C$ and $\|E_j\|_\infty \leq B_\chi$,

$$\left\| W_C(\alpha_i) \sum_{j=0}^{k-1} \alpha_i^j E_j \right\|_\infty \leq k \Gamma_c B_\chi.$$

Proof. Successive division of $G - G_C^0$ by its monic linear factors is exact: the other evaluation differences are units, so each remaining root survives a preceding division. Conversely, every U of degree at most $k - 1$ (or $U = 0$ for $k = 0$) gives a polynomial with the required evaluations. This is a bijection from the uniform conditional sharing fibre to $\mathcal{R}_q^k$, including the singleton fibre when $k = 0$. Finally, $W_C(\alpha_i)$ is a signed monomial times $\prod_{u \in C} (1 - \alpha_i \alpha_u^{-1})$. Multiplication by α_i^j preserves coefficient norms, and Lemma 2 gives the stated bound. $\square$

Lemma 10 (Coefficient-primitive masks). *For $g \in \mathcal{R}_q$, the coefficient-primitive condition is*

$$\gcd(q, g_0, \dots, g_{n-1}) = 1.$$

For such g and uniform $h \in \mathcal{R}_q$, every coefficient of hg is uniform in $\mathbb{Z}_q$. In the prescribed setup, the probability that any g_i is not coefficient-primitive is at most $\varepsilon_{\text{prim}}$.

Proof. A coefficient of hg is a linear form in the uniform coefficients of h , whose weights are a signed permutation of the coefficients of g . Its image is the ideal

they generate in $\mathbb{Z}_q$; the gcd condition makes this ideal $\mathbb{Z}_q$. Surjectivity gives uniformity because every fibre of a finite-group homomorphism has the same size. By [Lemma 4](#), each g_i is uniform in $\mathcal{R}_q$. For every prime $\ell \mid q$, the probability that all its n coefficients vanish modulo ℓ is ℓ^{-n} . A union over the distinct prime divisors and party indices proves the bound. The claim is marginal uniformity, not independence across coefficients or parties; a ring unit is sufficient but unnecessary. $\square$

Lemma 11 (Rounding-boundary coupling). *If a scalar residue $v \bmod L$ is uniform, the probability that some integral shift of magnitude at most B changes $\lfloor v \rfloor_r$ is at most $\beta_L(B)$. The shift may depend on v .*

Proof. There is one rounding boundary in each period of length L . Its two-sided radius- B neighbourhood contains at most $2\lceil B \rceil + 2$ integral residues; outside it every permitted shift stays in the same cell. The case $B = 0$ is exact. For a vector of unrounded values, a union bound applies to the marginal probabilities without any independence assumption. $\square$

Proof (Proof of [Theorem 7.1](#)). We construct one simulator for each adversary, independently of the environment, then compare complete executions by hybrids. Setup and admission are the exact interfaces in the theorem, not additional protocols whose security is inferred here.

Simulator and atomic publication. The simulator $\mathcal{S}$ runs $\mathcal{A}$ and forwards the public key and the actual corrupted states supplied by the ideal setup; it does not resample these states. It uses the given random-oracle resource without programming it and mirrors all local bindings, ticket serials, cancellations, and channel deliveries. For each label τ , it stores the committed unrounded values $u_{j,\tau}$ of honest packets, the published packets, and eventually a single polynomial P'_τ . Write $h_\tau = H_0(\tau)$ and define $\text{EncTarget}_\tau(v) = (q/p)v$ for BfV. When the simulated real wrapper is activated to publish a live honest job $e = (\tau, i, \text{serial})$, read $T = \text{Seen}_\tau$ before that publication and proceed as follows.

- (1) If P'_τ is not fixed and $|T| < t - 1$, sample a provisional $u \xleftarrow{\$} \mathcal{R}_q$ and submit the constant circuit $M_e(w) = \lfloor u \rfloor_r$ in $(\text{CommitPart}, e, T, M_e)$. Store $u_{i,\tau} = u$ only after successful commitment.
- (2) If P'_τ is not fixed and $|T| = t - 1$, form the polynomial-valued affine map $\Phi_{\tau,T}(Y; v)$ specified by

$$\begin{aligned} \Phi_{\tau,T}(0; v) &= b_\tau - \text{EncTarget}_\tau(v), \\ \Phi_{\tau,T}(\alpha_j; v) &= a_\tau f_j + h_\tau g_j & (j \in C), \\ \Phi_{\tau,T}(\alpha_j; v) &= u_{j,\tau} & (j \in T \setminus C). \end{aligned}$$

Submit the circuit

$$M_e(v) = \lfloor \Phi_{\tau,T}(\alpha_i; v) \rfloor_r, \quad M_e(\perp) = 0.$$

All interpolation data except the circuit input v are already known, so at this first threshold crossing the simulator submits the circuit before learning

- v_τ . The functionality evaluates $M_e(v_\tau)$, stores and exposes the packet, and releases v_τ to the simulator in the same transition. Upon this opening set $P'_\tau = \Phi_{\tau,T}(Y; v_\tau)$ and store its unrounded evaluation at α_i .
- (3) If P'_τ is fixed, commit the constant circuit $M_e(w) = \lfloor P'_\tau(\alpha_i) \rfloor_r$ and retain its unrounded value upon success. Already published packets are replayed without a new circuit or gate event.

A rejected or canceled commitment never enters the table of committed unrounded values; any stale proposal is discarded and recomputed for a live ticket and the current T . The circuits are polynomial-size: interpolation uses the public inverse Vandermonde matrix over $\mathcal{R}_q$, and the remaining arithmetic and rounding have polynomial bit complexity. Their oracle values are hardwired public constants; they contain no honest shares, hidden phase, or oracle gates. The $\perp$ branch of the crossing circuit is never used after a successful expected-state check. At the first threshold crossing, this computes the response and installs the packet atomically with the release of v_τ , without rewinding or releasing v_τ for an unfinished ticket. By Lemma 8, exactly k distinct honest packets precede crossing, including none when $k = 0$; C is never enlarged artificially. Corrupted responses use the supplied corrupted states and the prescribed deterministic algorithm, exactly as in the passive real execution. Final outputs require no message from $\mathcal{S}$: the functionality uses its stored packets and output profile.

Hybrid $H_0 \rightarrow H_1$: proof-only noise. Hybrid H_0 is the real execution. Use Lemma 9 and, at each label's first query, associate fresh errors $E_{\tau,j} \leftarrow \chi$ and set $T_{\tau,j} = h_\tau \beta_j + E_{\tau,j}$. Hybrid H_1 replaces each honest unrounded value by

$$a_\tau f_i + h_\tau G_C^0(\alpha_i) + W_C(\alpha_i) \sum_{j=0}^{k-1} \alpha_i^j T_{\tau,j},$$

then applies the unchanged rounding operation. The difference has coefficient norm at most B_{mask} , except with probability ε_χ . When $k > 0$, condition on the setup and the history preceding a label's first query, before sampling h_τ . The entire ciphertext is already fixed by its canonical label. Outside the setup event of Lemma 10, every coordinate of the original unrounded value $a_\tau f_i + h_\tau g_i$ is uniform in $\mathbb{Z}_q$, hence modulo L . Mark it bad if any shift bounded by B_{mask} can cross a rounding boundary. Lemma 11 and a union over all candidate labels, parties, and coefficients bound this transition by

$$\varepsilon_{\text{prim}} + \varepsilon_\chi + q_H N n \beta_L(B_{\text{mask}}).$$

The union includes discarded candidates and values never ultimately exposed. It therefore covers later adaptive selection without assuming uniformity after a hash answer or response has been observed. Couple executions until the first differing exposed byte; outside the marked events the schedules and subsequent decisions agree. When $k = 0$, this transition is the identity: there is no proof-only noise, and no primitive-mask exception is charged here.

Hybrid $H_1 \rightarrow H_2$: replacing the mask kernel. Replace the columns $(T_{\tau,j})_\tau$ one at a time by independent uniform $(Z_{\tau,j})_\tau$. For column j , a standard

RLWE reduction uses its samples $(h_\tau, T_{\tau,j})$, samples earlier columns uniformly, and computes later columns using independently sampled secrets on the same h_τ . It assigns the sample's first component to $H_0(\tau)$ only at first query and records it for reuse on subsequent queries. This gives the joint random-oracle distribution exactly and costs at most $k \text{Adv}_{\text{RLWE}}(T_{\text{red}}, q_H)$.

Unlike the ideal simulator, this reduction emulates setup internally: it generates s , F , and all base-FHE public and evaluation keys, and samples g_C and its interpolant G_C^0 . Only the embedded residual mask secret is unknown. Its corrupted shares have the exact conditional setup distribution because W_C vanishes on $\{0\} \cup C$. It can preserve every base-FHE side-information value and simulate the subsequent adaptive execution while changing no ciphertext component a_τ or b_τ . No uniformity assumption on evaluated ciphertexts or auxiliary-leakage assumption on the embedded RLWE secret is used. When $k = 0$ this hybrid transition is empty.

In $\mathbf{H}_2$ the unrounded responses are evaluations of

$$P_\tau(Y) = a_\tau F(Y) + h_\tau G_C^0(Y) + W_C(Y) Z_\tau(Y), \quad Z_\tau(Y) = \sum_{j=0}^{k-1} Z_{\tau,j} Y^j.$$

At first query, conditional on the earlier history and the setup, Z_τ is fresh uniform. Consequently, P_τ is uniform subject to

$$\deg P_\tau < t, \quad P_\tau(0) = a_\tau s, \quad P_\tau(\alpha_j) = a_\tau f_j + h_\tau g_j \quad (j \in C).$$

Subtracting the interpolant through $\{0\} \cup C$ leaves W_C times a uniform polynomial; the residual contribution of F only translates it. Fresh residual coefficients are independent across labels, although ciphertexts and their fixed constraints may depend on the earlier transcript.

Hybrid $\mathbf{H}_2$ versus the simulator. Before crossing there are k free ring coefficients. Evaluation at any at-most- k distinct honest indices is surjective: the factors $W_C(\alpha_i)$ are units and the remaining evaluation matrix is Vandermonde with unit differences. Thus sequential uniform sampling of unrounded values is exactly deferred sampling of P_τ , even with adaptive index choices. Keeping unrounded values, rather than only their rounded images, preserves the correct conditional distribution for later responses.

Couple these values with those recorded by $\mathcal{S}$. At the successful crossing commitment, $P'_\tau = \Phi_{\tau,T}(Y; m_\tau)$ and P_τ agree on T , while their constants differ by e_τ . Uniqueness of interpolation gives the identity in $\mathcal{R}_q[Y]$

$$P'_\tau(Y) - P_\tau(Y) = e_\tau \prod_{j \in T} (1 - Y \alpha_j^{-1}).$$

At every remaining party this shift has an integral lift of coefficient norm at most B_{out} . At first query, mark each original coordinate of $P_\tau(\alpha_i)$ bad if any shift of this magnitude can cross a rounding boundary. For $k > 0$, a single honest evaluation is uniform in $\mathcal{R}_q$ by kernel surjectivity, so the probability of a

marked coordinate is bounded by [Lemma 11](#) without a primitive-mask exception. For $k = 0$, $H_2 = H_0$; fresh h_τ and [Lemma 10](#) give the same bound outside an event of probability at most $\varepsilon_{\text{prim}}$. The universal shift event protects against every possible crossing set T , including one chosen from the responses; no union over crossing sets is needed. Hence this comparison costs at most $q_H N n \beta_L(B_{\text{out}})$ when $k > 0$, and at most that quantity plus $\varepsilon_{\text{prim}}$ when $k = 0$. All old, crossing, and subsequent response bytes then agree.

Outputs and UC scheduling. It remains to match honest outputs, not merely response transcripts. In H_2 , the identity $P_\tau(0) = a_\tau s$ gives the aggregation equation (28). The pointwise correctness proof in [Theorem 5.1](#) applies to these auxiliary unrounded polynomials by [Lemma 1](#). Every qualified set therefore returns m_τ , for arbitrary rounding residues within their deterministic bounds. Outside the boundary events, simulated bytes equal these bytes, so public re-computation agrees with every ideal output. No independence claim about the rounding residues is used.

The bookkeeping of setup, registration, local contexts, tickets, and receipts is identical in the coupled executions. A publication circuit is an internal description of the simulator’s next packet, not an extra protocol message; successful commitments match real local send events. Before such a send, cancellation may remove a pending job without exposing a value. After it, cancellation cannot remove its packet, privacy count, self-receipt, queued deliveries, or an already obtainable output. Stale serials and expected-state mismatches are rejected before v_τ is read. Local identifiers never create fresh cryptographic labels, so retries after cancellation and aliases retain the same unrounded and rounded values. By [Lemma 8](#), every recipient’s ready set is a subset of committed packets, even when recipients use different sets or receive them in different orders. Thus identical packets imply identical readiness, failed-request responses, local combining outputs, and subsequent public scheduling decisions. The environment may re-compute **Combine** from every public packet; it need not trust the functionality’s output announcement. No fairness assumption is used for indistinguishability: a permanently delayed job or message simply gives the corresponding incomplete prefix on both sides. The conditional availability claim in [Remark 2](#) follows separately from the wrapper’s local final-decryption rule, not from UC security alone. The same straight-line $\mathcal{S}$ works for every environment; reductions may run the environment to distinguish hybrids, not to construct $\mathcal{S}$. For $k > 0$ the primitive-mask exception occurs only in the first transition, while for $k = 0$ it occurs only in the last comparison. Thus it is charged once, and the triangle inequality gives (18). $\square$

Corollary D.2 (Encoded-target CKKS profile). *Use the numerical decoding map and deterministic conditions of [Corollary 5.1](#), with the registered CKKS targets of [Section 5.2](#). Replace the BFV interface and correctness hypotheses of [Theorem 7.1](#) by [Definition 7](#) and (10)–(11), keeping the same setup, oracle, and RLWE hypotheses. Then the Jambon wrapper realizes*

$$\mathcal{F}_{\text{ThFHE}}^{\text{str}}[\text{ThKeyGen}, N, t, \text{ckks}, \emptyset]$$

with realization error bounded by (18), where B_e bounds the coefficient error in $b_\tau - a_\tau s = \hat{m}_\tau + e_\tau \pmod{q}$.

Proof. Use $\text{EncTarget}_\tau(v) = v$ in the crossing circuit, so $P'_\tau(0) = b_\tau - \hat{m}_\tau$. The polynomial difference is still $e_\tau \prod_{j \in T} (1 - Y \alpha_j^{-1})$, and the first threshold-crossing circuit is constructed before $\hat{m}_\tau$ is released. Thus every response comparison and the bound (18) are unchanged; no plaintext modulus is needed. For a ready output request, the functionality applies the public finite-precision decoding map to its immutable response table and checks OutOK_τ . It neither asks the simulator to approve an output nor samples fresh error. In H_2 , the CKKS correctness proof applies pointwise to the underlying sampled polynomial by Lemma 1. Outside the boundary events, the recomputed value therefore satisfies OutOK_τ simultaneously for every delivered set and equals the real recipient's local output. Recording and reusing outputs under (τ, S) gives identical outputs for identical packet sets across recipients, while distinct sets may give different vectors within the permitted tolerance. By Lemma 8, a ready set implies that v_τ has been released to the simulator, including after local cancellation; there is no additional early leakage from the relation test. On a bad execution the functionality is still defined and may return $\perp_{\text{badtr}}$; this discrepancy is covered by the same realization bound. There is no additional probabilistic precision event under the deterministic hypotheses. $\square$

Remark 6 (Scope and parameter cost). The reduction for the zero-sharing coefficients uses uniform-secret RLWE; ciphertext privacy retains the base FHE assumptions in Proposition D.1. The theorem covers static semi-honest adversaries making classical queries to independent random oracles for distinct key-generation instances; it makes no claim for a shared global oracle or quantum superposition queries [7]. For each polynomial execution budget, the chosen q/r must make the boundary terms negligible; $T_j \leq 2^j$ does not establish polynomial threshold overhead. The setup exception is bounded by $\varepsilon_{\text{prim}}$ as a function of the actual moduli and ring degree, not by a universal numerical constant. CKKS security is relative to the encoded-target functionality, not to approximate-output-only leakage. Its no-wrap and precision conditions must hold together with the security constraint on q/r . If the pipeline's admissibility bounds have a stated total failure probability, add that probability to the realization bound. Replacing setup by a DKG requires a separate realization of the same instance of Definition 9, including its public transcript and delivery/failure semantics; that obligation is left to the DKG security analysis. The theorem does not cover simulator-chosen replacement key views, unanimity-based context agreement, public verification of malicious responses, or a network-wide abort protocol. Adding any such feature requires a matching real protocol and an extension of the simulator, rather than a change of functionality alone.

D.2 DKG Security Analysis

We prove the sharing distribution and simulate the dealing transcript, then state the backend guarantees required for composition. Corruptions remain static.

D.2.1 Output distribution and retained dealing state

Proof (Proof of Theorem 6.1). Fix the backend randomness and outputs, including $s, s_1^{\text{add}}, \dots, s_N^{\text{add}}$, before resharing and subsequent delivery decisions. All parties follow the semi-honest sampling rules. By (12),

$$\sum_j K_j(Y) = s + \sum_{\ell=1}^{t-1} \left(\sum_j A_{j,\ell} \right) Y^\ell, \quad \sum_j Z_j(Y) = \sum_{\ell=1}^{t-1} \left(\sum_j B_{j,\ell} \right) Y^\ell.$$

Each coefficient sum contains an independent uniform summand; all sums are independent of one another and of the backend record. Thus the target distribution holds, with its prescribed constant s , and local aggregation evaluates these polynomials irrespective of delivery order. Presampling all coins proves the identity even for incomplete executions; it makes no claim about the distribution conditioned on completion. $\square$

Remark 7 (Resharing using a masked opening). For Lemma 7, replace $\sum_j K_j(Y)$ by $\mathbf{m} + \sum_j \tilde{K}_j(Y)$. The output-distribution proof and completion count below are unchanged: fixed constants merely translate the evaluation constraints. Once $\mathbf{m}$ and corrupted masks are available, the sampler uses $f_C - \mathbf{m}\mathbf{1}$ for the corrupted evaluations. This does not simulate messages sent before opening $\mathbf{m}$. Such a schedule requires a causal simulator for the enlarged backend view, including the opening and corrupted masks at their specified release times; security for the original ξ_C alone does not supply it. Accordingly, Theorem D.1 proves composition for subsequent dealing. Early mask sharing retains these additional, unproved simulation obligations.

Lemma 12 (Exact completion count and dealing-view simulator). *Put $m = t - 1$, $c = |C| < t$, and $h = N - c$. Let $A_C : \mathcal{R}_q^m \rightarrow \mathcal{R}_q^c$ evaluate a zero-constant polynomial at the corrupted points. For either coefficient family, fix its aggregate coefficient vector, the corrupted dealers' coefficient vectors, and every honest dealer's evaluation vector at C . If their sums are consistent, the number of honest coefficient completions is exactly*

$$|\mathcal{R}_q|^{(m-c)(h-1)}. \quad (33)$$

Consequently the complete corrupted dealing view has a perfect polynomial-time sampler from (ξ_C, f_C, g_C) without requesting an additional secret or honest-share input. Conditioned on the presampled corrupted-visible algebraic dealing table and corrupted tapes, the residual zero-sharing polynomial is still uniform on the fibre specified by g_C . This conditioning excludes later response-dependent scheduling events, which are handled by the interactive transcript coupling.

Proof. By the invertible evaluation differences and Lemma 3, the first c columns of A_C form an invertible matrix. Thus A_C has a computable right inverse and kernel size $|\mathcal{R}_q|^{m-c}$, including the empty constraint when $c = 0$.

After subtracting corrupted coefficient vectors, each of $h - 1$ honest dealers has $|\mathcal{R}_q|^{m-c}$ coefficient choices with its prescribed evaluations. The aggregate sum fixes the last vector, which satisfies its constraints precisely when the evaluation sums are consistent. This proves (33), also for $c = m$ or $h = 1$. For key dealing, first subtract $s_j^{\text{add}} \mathbf{1}$ from dealer j 's evaluation vector; fixed constants therefore do not affect the count.

Here is the sampler for the visible transcript. First sample the corrupted dealers' coefficient vectors independently and uniformly, retain their full tapes, and compute their outgoing key and zero messages using their actual additive shares $s_C^{\text{add}} = (s_j^{\text{add}})_{j \in C}$. For the h honest incoming key vectors at C , sample $h - 1$ independent uniform vectors in $\mathcal{R}_q^c$ and determine the last by

$$\sum_{j \notin C} x_{j,C} = f_C - \sum_{j \in C} (s_j^{\text{add}} \mathbf{1} + A_C A_j).$$

Independently do the same for zero messages with prescribed sum

$$\sum_{j \notin C} y_{j,C} = g_C - \sum_{j \in C} A_C B_j.$$

Here $A_j = (A_{j,\ell})_\ell$ and $B_j = (B_{j,\ell})_\ell$ are coefficient vectors, and $x_{j,C}, y_{j,C}$ collect dealer j 's messages to corrupted recipients. The sampler uses no honest additive share s_j^{add} . For each fixed aggregate polynomial and backend record, the completion count makes all consistent incoming tables equiprobable. Their distribution depends on F, G only through f_C, g_C , and on the backend only through its declared corrupted state. Corrupted coefficients remain uniform because an honest dealer can supply the aggregate remainder. Applying this independently to both families also proves conditional uniformity of the residual zero polynomial given corrupted tapes and incoming messages.

Sample this table once and release entries at their scheduled deliveries. Corrupted outgoing messages follow from their tapes; private honest-to-honest contents remain hidden, while lengths, endpoints and delivery effects are simulated. Every corrupted intermediate state is determined by its tape and delivered entries, so the full-record coupling covers adaptive schedules, withheld messages and interleaving without conditioning on completion. Independent instances use fresh coins; aliases reuse the stored table. Static corruption requires no erased-state assumption. $\square$

Remark 8 (Additive zero dealing). For the additive profile, replace the polynomial module by $V = \{y \in \mathcal{R}_q^N : \sum_i y_i = 0\}$. Projection $V \rightarrow \mathcal{R}_q^c$ is surjective, with kernel size $|\mathcal{R}_q|^{N-c-1}$, since at least one coordinate lies outside C . The same sum-and-fibre argument gives $|\mathcal{R}_q|^{(N-c-1)(h-1)}$ completions after fixing the aggregate and corrupted rows. The simulator samples corrupted zero rows, samples $h - 1$ honest incoming vectors, and fixes the last by their sum g_C . Consequently the residual additive zero vector is uniform conditional on g_C and the corrupted-visible algebraic dealing table and tapes. This is not a

posterior-uniformity assertion after online responses or response-dependent delivery decisions. This claim does not require uniformly distributed backend key shares.

D.2.2 Composition and backend requirements

Theorem D.1 (Conditional DKG composition). *Assume the backend realizes the additive-key-generation component of Definition 14, including its specified retained state and behavior at every execution prefix, with error ε_B , and the authenticated private channels have realization error $\varepsilon_{\text{priv}}$ for the complete setup. The dealing extension supplies the independent F, G sharings and the dealing-readiness gates; these are not assumed as backend outputs. It has no additional simulation error with exact sampling. It realizes the setup resource $\mathcal{F}_{\text{KG}}[\text{ThKeyGen}_B, \xi_B]$ of Definition 14 with error at most*

$$\varepsilon_B + \varepsilon_{\text{priv}} + \varepsilon_{\text{samp}}.$$

Here a sampling or failure term already charged to the backend is not charged twice. Under the algebraic, admission, RLWE, and correctness hypotheses of the original strong-threshold theorem, the corresponding backend-adapted Ring-Shamir functionality $\mathcal{F}_{\text{ThFHE}}^{\text{str}}[\text{ThKeyGen}_B, N, t, \Pi, \xi_B]$ for $\Pi \in \{\text{bfv}, \text{ckks}\}$ is realized with error at most

$$\varepsilon_B + \varepsilon_{\text{priv}} + \varepsilon_{\text{samp}} + \varepsilon_{\text{online}}, \quad (34)$$

where $\varepsilon_{\text{online}}$ is the bound of Theorem 7.1, or its encoded-target CKKS counterpart. If, in addition, the backend's key distribution is the original one and a simulator given the original setup outputs reproduces the additional backend view with error $\varepsilon_{\text{compat}}$, the original setup interface $\mathcal{F}_{\text{KG}}[\text{ThKeyGen}, \emptyset]$ and online instance $\mathcal{F}_{\text{ThFHE}}^{\text{str}}[\text{ThKeyGen}, N, t, \Pi, \emptyset]$ are obtained with that extra error. This simulator must reproduce the joint transcript at every stage of message delivery, including incomplete executions; matching completed outputs alone is insufficient.

Proof. In the backend/private-channel hybrid, apply Lemma 12 to the actual corrupted record at the release times supplied by the backend premise. The sampler preserves keys, corrupted states and every scheduled prefix without honest-share queries or fairness. Data processing carries the sampling discrepancy of Remark 4 to these views; replacing the backend and private channels gives the first bound.

For the online phase, condition on the full base-key/backend record generated by the reduction. The key polynomial is still a uniform sharing of s . By Lemma 12, the retained zero-dealing view can be sampled from g_C and is independent of the residual zero coefficients used in Lemma 9. The RLWE reduction can therefore generate all backend and key-side information itself and sample the zero-dealing view without knowing a challenged residual mask secret. The proof-only noise, column replacement, and first-query boundary couplings

of [Theorem 7.1](#) apply unchanged. The online simulator receives actual f_C, g_C through the declared interface; readiness, publication and delivery rules are unchanged. [Theorem 6.1](#) and the original bounds supply pointwise correctness, proving the online term in (34). Finally, the separately assumed simulator removes the additional setup view, when that guarantee is available. $\square$

The theorem separates a proved information-theoretic dealing layer from a conditional backend instantiation. In particular, a backend with $s_1^{\text{add}} = s$ and all other $s_j^{\text{add}} = 0$ satisfies the additive identity, and re-sharing still gives the exact final polynomial distribution, but a corrupted first party retains s . Thus the identity and output distribution cannot imply the original setup security for an arbitrary backend. Nor does strong online simulation by itself establish ciphertext privacy with the retained backend state. The concrete CKKS backend must supply those guarantees under its stated assumptions.

D.2.3 Deterministic correctness refinements

Proof (Proof of [Remark 5](#)). Under the unchanged coefficient no-wrap condition, [Lemma 1](#) gives the integral equality

$$z_S = D(\hat{m}_\tau + e_\tau) - \sum_{i \in S} \lambda_i^S \rho_i, \quad \|\rho_i\|_\infty \leq (L-1)/2.$$

Each embedding root has modulus one, hence $|\sigma_j(e_\tau)| \leq nB_e$ and $|\sigma_j(\rho_i)| \leq n(L-1)/2$. Divide the equality by $D\Delta_{\text{ct}\tau}$ and use the triangle inequality in each embedding. The residue contribution is at most $n(L-1) \sum_i |\sigma_j(\lambda_i^S/D)|/(2\Delta_{\text{ct}\tau})$. Taking the maximum over embeddings and adding the numerical error proves (29). Finally $|\sigma_j(\lambda_i^S)| \leq \|\lambda_i^S\|_1$ gives $\Lambda_{\text{can}}(S) \leq M_1(S)/D$. This argument uses no residue-variance estimate; the coefficient bound preventing wraparound is still required. $\square$

Proof (Proof of [Theorem C.1](#)). The rounding-residue identity gives

$$Lp_i \equiv af_i + H_0(\tau)g_i + \rho_i \pmod{q}, \quad \|\rho_i\|_\infty \leq (L-1)/2.$$

Sum over all parties and use $\sum_i f_i = s$ and $\sum_i g_i = 0$. This gives the claimed phase identity and aggregate residue bound $N(L-1)/2$. For BFV, the stated bound places the phase within half a plaintext spacing. For CKKS, (31) identifies the centered integral lift with the desired message plus error; applying $\|\sigma_{\text{slot}}(w)\|_\infty \leq n\|w\|_\infty$ and adding ν_τ gives (32). All claims hold pointwise, with no independence assumption about residues. $\square$

D.2.4 The additive all-party profile This is a separate additive N/N construction, not Ring–Shamir with $t = N$. It uses the setup of [Definition 15](#) and the publication/delivery rules of [Definition 11](#). The declared setup view includes corrupted backend state. Removing that additional view requires a simulator from (pk, f_C, g_C) and its separately charged error, as in [Theorem D.1](#); the additive identity alone establishes neither setup security nor ciphertext privacy.

Theorem D.2 (Strong security with additive shares). *Use the static semi-honest execution, admission, publication, and delivery interfaces of [Section 5.2](#), with threshold $t = N$ and the additive profile of [Definition 15](#). The setup samples an arbitrary joint key/backend record with $\sum_{i=1}^N f_i = s$ in $\mathcal{R}_q$. Independently, it samples a uniform additive zero sharing g satisfying $\sum_{i=1}^N g_i = 0$. It provides the actual corrupted shares and the declared retained state to the simulator. Conditioned on the algebraic part of the declared corrupted setup view, the base-key/backend record, and g_C , excluding response-dependent scheduling events, assume that g is uniform on the additive zero-sharing fibre with the prescribed g_C . In particular, independent zero dealing satisfies this conditional-distribution requirement even when corrupted dealer tapes and incoming dealings are retained, by the zero-dealing transcript simulation.*

Put $c = |C| < N$ and $k = N - c - 1$. Use q_H , B_χ , β_L , and $\varepsilon_{\text{prim}}$ from the preceding analysis, and let T_{red} bound the running times of the reductions below, including their simulation overhead, with

$$\varepsilon_\chi = q_H k \Pr_{E \leftarrow \chi} [\|E\|_\infty > B_\chi], \quad B_{\text{mask}} = kB_\chi, \quad B_{\text{out}} = B_e.$$

Assume uniform-secret decisional RLWE and the deterministic correctness conditions of [Theorem C.1](#). Then, in the declared setup/admission hybrid, the additive wrapper realizes $\mathcal{F}_{\text{ThFHE}}^{\text{str}}[\text{ThKeyGen}^{\text{add}}, N, N, \Pi, \xi_B^{\text{add}}]$ for $\Pi \in \{\text{bfv}, \text{ckks}\}$, with

$$\begin{aligned} \varepsilon_{\text{add}} \leq & k \text{Adv}_{\text{RLWE}}(T_{\text{red}}, q_H) + \varepsilon_\chi + \varepsilon_{\text{prim}} \\ & + q_H N n (\beta_L(kB_\chi) + \beta_L(B_e)). \end{aligned} \quad (35)$$

Replacing the declared setup hybrid by a DKG realization, with any required simulation of additional setup state, adds that realization's error, including any stated sampling or base-key-generation failure probability. The online simulator requests no secret or honest-share input beyond the declared corrupted setup view and requires no erasures.

Proof. We give the changes to the proof of [Theorem 7.1](#), checking the different sharing distribution and reconstruction rule. Let $H = [N] \setminus C$ and fix $i^* \in H$ for the proof. Conditioned on the declared algebraic setup view, the zero shares have the representation

$$g_j = \beta_j \quad (j \in H \setminus \{i^*\}), \quad g_{i^*} = - \sum_{u \in C} g_u - \sum_{j \in H \setminus \{i^*\}} \beta_j, \quad (36)$$

where the k residual values β_j are independent uniform ring elements. Thus the basis is $e_j - e_{i^*}$; no uniformity of the key shares f_i is required. The zero-dealing sampler of [Remark 8](#) generates the additional corrupted view from g_C without these residual values.

Simulator. Use the core simulator's state and scheduling rules and the given, unprogrammed oracle; put $h_\tau = H_0(\tau)$. For the first k honest publications, sample uniform unrounded values and retain them only after successful commitment.

For the last honest identity i , let $T = \text{Seen}_\tau$ immediately before its publication, so $|T| = N - 1$, and replace the interpolation circuit by

$$\begin{aligned} u_{i,\tau}(v) &= b_\tau - \text{EncTarget}_\tau(v) - \sum_{j \in C} (a_\tau f_j + h_\tau g_j) - \sum_{j \in T \setminus C} u_{j,\tau}, \\ M(v) &= \lfloor u_{i,\tau}(v) \rfloor_r, \quad M(\perp) = 0. \end{aligned} \quad (37)$$

Here $\text{EncTarget}_\tau(v) = (q/p)v$ for BFV and v for CKKS. The circuit is fixed before release and evaluated atomically with publication and release of v_τ . Thereafter every honest identity is already in Seen_τ , so only stored packets can be replayed. Corrupted responses use their actual shares; stale or canceled proposals neither alter stored values nor release v_τ .

Noise and RLWE replacements. At each label's first query sample $E_{\tau,j} \leftarrow \chi$ and put $T_{\tau,j} = h_\tau \beta_j + E_{\tau,j}$. The proof-only perturbed values are

$$\begin{aligned} v_{j,\tau}^{(1)} &= a_\tau f_j + T_{\tau,j} & (j \in H \setminus \{i^*\}), \\ v_{i^*,\tau}^{(1)} &= a_\tau f_{i^*} - h_\tau \sum_{u \in C} g_u - \sum_{j \in H \setminus \{i^*\}} T_{\tau,j}. \end{aligned} \quad (38)$$

The errors sum to zero and change each honest value by at most kB_χ outside the tail event. Each g_i is marginally uniform; Lemmas 10 and 11 therefore give $\mathbf{1}_{k > 0} \varepsilon_{\text{prim}} + \varepsilon_\chi + q_H N n \beta_L(kB_\chi)$. As in the core proof, mark all candidate labels and coordinates at first query, before later sender or completion choices; no posterior uniformity is assumed.

Use the core proof's shared-first-component RLWE hybrids to replace the k columns $(T_{\tau,j})_\tau$ by independent uniform columns $(Z_{\tau,j})_\tau$. This contributes at most $k \text{Adv}_{\text{RLWE}}(T_{\text{red}}, q_H)$ to the distinguishing advantage. The reduction generates the complete key/backend record and f , samples g_C , and uses the conditional zero-view sampler above. Thus its only unknown secret is the challenged β_j ; all key correlations and private-channel metadata remain simulatable without extra RLWE leakage assumptions. The resulting vector is uniform subject to

$$\sum_{i=1}^N v_{i,\tau}^{(2)} = a_\tau s, \quad v_{u,\tau}^{(2)} = a_\tau f_u + h_\tau g_u \quad (u \in C). \quad (39)$$

Indeed, nonpivot coordinates are independent translates of $Z_{\tau,j}$ and the pivot is fixed by their sum. This holds for any fixed f and backend record, with fresh residual masks across labels.

Release and outputs. Any proper subset of the honest coordinates of this fixed-sum vector is uniform. The simulator's first k unrounded values therefore implement exact deferred sampling, including for identities selected from the preceding transcript. At the last publication, (37) changes only that value, by $e_\tau = b_\tau - a_\tau s - \text{EncTarget}_\tau(v_\tau)$, with $\|e_\tau\|_\infty \leq B_e$. The crossing coefficient is 1. The core first-query boundary argument now costs $q_H N n \beta_L(B_e) + \mathbf{1}_{k=0} \varepsilon_{\text{prim}}$, covering every possible last sender. For $k = 0$ there are no noise or RLWE

replacements; fresh h_τ and the primitive-mask lemma supply the bound. The primitive-mask exception is thus charged exactly once.

Both replacements preserve (39). For arbitrary rounding residues $\|\rho_i\|_\infty \leq (L-1)/2$,

$$z = \left[b - L \sum_i p_i \right]_q \equiv b - as - \sum_i \rho_i \pmod{q}.$$

Consequently Theorem C.1 applies pointwise, with $D = 1$: BFV uses $B_e + N(L-1)/2 < q/(2p)$; CKKS uses the no-wrap and precision bounds (31)–(32). No independence of rounding residues is needed. Outside the marked events, identical packets give identical outputs satisfying the ideal relation.

Lemma 8 applies because publication and delivery commands are unchanged. Readiness now requires all N responses; corrupted indices count toward release even before their packets arrive. The simulator preserves retries, cancellation, self-receipts and incomplete executions, using the declared setup view without honest-share queries or completion conditioning. Adding the three transition bounds proves (35); setup realization losses are charged separately. $\square$

The remaining subsections establish the static-malicious extension: the numerical DKG theorem preserves the full sampled application record, and the combined theorem adds commitment and online verification losses.

D.3 Temporary-HE assumptions and malicious DKG proof

We prove Section C.5.3 in the local-resource hybrid of Section 6.3; Theorem 7.3 separately charges R6 commitment losses.

D.3.1 Output distribution and assumptions Before completion decisions, inversion of the evaluation map, whose determinant is a unit by Lemma 3, for the $t-1$ ideal seed pairs (including corrupted indices) gives

$$F(Y) = s + \sum_{\ell=1}^{t-1} a_\ell Y^\ell, \quad G(Y) = \sum_{\ell=1}^{t-1} b_\ell Y^\ell, \quad (40)$$

$$a_\ell, b_\ell \xleftarrow{\$} \mathcal{R}_q \text{ independently.}$$

Both coefficient families are independent of the base-key record.

Temporary-HE correctness. For any admissible PPT adversary, HE-Corr bounds by ε_{HE} the probability that a valid R3 prefix violates

$$c_{0,o} + c_{1,o}s^{\text{tmp}} = \text{Emb}_o(M_o^{\text{base}}) + e_o, \quad \|e_o\|_\infty \leq B_{\text{ct},o}, \quad (41)$$

or flooded decoding correctness, before later abort decisions. The correctness bound must cover the complete fixed circuit/actual parameters: enlarged $B_{1,j}$, hint/input floods, level projections/embeddings, output flood sums and decoding

margins. The security error bounds assume this correctness bound; it supplies no concrete feasible circuit-correctness parameter set.

Temporary FHE. Use the direct-RLWE TempHE profile of [Section C.5.3](#), with actual same-key public/EVK entries at modulus M :

$$\begin{aligned} b_x &= -a_{\text{pk}}x + e_x^{\text{pk}}, \\ K_j^* &= \gamma_j x^2 - A_j^* x + e_j^*. \end{aligned} \tag{42}$$

Use $x \leftarrow \chi_{\text{tmp}}$, independent uniform a_{pk}, A_j^* , independent errors from the original key distribution, and exact level projections/encryption distributions. Let Adv_{tmp} bound multi-message IND-CPA with complete public EVK and one shared challenge bit, without secret/decryption/setup oracles. The direct key format is standard [\[20, App. E.1\]](#); any circular-security premise is part of this public-EVK assumption [\[20, Sect. 2.1\]](#).

For J columns, Adv_{aux} bounds distinguishing $(a_j, -a_j u + e_{0,j})_j$ from independent uniform second components, with shared $u \leftarrow \chi_u$. Use independent honest R1 errors $e_{1,i,j} \xleftarrow{\$} [-B_{1,j}, B_{1,j}]^{n_{\text{tmp},j}}$ as in [Section C.5.1](#). Comparisons between the original distributions exclude implementation losses: (50) charges them to ε_{red} over all reduction occurrences, with multiplicity.

IND-CPA with public auxiliary samples. We make explicit the auxiliary experiment used below; it grants no secret-state or setup oracle. In experiment Aug_β , for $\beta \in \{0, 1\}$, the challenger samples the original temporary key $(x, b_x, (A_j^*, K_j^*)_j)$ of (42) and gives the distinguisher the complete public key and evaluation key. Independently for every column, it samples

$$a_j \xleftarrow{\$} \mathcal{R}_M, \quad e_{1,j} \xleftarrow{\$} [-B_{1,j}, B_{1,j}]^{n_{\text{tmp},j}}, \quad z_j = a_j x + e_{1,j} \pmod{M},$$

and gives $(a_j, z_j)_j$ to the distinguisher. On each adaptive query (m_0, m_1) with equal public encoding length, it returns an ordinary TempHE encryption of m_β with independent encryption coins, at the declared level. The same bit is used for all queries. The distinguisher may keep the queried messages and its own randomness, but receives neither x nor the auxiliary errors or encryption coins. Define

$$\text{Adv}_{\text{augmented}}(\mathcal{D}) := |\Pr[\mathcal{D}^{\text{Aug}_0} = 1] - \Pr[\mathcal{D}^{\text{Aug}_1} = 1]|.$$

This is an intermediate experiment, not an additional hardness assumption. In particular, it does not allow arbitrary auxiliary leakage correlated with x ; the following reduction applies to exactly the samples above.

Lemma 13 (Simulating auxiliary samples from the public key). *From the actual public key alone, independently sample $r_j \leftarrow \chi_u$, $\nu_j \leftarrow \chi_{0,j}$, $\zeta_j \xleftarrow{\$} [-B_{1,j}, B_{1,j}]^{n_{\text{tmp},j}}$ and compute*

$$\begin{aligned} a_j &= a_{\text{pk}} r_j + \nu_j, & z_j &= -b_x r_j + \zeta_j = a_j x + \zeta_j + \Delta_j, \\ \Delta_j &= -e_x^{\text{pk}} r_j - \nu_j x, & D_j &:= h_u B_{\text{pk}} + B_{\text{tmp}} B_{\nu,j}. \end{aligned} \tag{L1}$$

With auxiliary noise $\chi_{0,j}$ and $B_{\nu,j} = B_{0,j}$, the norm bounds give $\|\Delta_j\|_\infty \leq D_j$. Jointly with the full public key/EVK and adaptive encryption answers, the distinguishing gap between these simulated samples and the samples of Aug_β , for either fixed β , is at most $\text{Adv}_{\text{lift}} + \varepsilon_{\text{lift}} + \theta_{\text{lift}}$, where

$$\text{Adv}_{\text{lift}} := \sum_{j=1}^J \text{Adv}_{\text{RLWE},j}^{(1)} \leq J \text{Adv}_{\text{aux}}, \quad \varepsilon_{\text{lift}} \leq \min \left\{ 1, \sum_j \frac{n_{\text{tmp},j} D_j}{2B_{1,j} + 1} \right\}.$$

Consequently, for every PPT $\mathcal{D}$,

$$\text{Adv}_{\text{augmented}}(\mathcal{D}) \leq \text{Adv}_{\text{tmp}} + 2\text{Adv}_{\text{lift}} + 2(\varepsilon_{\text{lift}} + \theta_{\text{lift}}). \quad (43)$$

Proof. First remove Δ_j from the simulated sample using the independent uniform noise ζ_j and (48), with the stated cutoff charge. The resulting sample is $(a_j, a_j x + \zeta_j)$ with $a_j = a_{\text{pk}} r_j + \nu_j$. Replace each a_j by uniform using a single-sample auxiliary-RLWE challenge. The reduction samples x and errors from the original key distribution, sets a_{pk} to minus the challenge coefficient, and takes the challenge's second component as a_j . It can therefore compute $a_j x + \zeta_j$, the full public key/EVK, the other columns and, for the fixed bit β , all adaptive encryption answers. It does not need the challenged r_j or ν_j . The common public coefficient is allowed in each single-column reduction; summing the column costs gives Adv_{lift} .

For the middle comparison, an ordinary public-EVK IND-CPA reduction receives the original public key/EVK, generates all simulated samples by (L1) using its own randomness, and forwards every message-pair query to its challenge oracle. This requires neither x nor e_x^{pk} . Apply the preceding replacement once for each challenge bit and use the triangle inequality to obtain (43). These comparisons preserve the distinguisher's local state in the auxiliary experiment, including its queried messages, public answers and its own randomness. Internal coins used to generate the simulated samples are excluded from this state. No assertion is made for additional secret-dependent leakage.

Error bounds. Registered states satisfy $\|s_i^{\text{tmp}}\|_1 \leq B_{\text{tmp}}$, $\|u_i\|_1 \leq h_u$, $\|e_i^{\text{pk}}\|_\infty \leq B_{\text{pk}}$, and $\|e_{b,i,j}\|_\infty \leq B_{b,j}$ for $b \in \{0,1\}$. Set

$$B_{\delta,j} = N(B_{\text{tmp}}B_{0,j} + h_u B_{1,j}).$$

Let $B_{\text{evk},j}$ and $B_{\text{in},b}$ bound the original evaluation-key error e_j^* and tape block b 's aggregate encryption shift in its output module. At an unscaled negacyclic level, suppose $\|v_{\text{enc}}\|_1 \leq h_{\text{enc},b}$, $\|e_1\|_\infty \leq B_{\text{enc1},b}$, and each projected public-key error is bounded by $B_{\text{pk},b}$. Then the ciphertext transformation below admits

$$B_{\text{in},b} = (N-1)(B_{\text{pk},b}h_{\text{enc},b} + B_{\text{tmp}}B_{\text{enc1},b}).$$

Any subsequent coefficient map additionally requires a proved operator-norm bound; this formula does not identify distinct levels or modules. Independent

coefficientwise uniform floods with bounds $B_{\eta,j}, B_{\xi,b}, B_{\rho,o}$ give

$$\begin{aligned}
\varepsilon_\delta &\leq \min \left\{ 1, \sum_j \frac{n_{\text{tmp},j} B_{\delta,j}}{2B_{\eta,j} + 1} \right\}, \\
\varepsilon_{\text{evk}} &\leq \min \left\{ 1, \sum_j \frac{n_{\text{tmp},j} B_{\text{evk},j}}{2B_{\eta,j} + 1} \right\}, \\
\varepsilon_{\text{in}} &\leq \min \left\{ 1, \sum_b \frac{n_b B_{\text{in},b}}{2B_{\xi,b} + 1} \right\}, \\
\varepsilon_{\text{out}} &\leq \min \left\{ 1, \sum_o \frac{n_o B_{\text{ct},o}}{2B_{\rho,o} + 1} \right\}.
\end{aligned} \tag{44}$$

Use widened $B_{1,j}$ in $B_{\delta,j}$ and sum over all columns, honest blocks and outputs per key-generation instance, with actual dimensions $n_{\text{tmp},j}, n_b, n_o$ (not necessarily n). Each honest flooding variable is independent of its shift before any value depending on that variable is revealed; successive hybrid costs add by triangle inequality, without independence of error events. Uniformly over unconditioned experiments, challenge bits, admissible adversaries and schedules, θ_{reg} bounds any honest sample outside its stated bounds, and θ_X , $X = \text{lift}, \delta, \text{evk}, \text{in}$, bounds any respective infinity-norm cutoff failure

$$(\Delta_j, \delta_{h,j}, e_j^*, \Delta_{\text{enc},b}) \quad \text{at} \quad (D_j, B_{\delta,j}, B_{\text{evk},j}, B_{\text{in},b}).$$

Put

$$\varepsilon_{\text{tail}} = 2(\theta_{\text{reg}} + \theta_{\text{lift}} + \theta_\delta + \theta_{\text{evk}} + \theta_{\text{in}}). \tag{45}$$

Output/decoding failures are in ε_{HE} . Sampler coupling costs are $\varepsilon_{\text{samp}} = \varepsilon_{\text{pre}} + \varepsilon_{\text{KG}} + \varepsilon_{\text{post}} + \varepsilon_{\text{red}}$ for distributions sampled by real protocol roles, central-key tapes, implemented simulator and all reductions, respectively. Count each key/zero-coefficient or temporary-role occurrence once; exact resource draws cost zero. J_{samp} rejection trials, each succeeding with probability at least $1/2$, cost at most $2^{-J_{\text{samp}}}$ per occurrence. Native hardness challengers are never truncated or conditioned; non-dyadic distributions need not have exact finite tapes.

The main-body theorem is [Theorem 7.2](#); its complete H0–H6 chain uses the lemmas below.

D.3.2 Distributional steps and proof All comparisons retain the public/corrupted transcript, delivery history and full sampled application record. They do not retain or expose the honest temporary secret x or unexposed honest resource witnesses. **Fixed state and relinearization-key replacement.** Every accepted message from a corrupted party must satisfy the prescribed relation on its registered values. In particular, each R4 partial is fixed before the party decides whether to send it. Write $s^{\text{tmp}} = \sum_i s_i^{\text{tmp}}$, $U = \sum_i u_i$, and

$$E_b = \sum_i e_{b,i}.$$

$$K + As^{\text{tmp}} = \gamma(s^{\text{tmp}})^2 + s^{\text{tmp}}E_0 + UE_1 + \sum_i \eta_i. \quad (46)$$

Fix an honest party h and put $x = s_h^{\text{tmp}}$, $T = \sum_{i \neq h} s_i^{\text{tmp}}$, and $\bar{h}_{0,-h} = \sum_{i \neq h} h_{0,i}$.

$$\begin{aligned} k_h &= \gamma x^2 + (\gamma T + \bar{h}_{0,-h} - A)x - Th_{0,h} + \eta_h + \delta_h, \\ \delta_h &= s^{\text{tmp}}e_{0,h} + E_1u_h. \end{aligned} \quad (47)$$

Within the registered norm bounds, $\|\delta_h\|_\infty \leq B_\delta$. For η uniform on $[-B, B]^{n_{\text{tmp}}}$ and independent of shift data V , overlap counting gives the unconditional bound

$$\text{SD}((V, \eta + \delta(V)), (V, \eta)) \leq \mathbb{E}[\min\{1, n_{\text{tmp}}\|\delta(V)\|_\infty/(2B+1)\}]. \quad (48)$$

Separating the cutoff-failure event gives a bound of $\varepsilon_X + \theta_X$; modular reduction and causal continuation preserve the bound. For relinearization-key contributions, corrupted states are fixed before the η_h -dependent R3 exposure, and the shift excludes η_h . Auxiliary-sample, evaluation-key and input-ciphertext replacements likewise use independent uniform floods. Retain adaptive sending, abort and pending decisions as transcript data under the same continuation, never as acceptance/completion conditions. Remove δ_h ($\varepsilon_\delta + \theta_\delta$), then replace $h_{0,h}$ by uniform using an auxiliary reduction knowing x (Adv_{aux}). The TempHE reduction uses (L1) without x, e^* :

$$A = A^* + \gamma T + \bar{h}_{0,-h}, \quad v_h = A - \sum_{i \neq h} v_i, \quad \tilde{k}_h = K^* - Th_{0,h} + \eta_h. \quad (49)$$

A^* is independent of visible R1 and the other fixed coins and states, so v_h is uniform and the reduction can defer generating its unopened record. K^* is first used in R3; smudge e^* at cost $\varepsilon_{\text{evk}} + \theta_{\text{evk}}$.

For $E = \sum_{i \neq h} e_i^{\text{pk}}$, the ciphertext transformation is

$$\begin{aligned} b &= b_x - a_{\text{pk}}T + E, & (c_0, c_1) &\mapsto (c_0 - Tc_1 + \xi, c_1), \\ \Delta_{\text{enc}} &= -Ev_{\text{enc}} - Te_1. \end{aligned}$$

Independent ξ absorbs Δ_{enc} without computing it ($\varepsilon_{\text{in}} + \theta_{\text{in}}$). Submit $(\omega, 0)$ for each honest tape block. The reduction itself samples every honest application tape ω_i and records the corrupted tapes submitted to the emulated commitment-and-proof resource. R1 fixes all these tapes before any R2 challenge ciphertext is exposed. Their XOR therefore specifies the same application key in both branches without knowledge of the challenge bit. Later sharing messages are generated by the conditional sampler below. The retained application secret is distinct from the unknown temporary secret x . The two endpoint bridges cost

$$\varepsilon_{\text{back}} \leq \varepsilon_{\text{crypt}} + \varepsilon_{\text{tail}} + \varepsilon_{\text{red}}. \quad (50)$$

Apply this ciphertext-replacement reduction only after the output-opening replacement below. The simulator’s own temporary secrets differ from the reduction’s unknown secret. After replacement, suppress honest domain failures and restore $\mathcal{S}_6$ ’s pending rule for its own sample outside the stated bounds (θ_{reg} per endpoint), including pre-R3 failures for $C = \emptyset$. Use unconditioned draws from the original distributions: unknown challenge tails are analysis events charged once per relevant endpoint, with no extra output-tail charge.

State available to the ciphertext reduction. The preceding reductions operate on the H3 execution, after the output opening has been replaced by **SimOpen**. For clarity, their continuation uses exactly the following data.

1. Public key/EVK, the samples of [Section D.3.1](#), and challenge ciphertexts. These determine all public evaluations. For party h ’s relinearization-key contribution, use (49); the other honest temporary secrets are sampled by the reduction.
2. The chosen honest application tapes, source masks and sharing coefficients, corrupted submissions to the emulated resource, and honest noise-row coordinates already delivered to corrupted parties. Thus T , the corrupted consistency checks and the full application record (pub, s, F, G) are computable without x . Unexposed honest resource records may remain deferred, as allowed by the interface; they are not recovered from the simulated openings.
3. Output partials generated by (51), and sharing messages generated from the retained application record by (52). All corrupted-recipient transfers are therefore computable; honest-to-honest transfers expose only metadata.

In these proof hybrids the reduction emulates the local resource; it does not submit the unknown x as a numerical witness. Honest receipts use the exact public statement and fixed metadata. Corrupted submissions are checked against their own registered values and incoming references, including submissions chosen adaptively from earlier messages. The restricted resource offers no operation that exports an honest temporary secret, tape, or unused witness. After output replacement, no later honest numerical message requires x . Consequently, a distinguisher of H3 and H4, including any causal use of the retained application record, gives a distinguisher in the specified auxiliary experiment. This argument would not apply to a resource that exposed additional honest temporary state.

Simulated opening. For fixed ρ and any assignment to the $c < N$ corrupted coordinates,

$$\begin{aligned} \#\{w \in \mathcal{A}_o^N : \sum_i w_i = \rho\} &= |\mathcal{A}_o|^{N-1}, \\ \#\{w : \sum_i w_i = \rho, w_{\text{corr}} = \bar{w}_{\text{corr}}\} &= |\mathcal{A}_o|^{N-c-1}. \end{aligned}$$

Observed honest-row coordinates are uniform independently of their sums: R2 exposure and subsequent choices leave an honest row’s uniformly sampled sum independent of the error shift. For $H = [N] \setminus \text{corr}$, conditional on these coordinates, corrupted rows and all sums, honest R4 partials are uniform with sum

$$\text{Emb}_o(M_o) + e_o + \rho_{H,o} + \rho_{\text{corr},o} - c_{0,o} - \sum_{i \in \text{corr}} \hat{p}_{i,o}.$$

Here $M_s = m$, public M_o is prescribed, and recorded states determine $\hat{p}_{i,o}$. $\text{SimOpen}(o, M_o)$ samples $\tilde{\rho}_{H,o}$ from the distribution of the sum of honest floods and sets

$$V_o = \text{Emb}_o(M_o) + \tilde{\rho}_{H,o} + \rho_{\text{corr},o} - c_{0,o} - \sum_{i \in \text{corr}} \hat{p}_{i,o}. \quad (51)$$

Sample all but one honest partial opening uniformly; set the remaining one so that their sum is V_o . One unused independent honest row's uniform flood hides e_o by (48), costing $\varepsilon_{\text{out}} + \varepsilon_{\text{HE}}$ jointly with the full application record. Data processing covers the same adaptive later sharing/abort/pending continuation, without conditioning on its outcomes.

Lemma 14 (Conditional simulation of key and zero sharings). *Condition on full F, G , source masks, m , and corrupted R_4 polynomials. The table of honest key and zero sharing messages to corrupted parties has an exact sampler depending on F, G only through $f_{\text{corr}}, g_{\text{corr}}$: sample all but one honest row uniformly and determine the remaining row by (52).*

Proof. For $c > 0$, the first c columns of $V_{\text{corr}} = (\alpha_i^\ell)_{i \in \text{corr}, 1 \leq \ell < t}$ have determinant

$$\left(\prod_{i \in \text{corr}} \alpha_i \right) \prod_{i < k, i, k \in \text{corr}} (\alpha_k - \alpha_i),$$

a unit; each evaluation fiber has size $|\mathcal{R}_q|^{t-1-c}$ ($c = 0$ is vacuous). Fix honest h^* , sample the other honest rows on corr independently uniformly, and set

$$\begin{aligned} x_{h^*, \text{corr}} &= f_{\text{corr}} - m \mathbf{1}_{\text{corr}} - \sum_{j \in \text{corr}} x_{j, \text{corr}} - \sum_{h \in H \setminus \{h^*\}} x_{h, \text{corr}}, \\ y_{h^*, \text{corr}} &= g_{\text{corr}} - \sum_{j \in \text{corr}} y_{j, \text{corr}} - \sum_{h \in H \setminus \{h^*\}} y_{h, \text{corr}}. \end{aligned} \quad (52)$$

For either family, the conditional probability of any first $|H| - 1$ rows is

$$\frac{|\mathcal{R}_q|^{(t-1-c)(|H|-1)}}{|\mathcal{R}_q|^{(t-1)(|H|-1)}} = |\mathcal{R}_q|^{-c(|H|-1)}.$$

The aggregate fixes the last coefficient vector; constants translate fibers and independent families multiply probabilities. Sample once after corrupted rows are fixed, then release in scheduler order, preserving complaints/incomplete prefixes. After removal of secret-dependent evidence, dependence on G only through g_{corr} preserves its uniform residual kernel given the pre-online algebraic view. This is not an information-theoretic uniformity claim conditioned on the real commitments and transcript. $\square$

The complete H0–H6 chain appears in the main-body proof of [Theorem 7.2](#).

D.4 Proof of malicious threshold decryption

We prove [Theorem 7.3](#) by checking the obligations added to the core proof by input validation, DKG and local proofs. The session-local resources follow the commit-and-prove pattern of [8, Sect. 7.1]; they are not global proofs or transferable NIZKs.

Proof. Fix one simulator per adversary, independently of $\mathcal{Z}$, and retain failed setup and incomplete executions.

0. Validated ciphertexts. Induction on local validation events gives $\text{Adm}_j(\tau) = 1 \Rightarrow \text{ct}_\tau \in \text{EvalClosure}(\text{ValidInput}, \text{Policy})$: roots require a matching (V0) acceptance or honest self-acceptance; derived nodes require valid ancestors and exact public recomputation. The pointwise conditions of [Section 7.3](#) therefore assign one output $\text{Dec}_\tau^*(s, \text{ct}_\tau)$ to every accepted proof for τ , even with maliciously chosen encryption randomness. Couple honest input tapes and failure branches; the ideal service produces the same ciphertext, and the simulator emulates honest notices without witnesses. Forward only actual corrupted witness submissions privately for the same (V0) check. Induction under the common key couples records, notices and local validation decisions, including invalid inputs, interleaving and proof sessions whose statements fail public checks. Starting a proof or extracting its submitted witness does not publish a pending root. All proof paths for the ciphertext share its canonical label and Seen_τ , without recipient synchronization.

1. Binding responses to DKG shares. A corrupted party's (D1) witness fixes its shares from actual incoming references, including its own contribution. An accepted (V1) witness for a different pair yields distinct openings of its registered commitment, at cost $\varepsilon_{\text{bind}}$. For two accepted responses using the same pair and label,

$$\begin{aligned} L(p - p') &\equiv \rho - \rho' \pmod{q}, \quad q = Lr, \quad \|\rho - \rho'\|_\infty < L \\ \implies \rho &= \rho', \quad p = p' \text{ in } \mathcal{R}_\tau. \end{aligned}$$

Reduce coefficientwise modulo L ; odd- L rounding supplies existence. Thus every accepted response is the prescribed rounding, without a commitment-knowledge assumption. Emulate notices exactly while honest statements are true and retain that emulation after replacing honest commitments. The resource rules of [Section 5.5](#) fix each session's statement, parties and first proof input, even before public registration checks pass; such sessions fill no response entry and are never reset by admission. After the first R6 completion notice, new source proofs, updates and transfers are rejected, while historical receipts remain verifiable. Static corruption requires no later honest-state opening.

2. Hiding before other replacements. Replace each honest C_i by $\text{Com}(\langle 0, 0 \rangle; \hat{\vartheta}_i)$. The reduction generates genuine setup, knows the two challenge messages and unchallenged coins, and retains numerical responses and simulated notices without the challenge opening. Its continuation includes malicious inputs, incomplete R6, aborts and interleaving. These at most $N - c$ replacements

cost $\varepsilon_{\text{hide}}$ and precede DKG and RLWE replacement, so commitments disclose no challenged residual coefficient.

3. DKG and causal setup views. [Lemma 14](#) samples honest dealings from $(m, f_C, g_C, \text{rows}_C)$ while preserving full F, G ; the zero-dealing view depends on G only through g_C . Corrupted R6 checks use actual tables; honest proofs use simulated notices and commitments to zero. The joint DKG theorem preserves (pub, s, F, G) and all release times at cost ε_{DKG} , allowing the same decryption continuation without giving its simulator s or honest openings.

Presample the ideal one-shot key and independent sharing coins independently of the mask-oracle tape, without changing their release times. Causal dealings and corrupted rows may depend on earlier oracle answers but disclose no additional residual zero-sharing randomness. At each candidate's first hash query, even before initialization, setup or admission, condition only on this independent algebraic record and preceding history. The label fixes ciphertext bytes and its oracle answer is fresh uniform. Mark boundary events for every candidate and party at that time; never condition on future rows, delivery, admission or successful completion.

4. Applying the core hybrids. The preceding replacements establish the hypotheses of [Lemma 9](#): the coefficients of U in $G = G_C^0 + W_C U$ are uniform given the permitted algebraic view. The RLWE reduction generates base/evaluation keys, F, g_C and the causal dealing view without the challenged coefficient. It knows s for internal decryption, whereas the final simulator receives neither s nor v_τ before release. Consequently the proof-only-noise and shared-first-component column replacements in [Section D.1](#) apply, with

$$\begin{aligned}\varepsilon_{\text{noise}} &\leq \mathbf{1}_{k>0} \varepsilon_{\text{prim}} + \varepsilon_\chi + q_H N n \beta_L(B_{\text{mask}}), \\ \varepsilon_{\text{RLWE}} &\leq k \text{Adv}_{\text{RLWE}}(T_{\text{red}}, q_H).\end{aligned}$$

Both steps are empty for $k = 0$. First-query answers are stored and reused, and commitments to zero require no additional secret.

5. Exposure and release. Let E_τ be the set of honest identities with a public response, including responses with unfinished proofs. Then $\text{Seen}_\tau = C \cup E_\tau$ and $D_{\tau,j}^{\text{ver}} \subseteq \text{Seen}_\tau$. The core deferred-sampling argument gives the first k unrounded values, including for identities chosen from the transcript. Immediately before the first threshold-reaching publication, set $T = \text{Seen}_\tau$, so $|T| = t - 1$. The simulator submits [\(S1\)](#) for this set before learning v_τ ; evaluation, publication and release are atomic. Use $\text{EncTarget}_\tau(v) = (q/p)v$ for BFV and $\text{Lift}_\tau(v)$ for CKKS. Thus

$$P'_\tau - P_\tau = e_\tau \prod_{j \in T} (1 - Y \alpha_j^{-1}), \quad \|e_\tau\|_\infty \leq B^*.$$

The core universal boundary event costs $\varepsilon_{\text{cross}} \leq q_H N n \beta_L(\Gamma_{t-1} B^*) + \mathbf{1}_{k=0} \varepsilon_{\text{prim}}$, covering every later crossing set without a union over sets. For $k = 0$, fresh hashes and primitive masks supply this bound; otherwise the primitive-mask exception was charged at noise addition. Together the three costs are at most $\varepsilon_{\text{online}}$.

6. Outputs and delivery. Couple initialization, key publication, local validation, response publication, proof notices, cancellation and delivery. Accepted corrupted responses are fixed by Step 1; honest packets agree outside the marked events. Matching acceptances or honest self-receipts give identical verified sets, so t verified identities imply prior release to the simulator. [Lemma 1](#) gives $z_S = [Du_\tau - R_S]_q$ with $\|R_S\|_\infty \leq \mathcal{B}(S)$. BFV correctness or the exact-output CKKS condition ([Z](#)) forces output v_τ ; packet coupling also preserves the public aggregate before quantization. The unchanged local execution rules retain buffers, published responses and proof sessions after cancellation; the simulator schedules using notified tickets and accepted publications, without private contexts. Abort preserves public and validation records, while rejected proposals alter no state and access no decryption value. Thus the [specified simulator](#) matches every recipient, retry and incomplete execution, with an unprogrammed oracle, no fairness or completion conditioning, and no honest-share queries. Summing the costs proves ([20](#)). $\square$

E Evaluation bounds and additional profiles

This appendix proves the bounds and checks supporting [Section 8](#).

E.1 Certified modulus budgets

We bound the reconstruction and simulation factors, then choose $L = q/r$ before r to avoid an implicit inequality in q .

Lemma 15 (Uniform integral coefficient bounds). *Use the points of ([1](#)), let $N' = 2^{\lceil \log_2 N \rceil} \leq 2n$, and put $D = 2^{\lceil \log_2 t \rceil}$. For $2 \leq t \leq N$, define*

$$U_2(t, N') = 2^{-t + \lceil \log_2 t \rceil + 1} \sqrt{N'/2} \prod_{a=1}^{\lfloor t/2 \rfloor} \sin^{-2}(\pi a/N'), \quad (53)$$

$$M_{\text{cert}}(t, N') = tU_2(t, N').$$

Then the fixed integral cleared coefficients satisfy $\|\lambda_i^S\|_1 \leq U_2(t, N')$ for every $|S| = t$ and $i \in S$, and hence $M_1^{\max} \leq M_{\text{cert}}$. For the simulation factors Γ_j of [Section 7.1](#), set $\hat{\Gamma}_0 = 1$ and, for $1 \leq j < t$, set

$$\hat{\Gamma}_j = \min \left\{ 2^j, 2^j \sqrt{N'/2} \prod_{a=1}^{\lfloor (j+1)/2 \rfloor - 1} \cos^2(\pi a/N') \right\}. \quad (54)$$

Empty products equal one, and $\Gamma_j \leq \hat{\Gamma}_j$. Both bounds depend on (t, N') , independently of n .

Proof. Apply the third and fourth bounds of KLL⁺26 [[17](#), Cor. 5.1], translating its ring degree and party count to n, N . Its Theorem 3.8 clears denominators

before applying the embedding-norm bound; hence the first bound holds for integral coefficients before reduction modulo q , without assuming they are centered representatives. Its Corollary 5.1 embeds the degree- $N'/2$ subring generated by $y = x^{2n/N'}$ into $\mathcal{R}$, preserving coefficient ℓ_1 norms. Apply the fourth bound at threshold $j+1$ to $\prod_{u \in T} (1 - Y\alpha_u^{-1})$ evaluated at α_i , together with the elementary bound 2^j .

This upper bound need not equal $M_1^{\max}$. Over $2 \leq t \leq N$, its unique maximum occurs at $t = 86, 170, 342, 682$ for $N = 256, 512, 1024, 2048$, respectively. Each one-third row is therefore within one bit of the largest reconstruction bound, but need not minimize the saved fraction, whose denominator also depends on $\hat{\Gamma}_{t-1}$.

Jambon budget. By (18), (23) allocates $2^{-\kappa}$ to rounding boundaries; RLWE, error tails, exceptional masks, and setup failures remain separate. Taking only $c = t - 1$ would omit the mask term for smaller corruption sets. Substituting $q = rL$ into (8), with the exact odd- L residue bound $(L - 1)/2$, gives (24). In the threshold-dominated range, the leading budgets are

$$\log_2 r \simeq \log_2 p + \log_2 M_{\text{cert}}, \quad \log_2 q = \log_2 r + \log_2 L. \quad (55)$$

Bit widths use the exact inequalities. Increasing n preserves the coefficient bound but increases the coefficient count and required gap.

Bounds for the strong native baseline. Put $N'_K = 2^{\lceil \log_2 \max\{N, 2t\} \rceil}$, $m = N'_K/2$, and $D = 2^{\lceil \log_2 t \rceil}$, and assume $N'_K \leq 2n$. The interpolation points are $\alpha_i = y^{i-1}$, where $y = x^{2n/N'_K}$ generates the subring $\mathbb{Z}[y]/(y^m + 1)$ of $\mathbb{Z}[x]/(x^n + 1)$. Write $\ell_i^{S \rightarrow 0}$ and $\ell_j^{T \rightarrow \alpha_i}$ for ordinary reconstruction and interpolation coefficients, before clearing denominators. For $t \geq 3$, define

$$U_K = 2^{-t + \lceil \log_2 t \rceil + 1} \sqrt{m} \prod_{a=1}^{\lfloor t/2 \rfloor} \sin^{-2}(\pi a / N'_K),$$

$$G_K = D \sqrt{m} \prod_{a=1}^{\lfloor (t-1)/2 \rfloor} \sin^{-2}(\pi a / N'_K), \quad M_K = t U_K.$$

For $t = 2$, instead set $U_K = m$, $G_K = 4$, and $M_K = 2m$. In both cases, use

$$\hat{\Gamma}_{t-1}(N'_K) = \min \left\{ 2^{t-1}, 2^{t-1} \sqrt{m} \prod_{a=1}^{\lfloor t/2 \rfloor - 1} \cos^2(\pi a / N'_K) \right\}.$$

These bounds apply to integral lifts before modular reduction: for every qualified S of size t , $\|D\ell_i^{S \rightarrow 0}\|_1 \leq U_K$. For every set T of $t - 1$ party indices, $i \notin T$, and $T^* = \{0\} \cup T$, the polynomials $D\ell_j^{T^* \rightarrow \alpha_i}$ are integral and have coefficient ℓ_1 norm at most G_K , while $\ell_0^{T^* \rightarrow \alpha_i}$ itself is integral and has norm at most $\hat{\Gamma}_{t-1}(N'_K)$. For $t \geq 3$, apply the integral denominator-clearing and subring arguments of

KLL⁺26 [17, Thm. 4.4, Cor. 5.4]; centered reduction cannot increase coefficient norms. The separate cosine bound at 0 uses $\ell_0^{T^* \rightarrow \alpha_i} = \prod_{j \in T} (1 - \alpha_i \alpha_j^{-1})$; the elementary product bound gives the additional minimum with 2^{t-1} .

The case $t = 2$ lies outside Corollary 5.4. For distinct points, let k be their nonzero exponent difference modulo $2m$, set $g = \gcd(k, 2m)$ and $h = m/g$. Because m is a power of two and $k \not\equiv 0 \pmod{2m}$, g divides m and k/g is odd. Thus $(y^k)^h = -1$ and

$$\frac{2}{1 - y^k} = \sum_{a=0}^{h-1} y^{ka}.$$

The right side has coefficient ℓ_1 norm at most $h \leq m$ after reduction by $y^m + 1$, so each coefficient $D\ell_i^{S \rightarrow 0}$ has norm at most m when $D = 2$. For $T = \{j\}$, the interpolation coefficients at α_i are $\ell_0^{\{0,j\} \rightarrow \alpha_i} = 1 - y^{i-j}$ and $\ell_j^{\{0,j\} \rightarrow \alpha_i} = y^{i-j}$. Their norms are at most 2 and 1, respectively, and hence the common bound $G_K = 4$ for all interpolation coefficients is valid after multiplication by D . We retain this common bound, although the coefficient at j alone admits 2.

Lemma 16 (Strong native flooding with a finite query budget). *Consider the second KLL⁺26 construction, with its explicit deterministic $e_{\text{RLWE},i}$ term retained, and the statistical transition from Hybrid H_1 to Hybrid H_2 in the proof of its Theorem 4.2 and Lemma 4.3 [17]. Set $C_{\text{dec}} = D^2$ and $C_{\text{sim}} = D$ over an odd ciphertext modulus $q_K > 2D^2$. Assume at most q_H distinct ciphertext labels, at most one invocation per identity and ciphertext, a corruption set fixed before response queries with cardinality less than t , and the error bounds $\|e_{ct}\|_\infty \leq B_e$ and $\|e_{\text{RLWE},i}\|_\infty \leq B_\chi$ on these invocations. The corruption set may be chosen after the public key, as in the source experiment. Define*

$$\begin{aligned} H_t &= (N - t + 1)q_H, \\ E_K &= DB_e \widehat{\Gamma}_{t-1}(N'_K) + B_\chi[(t-1)G_K + D]. \end{aligned}$$

For fresh, independent uniform flooding on $[-B_{\text{sm}}, B_{\text{sm}}]^n$ at each invocation, this transition has statistical distance at most

$$\frac{H_t n E_K}{2B_{\text{sm}} + 1}. \quad (56)$$

Consequently,

$$B_{\text{sm}} = \max \left\{ 1, \left\lceil \frac{2^\kappa H_t n E_K - 1}{2} \right\rceil \right\}$$

makes this contribution at most $2^{-\kappa}$. This bounds only the stated statistical transition; it does not include the source's RLWE, random-oracle, base-FHE, or error-tail terms.

Proof. Initialize each ciphertext's exposed-identity set with the corrupted identities. The hybrids agree until this set reaches t . Freeze its first $t-1$ identities as T_{ct} : their deterministic components are stored for honest parties and computed

from shares for corrupted parties. Corrupted responses are unchanged; interpolation at an identity in T_{ct} reproduces its stored component. Thus only identities outside T_{ct} can change, at most $N - t + 1$ per ciphertext and H_t overall. This counts from plaintext release without padding the corruption set or revealing the plaintext early.

Fix the history and underlying state at a changed query. Write $T = T_{ct}$, $T^* = \{0\} \cup T$, and $\ell_j = \ell_j^{T^* \rightarrow \alpha_i}$. Let e_{ct} be the BFV error and $\eta_j = e_{\text{RLWE},j}$ the deterministically hashed errors, including η_i at the queried identity. The source's H_1, H_2 response centers differ, up to sign, by $D\delta$ modulo q_K , for the integral polynomial

$$\delta = \sum_{j \in T} (D\ell_j)\eta_j - D\eta_i - (D\ell_0)e_{ct}.$$

The integral-lift bounds and $\|uv\|_\infty \leq \|u\|_1 \|v\|_\infty$ give

$$\|\delta\|_\infty \leq (t-1)G_K B_\chi + DB_\chi + D\hat{\Gamma}_{t-1}(N'_K)B_e = E_K.$$

All $D\ell_j$ are formed in the integral subring before reduction; ℓ_0 is already integral. Neither their lifts nor δ need be centered modulo q_K .

For U_b uniform on $[-b, b]^n \cap \mathbb{Z}^n$ and any integral vector δ , exact counting gives

$$\Delta(U_b, U_b + \delta) = 1 - \prod_{a=1}^n \max \left\{ 0, 1 - \frac{|\delta_a|}{2b+1} \right\} \leq \frac{n\|\delta\|_\infty}{2b+1}.$$

Since D is invertible modulo odd q_K , remove the common center and factor D ; modular reduction cannot increase distance. Each changed query therefore contributes at most $nE_K/(2B_{\text{sm}} + 1)$. Its fresh flooding sample is independent of the history, query, stored components, and errors defining δ . The conditional bound thus holds for every adaptively chosen query and T_{ct} ; sequential coupling sums it over at most H_t invocations. Finally, $2B_{\text{sm}} + 1 \geq 2^\kappa H_t nE_K$.

The resulting sufficient correctness budget. For odd $p > 2$, D^2 is invertible modulo p . Let $F_K = (B_e + p/4)D^2$. The full correctness condition of the second construction (KLL⁺26, Theorem 4.1) follows from

$$q_K > 2p[F_K + M_K(DB_\chi + B_{\text{sm}})], \quad q_K > 2D^2.$$

Indeed $\|C_{\text{dec}} \bmod q_K\|_1 = D^2$, and summing the RLWE-error and flooding coefficient bounds DU_K and U_K over t contributors gives the result. We use Figure 4's explicit-error protocol, not Remark 1's noise-omission variant. The finite-query lemma replaces the source's coarser flooding condition; the smaller radius need not satisfy that condition literally. Additional fresh responses must be counted separately, whereas Jambon repeats its recorded response. For comparison, substituting $2^\kappa H_t$ for 2^λ in the source's unsharpened Equation (4) gives a larger sufficient modulus; neither choice is claimed optimal.

E.2 Profile details and statistical terms

Table 2 uses (22) and thresholds

$$t_{1/3} = \lfloor N/3 \rfloor, \quad t_{1/2} = N/2, \quad t_{2/3} = \lfloor 2N/3 \rfloor. \quad (57)$$

They are the largest integers satisfying, respectively, $t < (N+1)/3$, $t < (N+1)/2$, and $t < (2N+1)/3$ on this grid.

All use the same response arithmetic and privacy condition $c < t$. Output from honest parties alone requires fair activation and delivery and $N - c \geq t$. At $c = t - 1$, this is $N \geq 2t - 1$, which the two-thirds profile violates; threshold fractions alone do not establish availability.

Each coefficient of the proof-only χ is the difference of two independent sums of 64 uniform bits: support $[-64, 64]$ gives $\varepsilon_\chi = 0$ at $B_\chi = 64$. For odd q , its number $\omega(q)$ of distinct prime factors satisfies $3^{\omega(q)} \leq q$. Thus Lemma 10 gives $\varepsilon_{\text{prim}} \leq N\omega(q)3^{-n} \leq N \log_3(q)3^{-n} < 2^{-80}$ on the tabulated grid.

E.3 Sources of the response-size reduction

Write $M_J = M_{\text{cert}}(t, N)$, $T_J = 2^\kappa N q_H n$ and $T_K = 2^\kappa (N - t + 1) q_H n$. The flooding-dominated terms give $B_{\text{sm}} \simeq T_K E_K / 2$, $L \simeq 2T_J(A + B)$, and

$$\begin{aligned} r &\simeq pM_J, \\ q_J &\simeq 2pM_J T_J(A + B), \\ q_K &\simeq pM_K T_K E_K. \end{aligned} \quad (58)$$

Consequently,

$$\frac{q_J}{q_K} \simeq \frac{2N}{N - t + 1} \frac{M_J}{M_K} \frac{A + B}{E_K}. \quad (59)$$

These approximations describe sufficient choices, not necessary moduli; reported widths use exact bounds. Besides rounding from $q_J = rL$ to r , the simulation arguments differ: KLL⁺26 charges RLWE errors at previously fixed nonzero points and ciphertext error at zero. Jambon’s output shift B involves only the coefficient at zero; A separately charges proof-only mask errors (Theorem 7.1). The savings therefore combine rounding and simulation bounds, with different admissible geometries above $t = N/2$; the common-geometry check below isolates the latter effect.

Interpreting the plotted KLL⁺26 response sizes. The source’s Theorem 5.5 optimizes its own bound, not necessarily our finite-query refinement. For the plotted power-of-two party counts, $N'_K = N$ below or at half threshold and $2N$ above it. Both N'_K and $D = 2^{\lceil \log_2 t \rceil}$ double at $t = N/2 + 1$, producing Table 6’s jumps. Other power-of-two crossings change D ; serialization adds 1/128-MiB steps. The additional inverse-sine product G_K means the KLL⁺26 sizes need not peak near one-third threshold. Lines join discrete samples, without asserting continuity in t .

Table 6. Certified KLL⁺26 width jumps across half threshold. The spacing parameter changes from N to $2N$ and D from $N/2$ to N . Response sizes are MiB per party, rounded to two decimals.

N	$w(q_K)$		Response (MiB)	
	$t = N/2$	$t = N/2 + 1$	$t = N/2$	$t = N/2 + 1$
256	479	728	3.74	5.69
512	826	1322	6.45	10.33
1024	1517	2505	11.85	19.57

Table 7. Separate ciphertext and response widths at $N = 1024$, in bits per coefficient. KLL⁺26 is the strong Figure 4 profile with the finite-query refinement. These are sufficient analytical parameters, not matched concrete lattice-security estimates.

t	$w(q_J)$	$w(q_K)$	$w(r)$
341	957	1438	506
512	1024	1517	460
682	960	2736	349
1024	627	2884	31

E.4 Certified minima and sensitivity

The grid covers every integer $2 \leq t \leq N$, through $N = 32768$. The interval-verification procedure uses directed decimal rounding, Machin’s identity with alternating-series remainders for π , and Taylor enclosures for sine products. Widths are determined by comparison with exact powers of two, not floating-point logarithms.

Put $M = M_J$, $C = 2DB_e$, $F = (B_e + p/4)D^2$ and $\delta = (M + C)/T_J$. The selected odd moduli satisfy

$$\begin{aligned}
p(M - \delta) &< r \leq p(M + \delta + 2), \\
2p \left[F + M_K \left(DB_\chi + \frac{T_K E_K - 1}{2} \right) \right] &< q_K \\
&\leq 2p \left[F + M_K \left(DB_\chi + \frac{T_K E_K + 1}{2} \right) \right] + 2.
\end{aligned}$$

The grid satisfies $T_K E_K > 3$ and the correctness cutoff exceeds $2D^2$. An enclosure crossing a serialization boundary must be refined until it lies in $(2^{w-1}, 2^w]$; otherwise no width is certified. Exact rational comparisons then determine minima. Independent regression checks can use the original nine standard-baseline rows and all 21 entries of the updated KLL⁺26 Table 3 with their respective inputs.

Table 8. Statistical-budget sensitivity against strong KLL⁺26. Each pair is a minimizing threshold and its saving; all other inputs are fixed.

N	t^*	$\kappa = 40$		$\kappa = 64$		$\kappa = 80$	
		Saving (%)	t^*	Saving (%)	t^*	Saving (%)	
256	36	66.29	48	68.27	48	69.44	
512	52	62.97	78	64.49	78	65.37	
1024	88	60.37	102	61.47	108	62.13	
2048	106	58.39	140	59.19	152	59.66	

Remark 9 (Finite-grid minimum). Use (22) and $(N, t) = (1024, 88)$, so $N'_J = N'_K = 1024$ and $D = 128$. The exact bounds give

$$M = 88 \cdot 2^{-80} \sqrt{512} \prod_{a=1}^{44} \sin^{-2}(\pi a/1024), \quad G_K = 128 \sqrt{512} \prod_{a=1}^{43} \sin^{-2}(\pi a/1024).$$

Here $\hat{I}_{87} = 2^{87}$, $A = 2^{92}$, $B = 2^{117}$, $L = 2^{216} + 2^{191} + 2^{100} + 1$, $H_t = 937 \cdot 2^{32}$, and

$$E_K = 2^{124} + 64(87G_K + 128).$$

Interval evaluation gives

$$2^{320} < r < 2^{321}, \quad 2^{809} < q_K < 2^{810}, \quad 1 - \frac{321}{810} = \frac{163}{270} \approx 60.37\%.$$

The displayed logarithms are $\log_2 r \approx 320.171573$ and $\log_2 q_K \approx 809.411628$; $w(q_J) = 537$ is not the transmitted-response width. Minimality follows from the full $255 + 511 + 1023 = 1789$ -pair sweep, not from this single row.

At $N = 256, 512, 1024$, imposing the same geometry $N'_J = N'_K$ gives respective all-threshold minima of 64.52%, 62.81% and 60.37%. Thus the 60% floor on this finite grid does not depend on the geometry difference. Using the unsharpened source Equation (4) with the same finite-query allocation and Jambon’s native geometry instead gives 68.35%, 64.55% and 61.51%. The main tables use the smaller, refined KLL⁺26 modulus.

Statistical-budget sensitivity. For each tabulated profile, the $\kappa = 40$ enclosure for r also contains all choices at $\kappa \geq 40$ within the same bit width. The KLL⁺26 flooding radius and modulus are nondecreasing, so savings and their finite-grid minimum cannot decrease; the minimizing threshold can change. This describes the communication ratio, not a recommended security level.

These analytical extensions and the following one-third limit give no all-threshold guarantee for unbounded N .

Proposition E.2 (One-third profile against strong KLL⁺26). *Let N tend to infinity through powers of two, set $t = \lfloor N/3 \rfloor$, and keep $p, B_e, B_\chi, q_H, \kappa$ fixed, with $B_\chi > 0$. Choose admissible power-of-two ambient degrees $n \geq N/2$ satisfying $\log n = O(\log N)$. Use Jambon’s proved reconstruction and gap bounds to choose*

Table 9. Extended 2048-party profiles against strong KLL⁺26, with the inputs and reporting convention of Table 2.

N	t	Width (bits)		Response (MiB)		Saved (%)
		$w(r)$	$w(q_K)$	KLL ⁺ 26	Jambon	
2048	682	985	2738	21.39	7.70	64.0
	1024	892	2894	22.61	6.97	69.2
	1365	669	5332	41.66	5.23	87.5

its least admissible rounded-response modulus r , and use the finite-query strong parameterization above to choose the least admissible odd q_K . Set

$$I_s = \int_0^{1/6} \log_2 \sin(\pi x) dx, \quad a = -\frac{1}{3} - 2I_s, \quad g = -2I_s = a + \frac{1}{3}.$$

For $w(u) = \lceil \log_2 u \rceil$, the saved fraction satisfies

$$1 - \frac{w(r)}{w(q_K)} = \frac{g}{a+g} + O\left(\frac{\log N}{N}\right). \quad (60)$$

The limiting fraction is approximately 0.6317.

Proof. In this profile $N'_K = N$, so both constructions use the same reconstruction bound $M = tU_K$. The sine-product estimate gives $\log_2 M = aN + O(\log N)$ and $\log_2 G_K = gN + O(\log N)$. Separating the integrable endpoint singularity $\log x$ from $\log(\sin(\pi x)/x)$ yields an $O(\log N)$ error in the corresponding Riemann sums. The cosine bound has $\log_2 \hat{\Gamma}_{t-1} = bN + O(\log N)$, where $b = 1/3 + 2 \int_0^{1/6} \log_2 \cos(\pi x) dx < 1/3$. Since $\sin(\pi x) < 1/2$ for $0 < x < 1/6$, one has $g > 1/3 > b$. With fixed positive B_χ , it follows that

$$\log_2 E_K = gN + O(\log N), \quad \log_2 B_{\text{sm}} = gN + O(\log N).$$

The polynomial factors D, t, H_t, n and integer rounding affect only the $O(\log N)$ terms. Therefore the flooding term dominates the complete strong correctness budget and $w(q_K) = (a+g)N + O(\log N)$. Jambon's gap is exponential in this profile, and substitution in its exact BFV response condition yields $w(r) = aN + O(\log N)$ with $a > 0$. Dividing these expressions proves the stated result. This is the limit of the specified sufficient parameter choices, not a lower bound over all thresholds or a concrete-security statement.

E.5 Sensitivity to tighter reconstruction bounds

Hold the simulation bounds, L and B_{sm} fixed, and hypothetically improve both reconstruction bounds by d bits:

$$m_J = \max\{D, 2^{-d}M_J\}, \quad m_K = \max\{D, 2^{-d}M_K\}.$$

The floor is necessary by the cleared reconstruction identity; these improved bounds remain unproved. With $\text{odd}_>(x)$ the least odd integer strictly above x ,

$$\begin{aligned} r(m_J) &= p \text{odd}_>((1 - L^{-1})m_J + 2DB_e/L), \\ q_K(m_K) &= \text{odd}_>(\max\{2p[F + m_K(DB_\chi + B_{\text{sm}})], 2D^2\}), \end{aligned} \quad (61)$$

where $F = (B_e + p/4)D^2$. In the dominant-term range, a common d -bit tightening removes about d bits from each width, preserving the absolute gap and increasing the saved fraction. The exact rounded minima in Table 4 are recomputed, not inferred from that approximation. At $(N, t) = (1024, 512)$ the width pairs are $(460, 1517)$, $(452, 1509)$, $(444, 1501)$ and $(428, 1485)$; the 1057-bit gap is unchanged. Any improvement to the interpolation bounds used in simulation must also change E_K and the flooding budget before repeating the comparison.

E.6 Transport and setup accounting

For common ring degree n , let U, V count transmitted ciphertext and numerical-response copies. Their bit counts are

$$\begin{aligned} C_J &= n[2Uw(q_J) + Vw(r)], \\ C_K &= n[2Uw(q_K) + Vw(q_K)]. \end{aligned} \quad (62)$$

Hence $C_J < C_K$ precisely when

$$V(w(q_K) - w(r)) > 2U(w(q_J) - w(q_K)). \quad (63)$$

Counting one ciphertext object and t responses gives $U = 1, V = t$, without treating broadcast as unit-cost transport. At $N = 1024$ and $t = 341, 512, 682$, this count saves 64.63%, 69.53%, 87.18%, respectively. Unicasting the ciphertext to all parties uses $U = N, V = t$ and instead saves 37.93%, 39.93%, 70.49%. At $t = N = 1024$, these two counts save 98.88% and 85.15%. These counts exclude proofs, authentication, evaluation records, and setup; $U = 0$ recovers incremental response traffic when the ciphertext is already available. Each modulus requires a separate computational-security assessment; if approved ring degrees differ, the response ratio is

$$\frac{nw(r)}{n_K w(q_K)}.$$

Point-to-point delivery from t responders to every other party costs $t(N - 1)nw(r)$ bits; additional responders increase this cost. Semi-honest sharing of the secret and zero adds $2N(N - 1)nw(q)$ bits to backend traffic; sharing only zero adds $N(N - 1)nw(q)$ bits. Malicious setup also uses temporary HE, commitments, and proofs (Section 6). Its fixed commitment per identity is reused across ciphertexts. Sending each b_C -byte commitment to every other party gives $K_{\text{setup}} = N(N - 1)b_C + K_{\text{other}}$, including all other setup messages and proofs in K_{other} . The local-ZK functionality requires $s(N - 1)$ distinct recipient proof sessions, not one transferable proof. Any repeated response, commitment, or hash in

those sessions counts toward b_V ; b_{off} covers remaining offer/authentication traffic in both directions. Input proofs and evaluation records lie outside $E(b_r + b_V + b_{\text{off}})$. These counts cover prescribed invocations, not adversarial packet flooding.

For a concrete malicious comparator on the same transport, let its per-session verification and metadata cost be o_K and its amortized setup budget be K_K/J_{dec} . The communication difference is then

$$E \left[\frac{n(w(q_K) - w(r))}{8} + o_K - b_V - b_{\text{off}} \right] + \frac{K_K - K_{\text{setup}}}{J_{\text{dec}}}.$$

An empirical comparison requires compatible realizations with specified security parameters, serialization, and setup costs; no proof-size or runtime measurements are asserted here.

E.7 CKKS precision budget

For the encoded-target CKKS core, the BFV parameters must additionally satisfy [Corollary 5.1](#) at proved target bound B_m , scale Δ , numerical decoding error ν , and allowed error ε_{dec} :

$$\begin{aligned} q &> 2D(B_m + B_e) + (L - 1)M_{\text{cert}}, \\ \frac{n}{\Delta} \left(B_e + \frac{(L - 1)M_{\text{cert}}}{2D} \right) + \nu &\leq \varepsilon_{\text{dec}}. \end{aligned} \tag{64}$$

The second inequality bounds L above at fixed scale and precision; it must coexist with [\(23\)](#)'s lower bound. Increasing q alone does not establish feasibility.

Malicious CKKS has an exact finite ideal target and additionally requires the public lift and perturbation stability in [\(Z\)](#) ([Section 7.3](#)). Replace B_e by B_{lift} in the output-shift term B of [\(23\)](#). The inequalities above alone do not establish that profile; no concrete malicious CKKS parameters are reported.

F Generative AI Usage

This work used generative AI tools (Opus 5.1, Fable 5.1, and GPT-6) during manuscript preparation for:

- **Technical review assistance.** Checking proof sketches, theorem statements, parameter calculations, and the Jambon construction for potential issues. The authors independently checked flagged issues and applied corrections only after manual verification.
- **Presentation refinement.** Assisting with drafting and improving wording, notation, paragraph structure, and section transitions. The authors reviewed and revised AI-assisted text and retained control over the final manuscript.

The tools served as automated reviewers and writing assistants. The authors retain full responsibility for the accuracy, originality, and integrity of all content.