LatPRE: Compact Lattice-Based Proxy Re-Encryption with Security against Honest Re-Encryption Attacks

Hongxiao Wang[1,2], Mengting Yao[1,*], Ron Steinfeld[2],
Muhammed F. Esgin[2], Amin Sakzad[2], Jian Weng[1]
[1]*Guangzhou University* [2]*Monash University*
hongxiao.wang@monash.edu, mengting.ymt@gmail.com, cryptjweng@gmail.com
{*ron.steinfeld, muhammed.esgin, amin.sakzad*}*@monash.edu*

***Abstract*—Proxy re-encryption (PRE) allows a proxy to transform ciphertexts for other recipients without learning their plaintexts. It is a versatile cryptographic tool enabling a wide range of real-world applications, e.g., email forwarding, file sharing, and digital identity management. For these applications, security against honest re-encryption attacks (HRA) is essential: decrypting re-encryptions of some ciphertexts originally under a public key should not help break other ciphertexts under the same key. However, existing lattice-based approaches incur substantial ciphertext overhead or impose restrictions such as private proxy state or encryption requiring receiver's secret key.**

In this work, we present LatPRE, a compact single-hop HRA-secure PRE scheme based on the standard module learning with errors (MLWE) assumption. To this end, we make two technical contributions: (i) a novel generic framework for HRA-secure PRE based on a new hint-CPA security notion and hint-assisted re-encryption simulation; and (ii) tighter spectral bounds for honestly sampled leakage matrices in Matrix Hint-MLWE, whose hardness follows from standard MLWE under suitable parameters. Our results substantially reduce the required Gaussian masking widths, thereby enabling a compact lattice instantiation. We believe both contributions to be of independent interest.

Finally, we implement LatPRE in C and evaluate parameter sets targeting 128-bit security and supporting up to 2^{40} source encryptions and 2^{20} outgoing rekeys per key. For 256-bit messages, re-encrypted ciphertexts occupy only 1.8–2.8 KB, and re-encryption takes about 0.2–0.3 ms on a standard PC. Compared with the state of the art, LatPRE achieves around 23–37 times smaller re-encrypted ciphertexts and 7–9 times faster re-encryption. To demonstrate its practicality, we also integrate LatPRE with Proton Mail's open-source framework to build a hybrid post-quantum encrypted email-forwarding prototype.

***Index Terms*—Proxy re-encryption, honest re-encryption attacks, module learning with errors, lattice-based cryptography, post-quantum cryptography.**

1. Introduction

Proxy re-encryption (PRE) [1] enables a proxy to transform ciphertexts under one user's public key into ciphertexts that can be decrypted by another user, without learning the underlying plaintexts. The original recipient (delegator) provides the proxy with a re-encryption key (rekey) for the new recipient (delegatee), allowing these transformations to proceed while the delegator is offline. We refer to a ciphertext directly encrypted for a delegator as a *source ciphertext* and the proxy's output as a *re-encrypted ciphertext*. Single-hop PRE supports a single re-encryption of a source ciphertext from a delegator to a delegatee (e.g., Alice → Bob and Alice → Carlo); multi-hop PRE additionally supports successive re-encryptions (e.g., Alice → Bob → Carlo).

In a typical setting, PRE allows a proxy to re-encrypt a source ciphertext of a symmetric data key for multiple delegatees while leaving the file or email encrypted with that key unchanged. Consequently, re-encryption cost and output size are particularly important in practice. This delegation capability makes PRE a versatile cryptographic tool for a broad range of real-world applications, including encrypted file sharing [2], [3], [4], email forwarding [5], [6], [7], digital identity management [8], [9], and payment PIN translation [10]. For example, Proton Mail,[1] an email service with over 100 million users, uses a single-hop PRE [1], [7] to enable end-to-end encrypted email auto-forwarding.[2]

However, standard chosen-plaintext attack (CPA) security for PRE does not guarantee confidentiality against delegatees who receive re-encrypted ciphertexts. Cohen [11] showed that a CPA-secure scheme can reveal the delegator's secret key after a delegatee obtains just one honestly re-encrypted ciphertext. To address this gap, they introduced a stronger *honest re-encryption attack* (HRA) security, which ensures that decrypting re-encryptions of some ciphertexts originally under a public key does not help break other ciphertexts under the same key. This security guarantee is essential for the sharing and forwarding applications discussed above.

[*] *Corresponding author.*

1. https://proton.me/mail
2. https://proton.me/blog/email-forwarding

TABLE 1. COMPARISON OF LATTICE-BASED PRE SCHEMES.

Scheme	Security	Public Enc.	No Proxy Secret	Impl.	$\lvert\mathsf{rk}\rvert$ (KB)	$\lvert\mathsf{ct}\rvert$ (KB)	$\lvert\mathsf{ct}'\rvert$ (KB)	ReEnc (ms)
[16]	HRA	✓	✓	✓	129.0	65.0	65.0	2.2
[17]*	bCCA2	✓	✓	×	4.3 GB	6.2	6.2	—
[18]*	HRA	✓	×	×	0.3 GB	748.5	726.5	—
[19]*	HRA	×	×	×	58.5	6.5	6.5	—
LatPRE-L	HRA	✓	✓	✓	52.5	3.7	**1.8**	**0.2**
LatPRE-M	HRA	✓	✓	✓	98.5	5.7	**2.2**	**0.3**
LatPRE-H	HRA	✓	✓	✓	110.3	8.1	**2.8**	**0.3**

* These papers do not report concrete communication sizes; for a fair comparison, we estimate them for single-hop delivery of 256-bit messages at 128-bit security. For [17], we only set both the decryption and re-encryption query bounds to one.

Notes: Public Enc. means that encryption requires no secret key. No Proxy Secret means that the proxy requires no secret beyond the rekeys. Impl. indicates whether an implementation is provided. $\lvert\mathsf{rk}\rvert$, $\lvert\mathsf{ct}\rvert$, and $\lvert\mathsf{ct}'\rvert$ denote the sizes of the rekey, source ciphertext, and re-encrypted ciphertext, respectively. bCCA2 denotes bounded CCA2. A dash (—) denotes an unavailable timing. LatPRE-L/M/H share the same 128-bit security and differ in usage budgets (see Table 5).

TABLE 2. COMPARISON OF SPECTRAL-NORM BOUNDS IN MATRIX HINT-MLWE.

Instance	Original $\sqrt{B}$	Our $\sqrt{B}$	Improve
[20]*	141.76	63.01	2.25×
[21]†	23,170.48	536.94	43.15×
LatPRE-L‡	2.52×10^{7}	3.74×10^{5}	67.25×
LatPRE-M‡	5.04×10^{9}	6.71×10^{7}	75.02×
LatPRE-H‡	9.91×10^{11}	1.21×10^{10}	81.58×

* Independent real 256×4096 CBD_1 matrix from [20, Lemma 4].
† The 128-bit parameters in [21, Table 3]. Our bound additionally assumes honestly generated recipient keys.
‡ Original $\sqrt{B}$ uses the bound of [21]; the real-matrix bound in [20] does not directly apply to LatPRE's structured ring matrices.

Meanwhile, advances in quantum computing are driving the transition to post-quantum cryptography. NIST finalized its first post-quantum standards in 2024 and urged organizations to begin migrating. Lattice-based schemes underpin its primary key-encapsulation and signature standards [12], [13]. However, most existing PRE constructions still rely on discrete-logarithm [1], [14] or bilinear-pairing assumptions [2], [5] that are vulnerable to quantum computers [15].

1.1. Existing Works and Challenges

Lattice-based PRE provides a promising post-quantum alternative [22], [23], [24], but existing constructions make different trade-offs between security, usability, and efficiency. Table 1 summarizes representative lattice-based schemes.

Specifically, Cohen et al. [16] construct lattice-based HRA-secure PRE using statistical noise flooding, leading to substantial ciphertext overhead. Sato and Shikata [17] achieve bounded CCA2 security with compact ciphertexts, but their rekey size grows quartically with the oracle-query bound. Yin et al. [18] avoid statistical noise flooding through discrete Gaussian padding, but retain a *private proxy secret* and rely on a plain-LWE analysis that does not directly extend to standard Ring-LWE. This proxy secret requires additional protection and secure transfer when replacing or replicating the proxy. Their subsequent work [19] builds on ring lattices but also retains a private proxy secret and requires the delegator's secret key for encryption, restricting applications with arbitrary senders, such as email forwarding. Other works address adaptive corruption [25], fine-grained delegation [23], [24], unbounded multi-hop PRE [26], [27], and CSIDH-based PRE [28]. However, most of these constructions remain theoretical and have yet to demonstrate practical efficiency.

A central challenge in constructing HRA-secure PRE from lattices is that lattice key switching carries the source ciphertext's decryption error into the re-encrypted ciphertext, potentially leaking information about the delegator's secret key. Adding fresh noise alone may be insufficient, since repeated re-encryptions of the same source ciphertext can enable averaging attacks. To resist such attacks, Yin et al. [18], [19] use a private proxy secret to derive a fixed mask for each source ciphertext, limiting usability. Statistical noise flooding instead hides the error by adding much larger noise, but increasing the required modulus and communication costs. Therefore, an efficient alternative must hide this leakage while accounting for repeated re-encryptions of the same source ciphertext.

Cohen [11] also provides a generic framework for PRE that derives HRA security from CPA security and re-encryption simulatability. The latter requires a simulator to reproduce the distributions of re-encrypted ciphertexts, given a source ciphertext, its message, and the delegatee's secret key, without the delegator's secret key or the associated rekey. This lets a CPA reduction answer the additional re-encryption queries permitted by HRA. However, realizing this property with compact lattice parameters remains challenging: re-encryption inherits a secret-dependent error from the source ciphertext that is not directly available to the simulator.

These limitations also matter for real-world deployment. For example, although Proton Mail now supports post-quantum protection for regular encrypted email communication[3], it cannot yet provide such protection for end-to-end encrypted email auto-forwarding[4].

Overall, these considerations lead to our question:

Question: Can we construct a compact lattice-based HRA-secure PRE with public encryption, fast re-encryption, and no additional proxy secret?

1.2. Contributions

In this work, we answer the above question affirmatively and make the following three contributions. We provide a technical overview of our approach in Section 1.3.

New generic framework for HRA-secure PRE. We first propose a new (Q, M)-hint CPA security notion for PRE,

3. https://proton.me/blog/introducing-post-quantum-encryption
4. https://proton.me/support/mail-post-quantum-protection

which requires confidentiality even when auxiliary hints are available for source ciphertexts and hidden rekeys. Here, Q and M bound source-encryption queries and distinct outgoing rekeys per key, respectively. We pair this notion with hint-assisted re-encryption simulatability and prove that the two properties imply HRA security (Section 3). The framework strengthens Cohen's CPA-to-HRA approach [11] by allowing hints to supply the information needed for simulation. The framework also preserves public encryption: any sender can encrypt using only public parameters and the delegator's public key. We believe this generic framework could be of independent interest and inspire other post-quantum instantiations, such as code-based schemes.

Refining Matrix Hint-MLWE with tighter bounds. Matrix Hint-MLWE [20], [21] captures the hardness of MLWE in the presence of masked linear hints about its secret and error, specified by a public leakage matrix. Its reduction from standard MLWE requires Gaussian masks whose width scales with $\sqrt{B}$, where $\sqrt{B}$ bounds the spectral norm of the leakage matrix. We refine this reduction by introducing a spectral condition that allows the spectral bound to fail with negligible probability (Section 4). Exploiting the algebraic structure and independent coefficient randomness of honestly sampled leakage matrices, we combine negacyclic Fourier decomposition with spherical averaging to derive tighter bounds on their spectral norms. Our analysis covers common coefficient distributions, including centered binomial distributions CBD_η, bounded uniform distributions, and discrete Gaussian distributions.

For CBD_η, we achieve an asymptotic reduction factor of $\sqrt{2pd\eta}$ in the sufficient masking width compared to the bound used in [21]; here p is the number of columns of the leakage matrix, d is the ring degree, and η is the coefficient bound of CBD_η. Table 2 summarizes the gains for [20], [21] and our three parameter sets; for LatPRE, our results reduce $\sqrt{B}$ by approximately 67–$82\times$ relative to directly applying the bound of [21]. Overall, these improvements can reduce the required moduli and communication costs, thereby benefiting other applications of Matrix Hint-MLWE as well. We therefore believe our analysis is of independent interest.

LatPRE: a compact instantiation and implementation. We instantiate our generic PRE framework from the standard Module Learning with Errors (MLWE) assumption to obtain LatPRE, a compact single-hop HRA-secure PRE scheme supporting public encryption without an additional proxy secret (Section 5). This construction is based on our refined Matrix Hint-MLWE and admits perfect hint-assisted re-encryption simulation. We provide three parameter sets, Light (L), Middle (M), and Heavy (H), with $(Q, M) = (2^{20}, 2^{10}), (2^{30}, 2^{15}), (2^{40}, 2^{20})$, respectively, targeting 128-bit security and accommodating different encryption rates and key lifetime requirements (Table 5). Here, Q and M bound the numbers of source encryptions and distinct outgoing rekeys for each delegator's public key, respectively. For example, L supports about two years at the rate of one encryption per minute, while H supports about 35 years at the rate of 1,000 encryptions per second.

For 256-bit messages, re-encrypted ciphertexts occupy only 1.8–2.8 KB, and our implementation performs re-encryption in only 0.2–0.3 ms. Compared with the state of the art [16], LatPRE achieves approximately 23–$37\times$ smaller re-encrypted ciphertexts and 7–$9\times$ faster re-encryption (Section 6). Finally, we also integrate LatPRE with Proton Mail's open-source framework to build a hybrid post-quantum encrypted email auto-forwarding prototype (Section 6.1). Our results show that, for a 1 MB email, our hybrid forwarding adds approximately 0.4 ms and 1.8 KB compared with Proton Mail's quantum-vulnerable forwarding.

1.3. Technical Overview

In this subsection, we provide a self-contained overview of our techniques for constructing a compact lattice-based HRA-secure PRE scheme. We start from Cohen's CPA-to-HRA framework [11], explain how hints enable our lattice instantiation, and finally address the parameter cost of supporting large usage budgets.

Recall: Cohen's CPA-to-HRA framework. Cohen [11] shows that HRA security can be derived from CPA security when re-encrypted ciphertexts can be simulated from the source ciphertext ct_i, its message m, and the delegatee's secret key sk_j, namely,

$$\mathsf{ReEnc}(\mathsf{rk}_{i\to j}, \mathsf{ct}_i) \approx \mathsf{ReEncSim}(\mathsf{ct}_i, \mathsf{m}, \mathsf{sk}_j),$$

where $\approx$ denotes statistical indistinguishability given the simulator's inputs. At a high level, this simulator lets a CPA reduction answer requests for the re-encryption of honestly generated source ciphertexts from honest delegators to *corrupted delegatees*, as allowed by the HRA game, while forwarding the remaining requests to its CPA game.

For example, consider the single-hop CPA-secure pairing-based construction [2]. Let $e : G_1 \times G_1 \to G_T$ be a non-degenerate bilinear pairing between prime-order groups, let g generate G_1, and set $Z = e(g, g)$. The delegator has public key $\mathsf{pk}_i = (Z^{a_1}, g^{a_2})$ and secret key $\mathsf{sk}_i = (a_1, a_2)$; the delegatee has $\mathsf{pk}_j = (Z^{b_1}, g^{b_2})$ and $\mathsf{sk}_j = (b_1, b_2)$. A source ciphertext of $m \in G_T$ is $\mathsf{ct}_i = (g^r, v)$, where $v = mZ^{a_1 r}$ for uniform r. The delegator computes the rekey $\mathsf{rk}_{i\to j} = (g^{b_2})^{a_1}$ and sends it to the proxy. The proxy outputs the re-encrypted ciphertext $(e(g^r, (g^{b_2})^{a_1}), v) = (Z^{a_1 b_2 r}, v)$. Given $\mathsf{ct}_i = (g^r, v)$, m, and $\mathsf{sk}_j = (b_1, b_2)$, a simulator can compute $((v/m)^{b_2}, v)$, exactly reproducing the above re-encrypted ciphertext. Therefore, Cohen's framework lifts the CPA security of this construction to HRA security.

However, in lattice-based PRE, re-encrypted ciphertext carries a secret-dependent decryption error that is not determined by the plaintext, obstructing this direct simulation. Statistical noise flooding can hide this dependence [16], but incurs substantial parameter overhead.

HRA security from (Q, M)-hint CPA. To address this obstacle, we first propose a new security notion for PRE, called *(Q, M)-hint CPA security*. Here, Q and M bound source-encryption queries and distinct outgoing rekeys per

key, respectively. Specifically, we augment CPA-secure PRE with an auxiliary algorithm $\mathsf{HintGen}$ that generates a hint $h_{\mathsf{ct}_i, i\to j}$, supplying the information needed to simulate re-encryption of source ciphertext ct_i to delegatee j. This notion requires CPA security even when the adversary is given such hints for honestly generated source ciphertexts under honest delegators' keys, with corrupted delegatees.

Accordingly, we strengthen Cohen's re-encryption simulatability by allowing the simulator to use the hint and maintain its own private state st. Specifically, we require the ciphertext outputs to satisfy

$$\mathsf{ReEnc}(\mathsf{rk}_{i\to j}, \mathsf{ct}_i) \approx \mathsf{ReEncSim}(\mathsf{ct}_i, h_{\mathsf{ct}_i, i\to j}, \mathsf{st}).$$

Besides public information, the simulator only receives the source ciphertext and its hint, and retains private state across calls. Statistical indistinguishability must hold over the complete query sequence. A variant that sets $\mathsf{HintGen} \equiv \perp$ and additionally provides the simulator with the message and the delegatee's secret key recovers Cohen's framework [11].

Thus, the reduction can obtain the corresponding hint and employ the simulator to answer the requests to re-encrypt source ciphertexts for honest delegators to corrupted delegatees. Consequently, HRA security can be derived from (Q, M)-hint CPA security together with hint-assisted simulatability. We note that the hints and their auxiliary algorithms are used only in the proof and are not part of the deployed protocol.

A lattice-based (Q, M)-hint CPA PRE. We construct the first lattice-based PRE scheme achieving (Q, M)-hint CPA security. We start from the ML-KEM-style CPA-secure PRE [17] and show how to extend it to achieve this goal.

Over $\mathcal{R}_q = \mathbb{Z}_q[X]/(X^d + 1)$, let $\mathbf{A} \in \mathcal{R}_q^{k\times \ell}$ be public and let user i have public key $\mathbf{b}_i = \mathbf{A}^\top \mathbf{s}_i + \mathbf{e}_i$ and secret key $\mathbf{s}_i$. For source encryption of a message $m \in \{0,1\}^d$ under the delegator's public key $\mathbf{b}_i$, it samples fresh randomness $(\mathbf{r}, \mathbf{e}_u, e_v)$ and outputs a source ciphertext $(\mathbf{u}, v)$, where

$$\mathbf{u} = \mathbf{A}\mathbf{r} + \mathbf{e}_u, \qquad v = \langle \mathbf{b}_i, \mathbf{r}\rangle + e_v + \lfloor q/2 \rfloor m.$$

Source decryption rounds $v - \langle \mathbf{s}_i, \mathbf{u}\rangle = \lfloor q/2 \rfloor m + \nu_{\mathrm{S}}$ to recover m, where the source decryption error is $\nu_{\mathrm{S}} = e_v + \langle \mathbf{e}_i, \mathbf{r}\rangle - \langle \mathbf{s}_i, \mathbf{e}_u\rangle$.

For rekey generation, let B_{g} be a gadget base and set $\mathbf{G} = \mathbf{I}_k \otimes (1, B_{\mathrm{g}}, \ldots, B_{\mathrm{g}}^{L_{\mathrm{g}}-1})$, with $L_{\mathrm{g}} = \lceil \log_{B_{\mathrm{g}}} q \rceil$ and $w = kL_{\mathrm{g}}$. It encrypts gadget multiples of $-\mathbf{s}_i$ under $\mathbf{b}_j$ and outputs

$$\mathbf{K}_{ij} = \mathbf{K}_{ij}^0 - (\mathbf{0}, \mathbf{G}^\top \mathbf{s}_i)^\top \in \mathcal{R}_q^{(k+1)\times w},$$

where $\mathbf{K}_{ij}^0 = (\mathbf{A}^\top, \mathbf{b}_j)^\top \mathbf{R}_{ij} + \mathbf{E}_{ij}$ for randomness $\mathbf{R}_{ij}, \mathbf{E}_{ij}$.

For re-encryption, the proxy decomposes $\mathbf{u}$ as $\mathbf{d}_{\mathsf{ct}} = \mathsf{BD}(\mathbf{u})$, with $\mathbf{G}\mathbf{d}_{\mathsf{ct}} = \mathbf{u}$, and outputs

$$(\mathbf{u}'^\top, v')^\top = (\mathbf{0}^\top, v)^\top + \mathbf{K}_{ij}\mathbf{d}_{\mathsf{ct}}.$$

Substituting $\mathbf{K}_{ij}$ gives $(\mathbf{0}^\top, v - \langle \mathbf{s}_i, \mathbf{u}\rangle)^\top + \mathbf{K}_{ij}^0 \mathbf{d}_{\mathsf{ct}}$.

To decrypt the re-encrypted ciphertext, the delegatee rounds $v' - \langle \mathbf{s}_j, \mathbf{u}'\rangle = \lfloor q/2 \rfloor m + \nu_{\mathrm{S}} + \boldsymbol{\epsilon}_{ij}^\top \mathbf{d}_{\mathsf{ct}}$, where $\boldsymbol{\epsilon}_{ij}^\top = \mathbf{e}_j^\top \mathbf{R}_{ij} + \mathbf{e}_{ij,v}^\top - \mathbf{s}_j^\top \mathbf{E}_{ij,u}$ and $\mathbf{E}_{ij} = (\mathbf{E}_{ij,u}^\top, \mathbf{e}_{ij,v})^\top$. Here, the delegatee can observe a decryption error containing ν_{S}, which may leak information about the source secret key.

To this end, we let the proxy add an encryption of a Gaussian mask $y_{\mathsf{ct},ij}$ of the form $(\mathbf{0}^\top, y_{\mathsf{ct},ij})^\top + \mathbf{f}_0$ to mask this source decryption error during the re-encryption, where $\mathbf{f}_0 = (\mathbf{A}^\top, \mathbf{b}_j)^\top \mathbf{r}_0 + \mathbf{e}_0$ is a fresh encryption of zero.

Meanwhile, to prevent averaging attacks, we set $\mathsf{rk}_{i\to j} = (\mathbf{K}_{ij}, \xi_{ij})$ with an independent PRF seed ξ_{ij} and derive $y_{\mathsf{ct},ij} = \mathsf{GSamp}_{\sigma_y}(\mathsf{PRF}_{\xi_{ij}}(\mathsf{ct}))$, where $\mathsf{GSamp}_{\sigma_y}$ uses the PRF output as its sampling randomness to generate a discrete Gaussian mask of width σ_y. This fixes the mask for each ciphertext–rekey pair *without a global proxy secret*. Furthermore, this deterministic mask derivation supports repeated re-encryption of the same source ciphertext under a fixed rekey without consuming either budget. The secret-dependent part of the source error is now masked as

$$h^{\mathrm{lin}}_{\mathsf{ct},i\to j} = \langle \mathbf{e}_i, \mathbf{r}\rangle - \langle \mathbf{s}_i, \mathbf{e}_u\rangle + y_{\mathsf{ct},ij},$$

so the delegatee decryption error becomes $e_v + h^{\mathrm{lin}}_{\mathsf{ct},i\to j} + \boldsymbol{\epsilon}_{ij}^\top \mathbf{d}_{\mathsf{ct}}$. Looking ahead, this linear leakage $h^{\mathrm{lin}}_{\mathsf{ct},i\to j}$ matches the masked linear hint structure of Matrix Hint-MLWE.

Towards simulatability, the phase hint can be generated by $\mathsf{HintGen}$ as

$$h_{\mathsf{ct},i\to j} = v - \langle \mathbf{s}_i, \mathbf{u}\rangle + y_{\mathsf{ct},ij},$$

so the re-encryption can be simulated with hint as

$$(\mathbf{u}'^\top, v')^\top = (\mathbf{0}^\top, h_{\mathsf{ct},i\to j})^\top + \mathbf{K}_{ij}^0 \mathbf{d}_{\mathsf{ct}} + \mathbf{f}_0,$$

where the simulator samples $\mathbf{K}_{ij}^0$ using the delegatee's public key, retains it for re-encryptions from i to j, and generates fresh $\mathbf{f}_0$ per request. This preserves the joint distribution and gives perfect hint-assisted simulation.

For hint-CPA security, we jointly analyze the linear leakage about the source secret and error. At most Q source encryptions and M distinct outgoing rekeys per source key yield at most QM linear hints. For $p = k + \ell$, set $\mathbf{z}_i = (\mathbf{e}_i^\top, \mathbf{s}_i^\top)^\top$ and let $\mathbf{L} \in \mathcal{R}^{Q\times p}$ have rows $\mathbf{L}_t = (\mathbf{r}_t^\top, -\mathbf{e}_{u,t}^\top)$. Stacking the linear hints gives

$$\mathbf{h}_i^{\mathrm{lin}} = \widetilde{\mathbf{L}}\mathbf{z}_i + \mathbf{y}_i, \qquad \widetilde{\mathbf{L}} = (\mathbf{L}^\top, \ldots, \mathbf{L}^\top)^\top \in \mathcal{R}^{QM\times p}, \quad (1)$$

where $\widetilde{\mathbf{L}}$ stacks M copies of $\mathbf{L}$ and serves as the *leakage matrix* in Matrix Hint-MLWE, while $\mathbf{y}_i$ collects the Gaussian masks.

Given such a Matrix Hint-MLWE sample, the reduction retrieves $(\mathbf{r}_t, \mathbf{e}_{u,t})$ from row $\mathbf{L}_t$ and samples fresh $e_{v,t}$ to generate a source ciphertext $\mathsf{ct}_t = (\mathbf{u}_t, v_t)$ of m under $\mathbf{b}_i$ as above. If the rekey from i to j has index ν, it computes the corresponding phase hint as $h_{\mathsf{ct}_t,i\to j} = v_t - \langle \mathbf{b}_i, \mathbf{r}_t\rangle + h^{\mathrm{lin}}_{i,t,\nu}$, without the source secret.

Overall, Matrix Hint-MLWE ensures that the honest source public key $\mathbf{b}_i$ remains computationally indistinguishable from uniform, even given these hints. We note that, under suitable parameter choices, the hardness of Matrix Hint-MLWE follows from standard MLWE. For the full construction and analysis, see Section 5; here we have unified key notation and omitted ciphertext compression for simplicity.

Tighter spectral bounds for leakage matrices. The construction above achieves (Q, M)-hint CPA security through Matrix Hint-MLWE, but its concrete efficiency depends on the Gaussian width required for the hint mask y. Specifically, for the hints in Equation (1), the required Gaussian width scales with a bound on the spectral norm of $\Gamma(\widetilde{\mathbf{L}})$, where $\Gamma(\cdot)$ denotes the integer coefficient matrix representing ring multiplication. Here, $\|\cdot\|_{\mathrm{op}}$ denotes the spectral norm, i.e., the largest singular value. Since $\|\Gamma(\widetilde{\mathbf{L}})\|_{\mathrm{op}}^2 = M\|\Gamma(\mathbf{L})\|_{\mathrm{op}}^2$, this width scales as $\sqrt{MB_Q}$, where B_Q bounds $\|\Gamma(\mathbf{L})\|_{\mathrm{op}}^2$. This coefficient-bound analysis in [21] gives the worst-case bound $B_Q^{\mathrm{wc}} = Qpd^2\beta^2$ when every coefficient has absolute value at most β. For large usage budgets (Q, M), this bound requires substantially wider Gaussian masks, increasing the modulus needed for correctness and hence the communication costs.

Our key observation is that this worst-case bound ignores the randomness of the leakage matrix. In our construction, $\mathbf{L}$ has honestly sampled, independent, centered CBD_1 coefficients, allowing substantially tighter bounds that hold with high probability. To use such bounds, we first refine the Matrix Hint-MLWE reduction to allow $\|\Gamma(\mathbf{L})\|_{\mathrm{op}}^2 \le B_Q$ to fail with probability at most δ_{spec}, adding at most δ_{spec} to the reduction loss.

The challenge is that $\Gamma(\mathbf{L})$ contains repeated negacyclic shifts of the same random coefficients, so its entries are not independent. To address this, we apply Fourier decomposition to obtain $\|\Gamma(\mathbf{L})\|_{\mathrm{op}} = \max_{\zeta^d=-1}\|\mathbf{L}(\zeta)\|_{\mathrm{op}}$, reducing the analysis to the $Q \times p$ complex frequency matrices $\mathbf{L}(\zeta)$. Since the frequency matrices are linear in the independent coefficients, we first control their linear exponential moments using the coefficient moment-generating function (MGF) bound $\mathbb{E}[e^{t\xi}] \le e^{\tau^2 t^2/2}$, for some $\tau > 0$ and all real t. A Gaussian integral then converts these bounds into an exponential-moment bound for $\|\mathbf{L}(\zeta)x\|_2^2$ for each fixed unit vector x. However, the direction attaining the spectral norm depends on the random matrix, so fixed-direction bounds alone do not directly control its spectral norm. We therefore use spherical averaging: a large eigenvalue of the Gram matrix $\mathbf{L}(\zeta)^*\mathbf{L}(\zeta)$ forces the exponential average over the complex unit sphere to be large. Markov's inequality and a union bound over the frequencies then give a spectral tail bound, from which B_Q can be computed by one-dimensional optimization.

Overall, for fixed p, d, τ and δ_{spec}, our spherical-averaging bound satisfies

$$B_Q^{\mathrm{sph},*} = d\tau^2 Q(1 + o(1)) \qquad (Q \to \infty).$$

For concrete parameter selection, we compute a finite-Q bound B_Q at the prescribed failure probability δ_{spec}. We then use MB_Q as the squared spectral-norm bound for $\Gamma(\widetilde{\mathbf{L}})$ in the Matrix Hint-MLWE reduction to select the Gaussian masking width σ_y.

2. Preliminaries

In this section, we introduce our notation and basic tools, recall the MLWE assumption, and provide formal definitions of PRE syntax, correctness, and CPA and HRA security.

2.1. Notation

Let λ denote the security parameter and $[n] = \{1, \ldots, n\}$ for a positive integer n. We write PPT for probabilistic polynomial time and $\mathsf{negl}(\lambda)$ for a negligible function. Parameters and distributions are implicitly indexed by λ in asymptotic security statements. We use $x \leftarrow \mathcal{D}$ for sampling from a distribution $\mathcal{D}$, $\mathcal{U}(S)$ for the uniform distribution on a finite set S, and $x := f(y)$ for deterministic assignment.

Rings and matrices. Let $\mathcal{R} = \mathbb{Z}[X]/(X^d + 1)$ and $\mathcal{R}_q = \mathcal{R}/q\mathcal{R}$, where $d \ge 2$ is a power of two and $q \ge 2$. Ring elements are represented by polynomials of degree less than d, with coefficients indexed from zero. Bold lowercase and uppercase letters denote ring vectors and matrices, respectively. Vectors are columns unless stated otherwise, and $\langle \mathbf{a}, \mathbf{b} \rangle = \mathbf{a}^\top \mathbf{b}$. We write $^\top$ for transpose, * for conjugate transpose, $\mathbf{I}_n$ for the $n \times n$ identity matrix, and $\otimes$ for the Kronecker product.

For $f \in \mathcal{R}$, let $\mathrm{coef}(f) \in \mathbb{Z}^d$ denote its coefficient column vector. Define its negacyclic multiplication matrix by $\Gamma(f) := \big(\mathrm{coef}(f), \mathrm{coef}(Xf), \ldots, \mathrm{coef}(X^{d-1}f)\big)$, where products are computed in $\mathcal{R}$. For $\mathbf{L} \in \mathcal{R}^{a\times b}$, $\Gamma(\mathbf{L}) \in \mathbb{Z}^{ad\times bd}$ replaces each polynomial entry by its negacyclic matrix. Writing $\mathrm{coef}(\mathbf{z})$ for the concatenated coefficient vector of $\mathbf{z} \in \mathcal{R}^b$, we have $\mathrm{coef}(\mathbf{Lz}) = \Gamma(\mathbf{L})\,\mathrm{coef}(\mathbf{z})$.

For ring elements and vectors, $\|\cdot\|_\infty$ is the largest absolute coefficient; for elements of $\mathcal{R}_q$, we use centered coefficient representatives. For real or complex vectors, $\|\cdot\|_2$ is the Euclidean norm; for a matrix H, $\|H\|_{\mathrm{op}} = \sup_{\|x\|_2=1}\|Hx\|_2$ is its spectral norm, i.e., its largest singular value. For Hermitian H, K, $H \succeq K$ means that $H - K$ is positive semidefinite; $\lambda_{\max}(H)$ denotes the largest eigenvalue. We write $\lfloor x \rceil$ for nearest-integer rounding and $\lfloor x \rceil_{a\to b} := \lfloor (b/a)x \rceil \bmod b$ for scaling and rounding from modulus a to b, applied coefficientwise.

Distributions. For a distribution χ on $\mathbb{Z}$, we also use χ for the distribution on $\mathcal{R}$ obtained by sampling all d coefficients independently. Products such as χ^a and $\chi^{a\times b}$ denote independent sampling of ring vectors and matrices. For a positive integer η, CBD_η denotes the centered binomial distribution (CBD) of $\sum_{j=1}^{\eta}(B_j - B'_j)$ for independent $B_j, B'_j \leftarrow \mathcal{U}(\{0,1\})$.

For $\sigma > 0$, the centered discrete Gaussian $\mathcal{D}_{\mathbb{Z},\sigma}$ assigns probability proportional to $\exp(-\pi x^2/\sigma^2)$ to each $x \in \mathbb{Z}$. We write $\mathcal{D}_\sigma$ for its coefficientwise product distribution on $\mathcal{R}$. Here σ is the Gaussian width; $\sigma/\sqrt{2\pi}$ is the standard deviation of the corresponding Gaussian.

For $D \ge 1$ and $\epsilon \in (0,1)$, the smoothing parameter $\eta_\epsilon(\mathbb{Z}^D)$ is the least $s > 0$ such that $\sum_{z\in\mathbb{Z}^D\setminus\{0\}} \exp(-\pi s^2\|z\|_2^2) \le \epsilon$. We use the bound [29] $\eta_\epsilon(\mathbb{Z}^D) \le \sqrt{\ln(2D(1+1/\epsilon))/\pi}$.

For distributions P, Q on a common countable space Ω, we write $P \approx_\epsilon Q$ if their statistical distance $\frac{1}{2}\sum_{x\in\Omega}|P(x) - Q(x)|$ is at most ϵ, and $P \approx_{\mathrm{c}} Q$ for computational indistinguishability.

Gadget decomposition. For odd $q > 2$ and an odd base $B_{\mathrm{g}} \ge 3$, let $L_{\mathrm{g}} = \lceil \log_{B_{\mathrm{g}}} q \rceil$, $w = kL_{\mathrm{g}}$, and

$\mathbf{G} = \mathbf{I}_k \otimes (1, B_g, \ldots, B_g^{L_g - 1}) \in \mathcal{R}_q^{k \times w}$. Let $\mathsf{BD} : \mathcal{R}_q^k \to \mathcal{R}^w$ denote the deterministic coefficientwise balanced base-B_g decomposition of centered representatives. For every $\mathbf{u} \in \mathcal{R}_q^k$, $\mathbf{d} = \mathsf{BD}(\mathbf{u})$ satisfies

$$\mathbf{G}\mathbf{d} = \mathbf{u} \pmod q, \qquad \|\mathbf{d}\|_\infty \le (B_g - 1)/2.$$

Pseudorandom functions. Let $\mathsf{PRF} : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$ be an efficiently computable keyed function family. For an oracle distinguisher $\mathcal{B}$, define

$$\mathsf{Adv}^{\mathsf{PRF}}_{\mathcal{B}}(\lambda) := \left| \Pr[\mathcal{B}^{\mathsf{PRF}_\xi}(1^\lambda) = 1] - \Pr[\mathcal{B}^{F}(1^\lambda) = 1] \right|,$$

where $\xi \leftarrow \mathcal{U}(\mathcal{K})$ and $F : \mathcal{X} \to \mathcal{Y}$ is a random function. We say that PRF is secure if this advantage is negligible in λ for every PPT distinguisher $\mathcal{B}$.

2.2. Lattice Assumption

We recall the Module-LWE assumption.

Definition 2.1 (Module-LWE)**.** Let k, ℓ be positive integers, let $q \ge 2$ be an integer, and let χ_s, χ_e be efficiently sampleable coefficient distributions. Define $\chi := \chi_e^\ell \times \chi_s^k$ and $\mathsf{par} := (\mathcal{R}, k, \ell, q, \chi)$. Independently sample $\mathbf{A} \leftarrow \mathcal{U}(\mathcal{R}_q^{k \times \ell})$, $(\mathbf{e}^\top, \mathbf{s}^\top)^\top \leftarrow \chi$, and $\mathbf{u} \leftarrow \mathcal{U}(\mathcal{R}_q^\ell)$, and set $\mathbf{b} = \mathbf{A}^\top \mathbf{s} + \mathbf{e} \bmod q$. The decisional Module-LWE problem $\mathsf{MLWE}_{\mathcal{R},k,\ell,q,\chi}$ is to distinguish $(\mathbf{A}, \mathbf{b})$ from $(\mathbf{A}, \mathbf{u})$. We say this problem is hard if, for every PPT adversary $\mathcal{A}$, the following advantage is negligible in λ:

$$\mathsf{Adv}^{\mathsf{MLWE}}_{\mathsf{par},\mathcal{A}} := \left| \Pr[\mathcal{A}(\mathbf{A}, \mathbf{b}) = 1] - \Pr[\mathcal{A}(\mathbf{A}, \mathbf{u}) = 1] \right|.$$

2.3. Proxy Re-Encryption

Following [11], [16], we define PRE with explicit per-key budgets Q for source-encryption queries and M for generating distinct outgoing rekeys.

Definition 2.2 (Proxy re-encryption)**.** A PRE scheme Π with message space $\mathcal{M}$ and ciphertext space $\mathcal{C}$ consists of six algorithms:

- $\mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda, Q, M)$: On input the unary security parameter 1^λ and budgets Q, M, it outputs the public parameters pp.
- $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$: On input the public parameters pp, it outputs a public–secret key pair $(\mathsf{pk}, \mathsf{sk})$.
- $\mathsf{ct}_i \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{pk}_i, \mathsf{m})$: On input the public parameters pp, a public key pk_i, and a message $\mathsf{m} \in \mathcal{M}$, it outputs a ciphertext $\mathsf{ct}_i \in \mathcal{C}$.
- $\mathsf{rk}_{i \to j} \leftarrow \mathsf{ReKeyGen}(\mathsf{pp}, \mathsf{sk}_i, \mathsf{pk}_j)$: On input the public parameters pp, the delegator's secret key sk_i, and the delegatee's public key pk_j, it outputs a re-encryption key $\mathsf{rk}_{i \to j}$ for $i \ne j$.
- $\mathsf{ct}_j / \perp \leftarrow \mathsf{ReEnc}(\mathsf{pp}, \mathsf{pk}_i, \mathsf{pk}_j, \mathsf{rk}_{i \to j}, \mathsf{ct}_i)$: On input the public parameters pp, public keys $\mathsf{pk}_i, \mathsf{pk}_j$, re-encryption key $\mathsf{rk}_{i \to j}$, and ciphertext ct_i, it outputs a ciphertext $\mathsf{ct}_j \in \mathcal{C}$ or $\perp$.
- $\mathsf{m} / \perp := \mathsf{Dec}(\mathsf{pp}, \mathsf{sk}, \mathsf{ct})$: On input the public parameters pp, a secret key sk, and a ciphertext ct, it deterministically outputs a message $\mathsf{m} \in \mathcal{M}$ or $\perp$.

For single-hop schemes, we treat re-encrypted ciphertexts as *terminal ciphertexts*: further re-encryption returns $\perp$.

Correctness. Let $\delta : \mathbb{N}^3 \to [0, 1]$. For every λ, Q, M and $\mathsf{m} \in \mathcal{M}$, let $\mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda, Q, M)$ and independently generate $(\mathsf{pk}_i, \mathsf{sk}_i), (\mathsf{pk}_j, \mathsf{sk}_j) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$ for a delegator i and a delegatee $j \ne i$. Generate $\mathsf{rk}_{i \to j} \leftarrow \mathsf{ReKeyGen}(\mathsf{pp}, \mathsf{sk}_i, \mathsf{pk}_j)$ and $\mathsf{ct}_i \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{pk}_i, \mathsf{m})$; compute $\mathsf{ct}_j \leftarrow \mathsf{ReEnc}(\mathsf{pp}, \mathsf{pk}_i, \mathsf{pk}_j, \mathsf{rk}_{i \to j}, \mathsf{ct}_i)$. We say Π is δ-correct for one hop if

$$\Pr[\mathsf{Dec}(\mathsf{pp}, \mathsf{sk}_i, \mathsf{ct}_i) \ne \mathsf{m} \ \vee \ \mathsf{Dec}(\mathsf{pp}, \mathsf{sk}_j, \mathsf{ct}_j) \ne \mathsf{m}] \le \delta(\lambda, Q, M).$$

Security. Fix $\mathsf{ATK} \in \{\mathsf{CPA}, \mathsf{HRA}\}$ and polynomially bounded $N, Q, M \ge 1$. Figure 1 defines the security game, $\mathsf{Game}^{\mathsf{ATK}}_{\Pi,\mathcal{A}}$. The adversary fixes Cor before key generation and then receives the corresponding secret keys; later corruption is disallowed. All user keys are honestly generated. Each key permits at most Q source-encryption queries and M outgoing rekeys. An outgoing rekey counts toward M only when first generated, and subsequent queries for the same rekey do not consume additional budget.

A record $\mathsf{CT}[a] = (i, \mathsf{ct}, \mathsf{ch})$ stores its decryptor i, ciphertext ct, and a flag ch marking the challenge along with its re-encryption descendants. Counters n_i, c_i count source encryptions and outgoing rekeys, respectively. Oracle queries require valid user indices and messages, with $i \ne j$ for rekey and re-encryption queries. A failed oracle requirement returns $\perp$ without changing state; a failed game requirement yields an independent uniform output bit. The only difference between HRA and CPA is that HRA permits queries to re-encrypt honestly generated source ciphertexts under honest delegators' public keys for corrupted delegatees, whereas CPA rejects such queries. Thus, HRA security implies CPA security.

We say Π is ATK-secure if the following advantage is negligible for every PPT adversary and polynomially bounded N, Q, M:

$$\mathsf{Adv}^{\mathsf{ATK}}_{\Pi,\mathcal{A}}(\lambda, N, Q, M) := \left| \Pr[\mathsf{Game}^{\mathsf{ATK}}_{\Pi,\mathcal{A}}(1^\lambda, N, Q, M) = 1] - \tfrac{1}{2} \right|.$$

3. A Generic Framework for HRA-Secure PRE

We first introduce (Q, M)-hint CPA security for PRE and then show that, together with hint-assisted re-encryption simulatability, it implies HRA security.

3.1. (Q, M)-Hint CPA Security for PRE

Definition 3.1 ((Q, M)-hint CPA PRE)**.** A (Q, M)-hint CPA PRE scheme $\widehat{\Pi} = (\Pi, \mathsf{HintGen})$ augments a PRE scheme Π of Definition 2.2 with the following PPT auxiliary algorithm:

- $\mathsf{h} \leftarrow \mathsf{HintGen}(\mathsf{pp}, \mathsf{pk}_i, \mathsf{sk}_i, \mathsf{pk}_j, \mathsf{rk}_{i \to j}, \mathsf{ct})$: On input the public parameters pp, the delegator's key pair $(\mathsf{pk}_i, \mathsf{sk}_i)$, the delegatee's public key pk_j, the rekey $\mathsf{rk}_{i \to j}$, and an honestly generated source ciphertext ct under pk_i, it outputs a hint h for this ciphertext and rekey.

The auxiliary algorithm is used only in the security experiments and is not part of the deployed protocol. Correctness is inherited from Π, as specified in Section 2.3.

Experiment $\mathsf{Game}^{\mathsf{ATK}}_{\Pi,\mathcal{A}}(1^\lambda, N, Q, M)$
$\mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda, Q, M)$
$(\mathsf{Cor}, \mathsf{st}_0) \leftarrow \mathcal{A}_0(\mathsf{pp}, N)$
req: $\mathsf{Cor} \subsetneq [N]$
$(\mathsf{pk}_i, \mathsf{sk}_i) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$ for every $i \in [N]$
$\mathsf{RK}, \mathsf{CT} \leftarrow \emptyset$; $t \leftarrow 0$
$n_i, c_i \leftarrow 0$ for $i \in [N]$
$\mathcal{O} \coloneqq \{\mathcal{O}_{\mathsf{ReKey}}, \mathcal{O}_{\mathsf{Enc}}, \mathcal{O}_{\mathsf{ReEnc}}\}$
$(i^\star, \mathsf{m}_0, \mathsf{m}_1, \mathsf{st}_1) \leftarrow \mathcal{A}_1^{\mathcal{O}}((\mathsf{pk}_i)_{i\in[N]}, (\mathsf{sk}_i)_{i\in\mathsf{Cor}}, \mathsf{st}_0)$
req: $i^\star \notin \mathsf{Cor}$, $|\mathsf{m}_0| = |\mathsf{m}_1|$
$b \leftarrow \{0,1\}$; $\mathsf{ct}^\star \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{pk}_{i^\star}, \mathsf{m}_b)$
$t \leftarrow t+1$; $\mathsf{CT}[t] \leftarrow (i^\star, \mathsf{ct}^\star, 1)$; $a^\star \leftarrow t$
$b' \leftarrow \mathcal{A}_2^{\mathcal{O}}(a^\star, \mathsf{ct}^\star, \mathsf{st}_1)$
return $[b' = b]$

Oracle $\mathcal{O}_{\mathsf{ReKey}}(i, j)$
req: $i \in \mathsf{Cor}$ or $j \notin \mathsf{Cor}$
if $\mathsf{RK}[i,j] = \emptyset$ **then**
req: $c_i < M$
$\mathsf{RK}[i,j] \leftarrow \mathsf{ReKeyGen}(\mathsf{pp}, \mathsf{sk}_i, \mathsf{pk}_j)$
$c_i \leftarrow c_i + 1$
end if
return $\mathsf{RK}[i,j]$

Oracle $\mathcal{O}_{\mathsf{Enc}}(i, \mathsf{m})$
req: $n_i < Q$
$\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{pk}_i, \mathsf{m})$
$n_i \leftarrow n_i + 1$; $t \leftarrow t+1$
$\mathsf{CT}[t] \leftarrow (i, \mathsf{ct}, 0)$
return (t, ct)

Oracle $\mathcal{O}_{\mathsf{ReEnc}}(i, j, a)$
req: $\mathsf{CT}[a] \neq \emptyset$
$(i_a, \mathsf{ct}, \mathsf{ch}) \leftarrow \mathsf{CT}[a]$
req: $i_a = i$ and ($\mathsf{ch} = 0$ or $j \notin \mathsf{Cor}$)
req: $\mathsf{ATK} = \mathsf{HRA}$ or $i \in \mathsf{Cor}$ or $j \notin \mathsf{Cor}$
if $\mathsf{ct} = \perp$ or ct is terminal **then return** $\perp$
if $\mathsf{RK}[i,j] = \emptyset$ **then**
req: $c_i < M$
$\mathsf{RK}[i,j] \leftarrow \mathsf{ReKeyGen}(\mathsf{pp}, \mathsf{sk}_i, \mathsf{pk}_j)$
$c_i \leftarrow c_i + 1$
end if
$\mathsf{ct}' \leftarrow \mathsf{ReEnc}(\mathsf{pp}, \mathsf{pk}_i, \mathsf{pk}_j, \mathsf{RK}[i,j], \mathsf{ct})$
$t \leftarrow t+1$; $\mathsf{CT}[t] \leftarrow (j, \mathsf{ct}', \mathsf{ch})$
return (t, ct')

Figure 1. CPA and HRA security games for PRE.

Oracle $\mathcal{O}_{\mathsf{Hint}}(i, j, a)$
req: a was returned by $\mathcal{O}_{\mathsf{Enc}}$
$(i_a, \mathsf{ct}, \mathsf{ch}) \leftarrow \mathsf{CT}[a]$
req: $i_a = i$ and $\mathsf{ch} = 0$
req: $i \notin \mathsf{Cor}$ and $j \in \mathsf{Cor}$
if $\mathsf{ct} = \perp$ or ct is terminal **then**
return $\perp$
end if
if $\mathsf{RK}[i,j] = \emptyset$ **then**
req: $c_i < M$
$\mathsf{RK}[i,j] \leftarrow \mathsf{ReKeyGen}(\mathsf{pp}, \mathsf{sk}_i, \mathsf{pk}_j)$
$c_i \leftarrow c_i + 1$
end if
if $\mathsf{H}[i,j,a] = \emptyset$ **then**
$\mathsf{H}[i,j,a] \leftarrow \mathsf{HintGen}(\mathsf{pp}, \mathsf{pk}_i, \mathsf{sk}_i, \mathsf{pk}_j, \mathsf{RK}[i,j], \mathsf{ct})$
end if
return $\mathsf{H}[i,j,a]$

Figure 2. Additional oracle for (Q, M)-hint CPA security. Initially $\mathsf{H} \leftarrow \emptyset$.

Security. Let $\mathsf{Game}^{\mathsf{hintCPA}}_{\widehat{\Pi},\mathcal{A}}(1^\lambda, N, Q, M)$ be the (Q, M)-hint CPA security game obtained by augmenting the CPA game in Figure 1 with $\mathcal{O}_{\mathsf{Hint}}$ in Figure 2. The adversary may query $\mathcal{O}_{\mathsf{Hint}}$ adaptively before and after the challenge, alongside the other oracles. Specifically, for each source ciphertext ct_i returned by $\mathcal{O}_{\mathsf{Enc}}$ under an honest delegator's public key, it may obtain a hint for re-encryption to a corrupted delegatee.

We say $\widehat{\Pi}$ is (Q, M)-hint CPA secure if the following advantage is negligible for every PPT adversary and polynomially bounded N, Q, M:

$$\mathsf{Adv}^{\mathsf{hintCPA}}_{\widehat{\Pi},\mathcal{A}}(\lambda, N, Q, M) \coloneqq \left|\Pr\left[\mathsf{Game}^{\mathsf{hintCPA}}_{\widehat{\Pi},\mathcal{A}}((1^\lambda, N, Q, M)) = 1\right] - \tfrac{1}{2}\right|.$$

3.2. From (Q, M)-Hint CPA to HRA Security

We extend the re-encryption simulatability of [11] with hints and private simulator state. We present the single-hop result and discuss multi-hop extensions later.

Definition 3.2 (Hint-assisted re-encryption simulator)**.** A hint-assisted re-encryption simulator is a PPT algorithm defined as follows:

- $(\mathsf{ct}'_j, \mathsf{st}') \leftarrow \mathsf{ReEncSim}(\mathsf{AUX}, \mathsf{st})$: On input auxiliary information $\mathsf{AUX} \coloneqq (\mathsf{pp}, \mathsf{pk}_i, \mathsf{pk}_j, \mathsf{ct}_i, \mathsf{h})$ and private state st, it outputs a simulated re-encrypted ciphertext ct'_j, together with updated state st'. Here AUX contains the public parameters pp, the public keys $\mathsf{pk}_i, \mathsf{pk}_j$, a fresh ciphertext ct_i, and the corresponding hint h.

The simulator initializes and maintains its private state st using only its auxiliary inputs and its own randomness. The message m, the delegator's and delegatee's secret keys, and the corresponding rekey $\mathsf{rk}_{i\to j}$ are not supplied as separate inputs.

Simulatability. A PRE scheme $\widehat{\Pi}$ is ϵ_{sim}-simulatable if there exists a PPT stateful simulator $\mathsf{ReEncSim}$ satisfying the following for any polynomial-time sequence of queries within budgets Q, M. All user keys are honestly generated, with corrupted users fixed before key generation. The distinguisher receives corrupted users' secret keys and may obtain rekeys except those from honest delegators to corrupted delegatees. Consider re-encryption requests on source ciphertexts returned by $\mathcal{O}_{\mathsf{Enc}}$ under honest delegators' public keys, with corrupted delegatees. Their corresponding hints may be obtained through $\mathcal{O}_{\mathsf{Hint}}$ in Section 3.1. We require

$$(\mathsf{AUX}, \mathsf{ct}')_{\mathrm{real}} \approx_{\epsilon_{\mathrm{sim}}} (\mathsf{AUX}, \mathsf{ct}')_{\mathrm{sim}},$$

where real replies are generated by ReEnc, and simulated replies are generated by successive calls to $\mathsf{ReEncSim}(\mathsf{AUX}, \mathsf{st})$ starting from the first such request, and $\approx_{\epsilon_{\mathrm{sim}}}$ denotes statistical distance at most ϵ_{sim}. Here AUX contains the public parameters, the delegator's and delegatee's public keys, the source ciphertext, and the corresponding hint. All other services remain unchanged.

We note that this statistical indistinguishability must hold over the entire query sequence, including all replies and permitted observations, with distance at most ϵ_{sim}. Rekeys and private simulator state are retained across calls.

The proof of the following theorem is provided in Appendix C.

Theorem 3.3 (HRA security from hint CPA security)**.** *If a single-hop PRE scheme $\widehat{\Pi}$ is (Q, M)-hint CPA secure and ϵ_{sim}-simulatable for negligible ϵ_{sim}, then $\widehat{\Pi}$ is HRA secure.*

More precisely, for every PPT HRA adversary $\mathcal{A}$ there exists a PPT (Q, M)-hint CPA adversary $\mathcal{B}$ such that

$$\mathsf{Adv}^{\mathsf{HRA}}_{\widehat{\Pi},\mathcal{A}} \le \mathsf{Adv}^{\mathsf{hintCPA}}_{\widehat{\Pi},\mathcal{B}} + \epsilon_{\mathrm{sim}}.$$

We note that our framework also captures Cohen's CPA-to-HRA approach [11] through the variant that sets $\mathsf{HintGen} \equiv \perp$ and additionally gives the simulator the source message m and the corrupted delegatee's secret key sk_j. In this case, (Q, M)-hint CPA security reduces to CPA security under the same budgets.

Remark 3.4 (Extension to the multi-hop setting)**.** Although our generic framework is currently formulated for single-hop PRE, the same approach could extend naturally to bounded multi-hop settings with suitable hint and simulation guarantees. An extended $\mathsf{HintGen}$ would provide hints not only for source ciphertexts produced by Enc, but also for re-encrypted ciphertexts, including simulated outputs of $\mathsf{ReEncSim}$. An extended $\mathsf{ReEncSim}$ would generate intermediate ciphertexts and retain their associated hints in its state to simulate subsequent re-encryptions. Hint CPA security and simulatability would need to cover these evolving hints and ciphertexts, and the reduction must process simulated outputs using only permitted information and oracle access. We note that Proton Mail's encrypted email auto-forwarding requires only single-hop PRE.

4. Refining the Concrete Analysis of Matrix Hint-MLWE

We refine the reduction from standard MLWE to Matrix Hint-MLWE by allowing the spectral bound to fail with negligible probability. We then derive tighter spectral bounds for honestly sampled leakage matrices, reducing the Gaussian masking widths required by our PRE instantiation.

4.1. Refined Matrix Hint-MLWE

We first recall the definition of Matrix Hint-MLWE [20], [21] as follows.

Definition 4.1 (Matrix Hint-MLWE)**.** Let k, ℓ, n_{h} be positive integers, let $q \ge 2$ be an integer, and put $p = k + \ell$. Let $\mathcal{S}_L$, χ_0, and χ_1 be efficiently sampleable distributions over $\mathcal{R}^{n_{\mathrm{h}} \times p}$, $\mathcal{R}^p$, and $\mathcal{R}^{n_{\mathrm{h}}}$, respectively. Independently sample $\mathbf{A} \leftarrow \mathcal{U}(\mathcal{R}_q^{k\times\ell})$, $\mathbf{L} \leftarrow \mathcal{S}_L$, $\mathbf{z} = (\mathbf{e}^\top, \mathbf{s}^\top)^\top \leftarrow \chi_0$, $\mathbf{y} \leftarrow \chi_1$, and $\mathbf{u} \leftarrow \mathcal{U}(\mathcal{R}_q^\ell)$, where $\mathbf{e} \in \mathcal{R}^\ell$ and $\mathbf{s} \in \mathcal{R}^k$. Set $\mathbf{b} = \mathbf{A}^\top\mathbf{s} + \mathbf{e} \bmod q$ and $\mathbf{h}^{\mathrm{lin}} = \mathbf{L}\mathbf{z} + \mathbf{y}$. The problem $\mathsf{MatrixHint\text{-}MLWE}^{n_{\mathrm{h}},\chi_1,\mathcal{S}_L}_{\mathcal{R},k,\ell,q,\chi_0}$ is to distinguish $(\mathbf{A}, \mathbf{b}, \mathbf{L}, \mathbf{h}^{\mathrm{lin}})$ from $(\mathbf{A}, \mathbf{u}, \mathbf{L}, \mathbf{h}^{\mathrm{lin}})$. For $\mathsf{par} \coloneqq ((\mathcal{R}, k, \ell, q, \chi_0), (n_{\mathrm{h}}, \chi_1, \mathcal{S}_L))$, we say this problem is hard if, for every PPT adversary $\mathcal{A}$, the following advantage is negligible in λ:

$$\mathsf{Adv}^{\mathsf{MH}}_{\mathsf{par},\mathcal{A}} \coloneqq \big|\Pr[\mathcal{A}(\mathbf{A}, \mathbf{b}, \mathbf{L}, \mathbf{h}^{\mathrm{lin}}) = 1] - \Pr[\mathcal{A}(\mathbf{A}, \mathbf{u}, \mathbf{L}, \mathbf{h}^{\mathrm{lin}}) = 1]\big|.$$

Here, following [20], [21], [30], we establish the hardness of Matrix Hint-MLWE from standard MLWE, additionally allowing the spectral bound to fail with probability at most δ_{spec}.

Theorem 4.2 (Hardness of Matrix Hint-MLWE)**.** *In Definition 4.1, write $D = pd$ and $\Gamma(\mathbf{L}) \in \mathbb{Z}^{n_{\mathrm{h}}d\times D}$. Let $\sigma_{\mathrm{S}}, \sigma_y, \sigma_{\mathrm{H}} > 0$ and define $\chi_0 \coloneqq \mathcal{D}^p_{\sigma_{\mathrm{S}}}$, $\chi_1 \coloneqq \mathcal{D}^{n_{\mathrm{h}}}_{\sigma_y}$, and $\chi_{\mathrm{H}} \coloneqq \mathcal{D}^p_{\sigma_{\mathrm{H}}}$. Suppose $B \ge 0$ and $\delta_{\mathrm{spec}} \in [0, 1]$ satisfy*

$$\Pr_{\mathbf{L}\leftarrow\mathcal{S}_L}[\|\Gamma(\mathbf{L})\|^2_{\mathrm{op}} > B] \le \delta_{\mathrm{spec}}.$$

For $\epsilon_{\mathrm{g}} \in (0, 1/2)$ and $\delta > 0$, suppose the sampleability and convolution conditions

$$\frac{1}{(1+\delta)\sigma_{\mathrm{H}}^2 + \delta_0} \ge \frac{1}{\sigma_{\mathrm{S}}^2} + \frac{B}{\sigma_y^2}, \quad \sigma_{\mathrm{H}} \ge \sqrt{1 + 1/\delta}\,\eta_{\epsilon_{\mathrm{g}}}(\mathbb{Z}^D) \tag{2}$$

hold, where $\delta_0 \coloneqq \ln(2D + 4)/\pi$ is the sampleability offset and $\eta_{\epsilon_{\mathrm{g}}}(\mathbb{Z}^D)$ is the smoothing parameter. Then, for every PPT adversary $\mathcal{A}$ against $\mathsf{MatrixHint\text{-}MLWE}^{n_{\mathrm{h}},\chi_1,\mathcal{S}_L}_{\mathcal{R},k,\ell,q,\chi_0}$, there exists a PPT adversary $\mathcal{B}$ against $\mathsf{MLWE}_{\mathcal{R},k,\ell,q,\chi_{\mathrm{H}}}$ such that

$$\mathsf{Adv}^{\mathsf{MH}}_{\mathsf{par}_0,\mathcal{A}} \le \mathsf{Adv}^{\mathsf{MLWE}}_{\mathsf{par}_1,\mathcal{B}} + 2\epsilon_{\mathrm{g}} + \delta_{\mathrm{spec}}.$$

where $\mathsf{par}_0 \coloneqq ((\mathcal{R}, k, \ell, q, \chi_0), (n_{\mathrm{h}}, \chi_1, \mathcal{S}_L))$ and $\mathsf{par}_1 \coloneqq (\mathcal{R}, k, \ell, q, \chi_{\mathrm{H}})$.

A proof sketch is provided in Appendix B. The following corollary gives typical choices of Gaussian widths for Matrix Hint-MLWE.

Corollary 4.3. *In the setting of Theorem 4.2, let $\epsilon_{\mathrm{g}}, \delta, \sigma_{\mathrm{H}}$ satisfy its parameter bounds and convolution condition. Choose $\sigma_{\mathrm{S}}, \sigma_y > 0$ such that*

$$\sigma_{\mathrm{S}} > \sqrt{(1+\delta)\sigma_{\mathrm{H}}^2 + \delta_0},\ \sigma_y \ge \sqrt{\frac{B}{((1+\delta)\sigma_{\mathrm{H}}^2 + \delta_0)^{-1} - \sigma_{\mathrm{S}}^{-2}}}.$$

These choices satisfy the sampleability condition in (2)*, so the reduction applies with advantage loss at most $2\epsilon_{\mathrm{g}} + \delta_{\mathrm{spec}}$.*

Here, B bounds $\|\Gamma(\mathbf{L})\|^2_{\mathrm{op}}$, the squared spectral norm of the coefficient matrix. With all other parameters fixed, the required masking width in Corollary 4.3 scales as $\sqrt{B}$. Therefore, tightening B allows smaller masks and can thereby reduce the modulus and communication costs.

4.2. Tightening the Spectral Bound B

Let $\mathbf{L} \in \mathcal{R}^{Q\times p}$, where Q is the number of rows and $p = k + \ell$. Matrix Hint-MLWE [20], [21] is typically instantiated with short public leakage matrices $\mathbf{L}$; for example, each polynomial entry a satisfies $\|a\|_\infty \le \beta$ for some bound $\beta > 0$. In [21], the bound B is derived using $\|\Gamma(\mathbf{L})\|_{\mathrm{op}} \le \sqrt{\|\Gamma(\mathbf{L})\|_1 \|\Gamma(\mathbf{L})\|_\infty}$, where $\|\cdot\|_1$ and $\|\cdot\|_\infty$ denote the maximum absolute column and row sums, respectively. The coefficient bound gives $\|\Gamma(\mathbf{L})\|_1 \le Qd\beta$ and $\|\Gamma(\mathbf{L})\|_\infty \le pd\beta$. Thus $\|\Gamma(\mathbf{L})\|^2_{\mathrm{op}} \le B^{\mathrm{wc}}_Q$, where

$$B^{\mathrm{wc}}_Q \coloneqq Qpd^2\beta^2. \tag{3}$$

Here we focus on matrices honestly sampled from specified distributions with independent, mean-zero coefficients, and exploit their randomness to derive substantially tighter high-probability bounds.

A unified MGF bound. Let ξ denote any integer coefficient of a polynomial entry of $\mathbf{L}$. Let its MGF be $\mathbb{E}[e^{t\xi}]$. We use a common subgaussian variance proxy $\tau^2 > 0$ satisfying

$$\mathbb{E}[e^{t\xi}] \le e^{\tau^2 t^2/2} \qquad \text{for every } t \in \mathbb{R}. \tag{4}$$

This bound summarizes the coefficient distributions through τ^2 for the subsequent spectral tail analysis. We list valid choices of τ^2 for commonly used coefficient distributions in Table 3.

TABLE 3. SUBGAUSSIAN VARIANCE PROXIES.

Family	Distribution	τ^2
Binomial	CBD_η	$\eta/2$
Uniform	$\mathcal{U}(\{-\beta,\dots,\beta\})$	$\beta(\beta+1)/3$
Bounded	$\lvert\xi\rvert \le \beta$	β^2
Gaussian	$\mathcal{D}_{\mathbb{Z},\sigma}$	$\sigma^2/(2\pi)$

Centered binomial. By the Bernoulli-sum definition [12], $\xi \sim \mathsf{CBD}_\eta$ satisfies $\mathbb{E}[e^{t\xi}] = \cosh(t/2)^{2\eta} \le e^{\eta t^2/4}$, using $\cosh x \le e^{x^2/2}$. Hence $\tau^2 = \eta/2$.
Bounded uniform. For an integer $\beta \ge 1$ and $\xi \sim \mathcal{U}(\{-\beta,\dots,\beta\})$, $\mathbb{E}[e^{t\xi}] = \frac{\sinh((2\beta+1)t/2)}{(2\beta+1)\sinh(t/2)}$, interpreted by continuity at $t = 0$. Euler's product for $\sinh$[5] gives $\ln \mathbb{E}[e^{t\xi}] \le \beta(\beta+1)t^2/6$, so $\tau^2 = \beta(\beta+1)/3$. A proof is provided in Appendix B.
Bounded distributions. If $\mathbb{E}\xi = 0$ and $\lvert\xi\rvert \le \beta$, Hoeffding's lemma [31] gives $\mathbb{E}[e^{t\xi}] \le e^{\beta^2 t^2/2}$, so $\tau^2 = \beta^2$.
Discrete Gaussian. For $\xi \sim \mathcal{D}_{\mathbb{Z},\sigma}$ with $\sigma > 0$, the standard discrete-Gaussian MGF bound [32] gives $\mathbb{E}[e^{t\xi}] \le e^{\sigma^2 t^2/(4\pi)}$. Hence $\tau^2 = \sigma^2/(2\pi)$.

Fourier reduction. Here, we apply Fourier decomposition to reduce the spectral analysis of $\Gamma(\mathbf{L})$ to that of the smaller matrices $\mathbf{L}(\zeta_j)$, as formalized below. The proofs of Lemmas 4.4 and 4.5 are provided in Appendix B.

Lemma 4.4 (Fourier decomposition)**.** *For $\zeta_j = \exp((2j+1)\pi \mathrm{i}/d)$,*

$$\|\Gamma(\mathbf{L})\|_{\mathrm{op}} = \max_{0 \le j < d} \|\mathbf{L}(\zeta_j)\|_{\mathrm{op}}. \tag{5}$$

Since the coefficients are real, conjugate frequency blocks have equal operator norms, so it suffices to consider one frequency from each of the $d/2$ conjugate pairs.

We then combine the coefficient MGF bound (4) with a Gaussian integral to obtain the following exponential-moment bound for each fixed frequency ζ and unit vector x.

Lemma 4.5 (Fixed-direction exponential moment)**.** *Let $\mathbf{L} \in \mathcal{R}^{Q\times p}$ have mutually independent coefficients satisfying* (4) *with $\tau > 0$. For every unit vector $x \in \mathbb{C}^p$, every root ζ of $X^d + 1$, and $0 < \theta < (d\tau^2)^{-1}$,*

$$\mathbb{E}_{\mathbf{L}} e^{\theta \|\mathbf{L}(\zeta)x\|_2^2} \le (1 - d\tau^2\theta)^{-Q}.$$

Spherical averaging. Since the fixed-direction bound holds for every unit vector, we use spherical averaging and Markov's inequality to obtain a spectral tail bound. For a uniform unit vector $x \in \mathbb{C}^p$ with $p \ge 2$, $U = |x_1|^2$ follows $\mathrm{Beta}(1, p-1)$ [33].

Its MGF $\Phi_p(t) = \mathbb{E}[e^{tU}]$ satisfies, for $t > 0$,

$$\Phi_p(t) = (p-1)\int_0^1 e^{tu}(1-u)^{p-2}\,du = \frac{(p-1)!}{t^{p-1}}\left(e^t - \sum_{j=0}^{p-2} t^j/j!\right).$$

Set $\Phi_p(0) = 1$ and $\Phi_1(t) = e^t$.

Combining Lemma 4.5 with spherical averaging yields the following spectral tail bound.

Theorem 4.6 (Spherical-averaging spectral bound)**.** *Let $\mathbf{L} \in \mathcal{R}^{Q\times p}$ have mutually independent centered coefficients satisfying* (4) *with $\tau > 0$. For every $B > 0$,*

$$\Pr[\|\Gamma(\mathbf{L})\|_{\mathrm{op}}^2 \ge B] \le \frac{d}{2} \inf_{0<\theta<(d\tau^2)^{-1}} \frac{(1-d\tau^2\theta)^{-Q}}{\Phi_p(\theta B)}.$$

Proof. Fix $\zeta^d = -1$ and $0 < \theta < (d\tau^2)^{-1}$, and put $H = \mathbf{L}(\zeta)^*\mathbf{L}(\zeta) \succeq 0$. Let x be an independent uniform unit vector in $\mathbb{C}^p$ and define $\mathcal{I}_\theta(H) = \mathbb{E}_x[e^{\theta x^* H x}]$. For each fixed H with $\lambda_{\max}(H) \ge B$, unitary invariance of x and $H \succeq 0$ give

$$\mathcal{I}_\theta(H) \ge \mathbb{E}_x e^{\theta B |x_1|^2} = \Phi_p(\theta B).$$

The identity also holds for $p = 1$, since $|x_1|^2 = 1$. Tonelli's theorem [34] and Lemma 4.5 give

$$\mathbb{E}_{\mathbf{L}}[\mathcal{I}_\theta(H)] = \mathbb{E}_x \mathbb{E}_{\mathbf{L}} e^{\theta\|\mathbf{L}(\zeta)x\|_2^2} \le (1-d\tau^2\theta)^{-Q}.$$

Markov's inequality therefore bounds $\Pr[\lambda_{\max}(H) \ge B]$ by $(1-d\tau^2\theta)^{-Q}/\Phi_p(\theta B)$. By Lemma 4.4, a union bound over the $d/2$ conjugate frequency pairs, followed by the infimum over θ, proves the claim. □

To select B with spectral failure probability at most δ_{spec}, as required by Theorem 4.2, we substitute $r = d\tau^2\theta$ in Theorem 4.6. This gives the following verification condition and one-dimensional optimization. The proofs of Corollaries 4.7 and 4.8 are provided in Appendix B.

Corollary 4.7 (Computable spectral bound)**.** *In the setting of Theorem 4.6, fix $\delta_{\mathrm{spec}} \in (0,1)$, and put $c = d\tau^2$, $\Psi_p(t) = \ln \Phi_p(t)$, and $\alpha = \ln(d/(2\delta_{\mathrm{spec}}))$. If $B > 0$ and $r \in (0,1)$ satisfy*

$$\Psi_p(rB/c) \ge \alpha - Q\ln(1-r) \tag{6}$$

then $\Pr[\|\Gamma(\mathbf{L})\|_{\mathrm{op}}^2 > B] \le \delta_{\mathrm{spec}}$. The infimum of such bounds is

$$B_Q^{\mathrm{sph},*} = c \inf_{0<r<1} \frac{\Psi_p^{-1}(\alpha - Q\ln(1-r))}{r}. \tag{7}$$

We write B_Q^{sph} for a chosen bound satisfying (6). At fixed Q, p, d and δ_{spec}, the optimized bound (7) scales linearly with τ^2.

The following corollary gives an explicit bound without numerical optimization.

5. https://dlmf.nist.gov/4.36.E1

Corollary 4.8 (Explicit bound)**.** *In the setting of Corollary 4.7, let* $Q > 1$, $\alpha = \ln(d/(2\delta_{\mathrm{spec}}))$, *and* $A_Q = \alpha + (p-1)\ln Q/2$. *Then the bound*

$$B_{\exp} = \frac{c}{1 - Q^{-1/2}} \left(Q + \sqrt{2QA_Q} + A_Q \right)$$

satisfies $\Pr[\|\Gamma(\mathbf{L})\|_{\mathrm{op}}^2 > B_{\exp}] \le \delta_{\mathrm{spec}}$.

For fixed p, d, τ and δ_{spec}, $A_Q = O(\ln Q)$ gives $B_{\exp} = c[Q + O(\sqrt{Q \ln Q})]$. Also, $\Phi_p(t) \le e^t$ and $-\ln(1-r) > r$ imply $B_Q^{\mathrm{sph},*} \ge cQ$. Together with $B_Q^{\mathrm{sph},*} \le B_{\exp}$, this yields

$$B_Q^{\mathrm{sph},*} = d\tau^2 Q(1 + o(1)) \qquad (Q \to \infty).$$

Compared with applying the worst-case bound (3) from [21] to the same honestly sampled matrices with coefficients bounded by β, our bound yields an asymptotic reduction factor of $pd\beta^2/\tau^2$ in B. Specifically, for CBD_η coefficients, $\beta = \eta$ and $\tau^2 = \eta/2$, giving asymptotic reduction factors of $2pd\eta$ in the squared-norm bound and $\sqrt{2pd\eta}$ in the sufficient Gaussian masking width, with the remaining Gaussian parameters fixed.

We also extend the argument to real matrices by averaging over the real unit sphere, as follows.

Remark 4.9 (Extension to real matrices)**.** Let $\mathbf{X} \in \mathbb{R}^{n \times m}$, $n, m \ge 1$, have mutually independent, centered entries satisfying the MGF bound (4) with $\tau > 0$. For $m \ge 2$, define $\Phi_m^{\mathbb{R}}(t) = \mathbb{E}[e^{tU}]$ for $U \sim \mathrm{Beta}(1/2, (m-1)/2)$, and set $\Phi_1^{\mathbb{R}}(t) = e^t$. For every $B > 0$,

$$\Pr[\|\mathbf{X}\|_{\mathrm{op}}^2 \ge B] \le \inf_{0 < \theta < (2\tau^2)^{-1}} \frac{(1 - 2\tau^2\theta)^{-n/2}}{\Phi_m^{\mathbb{R}}(\theta B)}.$$

The proof follows the Gaussian-integral and spherical-averaging arguments above, using $g \sim N(0, \mathbf{I}_n)$ and a uniform unit vector in $\mathbb{R}^m$. Transposing $\mathbf{X}$ preserves its operator norm and allows the smaller dimension to index the sphere. This extension can support tightening the real-matrix bound in [20], as shown in Table 2.

Matrix Hint-MLWE Instantiation for our PRE. Set $p = k + \ell$ and let M be a positive integer. Sample $\mathbf{r}_t \in \mathcal{R}^\ell$ and $\mathbf{e}_{u,t} \in \mathcal{R}^k$ for $t \in [Q]$ with mutually independent CBD_1 coefficients, and define

$$\mathbf{L} = \begin{pmatrix} \mathbf{r}_1^\top & -\mathbf{e}_{u,1}^\top \\ \vdots & \vdots \\ \mathbf{r}_Q^\top & -\mathbf{e}_{u,Q}^\top \end{pmatrix} \in \mathcal{R}^{Q \times p}, \quad \widetilde{\mathbf{L}} = (\mathbf{L}^\top, \ldots, \mathbf{L}^\top)^\top \in \mathcal{R}^{QM \times p}. \quad (8)$$

The final leakage matrix $\widetilde{\mathbf{L}}$ stacks M identical copies of $\mathbf{L}$, so $\|\Gamma(\widetilde{\mathbf{L}})\|_{\mathrm{op}}^2 = M\|\Gamma(\mathbf{L})\|_{\mathrm{op}}^2$. Applying Corollary 4.7 with $\tau^2 = 1/2$ therefore yields the bound $B_{Q,M}^{\mathrm{sph}} = MB_Q^{\mathrm{sph}}$. Table 2 quantifies the gains for our PRE over the bound in [21]. Our artifact also provides scripts verifying our spectral bounds at $\delta_{\mathrm{spec}} = 2^{-160}$.

TABLE 4. MAIN PARAMETERS AND NOTATION FOR LatPRE.

Notation	Description
Q, M	Encryption and outgoing-rekey budgets per key
d, q	Ring dimension and modulus
k, ℓ	Numbers of rows and columns of $\mathbf{A}$, respectively
$\sigma_{\mathrm{S}}, \sigma_{\mathrm{H}}$	Source Gaussian width and baseline MLWE width
σ_y	Gaussian masking width
χ, χ_{T}	Encryption randomness and terminal secret key distributions
$B_{\mathrm{g}}, L_{\mathrm{g}}, w$	Gadget base, digit length, and $w = kL_{\mathrm{g}}$ columns
t_v, t_u', t_v'	Compression bits for ciphertext components
ξ_{ij}	Independent 256-bit PRF seed in $\mathsf{rk}_{i \to j}$
$\mathbf{d}_{\mathsf{ct}}$	Gadget decomposition of the source header $\mathbf{u}$

5. LatPRE: A Lattice-Based Instantiation

We present LatPRE, a lattice-based (Q, M)-hint CPA-secure PRE scheme that achieves HRA security via Theorem 3.3. Our construction combines lattice key switching with Gaussian masks fixed for each ciphertext–rekey pair using independent rekey seeds. These masks hide the source decryption error, and the corresponding hints enable re-encryption simulation without the delegator's secret key. Our security analysis uses the refined Matrix Hint-MLWE reduction in Section 4, which establishes a reduction from standard MLWE with compact parameters.

Construction 5.1 ((Q, M)-hint CPA PRE from lattices)**.** Let λ be the security parameter, $d \ge 2$ a power of two, $q > 2$ odd, and $k, \ell, Q, M, \eta_{\mathrm{T}}$ positive integers. Here Q, M are the per-key encryption and outgoing-rekey budgets. Let $B_{\mathrm{g}} \ge 3$ be an odd gadget base, $L_{\mathrm{g}} = \lceil \log_{B_{\mathrm{g}}} q \rceil$, and $w = kL_{\mathrm{g}}$. Let $\sigma_{\mathrm{S}}, \sigma_{\mathrm{H}}, \sigma_y > 0$ be Gaussian widths and $1 \le t_v, t_u', t_v' < \lceil \log_2 q \rceil$ be integer compression parameters. Set $\chi = \mathsf{CBD}_1$ and $\chi_{\mathrm{T}} = \mathsf{CBD}_{\eta_{\mathrm{T}}}$. Let $\mathsf{PRF} : \{0,1\}^{256} \times \{0,1\}^* \to \{0,1\}^{t_{\mathrm{samp}}}$ be a pseudorandom function and $\mathsf{GSamp}_{\sigma_y} : \{0,1\}^{t_{\mathrm{samp}}} \to \mathcal{R}$ an efficient deterministic discrete Gaussian sampler targeting width σ_y. The sampling length t_{samp} is polynomially bounded and chosen for the required accuracy. The simulator uses an initially empty private table st, retaining one matrix $\mathbf{K}_{ij}^0$ in $\mathsf{st}[i, j]$ for each delegator–delegatee pair. For the message space $\mathcal{M} = \{0,1\}^d$, Figure 3 gives the construction, with notation summarized in Table 4.

Correctness. We show the correctness of Construction 5.1 as follows. We select parameters in Section 5.1 to achieve δ_{dec}-correctness with $\delta_{\mathrm{dec}} < 2^{-128}$. The proof is provided in Appendix D.

Theorem 5.2 (Correctness)**.** *For any fixed message* m, *let the random variables be generated as in Construction 5.1. Write* $\mathbf{E}_{ij} = (\mathbf{E}_{ij,u}^\top, \mathbf{e}_{ij,v})^\top$ *and* $\mathbf{e}_0 = (\mathbf{e}_{0,u}^\top, e_{0,v})^\top$. *Let* $\omega_v, \boldsymbol{\omega}_u', \omega_v'$ *be the centered errors from compressing and reconstructing* $v, \mathbf{u}', v'$, *respectively. Define the rekey and fresh phase errors by* $\boldsymbol{\epsilon}_{ij}^\top \coloneqq \widehat{\mathbf{e}}_j^\top \mathbf{R}_{ij} + \mathbf{e}_{ij,v}^\top - \widehat{\mathbf{s}}_j^\top \mathbf{E}_{ij,u}$ *and* $\epsilon_{j,0} \coloneqq \widehat{\mathbf{e}}_j^\top \mathbf{r}_0 + e_{0,v} - \widehat{\mathbf{s}}_j^\top \mathbf{e}_{0,u}$, *respectively, and the source and terminal decryption errors by* $\nu_{\mathrm{S}} \coloneqq e_v + \mathbf{e}_i^\top \mathbf{r} - \mathbf{s}_i^\top \mathbf{e}_u + \omega_v$ *and*

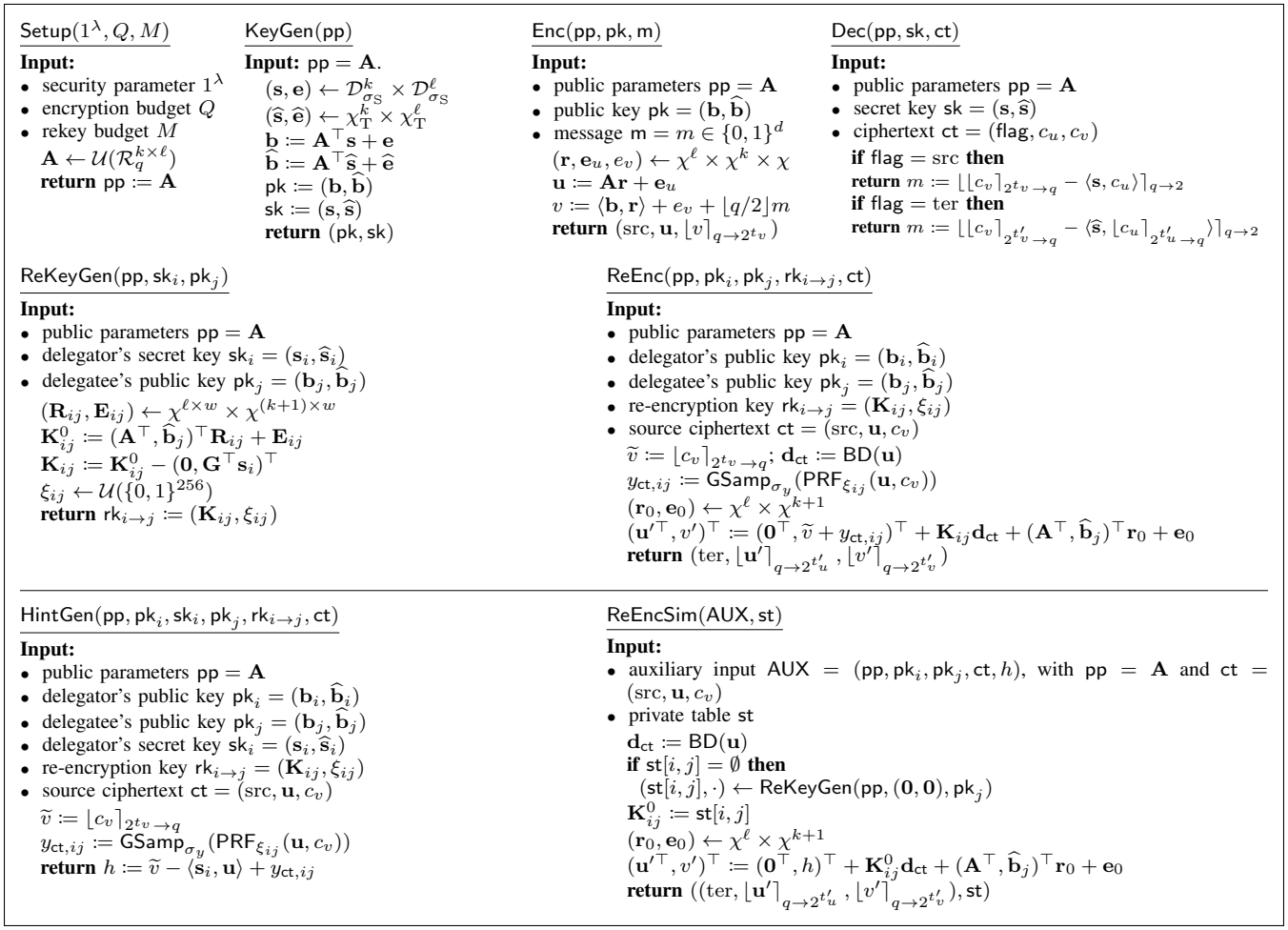

Figure 3. Construction of LatPRE, an HRA-secure PRE scheme, where HintGen and ReEncSim are used only in security proofs.

$\nu_{\mathrm{T}} := \nu_{\mathrm{S}} + y_{\mathsf{ct},ij} + \boldsymbol{\epsilon}_{ij}^{\top}\mathbf{d}_{\mathsf{ct}} + \epsilon_{j,0} + \boldsymbol{\omega}'_v - \widehat{\mathbf{s}}_j^{\top}\boldsymbol{\omega}'_u$. *Construction 5.1 is δ_{dec}-correct if, for every m,*

$$\Pr[\max\{\|\nu_{\mathrm{S}}\|_\infty, \|\nu_{\mathrm{T}}\|_\infty\} \geq \lfloor q/4 \rceil] \leq \delta_{\mathrm{dec}}. \tag{9}$$

Simulatability. We show that Construction 5.1 admits perfect hint-assisted re-encryption simulation, with $\epsilon_{\mathrm{sim}} = 0$. The proof is provided in Appendix D.

Theorem 5.3 (Simulatability)**.** *For any Q, M, Construction 5.1 is perfectly simulatable, with $\epsilon_{\mathrm{sim}} = 0$. Specifically, for every permitted query sequence within these budgets,*

$$(\mathbf{A}, \mathsf{pk}_i, \mathsf{pk}_j, \mathsf{ct}, h, \mathsf{ct}')_{\mathrm{real}} \approx_0 (\mathbf{A}, \mathsf{pk}_i, \mathsf{pk}_j, \mathsf{ct}, h, \mathsf{ct}')_{\mathrm{sim}},$$

where $\mathsf{pk}_i = (\mathbf{b}_i, \widehat{\mathbf{b}}_i)$, $\mathsf{pk}_j = (\mathbf{b}_j, \widehat{\mathbf{b}}_j)$, $\mathsf{ct} = (\mathrm{src}, \mathbf{u}, c_v)$, and $h = \widetilde{v} - \mathbf{s}_i^{\top}\mathbf{u} + y_{\mathsf{ct},ij}$. The re-encrypted ciphertext $\mathsf{ct}' = (\mathrm{ter}, \lfloor \mathbf{u}' \rceil_{q \to 2^{t'_u}}, \lfloor v' \rceil_{q \to 2^{t'_v}})$ is generated by ReEnc *in the real execution, or by* ReEncSim *in the simulated execution. Simulation starts from the first honest-to-corrupted request, and the above equation holds over all replies.*

Security. We establish (Q, M)-hint CPA security of our scheme under MLWE, Matrix Hint-MLWE, and PRF security. At a high level, MLWE hides honest terminal public keys, honest-to-honest rekeys, and the source challenge ciphertext. Matrix Hint-MLWE hides honest delegators' source public keys jointly with the hints exposed to corrupted delegatees, where PRF security and Gaussian sampling ensure that the masks are well-formed, i.e., computationally indistinguishable from ideal Gaussian samples. The proof is provided in Appendix D.

Theorem 5.4 ((Q, M)-hint CPA security)**.** *Let N, Q, M be positive integers, and let the remaining parameters and distributions be as in Construction 5.1. Our* LatPRE *is (Q, M)-hint CPA secure under PRF security and the hardness of* $\mathsf{MLWE}_{\mathcal{R},k,\ell,q,\chi_{\mathrm{T}}^p}$, $\mathsf{MLWE}_{\mathcal{R},\ell,k+1,q,\chi^{p+1}}$, *and* $\mathsf{MatrixHint\text{-}MLWE}^{Q_M,\mathcal{D}_{\sigma_y}^{Q_M},\mathcal{S}_{Q,M}}_{\mathcal{R},k,\ell,q,\mathcal{D}_{\sigma_{\mathrm{S}}}^p}$, *provided the statistical errors below are negligible. More precisely, for any PPT adversary $\mathcal{A}$ with N users and query budgets Q, M, there exist PPT adversaries $\mathcal{B}_{\mathrm{PRF}}, \mathcal{B}_{\mathrm{T}}, \mathcal{B}_{\mathrm{C}}, \mathcal{B}_{\mathrm{MH}}$ such that*

$$\begin{aligned}\mathsf{Adv}^{\mathsf{hintCPA}}_{\mathsf{LatPRE},\mathcal{A}} &\leq NM\,\mathsf{Adv}^{\mathsf{PRF}}_{\mathcal{B}_{\mathrm{PRF}}} + N\,\mathsf{Adv}^{\mathsf{MLWE}}_{\mathsf{par}_{\mathrm{T}},\mathcal{B}_{\mathrm{T}}} \\ &\quad + N\big((N-1)w+1\big)\mathsf{Adv}^{\mathsf{MLWE}}_{\mathsf{par}_{\mathrm{C}},\mathcal{B}_{\mathrm{C}}} + N\,\mathsf{Adv}^{\mathsf{MH}}_{\mathsf{par}_{\mathrm{MH}},\mathcal{B}_{\mathrm{MH}}} + \epsilon_{\mathrm{GS}} + \epsilon_{\mathrm{coll}}.\end{aligned}$$

where $p = k + \ell$, $\mathsf{par}_{\mathrm{T}} := (\mathcal{R}, k, \ell, q, \chi_{\mathrm{T}}^p)$, $\mathsf{par}_{\mathrm{C}} :=$

TABLE 5. PARAMETER SETS WITH $d = 256$, $\eta = 1$, AND $\sigma_{\mathrm{H}} = 8.74$. SEC. DENOTES THE CLASSICAL SECURITY TARGET IN BITS; DFR BOUNDS THE DECRYPTION FAILURE RATE. LIFETIME IS THE TIME TO EXHAUST Q AT THE ILLUSTRATIVE AVERAGE ENCRYPTION INTERVAL (ENC. INTERVAL).

Set	(Q, M)	$\lceil \log_2 q \rceil$	(k, ℓ)	η_{T}	σ_{S}	σ_y	$(B_{\mathrm{g}}, L_{\mathrm{g}}, w)$	(t_v, t'_u, t'_v)	Sec.	DFR	Enc. interval	Lifetime (years)
L	$(2^{20}, 2^{10})$	28	$(4, 5)$	12	37.0	4.9×10^{6}	$(643, 3, 12)$	$(7, 13, 5)$	128	2^{-128}	1 min	2.0
M	$(2^{30}, 2^{15})$	35	$(5, 6)$	16	61.7	8.5×10^{8}	$(3241, 3, 15)$	$(6, 13, 5)$	128	2^{-128}	0.1–1 s	3.4–34.0
H	$(2^{40}, 2^{20})$	42	$(6, 7)$	22	89.6	1.5×10^{11}	$(2086641, 2, 12)$	$(8, 14, 7)$	128	2^{-128}	0.1–1 ms	3.5–34.8

$(\mathcal{R}, \ell, k+1, q, \chi^{p+1})$, *and* $\mathsf{par}_{\mathrm{MH}} := ((\mathcal{R}, k, \ell, q, \mathcal{D}^{p}_{\sigma_{\mathrm{S}}}), (QM, \mathcal{D}^{QM}_{\sigma_y}, \mathcal{S}_{Q,M}))$. $\mathsf{Adv}^{\mathsf{PRF}}_{\mathcal{B}_{\mathrm{PRF}}}$ *denotes the PRF distinguishing advantage of* $\mathcal{B}_{\mathrm{PRF}}$. *Here* $\mathcal{S}_{Q,M}$ *is the distribution of* $\widetilde{\mathbf{L}}$ *defined in Equation* (8)*, and* $\epsilon_{\mathrm{GS}}, \epsilon_{\mathrm{coll}}$ *bound the cumulative Gaussian-sampling error and source-ciphertext collision probability, respectively.*

5.1. Parameter Setting

In this subsection, we discuss parameter selection for LatPRE in Construction 5.1. Write $p = k+\ell$, $\chi_0 = \mathcal{D}^{p}_{\sigma_{\mathrm{S}}}$, and $\chi_1 = \mathcal{D}^{QM}_{\sigma_y}$. As analyzed above, we require the following:

- $\mathsf{MLWE}_{\mathcal{R},k,\ell,q,\chi^{p}_{\mathrm{T}}}$ is hard at 128-bit classical security.
- $\mathsf{MLWE}_{\mathcal{R},\ell,k+1,q,\chi^{p+1}}$ is hard at 128-bit classical security.
- $\mathsf{MatrixHint\text{-}MLWE}^{QM,\chi_1,\mathcal{S}_{Q,M}}_{\mathcal{R},k,\ell,q,\chi_0}$ is hard at 128-bit classical security.
- δ_{dec}-correctness holds with $\delta_{\mathrm{dec}} < 2^{-128}$.

We estimate the practical hardness of MLWE using the lattice estimator[6] with the MATZOV classical reduction-cost model [35] under the geometric series assumption (GSA) [36]. For Matrix Hint-MLWE, we evaluate the underlying $\mathsf{MLWE}_{\mathcal{R},k,\ell,q,\mathcal{D}^{p}_{\sigma_{\mathrm{H}}}}$ instance through Theorem 4.2. Table 5 summarizes three parameter sets, Light (L), Middle (M), and Heavy (H), with different usage budgets to accommodate varying encryption rates and key lifetime requirements. Our artifact provides parameter-selection scripts and *additional parameter sets* with corresponding implementation results for reference.

We now present a step-by-step procedure for selecting the parameters.

First, we fix $d = 256$ to encrypt 256-bit messages, as in ML-KEM [12], and use $\chi = \mathsf{CBD}_{\eta}$ with $\eta = 1$ for source and re-encryption randomness. The terminal distribution is $\chi_{\mathrm{T}} = \mathsf{CBD}_{\eta_{\mathrm{T}}}$, with η_{T} selected jointly with the module ranks and modulus. We restrict q to primes satisfying $q \equiv 1 \pmod{2d}$, enabling a degree-d negacyclic NTT.

Second, for the Matrix Hint-MLWE parameters, we fix $\delta_{\mathrm{spec}} = 2^{-160}$, $\epsilon_{\mathrm{g}} = 2^{-150}$, and $\delta = 0.9$. For $D = pd$, the convolution condition in Theorem 4.2 holds whenever $\sigma_{\mathrm{H}} \geq \sqrt{(1+1/\delta)\ln(2D(1+1/\epsilon_{\mathrm{g}}))/\pi}$. We start from this sufficient lower bound and increase σ_{H} as needed to meet the intended security target. For $\tau^2 = 1/2$, Corollary 4.7 provides B^{sph}_{Q}. We set $B^{\mathrm{sph}}_{Q,M} = MB^{\mathrm{sph}}_{Q}$ and select $\sigma_{\mathrm{S}}, \sigma_y$ satisfying (2).

Third, we evaluate the source and re-encryption errors in Theorem 5.2 and require $\Pr[\max\{\|\nu_{\mathrm{S}}\|_\infty, \|\nu_{\mathrm{T}}\|_\infty\} \geq \lfloor q/4 \rceil] < 2^{-128}$. We use MGF bounds, Gaussian tail estimates, and CBD probability-mass-function (PMF) convolutions implemented in C, retaining the dependencies caused by the shared terminal secret and error.

Finally, we search jointly over $k, \ell, q, \eta_{\mathrm{T}}, \sigma_{\mathrm{S}}, \sigma_y$, the odd gadget base B_{g}, and the compression precisions (t_v, t'_u, t'_v). Our primary objective is to minimize the re-encrypted ciphertext size $|\mathsf{ct}'|$ while keeping the source ciphertext size $|\mathsf{ct}|$ and rekey size $|\mathsf{rk}|$ small. In particular, the gadget length $L_{\mathrm{g}} = \lceil \log_{B_{\mathrm{g}}} q \rceil$ determines the column count $w = kL_{\mathrm{g}}$. Increasing B_{g} reduces this count but increases rekey noise, so every selected candidate must pass the correctness check.

The theoretical ciphertext and key sizes follow directly from these parameters. Writing $t_q = \lceil \log_2 q \rceil$, their bit lengths are

$$|\mathsf{pk}| = 2\ell d t_q, \quad |\mathsf{sk}| = 2kdt_q, \quad |\mathsf{ct}| = d(kt_q + t_v),$$
$$|\mathsf{ct}'| = d(kt'_u + t'_v), \quad |\mathsf{rk}| = (k+1)wdt_q + 256.$$

The keys include both source and terminal components, and each rekey includes its independent 256-bit seed.

6. Implementation and Evaluation

We implement LatPRE in portable C[7] using the three parameter sets in Table 5. Polynomial arithmetic uses a degree-256 negacyclic NTT and Montgomery multiplication. We use domain-separated keyed SHAKE256 for the PRF, and expand the public matrix from a 32-byte seed. We adapt the discrete Gaussian samplers provided by OpenFHE [37].

As a baseline, we compare our implementation with the single-hop HRA-secure PRE [16][8]. Both implementations run single-threaded on a MacBook Air with an Apple M2 processor. We report median running times over 1,000 measurements.

Communication costs. Table 6 reports communication costs for 256-bit messages. Typically, re-encrypted ciphertext size is a primary concern in PRE because one source ciphertext may be re-encrypted for multiple delegatees. Our re-encrypted ciphertexts only occupy 1.8–2.8 KB, compared with 65.0 KB for [16], achieving a reduction of approximately 23–37×. In addition, source ciphertexts occupy 3.7–8.1 KB, compared with 65.0 KB for [16], representing an 8.0–17.5× reduction. These reductions mainly result from our use of a polynomial degree of 256 and much smaller moduli, compared with a degree of 4096 and a 54-bit ciphertext

6. https://github.com/malb/lattice-estimator

7. https://github.com/dingdangdangwhx/LatPRE-artifact

8. https://github.com/openfheorg/openfhe-network-examples

TABLE 6. COMMUNICATION COSTS (KB). S CT: SOURCE CIPHERTEXT; R CT: RE-ENCRYPTED CIPHERTEXT.

Scheme	PK	SK	Rekey	S CT	R CT	Reduction
LatPRE-L	8.8	7.0	52.5	3.7	**1.8**	**36.5×**
LatPRE-M	13.1	10.9	98.5	5.7	**2.2**	**29.7×**
LatPRE-H	18.4	15.8	110.3	8.1	**2.8**	**22.8×**
[16]	64.0	32.7	129.0	65.0	65.0	1.0×

TABLE 7. RUNNING TIMES (MS).

Scheme	Setup	KeyGen	ReKeyGen	Enc	Dec	ReEnc	Speedup
LatPRE-L	0.14	0.17	0.52	0.07	0.03	**0.24**	**9.1×**
LatPRE-M	0.21	0.23	0.78	0.08	0.04	**0.30**	**7.1×**
LatPRE-H	0.27	0.25	0.73	0.09	0.04	**0.32**	**6.7×**
[16]	1.09	0.62	0.68	0.54	0.07	2.15	1.0×

modulus (109-bit extended key modulus) in [16]; our bit-dropping compression further reduces ciphertext size.

Other communication costs include public keys, secret keys, and rekeys, which occupy 8.8–18.4, 7.0–15.8, and 52.5–110.3 KB, respectively, giving corresponding reductions of 3.5–7.3×, 2.1–4.7×, and 1.2–2.5×. These are one-time communication costs amortized over subsequent uses within the prescribed budgets.

Computational costs. Table 7 summarizes running times. All six algorithms have median running times below 1 ms. Specifically, re-encryption takes 0.24–0.32 ms, a 6.7–9.1× speedup over [16]. Encryption takes 0.07–0.09 ms, a 6.0–8.3× speedup, and decryption takes 0.03–0.04 ms. These speedups benefit from our smaller parameters, which reduce arithmetic costs. Setup, key generation, and rekey generation take 0.14–0.78 ms; these one-time computational costs are amortized over subsequent uses.

6.1. Application: Hybrid PQ Encrypted Email Auto-Forwarding for Proton Mail

We integrate LatPRE with Proton Mail's open-source classical forwarding mechanism [7] to build a hybrid post-quantum encrypted email auto-forwarding prototype. We evaluate its computational and communication costs against classical forwarding.

In this protocol, the delegator authorizes forwarding by generating the delegatee's key pairs for both schemes, securely delivering their secret keys to the delegatee, and providing the classical and LatPRE rekeys to the proxy. The sender encrypts two independent 256-bit key shares $x_{\mathrm{C}}, x_{\mathrm{P}}$ using Proton Mail's classical scheme and LatPRE, respectively, and derives the email encryption key $K = \mathsf{HKDF\text{-}SHA256}(x_{\mathrm{C}} \oplus x_{\mathrm{P}}; \mathsf{ctx})$. AES-256-GCM encrypts the email and its attachments with a fresh nonce ν and associated data ctx, which binds the protocol, suite, source public keys, and a fresh message identifier. The proxy re-encrypts the ciphertexts of x_{C} and x_{P} while preserving the encrypted email, nonce, and context, without requiring the delegator to remain online. The delegatee recovers x_{C} and x_{P}, derives the email encryption key K, and decrypts the email. We refer to Appendix A for the full protocol.

TABLE 8. EMAIL-FORWARDING RUNNING TIMES (MS).

Operation	Payload	Classical	Hybrid
KeyGen	—	0.12	0.47
Authorize	—	0.27	1.21
Encrypt	10 KB	0.16	0.41
	100 KB	0.17	0.43
	1 MB	0.33	0.65
Forward	10 KB	0.09	0.42
	100 KB	0.08	0.43
	1 MB	0.12	0.50
Decrypt	10 KB	0.09	0.28
	100 KB	0.10	0.30
	1 MB	0.34	0.47

TABLE 9. EMAIL-FORWARDING COMMUNICATION COSTS (KB).

Step / transmitted data	Classical	Hybrid
KeyGen: public keys	0.75	12.45
Authorize: full proxy package	1.05	71.11
Authorize: delegatee secret keys	1.14	10.52
Rekey in proxy package	0.03	52.56
Encrypt: per-email overhead	0.24	3.96
Forward: per-email overhead	0.24	2.02
Decrypt (local)	—	—

Evaluation. We use the Light parameters, $(Q, M) = (2^{20}, 2^{10})$, which allow each delegator to use the same key pair for approximately 2.87 years when receiving 1,000 emails per day and issue up to 1,024 auto-forwarding authorizations. After authorization, each rekey can be reused to forward subsequent emails to the same delegatee. We connect our C implementation to Proton Mail's `go-crypto` library[9], version `v1.5.2-proton`, through a Go adapter.

We compare classical and hybrid forwarding using identical email encryption and envelope formats, with payloads of 10 KB, 100 KB, and 1 MB. As above, we report medians over 1,000 single-threaded measurements on the Apple M2. Table 8 reports running times. For a 1 MB email, hybrid encryption, forwarding, and decryption take 0.65, 0.50, and 0.47 ms, respectively. KeyGen and Authorize take 0.47 and 1.21 ms. Setup is implicit and included in KeyGen.

Table 9 reports communication costs. For an email of P KB, encryption and forwarding in the classical scheme each transmit approximately $P + 0.24$ KB, while the hybrid scheme transmits approximately $P + 3.96$ KB and $P + 2.02$ KB, respectively. The hybrid rekey consists of the classical and LatPRE rekeys, totaling approximately 52.56 KB. The full proxy authorization package includes this rekey, the delegator's and delegatee's classical and LatPRE public keys, and metadata, totaling 71.11 KB.

9. https://github.com/ProtonMail/go-crypto/tree/v1.5.2-proton

References

[1] M. Blaze, G. Bleumer, and M. Strauss, "Divertible protocols and atomic proxy cryptography," in *Advances in Cryptology – EUROCRYPT '98*, 1998, pp. 127–144.

[2] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved proxy re-encryption schemes with applications to secure distributed storage," *ACM Transactions on Information and System Security*, vol. 9, no. 1, pp. 1–30, 2006.

[3] F. Hörandner, S. Ramacher, and S. Roth, "Selective end-to-end data-sharing in the cloud," *Journal of Banking and Financial Technology*, vol. 4, pp. 139–157, 2020.

[4] IronCore Labs, "IronCore's approach to cryptography," accessed: November 10, 2026. [Online]. Available: https://ironcorelabs.com/docs/data-control-platform/concepts/ironcore-cryptography/

[5] R. Canetti and S. Hohenberger, "Chosen-ciphertext secure proxy re-encryption," in *Proceedings of the 14th ACM Conference on Computer and Communications Security*, 2007.

[6] R. Bobba, J. Muggli, M. Pant, J. Basney, and H. Khurana, "Usable secure mailing lists with untrusted servers," in *Proc. Symp. Identity Trust Internet (IDtrust)*. ACM, 2009, pp. 103–116.

[7] F. Vial-Prado and A. Wussler, "OpenPGP email forwarding via diverted elliptic curve Diffie–Hellman key exchanges," arXiv:2211.06992, 2022. [Online]. Available: https://arxiv.org/abs/2211.06992

[8] F. Hörandner, S. Krenn, A. Migliavacca, F. Thiemer, and B. Zwattendorfer, "CREDENTIAL: A framework for privacy-preserving cloud-based data sharing," in *Proc. Int. Conf. Availability, Rel. Secur. (ARES)*. IEEE, 2016, pp. 742–749.

[9] Zyphe, "Data sharing and encryption," accessed: November 10, 2026. [Online]. Available: https://docs.zyphe.com/docs/decentralized-identity/data-sharing-and-encryption

[10] S. Gaddam, A. Luykx, R. Sinha, and G. Watson, "Reducing HSM reliance in payments through proxy re-encryption," in *30th USENIX Security Symposium*, 2021, pp. 4061–4078.

[11] A. Cohen, "What about bob? the inadequacy of CPA security for proxy reencryption," in *Proc. PKC*, 2019, pp. 287–316.

[12] National Institute of Standards and Technology, "Module-lattice-based key-encapsulation mechanism standard," National Institute of Standards and Technology, Federal Information Processing Standards Publication 203, 2024. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.203.pdf

[13] National Institute of Standards and Technology, "Module-lattice-based digital signature standard," National Institute of Standards and Technology, Federal Information Processing Standards Publication 204, 2024. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.204.pdf

[14] S. S. M. Chow, J. Weng, Y. Yang, and R. H. Deng, "Efficient unidirectional proxy re-encryption," in *Progress in Cryptology – AFRICACRYPT 2010*, 2010, pp. 316–332.

[15] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Journal on Computing*, vol. 26, no. 5, pp. 1484–1509, 1997.

[16] A. Cohen, D. B. Cousins, N. Genise, E. Kline, Y. Polyakov, and S. RV, "HRA-secure homomorphic lattice-based proxy re-encryption with tight security," *IACR Commun. Cryptol.*, vol. 2, no. 1, 2025.

[17] S. Sato and J. Shikata, "Bounded CCA2-secure proxy re-encryption from lattices," in *Selected Areas in Cryptography – SAC 2025*, 2026, pp. 425–455, preprint title: Bounded CCA2-Secure Proxy Re-Encryption Based on Kyber.

[18] H. Yin, J. Zhang, Y. Dong, D. Wojtczak, and E. G. Lim, "Lattice-based proxy re-encryption: Reconstruction attacks and HRA-secure constructions," Cryptology ePrint Archive, Paper 2025/704, 2025, revised November 10, 2026. [Online]. Available: https://eprint.iacr.org/2025/704

[19] H. Yin, J. Zhang, Y. Dong, D. Wojtczak, and E. G. Lim, "Single-hop HRA-secure owner-encrypted lattice-based proxy re-encryption without statistical noise flooding," Cryptology ePrint Archive, Paper 2026/1113, 2026. [Online]. Available: https://eprint.iacr.org/2026/1113

[20] T. Espitau, G. Niot, and T. Prest, "Flood and submerse: Distributed key generation and robust threshold signature from lattices," in *Advances in Cryptology – CRYPTO 2024*, 2024.

[21] H. Wang, R. Steinfeld, M.-J. O. Saarinen, M. F. Esgin, and S.-M. Yiu, "mmCipher: Batching post-quantum public key encryption made bandwidth-optimal," in *35th USENIX Security Symposium*, 2026, full version: Cryptology ePrint Archive 2025/1000. [Online]. Available: https://eprint.iacr.org/2025/1000

[22] N. Chandran, M. Chase, F.-H. Liu, R. Nishimaki, and K. Xagawa, "Re-encryption, functional re-encryption, and multi-hop re-encryption: A framework for achieving obfuscation-based security and instantiations from lattices," in *Proc. PKC*, 2014, pp. 95–112.

[23] Y. Zhou, S. Liu, S. Han, and H. Zhang, "Fine-grained proxy re-encryption: Definitions and constructions from LWE," in *Advances in Cryptology – ASIACRYPT 2023*, 2023.

[24] Y. Zhou, S. Liu, and S. Han, "Multi-hop fine-grained proxy re-encryption," in *Proc. PKC*, 2024, pp. 161–192.

[25] G. Fuchsbauer, C. Kamath, K. Klein, and K. Pietrzak, "Adaptively secure proxy re-encryption," in *Proc. PKC*, 2019, pp. 317–346.

[26] F. Zhao, H. Wang, and J. Weng, "Constant-size unbounded multi-hop fully homomorphic proxy re-encryption from lattices," in *Computer Security – ESORICS 2024*, 2024, pp. 238–258.

[27] X. Wan, Y. Wang, H. Xue, and M. Wang, "Unbounded multi-hop proxy re-encryption with HRA security: An LWE-based optimization," in *Information Security and Privacy – ACISP 2025*, 2025, pp. 124–144.

[28] H. Jo, S. Sato, and J. Shikata, "CCA2-secure unidirectional proxy re-encryption from cryptographic group actions," *IACR Commun. Cryptol.*, vol. 3, no. 2, 2026.

[29] D. Micciancio and O. Regev, "Worst-case to average-case reductions based on Gaussian measures," *SIAM Journal on Computing*, vol. 37, no. 1, pp. 267–302, 2007.

[30] D. Kim, D. Lee, J. Seo, and Y. Song, "Toward practical lattice-based proof of knowledge from Hint-MLWE," in *Advances in Cryptology – CRYPTO 2023*, 2023, pp. 549–580.

[31] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *J. Amer. Statist. Assoc.*, vol. 58, no. 301, pp. 13–30, 1963.

[32] D. Micciancio and C. Peikert, "Trapdoors for lattices: Simpler, tighter, faster, smaller," in *Advances in Cryptology – EUROCRYPT 2012*, 2012, pp. 700–718, full version: Cryptology ePrint Archive 2011/501.

[33] K. Życzkowski and H.-J. Sommers, "Induced measures in the space of mixed quantum states," *Journal of Physics A: Mathematical and General*, vol. 34, no. 35, pp. 7111–7125, 2001.

[34] S. Axler, *Measure, Integration & Real Analysis*, ser. Graduate Texts in Mathematics. Cham: Springer, 2020, vol. 282.

[35] MATZOV, "Report on the security of LWE: Improved dual lattice attack," MATZOV, Tech. Rep., 2022. [Online]. Available: https://zenodo.org/records/6412487

[36] C.-P. Schnorr, "Lattice reduction by random sampling and birthday methods," in *STACS 2003*, ser. Lecture Notes in Computer Science, vol. 2607. Springer, 2003, pp. 145–156.

[37] A. Al Badawi *et al.*, "OpenFHE: Open-source fully homomorphic encryption library," in *Proc. 10th Workshop Encrypted Comput. Appl. Homomorphic Cryptogr. (WAHC)*. ACM, 2022, pp. 53–63.

[38] D. Hsu, S. M. Kakade, and T. Zhang, "A tail inequality for quadratic forms of subgaussian random vectors," *Electronic Communications in Probability*, vol. 17, pp. 1–6, 2012, art. no. 52.

Appendix A.
Hybrid Email-Forwarding Protocol

$\underline{\mathsf{KeyGen}()}$

$(\mathsf{pk}_i^{\mathrm{C}}, \mathsf{sk}_i^{\mathrm{C}}) \leftarrow \mathsf{C.KeyGen}()$
$(\mathsf{pk}_i^{\mathrm{P}}, \mathsf{sk}_i^{\mathrm{P}}) \leftarrow \mathsf{P.KeyGen}()$
$P_i \coloneqq (\mathsf{pk}_i^{\mathrm{C}}, \mathsf{pk}_i^{\mathrm{P}})$
$S_i \coloneqq (\mathsf{sk}_i^{\mathrm{C}}, \mathsf{sk}_i^{\mathrm{P}})$
return (P_i, S_i)

$\underline{\mathsf{Authorize}(P_i, S_i)}$

$(\mathsf{pk}_j^{\mathrm{C}}, \mathsf{sk}_j^{\mathrm{C}}, \mathsf{rk}_{i\to j}^{\mathrm{C}}) \leftarrow \mathsf{C.Authorize}(\mathsf{sk}_i^{\mathrm{C}})$
$(\mathsf{pk}_j^{\mathrm{P}}, \mathsf{sk}_j^{\mathrm{P}}) \leftarrow \mathsf{P.KeyGen}()$
$\mathsf{rk}_{i\to j}^{\mathrm{P}} \leftarrow \mathsf{P.ReKeyGen}(\mathsf{sk}_i^{\mathrm{P}}, \mathsf{pk}_j^{\mathrm{P}})$
$P_j \coloneqq (\mathsf{pk}_j^{\mathrm{C}}, \mathsf{pk}_j^{\mathrm{P}})$
$S_j \coloneqq (\mathsf{sk}_j^{\mathrm{C}}, \mathsf{sk}_j^{\mathrm{P}})$
$R_{ij} \coloneqq (\mathsf{rk}_{i\to j}^{\mathrm{C}}, \mathsf{rk}_{i\to j}^{\mathrm{P}})$
return (P_j, S_j, R_{ij})

$\underline{\mathsf{Encrypt}(P_i, \mu)}$

$x_{\mathrm{C}}, x_{\mathrm{P}}, \iota \leftarrow \mathcal{U}(\{0,1\}^{256})$
$\nu \leftarrow \mathcal{U}(\{0,1\}^{96})$
$\mathsf{ctx} \coloneqq (\mathsf{domain}, \mathsf{suite}, H(\mathsf{pp}, P_i), \iota)$
$c_{\mathrm{C}} \leftarrow \mathsf{C.Enc}(\mathsf{pk}_i^{\mathrm{C}}, x_{\mathrm{C}})$
$c_{\mathrm{P}} \leftarrow \mathsf{P.Enc}(\mathsf{pk}_i^{\mathrm{P}}, x_{\mathrm{P}})$
$K \coloneqq \mathsf{KDF}(x_{\mathrm{C}} \oplus x_{\mathrm{P}}; \mathsf{ctx})$
$C \coloneqq \mathsf{AEAD.Enc}(K, \nu, \mu; \mathsf{ctx})$
return $E \coloneqq (\mathsf{ctx}, \nu, c_{\mathrm{C}}, c_{\mathrm{P}}, C)$

$\underline{\mathsf{Forward}(P_i, P_j, R_{ij}, E)}$

req: E is a source envelope for P_i
$(\mathsf{ctx}, \nu, c_{\mathrm{C}}, c_{\mathrm{P}}, C) \coloneqq E$
$c'_{\mathrm{C}} \coloneqq \mathsf{C.Transform}(\mathsf{rk}_{i\to j}^{\mathrm{C}}, c_{\mathrm{C}})$
$c'_{\mathrm{P}} \leftarrow \mathsf{P.ReEnc}(\mathsf{pk}_i^{\mathrm{P}}, \mathsf{pk}_j^{\mathrm{P}}, \mathsf{rk}_{i\to j}^{\mathrm{P}}, c_{\mathrm{P}})$
return $E' \coloneqq (\mathsf{ctx}, \nu, c'_{\mathrm{C}}, c'_{\mathrm{P}}, C)$

$\underline{\mathsf{Decrypt}(S_j, E')}$

$(\mathsf{ctx}, \nu, c'_{\mathrm{C}}, c'_{\mathrm{P}}, C) \coloneqq E'$
$x_{\mathrm{C}} \coloneqq \mathsf{C.Dec}(\mathsf{sk}_j^{\mathrm{C}}, c'_{\mathrm{C}})$
$x_{\mathrm{P}} \coloneqq \mathsf{P.Dec}(\mathsf{sk}_j^{\mathrm{P}}, c'_{\mathrm{P}})$
$K \coloneqq \mathsf{KDF}(x_{\mathrm{C}} \oplus x_{\mathrm{P}}; \mathsf{ctx})$
return $\mathsf{AEAD.Dec}(K, \nu, C; \mathsf{ctx})$

Figure 4. Hybrid post-quantum encrypted email forwarding. C is Proton Mail's classical scheme and P is LatPRE; Setup and the public parameters are implicit. μ is the email including attachments, H is SHA-256, KDF is HKDF-SHA256, and AEAD is AES-256-GCM. The delegator i securely delivers S_j to the delegatee j and provides (P_i, P_j, R_{ij}) to the proxy. Envelope metadata checks are omitted; any decryption failure returns $\perp$.

Appendix B.
Auxiliary Spectral Analyses

Proof sketch of Theorem 4.2. We adapt the discrete Gaussian reduction of [21, Theorem F.3]. Given an $\mathsf{MLWE}_{\mathcal{R},k,\ell,q,\chi_{\mathrm{H}}}$ input $(\mathbf{A}, \mathbf{b}_{\mathrm{H}})$, the reduction independently samples $\mathbf{L} \leftarrow \mathcal{S}_L$. Let G denote the event $\|\Gamma(\mathbf{L})\|_{\mathrm{op}}^2 \le B$; the reduction outputs a fixed bit if G fails. For every $\mathbf{L}$ satisfying G, the conditions in (2) permit the standard Gaussian reduction with advantage loss at most $2\epsilon_{\mathrm{g}}$. Both target distributions have the same $\mathbf{L}$ marginal, so the contribution of $\neg G$ to their distinguishing gap is at most $\Pr[\neg G] \le \delta_{\mathrm{spec}}$. Combining these losses proves the bound. □

MGF bound for bounded uniform coefficients. Let $\xi \sim \mathcal{U}(\{-\beta, \dots, \beta\})$ for an integer $\beta \ge 1$. For $t \ne 0$, put $c = 2\beta + 1$ and $a_n = t^2/(4\pi^2 n^2)$. Euler's product gives $\mathbb{E}[e^{t\xi}] = \frac{\sinh(ct/2)}{c\sinh(t/2)} = \prod_{n=1}^{\infty} \frac{1+c^2 a_n}{1+a_n}$. Using $\ln(1+u) \le u$ for $u \ge 0$ gives $\ln((1+c^2 a_n)/(1+a_n)) \le (c^2-1)a_n$. Hence

$$\ln \mathbb{E}[e^{t\xi}] \le (c^2-1) \sum_{n=1}^{\infty} a_n = \frac{(c^2-1)t^2}{24} = \frac{\beta(\beta+1)t^2}{6}.$$

Here we used $\sum_{n=1}^{\infty} n^{-2} = \pi^2/6$. The case $t = 0$ follows by continuity, proving the MGF bound with $\tau^2 = \beta(\beta+1)/3$. □

Proof of Lemma 4.4. The unitary Fourier matrix $F = (d^{-1/2}\zeta_j^h)_{0\le j,h<d}$ diagonalizes negacyclic convolution. Transforming each input and output block of $\Gamma(\mathbf{L})$ and regrouping by frequency gives a block diagonal matrix with blocks $\mathbf{L}(\zeta_j)$. Unitary invariance of the operator norm proves (5). □

Proof of Lemma 4.5. Fix x, ζ as above and put $y = \mathbf{L}(\zeta)x$. Since $d \ge 2$ is even, $\zeta^2 \ne 1$ and $\sum_{h=0}^{d-1} \zeta^{2h} = 0$. Hence $\sum_{h=0}^{d-1} \mathrm{Re}(c\zeta^h)^2 = d|c|^2/2$ for every $c \in \mathbb{C}$. Coefficient independence and (4) give, for any $v \in \mathbb{C}^Q$,

$$\mathbb{E}e^{\mathrm{Re}(v^* y)} \le \exp\left(\frac{\tau^2}{2} \sum_{t=1}^{Q} \sum_{a=1}^{p} \sum_{h=0}^{d-1} \mathrm{Re}(\bar{v}_t x_a \zeta^h)^2\right) = e^{d\tau^2 \|v\|_2^2/4}.$$

The inequality follows by factoring the expectation over independent coefficients and applying (4) to each factor; the last equality uses the preceding identity with $c = \bar{v}_t x_a$ and $\sum_{a=1}^{p} |x_a|^2 = 1$.

Write $y_{\mathbb{R}} = (\mathrm{Re}\, y, \mathrm{Im}\, y) \in \mathbb{R}^{2Q}$ and independently sample $g \leftarrow N(0, \mathbf{I}_{2Q})$. The Gaussian identity $e^{\theta\|y\|_2^2} = \mathbb{E}_g e^{\sqrt{2\theta}\langle g, y_{\mathbb{R}}\rangle}$ and Tonelli's theorem [34] give

$$\mathbb{E}_y e^{\theta\|y\|_2^2} \le \mathbb{E}_g e^{d\tau^2\theta\|g\|_2^2/2} = (1 - d\tau^2\theta)^{-Q}.$$

The last Gaussian integral is finite for $\theta < (d\tau^2)^{-1}$. This Gaussian-domination argument [38] uses the joint MGF bound above and requires no independence between the real and imaginary parts of y. □

Proof of Corollary 4.7. For each fixed $r = c\theta \in (0,1)$, the corresponding tail bound is at most δ_{spec} exactly when (6) holds, by taking logarithms. Since Ψ_p is continuous and strictly increasing from 0 to ∞, inversion followed by the infimum over r gives (7). □

Proof of Corollary 4.8. Put $\varepsilon = Q^{-1/2}$, $v = \sqrt{2A_Q/Q}$, and $r = v/(1+v)$. For $p \ge 2$, the Beta tail probability $\Pr[U \ge 1-\varepsilon] = \varepsilon^{p-1}$ gives $\Phi_p(t) \ge \varepsilon^{p-1} e^{(1-\varepsilon)t}$; this inequality also holds for $p = 1$. Using $-\ln(1-r) - r \le r^2/(2(1-r))$ gives

$$-Q\ln(1-r) - \frac{r(1-\varepsilon)B_{\exp}}{c} \le \frac{Qr^2}{2(1-r)} - r(\sqrt{2QA_Q} + A_Q) = -A_Q.$$

Thus the bound from Theorem 4.6 at $\theta = r/c$ is at most $(d/2)\varepsilon^{-(p-1)} e^{-A_Q} = \delta_{\mathrm{spec}}$. □

Appendix C.
Proof of Theorem 3.3

Oracle $\mathcal{O}_{\mathsf{ReEnc}}(i, j, a)$
  req: $\mathsf{CT}[a] \neq \emptyset$
  $(i_a, \mathsf{ct}, \mathsf{ch}) \leftarrow \mathsf{CT}[a]$
  req: $i_a = i$ and ($\mathsf{ch} = 0$ or $j \notin \mathsf{Cor}$)
  if $\mathsf{ct} = \perp$ or ct is terminal **then return** $\perp$
  if $\mathsf{RK}[i, j] = \emptyset$ **then**
    req: $c_i < M$
    $\mathsf{RK}[i, j] \leftarrow \mathsf{ReKeyGen}(\mathsf{pp}, \mathsf{sk}_i, \mathsf{pk}_j)$
    $c_i \leftarrow c_i + 1$
  end if
  if $i \notin \mathsf{Cor}$ and $j \in \mathsf{Cor}$ **then** ▷ G_1, G_2
    $\mathsf{h} \leftarrow \mathcal{O}_{\mathsf{Hint}}(i, j, a)$
  end if
  $\mathsf{ct}' \leftarrow \mathsf{ReEnc}(\mathsf{pp}, \mathsf{pk}_i, \mathsf{pk}_j, \mathsf{RK}[i, j], \mathsf{ct})$ ▷ G_0, G_1
  if $i \notin \mathsf{Cor}$ and $j \in \mathsf{Cor}$ **then** ▷ G_2
    $\mathsf{h} \leftarrow \mathsf{H}[i, j, a]$
    $\mathsf{AUX} \coloneqq (\mathsf{pp}, \mathsf{pk}_i, \mathsf{pk}_j, \mathsf{ct}, \mathsf{h})$
    $(\mathsf{ct}', \mathsf{st}) \leftarrow \mathsf{ReEncSim}(\mathsf{AUX}, \mathsf{st})$
  else
    $\mathsf{ct}' \leftarrow \mathsf{ReEnc}(\mathsf{pp}, \mathsf{pk}_i, \mathsf{pk}_j, \mathsf{RK}[i, j], \mathsf{ct})$
  end if
  $t \leftarrow t + 1$; $\mathsf{CT}[t] \leftarrow (j, \mathsf{ct}', \mathsf{ch})$
  return (t, ct')

Initially $\mathsf{H}, \mathsf{st} \leftarrow \emptyset$. Hint calls are internal; no hint is returned to $\mathcal{A}$.

Figure 5. Re-encryption oracle for G_0, G_1, G_2; other oracles follow Figure 1. Additions to Cohen's argument [11] are highlighted in boxes.

Proof. The proof is based on [11, Theorem 5]. Let $\mathcal{A}$ be a PPT adversary against HRA security in Figure 1. We bound its advantage through games G_0, G_1, G_2. Let E_i denote the event that $\mathcal{A}$ wins G_i. Figure 5 shows the oracle changes; all other steps follow Figure 1.

Game G_0**.** This is the real HRA experiment, so $\mathsf{Adv}^{\mathsf{HRA}}_{\widehat{\Pi},\mathcal{A}} = \left|\Pr[E_0] - \frac{1}{2}\right|$.

Game G_1**.** On an honest-to-corrupted re-encryption request, this game internally calls $\mathcal{O}_{\mathsf{Hint}}$ from Figure 2 to generate or retrieve the corresponding hint. These internal operations leave the adversary's view unchanged; hence $\Pr[E_1] = \Pr[E_0]$.

Game G_2**.** This game replaces honest-to-corrupted replies by successive $\mathsf{ReEncSim}$ calls using the source ciphertexts and corresponding hints, retaining the simulator state. By simulatability, replacing the real re-encryption replies with simulated ones changes the adversary's winning probability by at most ϵ_{sim}. Hence $|\Pr[E_1] - \Pr[E_2]| \leq \epsilon_{\mathrm{sim}}$.

Thus, there exists a (Q, M)-hint CPA adversary $\mathcal{B}$ that reproduces G_2 by using $\mathcal{O}_{\mathsf{Hint}}$ and $\mathsf{ReEncSim}$ for honest-to-corrupted replies and forwarding the remaining permitted queries to its challenger, giving $\left|\Pr[E_2] - \frac{1}{2}\right| = \mathsf{Adv}^{\mathsf{hintCPA}}_{\widehat{\Pi},\mathcal{B}}$. Combining the games yields the claimed bound. □

Appendix D.
Proofs for LatPRE

Correctness.

Proof of Theorem 5.2. All phase identities below are modulo q. For the source ciphertext, $\widetilde{v} = v + \omega_v$ and $\mathbf{b}_i = \mathbf{A}^\top \mathbf{s}_i + \mathbf{e}_i$ give

$$\begin{aligned}\widetilde{v} - \mathbf{s}_i^\top \mathbf{u} &= v + \omega_v - \mathbf{s}_i^\top(\mathbf{A}\mathbf{r} + \mathbf{e}_u) \\ &= \lfloor q/2 \rfloor m + e_v + \mathbf{e}_i^\top \mathbf{r} - \mathbf{s}_i^\top \mathbf{e}_u + \omega_v \\ &= \lfloor q/2 \rfloor m + \nu_{\mathrm{S}}.\end{aligned}$$

For the rekey, $\widehat{\mathbf{b}}_j^\top - \widehat{\mathbf{s}}_j^\top \mathbf{A} = \widehat{\mathbf{e}}_j^\top$ implies

$$\begin{aligned}(-\widehat{\mathbf{s}}_j^\top, 1)\mathbf{K}_{ij} &= -\mathbf{s}_i^\top \mathbf{G} + \widehat{\mathbf{e}}_j^\top \mathbf{R}_{ij} + \mathbf{e}_{ij,v}^\top - \widehat{\mathbf{s}}_j^\top \mathbf{E}_{ij,u} \\ &= -\mathbf{s}_i^\top \mathbf{G} + \boldsymbol{\epsilon}_{ij}^\top.\end{aligned}$$

Similarly, the fresh re-encryption term satisfies

$$(-\widehat{\mathbf{s}}_j^\top, 1)\left[(\mathbf{A}^\top, \widehat{\mathbf{b}}_j)^\top \mathbf{r}_0 + \mathbf{e}_0\right] = \epsilon_{j,0}.$$

Substituting these identities into ReEnc and using $\mathbf{G}\mathbf{d}_{\mathsf{ct}} = \mathbf{u}$, we obtain

$$\begin{aligned}v' - \widehat{\mathbf{s}}_j^\top \mathbf{u}' &= \widetilde{v} + y_{\mathsf{ct},ij} - \mathbf{s}_i^\top \mathbf{G}\mathbf{d}_{\mathsf{ct}} + \boldsymbol{\epsilon}_{ij}^\top \mathbf{d}_{\mathsf{ct}} + \epsilon_{j,0} \\ &= \widetilde{v} - \mathbf{s}_i^\top \mathbf{u} + y_{\mathsf{ct},ij} + \boldsymbol{\epsilon}_{ij}^\top \mathbf{d}_{\mathsf{ct}} + \epsilon_{j,0} \\ &= \lfloor q/2 \rfloor m + \nu_{\mathrm{S}} + y_{\mathsf{ct},ij} + \boldsymbol{\epsilon}_{ij}^\top \mathbf{d}_{\mathsf{ct}} + \epsilon_{j,0}.\end{aligned}$$

After reconstructing the terminal components, $\widetilde{\mathbf{u}}' = \mathbf{u}' + \boldsymbol{\omega}'_u$ and $\widetilde{v}' = v' + \omega'_v$, so

$$\begin{aligned}\widetilde{v}' - \widehat{\mathbf{s}}_j^\top \widetilde{\mathbf{u}}' &= v' - \widehat{\mathbf{s}}_j^\top \mathbf{u}' + \omega'_v - \widehat{\mathbf{s}}_j^\top \boldsymbol{\omega}'_u \\ &= \lfloor q/2 \rfloor m + \nu_{\mathrm{T}}.\end{aligned}$$

Since coefficientwise rounding recovers m when the noise norm is below $\lfloor q/4 \rfloor$, both decryptions succeed outside the event in (9), proving the claim. □

Simulatability.

Proof of Theorem 5.3. All identities below are modulo q. For each honest delegator i and corrupted delegatee j, the simulator samples $(\mathbf{R}^0_{ij}, \mathbf{E}^0_{ij}) \leftarrow \chi^{\ell \times w} \times \chi^{(k+1) \times w}$ through $\mathsf{ReKeyGen}(\mathsf{pp}, (\mathbf{0}, \mathbf{0}), \mathsf{pk}_j)$ and caches in $\mathsf{st}[i, j]$ the matrix

$$\mathbf{K}^0_{ij} \coloneqq (\mathbf{A}^\top, \widehat{\mathbf{b}}_j)^\top \mathbf{R}^0_{ij} + \mathbf{E}^0_{ij}. \tag{10}$$

For each call, it independently samples $(\mathbf{r}^0_0, \mathbf{e}^0_0) \leftarrow \chi^\ell \times \chi^{k+1}$ and sets $\mathbf{f}^0_0 = (\mathbf{A}^\top, \widehat{\mathbf{b}}_j)^\top \mathbf{r}^0_0 + \mathbf{e}^0_0$. Substituting (10) into $\mathsf{ReEncSim}$ gives

$$\begin{aligned}\left((\mathbf{u}')^\top, v'\right)^\top_{\mathrm{sim}} &= (\mathbf{0}^\top, h)^\top + \mathbf{K}^0_{ij}\mathbf{d}_{\mathsf{ct}} + \mathbf{f}^0_0 \\ &= (\mathbf{0}^\top, h)^\top + ((\mathbf{A}^\top, \widehat{\mathbf{b}}_j)^\top \mathbf{R}^0_{ij} + \mathbf{E}^0_{ij})\mathbf{d}_{\mathsf{ct}} + \mathbf{f}^0_0.\end{aligned}$$

In the real execution, $\mathsf{ReKeyGen}$ independently samples $(\mathbf{R}_{ij}, \mathbf{E}_{ij})$ from the same distribution and computes

$$\mathbf{K}_{ij} = (\mathbf{A}^\top, \widehat{\mathbf{b}}_j)^\top \mathbf{R}_{ij} + \mathbf{E}_{ij} - (\mathbf{0}, \mathbf{G}^\top \mathbf{s}_i)^\top.$$

With fresh $\mathbf{f}_0 = (\mathbf{A}^\top, \widehat{\mathbf{b}}_j)^\top \mathbf{r}_0 + \mathbf{e}_0$, where $(\mathbf{r}_0, \mathbf{e}_0) \leftarrow \chi^\ell \times \chi^{k+1}$, substitution into ReEnc yields

$$\begin{aligned}((\mathbf{u}')^\top, v')^\top_{\text{real}} &= (\mathbf{0}^\top, \widetilde{v} + y_{\mathsf{ct},ij})^\top + \mathbf{K}_{ij}\mathbf{d}_{\mathsf{ct}} + \mathbf{f}_0 \\ &= (\mathbf{0}^\top, h)^\top + ((\mathbf{A}^\top, \widehat{\mathbf{b}}_j)^\top \mathbf{R}_{ij} + \mathbf{E}_{ij})\mathbf{d}_{\mathsf{ct}} + \mathbf{f}_0,\end{aligned}$$

where the last equality uses $\mathbf{G}\mathbf{d}_{\mathsf{ct}} = \mathbf{u}$ and h.

Since the rekey is hidden and $\mathsf{HintGen}$ reads only its independent seed from the rekey, the randomness used to generate its rekey matrix is independent of observations made before the first re-encrypted ciphertext from delegator i to delegatee j is returned. Both executions sample this randomness from the same distribution and retain the resulting rekey. Thus, the distributions of re-encrypted ciphertexts and permitted observations coincide, and deterministic rounding preserves this equality, giving $\epsilon_{\text{sim}} = 0$. □

(Q, M)-Hint CPA Security.

Proof of Theorem 5.4. Let $\mathcal{A}$ be a PPT adversary against (Q, M)-hint CPA security of LatPRE. We bound its advantage through games $G_0, \ldots, G_6$. Let E_i denote the event that $\mathcal{A}$ wins G_i. The games are described in Figure 6; all other steps follow the security game in Section 3.1.

Game G_0. This is the real (Q, M)-hint CPA experiment, so $\mathsf{Adv}^{\mathsf{hintCPA}}_{\mathsf{LatPRE},\mathcal{A}} = |\Pr[E_0] - 1/2|$.

Game G_1. The game is the same as G_0 except that it aborts on source-ciphertext collisions and replaces the masks for honest-to-corrupted requests by ideal Gaussian samples.

Let Coll denote the event that two distinct source encryptions, including the challenge encryption, produce identical ciphertexts, i.e., $(\mathbf{u}_a, c_{v,a}) = (\mathbf{u}_{a'}, c_{v,a'})$ for $a \neq a'$. Under a fixed rekey, identical ciphertexts yield the same mask $y_{\mathsf{ct},ij}$. On Coll, the game returns an independent uniform bit. Let ϵ_{coll} denote the probability of Coll. A collision implies $\mathbf{u}_a = \mathbf{u}_{a'}$. Since $\chi = \mathsf{CBD}_1$, we obtain $\epsilon_{\text{coll}} \le \binom{NQ+1}{2} 2^{-kd}$.

For the at most NM hidden honest-to-corrupted rekey seeds, replace the PRFs by independent random functions, whose outputs on distinct ciphertexts are independent uniform strings $U \leftarrow \mathcal{U}(\{0,1\}^{t_{\text{samp}}})$. Then replace $\mathsf{GSamp}_{\sigma_y}(U)$ by ideal samples $y \leftarrow \mathcal{D}_{\sigma_y}$, at total statistical cost ϵ_{GS}. Repeated ciphertext–rekey pairs retain their masks. Hence $|\Pr[E_1] - \Pr[E_0]| \le NM\mathsf{Adv}^{\mathsf{PRF}}_{\mathcal{B}_{\text{PRF}}} + \epsilon_{\text{GS}} + \epsilon_{\text{coll}}$.

Game G_2. The game is the same as G_1 except that the challenger replaces each honest terminal public key $\widehat{\mathbf{b}}_j$ by an independent uniform vector $\mathbf{T}_j \leftarrow \mathcal{U}(\mathcal{R}_q^\ell)$. The honestly generated key satisfies $\widehat{\mathbf{b}}_j = \mathbf{A}^\top \widehat{\mathbf{s}}_j + \widehat{\mathbf{e}}_j$, where $\widehat{\mathbf{s}}_j \leftarrow \chi_{\text{T}}^k$ and $\widehat{\mathbf{e}}_j \leftarrow \chi_{\text{T}}^\ell$. Thus, each replacement is indistinguishable under the $\mathsf{MLWE}_{\mathcal{R},k,\ell,q,\chi_{\text{T}}^p}$ assumption. There are at most N honest users. Therefore, there exists a PPT adversary $\mathcal{B}_{\text{T}}$ such that $|\Pr[E_2] - \Pr[E_1]| \le N\mathsf{Adv}^{\mathsf{MLWE}}_{\mathsf{par}_{\text{T}},\mathcal{B}_{\text{T}}}$.

Game G_3. The game is the same as G_2 except that the challenger replaces each honest-to-honest rekey matrix $\mathbf{K}_{ij}$ by an independent uniform matrix $\mathbf{U}_{ij} \leftarrow \mathcal{U}(\mathcal{R}_q^{(k+1)\times w})$, leaving its seed unchanged. Set $\mathbf{P}_j \coloneqq (\mathbf{A}^\top, \mathbf{T}_j)^\top$. Each rekey column in G_2 has the form

$$\mathbf{K}_{ij,a} = \mathbf{P}_j \mathbf{r}_a + \mathbf{e}_a - (\mathbf{0}^\top, \mathbf{g}_a^\top \mathbf{s}_i)^\top,$$

where $\mathbf{r}_a \leftarrow \chi^\ell$, $\mathbf{e}_a \leftarrow \chi^{k+1}$, and $\mathbf{g}_a$ is the ath column of $\mathbf{G}$. Since $\mathbf{P}_j$ is uniform in G_2, the first two terms form a $\mathsf{MLWE}_{\mathcal{R},\ell,k+1,q,\chi^{p+1}}$ sample. The reduction knows $\mathbf{s}_i$, and translating a uniform sample by this known offset preserves uniformity. There are $N(N-1)w$ possible user-pair and column indices (i, j, a), with $i \neq j$ and $a \in [w]$. Therefore, there exists a PPT adversary $\mathcal{B}_{\text{rk}}$ such that $|\Pr[E_3] - \Pr[E_2]| \le N(N-1)w\mathsf{Adv}^{\mathsf{MLWE}}_{\mathsf{par}_{\mathbf{C}},\mathcal{B}_{\text{rk}}}$.

Each substituted matrix is used consistently in all rekey and re-encryption responses. In particular, an honest-to-honest re-encryption before rounding is

$$(\mathbf{0}^\top, \widetilde{v} + y_{\mathsf{ct},ij})^\top + \mathbf{U}_{ij}\mathbf{d}_{\mathsf{ct}} + \mathbf{P}_j \mathbf{r}_0 + \mathbf{e}_0. \tag{11}$$

This expression depends only on the public source ciphertext, the substituted matrix and seed, and fresh randomness, so the reduction can compute re-encryptions even for the challenge ciphertext.

Game G_4. The game is the same as G_3 except that the challenger rewrites the hint computation as follows.

For each honest user i with source secret key and error $(\mathbf{s}_i, \mathbf{e}_i)$, write $\mathbf{z}_i = (\mathbf{e}_i^\top, \mathbf{s}_i^\top)^\top$. Independently presample $(\mathbf{r}_t, \mathbf{e}_{u,t}) \leftarrow \chi^\ell \times \chi^k$ for its source encryptions $t \in [Q]$, and define

$$\mathbf{L}_i = \begin{pmatrix} \mathbf{r}_1^\top & -\mathbf{e}_{u,1}^\top \\ \vdots & \vdots \\ \mathbf{r}_Q^\top & -\mathbf{e}_{u,Q}^\top \end{pmatrix} \in \mathcal{R}^{Q\times p}, \quad \widetilde{\mathbf{L}}_i = (\mathbf{L}_i^\top, \ldots, \mathbf{L}_i^\top)^\top \in \mathcal{R}^{QM\times p}.$$

The latter stacks M copies of $\mathbf{L}_i$, as in Equation (8), and follows $\mathcal{S}_{Q,M}$. Let $\mathbf{Y}_i = (y_{t,\nu}) \leftarrow \mathcal{D}_{\sigma_y}^{Q\times M}$ independently, and let $\mathbf{y}_i$ be its columnwise concatenation. Define $\mathbf{h}_i^{\text{lin}} \coloneqq \widetilde{\mathbf{L}}_i \mathbf{z}_i + \mathbf{y}_i$, whose entry for source encryption t and rekey index ν is

$$h_{i,t,\nu}^{\text{lin}} = \langle \mathbf{e}_i, \mathbf{r}_t \rangle - \langle \mathbf{s}_i, \mathbf{e}_{u,t} \rangle + y_{t,\nu}.$$

To answer a permitted query $\mathcal{O}_{\mathsf{Hint}}(i, j, a)$, write the recorded ciphertext as $\mathsf{ct}_t = (\text{src}, \mathbf{u}_t, c_{v,t})$, where $t \in [Q]$ is its source-encryption index, and set $\widetilde{v}_t \coloneqq \lfloor c_{v,t} \rceil_{2^{t_v} \to q}$. Let $\nu \in [M]$ be the fixed index assigned to the hidden rekey $\mathsf{RK}[i, j]$ at creation. The challenger returns

$$h_{\mathsf{ct}_t, i\to j} = \widetilde{v}_t - \langle \mathbf{b}_i, \mathbf{r}_t \rangle + h_{i,t,\nu}^{\text{lin}} \pmod q. \tag{12}$$

Since $\mathbf{b}_i = \mathbf{A}^\top \mathbf{s}_i + \mathbf{e}_i$ and $\mathbf{u}_t = \mathbf{A}\mathbf{r}_t + \mathbf{e}_{u,t}$, the hint in Equation (12) equals $\widetilde{v}_t - \langle \mathbf{s}_i, \mathbf{u}_t \rangle + y_{t,\nu}$, exactly as in $\mathsf{HintGen}$. Presampling preserves the distribution of the source-encryption randomness and masks. Repeated queries retain their masks, and the collision abort is unchanged. Hence, $\Pr[E_4] = \Pr[E_3]$.

Game G_5. The game is the same as G_4 except that the challenger replaces each honest source public key $\mathbf{b}_i$ by an independent uniform vector $\mathbf{T}_i^{\text{S}} \leftarrow \mathcal{U}(\mathcal{R}_q^\ell)$. We reduce this change to Matrix Hint-MLWE as follows.

In G_4, the public key and linear hints satisfy

$$\mathbf{b}_i = (\mathbf{I}_\ell \mid \mathbf{A}^\top)\mathbf{z}_i \bmod q, \qquad \mathbf{h}_i^{\text{lin}} = \widetilde{\mathbf{L}}_i \mathbf{z}_i + \mathbf{y}_i.$$

Thus, $(\mathbf{A}, \mathbf{b}_i, \widetilde{\mathbf{L}}_i, \mathbf{h}_i^{\text{lin}})$ is a $\mathsf{MatrixHint\text{-}MLWE}^{QM,\mathcal{D}_{\sigma_y}^{QM},\mathcal{S}_{Q,M}}_{\mathcal{R},k,\ell,q,\mathcal{D}_{\sigma_{\text{S}}}^p}$ sample, allowing $\mathbf{b}_i$

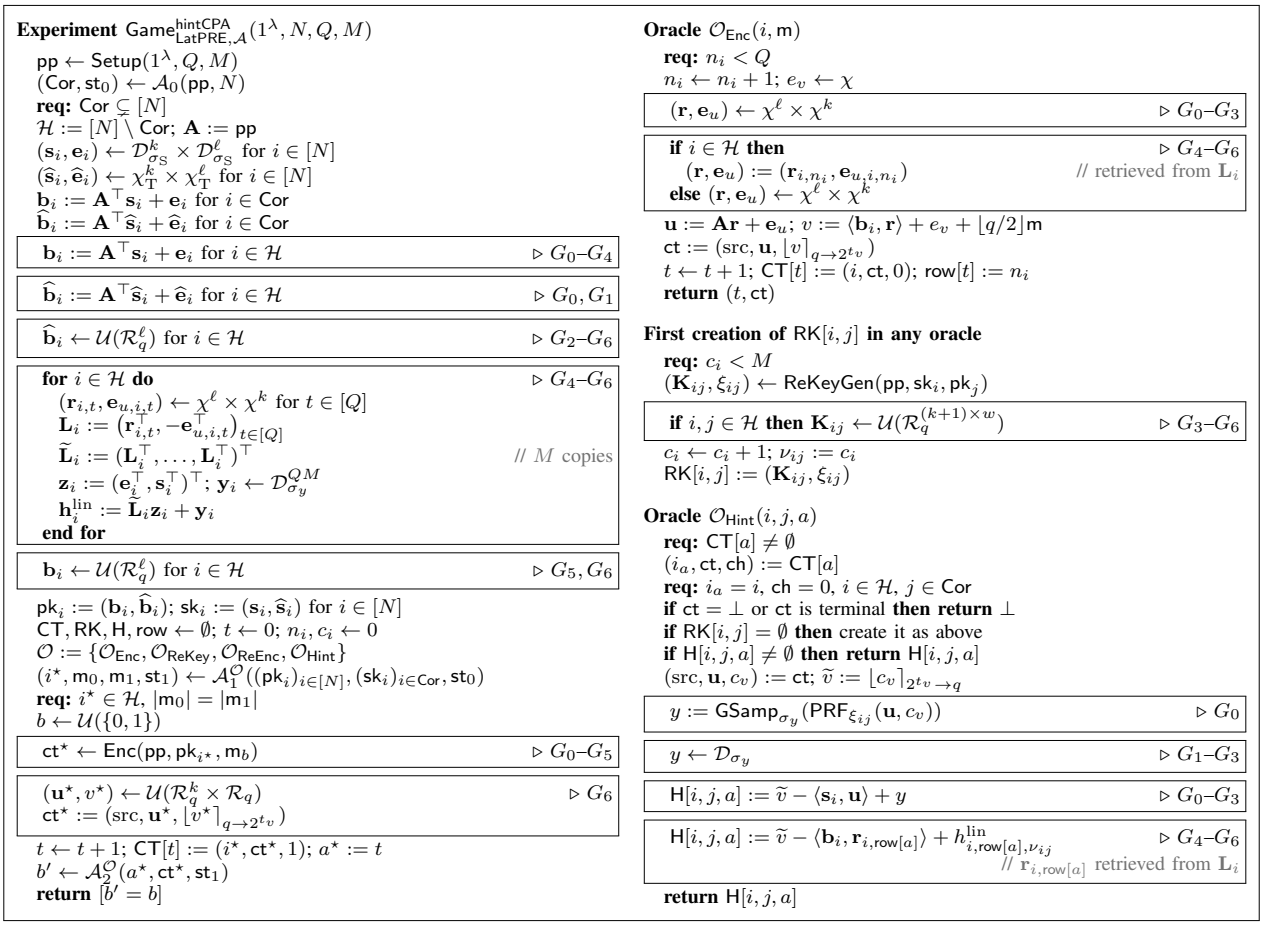

Figure 6. Games G_0–G_6 for the (Q, M)-hint CPA proof. Boxed steps apply only to the indicated games. The collision check introduced in G_1 is retained in all subsequent games.

to be replaced by $\mathbf{T}^{\mathrm{S}}_i$ while retaining the joint marginal of $(\widetilde{\mathbf{L}}_i, \mathbf{h}^{\mathrm{lin}}_i)$.

The reduction generates ct_t using $(\mathbf{r}_t, \mathbf{e}_{u,t})$ read from $\mathbf{L}_i$, fresh $e_{v,t} \leftarrow \chi$, and the current public key $\mathbf{b}_i$. It answers hint queries by Equation (12), using the corresponding entry of $\mathbf{h}^{\mathrm{lin}}_i$, thereby without the source secret key $\mathbf{s}_i$.

Furthermore, honest-to-honest rekeys are already uniform, and their replies follow (11). Honest-to-corrupted rekey matrices need not be generated: their disclosure and re-encryption services are forbidden, and hints follow Equation (12). Corrupted-source rekeys use known secrets. The challenge ciphertext is generated from the current public key with independent randomness. The reduction can therefore answer all permitted queries from the supplied Matrix Hint-MLWE sample. There are at most N honest users. Therefore, there exists a PPT adversary $\mathcal{B}_{\mathrm{MH}}$ such that $|\Pr[E_5] - \Pr[E_4]| \leq N\mathsf{Adv}^{\mathsf{MH}}_{\mathsf{par}_{\mathrm{MH}},\mathcal{B}_{\mathrm{MH}}}$.

Game G_6. The game is the same as G_5 except that the challenger samples $(\mathbf{u}^\star, v^\star) \leftarrow \mathcal{U}(\mathcal{R}^k_q \times \mathcal{R}_q)$ and returns $\mathsf{ct}^\star = (\mathrm{src}, \mathbf{u}^\star, \lfloor v^\star \rceil_{q\to 2^{t_v}})$. In G_5, the uncompressed challenge satisfies

$$(\mathbf{u}^{\star\top}, v^\star - \lfloor q/2 \rfloor m_b)^\top = (\mathbf{A}^\top, \mathbf{T}^{\mathrm{S}}_{i^\star})^\top \mathbf{r}^\star + (\mathbf{e}^{\star\top}_u, e^\star_v)^\top.$$

Since $(\mathbf{A}^\top, \mathbf{T}^{\mathrm{S}}_{i^\star})^\top$ is uniform in G_5, this is a $\mathsf{MLWE}_{\mathcal{R},\ell,k+1,q,\chi^{p+1}}$ sample. The reduction guesses the challenge-source label before setup and embeds its MLWE matrix in $\mathbf{A}$ and that label's uniform source key. If the guessed user is corrupted, the reduction aborts before returning the user keys. For a correct guess, it adds $(\mathbf{0}^\top, \lfloor q/2 \rfloor m_b)^\top$ to its supplied vector and applies the source compression. This gives the challenge of G_5 in the real branch and that of G_6 in the uniform branch, since translating a uniform vector preserves its distribution. For an incorrect guess, it outputs a fixed bit. Thus, there exists a PPT adversary $\mathcal{B}_{\mathrm{ct}}$ such that $|\Pr[E_6] - \Pr[E_5]| \leq N\mathsf{Adv}^{\mathsf{MLWE}}_{\mathsf{par}_{\mathrm{C}},\mathcal{B}_{\mathrm{ct}}}$. Uniform sampling and deterministic compression make the challenge distribution independent of b. No hint is generated for the challenge ciphertext, and its permitted re-encryptions follow (11). Hence, $\Pr[E_6] = 1/2$.

Combining the two reductions for $\mathsf{par}_{\mathrm{C}}$ into one weighted reduction $\mathcal{B}_{\mathrm{C}}$ and summing the game differences yields the claimed bound. □