Evasive Inner Product Functional Encryption Is Broken Too

Shweta Agrawal* Anuja Modi† Anshu Yadav‡ Shota Yamada§

Abstract

We show that the Evasive Inner Product Functional Encryption assumption introduced by Hsieh, Lin and Luo (Eurocrypt 2024) is false in its full generality. Our counterexample adds to a long sequence of works that have attacked different variants of Evasive LWE, taking down another instance that has been considered "public coin" (and hence safe) in the literature.

Our attack extends the counterexample against pseudorandom functional encryption (PRFE), developed by Agrawal, Modi, Yadav and Yamada (TCC 2025) to encompass the notion of Evasive Inner Product Functional Encryption (evIPFE) proposed by Hsieh, Lin and Luo (Eurocrypt 2024). The original attack relied crucially on the underlying functional encryption scheme being powerful enough to compute pseudorandom functions; our attack shows, perhaps surprisingly, that the much weaker "evasive" inner product functionality suffices to mount the attack.

We interpret our attack as showing that even the *slightest* power given to the adversary in choosing unrestricted private coins in the evasive LWE family of assumptions, renders it insecure.

1 Introduction

The Evasive Learning with Errors (Evasive LWE) assumption was recently introduced by [Wee22, Tsa22] to resolve the long-standing open problems of lattice-based broadcast encryption [Wee22] and witness encryption [Tsa22, VWW22]. Evasive LWE broadly states that revealing a Gaussian preimage to some structured matrix is safe so long as certain criteria are satisfied. Since its introduction, it has been successfully harnessed to tackle a large number of open problems in lattice based cryptography, which had thus far evaded solutions from more standard lattice assumptions [WWW22, AKYY23, ARYY23, HLL23, AKY24, WW24, MPV24, BDJ+25, AKY25]. We describe the Evasive LWE assumption next.

Evasive LWE. Below, we use curly underlines, i.e. $\underset{\sim}{\mathbf{x}}$ to denote a noisy version of $\mathbf{x}$, when the exact value of the noise is not important, $\$$ to denote the uniform distribution over the appropriate space and aux to denote some auxiliary information. Let $\mathbf{B}, \mathbf{P}$ be matrices of appropriate dimensions over some suitably chosen finite field and $\mathbf{s}$ be an LWE secret. Now, suppose that a "precondition" holds, namely

$$(\mathbf{B}, \mathbf{P}, \underset{\sim}{\mathbf{s}^\intercal\mathbf{B}}, \underset{\sim}{\mathbf{s}^\intercal\mathbf{P}}, \mathsf{aux}) \approx (\mathbf{B}, \mathbf{P}, \$, \$, \mathsf{aux})$$

for a sampler Samp outputting $(\mathbf{P}, \mathsf{aux})$. Then, the Evasive LWE assumption posits that the "post-condition" holds, i.e.,

$$(\mathbf{B}, \mathbf{P}, \underset{\sim}{\mathbf{s}^\intercal\mathbf{B}}, \mathbf{K}, \mathsf{aux}) \approx (\mathbf{B}, \mathbf{P}, \$, \mathbf{K}, \mathsf{aux})$$

*IIT Madras, India, shweta@cse.iitm.ac.in
†IIT Madras, India, anujamodi97@gmail.com
‡IIT Kanpur, India, anshuy@cse.iitk.ac.in
§AIST Tokyo, Japan, yamada-shota@aist.go.jp

where $\mathbf{K}$ is a low-norm matrix satisfying $\mathbf{BK} = \mathbf{P}$ and $\mathbf{B}$ is sampled uniformly. Note that the error in the term $\mathbf{s}^\intercal\mathbf{BK}$ obtained after multiplying $\mathbf{s}^\intercal\mathbf{B}$ and $\mathbf{K}$ is not i.i.d. unlike its counterpart $\mathbf{s}^\intercal\mathbf{P}$ in the pre-condition. However, this is assumed to be safe since the adversary never recovers the error without a large masking term. Thus, at a high level, the assumption posits that the pre-images $\mathbf{K}$ can only be used honestly to compute $\mathbf{s}^\intercal\mathbf{P}$, which in turn behaves as an LWE sample with fresh noise.

Public and Private Coin Variants. Broadly speaking, the Evasive LWE family of assumptions has been classified into the so-called "public-coin" and "private-coin" sub-categories, where in the former, random coins used by Samp are made available to the adversary, whereas in the latter, they are kept hidden.

Unfortunately, private-coin Evasive LWE has been invalidated by a wave of attacks [BUW24, BDJ+25, DJM+25, AMYY25, HJL25, HHY26]. The public-coin variant has remained immune to these attacks, except in unnatural parameter regimes [AMYY25]. The circular evasive LWE assumption, which seeks to extend the power of public-coin evasive LWE in a safe way [HLL23], was initially categorized as public-coin [HLL23, BDJ+25, CW25], but later shown to be insecure as well [AMYY25] [1]. The current rule of thumb is that public coin evasive LWE is safe while private coin is broken, hence it is extremely important to study variants of evasive LWE that are still considered public coin.

Evasive Inner Product Functional Encryption. A notable member of this family of assumptions which has been believed to be public coin is the Evasive Inner Product Functional Encryption (evIPFE) assumption, introduced by [HLL24a] (HLL henceforth). This evIPFE assumption was used to construct the evIPFE primitive (introduced in the same work with the same name)[2]. The latter was introduced to formulate a general framework for Attribute-Based Encryption (ABE) schemes based on evasive LWE assumption. This notion serves as the lattice counterpart of pairing-based IPFE, which together with Linear Secret Sharing was used to construct ABE [BW19].

In evasive IPFE, denoted by evIPFE, the ciphertext $\mathsf{ict}(\mathbf{u})$ encrypts $\mathbf{u} \in \mathbb{Z}_q^Z$ and the secret key $\mathsf{isk}(\mathbf{v})$ encodes $\mathbf{v} \in \mathbb{Z}_q^Z$, similar to standard IPFE. However, the correctness condition is somewhat relaxed, i.e., the decryption results in $y = \mathbf{u}^\intercal\mathbf{v} + e$ for some bounded error e. The security of evIPFE states that if noisy inner products are pseudorandom with freshly generated noise, then the ciphertext is pseudorandom even in the presence of secret keys. More formally, it is defined with respect to two samplers $\mathsf{Samp}_{\mathbf{V}}, \mathsf{Samp}_{\mathbf{U}}$. The sampler $\mathsf{Samp}_{\mathbf{V}}$ on input public randomness r_{pub}, outputs $(Z, Q_k, Q_c, \{\mathbf{v}_i\}_i)$ for $i \in [Q_k]$. The sampler $\mathsf{Samp}_{\mathbf{U}}$ on input private randomness r_{priv} and r_{pub} outputs $\{\mathbf{u}_j\}_j$ for $j \in [Q_c]$. The security notion requires that if the precondition holds, namely

$$\left\{(r_{\mathsf{pub}}, \{\mathbf{u}_j^\intercal\mathbf{v}_i + e_{i,j}\}_{i,j})\right\}_\lambda \approx \left\{(r_{\mathsf{pub}}, \{\$_{i,j}\}_{i,j})\right\}_\lambda$$

for above-described sampler, where $e_{i,j}$ are i.i.d, then

$$\left\{(r_{\mathsf{pub}}, \mathsf{impk}, \{\mathsf{ict}(\mathbf{u}_j)\}_j, \{\mathsf{isk}(\mathbf{v}_i)\}_i)\right\}_\lambda \approx \left\{(r_{\mathsf{pub}}, \mathsf{impk}, \{\mathsf{ict}(\$_j)\}_j, \{\mathsf{isk}(\mathbf{v}_i)\}_i)\right\}_\lambda .$$

The evIPFE was constructed by HLL, and the security was provided from the evIPFE assumption, which states the following. If the precondition holds, namely

$$\left\{(r_{\mathsf{pub}}, \{\mathbf{u}_j^\intercal\mathbf{v}_i + e_{i,j}\}_{i,j})\right\}_\lambda \approx \left\{(r_{\mathsf{pub}}, \{\$_{i,j}\}_{i,j})\right\}_\lambda$$

for samplers $\mathsf{Samp}_{\mathbf{V}}, \mathsf{Samp}_{\mathbf{U}}$, where $e_{i,j}$ are i.i.d, then

$$\left\{(r_{\mathsf{pub}}, \mathbf{A}, \mathbf{B}, \mathbf{K}, \{\mathbf{s}_j^\intercal\mathbf{B}, \mathbf{s}_j^\intercal\mathbf{A} + \mathbf{u}_j^\intercal \otimes \mathbf{g}^\intercal\}_j\right\}_\lambda \approx \left\{(r_{\mathsf{pub}}, \mathbf{A}, \mathbf{B}, \mathbf{K}, \{\$, \$\}_j\right\}_\lambda$$

[1] Brzuska et. al. [BUW24] noticed that it does not fall into the regime of public-coin in the strict sense.

[2] Henceforth, we omit the qualifier "primitive": we refer evIPFE primitive by just evIPFE, whereas the assumption is explicitly referred to as the evIPFE assumption.

where $\mathbf{K}$ is a low-norm matrix satisfying $\mathbf{BK} = \mathbf{AG}^{-1}(\mathbf{V})$, $\mathbf{V}$ is the column-wise concatenation of $\mathbf{v}_i$, $\mathbf{g}$ is the "gadget" vector $(1, 2, 4, \ldots, 2^{\lceil \log q \rceil})^\intercal$, and λ is the security parameter.

Our Results. In this work, we show that the evIPFE assumption [HLL24a], which has been seen as a fairly mild strengthening of the public-coin Evasive LWE, as stated, is false. Our attack suggests that allowing the adversary even the slightest control in choosing private coins in the evasive LWE family of assumptions can be problematic.

In more detail, we show that it is impossible to construct the evIPFE that is secure for all sampler distributions. Since HLL shows that their evIPFE assumption implies an evIPFE for all samplers, this translates into an attack against the variant of the evasive LWE assumption they use.

We next define the different versions of the evIPFE and the assumption, at a high level as specified by HLL, and precisely specify which versions our attacks target.

1. Versions of evIPFE:

   With/Without Structured Noise: The primitive defined above is the version without structured noise. In the version with structured noise, the decrypted output contains an additional instance of structured noise, along with the noisy inner product. In particular, the decryption result of the ciphertext $\mathsf{ict}(\mathbf{u})$ using a secret key $\mathsf{isk}(\mathbf{V})$, which now encodes a matrix $\mathbf{V}$, results in $y = \mathbf{u}^\intercal \mathbf{V} + \mathbf{e}^\intercal + e'\mathbf{g}^\intercal$. Here, $e'\mathbf{g}^\intercal$ is the structured noise. Now, the security notion requires that if the precondition holds, namely

   $$(\{\mathbf{V}_k, \mathbf{u}^\intercal \mathbf{V}_k + \mathbf{e}_k^\intercal + e'_k \mathbf{g}^\intercal\}_k) \approx (\{\mathbf{V}_k, \$\}_k)$$

   where $e'_k, \mathbf{e}_k$ are i.i.d noise, then

   $$(\mathsf{impk}, \mathsf{ict}(\mathbf{u}), \{\mathbf{V}_k, \mathsf{isk}(\mathbf{V}_k)\}_k) \approx (\mathsf{impk}, \$, \{\mathbf{V}_k, \mathsf{isk}(\mathbf{V}_k)\}_k)$$

   We first show that the version without structured noise is impossible. Since the version with structured noise is stronger than the version without, our impossibility also applies to this version.

   Restricted/Unrestricted: In the restricted version, $\mathsf{Samp}_{\mathbf{U}}$ is restricted to output $\mathbf{u}$ of the form $\mathbf{d}^\intercal \mathbf{A}$ where vector $\mathbf{d}$ is uniformly sampled from $\mathbb{Z}_q$ and public matrix $\mathbf{A}$ is the output of some deterministic algorithm. The unrestricted version is the one without such a restriction. We provide the impossibility against the unrestricted version.

2. Versions of evIPFE assumption:

   Conservative/Non-conservative: The assumption described previously is the non-conservative version. In the conservative version, only the precondition changes to a stricter form, i.e.,

   $$\left\{(r_{\mathsf{pub}}, \mathbf{A}, \mathbf{B}, \{\underbrace{\mathbf{s}_j^\intercal \mathbf{AG}^{-1}(\mathbf{V})}, \underbrace{\mathbf{s}_j^\intercal \mathbf{B}}, \underbrace{\mathbf{s}_j^\intercal \mathbf{A} + \mathbf{u}_j^\intercal \otimes \mathbf{g}^\intercal}\}_j\right\}_\lambda \approx \left\{(r_{\mathsf{pub}}, \mathbf{A}, \mathbf{B}, \{\$, \$, \$\}_j\right\}_\lambda$$

   which intuitively says that one can use $\mathbf{K} = \mathbf{B}^{-1}(\mathbf{AG}^{-1}(\mathbf{V}))$ only in an honest way by multiplying it with $\underbrace{\mathbf{s}_j^\intercal \mathbf{B}}$. Let $\mathsf{evIPFE}_{\mathsf{ipre}}$ and $\mathsf{evIPFE}_{\mathsf{spre}}$ be the pre-conditions of the normal and conservative versions respectively. Since HLL proved that $\mathsf{evIPFE}_{\mathsf{ipre}}$ implies $\mathsf{evIPFE}_{\mathsf{spre}}$ under the LWE assumption with noise super-polynomially smaller than pre-condition noise, our attack also translates to this version of the assumption under the same condition.

   Restricted/Unrestricted: This is the same as the restricted/unrestricted version of evIPFE. Our attack works against the unrestricted version.

The summary of the impossibility of different versions of evIPFE and the attack on different versions of evIPFE assumption are shown in Table 1. In here, "yes" indicates broken in this work, and "no" means it is not yet broken.

Table 1: Summary of impossibilities and attack on different versions of evIPFE and assumption implied by our counter-example (indicated by Yes/No).

	evIPFE **Versions**		**Assumption Versions**	
	w/o struct. noise	with struct. noise	Conservative	Non-conservative
Restricted	No	No	No	No
Unrestricted	Yes	Yes	Yes	Yes

Public and private coins in the evIPFE assumption. As discussed in HLL, it is crucial that $\mathbf{U}$, and hence the coins r_{priv} used by $\mathsf{Samp}_{\mathbf{U}}$ used to sample $\mathbf{U}$, be hidden. All other coins, in particular coins of $\mathsf{Samp}_{\mathbf{V}}$ are public, giving the impression of evIPFE assumption as a public-coin assumption since all the coins except those which must necessarily be private, are public. However, our attack shows that this seemingly unavoidable privacy granted to $\mathsf{Samp}_{\mathbf{U}}$ provides it enough flexibility to mount our attack, unless it is restricted to sample $\mathbf{U}$ only in a specific way that bars it from embedding any hidden clue in $\mathbf{U}$. The restricted evasive IPFE in HLL, to which our attack does not apply, does indeed limit $\mathsf{Samp}_{\mathbf{U}}$ to sample from a highly restricted distribution.

Implications of our attack. Our attack breaks the construction of Evasive IPFE (Construction 1) in [HLL24b] when $\mathsf{Samp}_{\mathbf{U}}$ is unrestricted, for both the structured and unstructured settings, but does not apply when $\mathsf{Samp}_{\mathbf{U}}$ is restricted. The Attribute-Based Encryption application in [HLL24b] uses the restricted version and therefore remains unaffected.

1.1 Technical Overview

Counter-example against Evasive IPFE. Below, we exhibit the impossibility of constructing the primitive evasive IPFE, defined by HLL, that is secure for all sampler distributions. Since HLL shows that evasive IPFE assumption implies evasive IPFE for all samplers, this translates into an attack against the evasive LWE assumption. Our counterexample is inspired by [AMYY25] (henceforth AMYY) counterexample against pseudorandom functional encryption (prFE) [AKY25], which builds upon prior work against pseudorandom obfuscation [BDJ+25].

We explain our counter-example from two complementary perspectives. For ease of understanding, in Section 1.1.1, we present the high-level ideas directly, and in Section 1.1.2, we first recall the AMYY attack and then explain how its ideas carry over to our setting.

1.1.1 Direct Overview of the attack

We show the counter-example against evIPFE by constructing a specific efficient sampler $(\mathsf{Samp}_{\mathbf{V}}, \mathsf{Samp}_{\mathbf{U}})$ for which the precondition holds but the post-condition fails, assuming Witness Encryption (WE) with pseudorandom ciphertexts and Learning With Errors (LWE). Recall that a WE scheme [GGSW13] is defined using a language $\mathcal{L} \in \mathsf{NP}$, where a message is encrypted with respect to a string $\mathsf{stmt} \in \{0,1\}^*$ and decryption is possible using a witness for $\mathsf{stmt} \in \mathcal{L}$. If $\mathsf{stmt} \notin \mathcal{L}$, the message remains hidden. The pseudorandom ciphertext property says that the WE ciphertext looks uniformly random.

Setup. Vectors $\mathbf{u}_j$ are sampled by $\mathsf{Samp}_{\mathbf{U}}$ using r_{priv}. Since the coins r_{priv} are private, $\mathsf{Samp}_{\mathbf{U}}$ uses this freedom to hide WE ciphertexts inside them. The WE ciphertexts encrypt the fixed messages 0 and 1

under a statement whose witness is simply the evIPFE ciphertexts and secret keys. In more detail, we construct $(\mathrm{Samp}_{\mathbf{V}}, \mathrm{Samp}_{\mathbf{U}})$ as follows.

$\mathrm{Samp}_{\mathbf{V}}$. It samples $\mathbf{v}_i \leftarrow \mathbb{Z}_q^Z$ using r_{pub} and outputs $\widetilde{\mathbf{v}}_i = (\mathbf{v}_i, 0)^\intercal$ for all $i \in [Q_k]$, together with $\widetilde{\mathbf{v}}_{Q_k+1} = (0, \ldots, 0, 1)^\intercal \in \mathbb{Z}_q^{Z+1}$.

$\mathrm{Samp}_{\mathbf{U}}$. It samples $\mathbf{u}_\ell \leftarrow \mathbb{Z}_q^Z$ for $\ell \in [Q_w]$ using r_{priv} and forms the WE statement $\mathsf{stmt} = \{\mathbf{u}_j^\intercal \mathbf{v}_i + e_{i,j}\}_{i\in[Q_k], j\in[Q_c]}$, where the $e_{i,j}$ are i.i.d noise and $Q_c < Q_w$. The NP language $\mathcal{L}$ contains all tuples for which there exist Q_c evIPFE ciphertexts and Q_k secret keys whose pairwise decryptions are close to the tuple. It then outputs $\{\tilde{\mathbf{u}}_\ell = (\mathbf{u}_\ell, \mathsf{w}_\ell \lfloor q/2 \rceil + z_\ell)^\intercal \in \mathbb{Z}_q^{Z+1}\}_{\ell\in[Q_w]}$, where w_ℓ is the ℓ-th bit of $(\mathsf{wct}(0), \mathsf{wct}(1))$, Q_w is its total bit length, and z_ℓ is chosen uniformly from $[-q/4, q/4]$. Here, $\mathsf{wct}(b)$ denotes the WE encryption of $b \in \{0,1\}$ under stmt. Placing w_ℓ in the most significant position lets it survive the noisy decryption, while z_ℓ makes the last coordinate look uniform over $\mathbb{Z}_q$.

Distinguishing the post-condition. Given the challenge, the distinguisher decrypts the evIPFE ciphertexts of $\tilde{\mathbf{u}}_\ell$ for all $\ell \in [Q_w]$ using the secret key for $\widetilde{\mathbf{v}}_{Q_k+1}$, rounds each result to a bit w_ℓ, and reassembles $(\mathsf{wct}(0), \mathsf{wct}(1))$. It then tries to decrypt both, using as witness the first Q_c evIPFE ciphertexts together with the secret keys for $\widetilde{\mathbf{v}}_1, \ldots, \widetilde{\mathbf{v}}_{Q_k}$, and outputs "real" if and only if it obtains $(0,1)$.

In the real world, rounding recovers $(\mathsf{wct}(0), \mathsf{wct}(1))$ exactly. Moreover, since $\langle \tilde{\mathbf{u}}_j, \widetilde{\mathbf{v}}_i \rangle = \mathbf{u}_j^\intercal \mathbf{v}_i$, these ciphertexts and keys decrypt to values close to stmt, so they form a valid witness and WE decryption yields $(0,1)$.

In the ideal world, however, the evIPFE ciphertexts encrypt random vectors, so decryption yields garbage: the recovered string is uniformly random rather than an encryption of $(0,1)$. Hence, WE decryption returns $(0,1)$ with probability at most $1/4$ (encrypting both 0 and 1 is what guarantees this bound), which distinguishes the post-condition.

Proof of the pre-condition. We now show that, even given r_{pub}, the noisy inner products look uniform, even though the WE ciphertext hidden among them depends on stmt. We first use noise flooding to hide the noise used to form stmt, and then use LWE to replace all inner products with random values; in particular, stmt becomes random. A random stmt almost never lies in $\mathcal{L}$, since the witnesses (the evIPFE ciphertexts and keys) are much shorter than the statement. We can therefore use the pseudorandomness of WE ciphertexts to replace $(\mathsf{wct}(0), \mathsf{wct}(1))$ with random bits, after which each last coordinate $\mathsf{w}_\ell \lfloor q/2 \rceil + z_\ell$ is statistically close to uniform over $\mathbb{Z}_q$.

1.1.2 An Overview via the AMYY attack strategy

PRFE Counterexample. Let us recall the notion of prFE and then show their attack. The syntax of prFE is the same as the standard notion of FE: There is a setup algorithm that generates master public key mpk and a master secret key msk, the key generation algorithm that takes as input the master secret key msk and a function f and outputs a secret key $\mathsf{sk}(f)$, and an encryption algorithm that takes as input mpk and a message x and outputs a ciphertext $\mathsf{ct}(x)$. By the decryption, you get the result $f(x)$. While the syntax is the same, the security notion of prFE is different: It essentially says that if the decryption result is pseudorandom, then the ciphertext should be pseudorandom as well. More formally, this is defined with respect to a sampler distribution: The sampler Samp outputs a set of messages $x_1, \ldots, x_{Q_c}$, functions $f_1, \ldots, f_{Q_k}$, and an auxiliary information aux. The security notion of prFE requires that if the following precondition

$$(\{f_i\}_{i\in[Q_k]}, \{f_i(x_j)\}_{i,j}, \mathsf{aux}) \approx_c (\{f_i\}_{i\in[Q_k]}, \{\$\}_{i,j}, \mathsf{aux})$$

for a sampler $\mathsf{Samp}(1^\lambda) \to (x_1, \ldots, x_{Q_c}, f_1, \ldots, f_{Q_k})$ holds, where \$ represents a random string of appropriate length, then the following post-condition should hold:

$$(\mathsf{ct}(x_1), \ldots, \mathsf{ct}(x_{Q_c}), \mathsf{sk}(f_1), \ldots, \mathsf{sk}(f_{Q_k}), \{f_i\}_i, \mathsf{aux}) \approx (\$, \ldots, \$, \mathsf{sk}(f_1), \ldots, \mathsf{sk}(f_{Q_k}), \{f_i\}_i, \mathsf{aux})$$

The security of prFE requires this condition to hold for all efficient samplers Samp.

The AMYY *Attack.* AMYY (building upon [BDJ+25]) show that for any prFE, there exists an efficient sampler Samp such that the precondition holds while the postcondition does not. To show the counterexample, they proceed in two stages: they first construct a sampler that generates auxiliary information aux using private randomness, and then modify the sampler to eliminate the need for aux. The core idea relies on an incompressibility argument following [BDJ+25]. Specifically, consider Q_c ciphertexts and Q_k secret keys, which together yield $Q_c \cdot Q_k$ decryption results that are pseudorandom. By choosing Q_c and Q_k sufficiently large, the combined description of the ciphertexts and keys is much smaller than the full set of outputs, thereby serving as a compressed representation. They then introduce aux so that this compressed description becomes useful for distinguishing the relevant distributions.

To realize this, they assume the existence of a witness encryption (WE) scheme. They instantiate the construction with $x_i = K_i$ (a PRF key) and $f_j = \mathsf{PRF}(\cdot, j)$, which maps a PRF key K to $\mathsf{PRF}(K, j)$. The resulting decryption values $R \stackrel{\text{def}}{=} \{r_{i,j} = \mathsf{Dec}(\mathsf{ct}(x_i), \mathsf{sk}_{f_j})\}_{i,j}$ are pseudorandom. The auxiliary information aux is then defined as a WE ciphertext of the statement: "there exist Q_c prFE ciphertexts and Q_k secret keys such that their combined decryption results equal R." The postcondition distribution becomes distinguishable, since on the LHS the prFE ciphertexts and secret keys serve as witnesses to decrypt the WE ciphertext, whereas on the RHS, if the prFE ciphertexts are random, decryption fails with high probability. The precondition is shown by replacing R with uniform randomness (possible in the absence of prFE ciphertexts and secret keys), and then observing that the WE ciphertext reveals nothing, since the statement is false for random R with overwhelming probability, which is shown by a simple counting argument.

In the final step, they eliminate the need for auxiliary information entirely. This is done by introducing an additional ciphertext–key pair whose combination directly yields a WE ciphertext. To ensure that the overall decryption results remain pseudorandom, the ciphertext WE must be pseudorandom. Accordingly, they assume the WE scheme satisfies the pseudorandom ciphertext property.

Counter-example against Evasive IPFE. We now present our attack on evIPFE assuming WE with pseudorandom ciphertext and LWE. We first note that if we are allowed to include aux generated using the private coin into the distributions, then the original attack of AMYY directly extends to our setting. Concretely, we encrypt random (secret) vectors $\mathbf{u}_j$ into ict ciphertexts, generate random vectors $\mathbf{v}_i$ using r_{pub}, and issue secret keys associated with these vectors. By the LWE assumption, the decryption results with i.i.d. noise, $R = \{\mathbf{u}_j^\intercal \mathbf{v}_i + e_{i,j}\}_{i,j}$, are pseudorandom. We then set aux to be a WE ciphertext for the statement: "there exist Q_c ciphertexts and Q_k secret keys such that all their decryption results, taken over all key–ciphertext pairs, *approximate* R." Here, the statement is slightly adapted to account for noisy decryption. Following arguments analogous to AMYY, we can show that the post-condition distribution is distinguishable. The indistinguishability of the pre-condition follows a similar proof, except that we must carefully account for the change of statement when applying the counting argument for WE security.

We now turn to removing the auxiliary information to obtain a full-fledged attack. As in the case of prFE, our goal is to recover the WE ciphertext from decryptions of the evIPFE ciphertexts, while keeping the pseudorandom decryption results guarantee. This can be achieved by generating a secret key for an additional identity vector $\mathbf{v}' = (0, \ldots, 0, 1)$. Using this key, we can recover the WE ciphertext by decrypting evIPFE ciphertexts that encrypt vectors formed by appending an extra $\mathbb{Z}_q$ element to $\mathbf{u}_j$. A naïve encoding—appending each bit of the WE ciphertext to $\mathbf{u}_j$—fails because evIPFE decryption is noisy.

Instead, we encode each bit w_ℓ of the WE ciphertext into the most significant bits, while randomizing the lower bits to maintain pseudorandomness. Specifically, we append $w_\ell \cdot \lfloor q/2 \rceil + z_\ell$, where w_ℓ is the ℓ-th bit of the WE ciphertext and z_ℓ is chosen uniformly from $[-q/4, q/4]$. By the pseudorandomness of WE ciphertexts, the most significant bits remain pseudorandom; and with the added noise z_ℓ, the decryption results are distributed pseudorandomly over the full domain $\mathbb{Z}_q$, as desired. We can also see that the private coin is only used for generating $\{\mathbf{u}_j\}_j$. We refer the reader to Section 3 for details.

Broader Comparison of our techniques with techniques of AMYY. Having discussed the technical ideas of our approach, we now compare it with AMYY. While the high-level strategy—embedding WE ciphertexts into $\mathsf{Samp}_{\mathbf{U}}$'s output, defining witnesses as real evIPFE ciphertexts and secret keys, and leveraging these to decrypt WE ciphertexts (recovering messages in the real world, obtaining garbage in the ideal world)—builds on AMYY's framework, adapting it to our setting requires two key modifications.

- AMYY relies on the underlying prFE scheme to evaluate a sufficiently powerful PRF, ensuring pseudorandom decryption values. However, PRF evaluation requires operations beyond inner products. However, we note that, in our setting, this additional power is unnecessary: sampling a random element (z_ℓ) from the appropriate distribution suffices.
- Directly embedding WE ciphertexts into $\mathsf{Samp}_{\mathbf{U}}$'s output fails in our case because evIPFE decryption is inherently noisy—a challenge absent in AMYY. We address this by encoding each WE ciphertext bit in the most significant bit position, with randomization in the low bits ($\mathsf{w}_\ell \cdot \lfloor q/2 \rceil + z_\ell$). Rounding recovers the intended bit while maintaining statistical near-uniformity of the resulting value.

Our main conceptual contribution is to demonstrate that a simple inner-product functionality suffices to execute the AMYY attack strategy, which initially appeared to require the full power of prFE.

2 Preliminaries

In this section, we present the notation and the required preliminaries.

Notation. We use bold capital letters to denote a matrix and bold small letters to represent vectors. By default, a vector $\mathbf{v}$ is a column vector and $\mathbf{v}^\intercal$ is a row vector. Throughout the paper, we use λ to denote the security parameter. Let $q, n \in \mathbb{N}$. Sometimes, we identify an element $x \in \mathbb{Z}_q$ with its centered representative in $[-\lfloor q/2 \rfloor, \lfloor q/2 \rfloor]$, and vice versa. We slightly overload the notation $|\cdot|$: for a scalar $v \in \mathbb{Z}$, $|v|$ denotes the absolute value of v; for a set S, $|S|$ denotes the number of elements in S and for any cryptographic object obj (in particular ciphertexts and keys), $|\mathsf{obj}|$ denotes the size of obj. For any element $v \in \mathbb{Z}_q$, we denote $(b_1, \ldots, b_{\lceil \log q \rceil}) = \mathsf{bits}(v)$ to be the bit representation of v such that $b_i \in \{0,1\}$ and b_1 is MSB. The notation is easily extended to the vector $\mathbf{v} \in \mathbb{Z}_q^n$. For any vector $\mathbf{v}$, $\mathbf{v}[i]$ denotes the ith element.

2.1 Lattice Preliminaries

Here, we recall some facts on lattices that are needed for the technical part. Throughout this section, n, m, and q are integers such that $n = \mathrm{poly}(\lambda)$ and $m \geq n\lceil \log q \rceil$. In the following, we define a function ρ_σ as $\rho_\sigma = \mathsf{Exp}(-\pi \|\mathbf{x}/\sigma\|_2^2)$, where $\|\cdot\|_2$ is the Euclidean norm. We will overload the notation to denote by $\rho_\sigma(S)$ the summation of $\rho_\sigma(\mathbf{x})$ over all $\mathbf{x} \in S$ for a countable set S. For a countable set S, $\mathcal{D}_{S,\sigma}$ refers to a distribution that outputs $\mathbf{x} \in S$ with probability $\rho_\sigma(\mathbf{x})/\rho(S)$.

Let $n, m \geq 1$ and $q \geq 2$ be integers such that $\log_2 q \leq m/n \in \mathbb{Z}$. Let

$$\mathbf{g} = (2^0, \ldots, 2^{m/n-1})^\intercal \quad \text{and} \quad \mathbf{G} = \mathbf{I}_n \otimes \mathbf{g}^\intercal$$

be the gadget vector and the gadget matrix. Given $\mathbf{p} \in \mathbb{Z}_q^n$, we denote by $\mathbf{G}^{-1}(\mathbf{p})$ the vector $\mathbf{k} \in \{0,1\}^m$, where $\mathbf{k}[(i-1)\cdot m/n+1], \ldots, \mathbf{k}[i \cdot m/n]$ are the bits of (the non-negative representative less than q of) $\mathbf{p}[i]$, low to high. It holds that $\mathbf{G}\mathbf{G}^{-1}(\mathbf{p}) = \mathbf{p}$. Given $K \in \mathbb{N}$, we let (note the different order in the Kronecker product)

$$\widetilde{\mathbf{g}} = (2^0, \ldots, 2^{K-1})^\intercal \quad \text{and} \quad \widetilde{\mathbf{G}} = \widetilde{\mathbf{g}}^\intercal \otimes \mathbf{I}.$$

When $2^K \geq q$, the vector $\widetilde{\mathbf{G}}^{-1}(\mathbf{p})$ is again the bit decomposition of $\mathbf{p} \in \mathbb{Z}_q^*$, arranged so that $\widetilde{\mathbf{G}}\widetilde{\mathbf{G}}^{-1}(\mathbf{p}) = \mathbf{p}$. The notations $\mathbf{G}^{-1}(\cdot), \widetilde{\mathbf{G}}^{-1}(\cdot)$ extend to matrices column-wise.

Lemma 2.1 (Smudging Lemma [WWW22]). Let λ be a security parameter. Take any $a \in \mathbb{Z}$ where $|a| \leq B$. Suppose $\gamma \geq B\lambda^{\omega(1)}$. Then the statistical distance between the distributions $\{z : z \leftarrow \mathcal{D}_{\mathbb{Z},\gamma}\}$ and $\{z + a : z \leftarrow \mathcal{D}_{\mathbb{Z},\gamma}\}$ is $\mathsf{negl}(\lambda)$.

Assumption 2.2 (The LWE Assumption [Reg09]). Let $n = n(\lambda), m = m(\lambda), \gamma = \gamma(\lambda)$ and $q = q(\lambda) > 2$ be integers.We say that the $\mathrm{LWE}(n, m, q, \gamma)$ hardness assumption holds if for any PPT adversary $\mathcal{A}$ we have

$$|\Pr[\mathcal{A}(\mathbf{A}, \mathbf{s}^\intercal\mathbf{A} + \mathbf{e}^\intercal) \to 1] - \Pr[\mathcal{A}(\mathbf{A}, \mathbf{v}^\intercal) \to 1]| \leq \mathsf{negl}(\lambda)$$

where the probability is taken over the choice of the random coins by the adversary $\mathcal{A}$ and $\mathbf{A} \leftarrow \mathbb{Z}_q^{n\times m}$, $\mathbf{s} \leftarrow \mathbb{Z}_q^n$, $\mathbf{e} \leftarrow \mathcal{D}_{\mathbb{Z},\gamma}^m$, and $\mathbf{v} \leftarrow \mathbb{Z}_q^m$. We also say that $\mathrm{LWE}(n, m, q, \gamma)$ problem is subexponentially hard if the above probability is bounded by $2^{-n^\epsilon} \cdot \mathsf{negl}(\lambda)$ for some constant $0 < \epsilon < 1$ for all PPT $\mathcal{A}$.

Lemma 2.3 ([MP12]). There are efficient algorithms $\mathsf{TrapGen}$ and $\mathsf{SampleD}$, and functions $m_0 \in \Theta(n \log q)$ and $\sigma_0 \in \omega(\sqrt{m \log m}) \cap \mathrm{O}(m)$ satisfying these conditions.

- $\mathsf{TrapGen}(1^n, 1^m, q)$ takes as input $n \geq 1$, $q \geq 2$, and $m \geq m_0(n, q)$. It outputs $(\mathbf{B}, \tau)$ such that $\mathbf{B} \in \mathbb{Z}_q^{n\times m}$ and $\mathbf{B}$ is $\mathrm{negl}(n)$-close to uniform over $\mathbb{Z}_q^{n\times m}$.
- $\mathsf{SampleD}(\mathbf{B}, \tau, \mathbf{p}, \sigma)$ takes as input $\mathbf{B}, \tau$ from $\mathsf{TrapGen}$, some $\mathbf{p} \in \mathbb{Z}_q^n$, and $\sigma \geq \sigma_0(n, m)$. It outputs $\mathbf{k} \in \mathbb{Z}^m$ such that $\mathbf{B}\mathbf{k} = \mathbf{p}$, $\|\mathbf{k}\| \leq \sigma\sqrt{m}$, and $\mathbf{k}$ is $\mathrm{negl}(n)$-close to $\mathcal{D}_{\Lambda_\mathbf{p}^\perp(\mathbf{B}),\sigma}$.

2.2 Evasive Inner Product Functional Encryption

Here we present verbatim the definitions of different versions of the Evasive Inner Product Functional Encryption scheme and the assumption, as introduced in [HLL24a], with identity space $\{1\}$.

2.2.1 Evasive IPFE (with structured noise).

In this section, we verbatim recall the definition of Evasive IPFE (evIPFE) with or without structured noise that corresponds to Definition 4 of [HLL24b]. Below, $\mathfrak{s} = \mathfrak{o}$ and $\mathfrak{s} = \mathfrak{g}$ (for which we use blue colors) correspond to terms without structure noise and with structure noise, respectively. For evIPFE without structured noise, we just ignore the blue terms and set $\mathfrak{s} = \mathfrak{o}$.

Syntax. A Evasive Inner Product Functional Encryption evIPFE scheme (with structured noise) consists of four polynomial time algorithms $(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ defined as follows.

$\mathsf{Setup}(1^\lambda, q, 1^K, 1^Z) \to (\mathsf{impk}, \mathsf{imsk})$. The setup algorithm takes as input the security parameter λ, modulus q, the noise parameter K, and dimension Z, and outputs a public key impk and a master secret key imsk. We implicitly assume that 1^λ is input to the rest of the algorithms.

$\mathsf{KeyGen}(\mathsf{imsk}, \mathbf{v}_\mathfrak{o}, \mathbf{V}_\mathfrak{g}) \to \mathsf{isk}$. The key generation algorithm takes as input the master secret key imsk, a key vector $\mathbf{v}_\mathfrak{o} \in \mathbb{Z}_q^Z$, and a key matrix $\mathbf{V}_\mathfrak{g} \in \mathbb{Z}_q^{Z\times K}$ and it outputs a secret key isk for $(\mathbf{v}_\mathfrak{o}, \mathbf{V}_\mathfrak{g})$. We implicitly assume that isk contains $(\mathbf{v}_\mathfrak{o}, \mathbf{V}_\mathfrak{g})$.

$\mathsf{Enc}(\mathsf{impk}, \mathfrak{s}, \mathbf{u}) \to \mathsf{ict}$. The encryption algorithm takes as input the public key impk, a noise structure $\mathfrak{s} \in \{\mathfrak{o}, \mathfrak{g}\}$, and an input vector $\mathbf{u} \in \mathbb{Z}_q^Z$ and outputs a ciphertext $\mathsf{ict} \in \mathcal{CT}$, where $\mathcal{CT}$ is the ciphertext space of $\mathbf{u}$ for noise structure $\mathfrak{s}$.

$\mathsf{Dec}(\mathsf{impk}, \mathsf{isk}, \mathsf{ict}, \mathfrak{s}) \to y$. The decryption algorithm takes as input impk, isk, ict, $\mathfrak{s}$ and outputs $y \in \mathbb{Z}_q$.

Definition 2.4 (Correctness). Let B be a function mapping (λ, q, K, Z) to natural numbers. The scheme is B-correct for $\mathfrak{o}$, if for all $\lambda, q, K, Z \in \mathbb{N}, \mathbf{V}_\mathfrak{g} \in \mathbb{Z}_q^{Z \times K}$, $\mathbf{v}_\mathfrak{o}, \mathbf{u} \in \mathbb{Z}_q^Z$, it holds that

$$\Pr\left[\begin{array}{l|l} (\mathsf{impk}, \mathsf{imsk}) \leftarrow \mathsf{Setup}(1^\lambda, q, 1^K, 1^Z) & \\ \mathsf{isk} \leftarrow \mathsf{KeyGen}(\mathsf{imsk}, \mathbf{v}_\mathfrak{o}, \mathbf{V}_\mathfrak{g}) & y = \mathbf{u}^\intercal \mathbf{v}_\mathfrak{o} + e_\mathfrak{o} \text{ for some} \\ \mathsf{ict} \leftarrow \mathsf{Enc}(\mathsf{impk}, \mathfrak{o}, \mathbf{u}) & e_\mathfrak{o} \in [-B(\lambda, q, K, Z), B(\lambda, q, K, Z)] \\ y = \mathsf{Dec}(\mathsf{impk}, \mathsf{isk}, \mathsf{ict}, \mathfrak{o}) & \end{array}\right] = 1.$$

For B-correctness for $\mathfrak{g}$, the requirement is

$$\Pr\left[\begin{array}{l|l} (\mathsf{impk}, \mathsf{imsk}) \leftarrow \mathsf{Setup}(1^\lambda, q, 1^K, 1^Z) & \\ \mathsf{isk} \leftarrow \mathsf{KeyGen}(\mathsf{imsk}, \mathbf{v}_\mathfrak{o}, \mathbf{V}_\mathfrak{g}) & \mathbf{y} = \mathbf{u}^\intercal \mathbf{V}_\mathfrak{g} + \tilde{\mathbf{g}}^\intercal \otimes e_\mathfrak{g} + \mathbf{e}_\mathfrak{g}^\intercal \text{ for some} \\ \mathsf{ict} \leftarrow \mathsf{Enc}(\mathsf{impk}, \mathfrak{g}, \mathbf{u}) & e_\mathfrak{g}, \mathbf{e}_\mathfrak{g}^\intercal \in [-B(\lambda, q, K, Z), B(\lambda, q, K, Z)]^* \\ \mathbf{y} = \mathsf{Dec}(\mathsf{impk}, \mathsf{isk}, \mathsf{ict}, \mathfrak{g}) & \end{array}\right] = 1.$$

The scheme is B-correct if both are satisfied.

Definition 2.5 (evIPFE **security).** Let $\mathrm{Samp} = (\mathrm{Samp}_\mathbf{V}, \mathrm{Samp}_\mathbf{U})$ be two algorithms with the following syntax.

$\mathrm{Samp}_\mathbf{V}$: On input 1^λ, and public randomness r_pub, outputs $(q, 1^K, 1^Z, 1^{Q_k}, 1^{Q_{c,\mathfrak{o}}}, 1^{Q_{c,\mathfrak{g}}}, \sigma_\mathsf{pre} \geq 0, \{\mathbf{v}_{i,\mathfrak{o}} \in \mathbb{Z}_q^Z, \mathbf{V}_{i,\mathfrak{g}} \in \mathbb{Z}_q^{Z \times K}\}_{i \in [Q_k]})$.

$\mathrm{Samp}_\mathbf{U}$: On input 1^λ, public randomness r_pub, and private randomness r_priv, outputs $(\{\mathbf{u}_{j_\mathfrak{s}} \in \mathbb{Z}_q^Z\}_{j_\mathfrak{s} \in [Q_{c,\mathfrak{s}}]})$ where $q, Z, Q_{c,\mathfrak{s}}$ for $\mathfrak{s} \in \{\mathfrak{o}, \mathfrak{g}\}$ agree with the output of $\mathrm{Samp}_\mathbf{V}(1^\lambda; r_\mathsf{pub})$.

Suppose $(\mathsf{impk}, \mathsf{imsk}) \leftarrow \mathsf{Setup}(1^\lambda, q, 1^K, 1^Z)$, and for all $i \in [Q_k], j_\mathfrak{s} \in [Q_{c,\mathfrak{s}}]$ and $\mathfrak{s} \in \{\mathfrak{o}, \mathfrak{g}\}$,

$$e_{i,j_\mathfrak{s}} \leftarrow \mathcal{D}_{\mathbb{Z},\sigma_\mathsf{pre}}, \quad \mathbf{e}_{i,j_\mathfrak{g}} \leftarrow \mathcal{D}_{\mathbb{Z},\sigma_\mathsf{pre}}^K, \quad \delta_{i,j_\mathfrak{o}} \leftarrow \mathbb{Z}_q, \quad \delta_{i,j_\mathfrak{g}} \leftarrow \mathbb{Z}_q^K, \quad \mathsf{isk}_i \leftarrow \mathsf{KeyGen}(\mathsf{imsk}, \mathbf{v}_{i,\mathfrak{o}}, \mathbf{V}_{i,\mathfrak{g}})$$
$$\mathsf{ict}_{j_\mathfrak{s}} \leftarrow \mathsf{Enc}(\mathsf{impk}, \mathfrak{s}, \mathbf{u}_{j_\mathfrak{s}}), \quad \mathbf{u}_{j_\mathfrak{s}}^\$ \leftarrow \mathbb{Z}_q^Z, \quad \mathsf{ict}_{j_\mathfrak{s}}^\$ \leftarrow \mathsf{Enc}(\mathsf{impk}, \mathfrak{s}, \mathbf{u}_{j_\mathfrak{s}}^\$)$$

Samp has *pseudorandom structurally noisy inner products*, denoted $\mathsf{IPFEsec}_\mathsf{pre}^\mathrm{Samp}$, if

$$\left\{\left(1^\lambda, r_\mathsf{pub} \quad \left\{\begin{array}{l} \mathbf{u}_{j_\mathfrak{o}}^\intercal \mathbf{v}_{i,\mathfrak{o}} + e_{i,j_\mathfrak{o}}, \\ \mathbf{u}_{j_\mathfrak{g}}^\intercal \mathbf{V}_{i,\mathfrak{g}} + \tilde{\mathbf{g}}^\intercal \otimes e_{i,j_\mathfrak{g}} + \mathbf{e}_{i,j_\mathfrak{g}}^\intercal \end{array}\right\}_{i \in [Q_k], j_\mathfrak{o} \in [Q_{c,\mathfrak{o}}], j_\mathfrak{g} \in [Q_{c,\mathfrak{g}}]}\right)\right\}_{\lambda \in \mathbb{N}}$$
$$\approx \left\{\left(1^\lambda, r_\mathsf{pub} \quad \left\{\begin{array}{l} \delta_{i,j_\mathfrak{o}} \\ \delta_{i,j_\mathfrak{g}}^\intercal \end{array}\right\}_{i \in [Q_k], j_\mathfrak{o} \in [Q_{c,\mathfrak{o}}], j_\mathfrak{g} \in [Q_{c,\mathfrak{g}}]}\right)\right\}_{\lambda \in \mathbb{N}}$$

The scheme is Samp-secure, denoted $\mathsf{IPFEsec}_\mathsf{post}^\mathrm{Samp}$, if

$$(1^\lambda, r_\mathsf{pub}, \mathsf{impk}, \{\mathsf{ict}_{j_\mathfrak{o}}\}_{j_\mathfrak{o} \in [Q_{c,\mathfrak{o}}]}, \{\mathsf{ict}_{j_\mathfrak{g}}\}_{j_\mathfrak{g} \in [Q_{c,\mathfrak{g}}]}, \{\mathsf{isk}_i\}_{i \in [Q_k]})$$
$$\approx (1^\lambda, r_\mathsf{pub}, \mathsf{impk}, \{\mathsf{ict}_{j_\mathfrak{o}}^\$\}_{j_\mathfrak{o} \in [Q_{c,\mathfrak{o}}]}, \{\mathsf{ict}_{j_\mathfrak{g}}^\$\}_{j_\mathfrak{g} \in [Q_{c,\mathfrak{g}}]}, \{\mathsf{isk}_i\}_{i \in [Q_k]})$$

The scheme is $\sigma_{\mathsf{pre},0}$-secure if it is Samp-secure for all efficient Samp such that $\sigma_\mathsf{pre} \leq \sigma_{\mathsf{pre},0}$ and $\mathsf{IPFEsec}_\mathsf{pre}^\mathrm{Samp}$ holds.

2.2.2 Evasive IPFE Assumption.

In this section, we verbatim recall the Evasive IPFE assumption that corresponds to Assumption 5 of [HLL24b].

Assumption 2.6. Let $\mathsf{Samp} = (\mathsf{Samp}_{\mathbf{V}}, \mathsf{Samp}_{\mathbf{U}})$ be two algorithms with the following syntax.

- $\mathsf{Samp}_{\mathbf{V}}(1^\lambda; r_{\mathsf{pub}})$, given randomness r_{pub}, outputs

$$1^n, 1^m, q, 1^Z, 1^J, 1^I, \qquad \sigma_{-1}, \qquad \sigma_{\mathsf{post}} \geq \sigma_{\mathsf{pre}} \geq 0, \qquad \mathbf{V} \in \mathbb{Z}_q^{Z \times J},$$

  where $m \geq m_0(n,q)$ and $\sigma_{-1} \geq \sigma_0(n,m)$ are constrained by Lemma 2.3.

- $\mathsf{Samp}_{\mathbf{U}}(1^\lambda, r_{\mathsf{pub}}; r_{\mathsf{priv}})$, given r_{pub} and additional randomness r_{priv}, outputs $\mathbf{U}^\intercal$, where $\mathbf{U} \in \mathbb{Z}_q^{Z \times I}$ and q, Z, I agree with the output of $\mathsf{Samp}_{\mathbf{V}}(1^\lambda; r_{\mathsf{pub}})$.

Suppose

$$\begin{aligned}
(\mathbf{B}, \tau) &\leftarrow \mathsf{TrapGen}(1^n, 1^m, q), \quad \mathbf{A} \leftarrow \mathbb{Z}_q^{n \times Z\lceil \log_2 q \rceil}, \quad \mathbf{D} \leftarrow \mathbb{Z}_q^{n \times I}, \\
\Delta_0 &\leftarrow \mathbb{Z}_q^{Z\lceil \log_2 q \rceil \times I}, \quad \Delta_1 \leftarrow \mathbb{Z}_q^{m \times I}, \quad \Delta_2 \leftarrow \mathbb{Z}_q^{J \times I}, \\
\mathbf{E}_{\mathsf{pre},0} &\leftarrow \mathcal{D}_{\mathbb{Z},\sigma_{\mathsf{pre}}}^{Z\lceil \log_2 q \rceil \times I}, \quad \mathbf{E}_{\mathsf{pre},1} \leftarrow \mathcal{D}_{\mathbb{Z},\sigma_{\mathsf{pre}}}^{m \times I}, \quad \mathbf{E}_{\mathsf{pre},2} \leftarrow \mathcal{D}_{\mathbb{Z},\sigma_{\mathsf{pre}}}^{J \times I}, \\
\mathbf{E}_{\mathsf{post},0} &\leftarrow \mathcal{D}_{\mathbb{Z},\sigma_{\mathsf{post}}}^{Z\lceil \log_2 q \rceil \times I}, \quad \mathbf{E}_{\mathsf{post},1} \leftarrow \mathcal{D}_{\mathbb{Z},\sigma_{\mathsf{post}}}^{m \times I}, \\
\mathbf{K} &\leftarrow \mathsf{SampleD}(\mathbf{B}, \tau, \mathbf{A}\mathbf{G}^{-1}(\mathbf{V}), \sigma_{-1})
\end{aligned}$$

The *inner-product* precondition $\mathsf{evIPFE}_{\mathsf{ipre}}^{\mathsf{Samp}}$ is

$$\{(1^\lambda, r_{\mathsf{pub}}, \mathbf{U}^\intercal\mathbf{V} + \mathbf{E}_{\mathsf{pre},2}^\intercal)\}_{\lambda \in \mathbb{N}} \approx \{(1^\lambda, r_{\mathsf{pub}}, \Delta_2^\intercal)\}_{\lambda \in \mathbb{N}}.$$

The *stricter* precondition $\mathsf{evIPFE}_{\mathsf{spre}}^{\mathsf{Samp}}$ is

$$\begin{aligned}
&\{(1^\lambda, r_{\mathsf{pub}}, \mathbf{A}, \mathbf{B}, \mathbf{D}^\intercal\mathbf{A}\mathbf{G}^{-1}(\mathbf{V}) + \mathbf{E}_{\mathsf{pre},2}^\intercal, \mathbf{D}^\intercal\mathbf{B} + \mathbf{E}_{\mathsf{pre},1}^\intercal, \mathbf{D}^\intercal\mathbf{A} + \mathbf{E}_{\mathsf{pre},0}^\intercal + \mathbf{U}^\intercal\mathbf{G})\}_{\lambda \in \mathbb{N}} \\
\approx &\{(1^\lambda, r_{\mathsf{pub}}, \mathbf{A}, \mathbf{B}, \quad \Delta_2^\intercal, \quad \Delta_1^\intercal, \quad \Delta_0^\intercal \quad)\}_{\lambda \in \mathbb{N}}.
\end{aligned}$$

The postcondition $\mathsf{evIPFE}_{\mathsf{post}}^{\mathsf{Samp}}$ is

$$\begin{aligned}
&\{(1^\lambda, r_{\mathsf{pub}}, \mathbf{A}, \mathbf{B}, \mathbf{K}, \mathbf{D}^\intercal\mathbf{B} + \mathbf{E}_{\mathsf{post},1}^\intercal, \mathbf{D}^\intercal\mathbf{A} + \mathbf{E}_{\mathsf{post},0}^\intercal + \mathbf{U}^\intercal\mathbf{G})\}_{\lambda \in \mathbb{N}} \\
\approx &\{(1^\lambda, r_{\mathsf{pub}}, \mathbf{A}, \mathbf{B}, \mathbf{K}, \quad \Delta_1^\intercal, \quad \Delta_0^\intercal \quad)\}_{\lambda \in \mathbb{N}}.
\end{aligned}$$

The $(n_0, \sigma_{\mathsf{post},0})$-*evasive IPFE assumption* states that $\mathsf{evIPFE}_{\mathsf{ipre}}^{\mathsf{Samp}}$ implies $\mathsf{evIPFE}_{\mathsf{post}}^{\mathsf{Samp}}$ for all efficient Samp such that $n \geq n_0$ and $\sigma_{\mathsf{post}} \geq \sigma_{\mathsf{post},0}$. The *conservative evasive IPFE assumption* states that $\mathsf{evIPFE}_{\mathsf{spre}}^{\mathsf{Samp}}$ implies $\mathsf{evIPFE}_{\mathsf{post}}^{\mathsf{Samp}}$ for all efficient Samp.

2.2.3 Restricted Versions

In this section, we verbatim recall Restricted evIPFE security and Restricted evIPFE assumption that respectively corresponds to Definition 6 and Assumption 6 of [HLL24b].

Definition 2.7 (Restricted evIPFE security). A sampler $\mathsf{Samp} = (\mathsf{Samp}_{\mathbf{V}}, \mathsf{Samp}_{\mathbf{U}})$ as per Definition 2.5 is restricted if $\mathsf{Samp}_{\mathbf{U}}(1^\lambda, r_{\mathsf{pub}}; r_{\mathsf{priv}})$ works as follows.

1. It uses a fixed deterministic algorithm $\mathsf{Samp}_{\mathbf{A}0}$ to compute

$$(1^{n'}, \{\mathbf{A}_{j_{\mathfrak{s}}}\}_{j_{\mathfrak{s}}\in[Q_{c,\mathfrak{s}}],\mathfrak{s}\in\{\mathfrak{o},\mathfrak{g}\}}) \leftarrow \mathsf{Samp}_{\mathbf{A}0}(1^\lambda, r_{\mathsf{pub}}),$$

where $\mathbf{A}_{j_{\mathfrak{s}}} \in \mathbb{Z}_q^{n'\times Z}$ for all $j_{\mathfrak{s}} \in [Q_{c,\mathfrak{s}}], \mathfrak{s} \in \{\mathfrak{o},\mathfrak{g}\}$ and the values of $q, Z, Q_{c,\mathfrak{s}}$ for $\mathfrak{s} \in \{\mathfrak{o},\mathfrak{g}\}$ agree with the output of $\mathsf{Samp}_{\mathbf{V}}(1^\lambda; r_{\mathsf{pub}})$.

2. It uses r_{priv} to sample $\mathbf{d}_0 \leftarrow \mathbb{Z}_q^{n'}$. The algorithm outputs $(\{\mathbf{u}_{j_{\mathfrak{s}}} \in \mathbb{Z}_q^Z\}_{j_{\mathfrak{s}}\in[Q_{c,\mathfrak{s}}]})$ for $\mathbf{u}_{j_{\mathfrak{s}}}^\intercal = \mathbf{d}_0^\intercal \mathbf{A}_{j_{\mathfrak{s}}}$ for all $j_{\mathfrak{s}} \in [Q_{c,\mathfrak{s}}], \mathfrak{s} \in \{\mathfrak{o},\mathfrak{g}\}$.

The evIPFE (with structured noise) is restricted-$\sigma_{\mathsf{pre},0}$-secure if it is Samp-secure (Definition 2.5) for all efficient restricted Samp such that $\sigma_{\mathsf{pre}} \le \sigma_{\mathsf{pre},0}$ and $\mathsf{IPFEsec}_{\mathsf{pre}}^{\mathsf{Samp}}$ holds.

Assumption 2.8 (Restricted evIPFE assumption). A sampler $\mathsf{Samp} = (\mathsf{Samp}_{\mathbf{V}}, \mathsf{Samp}_{\mathbf{U}})$ per Assumption 2.6 is restricted if $\mathsf{Samp}_{\mathbf{U}}(1^\lambda, r_{\mathsf{pub}}; r_{\mathsf{priv}})$ works as follows.

1. It uses a fixed deterministic algorithm $\mathsf{Samp}_{\mathbf{A}0}$ to compute

$$(1^{n'}, \{\mathbf{A}'_i\}_{i\in[I]}) \leftarrow \mathsf{Samp}_{\mathbf{A}0}(1^\lambda, r_{\mathsf{pub}}),$$

where $\mathbf{A}'_i \in \mathbb{Z}_q^{n'\times Z}$ for all $i \in [I]$ and the values of q, Z, I agree with the output of $\mathsf{Samp}_{\mathbf{V}}(1^\lambda; r_{\mathsf{pub}})$.

2. It uses r_{priv} to sample $\mathbf{d}_0 \leftarrow \mathbb{Z}_q^{n'}$. The algorithm outputs $\mathbf{U}^\intercal$ for $\mathbf{U} = (\mathbf{u}_1, \ldots, \mathbf{u}_I)$ and $\mathbf{u}_i^\intercal = \mathbf{d}_0^\intercal \mathbf{A}'_i$ for all $i \in [I]$.

The restricted (conservative) evIPFE assumption states that $\mathsf{evIPFE}_{\mathsf{ipre}}^{\mathsf{Samp}}$ ($\mathsf{evIPFE}_{\mathsf{spre}}^{\mathsf{Samp}}$) implies $\mathsf{evIPFE}_{\mathsf{post}}^{\mathsf{Samp}}$ for all efficient restricted Samp.

2.3 Witness Encryption with Pseudorandom Ciphertext

Definition 2.9 (Witness Encryption). [GGSW13, BDJ+25] A witness encryption scheme for an NP language $\mathcal{L}$ (with the corresponding witness relation R) consists of the following two polynomial-time algorithms:

$\mathsf{WE.Enc}(1^\lambda, \mathsf{stmt}, \mu) \to \mathsf{wct}$**.** [3] On input security parameter λ, an unbounded-length string stmt and message $\mu \in \{0,1\}$, outputs ciphertext wct.

$\mathsf{WE.Dec}(\mathsf{wit}, \mathsf{wct}) = \mu/\perp$**.** On input an unbounded length string wit and ciphertext wct, it either outputs message μ or the symbol $\perp$.

These algorithms satisfy the following conditions:

- **Correctness.** For any security parameter λ, for any $\mu \in \{0,1\}$, for any $\mathsf{stmt} \in \mathcal{L}$, such that $R(\mathsf{stmt}, \mathcal{L})$ holds i.e. $\mathsf{stmt} \in \mathcal{L}$, we have

$$\Pr\left[\mathsf{WE.Dec}(\mathsf{wit}, \mathsf{WE.Enc}(1^\lambda, \mathsf{stmt}, \mu)) = \mu\right] = 1.$$

- **Soundness.** For any PPT adversary $\mathcal{A}$, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for any $\mathsf{stmt} \notin \mathcal{L}$, we have:

$$\left|\Pr\left[\mathcal{A}(\mathsf{WE.Enc}(1^\lambda, \mathsf{stmt}, 0)) = 1\right] - \Pr\left[\mathcal{A}(\mathsf{WE.Enc}(1^\lambda, \mathsf{stmt}, 1)) = 1\right]\right| \le \mathsf{negl}(\lambda).$$

[3] In the main body, we drop the first input and implicitly assume 1^λ as an input.

- **Pseudorandom Ciphertext.** For any PPT adversary $\mathcal{A}$, there exists a negligible function $\mathsf{negl}(\cdot)$ such that for any $\mathsf{stmt} \notin \mathcal{L}$ and any $\mu \in \{0,1\}$, we have:

$$\left|\Pr\left[\mathcal{A}(\mathsf{WE.Enc}(1^\lambda, \mathsf{stmt}, \mu)) = 1\right] - \Pr[\mathcal{A}(\mathcal{U}) = 1]\right| \leq \mathsf{negl}(\lambda)$$

  where $\mathcal{U}$ is sampled uniformly from the ciphertext space.

Lemma 2.10. The pseudorandomness of WE ciphertexts can be achieved by combining (plain) WE with compute-and-compare/lockable obfuscation, which in turn is implied by LWE [WZ17, GKW17].

3 Counterexample against Evasive Inner-Product Functional Encryption

Theorem 3.1. Assuming the existence of WE and that the sub-exponential LWE assumption holds, there does not exist any Evasive IPFE without structured noise satisfying evIPFE security (Definition 2.5), for all λ.

Proof. To prove the above theorem, we show that for any evIPFE, there exist two efficient samplers $\mathrm{Samp}_{\mathbf{V}}$ and $\mathrm{Samp}_{\mathbf{U}}$ as defined below:

$\mathrm{Samp}_{\mathbf{V}}$: On input 1^λ, and public randomness r_{pub}, outputs $(q, 1^{Z'}, 1^{Q'_k}, 1^{Q'_c}, \sigma_{\mathsf{pre}} \geq 0, \{\tilde{\mathbf{v}}_i \in \mathbb{Z}_q^{Z'}\}_{i\in[Q'_k]})$.

$\mathrm{Samp}_{\mathbf{U}}$: On input 1^λ, public randomness r_{pub}, and private randomness r_{priv}, outputs $(\{\tilde{\mathbf{u}}_j \in \mathbb{Z}_q^{Z'}\}_{j\in[Q'_c]})$ where q, Z', Q'_c agree with the output of $\mathrm{Samp}_{\mathbf{V}}(1^\lambda; r_{\mathsf{pub}})$.

such that

1. (Pre-condition holds) For all PPT adversary $\mathcal{A}$,

$$\begin{aligned} D_0^{\mathsf{pre}} &:= (1^\lambda, r_{\mathsf{pub}}, \{\tilde{\mathbf{u}}_j^\mathsf{T}\tilde{\mathbf{v}}_i + \tilde{e}_{i,j}\}_{i\in[Q'_k], j\in[Q'_c]}) \\ \approx_c D_1^{\mathsf{pre}} &:= (1^\lambda, r_{\mathsf{pub}}, \{\delta_{i,j} \leftarrow \mathbb{Z}_q\}_{i\in[Q'_k], j\in[Q'_c]}) \end{aligned}$$

   where $\{\tilde{e}_{i,j} \leftarrow \mathcal{D}_{\mathbb{Z},\sigma_{\mathsf{pre}}}\}_{i\in[Q'_k], j\in[Q'_c]}$.

2. (Distinguishability of the post condition) There exists an efficient distinguisher for the following two distributions

$$\begin{aligned} D_0^{\mathsf{post}} &:= (1^\lambda, r_{\mathsf{pub}}, \mathsf{impk}, \{\mathsf{ict}_j\}_{j\in[Q'_c]}, \{\mathsf{isk}_i\}_{i\in[Q'_k]}), \\ D_1^{\mathsf{post}} &:= (1^\lambda, r_{\mathsf{pub}}, \mathsf{impk}, \{\mathsf{ict}_j^\$\}_{j\in[Q'_c]}, \{\mathsf{isk}_i\}_{i\in[Q'_k]}), \end{aligned}$$

   where $(\mathsf{impk}, \mathsf{imsk}) \leftarrow \mathsf{Setup}(1^\lambda, q, 1^{Z'})$, and for all $i \in [Q'_k], j \in [Q'_c]$, $\mathsf{isk}_i \leftarrow \mathsf{KeyGen}(\mathsf{imsk}, \tilde{\mathbf{v}}_i)$, $\mathsf{ict}_j \leftarrow \mathsf{Enc}(\mathsf{impk}, \tilde{\mathbf{u}}_j)$, $\tilde{\mathbf{u}}_j^\$ \leftarrow \mathbb{Z}_q^{Z'}$, and $\mathsf{ict}_j^\$ \leftarrow \mathsf{Enc}(\mathsf{impk}, \tilde{\mathbf{u}}_j^\$)$.

Defining the sampler. We first describe the ingredients needed to define our sampler.

- Let the sizes of any ciphertext and that of any secret key of evIPFE be bounded by polynomials $p_c = p_c(\lambda, Z')$ and $p_k = p_k(\lambda, Z')$, respectively, where Z' is the length of input vectors.
- Let Q_k and Q_c be such that

$$(\log(q/(2(\sigma'_{\mathsf{pre}} + B) + 1)))Q_kQ_c > p_cQ_c + p_kQ_k + \lambda + 1. \tag{1}$$

- Let $\mathsf{prm} = (\lambda, Z', p_c, p_k, Q_c, Q_k, \sigma_{\mathsf{pre}}, \sigma'_{\mathsf{pre}}, B)$, where $B = B(\lambda, q, Z')$ is the error bound due to evasive IPFE decryption and σ'_{pre} is the standard deviation of the error in the noisy inner product term and is such that $\sigma_{\mathsf{pre}} \geq |e|\lambda^{\omega(1)}$ where $e \leftarrow \mathcal{D}_{\mathbb{Z},\sigma'_{\mathsf{pre}}}$. Define the following NP language:

$$\mathcal{L}_{\mathsf{prm}} = \left\{ \begin{array}{l} \{y_{i,j} \in \mathbb{Z}_q\}_{i\in[Q_k],j\in[Q_c]} \mid \exists\ \{\mathsf{ict}_j\}_{j\in[Q_c]}, \{\mathsf{isk}_i\}_{i\in[Q_k]} \\ s.t. \forall\ i \in [Q_k], j \in [Q_c], \quad |y_{i,j} - \mathsf{Dec}(\mathsf{impk}, \mathsf{isk}_i, \mathsf{ict}_j)| \leq \sigma'_{\mathsf{pre}} + B \\ \text{and } |\mathsf{ict}_j| \leq p_c \text{ and } |\mathsf{isk}_i| \leq p_k \end{array} \right\} \tag{2}$$

  We assume that ict_j's contain the public key impk.

- Let WE be a witness encryption scheme for the language defined above in equation (2), wrt the above evIPFE. Thus, the witnesses are of size at most $Q_c p_c + Q_k p_k$. Let $p_w = p_w(Q_c p_c + Q_k p_k)$ be the bound on the length of the ciphertexts of WE. Thus, $p_w > Q_c, Q_k$.

  The relation between Z, Z'; Q_c, Q'_c and Q_k, Q'_k is defined while defining the samplers.

We are now ready to define our samplers, $\mathrm{Samp}_{\mathbf{V}}$ and $\mathrm{Samp}_{\mathbf{U}}$. For any $Z \in \mathbb{N}$, we define $\mathrm{Samp}_{\mathbf{V}}[Q_k, Q_c](1^\lambda; r_{\mathsf{pub}} = ((\mathbf{v}_1, \ldots, \mathbf{v}_{Q_k} \in \mathbb{Z}_q^Z), r'_{\mathsf{pub}}))$:

- Set $Z' = Z + 1$
- Let $Q_w := 2p_w \log q$. Also, since $p_w = \mathrm{poly}(\lambda, \log q, Q_c, Q_k)$, we assume $Q_w > Q_c$. In the otherwise case, we could set Q_w to be the maximum of Q_c and $2p_w \log q$.
- Set $Q'_k := Q_k + 1$, $Q'_c := Q_c + Q_w$.
- Parse r_{pub} as $((\mathbf{v}_1, \ldots, \mathbf{v}_{Q_k}), r'_{\mathsf{pub}})$ and set $\tilde{\mathbf{v}}_i := (\mathbf{v}_i, 0)^\intercal \in \mathbb{Z}_q^{Z+1}$ for all $i \in [Q_k]$.
- Set $\tilde{\mathbf{v}}_{Q_k+1} := (0, 0, \ldots, 0, 1)^\intercal \in \mathbb{Z}_q^{Z+1}$.
- Output $(q, 1^{Z'}, 1^{Q'_k}, 1^{Q'_c}, \sigma_{\mathsf{pre}} \geq 0, \sigma'_{\mathsf{pre}} \geq 0, \{\tilde{\mathbf{v}}_i\}_{i\in[Q'_k]})$. Here, q is set such that q/B is exponentially large.

$\mathrm{Samp}_{\mathbf{U}}[Q_k, Q_c](1^\lambda, r_{\mathsf{pub}}; r_{\mathsf{priv}})$:

- Compute $q, Z', Q'_k, Q_w, Q'_c, \sigma'_{\mathsf{pre}}$ such that they agree with $\mathrm{Samp}_{\mathbf{V}}[Q_k, Q_c](1^\lambda, r_{\mathsf{pub}})$.
- For $i \in [Q_k]$, compute $\mathbf{v}_i$ from r_{pub} in the same way as in $\mathrm{Samp}_{\mathbf{V}}$.
- For $j' \in [Q_c + Q_w]$, sample $\mathbf{u}_{j'} \leftarrow \mathbb{Z}_q^Z$.
- Set $\mathsf{stmt} = (\{\mathbf{u}_j^\intercal \mathbf{v}_i + e_{i,j}\}_{i\in[Q_k],j\in[Q_c]})$ where $\{e_{i,j} \leftarrow \mathcal{D}_{\mathbb{Z},\sigma'_{\mathsf{pre}}}\}_{i\in[Q_k],j\in[Q_c]}$.
- Compute $\mathsf{wct}_0 = \mathsf{WE.Enc}(\mathsf{stmt}, 0)$ and $\mathsf{wct}_1 = \mathsf{WE.Enc}(\mathsf{stmt}, 1)$.
  Set $(\mathsf{w}_1, \ldots, \mathsf{w}_{Q_w}) = \mathsf{bits}(\mathsf{wct}_0, \mathsf{wct}_1)$[4].
- For $\ell \in [Q_w]$, sample $z_\ell \leftarrow [-q/4 + B, q/4 - B]$.
- For $j \in [Q_c]$, sample $r_j \leftarrow [-q/2 + B, q/2 - B]$.
- Set $c_j = r_j$ for $j \in [Q_c]$ and $c_{Q_c+\ell} = \mathsf{w}_\ell \lfloor q/2 \rceil + z_\ell$ for $\ell \in [Q_w]$.
- Output $\{\tilde{\mathbf{u}}_{j'} = (\mathbf{u}_{j'}, c_{j'})^\intercal \in \mathbb{Z}_q^{Z+1}\}_{j'\in[Q_c+Q_w]}$.

[4]If length of wct_0 or wct_1 is less than p_w, we set the remaining w's as 0.

Parameter Constraints. We list the constraints on the parameters.

- q/B is exponentially large.
- $\sigma_{\mathsf{pre}} \geq |e|\lambda^{\omega(1)}$ where $e \leftarrow \mathcal{D}_{\mathbb{Z},\sigma'_{\mathsf{pre}}}$.
- Q_k and Q_c are such that $(\log(q/(2(\sigma'_{\mathsf{pre}}+B)+1)))Q_kQ_c > p_cQ_c + p_kQ_k + \lambda + 1$.

Distinguisher's strategy for post condition. Note that given r_{pub}, $\{\tilde{\mathbf{v}}_i\}_{i\in[Q_k+1]}$ can be computed efficiently. Hence, it suffices to prove that there exists a distinguisher, $\mathcal{A}^{\mathsf{post}}$, for

$$D_0^{\mathsf{post}} := (1^\lambda, \{\tilde{\mathbf{v}}_i\}_{i\in[Q_k+1]}, \mathsf{impk}, \{\mathsf{ict}_{j'} \leftarrow \mathsf{Enc}(\mathsf{impk}, \tilde{\mathbf{u}}_j)\}_{j'\in[Q_c+Q_w]}, \{\mathsf{isk}_i\}_{i\in[Q_k+1]})$$
$$D_1^{\mathsf{post}} := (1^\lambda, \{\tilde{\mathbf{v}}_i\}_{i\in[Q_k+1]}, \mathsf{impk}, \{\mathsf{ict}_{j'} \leftarrow \mathsf{Enc}(\mathsf{impk}, \tilde{\mathbf{u}}^{\$}_j)\}_{j'\in[Q_c+Q_w]}, \{\mathsf{isk}_i\}_{i\in[Q_k+1]}),$$

where, for all $j' \in [Q_c+Q_w]$, $\tilde{\mathbf{u}}^{\$}_{j'} \leftarrow \mathbb{Z}_q^{Z+1}$. $\mathcal{A}^{\mathsf{post}}$ does the following:

1. For $\ell \in [Q_w]$, compute $\widetilde{\mathsf{w}}_\ell = \mathsf{Dec}(\mathsf{impk}, \mathsf{isk}_{Q_k+1}, \mathsf{ict}_{Q_c+\ell})$.
2. For $\ell \in [Q_w]$, set $\mathsf{w}'_\ell = 0$ if $|\widetilde{\mathsf{w}}_\ell| \leq q/4$, else set $\mathsf{w}'_\ell = 1$.
3. Set $\widetilde{\mathsf{wct}} = \mathsf{PowerofTwo}(\mathsf{w}'_1, \ldots, \mathsf{w}'_{Q_w})$ and parse it as $\widetilde{\mathsf{wct}} = (\widetilde{\mathsf{wct}}_0, \widetilde{\mathsf{wct}}_1)$.
4. Set witness $\widetilde{\mathsf{wit}} = (\mathsf{ict}_1, \ldots, \mathsf{ict}_{Q_c}, \mathsf{isk}_1, \ldots, \mathsf{isk}_{Q_k})$.
5. Compute $\mu_0 = \mathsf{WE.Dec}(\widetilde{\mathsf{wit}}, \widetilde{\mathsf{wct}}_0)$ and $\mu_1 = \mathsf{WE.Dec}(\widetilde{\mathsf{wit}}, \widetilde{\mathsf{wct}}_1)$.
6. If $(\mu_0, \mu_1) = (0,1)$, output $\beta' = 0$,
   Else, if any of the above steps fail, or if $(\mu_0, \mu_1) \neq (0,1)$, output $\beta' = 1$.

Analyzing the distinguisher $\mathcal{A}^{\mathsf{post}}$. When $\beta = 0$, it is easy to see by inspection, that $\mathcal{A}^{\mathsf{post}}$ gets $(\mu_0, \mu_1) = (0,1)$ in Step 5. In particular, firstly, we observe that $\mathcal{A}^{\mathsf{post}}$ correctly recovers WE encryptions $\widetilde{\mathsf{w}}_\ell = \mathsf{w}'_\ell \lfloor q/2 \rceil + z_\ell + e'_{Q_k+1,Q_c+\ell}$ with $|e'_{Q_k+1,Q_c+\ell}| < B$, due to the B-correctness of evIPFE. Further, since, $|e'_{Q_k+1,Q_c+\ell}| < B$, $|z_\ell + e'_{Q_k+1,Q_c+\ell}| \leq [-q/4, q/4]$. Hence, $\mathcal{A}$ correctly recovers w'_ℓ. Next, we observe that $\widetilde{\mathsf{wit}}$ formed by first Q_c ciphertexts and Q_k keys indeed form a valid witness since for all $i \in [Q_k], j \in [Q_c]$, $|e_{i,j} - e'_{i,j}| \leq \sigma'_{\mathsf{pre}} + B$. Hence, by correctness of WE, $\mathsf{WE.Dec}(\widetilde{\mathsf{wit}}, \widetilde{\mathsf{wct}}_0) = 0$ and $\mathsf{WE.Dec}(\widetilde{\mathsf{wit}}, \widetilde{\mathsf{wct}}_1) = 1$. Hence, it outputs $\beta' = 0 = \beta$ with probability close to 1.

When $\beta = 1$, then the probability that $(\mu_0, \mu_1) = (0,1) \leq 1/4$. This is because, for all $\ell \in [Q_w]$, $\mathsf{ict}_{Q_c+\ell}$ encodes a uniformly random vector $\tilde{\mathbf{u}}^{\$}_{Q_c+\ell}$. Parse $\tilde{\mathbf{u}}^{\$}_{Q_c+\ell}$ as $(\mathbf{u}^{\$}_{Q_c+\ell}, \tilde{\mathsf{w}}_\ell)$. Now when $\mathsf{ict}_{Q_c+\ell}$ is decrypted using isk_{Q_k+1} in Step 1 , it will give some random $\tilde{\mathsf{w}}_\ell$ which in turn gives a random bit w'_ℓ. Thus, $\mathcal{A}$ obtains $\widetilde{\mathsf{wct}}_0$ and $\widetilde{\mathsf{wct}}_1$ as uniformly random WE ciphertexts. Hence, for any (valid or invalid) witness wit, $\Pr\left[\mathsf{WE.Dec}(\widetilde{\mathsf{wit}}, \widetilde{\mathsf{wct}}_0) = 0\right] = \Pr\left[\mathsf{WE.Dec}(\widetilde{\mathsf{wit}}, \widetilde{\mathsf{wct}}_1) = 0\right] = \epsilon$ for some $0 \leq \epsilon \leq 1$. Hence, $\Pr[(\mu_0, \mu_1) = (0,1)] \leq \epsilon(1-\epsilon) \leq 1/4$. This is because $\widetilde{\mathsf{wit}}$ and $(\widetilde{\mathsf{wct}}_0, \widetilde{\mathsf{wct}}_1)$ are independent. Hence, $\Pr[\beta' = 1 | \beta = 1] \geq 3/4$. Hence, the overall success probability of the adversary is close to at least $7/8$.

Analyzing the feasibility of equation (1). We note that Z' is independent of Q_k and Q_c and hence, both p_c and p_k (which depend on Z') are independent of both Q_c and Q_k. Hence, it is possible to set the values of Q_c and Q_k such that the equation (1) is satisfied.

Analyzing the pre-condition We prove the following claim.

Claim 1. Let there exist a WE with pseudorandom ciphertexts and let the LWE assumption hold. For samplers $(q, 1^{Z+1}, 1^{Q_k+1}, 1^{Q_c+Q_w}, \sigma_{\mathsf{pre}} \geq 0, \sigma'_{\mathsf{pre}} \geq 0, \{\tilde{\mathbf{v}}_i\}_{i\in[Q_k+1]}) \leftarrow \mathsf{Samp}_{\mathbf{V}}(1^\lambda; r_{\mathsf{pub}})$ and $\{\tilde{\mathbf{u}}_{j'} = (\mathbf{u}_{j'}, c_{j'})^\intercal\}_{j'\in[Q_c+Q_w]} \leftarrow \mathsf{Samp}_{\mathbf{U}}(1^\lambda, r_{\mathsf{pub}}; r_{\mathsf{priv}})$ as defined above, we have $D_0^{\mathsf{pre}} \approx_c D_1^{\mathsf{pre}}$ for $\sigma_{\mathsf{pre}} \geq |e|\lambda^{\omega(1)}$ where $e \leftarrow \mathcal{D}_{\mathbb{Z},\sigma'_{\mathsf{pre}}}$, and D_0^{pre} and D_1^{pre} are as defined below.

$$D_0^{\mathsf{pre}} := \left(\begin{array}{c} 1^\lambda,\ \ r_{\mathsf{pub}} = (\{\mathbf{v}_i\}_{i\in[Q_k]}, r'_{\mathsf{pub}}), \\ \{\mathbf{u}_{j'}{}^\intercal \mathbf{v}_i + \tilde{e}_{i,j'}\}_{i\in[Q_k],j'\in[Q_c+Q_w]},\ \{c_{j'} + \tilde{e}_{Q_k+1,j'}\}_{j'\in[Q_c+Q_w]} \end{array}\right)$$

$$\approx_c D_1^{\mathsf{pre}} := \left(\begin{array}{c} 1^\lambda,\ \ r_{\mathsf{pub}} = (\{\mathbf{v}_i\}_{i\in[Q_k]}, r'_{\mathsf{pub}}), \\ \{\delta_{i,j'} \leftarrow \mathbb{Z}_q\}_{i\in[Q_k],j'\in[Q_c+Q_w]},\quad \{\delta'_{Q_k+1,j'} \leftarrow \mathbb{Z}_q\}_{j'\in[Q_c+Q_w]} \end{array}\right).$$

Proof. We prove the claim via the following hybrids.

Hyb_0**:** This is D_0^{pre}. Note that here w_ℓ in $c_{Q_c+1}, \ldots, c_{Q_c+Q_w}$ is the ℓ-th bit of witness ciphertext (in bits) encoded with respect to $\mathsf{stmt} = (\{\mathbf{u}_j^\intercal \mathbf{v}_i + \tilde{e}_{i,j}\}_{i\in[Q_k],j\in[Q_c]})$.

Hyb_1**:** In this hybrid, for all $i \in [Q_k], j' \in [Q_c+Q_w]$, we change $\mathbf{u}_{j'}^\intercal \mathbf{v}_i + e_{i,j'}$ to $\mathbf{u}_{j'}^\intercal \mathbf{v}_i + \tilde{e}_{i,j'} + e_{i,j'}$ where for $i \in [Q_k], j' \in \{Q_c+1, \ldots, Q_c+Q_w\}$ we sample $\tilde{e}_{i,j'} \leftarrow \mathcal{D}_{\mathbb{Z},\sigma'_{\mathsf{pre}}}$ fresh, and for $i \in [Q_k], j \in [Q_c]$, we reuse the same $\tilde{e}_{i,j}$ used to compute stmt. This hybrid is indistinguishable from the previous hybrid by Lemma 2.1 (Smudging Lemma) as $\sigma_{\mathsf{pre}} \geq |e|\lambda^{\omega(1)}$ where $e \leftarrow \mathcal{D}_{\mathbb{Z},\sigma'_{\mathsf{pre}}}$.

Hyb_2**:** In this hybrid, for all $i \in [Q_k], j' \in [Q_c+Q_w]$, we change $\mathbf{u}_{j'}^\intercal \mathbf{v}_i + \tilde{e}_{i,j'} + e_{i,j'}$ to $y_{i,j'} \leftarrow \mathbb{Z}_q$. The indistinguishability follows from the hardness of LWE and from the fact that anything added to random is still random. Note that this also changes the computation of wct_0 and wct_1, which are now computed with respect to the statement formed by these (random) $y_{i,j}$'s for $i \in [Q_k], j \in [Q_c]$. Hence, we need to show that the following distribution is indistinguishable from D_1^{pre}.

$$(1^\lambda, r_{\mathsf{pub}} = (\{\mathbf{v}_i\}_{i\in[Q_k]}, r'_{\mathsf{pub}}), \{y_{i,j'} \leftarrow \mathbb{Z}_q\}_{i\in[Q_k],j'\in[Q_c+Q_w]}, \{c_{j'} + e_{Q_k+1,j'}\}_{j'\in[Q_c+Q_w]})$$

Hyb_3**:** In this hybrid, we change $\{c_{j'} + e_{Q_k+1,j'}\}_{j'\in[Q_c+Q_w]}$ to $\{\delta'_{Q_k+1,j'} \leftarrow \mathbb{Z}_q\}_{j'\in[Q_c+Q_w]}$. For $j \in [Q_c]$, we have $c_j + e_{Q_k+1,j} = r_j + e_{Q_k+1,j}$, which is already uniform over $\mathbb{Z}_q$ since r_j is uniform in $[-q/2+B, q/2-B]$ and independent of all other values and q/B is exponentially large. Hence, it suffices to argue about $j' = Q_c + \ell$ for $\ell \in [Q_w]$, for which $c_{Q_c+\ell} + e_{Q_k+1,Q_c+\ell} = \mathsf{w}_\ell \lfloor q/2 \rceil + z_\ell + e_{Q_k+1,Q_c+\ell}$. In particular, the challenger samples $\mathsf{w}_\ell \leftarrow \{0,1\}$ and $z_\ell \leftarrow [-q/4+B, q/4-B]$ for all $\ell \in [Q_w]$.

At first, note that for a randomly chosen set of $\{y_{i,j}\}_{i\in[Q_k],j\in[Q_c]}$, the probability of this being in $\mathcal{L}_{\mathsf{prm}}$ is negligible. This follows from the standard counting argument as follows: since any witness can be of length at most $Q_c p_c + Q_k p_k$, total number of possible witness is at most $2^{Q_c p_c + Q_k p_k + 1}$. We further observe that each witness can serve for at most $(2(\sigma'_{\mathsf{pre}} + B) + 1)^{Q_c Q_k}$ statements, due to the bounded decryption-error permitted in the description of $\mathcal{L}_{\mathsf{prm}}$ in (2). Thus $|\mathcal{L}_{\mathsf{prm}}| \leq 2^{Q_c p_c + Q_k p_k}(2(\sigma'_{\mathsf{pre}} + B) + 1)^{Q_c Q_k}$. On the other hand, total number of choices for $\{y_{i,j} \in \mathbb{Z}_q\}_{i\in[Q_k],j\in[Q_c]}$ is $2^{(\log q) Q_k Q_c}$. Hence, by the choice of the parameters, for a randomly chosen set of $\{y_{i,j}\}_{i,j}$, this being inside the language is $\leq 2^{-\lambda}$. Hence, the indistinguishability follows due to the pseudorandomness of WE ciphertexts, which can be obtained from plain WE and LWE (Lemma 2.10), z_ℓ is uniformly sampled from $[-q/4+B, q/4-B]$, and by the fact that B is exponentially smaller than $q/4$.

This completes the proof of Claim 1. □

This completes the proof of Theorem 3.1. □

We obtain the following implications as corollaries.

Corollary 3.2. Assuming the existence of a WE, and that the sub-exponential LWE assumption holds, the evasive IPFE assumption (conservative and non-conservative) as defined in Assumption 2.6 is false.

Proof. This follows directly from Theorem 5 and Corollary 7 in [HLL24a], which state that evasive IPFE assumptions (conservative as well as non-conservative) as defined in Assumption 2.6 imply evasive IPFE without structured noise, for which we show the impossibility above. □

Corollary 3.3. Assuming the existence of WE, and that sub-exponential LWE assumption holds, there does not exist any Evasive IPFE scheme with structured noise satisfying evIPFE security (Definition 2.5), for all λ.

Proof. Since the version of Evasive IPFE with structured noise is stronger than the version without, our impossibility also applies to this version. □

Remark 3.4 (Difficulty in attacking restricted evIPFE.). The attack above relies on embedding witness encryption ciphertexts in the private coins of $\mathsf{Samp}_{\mathbf{U}}$. In the restricted version of evIPFE, there are strict restrictions on the distribution on the private coins of $\mathsf{Samp}_{\mathbf{U}}$ which makes it impossible to embed witness encryption ciphertexts as required by the attack.

Acknowledgments. The first and the second authors are supported by the Arvind Raghunathan Center for the Theory of Computing (aRtCS), the Mphasis F1 Foundation, and the National Quantum Mission initiative of the Department of Science and Technology, India. Shota Yamada is supported by JPMJCR22M1 and JPMJKP24U3.

References

[AKY24] Shweta Agrawal, Simran Kumari, and Shota Yamada. Attribute Based Encryption for Turing Machines from Lattices. In *Crypto*, 2024. (Cited on page 1.)

[AKY25] Shweta Agrawal, Simran Kumari, and Shota Yamada. Pseudorandom fe, io and applications. In *Theory of Cryptography: 23rd International Conference, TCC 2025, Aarhus, Denmark, December 1–5, 2025, Proceedings, Part II*, page 222–258, Berlin, Heidelberg, 2025. Springer-Verlag. (Cited on page 1, 4.)

[AKYY23] Shweta Agrawal, Simran Kumari, Anshu Yadav, and Shota Yamada. Broadcast, trace and revoke with optimal parameters from polynomial hardness. In Carmit Hazay and Martijn Stam, editors, *EUROCRYPT 2023, Part III*, volume 14006 of *LNCS*, pages 605–636. Springer, Heidelberg, April 2023. (Cited on page 1.)

[AMYY25] Shweta Agrawal, Anuja Modi, Anshu Yadav, and Shota Yamada. Zeroizing attacks against evasive & circular evasive lwe. In *Theory of Cryptography: 23rd International Conference, TCC 2025, Aarhus, Denmark, December 1–5, 2025, Proceedings, Part II*, page 259–290, Berlin, Heidelberg, 2025. Springer-Verlag. (Cited on page 2, 4.)

[ARYY23] Shweta Agrawal, Mélissa Rossi, Anshu Yadav, and Shota Yamada. Constant input attribute based (and predicate) encryption from evasive and tensor LWE. In *CRYPTO 2023, Part IV*, LNCS, pages 532–564. Springer, Heidelberg, August 2023. (Cited on page 1.)

[BDJ+25] Pedro Branco, Nico Döttling, Abhishek Jain, Giulio Malavolta, Surya Mathialagan, Spencer Peters, and Vinod Vaikuntanathan. Pseudorandom obfuscation and applications. In Yael Tauman Kalai and Seny F. Kamara, editors, *Advances in Cryptology – CRYPTO 2025*, pages 663–698, Cham, 2025. Springer Nature Switzerland. (Cited on page 1, 2, 4, 6, 11.)

[BUW24] Chris Brzuska, Akin Ünal, and Ivy K. Y. Woo. Evasive lwe assumptions: Definitions, classes, and counterexamples. In *Advances in Cryptology – ASIACRYPT 2024: 30th International Conference on the Theory and Application of Cryptology and Information Security, Kolkata, India, December 9–13, 2024, Proceedings. Part IV*, page 418–449, Berlin, Heidelberg, 2024. Springer-Verlag. (Cited on page 2.)

[BW19] Ward Beullens and Hoeteck Wee. Obfuscating simple functionalities from knowledge assumptions. In Dongdai Lin and Kazue Sako, editors, *Public-Key Cryptography – PKC 2019*, pages 254–283, Cham, 2019. Springer International Publishing. (Cited on page 2.)

[CW25] Valerio Cini and Hoeteck Wee. Faster abe for turing machines from circular evasive lwe. In *EUROCRYPT*, page 94–125, Berlin, Heidelberg, 2025. Springer-Verlag. (Cited on page 2.)

[DJM+25] Nico Döttling, Abhishek Jain, Giulio Malavolta, Surya Mathialagan, and Vinod Vaikuntanathan. Simple and general counterexamples for private-coin evasive lwe. In Yael Tauman Kalai and Seny F. Kamara, editors, *Advances in Cryptology – CRYPTO 2025*, pages 73–92, Cham, 2025. Springer Nature Switzerland. (Cited on page 2.)

[GGSW13] Sanjam Garg, Craig Gentry, Amit Sahai, and Brent Waters. Witness encryption and its applications. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 467–476, 2013. (Cited on page 4, 11.)

[GKW17] Rishab Goyal, Venkata Koppula, and Brent Waters. Lockable obfuscation. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 612–621, 2017. (Cited on page 12.)

[HHY26] Tzu-Hsiang Huang, Wei-Hsiang Hung, and Shota Yamada. A note on obfuscation-based attacks on private-coin evasive lwe. *IACR Communications in Cryptology*, 2(4):20, 2026. (Cited on page 2.)

[HJL25] Yao-Ching Hsieh, Aayush Jain, and Huijia Lin. Lattice-based post-quantum io from circular security with random opening assumption. In Yael Tauman Kalai and Seny F. Kamara, editors, *Advances in Cryptology – CRYPTO 2025*, pages 3–38, Cham, 2025. Springer Nature Switzerland. (Cited on page 2.)

[HLL23] Yao-Ching Hsieh, Huijia Lin, and Ji Luo. Attribute-based encryption for circuits of unbounded depth from lattices. In *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 415–434. IEEE, 2023. (Cited on page 1, 2.)

[HLL24a] Yao-Ching Hsieh, Huijia Lin, and Ji Luo. A general framework for lattice-based abe using evasive inner-product functional encryption. In Marc Joye and Gregor Leander, editors, *Advances in Cryptology – EUROCRYPT 2024*, pages 433–464, Cham, 2024. Springer Nature Switzerland. (Cited on page 2, 3, 8, 16.)

[HLL24b] Yao-Ching Hsieh, Huijia Lin, and Ji Luo. A general framework for lattice-based ABE using evasive inner-product functional encryption. Cryptology ePrint Archive, Paper 2024/821, 2024. (Cited on page 4, 8, 10.)

[MP12] Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In *Eurocrypt*, 2012. (Cited on page 8.)

[MPV24] Surya Mathialagan, Spencer Peters, and Vinod Vaikuntanathan. Adaptively sound zero-knowledge snarks for up. In *Annual International Cryptology Conference*, pages 38–71. Springer, 2024. (Cited on page 1.)

[Reg09] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM*, 56(6):34:1–34:40, 2009. (Cited on page 8.)

[Tsa22] Rotem Tsabary. Candidate witness encryption from lattice techniques. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part I*, volume 13507 of *LNCS*, pages 535–559. Springer, Heidelberg, August 2022. (Cited on page 1.)

[VWW22] Vinod Vaikuntanathan, Hoeteck Wee, and Daniel Wichs. Witness encryption and null-IO from evasive LWE. In Shweta Agrawal and Dongdai Lin, editors, *ASIACRYPT 2022, Part I*, volume 13791 of *LNCS*, pages 195–221. Springer, Heidelberg, December 2022. (Cited on page 1.)

[Wee22] Hoeteck Wee. Optimal broadcast encryption and CP-ABE from evasive lattice assumptions. In Orr Dunkelman and Stefan Dziembowski, editors, *EUROCRYPT 2022, Part II*, volume 13276 of *LNCS*, pages 217–241. Springer, Heidelberg, May / June 2022. (Cited on page 1.)

[WW24] Brent Waters and Daniel Wichs. Adaptively secure attribute-based encryption from witness encryption. In *Theory of Cryptography Conference*, pages 65–90. Springer, 2024. (Cited on page 1.)

[WWW22] Brent Waters, Hoeteck Wee, and David J. Wu. Multi-authority ABE from lattices without random oracles. In Eike Kiltz and Vinod Vaikuntanathan, editors, *TCC 2022, Part I*, volume 13747 of *LNCS*, pages 651–679. Springer, Heidelberg, November 2022. (Cited on page 1, 8.)

[WZ17] Daniel Wichs and Giorgos Zirdelis. Obfuscating compute-and-compare programs under lwe. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 600–611, 2017. (Cited on page 12.)