Lattice-Based Post-Quantum iO from Circular Security with Random Opening Assumption

(Part II: zeroizing attacks against private-coin evasive LWE assumptions)

Yao-Ching Hsieh[*] Aayush Jain [†] Huijia Lin[‡]

Abstract

Indistinguishability obfuscation (iO) stands out as a powerful cryptographic primitive but remains notoriously difficult to realize under simple-to-state, post-quantum assumptions. Recent works have proposed lattice-inspired iO constructions backed by new "LWE-with-hints" assumptions, which posit that certain distributions of LWE samples retain security despite auxiliary information. However, subsequent cryptanalysis has revealed structural vulnerabilities in these assumptions, leaving us without any post-quantum iO candidates supported by simple, unbroken assumptions.

Motivated by these proposals, we introduce the *Circular Security with Random Opening* (CRO) assumption—a new LWE-with-hint assumption that addresses structural weaknesses from prior assumptions, and based on our systematic examination, does not appear vulnerable to known cryptanalytic techniques. In CRO, the hints are random "openings" of zero-encryptions under the Gentry–Sahai–Waters (GSW) homomorphic encryption scheme. Crucially, these zero-encryptions are efficiently derived from the original LWE samples via a special, carefully designed procedure, ensuring that the openings are marginally random. Moreover, the openings do not induce any natural leakage on the LWE noises. These two features—*marginally random hints and the absence of (natural) noise leakage*—rule out important classes of attacks that had undermined all previous LWE-with-hint assumptions for iO. Therefore, our new lattice-based assumption for iO provides a qualitatively different target for cryptanalysis compared to existing assumptions.

To build iO under this less-structured CRO assumption, we develop several new technical ideas. In particular, we devise an *oblivious LWE sampling* procedure, which succinctly encodes random LWE secrets and smudging noises, and uses a tailored-made homomorphic evaluation procedure to generate secure LWE samples. Crucially, all non-LWE components in this sampler, including the secrets and noises of the generated samples, are independently and randomly distributed, avoiding attacks on non-LWE components.

In the second part of this work, we investigate recent constructions of obfuscation for pseudorandom functionalities. We show that the same cryptanalytic techniques used to break previous LWE-with-hints assumptions for iO (Hopkins-Jain-Lin CRYPTO 21) can be adapted to construct counterexamples against the private-coin evasive LWE assumptions underlying these pseudorandom obfuscation schemes. Unlike prior counterexamples for private-coin evasive LWE assumptions, our new counterexamples take the form of zeroizing attacks, contradicting the common belief that evasive-LWE assumptions circumvent zeroizing attacks by restricting to "evasive" or pseudorandom functionalities.

[*] University of Washington, Email:ychsieh@cs.washington.edu

[†] Carnegie Mellon University, Email:aayushja@andrew.cmu.edu

[‡] University of Washington, Email:rachel@cs.washington.edu

Contents

1 Introduction

Indistinguishability obfuscation (iO) for general polynomial-size circuits [BGI+01, GKR08, GGH+13b] requires that for any two circuits C_0 and C_1 of the same size and functionality—meaning $\mathsf{C}_0(x) = \mathsf{C}_1(x)$ for all inputs x—the obfuscated circuits $\mathsf{iO}(\mathsf{C}_0)$ and $\mathsf{iO}(\mathsf{C}_1)$ should be computationally indistinguishable. Moreover, the obfuscator iO must run in probabilistic polynomial time and output a circuit C' that preserves functionality with probability 1, i.e., $\mathsf{C}'(x) = \mathsf{C}(x)$ for all x.

Since its inception, iO has been a powerful cryptographic primitive, enabling a broad range of applications in cryptography and complexity theory (see, e.g., [GGH+13b, SW14, BFM14, GGG+14, HSW13, KLW15, BPR15, CHN+16, GPS16, HJK+16]). However, constructing secure iO has remained a significant challenge. Following the first heuristic candidate [GGH+13b], a long line of work [GGH13a, GGH+13b, BGK+14, BR14, PST14, AGIS14, BMSZ16, CLT13, CLT15, GGH15, CHL+15, BWZ14, CGH+15, HJ16, BGH+15, Hal15, CLR15, MF15, MSZ16, DGG+18, Lin16, LV16, AS17, Lin17, LT17, GJK18, AJS18, Agr19, LM18, JLMS19, BIJ+20, AP20, GJLS21] explored a diverse range of hardness assumptions, including multilinear maps, affine determinant programs, and block-local PRGs, before culminating in the first provably secure iO construction [JLS21], based on four well-studied assumptions. This was later improved to rely on three assumptions [JLS22, RVV24], namely, the Decisional Linear (DLin) assumption on symmetric bilinear maps, Learning Parity with Noise (LPN) over large fields, and constant-local PRGs or sparse LPN.

Despite these advancements, a grand challenge remains: constructing iO that is secure against quantum adversaries. Current state-of-the-art constructions [JLS21, JLS22, RVV24] rely on bilinear maps, leaving them susceptible to quantum attacks. While some alternative approaches—such as those based on multilinear maps, or affine determinant programs, or random local mixing [CCMR24]—currently face no known quantum attacks, their security is not well understood, lacking reductions to simple-to-state hardness assumptions. This gap risks relying on "security by obscurity", limiting confidence in their approach. Besides post-quantum security, another important challenge is basing iO on simple-to-state hard problems w.r.t. a *single* mathematical structure, rather than *three* as the current state-of-the-art constructions do.

For both challenges, the ultimate long-term objective is to construct iO from a standard post-quantum assumption like Learning with Errors (LWE). However, our current understanding remains far from this goal. This raises the following natural and compelling question as an intermediate milestone toward that ultimate goal:

Can we build post-quantum iO *from a simple-to-state, principled, assumption?*

Recent Attempts. A recent exciting body of works have proposed lattice-inspired iO candidates [BDGM20, GP21, WW21, DQV+21, BDGM22], some of which are based on new, simple-to-state lattice assumptions. This includes the Circular Shielded Randomness Security (circ-SRL) assumption by [GP21, BDGM22], the Homomorphic Pseudorandom LWE Samples (HPLS) conjecture by [WW21], and the Subspace Flooding assumption by [DQV+21]. In addition, two very recent works [BDJ+24, AKY24] constructed iO for pseudorandom functionalities–termed Pseudorandom Obfuscation (PrO)–where the outputs of the circuits are pseudorandom, based on variants of private-coin evasive LWE assumptions, first introduced by [Wee22, Tsa22, VWW22] in the context of building attribute-based encryption and witness encryption.

Despite their differences, all these assumptions share a common structure:

> **LWE-with-hints assumptions** *posit that certain (circular) LWE samples retain some security (indistinguishability or pseudorandomness) even in the presence of specific* hints *that leak information about these samples.*

The presence of hints in these assumptions is crucial for achieving the functionality of iO, which requires revealing the outputs of the circuit evaluated on arbitrarily chosen inputs in the clear. Typically, these hints allow opening the output encoding derived via homomorphic evaluation from the original LWE samples. Then iO security requires the LWE samples to retain some security in the presence of hints, in order to argue that no information of the original circuit is revealed beyond the outputs. However, the hints introduce a delicate trade-off: do they leak too much information, possibly completely compromising LWE security? Prior works conjectured that the worst case does not happen.

Unfortunately, subsequent cryptanalysis [HJL21, JLLS23, BDJ+24] has demonstrated counterexamples or attacks against all aforementioned LWE-with-hint assumptions, leaving us without any iO constructions proven secure under simple, plausibly post-quantum assumptions.

Our Contributions. In this work, we show that even for the weaker notion of pseudorandom obfuscation, there are counterexamples to the private-coin evasive LWE assumptions underlying the recent constructions [AKY24, BDJ+24].

Moving beyond the attacks, we present a new iO construction based on a new, simple-to-state, post-quantum assumption, that we call the **Circular security with Random Opening (CRO) assumption**. CRO also has the LWE-with-hint format, and is falsifiable, instance-independent, and fully specified. Importantly, CRO avoids the structural vulnerabilities in prior assumptions that has been exploited in attacks, circumventing direct application of known attack strategies.

At a very high level, the CRO assumption considers real distributions consisting of circular LWE samples, denoted as encodings, together with hints $\mathbf{R}^*$ that are random "openings" of certain ciphertexts $\mathbf{C}^*$ of zeros under the Gentry-Sahai-Waters (GSW) homomorphic encryption scheme [GSW13]. The opened zero-ciphertext $\mathbf{C}^*$ can be efficiently derived from the LWE samples, using a carefully crafted procedure F, and the opening satisfies the constraint that $\mathbf{C}^* = F(\mathsf{encodings}) = \mathsf{GSW.Enc}(\mathsf{GSW.hpk}, 0; \mathbf{R}^*)$, where the public key $\mathsf{GSW.hpk}$ is contained in encodings. The assumption postulates that the real distributions are indistinguishable to ideal distributions where the LWE samples are replaced with *random* samples, while the hints are sampled from a simulated distribution still satisfying the constraint.

We perform a systematic study of prior attack strategies, revealing that all prior LWE-with-hints assumptions suffer from *structural vulnerabilities* either in their hints or in the leakage of LWE noises induced by the hint. Except for contrived counterexamples, all known attacks exploit these vulnerabilities by focusing solely on the hints or noise leakage, and are oblivious of the LWE samples otherwise. See Table 1 for a summary of the structural vulnerabilities in prior assumptions. We show that our CRO assumption introduces key structural differences, as highlighted below, that circumvent direct application of prior attacks.

1. *(Pseudo)Random Hints:* The *hints* in CRO are marginally random in the real distributions and pseudorandom in the ideal distributions, ensuring that the hints alone do not have any structural vulnerabilities.

2. *No Natural Noise Leakage:* Since our hints are "opening" of zero-ciphertexts that can be efficiently derived from the LWE samples available in the real distribution, it does not induce any natural noise leakage, circumventing zeroizing attacks. (See more discussion shortly below.)

3. *Pseudorandomness of LWE Samples Given Hints:* Different from prior LWE-with-hint assumptions underlying iO [GP21, WW21, DQV+21, BDGM22] which all postulate the indistinguishability security of LWE samples at the presence of hints, and *lack* natural pseudorandom variants of their assumptions, CRO gives a way to reason about the pseudorandomness of

LWE samples, given hints that enable non-evasive and non-pseudorandom functionalities.

We further formulate a weaker, but still sufficient, indistinguishability version, shorthanded as IND-CRO. We believe that the plausible pseudorandomness version, vetted against known cryptanalytic techniques, adds confidence to the security of CRO and IND-CRO.

In short, comparing with prior LWE-with-hint assumptions, CRO exhibits fewer structural vulnerabilities. As discussed in cryptanalysis in Section 2.3.2, the above features enable circumventing previous attack avenues in a principled way.

In order to base security on the less structured CRO assumption, our new iO construction develops several new ideas, building upon prior techniques especially [GP21, BDGM22]. We believe that these ideas might be instrumental for future constructions of iO and other advanced primitives.

Next, we describe the CRO assumption in more detail and provide a high-level overview of our construction. The formal definition and cryptanalysis of CRO are given in Section 2, while a detailed construction overview appears in Section 5.

1.1 Our Construction and Assumption in a Nutshell

It is well known that to construct iO, it suffices to build exponentially efficient iO, or xiO [LPST16], assuming LWE. xiO is the simpler task of obfuscating circuits Π that have polynomial-size truth tables TT. The obfuscator is allowed to run in time polynomial in the size of the entire truth table, with the only constraint that the resulting obfuscated circuit remains succinct – sublinear in the size of TT[1].

The work of [BDGM20], followed by [GP21, WW21, DQV$^+$21, BDGM22], proposed an appealing approach towards constructing xiO. The key idea is that, assuming (circular) LWE assumptions, one can hide a secret circuit Π in a homomorphic encoding $\mathsf{HEnc}(\Pi)$, from which an encoded truth table $\mathsf{Enc}(\mathrm{TT})$ can be efficiently computed (possibly under a slightly different encoding). The core challenge in achieving xiO is devising a way to safely and succinctly "open" $\mathsf{Enc}(\mathrm{TT})$, revealing TT and hopefully nothing else. The overall paradigm is depicted below.

$$\left.\begin{array}{r}\mathsf{HEnc}_{\mathbf{s}}\left(\Pi \,||\, f^{\mathrm{circ}}(\mathbf{s}) \,||\, \cdots\right) \overset{\text{HEval}}{\Longrightarrow} \mathsf{Enc}(\mathrm{TT}) \\ \text{succinct opening } \mathsf{open}\end{array}\right\} \Longrightarrow (\mathrm{TT}, \mathsf{leak})$$

We note that typically in these constructions, besides the original circuit Π, the homomorphic encoding also hides circular secret-dependent messages $f^{\mathrm{circ}}(\mathbf{s})$ to facilitate the final opening[2].

Prior works developed different encoding and succinct opening methods, and captured security of their scheme via different LWE-with-hint assumptions. Naturally, the LWE samples in the assumptions facilitate the homomorphic encoding $\mathsf{HEnc}(\Pi)$, while the hint hint facilitates opening open. Inevitably, the final encoding $\mathsf{Enc}(\mathrm{TT})$ also consists of LWE samples (derived via homomorphic evaluation), and opening them reveals not only TT but also additional leakage leak of the LWE noises, as indicated in the paradigm above.

A key issue in prior LWE-with-hint assumptions is that the hints and/or the noise leakage exhibit structural vulnerabilities, which have been exploited in attacks. Notably, prior cryptanalysis efforts focused entirely on hints and leakage, without attacking the LWE samples directly. Specifically, as summarized in Table 1, [HJL21] attacked the hints in the circ-SRL security assumption of [GP21], and

[1] Otherwise, a trivial construction would be to simply output the truth table as the obfuscated circuit.

[2] Sometimes more than one LWE secrets are involved and the key-dependent messages may depend on multiple secrets.

Assumption	hint = LWE secret	LWE noise leakage	hint = GSW randomness	hint = Lattice trapdoor
circ-SRL ([GP21])	✗	✗	Counterexample ([HJL21])	✗
HPLS ([WW21])	Non-random	Counterexample ([HJL21])	✗	✗
Subspace Flooding ([DQV⁺21])	Attack ([JLLS23])	Non-random	✗	✗
Private-coin ELWE ([Tsa22, VWW22])	✗	Counterexample (Section 7)	✗	$
CRO (Ours)	✗	✗	$	✗

Table 1: Characterization for different information leakage beyond LWE samples for existing assumptions toward iO/PrO. In the table, ✗ stands for no such leakage exist, $ stands for the leakage is marginally random (from a well-defined distribution), *Attack* stands for that there exist adversary breaking the assumption by focusing on the leakage, and *Counterexample* stands for that there exist specific implementation for the assumption which can be broken by focusing on the leakage.

the leakage in the HPLS conjecture [WW21]. Similarly, [JLLS23] attacked the hints in the subspace flooding assumption of [DQV⁺21]. Finally, in this work, in Section 7, we sketch attacks targeting leakage in private-coin evasive LWE assumptions underlying pseudorandom obfuscation [BDJ⁺24, AKY24].

Our Assumption CRO: Formally described in Figure 3, our CRO assumption postulates that a real distribution of circular LWE samples with hints is indistinguishable to an ideal distribution consisting of random samples and simulated hints. In the real distribution the circular LWE samples contain a Gentry-Sahai-Waters (GSW) public key GSW.hpk, GSW ciphertexts GSW.hct, and other LWE samples $\mathbf{C}$, where the latter two hide secret-dependent messages. Their distribution is set up in such a way that, using a *special and carefully designed procedure* F, one can efficiently derive certain specific GSW ciphertexts of zeros, $\mathbf{C}^* = F(\mathsf{GSW.hpk}, \mathsf{GSW.hct}, \mathbf{C})$.

The key idea in CRO is that the hints are *random openings* $\mathbf{R}^*$ of $\mathbf{C}^*$. An opening of $\mathbf{C}^*$ is a random string $\mathbf{R}$ satisfying $\mathbf{C}^* = \mathsf{GSW.Enc}(\mathsf{GSW.hpk}, \mathbf{0}\,;\,\mathbf{R})$, which corresponds to a small-norm matrix in GSW. Then, a random opening $\mathbf{R}^*$ is sampled as a random small-norm Gaussian matrix satisfying the same constraint, that is, $\mathbf{R}^* \leftarrow \mathcal{D}|_{\mathbf{C}^*=\mathsf{GSW.Enc}(\mathsf{GSW.hpk},0\,;\,\mathbf{R}^*)}$, where $\mathcal{D}$ is the distribution of random Gaussian matrix of appropriate dimension and Gaussian width.

The CRO assumption postulates that the real distribution of LWE encodings and opening $\mathbf{R}^*$, is indistinguishable to random encodings, and an equivocated opening $\hat{\mathbf{R}}^*$.

$$\overbrace{\big(\mathsf{encodings} = (\mathsf{GSW.hpk}, \mathsf{GSW.hct}, \mathbf{C}),\ \mathbf{R}^*\big)}^{\text{Real}} \approx \overbrace{\big(\mathsf{encodings} = (\$, \$, \$),\ \mathbf{R}^*\big)}^{\text{Ideal}},$$

In the ideal distribution, $\mathbf{C}^* = F(\mathsf{encodings})$ is computed in the same way using procedure F but evaluated on random encodings. $\mathbf{R}^*$ is also sampled in the same way w.r.t. $\mathbf{C}^*$, that is random small Gaussian matrix subject to constraint $\mathbf{C}^* = \mathsf{GSW.Enc}(\mathsf{GSW.hpk}, 0\,;\,\mathbf{R}^*)$. $\mathbf{R}^*$ is well-defined, corresponding to a "random opening" of $\mathbf{C}^*$ relative to a truly random "public key", owning to the equivocal properties of GSW when the public key is random.

Key Features of the CRO Assumption:

- *(Pseudo)random hints:* We prove that in the real distribution, $\mathbf{C}^*$ is, marginally, *a random GSW ciphertext of zeros*, and hence its random opening is, marginally, a truly random small-norm Gaussian matrix. In the ideal distribution, we show that $\mathbf{R}^*$ is pseudorandom. Therefore, no attacks focusing on hints alone can succeed. The (pseudo)randomness of $\mathbf{R}^*$ stems from the carefully designed distribution of LWE encodings and the procedure F for evaluating $\mathbf{C}^*$ from them.

  This stands in contrast to the structured hints in the circ-SRL assumption [GP21] and the subspace flooding assumption [DQV+21], which led to attacks [HJL21, JLLS23].

- *No natrual noise leakage:* The GSW public key and ciphertext have the form $\bar{\mathbf{B}}^T = (\mathbf{B}^T, \mathbf{B}^T\mathbf{r} + \mathbf{e})$ and $(\mathbf{C}^*)^T = (\mathbf{P}^T, \mathbf{P}^T\mathbf{r} + \mathbf{e}^*)$, and a random opening $\mathbf{R}^*$ satisfies $\bar{\mathbf{B}} \cdot \mathbf{R}^* = \mathbf{C}^*$. $\mathbf{R}^*$ may appear similar to a lattice trapdoor $\mathbf{R} \leftarrow \mathbf{B}^{-1}(\mathbf{P})$, satisfying $\mathbf{B} \cdot \mathbf{R} = \mathbf{P}$ [MP12], but there is a crucial distinction. A trapdoor $\mathbf{R} \leftarrow \mathbf{B}^{-1}(\mathbf{P})$ yields an approximate equality $\bar{\mathbf{B}}\mathbf{R} \approx \mathbf{C}^*$ and thus leaks LWE noises $\mathbf{e}^* - \mathbf{e}\mathbf{R}$, whereas an opening $\mathbf{R}^*$ yields an exact equality $\bar{\mathbf{B}}\mathbf{R}^* = \mathbf{C}^*$, leaking no information about LWE noises.

  This distinguishes CRO from evasive LWE-type assumptions where the hints are lattice trapdoors. It also rules out attacks that only combine encodings and hint in the most natural way – multiplying $\bar{\mathbf{B}}$ and $\mathbf{R}^*$ yields $\mathbf{C}^* = \bar{\mathbf{B}}\mathbf{R}^*$ which can already be efficiently computed from the original LWE encodings in the assumption, giving no additional information.

  In contrast, in Section 7 we describe new attacks on private-coin evasive LWE assumptions underlying recent construction of pseudorandom obfuscation [AKY24, BDJ+24]. Unlike prior attacks on private-coin evasive LWE [VWW22, BÜW24], our attack exploits structure in the noise leakage obtained after computing $\bar{\mathbf{B}} \cdot \mathbf{R}$. Hence, our attack is similar in principle to previous zeroizing attacks (e.g., [GGH+13b]), and falsifies prior intuition that evasive LWE assumptions are not subject to zeroizing attacks.

  The attacks of [HJL21] on the HPLS conjecture [WW21] also focus on noise leakage only, though their hints are different, and are functions of the LWE secrets.

- *Pseudorandom vs Indistinguishability Assumptions:* The distribution of encodings in CRO follow the same principle behind circular-security of LWE and key-dependent-message security of GSW. Therefore, attacks on the encodings *alone* would undermine widely adopted circular security assumptions (e.g., [BV11, GSW13]). When combined with the opening $\mathbf{R}^*$, there is an efficiently verifiable constraint $\mathbf{C}^* = F(\mathsf{encodings}) = \mathsf{GSW.Enc}(\mathsf{GSW.hpk}, 0\,;\,\mathbf{R}^*)$. CRO gives a new way to reason about the pseudorandomness of the LWE encodings at the presence of hint, stating that the real LWE encodings can be switched to random in an indistinguishable way, if $\mathbf{R}^*$ is simulaneously equivocated to maintain the constraint.

  As discussed before, prior LWE-with-hint assumptions underlying iO [GP21, WW21, DQV+21, BDGM22] postulate only indistinguishability security[3]. At first glance, indistinguishability may appear weaker and more preferable. The subtle issue, however, is that these assumptions *do not have* a natural stronger pseudorandom variant. In our view, the lack of a pseudorandom variant is precarious: *can LWE samples that lack pseudorandomness still retain any security*? This seems to be at odds with the common intuition that security of LWE-based schemes typically relies on pseudorandomness.

[3] Private-coin evasive LWE assumptions are pseudorandomness type assumptions. However, they only enable pseudorandom functionalities.

Therefore, we view the plausible pseudorandomness of CRO, vetted against known cryptanalytic techniques, as a strength of the assumption. At the same time we formulate a weaker (but sufficient) indistinguishability-based variant $\mathsf{IND\text{-}CRO}$ (described formally in Figure 9).

- *Remaining Challenge in Cryptanalysis:* The above three features entail that attacks on CRO must combine $\mathsf{encodings}$ and hint in ways more sophisticated than simply computing $\bar{\mathbf{B}} \cdot \mathbf{R}^*$. However, to the best of our knowledge, it is unclear how to extend current cryptanalytic techniques (e.g., lattice attacks) to leverage $\mathbf{R}^*$ in a non-trivial way. In the literature, such behavior has only arisen in contrived counterexamples – for instance, in prior cryptanalysis of private-coin evasive LWE [VWW22, BÜW24], the attacker receives auxiliary information (an obfuscated circuit) that helps leverage the trapdoors. By contrast, CRO does not provide auxiliary information and instead features a natural distribution of $\mathsf{encodings}$ and hint.

Highlights of Our Construction In order to base security on CRO, our construction of xiO, building upon [GP21, BDGM22], carefully combines several new ideas. The main component is *oblivious LWE sampling*, whose goal is to generate LWE samples $\tilde{\mathbf{s}}\mathbf{A} + \tilde{\mathbf{e}}$, where $\tilde{\mathbf{s}} \in \mathbb{Z}_q^n$ and $\mathbf{A} \in \mathbb{Z}_q^{n\times\ell}$, from a *succinct* encoding that is much shorter than the bit length $\ell \log q$ of generated samples. Crucially, the security guarantee is that the secrets $\tilde{\mathbf{s}}$ and $\tilde{\mathbf{e}}$ remain pseudorandom and can be used to re-randomize other LWE samples. Let us briefly highlight some key ideas in our construction; we refer the reader to Section 4 for a detailed overview of how these ideas are implemented.

(1) We introduce new ways of encoding random LWE secrets $\mathbf{s}$ and smudging noises $\mathbf{e}$ inside the oblivious LWE sampler or xiO encoding. Specifically, the encoding contains GSW ciphertexts of $\mathbf{s}$, along with LWE samples using noises $\mathbf{e}$ modulo a *small* modulus $\Delta \ll q$. This encoding differs from prior approaches, which either store $\mathbf{s}$ and $\mathbf{e}$ in the CRS, derive them from a PRF, or expand them from $\mathbf{sB} + \mathbf{e}$ and a trapdoor $\mathbf{B}^{-1}(\mathbf{P})$.

(2) We design a carefully crafted homomorphic evaluation procedure to derive a GSW ciphertext hct' encrypting $(\mathbf{sA} + \mathbf{e}) \bmod q$. This special procedure is crucial to ensure that the $\mathsf{hint} = \mathbf{R}^*$ in CRO has a random marginal distribution in the real distribution and pseudorandom in the ideal distribution, thereby avoiding certain attacks. In contrast, prior constructions rely on generic homomorphic evaluation procedure, which ended up leading to counterexamples [HJL21].

(3) Next, the GSW ciphertext hct' of $(\mathbf{sA} + \mathbf{e}) \bmod q$ is homomorphically decrypted using the dual GSW scheme (a.k.a. the packed dual-Regev encryption), producing the final samples $\tilde{\mathbf{s}}\mathbf{A} + \tilde{\mathbf{e}}$. Owing to (1) and (2), we can show that $\tilde{\mathbf{s}}$ and $\tilde{\mathbf{e}}$ are both *truly random* in their marginal joint distribution. This allows the security reduction to CRO to internally emulate $\tilde{\mathbf{s}}$ and $\tilde{\mathbf{e}}$ by sampling them randomly. As a result, the CRO assumption itself only contains an opening $\mathbf{R}^*$ and does not incur natural noise leakage.

(4) Finally, it is essential to *rerandomize* the GSW ciphertext hct' before performing homomorphic decryption. We achieve this using public randomness $\mathbf{R}^*$ in the CRS by setting $\mathsf{hct}' = \mathsf{hct}' + \bar{\mathbf{B}}\mathbf{R}^*$, following the approach of [GP21]. Rerandomization is key to achieving *simulation-based* security for oblivious LWE sampler and xiO (instead of mere indistinguishability). Indeed, the simulator can "program" the truth table TT into $\mathbf{R}^*$ in the CRS.

    Recall that CRO conjectures the *pseudorandomness* of the LWE samples when the hint is simultaneously equivocated. Intuitively, (as typically is the case when using pseudorandom assumptions) to maintain the correctness of the oblivious LWE sampler/xiO when the LWE samples switch to random, we need to "program" the outputs (fresh LWE samples or TT) into the CRS. The stronger simulation security of xiO is interesting on its own.

1.2 Paper Organization

In Section 2, we formally define our new CRO assumption. We also provide cryptanalysis of this assumption and compare it with related assumptions. Section 3 presents the necessary preliminaries for our iO construction, including a recall of functional encodings, a key primitive that implies iO.

In Section 4, we provide an overview of our functional encoding construction, highlighting the motivation behind our design and its essential algebraic components. Section 5 introduces a formal construction of functional encoding schemes for all polynomial-sized circuits based on the CRO assumption.

In Section 6, we define a weaker indistinguishability variant of the CRO assumption, denoted IND-CRO, and demonstrate how it implies an oblivious LWE sampler, a primitive known to imply functional encoding schemes.

Finally, in Section 7, we present a counterexample to the private-coin evasive LWE scheme.

2 Circular Security with Random Opening (CRO)

2.1 Preliminaries

Notations We denote the set of integers by $\mathbb{Z}$, and the set of integers modulo q by $\mathbb{Z}_q$. Vectors are represented as boldface lowercase letters, while matrices are denoted by boldface uppercase letters. For a matrix $\mathbf{W} \in \mathbb{Z}_q^{(n+1)\times m}$, we use $\overline{\mathbf{W}} \in \mathbb{Z}_q^{n\times m}$ to denote the matrix $\mathbf{W}$ excluding its bottom row, and we use $\underline{\mathbf{W}} \in \mathbb{Z}_q^{1\times m}$ to denote the bottom row of the matrix $\mathbf{W}$. The n-dimensional identity matrix is denoted as $\mathbf{I}_n$, and zero vectors/matrices of corresponding dimensions are written as $\mathbf{0}^n$ and $\mathbf{0}^{n\times m}$, respectively. We use $\mathbf{1}_i$, $\mathbf{1}_{i,j}$ to denote the unit vector/matrix with a 1 on the i-th/(i,j)-th index and 0 elsewhere. The notation $\mathbf{1}$ will only be used when the dimension of the unit vector/matrix is clear from the context. For a vector $\mathbf{a} \in \mathbb{Z}_q^n$, we denote the bitwise decomposition $\mathsf{bits}(\mathbf{a}) \in \{0,1\}^{n\lceil \log q\rceil}$, which consists of the bitwise representation of each entry of $\mathbf{a}$, ordered from the least significant to the most significant bit. For matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, we define $\|\mathbf{A}\|$ as the maximal absolute value of its entries, and $\mathsf{vec}(\mathbf{A}) \in \mathbb{Z}_q^{nm}$ as the column-wise vectorization of $\mathbf{A}$. The discrete Gaussian distribution over the integers with standard deviation σ is denoted by $\mathcal{D}_\sigma$. For two matrices $\mathbf{A}$, $\mathbf{B}$, we define their Kronecker product as $\mathbf{A} \otimes \mathbf{B}$. We define the gadget vector and the gadget matrix as follows:

$$\mathbf{g}_q = \left(1, 2, \ldots, 2^{\lceil \log q\rceil - 1}\right)^\intercal \in \mathbb{Z}_q^{\lceil \log q\rceil}, \quad \mathbf{G}_{n,q} = \mathbf{I}_n \otimes \mathbf{g}_q^\intercal \in \mathbb{Z}_q^{n\times n\lceil \log q\rceil}.$$

We omit the subscripts n, q when the dimensions/modulus are clear from the context. Given vector $\mathbf{p} \in \mathbb{Z}_q^n$, we define $\mathbf{G}^{-1}(\mathbf{p}) \in \{0,1\}^{n\lceil \log q\rceil}$ as the bitwise decomposition $\mathsf{bits}\mathbf{p}$. In particular, we have $\mathbf{G}\mathbf{G}^{-1}(\mathbf{p}) = \mathbf{p}$. We also extend the notation to matrices in a column-wise manner.

Standard Lemmas We introduce a few useful lemmas that will be used in this section.

Lemma 1 (Corollary of the Leftover Hash lemma ([HILL99, MM11]))**.** *Let n, m, q be integers such that $m > cn\log q$ for some sufficiently large constant c, and $q \geq 2$. Then, for every $k = \mathsf{poly}(n)$, the statistical distance of the following two distributions is bounded by $2^{-\Omega(n)}$*

$$\left\{(\mathbf{B}, \mathbf{BR}) \middle| \mathbf{B} \leftarrow \mathbb{Z}_q^{n\times m}, \mathbf{R} \leftarrow \{0,1\}^{m\times k}\right\} \approx_s \left\{(\mathbf{B}, \mathbf{U}) \middle| \mathbf{B} \leftarrow \mathbb{Z}_q^{n\times m}, \mathbf{U} \leftarrow \mathbb{Z}_q^{n\times k}\right\}.$$

Furthermore, the lemma also holds for $\mathbf{R} \leftarrow \mathcal{D}_\sigma^{m\times k}$ for $\sigma = \omega(\sqrt{\log n})$.

The following are a few well-known statistical lemmas. We omit the proofs here,

Lemma 2 (Gaussian Tail)**.** *For all $\lambda \in \mathbb{N}$ and $\sigma > 0$,*

$$\Pr[|x| \geq \sqrt{\lambda}\sigma | x \leftarrow \mathcal{D}_\sigma] \leq 2^{-\lambda}$$

Lemma 3 (Discrete Gaussian Smudging)**.** *For all $\lambda \in \mathbb{N}$, $y \in \mathbb{Z}$, and $\sigma \geq 2^\lambda |y|$, the statistical distance between $y + \mathcal{D}_\sigma$ and $\mathcal{D}_\sigma$ is at most $2^{-\Omega(\lambda)}$.*

Lemma 4 (Rounding lemma)**.** *Let $n, q, \Delta \in \mathbb{N}$ such that $\Delta | q$. Let $(\mathbf{v}, \mathbf{e}) \in (\mathbb{Z}_q^n)^2$ be (joint) random variables where*

- *The marginal distribution of $\mathbf{v}$ is ϵ-close to uniform in $\mathbb{Z}_q^n$.*
- *$\mathbf{e}$ has bounded norm, $\Pr[\|\mathbf{e}\| \geq B] \leq \epsilon'$.*

Then

$$\Pr\left[\left\lfloor \frac{\mathbf{v}}{\Delta} \right\rfloor \neq \left\lfloor \frac{\mathbf{v} + \mathbf{e}}{\Delta} \right\rfloor\right] \leq \frac{2Bn}{\Delta} + \epsilon + \epsilon'.$$

Lattice Trapdoor We recall the notion of lattice trapdoors from [GPV08, MP12].

Lemma 5 (Lattice Trapdoor[GPV08, MP12, BLP+13])**.** *Let n, m, q be integers such that $m \geq 3n \log q$. There exist efficient algorithms $\mathsf{TrapGen}, \mathsf{SampPre}$:*

- *$\mathsf{TrapGen}(1^n, q, m)$ takes as input the lattice parameters and outputs matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and its trapdoor $\mathbf{T}$.*
- *$\mathsf{SampPre}(\mathbf{A}, \mathbf{T}, \mathbf{Y}, \sigma)$ takes as input a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, its trapdoor $\mathbf{T}$, a target $\mathbf{Y} \in \mathbb{Z}_q^{n \times k}$, and a width parameter σ, and outputs a preimage $\mathbf{R} \in \mathbb{Z}_q^{m \times k}$.*

The algorithms satisfy the following two properties.

Statistical Randomness: *The distribution of $\mathbf{A}$ sampled from $\mathsf{TrapGen}$ is 2^{-n}-close to uniform:*

$$\left\{ \mathbf{A} \middle| (\mathbf{A}, \mathbf{T}) \leftarrow \mathsf{TrapGen}(1^n, q, m) \right\} \approx_s^{2^{-n}} \left\{ \mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m} \right\}.$$

Preimage Sampling *For every polynomial $k = k(n)$, every ensembles of target matrices $\{\mathbf{Y}_n \in \mathbb{Z}_q^{n \times k}\}_n$, and every width parameter $\sigma > m \log n$, the preimage obtained from $\mathsf{SampPre}$ is 2^{-n}-close the conditional Gaussian distribution:*

$$\left\{ (\mathbf{A}, \mathbf{R}) \,\middle|\, \begin{array}{rl} (\mathbf{A}, \mathbf{T}) & \leftarrow\!\!\$ \ \mathsf{TrapGen}(1^n, q, m) \\ \mathbf{R} & \leftarrow\!\!\$ \ \mathsf{SampPre}(\mathbf{A}, \mathbf{T}, \mathbf{Y}, \sigma) \end{array} \right\}_n \approx_s^{2^{-n}} \left\{ (\mathbf{A}, \mathbf{R}) \,\middle|\, \begin{array}{rl} (\mathbf{A}, \mathbf{T}) & \leftarrow\!\!\$ \ \mathsf{TrapGen}(1^n, q, m) \\ \mathbf{R} & \leftarrow\!\!\$ \ \mathcal{D}_\sigma^{m \times k}|_{\mathbf{AR}=\mathbf{Y}} \end{array} \right\}_n$$

Learning with Error We recall the LWE assumption.

Definition 1 (LWE [Reg05])**.** *Let $n = n(\lambda)$ and $q = q(\lambda)$ be integers, and let $\sigma = \sigma(\lambda) > 0$ be a noise parameter. We say that the ϵ-$\mathsf{LWE}_{n,q,\sigma}$ assumption holds if for all $m = \mathsf{poly}(n)$, the following ensembles are $\epsilon(\lambda)$ indistinguishable to all polynomial-sized adversaries:*

$$\left\{ (\mathbf{A}, \mathbf{s}^\intercal \mathbf{A} + \mathbf{e}^\intercal) \,\middle|\, \mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}, \mathbf{s} \leftarrow \mathbb{Z}_q^n, \mathbf{e} \leftarrow \mathcal{D}_\sigma^m \right\} \approx_c^\epsilon \left\{ (\mathbf{A}, \mathbf{u}^\intercal) \,\middle|\, \mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}, \mathbf{u} \leftarrow \mathbb{Z}_q^m \right\}$$

In this work, we rely on sub-exponential security for LWE with sub-exponential modulus-to-noise ratio. Namely, i.e., $n = \mathsf{poly}(\lambda)$, $q/\sigma = \Theta(2^{n^\delta})$ for some constant $\delta \in (0, 1)$. We assume ϵ-$\mathsf{LWE}_{n,q,\sigma}$ holds for $\epsilon(\lambda) = 2^{-\lambda^\rho}$ for some constant $\rho > 0$.

2.1.1 Homomorphic Encryption

We define our abstraction of GSW FHE scheme, which hides most details of the construction and highlights the properties that are crucial for understanding the assumption structure. Note that GSW supports evaluating mixed circuits $f : \{0,1\}^k \to \mathbb{Z}_q^\ell$ which first computes the bitwise representation of the output and then packs the bits into $\mathbb{Z}_q$ elements. We therefore consider function class $\mathcal{F}$ mapping to $\mathbb{Z}_q$ vectors and define approximate correctness for decryption.

Definition 2 (Homomorphic Encryption.)**.** *Let n be a positive integer, and all other parameters are implicitly dependent on n. A homomorphic encryption scheme with message space $\mathcal{M}$, key space $\mathcal{K}$, and encryption space C, supporting function class $\mathcal{F}$ mapping vectors over $\mathcal{M}$ to vectors over a ring $\mathcal{R}$ that is contained in $\mathcal{M}$, consists of the following algorithms:*

- $\mathsf{PKGen}(\mathbf{r})$ *takes as input a randomly sampled secret key $\mathbf{r} \leftarrow \mathbb{Z}_q^n$ and outputs a public key* $\mathsf{hpk} \in \mathcal{K}$.
- $\mathsf{Enc}(\mathsf{hpk}, \mathbf{m}; \mathbf{R})$ *takes as input a public key* hpk*, a message $\mathbf{m} \in \mathcal{M}^k$ for some dimension k, and encryption randomness $R \leftarrow \mathcal{D}_{\text{enc}}^k$ sampled according to $\mathcal{D}_{\text{enc}}$, outputs a ciphertext* hct *which is a vector over C^k. (Sometimes the notation* $\mathsf{hct}(\mathbf{m})$ *is used in order to explicitly indicate the encrypted message.)*
- $\mathsf{Eval}(\mathsf{hct}(\mathbf{m}), f)$ *takes as input a ciphertext* $\mathsf{hct}(\mathbf{m})$*, a circuit $f \in \mathcal{F}$, and outputs a ciphertext* hct_f *of the output $f(\mathbf{m})$.*
- $\mathsf{Dec}(\mathsf{hsk}, \mathsf{hct})$ *takes as input a secret key* hsk *and a ciphertext* hct*, and outputs a message $\mathbf{m} \in \mathcal{M}^k \cup \{\perp\}$.*

We require a homomorphic encryption scheme to be correct and secure as defined below.

$\alpha(n)$**-Approximate Decryption Correctness:** *For every $k \in \mathbb{Z}$, message $m \in \mathcal{M}^k$, and function $f \in \mathcal{F}$ taking inputs of k elements, the output ciphertext of the homomorphic encryption decrypts to some message that is $\alpha(n)$-close to the correct evaluation outcome $f(\mathbf{m})$ under the Euclidean norm.*

$$\Pr\left[\left\|\mathsf{Dec}(\mathsf{hsk}, \mathsf{hct}_f) - f(\mathbf{m})\right\| \le \alpha(n) \;\middle|\; \begin{array}{c} \mathbf{r} \leftarrow \mathbb{Z}_q^n \\ \mathsf{hpk} \leftarrow \mathsf{PKGen}(\mathbf{r}) \\ \mathbf{R} \leftarrow \mathcal{D}_{\text{enc}}^k \\ \mathsf{hct} = \mathsf{Enc}(\mathsf{hpk}, \mathbf{m}; \mathbf{R}) \\ \mathsf{hct}_f = \mathsf{Eval}(\mathsf{hct}, f) \end{array}\right] = 1$$

$\epsilon(n)$**-Pseudorandom Public Key and Ciphertext:** *For every polynomial $k = k(n)$, every ensemble of messages $\{\mathbf{m} \in \mathcal{M}^k\}_n$, the following ensembles are $\epsilon(n)$-indistinguishable to all polynomial-sized adversaries:*

$$\left\{(\mathsf{hpk}, \mathsf{hct}) \;\middle|\; \begin{array}{rl} \mathbf{r} & \leftarrow \mathbb{Z}_q^n \\ \mathsf{hpk} & \leftarrow \mathsf{PKGen}(\mathbf{r}) \\ \mathbf{R} & \leftarrow \mathcal{D}_{\text{enc}}^k \\ \mathsf{hct} & = \mathsf{Enc}(\mathsf{hpk}, \mathbf{m}; \mathbf{R}) \end{array}\right\}_n \approx_c^\epsilon \left\{(\mathsf{hpk}, \mathsf{hct}) \;\middle|\; \begin{array}{rl} \mathsf{hpk} & \leftarrow \mathcal{K} \\ \mathsf{hct} & \leftarrow C^k \end{array}\right\}_n$$

We also formulate the following additional properties, which are satisfied by the GSW scheme.

Definition 3. *A homomorphic encryption scheme is statistically $(\mathcal{D}_{\text{rand}}, \epsilon(n))$-**rerandomizable**, if for every polynomial $k = k(n)$, polynomial $\ell = \ell(n)$, ensemble of function-message pairs $\{\mathbf{m} \in \mathcal{M}^k, f \in \mathcal{F}_{k,\ell}\}_n$,* *where $\mathcal{F}_{k,\ell}$ is the subset of function class $\mathcal{F}$ that maps $\mathcal{M}^k$ to $\mathcal{R}^\ell$, it holds that for sufficiently large $n \in \mathbb{Z}$,*

$$\Pr\left[\mathsf{SD}\left(\begin{array}{r} \left(\mathsf{hct}_f \boxplus (-f(\mathbf{m})) \boxplus \mathsf{hct}_\mathbf{0} \,\middle|\, \mathsf{hct}_\mathbf{0} \leftarrow \mathsf{Enc}(\mathsf{hpk}, \mathbf{0}^\ell; \mathcal{D}_{\mathrm{rand}})\right), \\ \left(\mathsf{hct}_\mathbf{0} \,\middle|\, \mathsf{hct}_\mathbf{0} \leftarrow \mathsf{Enc}(\mathsf{hpk}, \mathbf{0}^\ell; \mathcal{D}_{\mathrm{rand}})\right) \end{array} \right) \leq \epsilon(n) \,\middle|\, \begin{array}{r} \mathbf{r} \leftarrow \mathbb{Z}_q^n \\ \mathsf{hpk} \leftarrow \mathsf{PKGen}(\mathbf{r}) \\ \mathbf{R} \leftarrow \mathcal{D}_{\mathrm{enc}}^k \\ \mathsf{hct} = \mathsf{Enc}(\mathsf{hpk}, \mathbf{m}; \mathbf{R}) \\ \mathsf{hct}_f = \mathsf{Eval}(\mathsf{hct}, f) \end{array} \right] = 1$$

where $\mathbf{0}$ denotes the zero element in ring $\mathcal{R}$, and $\boxplus$ is the homomorphic addition operation over two ciphertexts or over a ciphertext and a constant, implicitly defined by Eval, *and* $\mathsf{SD}(A, B)$ *denotes the statistical distance between two distributions.*

Definition 4. *A homomorphic encryption scheme has $\epsilon(n)$*-**equivocal mode** *if there are two additional algorithms:*

- $\mathsf{TDGen}(1^n, q)$ *samples a public key* hpk *together with a trapdoor* $\mathbf{T}$.
- $\mathsf{TDSamp}(\mathsf{hpk}, \mathbf{T}, \mathsf{hct})$ *on input a public key* hpk *with a trapdoor* $\mathbf{T}$, *and a target ciphertext* $\mathsf{hct} \in \mathcal{C}^\ell$, *samples a matching encryption randomness* $\mathbf{R}$ *satisfying* $\mathsf{Enc}(\mathsf{hpk}, \mathbf{0}^\ell \,;\, \mathbf{R}) = \mathsf{hct}$.

These two algorithms satisfy the following two statistical properties:

$$\left\{ \mathsf{hpk} \,\middle|\, \begin{array}{rl} \mathbf{r} & \leftarrow \mathbb{Z}_q^n \\ \mathsf{hpk} & \leftarrow \mathsf{PKGen}(\mathbf{r}) \end{array} \right\}_n \approx_s^\epsilon \left\{ \mathsf{hpk} \,\middle|\, (\mathsf{hpk}, \mathbf{T}) \leftarrow \mathsf{TDGen}(1^n, q) \right\}_n$$

For every polynomial $\ell = \ell(n)$, every ensemble of ciphertexts $\{\mathsf{hct} \in \mathcal{C}^\ell\}_n$,

$$\left\{ (\mathsf{hpk}, \mathbf{R}^*) \,\middle|\, \begin{array}{rl} (\mathsf{hpk}, \mathbf{T}) & \leftarrow \mathsf{TDGen}(1^n, q) \\ \mathbf{R}^* & \leftarrow \mathcal{D}_{\mathrm{rand}}|_{\mathsf{Enc}(\mathsf{hpk}, \mathbf{0}^\ell; \mathbf{R}^*) = \mathsf{hct}} \end{array} \right\}_n \approx_s^\epsilon \left\{ (\mathsf{hpk}, \mathbf{R}^*) \,\middle|\, \begin{array}{rl} (\mathsf{hpk}, \mathbf{T}) & \leftarrow \mathsf{TDGen}(1^n, q) \\ \mathbf{R}^* & \leftarrow \mathsf{TDSamp}(\mathsf{hpk}, \mathbf{T}, \mathsf{hct}) \end{array} \right\}_n$$

2.1.2 GSW encryption

In this section, we recall the construction of the GSW encryption scheme [GSW13] under the definitional framework of definition 2.

The GSW encryption has message space $\mathcal{M} = \{0, 1\} \cup \mathbb{Z}_q^{m'}$, key space $\mathbb{Z}_q^n$, and ciphertext space $\mathbb{Z}_q^{(n+1)\times m'}$, where $m' = (n+1)\lceil \log q \rceil$ is the width of the gadget matrix $\mathbf{G}_{n+1}$.

We first define the function class $\mathcal{F}_{d,M}$ which the GSW HE scheme supports.

Definition 5 (Bounded depth packed circuit). *The function class $\mathcal{F}_{d,M}$ of packed circuit with depth bound d and output size M consists of functions of the form*

$$f : \{0, 1\}^k \times \mathbb{Z}_q^{k'm'} \to \mathbb{Z}_q^M, \quad f(\mathbf{x}, \mathbf{v}) = L(C(\mathbf{x}), \mathbf{v}),$$

where $L : \mathbb{Z}_q^{\ell' m'} \times \mathbb{Z}_q^{k'm'} \to \mathbb{Z}_q^M$ is some linear function over $\mathbb{Z}_q$ with $\{0, 1, -1\}$-coefficients, and $C : \{0, 1\}^k \to \mathbb{Z}_q^{\ell' m'}$ is a function described by a polynomial sized depth d circuit which computes bit outputs $\{0, 1\}^{\ell' m' \lceil \log q \rceil}$, subsequently packed into vectors in $\mathbb{Z}_q^{\ell' m'}$.

The GSW HE schemes supports bounded depth packed circuits $\mathcal{F}_{d,M}$ for appropriate $q = m'^{\Omega(d)}$ and M is a multiple of m'.

The scheme consists of the following algorithms:

- $\mathsf{PKGen}(\mathbf{r})$: Sample a public matrix $\mathbf{B} \leftarrow \mathbb{Z}_q^{n\times m}$ for appropriate $m = \Theta(n \log q)$ and a noise vector $\mathbf{e} \leftarrow \mathcal{D}_\sigma^n$. Output the public key

$$\mathsf{hpk} = \overline{\mathbf{B}} = \begin{pmatrix} \mathbf{B} \\ \mathbf{r}^\top\mathbf{B} + \mathbf{e}^\top \end{pmatrix} \in \mathbb{Z}_q^{(n+1)\times m}.$$

- $\mathsf{Enc}(\mathsf{hpk}, \mathbf{x}; \mathbf{R})$: For a single element $x \in \mathcal{M} = \{0,1\} \cup \mathbb{Z}_q^{m'}$, the encryption of x is computed by

$$\mathsf{hct}(x) = \overline{\mathbf{B}}\mathbf{R} + x \cdot \mathbf{G}_{n+1} \text{ (if } x \in \{0,1\}), \quad \mathsf{hct}(x) = \overline{\mathbf{B}}\mathbf{R} + \begin{pmatrix} \mathbf{0}^{n\times m'} \\ x^\top \end{pmatrix} \text{ (if } x \in \mathbb{Z}_q^{m'}),$$

  where the encryption randomness $\mathbf{R}$ is sampled from $\mathcal{D}_{\mathrm{enc}} = \{0,1\}^{m\times m'}$. For vectors $\mathbf{x} \in \mathcal{M}^k$, encryption can be performed element-wise.

- $\mathsf{Eval}(\mathsf{hct}(\mathbf{x}), f)$: The GSW scheme supports homomorphic evaluation of functions using the following operations:

  – **HAdd:** Ciphertext for bits and vectors are respectively additive homomorphic.

$$\mathsf{hct}(b_1) \boxplus \mathsf{hct}(b_2) = (\overline{\mathbf{B}}\mathbf{R}_1 + b_1\mathbf{G}) + (\overline{\mathbf{B}}\mathbf{R}_2 + b_2\mathbf{G}) = \overline{\mathbf{B}}(\mathbf{R}_1 + \mathbf{R}_2) + (b_1 + b_2)\mathbf{G} = \mathsf{hct}(b_1 + b_2),$$
$$\mathsf{hct}(\mathbf{v}_1) \boxplus \mathsf{hct}(\mathbf{v}_2) = (\overline{\mathbf{B}}\mathbf{R}_1 + \begin{pmatrix} \mathbf{0} \\ \mathbf{v}_1^\top \end{pmatrix}) + (\overline{\mathbf{B}}\mathbf{R}_2 + \begin{pmatrix} \mathbf{0} \\ \mathbf{v}_2^\top \end{pmatrix}) = \overline{\mathbf{B}}(\mathbf{R}_1 + \mathbf{R}_2) + \begin{pmatrix} \mathbf{0} \\ \mathbf{v}_1^\top + \mathbf{v}_2^\top \end{pmatrix} = \mathsf{hct}(\mathbf{v}_1 + \mathbf{v}_2).$$

  Extending this to plaintext vectors, we have

$$\mathsf{hct}(\mathbf{v}_1) \boxplus \mathbf{v}_2 = (\overline{\mathbf{B}}\mathbf{R} + \begin{pmatrix} \mathbf{0} \\ \mathbf{v}_1^\top \end{pmatrix}) + \begin{pmatrix} \mathbf{0} \\ \mathbf{v}_2^\top \end{pmatrix} = \overline{\mathbf{B}}\mathbf{R} + \begin{pmatrix} \mathbf{0} \\ \mathbf{v}_1^\top + \mathbf{v}_2^\top \end{pmatrix} = \mathsf{hct}(\mathbf{v}_1 + \mathbf{v}_2).$$

  – **HMult:** Ciphertext for bits are multiplicative homomorphic.

$$\mathsf{hct}(b_1) \boxtimes \mathsf{hct}(b_2) = \mathsf{hct}(b_1) \cdot \mathbf{G}^{-1}(\mathsf{hct}(b_2)) = \overline{\mathbf{B}}\mathbf{R}_\times + b_1 b_2 \mathbf{G},$$

  where $\mathbf{R}_\times = \mathbf{R}_1\mathbf{G}^{-1}(\mathsf{hct}(b_2)) + b_1\mathbf{R}_2$.

  – **Packing:** Ciphertext for bits can be packed to ciphertext for vectors. Given a vector $\mathbf{v} \in \mathbb{Z}_q^m$ with bitwise representation $\mathbf{v}^\top = \sum 2^t(v_{1,t}, \ldots, v_{m,t})$,

$$\begin{aligned} \mathsf{Pack}(\{\mathsf{hct}(v_{i,t})\}) &= \sum \mathsf{hct}(v_{i,t})\mathbf{G}^{-1}(2^t \cdot \mathbf{1}_{n+1,i}) \\ &= \overline{\mathbf{B}}\left(\sum \mathbf{R}_{i,t}\mathbf{G}^{-1}(2^t \cdot \mathbf{1}_{n+1,i})\right) + \begin{pmatrix} \mathbf{0} \\ \mathbf{v}^\top \end{pmatrix} = \mathsf{hct}(\mathbf{v}), \end{aligned}$$

  where $\mathbf{1}_{n+1,i}$ is the unit matrix which is one at the $(n+1, i)$-th coordinate and 0 elsewhere.

  With these homomorphic operation, the scheme evaluates $f(\mathbf{x}, \mathbf{v}) = L(C(\mathbf{x}), \mathbf{v})$ as follows:

  1. Evaluate $\mathsf{hct}(\mathbf{x}) \to \mathsf{hct}(\mathsf{bits}(C(\mathbf{x})))$ using HAdd and HMult.
  2. Pack $\mathsf{hct}(\mathsf{bits}(C(\mathbf{x})))$ into $\mathsf{hct}(C(\mathbf{x}))$,
  3. Evaluate $\mathsf{hct}(C(\mathbf{x}), \mathbf{v}) \to \mathsf{hct}(L(C(\mathbf{x}), \mathbf{v}))$ using HAdd.

- $\mathsf{Dec}(\mathsf{hsk}, \mathsf{hct})$: Ciphertext for vectors can be linearly decrypted by

$$((-\mathbf{r}^\intercal, 1) \cdot \mathsf{hct}(\mathbf{v}))^\intercal = \left((-\mathbf{r}^\intercal, 1)\left(\begin{pmatrix}\mathbf{B} \\ \mathbf{r}^\intercal\mathbf{B} + \mathbf{e}^\intercal\end{pmatrix}\mathbf{R} + \begin{pmatrix}\mathbf{0} \\ \mathbf{v}^\intercal\end{pmatrix}\right)\right)^\intercal = (\mathbf{v}^\intercal + \mathbf{e}^\intercal\mathbf{R})^\intercal \approx \mathbf{v}.$$

The GSW scheme satisfies all following desired properties listed in the previous section.

Approximate Decryption Correctness The decryption error of the output ciphertext

$$\mathsf{ct}_f = \overline{\mathbf{B}}\mathbf{R}_f \boxplus f(\mathbf{x}) \leftarrow \mathsf{Eval}(\mathsf{hct}(\mathbf{x}), f)$$

has norm bounded by $\left\|\mathbf{e}^\intercal\mathbf{R}_f\right\| \le m^{O(d)}\|\mathbf{e}\| = m^{O(d)}O(\lambda\sigma)$.

Pseudorandom Public Key and Ciphertext Assuming (sub-exponential) LWE and appropriate $m = \Omega(n \log q)$, the GSW scheme has (sub-exponential) pseudorandom public key and ciphertext.

Rerandomizable By a standard smudging argument (lemma 3), the GSW scheme is statistically rerandomizable using fresh ciphertext with randomness from large gaussian $\mathcal{D}_{\mathrm{rand}} = \mathcal{D}_{\sigma_0}^{m \times \ell m'}$. Namely,

$$(\overline{\mathbf{B}}\mathbf{R}_f \boxplus \mathbf{v}) + \overline{\mathbf{B}}\mathbf{R}_0 \approx_s^{2^{-\lambda}} \overline{\mathbf{B}}\mathbf{R}_0 \boxplus \mathbf{v} \quad \text{when } \mathbf{R}_0 \leftarrow \mathcal{D}_\sigma^{m \times \ell m'} \text{ and } \sigma \ge 2^\lambda\left\|\mathbf{R}_f\right\|.$$

Equivocal mode For appropriate $m = \Omega(n \log q)$, random GSW public keys $\overline{\mathbf{B}}$ are 2^{-n}-close to matrices sampled with trapdoor (lemma 5). Furthermore, given any ciphertext $\mathsf{hct} \in \mathcal{C}^\ell = \mathbb{Z}_q^{(n+1)\times \ell m}$, one can sample encryption randomness $\mathbf{R}^*$ satisfying $\overline{\mathbf{B}}\mathbf{R}^* = \mathsf{hct}$ from the conditional Gaussian distribution using the trapdoor of $\overline{\mathbf{B}}$ (with up to 2^{-n} statistical error).

2.2 Assumption Formulation

In this section, we present our assumption. In order to ease the exposition and build intuition towards our assumption, we introduce it in two stages. Each stage will be associated with a key concept central to our assumption.

2.2.1 Building Intuition via Abstract Formulation

We will first describe our assumption in two stages, in an abstract way w.r.t. a homomorphic encryption scheme according to Definition 2. We believe the abstract versions better convey the rationale (without the burden of concrete algebra). We emphasize that the abstract versions are only for exposition, our actual assumption is w.r.t. the concrete GSW HE scheme [GSW13].

Stage 1: Circular Security. Our first stage is simply a variant of well-studied circular security of the Homomorphic Encryption (HE) scheme in the context of Bootstrapping [Gen09]. The assumption f^{circ}-circular security is described in Figure 1. The assumption posits indistinguishability between a real and an ideal distribution. Let $\mathbb{Z}_q$ be the ambient space of the HE ciphertexts and the LWE samples, and n the dimension of HE secret keys and LWE secret vectors. In the real distribution, we have two main components. The first component consists of an honestly homomorphic encryption public key hpk generated using secret key $\mathbf{r} \in \mathbb{Z}_q^n$, a circularly encrypted ciphertext hct encrypting the secret key $\mathbf{r}$ and a fresh encryption hct_0 of zeros. The second component simply consists of LWE samples with respect to random matrices $\mathbf{A}$ and $\mathbf{D}$ encoding secret related terms. The first LWE sample ct_1 with respect to $\mathbf{A}$ uses independent secret $\mathbf{U}$ and encodes a matrix $\mathbf{I}_\ell \otimes \mathbf{G}_n^\top \mathbf{r}$ that depends on $\mathbf{r}$, where n is the dimension of the secret key $\mathbf{r}$ and ℓ is an integer parameter polynomial in n and should be thought of as much larger than n. The second sample ct_2 with respect to $\mathbf{D}$ uses $\mathbf{r}$ as the secret vector and encodes a $\mathbb{Z}_q$-vector-valued function f^{circ} on $\mathbf{U}$ and hct_0. The distribution outputs $(\mathsf{hpk}, \mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)$.

The ideal distribution is exactly the same except that all the components are now replaced with randomly chosen vectors of appropriate size over $\mathbb{Z}_q$.

f^{circ}-circular security

The assumption is parameterized by a polynomial sized circuit f^{circ} with domain/codomain implicitly defined below.

$\mathcal{D}_0$: **Real distribution**	$\mathcal{D}_1$: **Ideal distribution**
HE Components: • Secret key $\mathbf{r} \leftarrow \mathbb{Z}_q^n$ • Public key $\mathsf{hpk} \leftarrow \mathsf{KeyGen}(\mathbf{r})$ • Ciphertext $\mathsf{hct} \leftarrow \mathsf{HE.Enc}(\mathsf{hpk}, \mathbf{r}; \mathcal{D}_{\mathrm{enc}})$ • Mask $\mathsf{hct}_0 \leftarrow \mathsf{HE.Enc}(\mathsf{hpk}, \mathbf{0}^{M'}; \mathcal{D}_{\mathrm{rand}})$	**HE Components:** • Public key $\mathsf{hpk} \leftarrow \$$ • Ciphertext $\mathsf{hct} \leftarrow \$$ • Mask $\mathsf{hct}_0 \leftarrow \$$
LWE Components: • Public matrices $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times \ell}, \mathbf{D} \leftarrow \mathbb{Z}_q^{n \times M}$ • $\mathsf{ct}_1 \leftarrow \utilde{\mathbf{U}^\top \mathbf{A} + \mathbf{I}_\ell \otimes \mathbf{G}_n^\top \mathbf{r}}$, where $\mathbf{U} \leftarrow \mathbb{Z}_q^{n \times \ell n \lceil \log q \rceil}$ • $\mathsf{ct}_2 \leftarrow \utilde{\mathbf{r}^\top \mathbf{D} + f^{\mathrm{circ}}(\mathbf{U}, \mathsf{hct}_0)}$	**LWE Components:** • Public matrices $\mathbf{A}, \mathbf{D} \leftarrow \$$ • $\mathsf{ct}_1 \leftarrow \$$ • $\mathsf{ct}_2 \leftarrow \$$
Output: $(\mathsf{hpk}, \mathsf{enc})$, where encoding $\mathsf{enc} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_0, \mathsf{ct}_1)$	

Figure 1: f^{circ}-circular security of HE scheme. We use the curly notation $\utilde{\cdot\cdot}$ to hide the noise terms.

We note that by inspection, only looking at the HE component corresponds to the standard circular security assumption used for bootstrapping GSW homomorphic encryption. Similarly, examining the LWE components independently (ignoring the dependence on the public ciphertext hct_0 for the time-being) is close to the standard 2-circular security assumption. Our assumption posits indistinguishability of these two components together. We discuss cryptanalysis of this shortly in Section 2.3.2.

Stage 2: Security with Re-Randomized Opening. As such, the first assumption is not useful for iO because one can never use these circularly encrypted ciphertexts to learn outputs securely in the clear. Our next assumption modifies the circular security assumption in a way that allows us to securely learn the outputs.

We consider the assumption, denoted as CRO, to be the f^{circ}-circular security (Figure 1) with an additional *opening* component described in Figure 2. Since we work with the GSW encryption scheme, we provide a concrete version of this assumption in Figure 3.

In this assumption, we consider a function f (with potentially $\mathbb{Z}_q$-vector outputs) to be homomorphically evaluated. One computes $\mathsf{hct}_f = \mathsf{Eval}(f, \mathsf{hct})$. The assumption aims to securely open the randomness to hct_f that allows one to learn $f(\mathbf{r})$.

One intuitive way to achieve this would be to release randomness $\mathbf{R}_f$ such that $\mathsf{hct}_f = \mathsf{HE.Enc}(\mathsf{hpk}, f(\mathbf{r}); \mathbf{R}_f)$. Such a randomness can be computed as a deterministic function of the randomness used in the initial ciphertexts used to compute hct_f relying on the well-known randomness homomorphism structure in the GSW encryption scheme. Unfortunately, leaking $\mathbf{R}_f$ this was could jeopardize security as it might have a structure that enables leaking sensitive information. In fact, leveraging the prior attack techniques developed in [HJL21] one might be able to show explicit attacks. Alternatively, one might try to release a random Gaussian opening $\tilde{\mathbf{R}}_f$ subject to the equation $\mathsf{hct}_f = \mathsf{HE.Enc}(\mathsf{hpk}, f(\mathbf{r}); \tilde{\mathbf{R}}_f)$, hoping that the additional randomness helps with security. This can be done, for example, by relying on a trapdoor matrix for the LWE coefficient matrix used to generate hpk. Unfortunately, here too, one could find structural vulnerabilities enabling explicit attacks. [4]

A reasonable approach to handle this is to introduce some sort of shield (as also considered by Gay and Pass [GP21]). Namely, we consider a fresh ciphertext hct_0 that encrypts 0, encrypted with randomness sampled from a special re-randomizing randomness distribution $\mathcal{D}_{\text{rand}}$ capable of smudging the evaluated randomness inside hct_f (see the rerandomizability property of HE, Definition 3). For GSW, the distribution $\mathcal{D}_{\text{rand}}$ consists of i.i.d. samples from a wide enough Gaussian distribution. Then, one can release an opening of the re-randomized ciphertext $\mathsf{hct}^*_f = \mathsf{hct}_f + \mathsf{hct}_0$. The opening is simply $\mathbf{R}^* = \mathbf{R}_f + \mathbf{R}_0$.

If the function $f(\star)$ did not depend on the secret, and in addition there were no circular encryptions of the secrets in the real distribution, the security of such a distribution can be proved under LWE (see [GP21] for details). Note that however, since the ciphertext hct encrypts the secret $\mathbf{r}$, one has to be careful on which functions $f(\star)$ should be allowed to learn. Therefore, for the assumption to make sense, we consider functions $f(\star)$ whose output can be computed publicly. We capture this by the safety constraint in (1), where we require that in the real distribution, with high probability $f(\mathbf{r}) = \tilde{f}(\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)$ for an efficient function $\tilde{f}$. This means revealing the function output is benign, since it can already be computed efficiently from the encodings themselves.

This describes the *real* distribution in Figure 3. Namely, the distribution consists of HE and LWE encodings along with fresh encryption hct_0 and the opening $\mathbf{R}^*$. We note that $f(\star)$ is a function that satisfies a safety constraint outlined in Figure 2. Namely, the constraint requires that with overwhelming probability $f(\mathbf{r}) = \tilde{f}(\mathsf{enc})$ where enc contains $(\mathsf{hpk}, \mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)$ for some efficient functions $f, \tilde{f}$. One can also observe that instead of releasing $\mathbf{R}^* = \mathbf{R}_0 + \mathbf{R}_f$, we release $\mathbf{R}^*$ subject to $\mathsf{hct}^*_f = \mathsf{HE.Enc}(\mathsf{hpk}, \mathbf{0}; \mathbf{R}^*) \boxplus f(\mathbf{r}) \overset{\text{w.h.p.}}{=} \mathsf{HE.Enc}(\mathsf{hpk}, \mathbf{0}; \mathbf{R}^*) \boxplus \tilde{f}(\mathsf{enc})$. For the GSW encryption scheme, the distribution of the openings generated deterministically as $\mathbf{R}_0 + \mathbf{R}_f$ and through random sampling are statistically close so long as $\mathcal{D}_{\text{rand}}$ is a wide-enough Gaussian. This choice is made as it syntactically unifies our presentation of the real and ideal distributions.

At this point, we remark that Gay and Pass also proposed an assumption that gave rise to similar structures, but as we point out in Section 2.3.3 there are major differences. Notably, in our case the function $f(\star)$ is independent of hct_0. This causes $\mathbf{R}_f$ and $\mathbf{R}_0$ to be independent. As a consequence

[4] The attack works by leveraging ideas inspired by [HJL21]. Namely, the attack exploits that such an $\widetilde{\mathbf{R}}_f$ must satisfy the equation $\mathbf{e}\widetilde{\mathbf{R}}_f = \mathbf{e}\mathbf{R}_f$ where $\mathbf{e}$ is the error vector in the LWE sample contained inside hpk. This equation can be used to infer non-trivial information about $\mathbf{R}_f$. We omit the details as they are not very central to the proposal in this paper.

Opening procedure $\mathsf{Open}(f, \widetilde{f}, \cdot)$

Constraint: The opening is parameterized with two efficiently computable function f and $\widetilde{f}$ where f is in the function class supported by HE. f maps tuple $(\mathbf{r}, \mathbf{A}, \mathbf{D})$ to $\mathcal{R}^{M'}$, while $\widetilde{f}$ maps enc to $\mathcal{R}^{M'}$. The procedure is only defined if $(f, \widetilde{f})$ satisfies the following constraint.

$$\textbf{safety constraint:} \quad \Pr[f(\mathbf{r}, \mathbf{A}, \mathbf{D}) = \widetilde{f}(\mathsf{enc})] \geq 1 - \epsilon(\lambda), \tag{1}$$

where the probability is taken over the sampling of $(\mathbf{r}, \mathbf{A}, \mathbf{D}, \mathsf{enc})$ according to distribution $\mathcal{D}_0$. We require ϵ to be negligible when considering polynomial security of CRO, and require $\epsilon = 2^{-\lambda^{\delta}}$ when considering sub-exponential security.

Procedure $\mathsf{Open}(f, \widetilde{f}, (\mathsf{hpk}, \mathsf{enc}))$:
Parse $\mathsf{enc} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_0, \mathsf{ct}_1)$, the opening is computed as follow.

- Perform homomorphic evaluation over hct to get $\mathsf{hct}_f = \mathsf{HE.Eval}(\mathsf{hct}, f_{\mathbf{A},\mathbf{D}})$, where function $f_{\mathbf{A},\mathbf{D}}(\cdot) = f(\cdot, \mathbf{A}, \mathbf{D})$.

  Comment: When enc *follows the real distribution $\mathcal{D}_0$, since* hct *is an honest encryption of* $\mathbf{r}$, *by correctness of* HE, hct_f *is a valid encryption of* $f_{\mathbf{A},\mathbf{D}}(\mathbf{r}) = f(\mathbf{r}, \mathbf{A}, \mathbf{D})$. *Following the safety constraint (Equation (1)),* hct_f *is with overwhelming probability a valid encryption of* $\widetilde{f}(\mathsf{enc})$.

- Rerandomize ciphertext using the mask $\mathsf{hct}_f^* = \mathsf{hct}_f + \mathsf{hct}_0 \boxminus \widetilde{f}(\mathsf{enc})$.

  Comment: When enc *follows the real distribution $\mathcal{D}_0$, by the rerandomizable property of* HE *(Definition 3), the distribution of* hct_f^* *is statistically close to* $\mathsf{HE.Enc}(\mathsf{hpk}, \mathbf{0}^{M'}; \mathcal{D}_{\mathrm{rand}})$.

- Sample a random opening $\mathbf{R}^*$ of hct_f^* with respect to hpk, i.e.,

$$\mathbf{R}^* \leftarrow \mathcal{D}_{\mathrm{rand}}|_{\mathsf{hct}_f^* = \mathsf{HE.Enc}(\mathsf{hpk}, \mathbf{0}^{M'}; \mathbf{R}^*)}.$$

  Comment: When enc *follows the real distribution $\mathcal{D}_0$, by the above discussion, the marginal distribution of* $\mathbf{R}^*$ *is statistically close to* $\mathcal{D}_{\mathrm{rand}}$ *(Theorem 1). When* enc *follows the ideal distribution $\mathcal{D}_1$, by the equivocation property of* HE *(Definition 4) and the* f^{circ}*-circular security of HE (Figure 1), the marginal distribution of* $\mathbf{R}^*$ *is pseudorandom (Theorem 2).*

Figure 2: $(f, \widetilde{f})$-opening for f^{circ}-circularly secure HE scheme. We note that though the opening procedure below is not necessarily efficient, it is possible to efficiently sample its output $\mathbf{R}^*$ together with $\mathcal{D}_0$ or $\mathcal{D}_1$.

$\mathbf{R}^* = \mathbf{R}_f + \mathbf{R}_0$ behaves like a standard Gaussian matrix over integers. In the assumption proposed by Gay and Pass, homomorphic evaluation of f can depend on hct_0, leading $\mathbf{R}^*$ to have an extractable bias, allowing for an efficient distinguisher in the assumption for a properly chosen f [HJL21].

We now describe the *ideal* distribution in our assumption. Correspondingly, the new ideal distribution contains $(\mathsf{hpk}, \mathsf{enc}) \leftarrow \mathcal{D}_1$ together with $\mathbf{R}^* \leftarrow \mathsf{Open}(f, \widetilde{f}, (\mathsf{hpk}, \mathsf{enc}))$. Here, both hpk, enc are sampled as uniformly random matrices in their co-domains as opposed to being generated honestly. While we do this, we make sure that the Open procedure is still well defined.

In the case, the intermediate ciphertexts hct_f and hct^*_f are now computed efficiently from random public key hpk and random encodings enc by using the homomorphic evaluation procedures. The opening $\mathbf{R}^*$ is now sampled so that it satisfies the equation $\mathsf{hct}_{f^*} = \mathsf{HE.Enc}(\mathsf{hpk}, \mathbf{0}; \mathbf{R}^*) \boxplus \widetilde{f}(\mathsf{enc})$. One might wonder why can this be done? We note that when the public keys are random, GSW enjoys an equivocal mode as defined in Definition 4 which guarantees such pre-images that can be sampled using a trapdoor matrix for hpk. We also ensure that $\mathbf{R}^*$ is sampled according to a discrete Gaussian of the same width as in the case of real distribution.

2.2.2 Concrete Assumptions

We consider circular security with random opening assumption specifically with respect to the GSW encryption scheme. We formulate three versions. First, a parameterized assumption w.r.t. some appropriate tuple $(f^{\text{circ}}, f, \widetilde{f})$, referred to as the $(f^{\text{circ}}, f, \widetilde{f})$-CRO assumptions. Next, in quest of identifying a fully-specified assumption sufficient for iO, we provide a completely specified single assumption with concrete $(f^{\text{circ}}, f, \widetilde{f})$ needed for our iO construction later. Throughout, this is referred to as the CRO assumption that we use. In section 6, we weaken the pseudorandomness requirement in CRO above and propose a weaker indistinguishability version, referred to as the IND-CRO assumption.

The $(f^{\text{circ}}, f, \widetilde{f})$-CRO assumptions For a tuple of appropriate functions $(f^{\text{circ}}, f, \widetilde{f})$, with appropriate domains/co-domains, and satisfying the safety constraint (Equation (1)), the abstract assumption instantiated with GSW, which is an HE scheme satisfying all needed properties (see Section 2.1.2), gives the following assumption.

Definition 6 ($(f^{\text{circ}}, f, \widetilde{f})$-Circular Security with Random Opening (CRO) Assumption)**.** *Let λ be the security parameter. Let $n, m, d, k, \ell, M, \sigma$ be integer parameters that are polynomial in λ, and q, σ_0 be (potentially superpolynomial) integer parameters where $m = \Omega(n \log q)$ and $\sigma_0 = 2^\lambda m^{\Omega(d)}$ are sufficiently large. Let $f \in \mathcal{F}_{d,M}$ be a bounded depth packed circuit (definition 5) which parses its input as bits and have depth bound d and output length M, where M w.l.o.g. is a multiple of $(n+1)\lceil \log q \rceil$[5], and f^{circ} and $\widetilde{f}$ be efficiently computable functions with domain/codomain implicitly defined in Figure 3.*

We say that the (subexponential) $(f^{\text{circ}}, f, \widetilde{f})$-CRO assumption holds if $\mathcal{D}_0$ and $\mathcal{D}_1$ in Figure 3 are (sub-exponentially) indistinguishable to all polynomial time attackers.

$$\begin{aligned}
&\{(\mathsf{hpk}, \mathsf{enc} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2), \mathsf{hint} = \mathbf{R}^*) \mid (\mathsf{hpk}, \mathsf{enc}, \mathsf{hint}) \leftarrow \mathcal{D}_0\}_\lambda \\
\approx &\{(\mathsf{hpk}, \mathsf{enc} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2), \mathsf{hint} = \mathbf{R}^*) \mid (\mathsf{hpk}, \mathsf{enc}, \mathsf{hint}) \leftarrow \mathcal{D}_1\}_\lambda
\end{aligned}$$

Next, we provide the fully-specified version (referred to as the CRO assumption) that is needed for our iO construction. The main difference between these assumptions is that this assumption is a particular instantiation of the previous assumption working with specific parameters (such as modulus, dimension, etc.) and specific functions $(f^{\text{circ}}, f, \widetilde{f})$ satisfying the safety constraint, as needed by our iO construction. Note that our iO construction needs to use this assumption for a *single* choice of $(f^{\text{circ}}, f, \widetilde{f})$. These functions do not depend on which circuit is being obfuscated, but only on the input/output length and the circuit size.

The CRO assumption In fact, for our construction of iO, it suffices to assume the CRO assumption for specific tuples of functions. By default, CRO-assumption refers to this version.

[5] One can always append zeros to the function output to make M a multiple of $(n+1)\lceil \log q \rceil$. This is a technicality due to the interface of GSW that we formalized in accordance with the abstract definition of HE, requiring that when encrypting a $\mathbb{Z}_q$ vector, the length of the vector is $(n+1)\lceil \log q \rceil$.

$(f^{\text{circ}}, f, \widetilde{f})$-Circular Security with Random Opening

Real Distribution $\mathcal{D}_0$ / Ideal Distribution $\mathcal{D}_1$

HE (GSW) Components (Real):

- $\mathbf{r} \leftarrow \mathbb{Z}_q^n$.
- $\mathsf{hpk} = \overline{\mathbf{B}} = \begin{pmatrix} \mathbf{B} \\ \mathbf{r}^\intercal \mathbf{B} + \mathbf{e}^\intercal \end{pmatrix}$, $\mathbf{B} \leftarrow \mathbb{Z}_q^{n \times m}$, $\mathbf{e} \leftarrow \mathcal{D}_\sigma^m$.
- $\mathsf{hct} = \overline{\mathbf{B}}\mathbf{R} + \mathsf{bits}(\mathbf{r})^\intercal \otimes \mathbf{G}_{n+1}$, $\mathbf{R} \leftarrow \{0,1\}^{m \times n(n+1)\lceil \log q \rceil^2}$.
- $\mathsf{hct}_0 = \overline{\mathbf{B}}\mathbf{R}_0$, $\mathbf{R}_0 \leftarrow \mathcal{D}_{\sigma_0}^{m \times M}$.

LWE Components (Real):

- $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times \ell}, \mathbf{D} \leftarrow \mathbb{Z}_q^{n \times k}$.
- $\mathsf{ct}_1 = \mathbf{U}^\intercal \mathbf{A} + \mathbf{E}_\mathbf{A} + \mathbf{I}_\ell \otimes \mathbf{G}_n^\intercal \mathbf{r}$, $\mathbf{U} \leftarrow \mathbb{Z}_q^{n \times \ell n \lceil \log q \rceil}$, $\mathbf{E}_\mathbf{A} \leftarrow \mathcal{D}_\sigma^{\ell n \lceil \log q \rceil \times \ell}$.
- $\mathsf{ct}_2 = \mathbf{r}^\intercal \mathbf{D} + \mathbf{e}_\mathbf{D}^\intercal + f^{\text{circ}}(\mathbf{U}, \mathsf{hct}_0)$, $\mathbf{e}_\mathbf{D} \leftarrow \mathcal{D}_\sigma^k$

HE (GSW) Components (Ideal):

- $\mathsf{hpk} \leftarrow \mathbb{Z}_q^{(n+1) \times m}$
- $\mathsf{hct} \leftarrow \mathbb{Z}_q^{(n+1) \times n(n+1)\lceil \log q \rceil^2}$.
- $\mathsf{hct}_0 \leftarrow \mathbb{Z}_q^{(n+1) \times M}$.

LWE Components (Ideal):

- $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times \ell}, \mathbf{D} \leftarrow \mathbb{Z}_q^{n \times k}$.
- $\mathsf{ct}_1 \leftarrow \mathbb{Z}_q^{\ell n \lceil \log q \rceil \times \ell}$.
- $\mathsf{ct}_2 \leftarrow \mathbb{Z}_q^{1 \times k}$.

$\mathsf{Open}(f, \widetilde{f}, (\mathsf{hpk}, \mathsf{enc} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)))$:
Functions $(f, \widetilde{f})$ satisfies the safety constraint (1), i.e., with overwhelming probability over the sampling of $(\mathsf{hpk}, \mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)$ according to $\mathcal{D}_0$, it holds that $f(\mathbf{r}, \mathbf{A}, \mathbf{D}) = \widetilde{f}(\mathsf{enc})$.

1. $\mathsf{hct}_f = \mathsf{HE.Eval}(\mathsf{hct}, f_{\mathbf{A},\mathbf{D}}) \overset{\text{in } \mathcal{D}_0}{=} \overline{\mathbf{B}}\mathbf{R}_f \boxplus f(\mathbf{r}, \mathbf{A}, \mathbf{D})$, where function $f_{\mathbf{A},\mathbf{D}}(\cdot) = f(\cdot, \mathbf{A}, \mathbf{D})$.
2. $\mathsf{hct}_f^* = \mathsf{hct}_f \boxplus (-\widetilde{f}(\mathsf{enc})) \boxplus \mathsf{hct}_0 \overset{\text{in } \mathcal{D}_0}{\approx_s} \overline{\mathbf{B}}(\mathbf{R}_f + \mathbf{R}_0)$.
3. $\mathbf{R}^* \leftarrow \mathcal{D}_{\sigma_0}^{m \times k}\Big|_{\mathsf{hct}_f^* = \overline{\mathbf{B}}\mathbf{R}^*}$.

Output: $(\mathsf{hpk}, \mathsf{enc} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2), \mathbf{R}^*)$.

Figure 3: $n, m, q, d, k, \ell, M, \sigma, \sigma_0$ are λ-dependent parameters where $n, m, d, k, \ell, M, \sigma$ are polynomials in λ, while q, σ_0 may be superpolynomial in λ satisfying $(n+1)\lceil \log q \rceil | M$, $m = \Omega(n \log q)$, and $\sigma_0 = 2^\lambda m^{\Omega(d)}$, where m, σ_0 are sufficiently large. Circuit $f \in \mathcal{F}_{d,M}$ is a bounded depth packed circuit (definition 5) with depth bound d and output length M. We assume that f parses its input as bits.

Assumption 1 (Circular Security with Random Opening (CRO) Assumption)**.** *Let λ be the security parameter, and n, q, σ be LWE parameters dependent on λ, where $n = \mathsf{poly}(\lambda)$, $\sigma = \mathsf{poly}(\lambda)$, $q \leq 2^{n^\delta}$ for some constant $\delta \in (0,1)$, q is a multiple of Δ such that $q/\Delta \geq 2^\lambda$, and $\Delta \geq (2n \log q)^\lambda$. The (subexponential) CRO assumption with parameters (n, q, σ, Δ) states that for an appropriate $m = \Theta(n \log q)$, $\sigma_0 = \Delta/2^{\Theta(\lambda)}$, and every efficiently computable polynomials $Q : \mathbb{Z} \to \mathbb{Z}$ and $\ell : \mathbb{Z} \to \mathbb{Z}$, the (subexponential) $(f^{\text{circ}}, f, \widetilde{f})$-CRO assumption holds for the following function tuple.*

$$\mathsf{hct}_0 = \begin{pmatrix} \overline{\mathsf{hct}}_{0,i} \in \mathbb{Z}_q^{n\times\ell} \\ \underline{\mathsf{hct}}_{0,i} \in \mathbb{Z}_q^{1\times\ell} \end{pmatrix}_{i\in[Q]} \qquad \mathbf{D} = \left(\mathbf{D}_i \in \mathbb{Z}_q^{n\times n\lceil \log q\rceil}\right)_{i\in[Q]} \qquad \mathsf{ct}_2 = \left(\mathsf{ct}_{2,i} = \mathbf{r}^\intercal \mathbf{D}_i + \mathbf{e}_{\mathbf{D},i} + f_i^{\text{circ}}(\mathbf{U}, \mathsf{hct}_{0,i})\right)_{i\in[Q]}$$

$$f^{\text{circ}} = \left(f_i^{\text{circ}}\right)_{i\in[Q]} \qquad f = \left(f_i\right)_{i\in[Q]} \qquad \widetilde{f} = \left(\widetilde{f_i}\right)_{i\in[Q]}$$

$$\begin{aligned} f_i^{\text{circ}}(\mathbf{U}, \mathsf{hct}_0) &= -\mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\intercal \cdot (\mathbf{U}^\intercal \mathbf{G}) && \in \mathbb{Z}_q^{1\times n\lceil \log q\rceil} \\ f_i(\mathbf{r}, \mathbf{A}, \mathbf{D})^\intercal &= \Delta \left\lfloor \frac{\mathbf{r}^\intercal \mathbf{D}_i \cdot \mathbf{G}^{-1}(\mathbf{A})}{\Delta} \right\rceil && \in \mathbb{Z}_q^{1\times\ell} \\ \widetilde{f_i}(\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)^\intercal &= \Delta \left\lfloor \frac{\mathsf{ct}_{2,i} \cdot \mathbf{G}^{-1}(\mathbf{A}) + \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\intercal \mathsf{ct}_1 + \underline{\mathsf{hct}}_{0,i}}{\Delta} \right\rceil && \in \mathbb{Z}_q^{1\times\ell} \end{aligned}$$

Note that f is computable by a packed circuit of depth $d = O(\log(n \log q))$. The corresponding distributions $\mathcal{D}_0$, $\mathcal{D}_1$ in Figure 3 have parameters $(n, m = \Theta(n \log q), q, d, k = Qn\lceil \log q\rceil, \ell, M = Q\ell, \sigma, \sigma_0)$.

The CRO assumption is almost fully specified modulo the circuit that implements $f_{\mathbf{A},\mathbf{D}}$[6]. The circuit hard-codes $\mathbf{G}^{-1}(\mathbf{A})$ and $\mathbf{D}$, and performs matrix multiplication, tensor products, modulo q, and rounding. We simply choose canonical circuits implementing these operations.

We show below that $f, \widetilde{f}$ considered in the CRO assumption indeed satisfies the safety constraint. A reader can safely skip the proof of this lemma below without affecting the understanding about the assumption or our construction:

Lemma 6. *The following safety constraint holds w.r.t. the distribution $\mathcal{D}_0$ and functions $(f^{\text{circ}}, f, \widetilde{f})$ specified in Assumption 1.*

$$\textbf{safety constraint:} \quad \Pr[f(\mathbf{r}, \mathbf{A}, \mathbf{D}) = \widetilde{f}(\mathsf{enc})] \geq 1 - 2^{-\Omega(\lambda)}, \tag{2}$$

where the probability is taken over the sampling of $(\mathbf{r}, \mathbf{A}, \mathbf{D}, \mathsf{enc})$ according to $\mathcal{D}_0$.

We defer the proof to section 2.4

Remark 1 (Falsifiability of the CRO assumption)**.** *We remark that the CRO assumption is falsifiable, in particular, the two distributions $\mathcal{D}_b$ can be efficiently sampled in a statistically close way, and the main question is whether the opening $\mathbf{R}^*$ can be sampled efficiently.*

We start with the ideal distribution $\mathcal{D}_1$, where $\mathsf{hpk} = \overline{\mathbf{B}}$ and all encodings $\mathsf{enc} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)$ are random. Using the lattice trapdoor techniques recalled in Lemma 5, the sampler can sample $\overline{\mathbf{B}}$ with a trapdoor $\mathbf{T}$, and use the trapdoor to efficiently sample opening $\mathbf{R}^* \leftarrow \overline{\mathbf{B}}^{-1}(\mathsf{hct}_f^*)$. Therefore, $\mathcal{D}_1$ is efficiently sampleable.

In the real distribution $\mathcal{D}_0$, $\overline{\mathbf{B}}$ contains LWE samples in the last row and does not have trapdoors. Hence we need a different efficient sampling procedure. We claim that $\mathcal{D}_0$ is statistically close to $(\mathsf{hpk}, \mathsf{enc} =$

[6] $(f^{\text{circ}}, f, \widetilde{f})$ is specified as above and the GSW homomorphic evaluation algorithms as in Section 2.1.2

$(\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2), (\mathbf{R}_f + \mathbf{R}_0))$ *which can be sampled efficiently. This follows from the fact that the following ways of sampling* $\mathbf{R}^*$ *are all statistically close:*

$$
\begin{aligned}
&\mathbf{R}^* \leftarrow \mathcal{D}_{\sigma_0}^{m\times k}\Big|_{\mathsf{hct}^*_f = \mathsf{hct}_f \boxplus (-\widetilde{f}(\mathsf{enc})) \boxplus \mathsf{hct}_0 = \overline{\mathbf{B}}\mathbf{R}^*} && [\textit{as in } \mathcal{D}_0]\\
&\mathbf{R}^* \leftarrow \mathcal{D}_{\sigma_0}^{m\times k}\Big|_{\overline{\mathbf{B}}\mathbf{R}_f \boxplus \mathsf{hct}_0 = \overline{\mathbf{B}}\mathbf{R}^*} && [\mathsf{hct}_f \boxplus (-\widetilde{f}(\mathsf{enc})) \overset{\textit{in } \mathcal{D}_0}{\approx_s} \overline{\mathbf{B}}\mathbf{R}_f]\\
&\mathbf{R}'_0 \leftarrow \mathcal{D}_{\sigma_0}^{m\times k} - \mathbf{R}_f\Big|_{\mathsf{hct}_0 = \overline{\mathbf{B}}\mathbf{R}'_0}, \ \mathbf{R}^* = \mathbf{R}'_0 + \mathbf{R}_f && [\textit{identical distribution}]\\
&\mathbf{R}'_0 \leftarrow \mathcal{D}_{\sigma_0}^{m\times k}\Big|_{\mathsf{hct}_0 = \overline{\mathbf{B}}\mathbf{R}'_0}, \ \mathbf{R}^* = \mathbf{R}'_0 + \mathbf{R}_f && [\mathcal{D}_{\sigma_0}^{m\times k} \approx_s \mathcal{D}_{\sigma_0}^{m\times k} - \mathbf{R}_f]\\
&\mathbf{R}^* = \mathbf{R}_0 + \mathbf{R}_f && [\textit{identical distribution } \mathbf{R}_0 \equiv \mathbf{R}'_0]
\end{aligned}
$$

A reader might find some superficial similarities between CRO and two assumptions considered in prior works: Evasive LWE [Wee22, Tsa22, VWW22] and 2-Circ SRL security [GP21]. We stress that there are many important differences in our assumptions that are crucial. It is these differences that make our assumptions provably robust against natural applications of all known attacks applicable to Evasive LWE and 2-Circ SRL security. We discuss these aspects and a detailed comparison in the next section on cryptanalysis.

Before that, we first formally define the weaker circular security assumption implied by CRO so that we can refer to and analyze it later.

CRO Implies the Weaker f^{circ}-Circular Security Assumption As mentioned before, the CRO assumption strengthens a more basic circular security assumption where the distinguisher is given all the LWE encodings without the opening $\mathbf{R}^*$.

Definition 7 (f^{circ}-Circular Security Assumption)**.** *Let $\lambda, n, m, d, k, \ell, M, \sigma, q, \sigma_0$ and f^{circ} be parameters and a function as specified in Definition 6.*

We say that the (subexponential) f^{circ}-circular security assumption holds if $\mathcal{D}_0$ and $\mathcal{D}_1$ without $\mathbf{R}^*$ in Figure 3 are (sub-exponentially) indistinguishable to all polynomial time attackers.

$$
\begin{aligned}
&\{(\mathsf{hpk}, \mathsf{enc} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)) \mid (\mathsf{hpk}, \mathsf{enc}, \mathsf{hint}) \leftarrow \mathcal{D}_0\}_\lambda\\
\approx &\{(\mathsf{hpk}, \mathsf{enc} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)) \mid (\mathsf{hpk}, \mathsf{enc}, \mathsf{hint}) \leftarrow \mathcal{D}_1\}_\lambda
\end{aligned}
$$

Assumption 2 (Base Circular Security Assumption)**.** *The base circular security assumption with respect to parameters (n, q, σ, Δ) satisfying the conditions specified in Assumption 1, states that the f^{circ}-circular security assumption with f^{circ} and additional parameters $(m, d, k, \ell, M, \sigma_0)$ described in Assumption 1 holds.*

The CRO assumption implies the base circular security assumption.

2.3 Cryptanalysis

In this section, we discuss natural avenues to break our assumption, including prior cryptanalytic attempts on related assumptions implying iO and PrO.

As mentioned previously, our assumption falls into the category of **LWE-with-hints** assumptions. Indeed, as described in Figure 3 we have HE and LWE components, in addition to opening $\mathbf{R}^*$ that is referred to as hint in this case. We start by characterizing/summarizing the state of attacks by discussing what parts of the assumption these attacks focus on as well as cryptanalytic techniques. Then, we will understand our assumption in light of this characterization.

2.3.1 Characterization of Prior Attacks

Broadly speaking, all known prior attacks on assumptions fitting into **LWE-with-hints** framework [JLMS19, AJL+19, WW21, GP21, BDGM22, DQV+21, BDJ+24, AKY24] could be characterized into three categories:

- **Type 1 Attacks:** These are attacks that only exploit the structure present in the hints or any immediately derivable leakage from the hint. These typically are the most worrisome kinds of attacks. Most of the previous attacks such as [HJL21, JLLS23] and even the attacks in this work, section 7, on private-coin evasive LWE are of this kind.
- **Type 2 Attacks:** These are attacks that exploit only the structure present in the (circularly encrypted) samples but do not make use of any additional hint.
- **Type 3 Attacks:** These are attacks that make use of hints, together with samples, as well as any auxiliary information, if any, present in the assumption.

These attacks can also be characterized on the basis of cryptanalytic techniques:

- **Algebraic Attacks:** These attacks entail setting up systems of equations and solving them algebraically to recover various secrets involved in the assumption. The attack described by [JLLS23] on the assumption in [DQV+21] was a **Type 1** attack that set linear equations based on the provided hint and recovered the secrets involved.
- **Correlation Attacks:** These are attacks that examine the hint or immediately derivable leakage from the hint. These attacks extract a biased bit of information either correlated with secrets or that enables an efficient distinguisher. Attacks on the assumptions considered in previous schemes [WW21, BDGM22, GP21] described in [HJL21] and the attack on private-coin evasive LWE underlying pseudorandom obfuscation [BDJ+24, AKY24] in this work are **Type 1** correlation attacks. Finally, we note that there were simple-to-state, instance-independent and falsifiable **LWE-with-hints** assumptions that implied iO together with Bilinear maps [JLMS19, AJL+19, Agr19]. Earlier versions of these assumptions also had **Type 1** correlation based attacks based on sum-of-squares [BHJ+19].
- **Corner Cases:** Often, a family of assumptions is meant to capture certain security heuristic, such as the evasive LWE assumption family, or the circular security assumption family. An important type of cryptanalysis is searching for corner cases, which are attacks that exploit the freedom in the assumption family, or in coming up with variants of the assumption family, to find broken corner cases. When the corner cases are contrived and/or unnatural, they do not completely invalidate the security heuristic especially in "natural" applications (reminiscent of the random oracle heuristic). Nevertheless, they show how robust a security heuristic is, and we would ideally like to have assumptions that do not have broken corner cases.

  So far, all direct attacks on **LWE-with-hints** assumptions underlying iO or PrO are of **Type 1**. For the broader class of private-coin evasive LWE assumptions which allows for general auxiliary information, there had been **Type 3** attacks [VWW22, BÜW24] that designed complex auxiliary information containing obfuscated programs. The obfuscated programs can for example help utilize the matrix trapdoors in the post-conditions of evasive LWE in a way that cannot be matched in the pre-condition without matrix trapdoors. Note our CRO and IND-CRO assumptions do not allow any auxiliary input.

Furthermore, there had been attacks on circular security of lattice-based schemes that feature unnatural distributions [GKW17b, GKW17a, WZ17]. In a nutshell, these encryption schemes rely on the cycle-tester framework typically instantiated using lockable/compute-and-compare obfuscation.

However, these corner cases are not known to be applicable to prior assumptions towards iO or PrO, nor to our CRO and IND-CRO assumptions, which contain (circular) LWE samples that follow natural distributions, unlike these in the corner cases.

Finally, typically the more freedom an assumption family allows, or the more under-specified an assumption is, the more prone it can be to existence of corner cases. Our CRO and IND-CRO assumptions are fully-specified, thereby, reducing the room for such corner cases.

- **Lattice-Based Attacks:** Finally, an important class of attacks are lattice-based attacks. The goal here is to somehow translate the problem of LWE samples with hints into an efficiently solvable lattice problem. This is an important potential class of attacks, however, when the hints are matrix trapdoors (as in evasive LWE) or openings (as in this work) it is unclear how to use them in lattice attacks beyond the straightforward way – simply using the hints to obtain new LWE samples by multiplying $\overline{\mathbf{B}}$ with the hint, and then attacking the original LWE samples together with these new samples, ignoring the hints. Recall that in CRO and IND-CRO, using the openings in this way only gives $\overline{\mathbf{B}}\mathbf{R}^*$ which can already be efficiently computed from the original LWE encodings, reducing to **Type 2** attacks that ignore the hints.

  In fact, the question whether there are lattice techniques that can significantly speed up attacks on LWE encodings by using matrix trapdoors in a non-straightforward way is a question implicitly posted by the evasive LWE assumptions. As discussed later in Section 2.3.3, despite various attacks on private coin evasive LWE, no such lattice techniques have been developed so far. We believe that this is a highly important question to investigate.

2.3.2 Security Against Attacks

Armed with the characterization of previous attacks and types of attacks, we now discuss the plausibility of our assumption with respect to these attacks.

Resistance against Type - 1 Attacks. Perhaps a major silver lining in our assumption is that one can show provable resistance against **Type 1** attacks. These are typically the most devastating attacks as witnessed in most attacks to recent **LWE-with-hints** assumptions [BHJ+19, HJL21, JLLS23].

As it turns out, our hint $\mathbf{R}^*$ (in the real distribution) as seen in Figure 3 is statistically closely distributed to a canonical discrete Gaussian distribution.

Theorem 1. *In the real distribution $\mathcal{D}_0$ of the $(f^{\mathrm{circ}}, f, \widetilde{f})$-CRO assumption (Figure fig. 3), if $f, \widetilde{f}$ satisfies the (subexponential) safety condition (equation 1), then the marginal distribution of the opening $\mathbf{R}^*$ in $\mathcal{D}_0$ is (subexponentially) statistically-close to a fresh discrete Gaussian $\mathcal{D}_{\sigma_0}^{m\times M}$.*

Furthermore, we can also prove that in the ideal distribution, the hints $\mathbf{R}^*$ are computationally indistinguishable to the same Gaussian distribution, assuming the f^{circ}-circular security assumption described in Figure 1.

Theorem 2. *In the ideal distribution $\mathcal{D}_1$ of the $(f^{\mathrm{circ}}, f, \widetilde{f})$-CRO assumption (Figure fig. 3), if $f, \widetilde{f}$ satisfies the (subexponential) safety condition (equation 1), then assuming the (subexponential) f^{circ}-circular assumption (definition 7), the marginal distribution of the opening $\mathbf{R}^*$ in $\mathcal{D}_1$ is (subexponentially) indistinguishable to a fresh discrete Gaussian $\mathcal{D}_{\sigma_0}^{m\times M}$ for all polynomial-sized adversaries.*

We defer the proof of these statements to Section 2.4.

Moreover, in our assumption there is no additional "immediate natural leakage" enabled by the hint. If $\mathbf{R}^*$ is used in the straightforward way, one can only compute:

$$\mathsf{hct}^*_f = \overline{\mathbf{B}}\mathbf{R}^* = \mathsf{hct}_f \boxplus \mathsf{hct}_0 \boxminus \widetilde{f}(\mathsf{enc}),$$

which is a known function of the LWE/HE encodings. In that sense, our assumption does not produce additional natural leakage, unlike some of the prior assumptions [WW21, DQV+21], and like [GP21].

Resistance against Type - 2 Attacks. Next, we discuss security of just the HE/LWE encodings, i.e., the f^{circ}-circular security. The non-standardness in our LWE samples comes from the fact that they are circularly encoded. If one is not careful with the dependency of various secrets and public matrices, it is easy to construct easy-to-attack distributions.

For example, if only one secret $\mathbf{s}$ is involved, it is problematic to consider samples with the pattern $\{\mathbf{s}\mathbf{A}_1 + f_1(\mathbf{s}, \mathbf{A}_2) + \mathbf{e}_1, \mathbf{s}\mathbf{A}_2 + f_2(\mathbf{s}, \mathbf{A}_1) + \mathbf{e}_2\}$, where f_1, f_2 are efficiently computable “circular" functions. The reason for this is that one could choose $f_1(\mathbf{s}, \mathbf{A}_2) = -\mathbf{s}\mathbf{A}_2$ and $f_2(\mathbf{s}, \mathbf{A}_1) = -\mathbf{s}\mathbf{A}_1$ producing samples that add up to a small norm vector $\mathbf{e}_1 + \mathbf{e}_2$. These counterexamples do not apply when one of f_1, f_2 becomes independent of the coefficient matrices. These problematic patterns would also be an issue in the two-secret settings. For instance, samples of the form $\{\mathbf{s}_1\mathbf{A} + f_1(\mathbf{s}_2, \mathbf{B}) + \mathbf{e}_1, \mathbf{s}_2\mathbf{B} + f_2(\mathbf{s}_1, \mathbf{A}) + \mathbf{e}_2\}$ is prone to the exact same counterexample.

It seems problematic when the dependency on the public matrix is also circular. In the above example, we have function of $\mathbf{B}$ encoded by LWE samples of matrix $\mathbf{A}$, and function of $\mathbf{A}$ circularly encoded by LWE samples of matrix $\mathbf{B}$.

Our assumption, on the other hand, follows a good circular security pattern in the two-secret setting, where the dependency on the public matrices is non-circular. As described in Assumption 1, we have samples of the form $\{\mathbf{s}_2\mathbf{A} + f_0(\mathbf{s}_2) + \mathbf{e}_0, \mathbf{s}_1\mathbf{B} + f_1(\mathbf{s}_2) + \mathbf{e}_1, \mathbf{s}_2\mathbf{C} + f_2(\mathbf{s}_1, \mathbf{A}) + \mathbf{e}_2\}$. Note that the only sample featuring a matrix in the encoded term is the third sample, but crucially, randomness $(\mathbf{C}, \mathbf{e}_2)$ used in this sample is not circularly encoded.

One can further generalize the above case into a circular encoding pattern that seems safe, without known counterexamples. In the single secret setting, one can assign an order to the encodings, such that, the i'th encoding encodes a function $f_i(\mathbf{s}, \{\mathbf{A}_j, \mathbf{e}_j\}_{j<i})$ of the secret $\mathbf{s}$ and randomness $(\mathbf{A}_j, \mathbf{e}_j)$ used in previous encodings $j < i$, using fresh and independent randomness $(\mathbf{A}_i, \mathbf{e}_i)$. That is, we have samples of form $\{\mathbf{s}\mathbf{A}_i + f_i(\mathbf{s}, \{\mathbf{A}_j, \mathbf{e}_j\}_{j<i})\}_{i\in[\ell]}$, where all $\mathbf{A}_i, \mathbf{e}_i$ are randomly sampled. In the case of multiple secrets, the circular functions f_i can depend on all secrets. We leave it as an exciting open question to identify a set of efficient functions obeying this dependency pattern that leads to an efficient attack.

So far, the only circular security counterexamples on lattice-based schemes [GKW17b] make use of specific designs consisting of a cycle-tester framework containing lockable-obfuscation of carefully chosen programs. These structures are absent in our assumption.

Resistance to Type-3 Attacks. As mentioned above, currently there lack cryptanalytic techniques that can leverage the hint $\mathbf{R}^*$ in a non-straightforward way. The only exception is the corner cases of private-coin evasive LWE [Tsa22, VWW22] that leverage complex auxiliary information depending on LWE matrices and/or secrets. Again, our assumption contains no auxiliary information. If $\mathbf{R}^*$ were used in the straightforward way, it produces encodings that can be efficiently computed from the original encodings in the assumption distributions, reducing security to the f^{circ}-circular security. We leave it as an exciting question to find such attacks on our assumption.

2.3.3 Comparison with Previous Assumptions

Our assumption bears some resemblance with two prior assumptions considered in the literature. We discuss these assumptions and mention differences and similarities. We also describe why our assumption might be on firmer foundations compared to these assumptions.

Comparison with Evasive LWE Evasive LWE was introduced by Wee and Tsabary independently in two works [Wee22, Tsa22]. Since then the assumption has been used in myriad works [VWW22, WWW22, HLL23, ARYY23, HLL24, MPV24, CM24, BDJ+24, AKY24] for a variety of applications including witness encryptions and advanced encryption schemes.

In its simplest form, the assumption roughly posits that if a certain pre-condition of the following kind holds:

$$(\mathbf{sB} + \mathbf{e}_1, \mathbf{sP} + \mathbf{e}_2, \mathsf{aux}) \approx_c (\$, \$, \mathsf{aux})$$

Then, the following post-condition holds:

$$(\mathbf{sB} + \mathbf{e}_1, \mathbf{T}, \mathsf{aux}) \approx_c (\$, \mathbf{T}, \mathsf{aux}),$$

where $\mathbf{T}$ is sampled as a random low-norm trapdoor satisfying $\mathbf{BT} = \mathbf{P}$. There are several versions of evasive LWE depending on how the matrices and the secrets are chosen and their relationship with the auxiliary information. We won't be making this distinction for our discussion. We refer the reader to [BÜW24] for a case-study of evasive LWE.

At a superficial level, it might seem that our CRO assumption is similar to evasive LWE in that both postulate indistinguishability of samples in the presence of a trapdoor, however, there are major differences between the two assumptions. First, our assumption does not have a pre-condition at all. It posits indistinguishability of two *fully-specified* distributions. Moreover, there is no additional auxiliary input in our assumption, which is an important source of troubles in defining evasive LWE assumption (see counterexamples of [VWW22, BÜW24]).

Perhaps, most importantly the governing heuristics are different. The evasive LWE assumptions rest upon two heuristics:

- The first heuristic is that the most effective way of leveraging the trapdoor $\mathbf{T}$ is the straightforward way–simply multiplying $\mathbf{T}$ with $\mathbf{sB} + \mathbf{e}_1$ to derive an LWE sample of the form $\mathbf{sP} + \mathbf{e}_1\mathbf{T}$. Therefore, the post-condition reduces to the security of samples $\mathbf{B}, \mathbf{P}, \mathbf{sB} + \mathbf{e}_1, \mathbf{sP} + \mathbf{e}_1\mathbf{T}$.
- Note that the second sample contains structured noise $\mathbf{e}_1\mathbf{T}$, and it's unclear how to reason about its security. The second heuristic is that if $\mathbf{B}, \mathbf{P}, \mathbf{sB} + \mathbf{e}_1, \mathbf{sP} + \mathbf{e}_2$ with fresh noise $\mathbf{e}_2$ is secure, then there is no effective way of leveraging the structured noise $\mathbf{e}_1\mathbf{T}$.

Existing attacks [VWW22, BÜW24] found corner cases of the first heuristic, when there is complex auxiliary information that enables using the trapdoor in a non-straightforward way. However, these corner cases are ad-hoc. So far, there has not been systematic algorithmic advance, such as the development of new lattice techniques, that can make use of trapdoor in a non-straightforward way. In comparison, CRO does not contain auxiliary information, but still relies on the lack of algorithmic techniques that can leverage trapdoors or openings in a non-straightforward way.

Furthermore, in Section 7, we present a new attack on private coin evasive LWE that invalidates the second heuristic behind evasive LWE. Jumping ahead, our attack uses $\mathbf{T}$ in the straightforward way and exploits the structured noise $\mathbf{e}_1^\intercal\mathbf{T}$ to break the assumption.

Since CRO contains opening instead of trapdoor, one can't hope to use $\mathbf{R}^*$ to learn new LWE samples as explained in the introduction. Therefore, a significant distinction is that CRO does not rely on the second heuristic.

Comparison with 2-CircSRL security. Another assumption that is closely related to CRO is the 2-Circ-SRL security introduced by Gay and Pass [GP21]. The assumption is described for completeness in Figure 4. The assumption is also meant to enable releasing secure openings of computed output on circularly encrypted ciphertexts. Moreover, similarly to us, before the opening is released, the evaluated ciphertext is rerandomized by adding to an encryption of 0 (see last step in Figure 4). Also similarly to us, the function f is restricted so that its output is "safe", in the sense of being publicly computable (in the second step in Figure 4, the output α is computed by the attacker).

2-Circ SRL Security

SRL Game G^b:

- Adversaries provides message m_0, m_1.
- Challenger prepares the following 2-circ components of (rerandomizable) homomorphic encryption HE_1 and homomorphic encryption HE_2, including
  - Public key $\mathsf{hpk}_1, \mathsf{hpk}_2$.
  - Circular ciphertext $\mathsf{hct}_1 = \mathsf{HE}_1.\mathsf{Enc}(\mathsf{hsk}_2; R)$, $\mathsf{hct}_2 = \mathsf{HE}_2.\mathsf{Enc}(\mathsf{hsk}_1)$.
  - Mask $\mathsf{hct}_0 = \mathsf{HE}_1.\mathsf{Enc}(0; R_0)$.
  - Challenge ciphertext $\mathsf{hct}_{\mathsf{ch}} = \mathsf{HE}_1.\mathsf{Enc}(m_b)$.
- Adversary specify f, α such that $f(\mathsf{hsk}_2) = \alpha$.
- Challenger computes homomorphic randomness R_f for ciphertext $\mathsf{hct}_f = \mathsf{HE}_1.\mathsf{Eval}(\mathsf{hct}_1, f)$, and releases $R^* = R_f + R_0$.

Figure 4: Two HE schemes are 2-Circ SRL secure if $G^0 \approx_c G^1$.

Beyond these similarities, there are important differences between CRO and 2-Circ-SRL security. In SRL security, the function f and output α are chosen by the attacker, whereas in CRO, f and $\widetilde{f}$ are fixed functions, where $\widetilde{f}$ specifies how the output can be computed from the public encodings. When using SRL to construct iO, the function f depends on the circuit obfuscated and hence SRL is instance dependent, whereas CRO is instance independent.

More importantly, in SRL, both f and α depend on all LWE encodings, including the zero-encryption hct_0 used for re-randomizing the output ciphertext hct_f produced by homomorphic evaluation. When the homomorphic evaluation can depend on hct_0, re-randomization using hct_0 no longer has guarantees.

The crucial distinction in the case of CRO is the function $f(\star)$ is **independent** of the fresh encryption of zero hct_0, and hence $\mathbf{R}_0$. This implies that the opening $\mathbf{R}^* = \mathbf{R}_f + \mathbf{R}_0$ is statistically random because $\mathbf{R}_f$ is independent of $\mathbf{R}_0$ (Theorem 1 and 2). In contrast in SRL, $\mathbf{R}^*$ is not random, as f, and hence $\mathbf{R}_f$, are correlated to $\mathbf{R}_0$. Furthermore, the work of [HJL21] showed that such correlation could be used to make $\mathbf{R}^*$ reveal a secret bit, which led to counterexamples to SRL.

Moreover, the SRL security assumes indistinguishability of circular LWE samples hiding messages m_0 or m_1 (similar to PKE security) at the presence of an opening $\mathbf{R}^*$ revealing output α of function f. At a first glance, it appears LWE-with-hints assumptions that aim at "controlled opening" of LWE samples – revealing some function outputs but nothing else – cannot posit the pseudorandomness of LWE samples, because the correctness condition no longer holds when the

LWE samples were switched to random. Indeed, SRL and other LWE-with-hints assumptions toward iO [WW21, DQV+21] have indistinguishability type.

CRO offers a new way to reason about pseudorandomness of the (circular) LWE samples: It posits that the LWE samples are pseudorandom, if the (random) opening $\mathbf{R}^*$ in the real distribution can be switched to a simulated opening maintaining correctness in the ideal distribution. We believe that this new type of pseudorandomness assumption is interesting.

2.4 Deferred Proofs

Proof of lemma 6. For each $i \in [Q]$ we have

$$\widetilde{f}_i(\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)^\intercal = \Delta \left\lfloor \frac{\mathsf{ct}_{2,i} \cdot \mathbf{G}^{-1}(\mathbf{A}) + \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\intercal \mathsf{ct}_1 + \underline{\mathsf{hct}}_{0,i}}{\Delta} \right\rceil$$

Using the same variable in the description of $\mathcal{D}_0$ and in Assumption 1, with the additional notation that $\mathbf{R}_0 = (\mathbf{R}_{0,i})_{i\in[Q]}$ such that $\mathsf{hct}_{0,i} = \overline{\mathbf{B}}\mathbf{R}_{0,i}$, we can expand each terms in $\widetilde{f}_i$ by

$$\begin{aligned}
\mathsf{ct}_{2,i} \cdot \mathbf{G}^{-1}(\mathbf{A}) &= \left(\mathbf{r}^\intercal \mathbf{D}_i + \mathbf{e}_{\mathbf{D},i} - \mathsf{vec}(\mathbf{G}^{-1}(-\mathbf{B}\mathbf{R}_{0,i}))^\intercal \cdot (\mathbf{U}^\intercal \mathbf{G})\right) \cdot \mathbf{G}_n^{-1}(\mathbf{A}) \\
&= \mathbf{r}^\intercal \mathbf{D}_i \mathbf{G}_n^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{D},i}\mathbf{G}_n^{-1}(\mathbf{A}) - \mathsf{vec}(\mathbf{G}^{-1}(-\mathbf{B}\mathbf{R}_{0,i}))^\intercal \mathbf{U}^\intercal \mathbf{A} \\
\mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\intercal \mathsf{ct}_1 &= \mathsf{vec}(\mathbf{G}^{-1}(-\mathbf{B}\mathbf{R}_{0,i}))^\intercal \left(\mathbf{U}^\intercal \mathbf{A} + \mathbf{E}_\mathbf{A} + \mathbf{I}_\ell \otimes \mathbf{G}_n^\intercal \mathbf{r}\right) \\
&= \mathsf{vec}(\mathbf{G}^{-1}(-\mathbf{B}\mathbf{R}_{0,i}))^\intercal \mathbf{U}^\intercal \mathbf{A} + \mathsf{vec}(\mathbf{G}^{-1}(-\mathbf{B}\mathbf{R}_{0,i}))^\intercal \mathbf{E}_\mathbf{A} - \mathbf{r}^\intercal \mathbf{B}\mathbf{R}_{0,i} \\
\underline{\mathsf{hct}}_{0,i} &= (\mathbf{r}^\intercal \mathbf{B} + \mathbf{e}^\intercal)\mathbf{R}_{0,i} \\
&= \mathbf{r}^\intercal \mathbf{B}\mathbf{R}_{0,i} + \mathbf{e}^\intercal \mathbf{R}_{0,i}
\end{aligned}$$

Therefore,

$$\widetilde{f}_i(\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)^\intercal = \Delta \left\lfloor \frac{\mathbf{r}^\intercal \mathbf{D}_i \mathbf{G}^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{D},i}\mathbf{G}^{-1}(\mathbf{A}) + \mathsf{vec}(\mathbf{G}^{-1}(-\mathbf{B}\mathbf{R}_{0,i}))^\intercal \mathbf{E}_\mathbf{A} + \mathbf{e}^\intercal \mathbf{R}_{0,i}}{\Delta} \right\rceil.$$

Note that $\mathbf{r}, \mathbf{A}, \mathbf{D}$ are all sampled at random, therefore the marginal distribution of $\mathbf{r}^\intercal \mathbf{D}_i \mathbf{G}^{-1}(\mathbf{A})$ is random in $\mathbb{Z}_q^{1\times\ell}$. Furthermore, by the Gaussian tail bound (lemma 2), with probability $1 - 2^{-\Omega(\lambda)}$ the noise term would have norm bounded by

$$\left\|\mathbf{e}_{\mathbf{D},i}\mathbf{G}^{-1}(\mathbf{A}) + \mathsf{vec}(\mathbf{G}^{-1}(-\mathbf{B}\mathbf{R}_{0,i}))^\intercal \mathbf{E}_\mathbf{A} + \mathbf{e}^\intercal \mathbf{R}_{0,i}\right\| \le \sqrt{\lambda}\ell\sigma + \sqrt{\lambda}\ell n\lceil \log q\rceil\sigma + \sqrt{\lambda}\sigma\sigma_0 = \mathsf{poly}(\lambda)\sigma_0.$$

Combined with the setup that $\sigma_0 = \Delta/2^{\Theta(\lambda)}$, we can apply the rounding lemma 4 to conclude that

$$\begin{aligned}
\widetilde{f}_i(\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)^\intercal &= \Delta \left\lfloor \frac{\mathbf{r}^\intercal \mathbf{D}_i \mathbf{G}^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{D},i}\mathbf{G}^{-1}(\mathbf{A}) + \mathsf{vec}(\mathbf{G}^{-1}(-\mathbf{B}\mathbf{R}_{0,i}))^\intercal \mathbf{E}_\mathbf{A} + \mathbf{e}^\intercal \mathbf{R}_{0,i}}{\Delta} \right\rceil \\
&\overset{\text{w.h.p.}}{=} \Delta \left\lfloor \frac{\mathbf{r}^\intercal \mathbf{D}_i \mathbf{G}^{-1}(\mathbf{A})}{\Delta} \right\rceil \\
&= f_i(\mathbf{r}, \mathbf{A}, \mathbf{D})^\intercal
\end{aligned}$$

where the second equality holds with probability $1 - 2^{-\Omega(\lambda)}$. Finally, with a union bound over $i \in [Q]$, we conclude that the safety constraint holds with probability $1 - 2^{-\Omega(\lambda)}$. □

Proof for Theorem 1. Observe that in $\mathcal{D}_0$, the opening is sampled by $\mathbf{R}^* \leftarrow \mathcal{D}_{\mathrm{rand}}^{m\times M}|_{\mathsf{hct}_f^* = \overline{\mathbf{B}}\mathbf{R}^*}$, which expands to

$$\mathbf{R}^* \leftarrow \mathcal{D}_{\mathrm{rand}}^{m\times M}|_{\overline{\mathbf{B}}\mathbf{R}^* = \mathsf{hct}_f \boxplus (-\widetilde{f}(\mathsf{enc})) \boxplus \mathsf{hct}_0},$$

where the right hand side of the condition can be further expanded by

$$\mathsf{hct}_f \boxplus (-\widetilde{f}(\mathsf{enc})) \boxplus \mathsf{hct}_0 = \overline{\mathbf{B}}(\mathbf{R}_f + \mathbf{R}_0) \boxplus (f(\mathbf{r}, \mathbf{A}, \mathbf{D}) - \widetilde{f}(\mathsf{enc})).$$

By the safety constraint, $f(\mathbf{r}, \mathbf{A}, \mathbf{D}) - \widetilde{f}(\mathsf{enc})$ is zero for probability $1 - \epsilon(\lambda)$. Furthermore, the matrix $\mathbf{R}_f$ has norm bound $\|\mathbf{R}_f\| = m^{O(d)}$, implying that $\sigma_0 > 2^{\lambda}\|\mathbf{R}_f\|$. Therefore by the smudging lemma (lemma 3), the sum of variables $(\mathbf{R}_f + \mathbf{R}_0)$ distributes $2^{-\Omega(\lambda)}$-close to $\mathbf{R}_0 \leftarrow \mathcal{D}_{\sigma_0}^{m\times M}$. Therefore, the marginal distribution of $\mathbf{R}^*$ is $\left(2^{-\Omega(\lambda)} + \epsilon(\lambda)\right)$-close to

$$\mathbf{R}^* \leftarrow \mathcal{D}_{\mathrm{rand}}^{m\times M}|_{\overline{\mathbf{B}}\mathbf{R}^* = \overline{\mathbf{B}}\mathbf{R}_0}, \quad \text{where } \mathbf{R}_0 \leftarrow \mathcal{D}_{\sigma_0}^{m\times M},$$

which gives exactly the discrete Gaussian distribution $\mathcal{D}_{\sigma_0}^{m\times Lm'}$. □

Proof for Theorem 2. We consider the following sequence of hybrids.

- $\mathcal{H}_0$: This is the marginal distribution of $\mathbf{R}^*$ in the ideal distribution $\mathcal{D}_1$. The opening is sampled by

  $$\mathbf{R}^* \leftarrow \mathcal{D}_{\mathrm{rand}}^{m\times M}|_{\overline{\mathbf{B}}\mathbf{R}^* = \mathsf{hct}_f \boxplus (-\widetilde{f}(\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)) \boxplus \mathsf{hct}_0},$$

  where $\overline{\mathbf{B}}, \mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2 \leftarrow \$$

- $\mathcal{H}_1$: This hybrid replaces the random matrix $\overline{\mathbf{B}}$ with a matrix sampled with trapdoor $(\mathbf{B}_0, \mathbf{T}) \leftarrow \mathsf{TrapGen}(1^{n+1}, q, m)$ (variable name changed for clarity). The opening is sampled by

  $$\mathbf{R}^* \leftarrow \mathcal{D}_{\mathrm{rand}}^{m\times M}|_{\mathbf{B}_0\mathbf{R}^* = \mathsf{hct}_f \boxplus (-\widetilde{f}(\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)) \boxplus \mathsf{hct}_0}.$$

- $\mathcal{H}_2$: This hybrid samples $\mathbf{R}^*$ efficiently using the trapdoor, i.e.,

  $$\mathbf{R}^* \leftarrow \mathsf{SampPre}(\mathbf{B}_0, \mathbf{T}, \mathsf{hct}_f \boxplus (-\widetilde{f}(\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)) \boxplus \mathsf{hct}_0, \sigma_0).$$

- $\mathcal{H}_3$: This hybrid samples $\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2$ following the real distribution $\mathcal{D}_0$ of the $(f^{\mathrm{circ}}, f, \widetilde{f})$-CRO assumption (Figure fig. 3). Namely,

  - $\mathsf{hct} = \overline{\mathbf{B}}\mathbf{R} + \mathsf{bits}(\mathbf{r})^{\intercal} \otimes \mathbf{G}$, $\mathsf{hct}_0 = \overline{\mathbf{B}}\mathbf{R}_0$, where $\overline{\mathbf{B}} = \begin{pmatrix} \mathbf{B} \\ \mathbf{r}^{\intercal}\mathbf{B} + \mathbf{e}^{\intercal} \end{pmatrix}$.
  - $\mathbf{A}, \mathbf{D} \leftarrow \$$, $\mathsf{ct}_1 \leftarrow \mathbf{U}^{\intercal}\mathbf{A} + \mathbf{I}_{\ell} \otimes \mathbf{G}_n^{\intercal}\mathbf{r}$, $\mathsf{ct}_2 \leftarrow \mathbf{r}^{\intercal}\mathbf{D} + f^{\mathrm{circ}}(\mathbf{U}, \mathsf{hct}_0)$.

  We can expand the opening sampling equation to

  $$\mathbf{R}^* \leftarrow \mathsf{SampPre}(\mathbf{B}_0, \mathbf{T}, \overline{\mathbf{B}}(\mathbf{R}_f + \mathbf{R}_0) \boxplus (f(\mathbf{r}, \mathbf{A}, \mathbf{D}) - \widetilde{f}(\mathsf{enc})), \sigma_0).$$

- $\mathcal{H}_4$: This hybrid samples the opening by

  $$\mathbf{R}^* \leftarrow \mathsf{SampPre}(\mathbf{B}_0, \mathbf{T}, \overline{\mathbf{B}}\mathbf{R}_0, \sigma_0),$$

  where $\overline{\mathbf{B}} = \begin{pmatrix} \mathbf{B} \\ \mathbf{r}^{\intercal}\mathbf{B} + \mathbf{e}^{\intercal} \end{pmatrix}$, $\mathbf{R}_0 \leftarrow \mathcal{D}_{\sigma_0}^{m\times Lm}$.

- $\mathcal{H}_5$: This hybrid switches $\overline{\mathbf{B}}$ to random and samples the opening by

$$\mathbf{R}^* \leftarrow \mathsf{SampPre}(\mathbf{B}_0, \mathbf{T}, \overline{\mathbf{B}}\mathbf{R}_0, \sigma_0),$$

  where $\overline{\mathbf{B}} \leftarrow \mathbb{Z}_q^{(n+1)\times m}$, $\mathbf{R}_0 \leftarrow \mathcal{D}_{\sigma_0}^{m\times Lm'}$.

Observe that

- $\mathcal{H}_0$ is 2^{-n}-close to $\mathcal{H}_1$ by the statistical randomness property of lattice trapdoors (lemma 5).
- $\mathcal{H}_1$ is 2^{-n}-close to $\mathcal{H}_2$ by the preimage sampling property of lattice trapdoors (lemma 5).
- $\mathcal{H}_2$ and $\mathcal{H}_3$ are (subexponentially) indistinguishable assuming the (subexponential) f^{circ}-circular security assumption.
- $\mathcal{H}_3$ is (subexponentially)-close to $\mathcal{H}_4$. This follows identical arguments from the proof of Theorem 1.
- $\mathcal{H}_4$ and $\mathcal{H}_5$ are (subexponentially) indistinguishable assuming (subexponential) LWE, which is implied by (subexponential) f^{circ}-circular security assumption.

Finally, by the leftover hash lemma (lemma 1), the distribution of $\overline{\mathbf{B}}\mathbf{R}_0$ in $\mathcal{H}_5$ is 2^{-n}-close to uniform. Furthermore, combining the leftover hash lemma and the preimage sampling property of lattice trapdoor, a preimage sampled from the SampPre algorithm with respect to a uniformly random target is 2^{-n}-close to the discrete Gaussian distribution. Therefore the output distribution of $\mathcal{H}_5$ is $2^{-\Omega n}$-close to the discrete Gaussian distribution $\mathcal{D}_{\sigma_0}^{m\times M}$.

With these, we conclude that the marginal distribution of $\mathbf{R}^*$ in $\mathcal{D}_1$ is (subexponentially) indistinguishable to the discrete Gaussian distribution. □

3 Preliminaries for Construction

3.1 Indistinguishability Obfuscation (iO)

Definition 8 (Indistinguishability Obfuscation (iO) [BGI+01, GGH+13b]). *An indistinguishability obfuscation scheme for a circuit class $\{C_\lambda\}_{\lambda\in\mathbb{N}}$ consists of the following two efficient algorithms:*

- $\mathsf{Obf}(1^\lambda, \Pi)$: *On input a security parameter λ and a circuit $\Pi \in C_\lambda$, outputs an obfuscated circuit $\widetilde{\Pi}$.*
- $\mathsf{Eval}(1^\lambda, \widetilde{\Pi}, x)$: *On input a security parameter λ, an obfuscated circuit $\widetilde{\Pi}$, and an input x, outputs an evaluation result y.*

We require an iO scheme to satisfy the following properties:

Correctness: *For all $\lambda, n \in \mathbb{N}$, every circuit $\Pi \in C_\lambda$, with input length n, and every $x \in \{0,1\}^n$,*

$$\Pr[\mathsf{Eval}(1^\lambda, \mathsf{Obf}(1^\lambda, \Pi_\lambda), x) = \Pi_\lambda(x)] = 1.$$

$\epsilon(\lambda)$-IND-security *For every pair of circuit sequences $\{\Pi_\lambda^0\}, \{\Pi_\lambda^1\} \in \{C_\lambda\}$ such that Π_λ^0 and Π_λ^1 are functionally equivalent, the following ensembles are $\epsilon(\lambda)$-indistinguishable to all polynomial-sized adversaries::*

$$\{\mathsf{Obf}(1^\lambda, \Pi_\lambda^0)\}_\lambda \approx_c^\epsilon \{\mathsf{Obf}(1^\lambda, \Pi_\lambda^1)\}_\lambda.$$

3.2 Exponentially-Efficient iO (xiO)

Definition 9 (xiO with preprocessing [LPST16])**.** *An exponentially-efficient indistinguishability obfuscation scheme with preprocessing for a circuit class $\{C_\lambda\}_{\lambda\in\mathbb{N}}$ consists of the following three efficient algorithms:*

- $\mathsf{Gen}(1^\lambda, 1^N, 1^s)$*: On input a security parameter λ, the circuit input space size $N = 2^n$, and an upper bound s on the circuit size, outputs an common reference string* crs*. We assume that* crs *implicitly includes the parameters $(1^\lambda, 1^N, 1^s)$.*
- $\mathsf{Obf}(\mathsf{crs}, \Pi)$*: On input the common reference string* crs *and a circuit $\Pi \in C_\lambda$ with size at most s and taking $n = \log N$ bit input, outputs an obfuscated circuit $\widetilde{\Pi}$.*
- $\mathsf{Eval}(\mathsf{crs}, \widetilde{\Pi}, x)$*: On input the common reference string* crs*, an obfuscated circuit $\widetilde{\Pi}$, and an input x, outputs an evaluation result y.*

We require an xiO *scheme to satisfy the following properties:*

Correctness: *For all $\lambda, N, s \in \mathbb{N}$, every circuits $\Pi \in C_\lambda$ with size at most s and taking $n = \log N$ bit input, and every input $x \in \{0,1\}^n$,*

$$\Pr\left[\mathsf{Eval}(\mathsf{crs}, \widetilde{\Pi}, x) = \Pi(x) \;\middle|\; \mathsf{crs} \leftarrow \mathsf{Gen}(1^\lambda, 1^N, 1^s),\ \widetilde{\Pi} \leftarrow \mathsf{Obf}(\mathsf{crs}, \Pi)\right] = 1$$

Succinctness *There exist a constant $\epsilon \in (0,1)$ and some polynomial* poly *such that for all $\lambda, N, s \in \mathbb{N}$, all circuits $\Pi \in C_\lambda$ with $n = \log N$ bit input and size at most s, all* crs*, and all obfuscated circuit $\widetilde{\Pi}$ in the support of* $\mathsf{Obf}(1^\lambda, \mathsf{crs}, \Pi)$*, the size of the obfuscated circuit is bounded by*

$$|\widetilde{\Pi}| \leq N^{1-\epsilon} \cdot \mathsf{poly}(\lambda, s).$$

$\epsilon(\lambda)$**-IND-security** *For all polynomial $N(\cdot), s(\cdot)$ and every pair of circuit sequences $\{\Pi^0_\lambda\}, \{\Pi^1_\lambda\} \in \{C_\lambda\}$ consisting of circuits with size at most $s(\lambda)$ and taking $n(\lambda) = \log N(\lambda)$ bit input, if Π^0_λ and Π^1_λ are functionally equivalent for all $\lambda \in \mathbb{N}$, the following ensembles are $\epsilon(\lambda)$-indistinguishable to all polynomial-sized adversaries:*

$$\left\{(\mathsf{crs}, \widetilde{\Pi}) \;\middle|\; \begin{array}{ll} \mathsf{crs} & \leftarrow \mathsf{Gen}(1^\lambda, 1^{N(\lambda)}, 1^{s(\lambda)}) \\ \widetilde{\Pi} & \leftarrow \mathsf{Obf}(\mathsf{crs}, \Pi^0_\lambda) \end{array}\right\}_\lambda \approx^\epsilon_c \left\{(\mathsf{crs}, \widetilde{\Pi}) \;\middle|\; \begin{array}{ll} \mathsf{crs} & \leftarrow \mathsf{Gen}(1^\lambda, 1^{N(\lambda)}, 1^{s(\lambda)}) \\ \widetilde{\Pi} & \leftarrow \mathsf{Obf}(\mathsf{crs}, \Pi^1_\lambda) \end{array}\right\}_\lambda$$

Theorem 3 ([LPST16, BDGM22])**.** *Assuming the sub-exponential hardness of the learning with errors (LWE) problem, and a sub-exponentially secure* xiO *scheme with preprocessing for polynomial-sized circuits,* iO *exists for all polynomial-sized circuits.*

3.3 Functional Encodings

In this section, we recall a slight variant of the functional encoding scheme introduced in [WW21], which is a useful primitive implying iO.

Definition 10 (Functional Encoding)**.** *A functional encoding scheme in the CRS model for a circuit class $\{C_\lambda\}_{\lambda\in\mathbb{N}}$ consists of the following efficient algorithms:*

- $\mathsf{Gen}(1^\lambda, 1^Q, 1^K, 1^L, 1^d)$ *takes as input a security parameter* λ*, a bound* Q *on the number of openings, the circuit input/output length* $1^K, 1^L$*, and a bound* 1^d *on circuit depth. The algorithm outputs a common reference string* crs*. We assume that* crs *implicitly encodes the parameters* $(1^\lambda, 1^Q, 1^K, 1^L, 1^d)$*.*

- $\mathsf{Enc}(\mathsf{crs}, \mathbf{x}; R)$ *takes as input the common reference string* crs *and an input* $\mathbf{x} \in \{0,1\}^K$*. With randomness* R*, the algorithm outputs an encoding* ct*.*

- $\mathsf{Open}(\mathsf{crs}, g, i, \mathbf{x}, R)$ *takes as input the common reference string* crs*, a depth* d *circuit* $g \colon \{0,1\}^K \to \{0,1\}^L$ *in* C_λ*, an index* $i \in [Q]$*, an input* $\mathbf{x} \in \{0,1\}^K$*, and a encoding randomness* R*. The algorithm* deterministically *outputs* ρ_i *as the opening corresponding to the* i*-th function* f*.*

- $\mathsf{Dec}(\mathsf{crs}, g, i, \mathsf{ct}, \rho)$ *takes as input the common reference string* crs*, a depth* d *circuit* $g \colon \{0,1\}^K \to \{0,1\}^L$ *in* C_λ*, an index* $i \in [Q]$*, an encoding* ct*, and an opening* ρ*. The algorithm* deterministically *outputs the decoding result* $\mathbf{y} \in \{0,1\}^L$*.*

The algorithms are required to satisfy the following properties:

Correctness *For all* $\lambda, Q, K, L, d \in \mathbb{N}$*, all depth* d *circuit* $g \colon \{0,1\}^K \to \{0,1\}^L$ *in* C_λ*, all input* $\mathbf{x} \in \{0,1\}^K$*, and all index* $i \in [Q]$*, it holds that*

$$\Pr\left[\mathsf{Dec}(\mathsf{crs}, g, i, \mathsf{ct}, \rho) = g(\mathbf{x}) \;\middle|\; \begin{array}{c} \mathsf{crs} \leftarrow \mathsf{Gen}(1^\lambda, 1^Q, 1^K, 1^L, 1^d) \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{crs}, \mathbf{x}; R) \\ \rho \leftarrow \mathsf{Open}(\mathsf{crs}, g, i, \mathbf{x}, R) \end{array} \right] = 1$$

Succinctness *There exist a constant* $\varepsilon \in (0,1)$ *such that for all* $\lambda, Q, K, L, d \in \mathbb{N}$*, all* crs*, all depth* d *circuit* $g \colon \{0,1\}^K \to \{0,1\}^L$ *in* C_λ*, all* $i \in [Q]$*, all* $\mathbf{x} \in \{0,1\}^K$*, and all* R*, the size of the encoding* $\mathsf{ct} = \mathsf{Enc}(\mathsf{crs}, \mathbf{x}; R)$ *and the opening* $\rho = \mathsf{Open}(\mathsf{crs}, g, i, \mathbf{x}, R)$ *are bounded by:*

$$|\mathsf{ct}| \le (L^{1-\epsilon} Q + \mathsf{poly}(L)) \cdot \mathsf{poly}(\lambda, K, d), \quad |\rho| \le L^{1-\varepsilon} \cdot \mathsf{poly}(\lambda, K, d).$$

$\epsilon(\lambda)$ **SIM-security** *There exists an efficient simulator* Sim *such that for all polynomials* $Q = Q(\lambda), k = k(\lambda), L = L(\lambda), d = d(\lambda)$*, every ensemble of messages-function tuple* $\{\mathbf{x}, g_1, \ldots, g_Q\}_\lambda$ *where input* $x \in \{0,1\}^K$ *and each function* $g_i : \{0,1\}^K \to \{0,1\}^L$ *are depth* d *circuits in* C_λ*, the following ensembles are* $\epsilon(n)$*-indistinguishable to all polynomial-sized adversaries:*

$$\left\{ (\mathsf{crs}, \mathsf{ct}, \{\rho_i\}_{i \in [Q]}) \;\middle|\; \begin{array}{cl} \mathsf{crs} & \leftarrow \mathsf{Gen}(1^\lambda, 1^Q, 1^K, 1^L, 1^d) \\ \mathsf{ct} & \leftarrow \mathsf{Enc}(\mathsf{crs}, \mathbf{x}; R) \\ \rho_i & \leftarrow \mathsf{Open}(\mathsf{crs}, g_i, i, \mathbf{x}, R) \end{array} \right\}_\lambda \approx_c^\epsilon \left\{ \mathsf{Sim}(1^\lambda, \{g_i, g_i(\mathbf{x})\}_{i \in [Q]}) \right\}_\lambda$$

Theorem 4 ([WW21])**.** *Assuming a (sub-exponential) secure functional encoding scheme for polynomial-sized circuits, there exists a (sub-exponential) secure* xiO *scheme with preprocessing for polynomial-sized circuits.*

Remark 2. *In [WW21], functional encodings were defined with a stronger succinctness requirement, where the encoding size is bounded by* $\mathsf{poly}(\lambda, K, L, d)$*, independent of the number of openings. Nevertheless, the FE to* xiO *transformation given in [WW21] remains valid as long as the parameters* Q *and* L *are chosen such that* $N = QL$ *and the total size* $(|\mathsf{ct}| + Q|\rho|)$ *is sublinear in* N*, which is also achievable under our definition.*

Remark 3. *One might observe that functional encoding schemes inherently provide a simulation-secure* xiO *scheme with preprocessing, leading to an exponentially efficient pseudorandom obfuscation (*xPrO*) with preprocessing. However, the known transformation from* xiO *with preprocessing to* iO *does not directly extend to transforming* xPrO *with preprocessing into pseudorandom obfuscation (*PrO*). Consequently, the existence of functional encoding schemes does not contradict the impossibility result for* PrO *established in [BDJ+24]. This leaves open the plausibility of constructing secure functional encoding schemes.*

3.4 Dual-GSW Encryption

In this section, we recall the secret key dual-GSW encryption scheme (also known as the Packed dual-Regev encryption scheme), formulated according to the syntax of a functional encoding scheme. It satisfies correctness and succinctness, but not simulation security. The algebraic structure of the dual-GSW encryption scheme closely resembles that of the GSW encryption scheme [GSW13], as recalled in section 2.1.

The dual-GSW encryption scheme consists of the following algorithms.

- $\mathsf{Gen}(1^\lambda, 1^Q, 1^K, 1^L, 1^d)$: The algorithm picks an encoding size κ, modulus p, q such that $p|q$, $q/p > 2^\kappa$, and $p \gg \ell^{\Theta(d)}\mathsf{poly}(\sigma)$, and dimension $\ell \geq \max(L/\kappa, \log q)$. It outputs a random matrix $\mathbf{A} \leftarrow \mathbb{Z}_q^{n\times\ell}$.

- $\mathsf{Enc}(\mathsf{crs}, \mathbf{x}; R = (\mathbf{U}, \mathbf{E}))$: Vector $\mathbf{x} \in \{0,1\}^K$ of bits is encrypted by

$$\mathsf{dct} = \mathbf{U}^\intercal\mathbf{A} + \mathbf{E} + \mathbf{x} \otimes \mathbf{G}_\ell^\intercal, \text{where } \mathbf{U} \leftarrow \mathbb{Z}_q^{n\times K\ell\lceil\log q\rceil}, \mathbf{E} \leftarrow \mathcal{D}_\sigma^{K\ell\lceil\log q\rceil\times\ell}.$$

  Note that $\mathbf{G}_\ell$ has dimension $\ell \times \ell\lceil\log q\rceil$, and so are the $\mathbf{G}$ in the rest of the description.

- $\mathsf{Open}(\mathsf{crs}, g, i, \mathbf{x}, R)$: We first start with describing the homomorphic evaluation operations supported by the dual GSW scheme. Then use them to describe the opening algorithm.

  - **HAdd:** For bits $b_1, b_2 \in \{0,1\}$,

$$\mathsf{dct}(b_1) \boxplus \mathsf{dct}(b_2) = (\mathbf{U}_1^\intercal\mathbf{A} + \mathbf{E}_1 + b_1 \otimes \mathbf{G}^\intercal) + (\mathbf{U}_2^\intercal\mathbf{A} + \mathbf{E}_2 + b_2 \otimes \mathbf{G}^\intercal) = \mathbf{U}_+^\intercal\mathbf{A} + \mathbf{E}_+ + (b_1 + b_2)\mathbf{G}^\intercal,$$

    where $\mathbf{U}_+ = (\mathbf{U}_1 + \mathbf{U}_2)$, $\|\mathbf{E}_+\| \leq 2\|\mathbf{E}_1, \mathbf{E}_2\|$.

  - **HMult:** Ciphertext for bits are multiplicative homomorphic. For bits $b_1, b_2 \in \{0,1\}$,

$$\mathsf{dct}(b_1) \boxtimes \mathsf{dct}(b_2) = (\mathbf{G}^{-1}(\mathsf{dct}_1^\intercal))^\intercal \cdot \mathsf{dct}_2 = \mathbf{U}_\times^\intercal\mathbf{A} + \mathbf{E}_\times + b_1 b_2\mathbf{G}^\intercal$$

    where $\mathbf{U}_\times^\intercal = \mathbf{G}^{-1}(\mathsf{ct}_1^\intercal)^\intercal\mathbf{U}_2^\intercal + b_2\mathbf{U}_1^\intercal$[7], and $\mathbf{E}_\times \leq 2\ell\|\mathbf{E}_1, \mathbf{E}_2\|$.

  - **Packing:** Ciphertext for bits can be packed to ciphertext for vectors. Given a vector $\mathbf{v} \in \mathbb{Z}_q^\ell$ with bitwise representation $\mathbf{v}^\intercal = \sum 2^t(v_{1,t}, \ldots, v_{m,t})$,

$$\mathsf{Pack}(\{\mathsf{dct}(v_{i,t})\}) = \sum \mathbf{G}^{-1}(2^t \cdot \mathbf{1}_i)^\intercal\mathsf{dct}(v_{i,t}) = \mathbf{u}_{\mathsf{Pack}}^\intercal\mathbf{A} + \mathbf{e}_{\mathsf{Pack}}^\intercal + \mathbf{v}^\intercal,$$

    where $\mathbf{1}_i$ is the unit vector which is one at the i-th coordinate and 0 elsewhere, $\mathbf{u}_{\mathsf{Pack}} = \sum \mathbf{U}\mathbf{G}^{-1}(2^t \cdot \mathbf{1}_i)$, and $\|\mathbf{e}_{\mathsf{Pack}}\| \leq \mathsf{poly}(\ell, \log q)\|\mathbf{E}\|$.

  Combining these operations, any polynomial-sized depth d circuit $C\colon \{0,1\}^K \to \mathbb{Z}_q^\ell$ can be homomorphically evaluated as

$$\mathsf{Eval}(\mathsf{dct}(\mathbf{x}), C) = \mathbf{u}_C^\intercal\mathbf{A} + \mathbf{e}_C^\intercal + C(\mathbf{x}), \quad \mathsf{EvalU}(\mathsf{dct}(\mathbf{x}), C, \mathbf{x}, \mathbf{U}) = \mathbf{u}_C,$$

  where $\|\mathbf{e}_C\| \leq \ell^{O(d)}\|\mathbf{E}\|$, and the randomness evaluating algorithm EvalU is efficiently computable.

  Equipped with the above homomorphic evaluation operation, the opening algorithm proceeds as follows: It parses the encryption randomness as $R = (\mathbf{U}, \mathbf{E})$. For an input function

[7] Notice that $(\mathbf{G}^{-1}(\mathbf{P}^\intercal))^\intercal \cdot \mathbf{G}^\intercal = (\mathbf{G} \cdot \mathbf{G}^{-1}(\mathbf{P}^\intercal))^\intercal = \mathbf{P}$.

$g\colon \{0,1\}^K \to \{0,1\}^L$ represented as a circuit C, the algorithm prepares another circuit C_g $C_g\colon \{0,1\}^K \to \mathbb{Z}_q^\ell$ with $\mathbb{Z}_q$ outputs. For the parameters p and κ chosen at setup, an output element y_i of C_g satisfies that $y_i \bmod p = 0$ and y/p is the integer representation of the i'th κ-bit chunk of the output of g. Note that if g has depth d_g, C_g can be implemented in depth $d_g + O(\log \lambda)$.

It outputs the opening $\rho = \mathbf{u}_{C_g} = \mathsf{EvalU}(\mathsf{dct}(\mathbf{x}), C_g, \mathbf{x}, \mathbf{U})$.

- $\mathsf{Dec}(\mathsf{crs}, g, i, \mathsf{dct}, \rho = \mathbf{u})$: Given g represented as a circuit C, prepare the same circuit C_g described above. Homomorphically evaluate C_g to obtain output ciphertext $\mathsf{dct}_g = \mathsf{Eval}(\mathsf{dct}, \widetilde{g})$, and recover the output $g(\mathbf{x})$ by computing
$$\left\lfloor \frac{\mathsf{dct}_g - \mathbf{u}^\intercal \mathbf{A}}{p} \right\rceil .$$

The dual GSW scheme is correct since $pp \gg \ell^{\Theta(d)} \|\mathbf{E}\|$, indicating that all errors introduced in the homomorphic evaluation procedure do not exceed the rounding boundary.

With appropriate parameters, it also satisfies succinctness since $|\mathsf{dct}(\mathbf{x})|$ is independent of Q, and the length of each opening that $|\rho| = n \log q$ is sublinear in L. Note first that for LWE to hold, we need $2^{n^{1-\epsilon}} > (q/\sigma)$ for some $\epsilon > 0$, where $\sigma = \mathsf{poly}(\lambda)$ is the width of the noises in LWE samples, which holds if setting $n = (\log q)^{1+\epsilon'}$ for some $\epsilon' > 0$ (dependent on ϵ). We also have the constraint that $q > 2^\kappa p \gg 2^\kappa \ell^{\Theta(d)} \sigma$. Therefore, the succinctness of opening $|\rho| = n \log q = (\log q)^{2+\epsilon'} = L^{1-\Omega(1)} \mathsf{poly}(\lambda, K, d)$ follows if $(\kappa + \Theta(d) \log \ell + \lambda)^{2+\epsilon'} = L^{1-\Omega(1)} \mathsf{poly}(\lambda, K, d)$. This holds when the parameter κ is smaller than $L^{(1-\epsilon'')/(2+\epsilon')}$ for ϵ' determined by LWE security and ϵ'' an arbitrarily small positive constant.

Unfortunately, dual GSW does not satisfy the simulation security required by definition 10. Indeed, the opening $\mathbf{u}_f$ can leak information of the encryption randomness $\mathbf{U}$.

We also note that there is an alternative dual GSW encoding that supports linear homomorphism. Namely, we can encode a vector $\mathbf{v} \in \mathbb{Z}_q^n$ by
$$\mathsf{dct}(\mathbf{v}) = \mathbf{U}^\intercal \mathbf{A} + \mathbf{E} + \mathbf{I}_\ell \otimes \mathbf{G}_n^\intercal \mathbf{v}.$$

For matrices $M \in \mathbb{Z}_q^{n \times \ell}$, the encoding satisfies homomorphic relation
$$\mathsf{vec}(\mathbf{G}^{-1}(\mathbf{M}))^\intercal \mathsf{dct}(\mathbf{v}) = \mathbf{u}_\mathbf{M}^\intercal \mathbf{A} + \mathbf{E}' + \mathbf{v}^\intercal \mathbf{M}, \quad \text{where } \mathbf{u}_\mathbf{M} = \mathbf{U} \cdot \mathsf{vec}(\mathbf{G}^{-1}(\mathbf{M})),\ \|\mathbf{E}'\| \leq \ell n \|\mathbf{E}\|, \tag{3}$$
where the equation follows by the fact $\mathsf{vec}(\mathbf{AB}) = (\mathbf{I} \otimes \mathbf{A})\mathsf{vec}(\mathbf{B})$.

4 Overview for Functional Encoding Construction

GSW and dual-GSW, through the Lens of Functional Encoding Both the GSW and dual GSW homomorphic encryption scheme can be converted into a functional encryption scheme: To encode an input $\mathbf{x}$, simply encrypt it $\mathsf{hct} = \mathsf{Enc}(\mathbf{x}; \rho)$. To open the output of a function g, first perform homomorphic evaluation to obtain $\mathsf{hct}_g = \mathsf{Enc}(\mathbf{x}; \rho_g)$ and use the randomness ρ_g underlying the output ciphertext as the opening. More specifically,
$$\overline{\mathbf{B}} = \begin{pmatrix} \mathbf{B} \\ \mathbf{r}^\intercal \mathbf{B} + \mathbf{e}_\mathbf{B}^\intercal \end{pmatrix},\ \mathsf{GSW.hct}(\mathbf{x}) = \overline{\mathbf{B}}\mathbf{R} + \mathbf{x}^\intercal \otimes \mathbf{G} \implies \mathsf{GSW.hct}_g = \overline{\mathbf{B}}\mathbf{R}_g + \begin{pmatrix} \mathbf{0} \\ g(\mathbf{x}) \end{pmatrix}$$
$$\mathsf{dGSW.hct}(\mathbf{x}) = \mathbf{U}^\intercal \mathbf{A} + \mathbf{x} \otimes \mathbf{G}^\intercal + \mathbf{E} \implies \mathsf{dGSW.hct}_g = \mathbf{u}_g^\intercal \mathbf{A} + g(\mathbf{x}) + \mathbf{e}_g^\intercal$$

Both GSW and dual-GSW allow packing and $g(\mathbf{x})$ can be a $\mathbb{Z}_q$ vector of some dimension ℓ, and bit length close to $L = \ell \log q$ (modulo low order bits). The GSW opening is not succinct, as $\rho_g = \mathbf{R}_g$ has size $n \log q \cdot \ell \cdot \log q$. But dual-GSW does have succinct openings, with $\rho_g = \mathbf{u}_g$ of size $n \cdot \log q$, which is sublinear in L. Note that GSW does not have noise leakage, whereas dual-GSW leaks $\mathbf{e}_g$.

The main issue is that their openings reveal more information than $g(\mathbf{x})$. In both cases, $\mathbf{R}_g, \mathbf{u}_g$ is a linear function (dependent on $\mathbf{x}, g, \mathsf{hct}$) of the original randomness $\mathbf{R}, \mathbf{U}$. The revelation of them could completely compromise security.

One way to create a safe opening is through re-randomization: If there is additionally a fresh ciphertext of zero hct_0 generated using randomness from an appropriate distribution, we can instead open $\mathsf{hct}'_g = \mathsf{hct}_g + \mathsf{hct}_0$. The randomness of hct_0 can ensure that the re-randomized opening and noise leakage reveals only $g(\mathbf{x})$. More specifically, in GSW, the randomness in hct_0 is $\widetilde{\mathbf{R}}$, consisting of i.i.d. sufficiently wide discrete Gaussian samples, and the opening becomes $\mathbf{R}^* = \mathbf{R}_g + \widetilde{\mathbf{R}}$, while in dual-GSW, the randomness in hct_0 consists of random $\mathbf{s}$ and smudging noise $\mathbf{e}$, and the opening becomes $\tilde{\mathbf{s}} = \mathbf{u}_g + \mathbf{s}$, leaking noise $\tilde{\mathbf{e}} = \mathbf{e}_g + \mathbf{e}$. The fact that they reveal only $g(\mathbf{x})$ can be proven using the standard simulation technique that "programs" the output $g(\mathbf{x})$ into hct_0 (e.g., see [WW21] for such a proof). Interestingly, GSW admits an alternative simulation strategy that "programs" $g(\mathbf{x})$ into the opening $\mathbf{R}^*$ (e.g., see [GP21] for such a proof).

The problem is we need fresh and independent zero-ciphertexts $\{\mathsf{hct}_{0,i}\}_{i \in [Q]}$ for each functional opening for g_i. There is no place for these zero-ciphertexts: They are too large, larger than $Q \cdot L$ bits, to be put in the succinct functional encoding. On the other hand, leaving them in the CRS renders them useless, as giving re-randomized openings such as $\widetilde{\mathbf{R}}, \tilde{\mathbf{s}}$ requires knowing the secrets related to the CRS.

Version 0: Combining GSW with dual-GSW In this work, we will leverage both the succinct opening of dual-GSW and the GSW simulation strategy of programming into randomness $\mathbf{R}^*$. A technique of combining them, introduced by [BDGM20, BDGM22, GP21], is to perform homomorphic evaluation using GSW, followed by homomorphic decryption using dual-GSW, as shown in Figure 5.

An encoding of $\mathbf{x}$ contains a GSW public key $\mathsf{hpk} = \overline{\mathbf{B}}$ with secret $\mathbf{r} \in \mathbb{Z}_q^n$, and a ciphertext $\mathsf{hct}(\mathbf{x})$. It also contains a dual-GSW ciphertext dct using public matrix $\mathbf{A}$ and encrypting the GSW secret $\mathbf{r}$ in a special form $\mathbf{I}_\ell \otimes \mathbf{G}_n^\top \mathbf{r} \in \mathbb{Z}_q^{\ell \cdot n \cdot \log q \times \ell}$.

Opening the output of a function g_i proceeds as described in bottom part of Figure 5: Step 1) computes a GSW-ciphertext hct_{g_i} of $g_i(\mathbf{x})$, followed by Step 2) that homomorphically decrypts hct_{g_i} under dual-GSW by computing dct_{g_i}.

$$\mathsf{dct}_{g_i} = \underbrace{\mathsf{vec}\big(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{g_i})\big)^\top}_{\mathbf{v}_{g_i} = \mathsf{vec}(\mathbf{G}^{-1}(-\mathbf{B}\mathbf{R}_{g_i})} \cdot \mathsf{dct} = \underbrace{\mathbf{v}_{g_i}^\top \cdot \mathbf{U}^\top}_{\mathbf{u}_{g_i}^\top} \cdot \mathbf{A} + \underbrace{\mathbf{v}_{g_i}^\top \cdot \mathbf{E}_{\mathbf{A}}}_{\mathbf{e}_{g_i}^\top} + \underbrace{\mathbf{v}_{g_i}^\top \cdot (\mathbf{I}_\ell \otimes \mathbf{G}^\top \mathbf{r})}_{-\mathbf{r}^\top \cdot \mathbf{B}\mathbf{R}_{g_i}} \tag{5}$$

Adding hct_{g_i} and dct_{g_i} gives a dual-GSW ciphertext of $g_i(\mathbf{x})$ as shown in Equation (4), which can be succinctly opened by revealing $\mathbf{u}_{g_i}$. However, just as dual-GSW, revealing $\mathbf{u}_{g_i}$ and leaking $\mathbf{e}_{g_i}$ may completely compromise security.

Version I: Special Encoding of Secrets $(\mathbf{s}, \mathbf{e})$ of Zero-Ciphertexts In order to hide $\mathbf{u}_{g_i}, \mathbf{e}_{g_i}$, we attempt to re-randomize the final dual-GSW ciphertext. Since, as discussed above, there is no suitable place for storing zero-ciphertexts $\{\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top\}_{i \in [Q]}$, we instead encrypt their secrets as described in Figure 6. Version I encrypts all $\mathbf{s}_i$'s using GSW, and hides all $\mathbf{e}_i$'s in LWE samples $\mathbf{c}_i^\top = (\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top \mod \Delta)$ modulo *a much smaller modulus* $\Delta \ll q$. (The dual-GSW components stay the same.) Note that the LWE samples $\{\mathbf{c}_i\}_{i \in [Q]}$ are succinct, of size $Q \cdot \ell \cdot \log \Delta \ll Q \cdot \ell \cdot \log q \approx Q \cdot L$. This also shows we

Encoding of $\mathbf{x}$:

GSW components

$$\mathsf{hpk} = \overline{\mathbf{B}} = \begin{pmatrix} \mathbf{B} \\ \mathbf{r}^\top \mathbf{B} + \mathbf{e}_\mathbf{B}^\top \end{pmatrix}.$$

$$\mathsf{hct}(\mathbf{x}) = \overline{\mathbf{B}}\mathbf{R} + \mathbf{x}^\top \otimes \mathbf{G}_{n+1} \ .$$

dGSW components

$$\mathbf{A}$$

$$\mathsf{dct} = \mathbf{U}^\top \mathbf{A} + \mathbf{E}_\mathbf{A} + \mathbf{I}_\ell \otimes \mathbf{G}_n^\top \mathbf{r}.$$

Opening $\mathbf{u}_{g_i}$ for g_i:

(1) Evaluate g_i.

$$\mathsf{hct}_{g_i} = \mathsf{Eval}(\mathsf{hct}(\mathbf{x}), g_i),$$

$$\begin{pmatrix} \overline{\mathsf{hct}}_{g_i} \\ \underline{\mathsf{hct}}_{g_i} \end{pmatrix} = \begin{pmatrix} \mathbf{B}\mathbf{R}_{g_i} \\ (\mathbf{r}^\top \mathbf{B} + \mathbf{e}_\mathbf{B}^\top)\mathbf{R}_{g_i} + g_i(\mathbf{x}) \end{pmatrix}.$$

(2) Linear decryption of hct_{g_i}

$$\begin{aligned} \mathbf{v}_{g_i} &= \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{g_i})) \\ &= \mathsf{vec}(\mathbf{G}^{-1}(-\mathbf{B}\mathbf{R}_{g_i})) \\ \mathsf{dct}_{g_i} &= \mathbf{v}_{g_i}^\top \cdot \mathsf{dct} = \mathbf{u}_{g_i}^\top \mathbf{A} + \mathbf{e}_{g_i}^\top - \mathbf{r}^\top(\mathbf{B}\mathbf{R}_{g_i}). \end{aligned}$$

$$\textbf{Correctness:} \quad \underline{\mathsf{hct}}_{g_i} + \mathsf{dct}_{g_i} = \mathbf{u}_{g_i}^\top \mathbf{A} + g_i(\mathbf{x}) + (\mathbf{e}_\mathbf{B}^\top \mathbf{R}_{g_i} + \mathbf{e}_{g_i}^\top) \tag{4}$$

Figure 5: Combining GSW and dual-GSW. The matrix/vectors are sampled as $\mathbf{r} \leftarrow \mathbb{Z}_q^n$, $\mathbf{B} \leftarrow \mathbb{Z}_q^{n\times m}$, $\mathbf{e_B} \leftarrow \mathcal{D}_\sigma^m$, $\mathbf{R} \leftarrow \{0,1\}^{m\times(n+1)\lceil \log q\rceil \cdot |\mathbf{x}|}$, $\mathbf{U} \leftarrow \mathbb{Z}_q^{n\times \ell n\lceil \log q\rceil}$, $\mathbf{A} \leftarrow \mathbb{Z}_q^{n\times\ell}$, $\mathbf{E_A} \leftarrow \mathcal{D}_\sigma^{\ell n\lceil \log q\rceil\times\ell}$. The function g_i has outputs in $\mathbb{Z}_q^\ell$. Hence variables derived from the homomorphic evaluation have dimensions: $\mathbf{R}_{g_i} \in \mathbb{Z}_q^{m\times\ell}$, $\mathbf{u}_{g_i} \in \mathbb{Z}_q^n$, $\mathbf{e}_{g_i} \in \mathbb{Z}_q^\ell$. The opening $\mathbf{u}_{g_i}$ is succinct: $|\mathbf{u}_{g_i}| = n \log q \ll \ell \log q = |g_i(\mathbf{x})|$.

cannot afford, for succinctness, to encrypt all the smudging noises $\mathbf{e}_i$ in any regular ciphertexts mod q.

Encoding:

GSW components

$$\mathsf{hpk} = \begin{pmatrix} \mathbf{B} \\ \mathbf{r}^\top \mathbf{B} + \mathbf{e}_\mathbf{B}^\top \end{pmatrix}.$$

$$\{\mathsf{hct}(\mathbf{s}_i) = \overline{\mathbf{B}}\mathbf{R} + \mathsf{bits}(\mathbf{s}_i)^\top \otimes \mathbf{G}_{n+1}\}_i.$$

Connecting components

$$\{\mathbf{c}_i^\top = (\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top \bmod \Delta)\}_i.$$

dGSW components

$$\mathbf{A}$$

$$\mathsf{dct} = \mathbf{U}^\top \mathbf{A} + \mathbf{E} + \mathbf{I}_\ell \otimes \mathbf{G}_n^\top \mathbf{r}.$$

Oblivious LWE Sampling:

(1) Evaluate $f_i(\mathbf{s}_i) = \Delta \left\lfloor \frac{\mathbf{s}_i^\top \mathbf{A}}{\Delta} \right\rceil$.

$$\mathsf{hct}_{f_i} = \mathsf{Eval}(\mathsf{hct}(\mathbf{s}_i), f_i),$$

$$\begin{pmatrix} \overline{\mathsf{hct}}_{f_i} \\ \underline{\mathsf{hct}}_{f_i} \end{pmatrix} = \begin{pmatrix} \mathbf{B}\mathbf{R}_{f_i} \\ (\mathbf{r}^\top \mathbf{B} + \mathbf{e}_\mathbf{B}^\top)\mathbf{R}_{f_i} + f_i(\mathbf{s}_i) \end{pmatrix}$$

(2) Add $\mathbf{c}_i$

$$\underline{\mathsf{hct}}_{f_i} + \mathbf{c}_i = (\mathbf{r}^\top \mathbf{B} + \mathbf{e}_\mathbf{B}^\top)\mathbf{R}_{f_i} + \mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top$$

(3) Linear decryption for hct_{f_i}

$$\begin{aligned} \mathbf{v}_{f_i} &= \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{f_i})) \\ \mathsf{dct}_{f_i} &= \mathbf{v}_{f_i}^\top \cdot \mathsf{dct} \\ &= \mathbf{u}_{f_i}^\top \mathbf{A} + \mathbf{e}_{f_i}^\top - \mathbf{r}^\top(\mathbf{B}\mathbf{R}_{f_i}) \end{aligned}$$

$$\begin{aligned} \textbf{Correctness :} \quad & \forall i \in [Q],\ \underline{\mathsf{hct}}_{f_i} + \mathbf{c}_i^\top + \mathsf{dct}_{f_i} = \tilde{\mathbf{s}}_i^\top \mathbf{A} + \tilde{\mathbf{e}}_i^\top \\ & \text{where } \tilde{\mathbf{s}}_i = \mathbf{u}_{f_i} + \mathbf{s}_i,\ \tilde{\mathbf{e}}_i^\top = \mathbf{e}_i^\top + \mathbf{e}_\mathbf{B}^\top \mathbf{R}_{f_i} + \mathbf{e}_{f_i}^\top \end{aligned} \tag{6}$$

Figure 6: Version I: Encrypt secrets $(\mathbf{s}_i, \mathbf{e}_i)$ of zero-ciphertexts. Note that $\mathbf{e}_i$'s are encoded in LWE samples with modulus Δ, where $\Delta \gg \|\mathbf{e}_i\|$. The output of f_i has roughly bit length $L = \ell \log q$. By setting $\log \Delta \ll \log q$ and $\log q$ to be sublinear in L, each $\mathbf{c}_i$ is succinct with length $L^{1-\epsilon}$. The marginal distribution of $\{\tilde{\mathbf{e}}_i\}$ is statistically close to iid Gaussian.

In the following, we will temporarily switch to the goal of oblivious LWE sampling, which captures the key ideas. Intuitively, we can think of computing the function f_i with output

$\mathbf{s}_i^\intercal \mathbf{A} + \mathbf{e}_i^\intercal \bmod q$, and the final dual-GSW ciphertext $\tilde{\mathbf{s}}_i^\intercal \mathbf{A} + \tilde{\mathbf{e}}_i^\intercal = (\mathbf{u}_{g_i} + \mathbf{s}_i)^\intercal \mathbf{A} + (\mathbf{e}_{g_i} + \mathbf{e}_i)^\intercal$ will be the generated LWE samples. (Note we use f_i to denote the functions related to oblivious LWE sampling, to not confuse with the functions g_i computed using functional encoding.) Our goal is to ensure that LWE secrets $(\tilde{\mathbf{s}}_i, \tilde{\mathbf{e}}_i)$ are pseudorandom, given the encoding and CRS (currently empty) from which they are generated. Eventually, this will be shown via simulation – the encodings and CRS can be simulated from $\tilde{\mathbf{s}}_i^\intercal \mathbf{A} + \tilde{\mathbf{e}}_i^\intercal$ with truly random $\tilde{\mathbf{s}}_i, \tilde{\mathbf{e}}_i$.

To this end, our first attempt at generating $\tilde{\mathbf{s}}^\intercal \mathbf{A} + \tilde{\mathbf{e}}^\intercal$ is described in the bottom part of Figure 6. Step 1) uses GSW to homomorphically evaluate the function $f_i(\mathbf{s}_i) = \Delta \lfloor \mathbf{s}_i^\intercal \mathbf{A} / \Delta \rceil$ to get hct_{f_i}; Step 2) adds $\mathbf{c}_i = (\mathbf{s}_i \mathbf{A} + \mathbf{e}_i \bmod \Delta)$ to the last row of hct_{f_i} to obtain a GSW ciphertext of $f_i(\mathbf{s}_i) + \mathbf{c}_i = (\mathbf{s}_i^\intercal \mathbf{A} + \mathbf{e}_i^\intercal \bmod q)$; Step 3) homomorphically decrypts hct_{f_i} under dual-GSW as done in Equation (5) to produce the final LWE samples $\tilde{\mathbf{s}}_i^\intercal \mathbf{A} + \tilde{\mathbf{e}}_i^\intercal$.

Advantage and Drawbacks The advantage of Version I is that the LWE noises $\{\tilde{\mathbf{e}}_i\}_i$ follow the distribution of iid Gaussian $\tilde{\mathbf{e}}_{i,j} \sim \mathcal{D}_{\sigma_0}$. This is because $\tilde{\mathbf{e}}_i = \mathbf{e}_i + \mathbf{e}_\mathbf{B} \mathbf{R}_{f_i} + \mathbf{e}_{f_i}$, note that both $\mathbf{e}_\mathbf{B} \mathbf{R}_{f_i}$ and $\mathbf{e}_{f_i}$ resulting from homomorphic evaluation in Step 1) and 3) have bounded norm and are independent of the smudging noise $\mathbf{e}_i$. By sampling $\mathbf{e}_i$ according to $\mathcal{D}_{\sigma_0}$ with sufficient width $\sigma_0 \gg ||\mathbf{e}_{f_i} + \mathbf{e}_\mathbf{B} \mathbf{R}_{f_i}||$, $\tilde{\mathbf{e}}_i$ distributes statistically closely to iid Gaussian $\mathcal{D}_{\sigma_0}$. This means the noises $\tilde{\mathbf{e}}_i$ alone are safe to reveal.

It may appear that the LWE secret $\tilde{\mathbf{s}}_i$ is random, because of the randomness in $\mathbf{s}_i$. This is false because $\mathbf{u}_{f_i}$ may be correlated with $\mathbf{s}_i$. Recall that $\mathbf{u}_{f_i}^\intercal = \mathbf{v}_{f_i}^\intercal \cdot \mathbf{U}^\intercal$ and $\mathbf{v}_{f_i} = \mathsf{vec}\big(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{f_i})\big)$. Note the top part $\overline{\mathsf{hct}}_{f_i}$ of the GSW output ciphertext hct_{f_i} is correlated with the encrypted secret $\mathsf{bits}(\mathbf{s}_i)$, and so is $\mathbf{u}_{f_i}$. Therefore, revealing $\tilde{\mathbf{s}}_i$ may leak information $\mathbf{u}_{f_i}$ which may compromise security.

Comparisons In table 2, we provide a comparison on the distribution of the leakage/opening components for existing lattice-based iO constructions.

In all prior constructions of xiO and pseudorandom random obfuscation [DQV+21, WW21, AKY24, BDJ+24], except for [GP21, BDGM22], the marginal distribution of noise leakage is far from random. The structure in the noise leakage was leveraged in showing counterexamples [HJL21] against certain instances of [WW21] and in the attack in section 7 against private-coin evasive LWE [AKY24, BDJ+24].

The distinction lies in how the smudging noise $\mathbf{e}_i$ is encoded or generated. In prior works, they are generated either through homomorphic decryption of the CRS, or homomorphic evaluation of a PRF, or expanded from a few samples $(\mathbf{s}^\intercal \mathbf{B} + \mathbf{e}^\intercal)$ using a trapdoor $\mathbf{K} = \mathbf{B}^{-1}(\mathbf{P})$. In these examples, the generated smudging noise $\mathbf{e}_i$ is not random. The key idea in Version I is that random $\mathbf{e}_i$ is directly encoded in LWE samples with small modulus, and added to $\mathbf{e}_{f_i}$.

Version II: Special Homomorphic Evaluation Procedure We now fix the drawback in Version I that $\tilde{\mathbf{s}}_i = \mathbf{u}_{f_i} + \mathbf{s}_i$ is not marginally random. To this end, we remove the correlation between $\mathbf{u}_{f_i}$ and $\mathbf{s}_i$, by carefully designing a special procedure for homomorphically evaluating $\mathbf{s}_i^\intercal \mathbf{A} + \mathbf{e}_i^\intercal$. Our key observation is as follows: Given LWE sample $\mathbf{c}_{\mathbf{s},i}^\intercal = \mathbf{r}^\intercal \mathbf{D}_i + \mathbf{s}_i \mathbf{G} + \mathbf{e}_{\mathbf{s},i}^\intercal$ together with $\mathbf{c}_i$ introduced

Construction	LWE secret	LWE noise	GSW randomness	Lattice trapdoor
[GP21]	\$	\$	Counterexample ([HJL21])	✗
[BDGM22]	\$	\$	Counterexample[1] ([HJL21])	✗
[WW21]	Non-random	Counterexample ([HJL21])	✗	✗
[DQV⁺21]	Attack ([JLLS23])	Non-random	✗	✗
[BDJ⁺24] [AKY24]	✗	Counterexample (Section 7)	✗	\$
Ours	\$[2]	\$[2]	\$[2]	✗

[1] While [HJL21] did not directly give a counterexample for [BDGM22], the construction shares similar weakness structure as in [GP21], and it is pointed out in [HJL21] that the counter example is likely to extend.
[2] In our construction, these three leakages are jointly random.

Table 2: Characterization for different information leakage beyond LWE samples for existing iO/PrO constructions. These leakages typically come from the "opening components" required to enable evaluation procedures. In the table, ✗ stands for no such leakage exist, \$ stands for the leakage is marginally random (from a well-defined distribution), *Attack* stands for that there exist adversary breaking the construction through the leakage, and *Counterexample* stands for that there exist specific implementation for the construction which can be broken through the leakage.

above, we can obtain $\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top$ hidden by a pad $\mathsf{PAD}_i(\mathbf{r})$ dependent only on $\mathbf{r}$.

$$\begin{aligned}
\Delta \cdot \left\lfloor \frac{\mathbf{c}_{\mathbf{s},i}^\top \mathbf{G}^{-1}(\mathbf{A}) - \mathbf{c}_i^\top}{\Delta} \right\rceil + \mathbf{c}_i^\top &= \Delta \cdot \left\lfloor \frac{\mathbf{r}^\top \mathbf{D}_i \mathbf{G}^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{s},i}^\top \mathbf{G}^{-1}(\mathbf{A}) - \mathbf{e}_i^\top}{\Delta} + \frac{\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top - (\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top \bmod \Delta)}{\Delta} \right\rceil + \mathbf{c}_i^\top \\
&= \Delta \cdot \left\lfloor \frac{\mathbf{r}^\top \mathbf{D}_i \mathbf{G}^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{s},i}^\top \mathbf{G}^{-1}(\mathbf{A}) - \mathbf{e}_i^\top}{\Delta} \right\rceil + \Delta \cdot \left\lfloor \frac{\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top}{\Delta} \right\rceil + (\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top \bmod \Delta) \\
&\overset{w.h.p}{=} \Delta \cdot \left\lfloor \frac{\mathbf{r}^\top \mathbf{D}_i \mathbf{G}^{-1}(\mathbf{A})}{\Delta} \right\rceil + \mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top \\
&= \mathsf{PAD}_i(\mathbf{r}) + \mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top \qquad (7)
\end{aligned}$$

where the second last equality holds with high probability when the noises $\mathbf{e}_{\mathbf{s},i}^\top \mathbf{G}^{-1}(\mathbf{A}) - \mathbf{e}_i$ are much smaller than Δ.

The Version II encoding is described in Figure 7. It includes a circular GSW ciphertext $\mathsf{hct}(\mathbf{r})$ and LWE samples $\mathbf{c}_{\mathbf{s},i}^\top = \mathbf{r}^\top \mathbf{D} + \mathbf{s}_i^\top \mathbf{G} + \mathbf{e}_{\mathbf{s},i}^\top$, in addition to $\mathbf{c}_i, \mathsf{dct}$ as before. The evaluation proceeds as follows. Step 1) uses GSW homomorphic evaluation to obtain a ciphertext hct_{f_i} encrypting the pad $f_i(\mathbf{r}) = -\mathsf{PAD}_i(\mathbf{r})$. Step 2) computes $\mathsf{PAD}_i(\mathbf{r}) + \mathbf{s}_i \mathbf{A} + \mathbf{e}_i$ as in Equation (7), and Step 3) homomorphically decrypts the GSW ciphertext under dual-GSW to obtain the LWE sample $\tilde{\mathbf{s}}_i^\top \mathbf{A} + \tilde{\mathbf{e}}_i^\top$. The overall correctness is summarized in Equation (8).

Advantage The advantage of Version II is that the joint distribution of the LWE secret $\tilde{\mathbf{s}}_i$ and noise $\tilde{\mathbf{e}}_i$ is, marginally, random. $\tilde{\mathbf{s}}_i$ is uniformly random over $\mathbb{Z}_q$ since $\mathbf{s}_i$ is random and independent of $\mathbf{u}_{f_i}$ (and $\tilde{\mathbf{e}}_i$ is iid random Gaussian as in Version I). Recall again that $\mathbf{u}_{f_i}^\top = \mathbf{v}_{f_i}^\top \cdot \mathbf{U}^\top$ and $\mathbf{v}_{f_i} = \mathsf{vec}\big(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{f_i})\big)$

Encoding

GSW components	Connecting components	dGSW components
$\mathsf{hpk} = \begin{pmatrix} \mathbf{B} \\ \mathbf{r}^\top \mathbf{B} + \mathbf{e}_\mathbf{B}^\top \end{pmatrix}$. $\mathsf{hct}(\mathbf{r}) = \overline{\mathbf{B}}\mathbf{R} + \mathsf{bits}(\mathbf{r})^\top \otimes \mathbf{G}_{n+1}$.	$\{\mathbf{c}_i^\top = (\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top \mod \Delta)\}_{i \in [Q]}$. $\{\mathbf{D}_i\}_{i \in [Q]}$. $\{\mathbf{c}_{\mathbf{s},i}^\top = \mathbf{r}^\top \mathbf{D}_i + \mathbf{e}_{\mathbf{s},i} + \mathbf{s}_i^\top \mathbf{G}\}_{i \in [Q]}$	$\mathbf{A}$ $\mathsf{dct} = \mathbf{U}^\top \mathbf{A} + \mathbf{E} + \mathbf{I}_\ell \otimes \mathbf{G}^\top \mathbf{r}$

Oblivious LWE Sampling:

(1) Evaluate	**(2)** Round and Mult by Δ and Add $\mathbf{c}_i$.	**(3)** Linear decryption for hct_{f_i}.
$f_i(\mathbf{r}) = -\Delta \left\lfloor \frac{\mathbf{r}^\top \mathbf{D}_i \mathbf{G}^{-1}(\mathbf{A})}{\Delta} \right\rceil$. $\mathsf{hct}_{f_i} = \mathsf{Eval}(\mathsf{hct}(\mathbf{s}_i), f_i)$, $\begin{pmatrix} \overline{\mathsf{hct}}_{f_i} \\ \underline{\mathsf{hct}}_{f_i} \end{pmatrix} = \begin{pmatrix} \mathbf{B}\mathbf{R}_{f_i} \\ (\mathbf{r}^\top \mathbf{B} + \mathbf{e}_\mathbf{B}^\top)\mathbf{R}_{f_i} + f_i(\mathbf{s}_i) \end{pmatrix}$	$\Delta \left\lfloor \frac{\mathbf{c}_{\mathbf{s},i}^\top \mathbf{G}^{-1}(\mathbf{A}) - \mathbf{c}_i^\top}{\Delta} \right\rceil + \mathbf{c}_i$ $= \Delta \left\lfloor \frac{\mathbf{r}^\top \mathbf{D} \mathbf{G}^{-1}(\mathbf{A})}{\Delta} \right\rceil + \mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top$	$\mathbf{v}_{f_i} = \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{f_i}))$ $\mathsf{dct}_{f_i} = \mathbf{v}_{f_i}^\top \cdot \mathsf{dct}$ $= \mathbf{u}_{f_i}^\top \mathbf{A} + \mathbf{e}_{f_i}^\top - \mathbf{r}^\top (\mathbf{B}\mathbf{R}_{f_i})$

$$\textbf{Correctness}: \quad \forall i \in [Q],\ \underline{\mathsf{hct}}_{f_i} + \Delta \left\lfloor \frac{\mathbf{c}_{\mathbf{s},i}^\top \mathbf{G}^{-1}(\mathbf{A}) - \mathbf{c}_i^\top}{\Delta} \right\rceil + \mathbf{c}_i^\top + \mathsf{dct}_{f_i} = \tilde{\mathbf{s}}_i^\top \mathbf{A} + \tilde{\mathbf{e}}_i^\top$$
$$\text{where } \tilde{\mathbf{s}}_i^\top = \mathbf{u}_{f_i}^\top + \mathbf{s}_i^\top,\ \tilde{\mathbf{e}}_i^\top = \mathbf{e}_i^\top + \mathbf{e}_\mathbf{B}^\top \mathbf{R}_{f_i} + \mathbf{e}_{f_i}^\top \quad (8)$$

Figure 7: Version II: Special Homomorphic Evaluation Procedure for Computing GSW ciphertexts of $\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top$. Each $\mathbf{c}_{\mathbf{s},i}$ has dimension $n \log q$ and bit length $n \log q \log q$, which is sublinear in $L = \ell \log q$ if $n \log q \ll \ell$.

depends on the top part of the ciphertext hct_{f_i}. Different from Version I, hct_{f_i} is now the result of evaluating $f_i(\mathbf{r})$ and hence is only correlated with $\mathsf{hct}(\mathbf{r})$ and f_i which depends on matrices $\mathbf{D}, \mathbf{A}$, and hence independent of $\mathbf{s}_i$.

Comparison In prior constructions [DQV+21, WW21], the LWE secrets $\tilde{\mathbf{s}}$ produced in the scheme are far from random. In particular, this was leveraged by [JLLS23] to launch a polynomial-time attack on [DQV+21].

Drawbacks Now that $(\tilde{\mathbf{s}}_i, \tilde{\mathbf{e}}_i)$ is jointly random, can we reduce the security of Version II to some LWE-with-hints assumption with random hints? We show below that this could be done, however, the resulting assumption needs to postulate security of LWE-based encodings with an "unnatural" distribution, in particular, they are provably not pseudorandom given the hints.

We observe that there is a reduction $\mathcal{R}_2$ that given a sample from the following "smaller" distribution can emulate the distribution of Version II:

$$\mathsf{Real}_{\mathsf{v2}} : (\mathsf{hpk}, \mathsf{hct}(\mathbf{r}), \{\mathbf{D}_i, \widehat{\mathbf{c}}_i\}_i, \mathbf{A}, \mathsf{dct}),\ \text{where } \widehat{\mathbf{c}}_i^\top = \mathbf{r}^\top \mathbf{D}_i - \mathbf{u}_{f_i}^\top \mathbf{G} + \widehat{\mathbf{e}}_i^\top$$

Above, components $\mathsf{hpk}, \mathsf{hct}(\mathbf{r}), \mathbf{D}, \mathbf{A}, \mathsf{dct}$ are sampled exactly as in Version II. Therefore, the reduction $\mathcal{R}_2$ just needs to emulate the missing components $(\tilde{\mathbf{s}}_i, \tilde{\mathbf{e}}_i, \mathbf{c}_i, \mathbf{c}_{\mathbf{s},i})$ in Version II. Leveraging that $(\tilde{\mathbf{s}}_i, \tilde{\mathbf{e}}_i)$ are random, $\mathcal{R}_2$ *can sample them internally*, which implicitly defines $\mathbf{s}_i^\top = \tilde{\mathbf{s}}_i^\top - \mathbf{u}_{f_i}^\top$ and $\mathbf{e}_i^\top = \tilde{\mathbf{e}}_i^\top - (\mathbf{e}_{f_i}^\top + \mathbf{e}_\mathbf{B}^\top \mathbf{R}_{f_i})$. Next, the correctness constraint (Equation (8)) gives a way to emulate $\mathbf{c}_i$ as follows:

$$\mathbf{c}_i \textbf{ emulation:} \quad \mathbf{c}_i^\top = \left(\tilde{\mathbf{s}}_i^\top \mathbf{A} + \tilde{\mathbf{e}}_i^\top\right) - \underline{\mathsf{hct}}_{f_i} - \mathsf{dct}_{f_i} \pmod{\Delta} \quad (9)$$

Finally, $\mathbf{c}_{\mathbf{s},i}$, which should encrypt $\mathbf{s}_i^\intercal\mathbf{G} = \tilde{\mathbf{s}}_i^\intercal\mathbf{G} - \mathbf{u}_{f_i}^\intercal\mathbf{G}$ can be emulated using $\widehat{\mathbf{c}}_i$:

$$\mathbf{c}_{\mathbf{s},i} \textbf{ emulation:} \quad \mathbf{c}_{\mathbf{s},i}^\intercal = \widehat{\mathbf{c}}_i^\intercal + \tilde{\mathbf{s}}_i^\intercal\mathbf{G} \tag{10}$$

Hence, any property of the distribution in Version II translates to some property of $\mathsf{Real}_{\mathrm{v2}}$, and vice versa.

Examining the distribution $\mathsf{Real}_{\mathrm{v2}}$, it contains circular LWE encodings – $\mathsf{hct}(\mathbf{r})$ circularly encrypts $\mathbf{r}$ under $\mathbf{r}$, dct encrypts $\mathbf{I}_\ell \otimes \mathbf{G}^\intercal \cdot \mathbf{r}$ under $\mathbf{U}$, and $\widehat{\mathbf{c}}_i$ encrypts $\mathbf{u}_{f_i}$ under $\mathbf{r}$. It appears that by the commonly used circular LWE security rationale, one could postulate the pseudorandomness of $\mathsf{Real}_{\mathrm{v2}}$, which would imply some form of the security of Version II.

However, this is false. $\mathsf{Real}_{\mathrm{v2}}$ is provably not pseudorandom, because the correctness condition (Equation (8)) of Version II translates (via $\mathcal{R}_2$) into an efficiently verifiable constraint on $\mathsf{Real}_{\mathrm{v2}}$ that truly random encodings do not satisfy.

To unravel the apparent contradiction, it is instrumental to note that the encoding $\widehat{\mathbf{c}}$ is not a "safe" circular encoding. In general, a circular encoding $\mathbf{t}^\intercal\mathbf{H} + f(\mathbf{t}) + \mathbf{e}^\intercal$ is only secure if the encrypted message $f(\mathbf{t})$ is independent of the encoding randomness $(\mathbf{H}, \mathbf{e})$ (a trivial counterexamples is $f(\mathbf{t}) = -\mathbf{t}^\intercal \cdot \mathbf{H}$). However, $\widehat{\mathbf{c}}$ violates this rule-of-thumb: The message $\mathbf{u}_{f_i}$ depends on the random matrix $\mathbf{D}_i$ used to encode it. The correlation exists because $\mathbf{u}_{f_i}$ depends on the function f_i, which computes $f_i(\mathbf{r}) = -\Delta\lfloor \mathbf{r}^\intercal\mathbf{D}_i\mathbf{G}^{-1}(\mathbf{A})/\Delta\rceil$ and is dependent on $\mathbf{D}_i$.

We distill a take-away message from the above discussion. For any assumption that contains LWE-based encodings, we view the lack of plausible pseudorandomness of the encodings problematic, as it stands at odds with our intuition that security based on LWE encodings relies on their pseudorandomness[8].

Therefore, our goal is to formulate an LWE-with-hint assumption, where the LWE encodings in the real distribution are switched to random in the ideal distribution. Towards this, in Version III we will introduce a URS (uniform random CRS), and show simulation security, namely, the encodings and URS can be simulated using $\tilde{\mathbf{s}}_i^\intercal\mathbf{A} + \tilde{\mathbf{e}}_i^\intercal$.

Comparison In all prior oblivious LWE sampler and xiO constructions [BDGM20, DQV+21, WW21, GP21, BDGM22], the underlying hardness / assumption postulates indistinguishability security, and lack natural pseudorandomness variants of their assumptions.

Version III: GSW Rerandomization, Pseudorandom LWE-with-Hint, and Simulation Security Towards the aforementioned goal of relying on pseudorandom LWE-with-hint assumption and achieving simulation security, Version III uses a technique introduced in [GP21] that re-randomizes the GSW ciphertext before dual-GSW homomorphic decryption, as described in Figure 8. The re-randomization uses sufficiently wide random Gaussian matrices $\mathbf{R}^* = \{\mathbf{R}_i^*\}_{i\in[Q]}$ contained in the URS. In particular, after Step 1) obtaining the GSW ciphertext hct_{f_i}, Step 2) "rerandomizes" the ciphertext to

$$\mathsf{hct}' = \mathsf{hct}_{f_i} + \overline{\mathbf{B}}\mathbf{R}_i^* = \begin{pmatrix} \mathbf{B}\widetilde{\mathbf{R}}_i \\ (\mathbf{r}^\intercal\mathbf{B} + \mathbf{e}_{\mathbf{B}}^\intercal)\widetilde{\mathbf{R}}_i + f_i(\mathbf{r}) \end{pmatrix}, \quad \text{where } \widetilde{\mathbf{R}}_i = (\mathbf{R}_{f_i} + \mathbf{R}_i^*)$$

Following that, evaluation proceeds identically as in Version II.

Advantage We first formulate a distribution $\mathsf{Real}_{\mathrm{v3}}$ from which Version III can be emulated, and

[8] This should be separated from LWE-based constructions, e.g., NIZK, where pseudorandomness does not hold but ZK or indistinguishability holds. Such behaviors are the result of careful design, whereas when formulating assumptions, we are considering LWE encodings that we do not fully know how to analyze.

Encoding:

Common reference string: $\mathsf{crs} = \{\mathbf{R}_i^* \leftarrow \mathcal{D}_{\sigma_0}^{m\times\ell}\}_{i\in[Q]}$.

GSW components

$$\mathsf{hpk} = \begin{pmatrix} \mathbf{B} \\ \mathbf{r}^\top\mathbf{B} + \mathbf{e}_{\mathbf{B}}^\top \end{pmatrix}$$

$$\mathsf{hct}(\mathbf{r}) = \overline{\mathbf{B}}\mathbf{R} + \mathsf{bits}(\mathbf{r})^\top \otimes \mathbf{G}_{n+1}$$

Connecting components

$$\{\mathbf{c}_i^\top = (\mathbf{s}_i^\top\mathbf{A} + \mathbf{e}_i^\top \mod \Delta)\}_{i\in[Q]}$$

$$\{\mathbf{D}_i\}_{i\in[Q]}$$

$$\{\mathbf{c}_{\mathbf{s},i}^\top = \mathbf{r}^\top\mathbf{D}_i + \mathbf{e}_{\mathbf{s},i}^\top + \mathbf{s}_i^\top\mathbf{G}\}_{i\in[Q]}$$

dGSW components

$$\mathbf{A}$$

$$\mathsf{dct} = \mathbf{U}^\top\mathbf{A} + \mathbf{E} + \mathbf{I}_\ell \otimes \mathbf{G}^\top\mathbf{r}$$

Oblivious LWE Sampling:

(1) Evaluate

$$f_i(\mathbf{r}) = -\Delta\left\lfloor \frac{\mathbf{r}^\top\mathbf{D}_i\mathbf{G}^{-1}(\mathbf{A})}{\Delta} \right\rceil$$

$$\mathsf{hct}_{f_i} = \mathsf{Eval}(\mathsf{hct}(\mathbf{s}_i), f_i),$$

$$\begin{pmatrix} \overline{\mathsf{hct}}_{f_i} \\ \underline{\mathsf{hct}}_{f_i} \end{pmatrix} = \begin{pmatrix} \mathbf{B}\mathbf{R}_{f_i} \\ (\mathbf{r}^\top\mathbf{B} + \mathbf{e}_{\mathbf{B}}^\top)\mathbf{R}_{f_i} + f_i(\mathbf{s}_i) \end{pmatrix}$$

(2) Rerandomization

$$\mathsf{hct}_i' = \mathsf{hct}_{f_i} + \overline{\mathbf{B}}\mathbf{R}_i^*$$

$$\begin{pmatrix} \overline{\mathsf{hct}}_i' \\ \underline{\mathsf{hct}}_i' \end{pmatrix} = \begin{pmatrix} \mathbf{B}\widetilde{\mathbf{R}}_i \\ (\mathbf{r}^\top\mathbf{B} + \mathbf{e}_{\mathbf{B}}^\top)\widetilde{\mathbf{R}}_i + f_i(\mathbf{r}) \end{pmatrix},$$

where $\widetilde{\mathbf{R}}_i = \mathbf{R}_{f_i} + \mathbf{R}_i^*$

(3) Round and Mult by Δ and Add

$$\left\lfloor \frac{\mathbf{c}_{\mathbf{s},i}^\top\mathbf{G}^{-1}(\mathbf{A}) - \mathbf{c}_i^\top}{\Delta} \right\rceil + \mathbf{c}_i^\top$$

$$= \left\lfloor \frac{\mathbf{r}^\top\mathbf{D}\mathbf{G}^{-1}(\mathbf{A})}{\Delta} \right\rceil + \mathbf{s}_i^\top\mathbf{A} + \mathbf{e}_i^\top$$

(4) Linear decryption for hct_i'

$$\mathbf{v}_i = \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_i'))$$

$$\begin{aligned} \mathsf{dct}_i &= \mathbf{v}_i^\top \cdot \mathsf{dct} \\ &= \mathbf{u}_i^\top\mathbf{A} + \mathbf{e}_i'^\top - \mathbf{r}^\top(\mathbf{B}\widetilde{\mathbf{R}}_i) \\ &(\mathbf{u}_i^\top = \mathbf{v}_i^\top\mathbf{U}^\top,\ \mathbf{e}_i'^\top = \mathbf{v}_i^\top\mathbf{E}) \end{aligned}$$

$$\textbf{Correctness}: \quad \forall i \in [Q],\ \underline{\mathsf{hct}}_i' + \Delta\left\lfloor \frac{\mathbf{c}_{\mathbf{s},i}^\top\mathbf{G}^{-1}(\mathbf{A}) - \mathbf{c}_i^\top}{\Delta} \right\rceil + \mathbf{c}_i^\top + \mathsf{dct}_i = \tilde{\mathbf{s}}_i^\top\mathbf{A} + \tilde{\mathbf{e}}_i^\top$$
$$\text{where } \tilde{\mathbf{s}}_i = \mathbf{u}_i + \mathbf{s}_i,\ \tilde{\mathbf{e}}_i^\top = \mathbf{e}_i^\top + \mathbf{e}_{\mathbf{B}}^\top\widetilde{\mathbf{R}}_i + \mathbf{e}_i'^\top,\ \mathbf{e}_i'^\top = \mathbf{v}_i\mathbf{E} \tag{11}$$

Figure 8: Version III: Rerandomizing the GSW ciphertext.

show that the LWE encodings in $\mathsf{Real}_{\mathrm{v3}}$ now follow sound circular security rationale.

$$\mathsf{Real}_{\mathrm{v3}} : \mathsf{encodings} = \left(\mathsf{hpk} = \overline{\mathbf{B}},\ \mathsf{hct}(\mathbf{r}),\ \{\mathsf{hct}_{0,i} = \overline{\mathbf{B}}\widetilde{\mathbf{R}}_i\},\ \mathbf{D},\ \{\widehat{\mathbf{c}}_i\},\ \mathbf{A}, \mathsf{dct}\right),\ \mathsf{hint} = \{\mathbf{R}_i^*\}$$
$$\text{where } \widetilde{\mathbf{R}}_i = \mathbf{R}_{f_i} + \mathbf{R}_i^*,\ \widehat{\mathbf{c}}_i = \mathbf{r}^\top\mathbf{D}_i + \widehat{\mathbf{e}}_i^\top - \mathbf{u}_i^\top\mathbf{G},\ \mathbf{u}_i^\top\mathbf{G} = \mathsf{vec}(\mathbf{G}^{-1}(-\mathbf{B}\widetilde{\mathbf{R}}))^\top\mathbf{U}^\top\mathbf{G} = f^{\mathrm{circ}}(\mathbf{U}, \mathsf{hct}_0)\ .$$

Additionally, $(\mathsf{hpk}, \mathsf{hct}(\mathbf{r}), \mathbf{D}, \mathbf{A}, \mathsf{dct}, \mathbf{R}^*)$ are sampled, and $\mathbf{R}_{f_i}, \mathbf{u}_i$ computed just as in Version III. To emulate the full distribution of Version III, a reduction $\mathcal{R}_3$ given a sample from $\mathsf{Real}_{\mathrm{v3}}$ needs to emulate the missing terms $(\tilde{\mathbf{s}}_i, \tilde{\mathbf{e}}_i, \mathbf{c}_i, \mathbf{c}_{\mathbf{s},i})$ similarly to $\mathcal{R}_2$. Because the distribution of $\tilde{\mathbf{s}}_i, \tilde{\mathbf{e}}_i$ are random and independent of all the components in $\mathsf{Real}_{\mathrm{v3}}$, they can be sampled by $\mathcal{R}_3$ internally, $\mathbf{c}_{\mathbf{s},i}$ is emulated by $\widehat{\mathbf{c}}_i + \tilde{\mathbf{s}}_i^\top\mathbf{G}$ as in Equation (9), and $\mathbf{c}_i$ is emulated according to the new correctness condition (Equation (11)):

$$\textbf{New } \mathbf{c}_i \textbf{ emulation:} \quad \mathbf{c}_i = \left(\tilde{\mathbf{s}}_i^\top\mathbf{A} + \tilde{\mathbf{e}}_i^\top\right) - \underline{\mathsf{hct}}_{0,i} - \mathsf{dct}_i \pmod{\Delta} \tag{12}$$

Thanks to re-randomization, the LWE encodings $\mathsf{encodings}$ in $\mathsf{Real}_{\mathrm{v3}}$ are now safe circular encoding. $\mathsf{hct}(\mathbf{r}), \mathsf{hct}_{0,i}, \mathsf{dct}$ are standard circular encodings. We further observe that $\widehat{\mathbf{c}}_i$ now encrypts a message $\mathbf{u}_i$ using *independent* randomness $(\mathbf{D}, \widehat{\mathbf{e}})$. This is because $\mathbf{u}_i$ depends on $\mathbf{U}$ and $\mathbf{B}\widetilde{\mathbf{R}}_i$. Thanks to smudging, $\widetilde{\mathbf{R}}_i = \mathbf{R}_{f_i} + \mathbf{R}_i^*$ is a random Gaussian matrix and hence $\mathbf{u}_i$ is independent of $\mathbf{D}$ and $\widehat{\mathbf{e}}$. Therefore, by circular security rationale, the $\mathsf{encodings}$ alone is pseudorandom. This overcomes the drawback of Version II.

Our Pseudorandom LWE-with-hints Assumption: We explore whether the LWE encodings $\mathsf{encodings}$ is still pseudorandom, at the presence of hint, by formulating an LWE-with-hints assumption. While $\mathsf{encodings}$ alone is pseudorandom by circular security, and $\mathbf{R}_i^*$'s are marginally random, their joint distribution is subject to a constraint implied by the correctness condition of Version III. Hence, the main question is when $\mathsf{encodings}$ is switched to random in an ideal distribution, how should the distribution of $\mathbf{R}_i^*$ change accordingly to ensure that the constraint is still satisfied?

Let's examine the distribution of $\mathbf{R}_i^*$. In the real distribution $\mathsf{Real}_{\mathrm{v3}}$, it is a random Gaussian matrix subject to the following constraint:

$$\mathbf{R}_i^* \leftarrow \mathcal{D}_{\sigma_0}^{m\times\ell},\ \textit{conditioned on } \mathsf{hct}_{f_i} + \overline{\mathbf{B}}\mathbf{R}_i^* = \mathsf{hct}_{0,i} + \begin{pmatrix} \mathbf{0} \\ f_i(\mathbf{r}) \end{pmatrix}$$

Furthermore, the correctness equality of Version III is equivalent to an equality showing that $f_i(\mathbf{r})$ can be computed publicly from existing LWE encodings $(\mathsf{hct}_{0,i}, \widehat{\mathbf{c}}_i, \mathsf{dct})$ with overwhelming probability.

Claim 1. *The correctness constraint in Version III is equivalent to the following equality:*

$$f_i(\mathbf{r}) \overset{w.h.p}{=} \widetilde{f}_i(\mathsf{hct}_{0,i}, \widehat{\mathbf{c}}_i, \mathsf{dct}) = \left\lfloor \frac{-\widehat{\mathbf{c}}_i^{\mathsf{T}}\mathbf{G}^{-1}(\mathbf{A}) + \mathsf{dct}_i - \underline{\mathsf{hct}}_{0,i}}{\Delta} \right\rceil \cdot \Delta,\ \ \mathsf{dct}_i = \mathrm{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i}))^{\mathsf{T}} \cdot \mathsf{dct}$$

Proof. The proof follows from algebraic derivation, starting from the correctness of Version III.

$$\begin{aligned}
\tilde{\mathbf{s}}_i^{\mathsf{T}}\mathbf{A} + \tilde{\mathbf{e}}_i^{\mathsf{T}} &= \underline{\mathsf{hct}}_{0,i} + f_i(\mathbf{r}) + \Delta\left\lfloor \frac{\mathbf{c}_{\mathbf{s}_i,i}^{\mathsf{T}}\mathbf{G}^{-1}(\mathbf{A}) - \mathbf{c}_i^{\mathsf{T}}}{\Delta} \right\rceil + \mathbf{c}_i^{\mathsf{T}} + \mathsf{dct}_i \\
f_i(\mathbf{r}) &= \tilde{\mathbf{s}}_i^{\mathsf{T}}\mathbf{A} + \tilde{\mathbf{e}}_i^{\mathsf{T}} - \underline{\mathsf{hct}}_{0,i} - \mathsf{dct}_i - \mathbf{c}_i^{\mathsf{T}} - \Delta\left\lfloor \frac{\mathbf{c}_{\mathbf{s}_i,i}^{\mathsf{T}}\mathbf{G}^{-1}(\mathbf{A}) - \mathbf{c}_i^{\mathsf{T}}}{\Delta} \right\rceil \\
&= \tilde{\mathbf{s}}_i^{\mathsf{T}}\mathbf{A} + \tilde{\mathbf{e}}_i^{\mathsf{T}} - \underline{\mathsf{hct}}_{0,i} - \mathsf{dct}_i - \mathbf{c}_i^{\mathsf{T}} - \left(\mathbf{c}_{\mathbf{s}_i,i}^{\mathsf{T}}\mathbf{G}^{-1}(\mathbf{A}) - \mathbf{c}_i^{\mathsf{T}}\right) + \left(\left(\mathbf{c}_{\mathbf{s}_i,i}^{\mathsf{T}}\mathbf{G}^{-1}(\mathbf{A}) - \mathbf{c}_i^{\mathsf{T}}\right) \bmod \Delta\right) \\
&= \tilde{\mathbf{s}}_i^{\mathsf{T}}\mathbf{A} + \tilde{\mathbf{e}}_i^{\mathsf{T}} - \underline{\mathsf{hct}}_{0,i} - \mathsf{dct}_i - \mathbf{c}_i^{\mathsf{T}} - (\widehat{\mathbf{c}}_i^{\mathsf{T}}\mathbf{G}^{-1}(\mathbf{A}) + \tilde{\mathbf{s}}_i^{\mathsf{T}}\mathbf{A} - \mathbf{c}_i^{\mathsf{T}}) + ((\widehat{\mathbf{c}}_i^{\mathsf{T}}\mathbf{G}^{-1}(\mathbf{A}) + \tilde{\mathbf{s}}_i^{\mathsf{T}}\mathbf{A} - \mathbf{c}_i^{\mathsf{T}}) \bmod \Delta) \\
&= \tilde{\mathbf{e}}_i^{\mathsf{T}} - \underline{\mathsf{hct}}_{0,i} - \mathsf{dct}_i - \widehat{\mathbf{c}}_i^{\mathsf{T}}\mathbf{G}^{-1}(\mathbf{A}) + \left(\left(\widehat{\mathbf{c}}_i^{\mathsf{T}}\mathbf{G}^{-1}(\mathbf{A}) + \tilde{\mathbf{s}}_i^{\mathsf{T}}\mathbf{A} - \left(\tilde{\mathbf{s}}_i^{\mathsf{T}}\mathbf{A} + \tilde{\mathbf{e}}_i^{\mathsf{T}} - \underline{\mathsf{hct}}_{0,i} - \mathsf{dct}_i\right)\right) \bmod \Delta\right) \\
&= \Delta\left\lfloor \frac{\tilde{\mathbf{e}}_i^{\mathsf{T}} - \underline{\mathsf{hct}}_{0,i} - \mathsf{dct}_i - \widehat{\mathbf{c}}_i^{\mathsf{T}}\mathbf{G}^{-1}(\mathbf{A})}{\Delta} \right\rceil \\
&\overset{w.h.p}{=} \Delta\left\lfloor \frac{-\underline{\mathsf{hct}}_{0,i} - \mathsf{dct}_i - \widehat{\mathbf{c}}_i^{\mathsf{T}}\mathbf{G}^{-1}(\mathbf{A})}{\Delta} \right\rceil = \widetilde{f}_i(\mathsf{hct}_{0,i}, \widehat{\mathbf{c}}_i, \mathsf{dct})
\end{aligned}$$

The fourth equality follows from emulation of $\mathbf{c}_{\mathbf{s}_i,i}$ by (10). The fifth equality follows from emulation of $\mathbf{c}_i$ by (12). The last equality follows with high probability since the noise $\tilde{\mathbf{e}}_i$ is small. □

We can now formulate our pseudorandom LWE-with-hints assumption:

$$\begin{array}{rl}
& \mathsf{encodings} = (\ \ \mathsf{hpk} = \overline{\mathbf{B}},\ \ \mathsf{hct}(\mathbf{r}),\ \ \{\mathsf{hct}_{0,i} = \overline{\mathbf{B}}\widetilde{\mathbf{R}}_i\},\ \ \mathbf{D},\ \ \{\widehat{\mathbf{c}}_i\},\ \ \mathbf{A},\ \ \mathsf{dct}\ \ ),\ \ \mathsf{hint} = \{\mathbf{R}_i^*\} \\
\approx & \mathsf{encodings} = (\ \ \$,\ \ \$,\ \ \{\$\},\ \ \$,\ \ \{\$\},\ \ \$,\ \ \$\ \ ),\ \ \mathsf{hint} = \{\mathbf{R}_i^*\}
\end{array}$$

$$\text{where } \mathbf{R}_i^* \leftarrow \mathcal{D}_{\sigma_0}^{m\times\ell},\ \textit{conditioned on } \mathsf{hct}_{f_i} + \overline{\mathbf{B}}\mathbf{R}_i^* = \mathsf{hct}_{0,i} + \begin{pmatrix} \mathbf{0} \\ \widetilde{f}_i(\mathsf{hct}_{0,i}, \widehat{\mathbf{c}}_i, \mathsf{dct}) \end{pmatrix}$$

Note that the constraint on $\mathbf{R}^*$ is efficiently verifiable. In the real distribution, the $\mathsf{encodings}$ contains honestly generated LWE encodings, and $\mathbf{R}^*$ follows the Gaussian distribution subject to the constraint, while in the ideal distribution, $\mathsf{encodings}$ is truly random and $\mathbf{R}^*$ is still Gaussian subject to the constraint. Furthermore, the marginal distribution of $\mathbf{R}_i^*$ is truly random Gaussian in the real distribution (Theorem 1), and is pseudorandom Gaussian in the ideal distribution (Theorem 2). Our assumption postulates that these two distributions are indistinguishable. We show that it resists existing attacks and cryptanalytic techniques in Section 2.3.2.

Simulation Security: Our assumption immediately enables proving simulation security. Given $\tilde{\mathbf{s}}_i^\intercal \mathbf{A} + \tilde{\mathbf{e}}_i^\intercal$, a simulator simulates the LWE encodings and the CRS in Version III as follows: It samples $\mathsf{encodings}$ at random and $\mathbf{R}_i^*$ as in the ideal distribution. Note that sampling $\mathbf{R}_i^*$ as Gaussian with width σ_0 conditioned on $\overline{\mathbf{B}}\mathbf{R}_i^*$ being equal to a target matrix can be done efficiently if $\overline{\mathbf{B}}$ in $\mathsf{encodings}$ is sampled together with a trapdoor. Then the simulator invokes $\mathcal{R}_3$ to simulate the rest components in Version III. We note that the output LWE samples $\tilde{\mathbf{s}}_i^\intercal \mathbf{A} + \tilde{\mathbf{e}}_i^\intercal$ are "programmed" in $\mathbf{R}_i^*$ in the CRS (note this is the alternative simulation strategy of GSW).

Construction of Functional Encoding: Once we have an oblivious LWE sampler, it becomes easy to construct a functional encoding. The high-level idea is that the CRS of the functional encoding is exactly the CRS of the oblivious LWE sampler, namely $\mathbf{R}^*$. The functional encoding of an input $\mathbf{x}$ includes all the encodings in the oblivious LWE sampler, and additionally a dual-GSW ciphertext of the binary input $\mathbf{x}$.

$$\mathsf{dct}(\mathbf{x}) = \mathbf{W}^\intercal \mathbf{A} + \mathbf{E}_\mathbf{x} + \mathbf{x} \otimes \mathbf{G}_\ell^\intercal \overset{\mathsf{dGSW.Eval}}{\Longrightarrow} \mathsf{dct}_{g_i} = \mathbf{w}_{g_i}^\intercal \mathbf{A} + \mathbf{e}_{g_i}^\intercal + g_i(\mathbf{x})^\intercal, \text{ for } g_i(\mathbf{x}) \in \mathbb{Z}_q^\ell$$

Using the homomorphic evaluation of dual-GSW, we can obtain a ciphertext of the output $g_i(\mathbf{x}) \in \mathbb{Z}_q^\ell$. To reveal the output, instead of opening $\mathbf{w}_{g_i}$ which compromises security, we re-randomize dct_{g_i} using the obliviously sampled LWE samples and open $\mathbf{w}_{g_i} + \tilde{\mathbf{s}}_i$, which reveals $g_i(\mathbf{x}) + \mathbf{e}_{g_i} + \tilde{\mathbf{e}}_i$ and hence the high order bits of $g_i(\mathbf{x})$. Thanks to the fact that $\tilde{\mathbf{s}}_i, \tilde{\mathbf{e}}_i$ are pseudorandom, $\mathbf{w}_{g_i}$ and $\mathbf{e}_{g_i}$ are now hidden. By a similar simulation strategy as above, we can show simulation security of the functional encoding.

5 Constructing Functional Encoding from CRO

We now present the formal construction and proof of functional encoding from the CRO assumption.

Construction 1. *In the construction, we always use the discrete Gaussian distribution with bounded norm. In particular, we use the notation $\overline{\mathcal{D}}_\sigma$ to denote the conditional distribution $x \leftarrow \mathcal{D}_\sigma\big|_{|x| \leq \sigma\sqrt{\lambda}}$.*

- $\mathsf{Gen}(1^\lambda, 1^Q, 1^K, 1^L, 1^d)$: *The algorithm picks appropriate parameters* $\mathsf{pp} = (n, m, \Delta, \kappa, p, q, \ell, \sigma, \sigma_0, \sigma_\mathbf{e})$ *samples GSW rerandomization matrix*

$$\mathbf{R}^* = (\mathbf{R}_1^*, \ldots, \mathbf{R}_Q^*) \leftarrow \overline{\mathcal{D}}_{\sigma_0}^{m \times Q\ell},$$

  and output $\mathsf{crs} = (\mathsf{pp}, \mathbf{R}^*)$.

  Note: The parameters need to satisfy certain relations. We specify them below when a relation is first needed, and set the parameters at the end of the section.

- $\mathsf{Enc}(\mathsf{crs}, \mathbf{x}; R)$: *The algorithm on input the* crs, *and a binary input* $\mathbf{x} \in \{0,1\}^K$, *generates the following components.*

- *GSW components:*
  - *Public key* $\mathsf{hpk} = \overline{\mathbf{B}} = \begin{pmatrix} \mathbf{B} \\ \mathbf{r}^\intercal\mathbf{B} + \mathbf{e}_\mathbf{B}^\intercal \end{pmatrix}$*, where* $m = \Theta(n \log q)$*,* $\mathbf{r} \leftarrow \mathbb{Z}_q^n$*,* $\mathbf{B} \leftarrow \mathbb{Z}_q^{n\times m}$*,* $\mathbf{e}_\mathbf{B} \leftarrow \overline{\mathcal{D}}_\sigma^m$
  - *Ciphertext* $\mathsf{hct} = \overline{\mathbf{B}}\mathbf{R} + \mathsf{bits}(\mathbf{r})^\intercal \otimes \mathbf{G}$*, where* $\mathbf{R} \leftarrow \{0,1\}^{m\times n(n+1)\lceil \log q\rceil^2}$
- *Connecting components*
  - *Public matrix* $\mathbf{A} \leftarrow \mathbb{Z}_q^{n\times \ell}$*, where* ℓ *is an appropriate sublinear quantity in the function output length L as described in Theorem 6.*
  - *Public matrix* $\mathbf{D} \leftarrow \mathbb{Z}_q^{n\times Qn\lceil \log q\rceil}$*, where* $\mathbf{D} = (\mathbf{D}_1, \dots, \mathbf{D}_Q)$*.*
  - *For* $i \in [Q]$*,* $\mathbf{c}_i = (\mathbf{s}_i^\intercal\mathbf{A} + \mathbf{e}_i^\intercal \pmod{\Delta})$*, where* $\mathbf{s}_i \leftarrow \mathbb{Z}_q^n$*,* $\mathbf{e}_i \leftarrow \overline{\mathcal{D}}_{\sigma_\mathbf{e}}^\ell$*.*
  - *For* $i \in [Q]$*,* $\mathbf{c}_{\mathbf{s},i} = \mathbf{r}^\intercal\mathbf{D}_i + \mathbf{e}_{\mathbf{s},i}^\intercal + \mathbf{s}_i^\intercal\mathbf{G}_n$*, where* $\mathbf{e}_{\mathbf{s},i} \leftarrow \overline{\mathcal{D}}_\sigma^{n\lceil \log q\rceil}$
- *Dual GSW components*
  - *Ciphertext for secret key* $\mathsf{dct} = \mathbf{U}^\intercal\mathbf{A} + \mathbf{E}_\mathbf{A} + \mathbf{I}_\ell \otimes \mathbf{G}_n^\intercal\mathbf{r}$*, where* $\mathbf{U} \leftarrow \mathbb{Z}_q^{n\times \ell n\lceil \log q\rceil}$*,* $\mathbf{E}_\mathbf{A} \leftarrow \overline{\mathcal{D}}_\sigma^{\ell n\lceil \log q\rceil\times \ell}$*.*
  - *Ciphertext for message* $\mathsf{dct}_\mathbf{x} = \mathbf{W}^\intercal\mathbf{A} + \mathbf{E}_\mathbf{x} + \mathbf{x} \otimes \mathbf{G}_\ell^\intercal$*, where* $\mathbf{w} \leftarrow \mathbb{Z}_q^{n\times K\ell\lceil \log q\rceil}$*,* $\mathbf{E}_\mathbf{x} \leftarrow \overline{\mathcal{D}}_\sigma^{K\ell\lceil \log q\rceil\times \ell}$*.*

The algorithm outputs

$$\mathsf{ct} = (\mathsf{hpk}, \mathsf{hct}, \mathbf{D}, \{\mathbf{c}_i, \mathbf{c}_{\mathbf{s}_i}\}_{i\in[Q]}, \mathbf{A}, \mathsf{dct}, \mathsf{dct}_\mathbf{x}).$$

- $\mathsf{Open}(\mathsf{crs}, g, i, \mathbf{x}, R)$*: The algorithm first runs the encryption algorithm* $\mathsf{Enc}(\mathsf{crs}, \mathbf{x}; R)$ *to recompute all intermediate variables generated in the encryption process as described above, and then computes the opening as follows.*

  0. *Parse function* $g \colon \{0,1\}^K \to \{0,1\}^L$ *as* $g \colon \{0,1\}^K \to \mathbb{Z}_{2^\kappa}^\ell$*, where every* κ*-bit chunk of the output is parsed as a* κ *bit integer in* $[0, 2^\kappa - 1]$*. Define* $\widetilde{g} \colon \{0,1\}^K \to \mathbb{Z}_q^\ell$ *where* $\widetilde{g}(\mathbf{x}) = p \cdot g(\mathbf{x}) \bmod q$*. We require that* $q > 2^\kappa p$*.*
  1. *Homomorphically evaluate the function* $\widetilde{g}$ *over the dual-GSW ciphertext* $\mathsf{dct}_\mathbf{x}$ *to obtain*

  $$\mathsf{dct}_{\widetilde{g}} = \mathbf{w}_{\widetilde{g}}^\intercal\mathbf{A} + \mathbf{e}_{\widetilde{g}}^\intercal + \widetilde{g}(\mathbf{x})^\intercal,$$

  where $\mathbf{w}_{\widetilde{g}}$ *is efficiently computable.*

  2. *H̄omomorphically evaluate function* f_i *where* $f_i(\mathbf{r})^\intercal = \Delta\left\lfloor \frac{\mathbf{r}^\intercal\mathbf{D}_i\mathbf{G}^{-1}(\mathbf{A})}{\Delta}\right\rceil$ *over the GSW ciphertext* hct *to obtain*

  $$\mathsf{hct}_{f_i} = \begin{pmatrix} \overline{\mathsf{hct}}_{f_i} \\ \underline{\mathsf{hct}}_{f_i} \end{pmatrix} = \begin{pmatrix} \mathbf{B}\mathbf{R}_{f_i} \\ (\mathbf{r}^\intercal\mathbf{B} + \mathbf{e}_\mathbf{B}^\intercal)\mathbf{R}_{f_i} + f_i(\mathbf{r})^\intercal \end{pmatrix}$$

  3. *Rerandomize* $-\mathsf{hct}_{f_i}$ *by*

  $$\begin{pmatrix} \overline{\mathsf{hct}}'_i \\ \underline{\mathsf{hct}}'_i \end{pmatrix} = \mathsf{hct}'_i = \overline{\mathbf{B}}\mathbf{R}_i^* - \mathsf{hct}_{f_i} = \begin{pmatrix} \mathbf{B}\widetilde{\mathbf{R}}_i \\ (\mathbf{r}^\intercal\mathbf{B} + \mathbf{e}_\mathbf{B}^\intercal)\widetilde{\mathbf{R}}_i - f_i(\mathbf{r})^\intercal \end{pmatrix},$$

  where $\widetilde{\mathbf{R}}_i = \mathbf{R}_i^* - \mathbf{R}_{f_i}$*.*

4. *Homomorphically decrypt* hct'_i *using dual GSW ciphertext* dct *by*

$$\mathbf{v}_i = \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}'_i)), \quad \mathsf{dct}_i = \mathbf{v}_i^\top \cdot \mathsf{dct} = \mathbf{u}_i^\top \mathbf{A} + \mathbf{e}_i'^\top - \mathbf{r}^\top (\mathbf{B}\widetilde{\mathbf{R}}_i),$$

where $\mathbf{u}_i^\top = \mathbf{v}_i^\top \mathbf{U}^\top$ *is efficiently computable.*

Finally, the algorithm outputs the opening $\rho_i = \mathbf{u}_i + \mathbf{w}_{\widetilde{g}} + \mathbf{s}_i$.

- $\mathsf{Dec}(\mathsf{crs}, g, i, \mathsf{ct}, \rho)$: *The algorithm first parses*

$$\mathsf{ct} = (\mathsf{hpk}, \mathsf{hct}, \mathbf{D}, \{\mathbf{c}_i, \mathbf{c}_{\mathbf{s}_i}\}_{i \in [Q]}, \mathbf{A}, \mathsf{dct}, \mathsf{dct}_{\mathbf{x}}).$$

Following the same procedure as the opening algorithm, the decryption algorithm computes

- *Dual GSW ciphertext*

$$\mathsf{dct}_{\widetilde{g}} = \mathbf{w}_{\widetilde{g}}^\top \mathbf{A} + \mathbf{e}_{\widetilde{g}}^\top + \widetilde{g}(\mathbf{x})^\top.$$

- *Rerandomized GSW ciphertext*

$$\begin{pmatrix} \overline{\mathsf{hct}}'_i \\ \underline{\mathsf{hct}}'_i \end{pmatrix} = \mathsf{hct}'_i = \begin{pmatrix} \mathbf{B}\widetilde{\mathbf{R}}_i \\ (\mathbf{r}^\top \mathbf{B} + \mathbf{e}_{\mathbf{B}}^\top)\widetilde{\mathbf{R}}_i - f_i(\mathbf{r})^\top \end{pmatrix}.$$

- *Dual GSW ciphertext*

$$\mathsf{dct}_i = \mathbf{u}_i^\top \mathbf{A} + \mathbf{e}_i'^\top - \mathbf{r}^\top (\mathbf{B}\widetilde{\mathbf{R}}_i).$$

The algorithm approximately decodes $\widetilde{g}(\mathbf{x})$ *by*

$$\widetilde{g}(\mathbf{x})^\top \approx \tilde{\mathbf{y}}^\top = \underline{\mathsf{hct}}'_i + \mathsf{dct}_i + \mathsf{dct}_{\widetilde{g}} + \Delta \left\lfloor \frac{\mathbf{c}_{\mathbf{s},i}^\top \mathbf{G}^{-1}(\mathbf{A}) - \mathbf{c}_i^\top}{\Delta} \right\rceil + \mathbf{c}_i^\top - \rho^\top \mathbf{A}. \tag{13}$$

The algorithm outputs bit string y*, which extracted from* $\mathsf{bits}(\left\lfloor \frac{\tilde{\mathbf{y}}}{p} \right\rceil)$.

Theorem 5 (Correctness). *Suppose* $\Delta | q$, $m = \Omega(n \log q)$, $q \leq 2^n$, $\sigma_0 \geq m^{\Omega(\log n)}$, $\sigma_{\mathbf{e}} \geq \ell^{\Omega(d)} \sigma \sigma_0$ $\Delta \geq 2^\lambda \sigma_{\mathbf{e}}$, *and* $p \geq 8\Delta$, *where* $m, \sigma_0, \sigma_{\mathbf{e}}$ *are sufficiently large, then Construction 1 is correct.*

Moreover, the following correctness condition holds with probability $1 - 2^{-\Omega(\lambda)}$.

$$\underline{\mathsf{hct}}'_i + \mathsf{dct}_i + \mathsf{dct}_{\widetilde{g}} + \Delta \left\lfloor \frac{\mathbf{c}_{\mathbf{s},i}^\top \mathbf{G}^{-1}(\mathbf{A}) - \mathbf{c}_i^\top}{\Delta} \right\rceil + \mathbf{c}_i^\top - \rho^\top \mathbf{A} = \widetilde{g}(\mathbf{x})^\top + \tilde{\mathbf{e}}_i^\top \tag{14}$$

$$\tilde{\mathbf{e}}^\top = \mathbf{e}_{\mathbf{B}}^\top \widetilde{\mathbf{R}}_i + \mathbf{e}_{\widetilde{g}}^\top + \mathbf{e}_i'^\top + \mathbf{e}_i^\top$$

Proof. First observe that in the decryption equation 13, the connecting components satisfy

$$\begin{aligned}
\Delta \cdot \left\lfloor \frac{\mathbf{c}_{\mathbf{s},i}^\top \mathbf{G}^{-1}(\mathbf{A}) - \mathbf{c}_i^\top}{\Delta} \right\rceil + \mathbf{c}_i^\top &= \Delta \cdot \left\lfloor \frac{\mathbf{r}^\top \mathbf{D}_i \mathbf{G}^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{s},i}^\top \mathbf{G}^{-1}(\mathbf{A}) + \mathbf{s}_i^\top \mathbf{A} - (\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top \bmod \Delta)}{\Delta} \right\rceil + \mathbf{c}_i^\top \\
&= \Delta \cdot \left\lfloor \frac{\mathbf{r}^\top \mathbf{D}_i \mathbf{G}^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{s},i}^\top \mathbf{G}^{-1}(\mathbf{A}) - \mathbf{e}_i^\top}{\Delta} + \frac{\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top - (\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top \bmod \Delta)}{\Delta} \right\rceil + \mathbf{c}_i^\top \\
&= \Delta \cdot \left\lfloor \frac{\mathbf{r}^\top \mathbf{D}_i \mathbf{G}^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{s},i}^\top \mathbf{G}^{-1}(\mathbf{A}) - \mathbf{e}_i^\top}{\Delta} \right\rceil + \Delta \cdot \left\lfloor \frac{\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top}{\Delta} \right\rceil + (\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top \bmod \Delta) \\
&= \Delta \cdot \left\lfloor \frac{\mathbf{r}^\top \mathbf{D}_i \mathbf{G}^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{s},i}^\top \mathbf{G}^{-1}(\mathbf{A}) - \mathbf{e}_i^\top}{\Delta} \right\rceil + \mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top.
\end{aligned}$$

Therefore, by expanding the decryption equation, we get

$$\begin{aligned}
\tilde{\mathbf{y}}^{\top} &= (\mathbf{r}^{\top}\mathbf{B} + \mathbf{e}_{\mathbf{B}}^{\top})\widetilde{\mathbf{R}}_i - \Delta\left\lfloor\frac{\mathbf{r}^{\top}\mathbf{D}_i\mathbf{G}^{-1}(\mathbf{A})}{\Delta}\right\rceil + \mathbf{w}_{\widetilde{g}}^{\top}\mathbf{A} + \mathbf{e}_{\widetilde{g}}^{\top} + \widetilde{g}(\mathbf{x})^{\top} + \mathbf{u}_i^{\top}\mathbf{A} + \mathbf{e}_i'^{\top} - \mathbf{r}^{\top}(\mathbf{B}\widetilde{\mathbf{R}}_i) \\
&\qquad + \Delta\cdot\left\lfloor\frac{\mathbf{r}^{\top}\mathbf{D}_i\mathbf{G}^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{s},i}^{\top}\mathbf{G}^{-1}(\mathbf{A}) - \mathbf{e}_i^{\top}}{\Delta}\right\rceil + \mathbf{s}_i^{\top}\mathbf{A} + \mathbf{e}_i^{\top} - (\mathbf{u}_i + \mathbf{w}_{\widetilde{g}} + \mathbf{s}_i)^{\top}\mathbf{A} \\
&= \widetilde{g}(\mathbf{x})^{\top} + \mathbf{e}_{\mathbf{B}}^{\top}\widetilde{\mathbf{R}}_i + \mathbf{e}_{\widetilde{g}}^{\top} + \mathbf{e}_i'^{\top} + \mathbf{e}_i^{\top} + \Delta\eta_i, \qquad (15) \\
&\qquad \text{where } \eta_i = \left(\left\lfloor\frac{\mathbf{r}^{\top}\mathbf{D}_i\mathbf{G}^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{s},i}^{\top}\mathbf{G}^{-1}(\mathbf{A}) - \mathbf{e}_i^{\top}}{\Delta}\right\rceil - \left\lfloor\frac{\mathbf{r}^{\top}\mathbf{D}_i\mathbf{G}^{-1}(\mathbf{A})}{\Delta}\right\rceil\right).
\end{aligned}$$

The error terms in the above equation are bounded by

$$\begin{aligned}
\left\|\mathbf{e}_{\mathbf{B}}^{\top}\widetilde{\mathbf{R}}_i\right\| &\le \left\|\mathbf{e}_{\mathbf{B}}^{\top}\mathbf{R}_{f_i}\right\| + \left\|\mathbf{e}_{\mathbf{B}}^{\top}\mathbf{R}_i^{*}\right\| \le \sigma\sqrt{\lambda}\cdot m\cdot(m^{O(\log n)} + \sigma_0\sqrt{\lambda}) \le 2m\lambda\sigma\sigma_0, \\
\left\|\mathbf{e}_{\widetilde{g}}\right\| &\le \ell^{O(d)}\sigma\sqrt{\lambda} \le \sigma_{\mathbf{e}}, \quad ,\left\|\mathbf{e}_i'\right\| \le \ell n\sigma\sqrt{\lambda}, \quad \|\mathbf{e}_i\| \le \sigma_{\mathbf{e}}, \\
|\eta_i| &\le 1 \text{ since } \left\|\mathbf{e}_{\mathbf{s},i}^{\top}\mathbf{G}^{-1}(\mathbf{A}) - \mathbf{e}_i^{\top}\right\| \le \Delta + m\sigma\sqrt{\lambda} + \sigma_{\mathbf{e}}\sqrt{\lambda} \le \Delta
\end{aligned}$$

The norm bound $\left\|\mathbf{R}_{f_i}\right\| = m^{O(\log n)}$ follows from the observation that the function f_i can be computed by a circuit of depth $O(\log(n\log q)) = O(\log n)$. Therefore, the decoding error of $\tilde{\mathbf{y}}$ has norm bound

$$\left\|\left\lfloor\frac{\tilde{\mathbf{y}}}{p}\right\rceil - \left\lfloor\frac{\widetilde{g}(\mathbf{x})}{p}\right\rceil\right\| = \left\|\left\lfloor\frac{\tilde{\mathbf{y}} - \widetilde{g}(\mathbf{x})}{p}\right\rceil\right\| \le \left\lfloor\frac{5\sigma_{\mathbf{e}} + \Delta}{p}\right\rceil \le \left\lfloor\frac{1}{4}\right\rceil = 0,$$

where the first equation follows from the fact that $\widetilde{g}(\mathbf{x})$ is always a multiple of p. Therefore, $\left\lfloor\frac{\tilde{\mathbf{y}}}{p}\right\rceil = \left\lfloor\frac{\widetilde{g}(\mathbf{x})}{p}\right\rceil$, indicating that the decryption algorithm recovers $g(\mathbf{x})$ perfectly. Moreover, by the observation that

$$\left\|\mathbf{e}_{\mathbf{s},i}^{\top}\mathbf{G}^{-1}(\mathbf{A}) - \mathbf{e}_i^{\top}\right\| \le \Delta + m\sigma\sqrt{\lambda} + \sigma_{\mathbf{e}}\sqrt{\lambda} \le 2^{-\Omega(\lambda)}\Delta,$$

along with the fact that $\mathbf{r}^{\top}\mathbf{D}_i\mathbf{G}^{-1}(\mathbf{A})$ is marginally random, the rounding lemma guarantees that with probability $1 - 2^{-\Omega(\lambda)}$,

$$\left\lfloor\frac{\mathbf{r}^{\top}\mathbf{D}_i\mathbf{G}^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{s},i}^{\top}\mathbf{G}^{-1}(\mathbf{A}) - \mathbf{e}_i^{\top}}{\Delta}\right\rceil = \left\lfloor\frac{\mathbf{r}^{\top}\mathbf{D}_i\mathbf{G}^{-1}(\mathbf{A})}{\Delta}\right\rceil.$$

Therefore $\eta_i = 0$ with overwhelming probability. Hence, combining the decryption equation (13) with equation (15), we conclude that the equation (14) holds with overwhelming probability. □

Theorem 6 (Succinctness). *Construction 1 is succinct if there exists constants $\alpha, \beta, \gamma \in (0,1)$ such that $n \le L^{\alpha}\cdot\mathsf{poly}(\lambda, k, d)$, $\kappa \le L^{\beta}$, $\ell \le L^{\gamma}$ where $\beta + \gamma \ge 1$ and $\alpha + \beta < 1/2$, $m = \Theta(n\log q)$, $q = 2^{\Theta(\kappa)}p$, and parameters $(\log\Delta, \log p) = \mathsf{poly}(\lambda, K, d, \log L)$.*

Proof. The size of the opening is bounded by

$$|\rho_i| = n\log q = n(\log p + O(\kappa)) = L^{\alpha+\beta}\mathsf{poly}(\lambda, K, d),$$

where $\alpha + \beta < 1/2$. For the encoding, we can separate the components into Q-dependent and Q-independent terms. The Q independent terms have size bounded by

$$|\mathsf{hpk}|, |\mathsf{hct}|, |\mathbf{A}|, |\mathsf{dct}|, |\mathsf{dct}_{\mathbf{x}}| = \mathsf{poly}(n, m, K, \ell, \log q) = \mathsf{poly}(L) \cdot \mathsf{poly}(\lambda, K, d).$$

On the other hand, the Q-dependent terms have size bounded by

$$\begin{aligned} |\mathbf{D}| &= Qnm \log q \leq QL^{2\alpha+2\beta} \cdot \mathsf{poly}(\lambda, K, d), \\ |\{\mathbf{c}_{\mathbf{s},i}\}_{i \in Q}| &= Qm \log q \leq QL^{\alpha+2\beta} \cdot \mathsf{poly}(\lambda, K, d), \\ |\{\mathbf{c}_i\}_{i \in Q}| &= Q\ell \log \Delta \leq QL^{\gamma} \cdot \mathsf{poly}(\lambda, K, d). \end{aligned}$$

Therefore the overall size of the encoding is bounded by $(QL^{1-\epsilon} + \mathsf{poly}(L)) \cdot \mathsf{poly}(\lambda, K, d)$ for constant $\epsilon = 1 - \max(2(\alpha + \beta), \gamma)$. □

Theorem 7 (SIM-security). *Assuming $n = \mathsf{poly}(\lambda)$, $q < 2^n$, $m = \Omega(n \log q)$, $\sigma_0 \geq 2^{\lambda} m^{\Omega(\log n)} \sigma$, $\sigma_{\mathbf{e}} \geq 2^{\lambda} \sigma \sigma_0 \ell^{\Omega(\log n + d)}$, and $\Delta \geq 2^{\lambda} \ell^{\Omega(d)} \sigma \sigma_0$, where $m, \sigma_0, \sigma_{\mathbf{e}}, \Delta$ are sufficiently large, and assuming that the* CRO *assumption (assumption 1) is (sub-exponentially) secure for parameters (n, q, Δ, σ), then construction 1 is (sub-exponentially) SIM-secure.*

Proof. We consider the following sequence of hybrids. We inline intuition on why neighboring hybrids are indistinguishable and provide formal proofs after all hybrids are described.

- $\mathcal{H}_0$: This is the LHS of the SIM-security definition in definition 10. For message-function tuple $(\mathbf{x}, g_1, \ldots, g_n)$, the output distribution of this hybrid is described by

$$\left\{ (\mathsf{crs}, \mathsf{ct}, \{\rho_i\}_{i \in [Q]}) \;\middle|\; \begin{array}{rl} \mathsf{crs} & \leftarrow \mathsf{Gen}(1^{\lambda}, 1^{Q}, 1^{k}, 1^{L}, 1^{d}) \\ \mathsf{ct} & \leftarrow \mathsf{Enc}(\mathsf{crs}, \mathbf{x}; R) \\ \rho_i & \leftarrow \mathsf{Open}(\mathsf{crs}, g_i, i, \mathbf{x}, R) \end{array} \right\}$$

- $\mathcal{H}_1$: Same as $\mathcal{H}_0$, except that in all algorithms, the samples from the bounded discrete Gaussian $\overline{\mathcal{D}}$ are replaced with samples from the standard discrete Gaussian $\mathcal{D}$. Hybrid $\mathcal{H}_1$ is statistically close to $\mathcal{H}_0$ following the fact that Gaussian distributions have small tails (Lemma 2).

- $\mathcal{H}_2$: Same as $\mathcal{H}_1$, but in the encryption algorithm, the randomness $\mathbf{s}_i, \mathbf{e}_i$ for the connecting components are computed differently. The hybrid first samples

$$\tilde{\mathbf{s}}_i \leftarrow \mathbb{Z}_q^n, \quad \tilde{\mathbf{e}}_i \leftarrow \mathcal{D}_{\sigma_{\mathbf{e}}}^{\ell},$$

then programs

$$\mathbf{s}_i = \tilde{\mathbf{s}}_i - \mathbf{u}_i - \mathbf{w}_{\widetilde{g}_i}, \quad \mathbf{e}_i^{\top} = \tilde{\mathbf{e}}_i - \mathbf{e}_{\mathbf{B}}^{\top} \widetilde{\mathbf{R}}_i - \mathbf{e}_i'^{\top} - \mathbf{e}_{\widetilde{g}_i}^{\top}.$$

Note that the opening $\rho_i = \tilde{\mathbf{s}}_i$ for all $i \in [Q]$. Hybrid $\mathcal{H}_2$ is statistically close to $\mathcal{H}_1$ because the marginal distribution of $\tilde{\mathbf{s}}_i, \tilde{\mathbf{e}}_i$ in $\mathcal{H}_1$ is uniformly random over $\mathbb{Z}_q$ and statistically close to i.i.d. Gaussian $\mathcal{D}_{\sigma_{\mathbf{e}}}$. Therefore, it is statistically close to sample them first, and then reverse compute $\mathbf{s}_i$ and $\mathbf{e}_i$. See Lemma 8.

- $\mathcal{H}_3$: Same as $\mathcal{H}_2$, but in the encryption algorithm, the connecting components are computed differently. The hybrid sets

$$\mathbf{c}_i^{\top} = (\tilde{\mathbf{s}}_i^{\top} \mathbf{A} + \tilde{\mathbf{e}}_i^{\top} - \mathsf{dct}_{f_i} - \underline{\mathsf{hct}_i'} - \mathsf{dct}_{\widetilde{g}_i} + \widetilde{g}(\mathbf{x})^{\top} \bmod \Delta).$$

For elements $\mathbf{c}_{\mathbf{s},i}$, the hybrid first computes intermediate elements

$$\widehat{\mathbf{c}}_i^{\intercal} = \mathbf{r}^{\intercal}\mathbf{D}_i + \mathbf{e}_{\mathbf{s},i}^{\intercal} - \mathbf{u}_i^{\intercal}\mathbf{G} = \mathbf{r}^{\intercal}\mathbf{D}_i + \mathbf{e}_{\mathbf{s},i}^{\intercal} - \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}_i'})) \cdot \mathbf{U}^{\intercal}\mathbf{G}$$

then set $\mathbf{c}_{\mathbf{s},i}^{\intercal} = \widehat{\mathbf{c}}_i^{\intercal} + \tilde{\mathbf{s}}_i^{\intercal}\mathbf{G} - \mathbf{w}_{\widetilde{g}_i}\mathbf{G}$. Note that $\mathbf{w}_{\widetilde{g}_i}$ is efficiently computable given secret $\mathbf{W}$, ciphertext, $\mathsf{dct}_{\mathbf{x}}$, function g_i, and message $\mathbf{x}$.

Hybrid $\mathcal{H}_3$ and $\mathcal{H}_2$ are identically distributed. The emulation of $\mathbf{c}_i^{\intercal}$ is perfect following from the correctness condition Equation (14) in Theorem 5. The emulation of $\mathbf{c}_{\mathbf{s},i}$ is also perfect because $\mathbf{c}_{\mathbf{s},i}$ is supposed to encode $\mathbf{s}_i\mathbf{G}$ which by the programming step in $\mathcal{H}_2$ equals to $(\tilde{\mathbf{s}}_i - \mathbf{u}_i - \mathbf{w}_{\widetilde{g}_i})\mathbf{G}$.

- $\mathcal{H}_4$: Same as $\mathcal{H}_3$, but in the generation of crs, the hybrid samples $\mathbf{R}^*$ indirectly. Namely, the algorithm first sample $\mathbf{R}_0 = (\mathbf{R}_{0,1}, \ldots, \mathbf{R}_{0,Q}) \leftarrow \mathcal{D}_{\sigma_0}^{m\times Q\ell}$, compute ciphertexts $\mathsf{hct}_{0,i} = \overline{\mathbf{B}}\mathbf{R}_{0,i}$ for all $i \in [Q]$, then reverse sample $\mathbf{R}^*$ from the conditional distribution

  $$\mathbf{R}_i^* \leftarrow \mathcal{D}_{\sigma_0}^{m\times\ell}|_{\overline{\mathbf{B}}\mathbf{R}_i^*=\mathsf{hct}_{0,i}}.$$

  Note that in this hybrid, the GSW rerandomization can be written by

  $$\mathsf{hct}_i' = \mathsf{hct}_{0,i} - \mathsf{hct}_{f_i}$$

  therefore one can equivalently write the reverse sample of $\mathbf{R}^*$ by

  $$\mathbf{R}_i^* \leftarrow \mathcal{D}_{\sigma_0}^{m\times\ell}|_{\overline{\mathbf{B}}\mathbf{R}_i^*=\mathsf{hct}_i'+\mathsf{hct}_{f_i}}$$

  Hybrid $\mathcal{H}_4$ is identically distributed to $\mathcal{H}_3$ since it only changes the order of sampling to hct_0 first and $\mathbf{R}^*$ second, and replaces hct_0 using equality.

- $\mathcal{H}_5$: Same as $\mathcal{H}_4$, but rewrites the GSW rerandomization equation by

  $$\mathsf{hct}_i' = \mathsf{hct}_{0,i} \boxminus f_i(\mathbf{r})$$

  Notice that this implies $\overline{\mathsf{hct}_i'} = \overline{\mathsf{hct}_{0,i}}$. The sampling of $\mathbf{R}^*$ is subsequently replaced by

  $$\mathbf{R}_i^* \leftarrow \mathcal{D}_{\sigma_0}^{m\times\ell}|_{\overline{\mathbf{B}}\mathbf{R}_i^*=\mathsf{hct}_{f_i}+\mathsf{hct}_{0,i}\boxminus f_i(\mathbf{r})}$$

  Hybrid $\mathcal{H}_5$ is statistically close to $\mathcal{H}_4$ because by the re-randomizability of GSW, $\mathsf{hct}_{0,i} \boxminus f_i(\mathbf{r})$ is statistically close to $\mathsf{hct}_{0,i} - \mathsf{hct}_{f_i}$. The former is a fresh encryption of $f_i(\mathbf{r})$ using wide Gaussian $\mathcal{D}_{\sigma_0}$ and the latter is a re-randomized encryption of $f_i(\mathbf{r})$.

- $\mathcal{H}_6$: Define function $\widetilde{f}_i$ as

  $$\widetilde{f}_i(\mathsf{hct}, \mathsf{hct}_{0,i}, \mathbf{A}, \mathbf{D}_i, \mathsf{dct}, \widehat{\mathbf{c}}_i)^{\intercal} = \Delta \left\lfloor \frac{\widehat{\mathbf{c}}_i^{\intercal}\mathbf{G}^{-1}(\mathbf{A}) + \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}_{0,i}}))^{\intercal} \cdot \mathsf{dct} + \underline{\mathsf{hct}_{0,i}}}{\Delta} \right\rceil,$$

  and replace $f_i(\mathbf{r})$ by $\widetilde{f}_i(\mathsf{hct}, \mathsf{hct}_{0,i}, \mathbf{A}, \mathbf{D}_i, \mathsf{dct}, \widehat{\mathbf{c}}_i)$ in the sampling equation of $\mathbf{R}_i^*$, namely,

  $$\mathbf{R}_i^* \leftarrow \mathcal{D}_{\sigma_0}^{m\times\ell}|_{\overline{\mathbf{B}}\mathbf{R}_i^*=\mathsf{hct}_{f_i}+\mathsf{hct}_{0,i}\boxminus\widetilde{f}_i(\mathsf{hct},\mathsf{hct}_{0,i},\mathbf{A},\mathbf{D}_i,\mathsf{dct},\mathbf{c}_{\mathbf{u},i})}.$$

  All other variables are sampled as in $\mathcal{H}_5$.

Hybrid $\mathcal{H}_6$ is statistically close to $\mathcal{H}_5$ because $f_i(\mathbf{r}) = \widetilde{f}_i(\mathsf{hct}, \mathsf{hct}_{0,i}, \mathbf{A}, \mathbf{D}_i, \mathsf{dct}, \widehat{\mathbf{c}}_i)$ with overwhelming probability, which follows from arguments similar to claim 1. See Lemma 12.

Observe that in $\mathcal{H}_6$, the sampling of $(\mathsf{hpk}, \mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{dct}, \widehat{\mathbf{c}})$ matches that of $(\mathsf{hpk}, \mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)$ in the real distribution $\mathcal{D}_0$ of the CRO assumption, Assumption 1. In particular, $\mathsf{dct} = \mathsf{ct}_1$ and $\widehat{\mathbf{c}}$ encrypts the same circular message as ct_2, $\mathbf{u}_i = f^{\mathrm{circ}}(\mathbf{U}, \mathsf{hct}_0) = -\mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_i)) \cdot \mathbf{U}^\intercal \mathbf{G}$, given $\overline{\mathsf{hct}}_i = \overline{\mathsf{hct}}'_i$. The function $f_i(\mathbf{r})$ is the same as described in the CRO assumption, and the above function $\widetilde{f}_i$ computes f_i using public encodings, corresponding to the safety constraint in the CRO assumption.

- $\mathcal{H}_7$: This hybrid samples the following components as random.

  $$(\mathsf{hpk}, \mathsf{hct}, \mathsf{dct}, \{\mathsf{hct}_{0,i}, \widehat{\mathbf{c}}_i\}_{i \in [Q]}) \leftarrow \mathbb{Z}_q^{(n+1)\times m} \times \mathbb{Z}_q^{(n+1)\times n(n+1)\lceil \log q \rceil^2} \times \mathbb{Z}_q^{\ell n \lceil \log q \rceil \times \ell} \times (\mathbb{Z}_q^{(n+1)\times \ell} \times \mathbb{Z}_q^{n\lceil \log q \rceil})^Q,$$

  and compute/sample the remaining elements as in $\mathcal{H}_6$.

  Hybrid $\mathcal{H}_7$ is computationally indistinguishable to $\mathcal{H}_6$ following the CRO assumption (Assumption 1) w.r.t. parameters (n, q, Δ, σ). As sketched in Hybrid $\mathcal{H}_6$ and as detailed in Lemma 13, the real and ideal distribution of CRO matches exactly $\mathcal{H}_6$ and $\mathcal{H}_7$.

- $\mathcal{H}_8$: Same as $\mathcal{H}_7$, except that the GSW public key is sampled with trapdoor (lemma 5), namely,

  $$(\mathsf{hpk} = \overline{\mathbf{B}}, \mathbf{T}) \leftarrow \mathsf{TrapGen}(1^{n+1}, 1^m).$$

  Hybrid $\mathcal{H}_8$ and $\mathcal{H}_7$ are statistically close by properties of lattice trapdoor sampling.

- $\mathcal{H}_9$: Same as $\mathcal{H}_8$, except that the the conditional sampling of $\mathbf{R}^*$ is replaced by the efficient procedure

  $$\mathbf{R}_i^* \leftarrow \mathsf{SampPre}(\mathbf{A}, \mathbf{T}, (\mathsf{hct}_{f_i} + \mathsf{hct}_{0,i} \boxminus \widetilde{f}_i(\mathsf{hct}, \mathsf{hct}_{0,i}, \mathbf{A}, \mathbf{D}_i, \mathsf{dct}, \widehat{\mathbf{c}}_i)), \sigma_0)$$

  Hybrid $\mathcal{H}_8$ and $\mathcal{H}_7$ are statistically close by properties of preimage sampling using lattice trapdoors.

- $\mathcal{H}_{10}$: Same as $\mathcal{H}_9$, except now $\mathbf{c}_{\mathbf{s},i} \leftarrow \mathbb{Z}_q^m$ are sampled at random, while $\widehat{\mathbf{c}}_i^\intercal = \mathbf{c}_{\mathbf{s},i}^\intercal - \tilde{\mathbf{s}}_i^\intercal \mathbf{G} + \mathbf{w}_{\widetilde{g}_i}^\intercal \mathbf{G}$.

  Hybrid $\mathcal{H}_{10}$ is identically distributed as $\mathcal{H}_9$, since in the latter $\widehat{\mathbf{c}}_i \leftarrow \mathbb{Z}_q^m$ and $\mathbf{c}_{\mathbf{s},i}^\intercal = \widehat{\mathbf{c}}_i + \tilde{\mathbf{s}}_i^\intercal \mathbf{G} - \mathbf{w}_{\widetilde{g}_i}^\intercal \mathbf{G}$.

- $\mathcal{H}_{11}$: Define function $\widetilde{f}'_i$ as

  $$\begin{aligned} &\widetilde{f}'_i(\mathsf{hct}, \mathsf{hct}_{0,i}, \mathbf{A}, \mathbf{D}_i, \mathsf{dct}, \mathsf{dct}_{\mathbf{x}}, \mathbf{c}_{\mathbf{s},i}, \tilde{\mathbf{s}}_i, \widetilde{g}_i(\mathbf{x}))^\intercal \\ =&\Delta \left\lfloor \frac{\mathbf{c}_{\mathbf{s},i}^\intercal \mathbf{G}^{-1}(\mathbf{A}) - \tilde{\mathbf{s}}_i^\intercal \mathbf{A} + \mathsf{dct}_{\widetilde{g}_i}^\intercal - \widetilde{g}_i(\mathbf{x})^\intercal + \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\intercal \cdot \mathsf{dct} + \underline{\mathsf{hct}}_{0,i}}{\Delta} \right\rceil, \end{aligned}$$

  and replace $\widetilde{f}_i(\mathsf{hct}, \mathsf{hct}_{0,i}, \mathbf{A}, \mathbf{D}_i, \mathsf{dct}, \mathsf{dct}_{\mathbf{x}}, \widehat{\mathbf{c}}_i)$ by $\widetilde{f}'_i(\mathsf{hct}, \mathsf{hct}_{0,i}, \mathbf{A}, \mathbf{D}_i, \mathsf{dct}, \mathbf{c}_{\mathbf{s},i}, \tilde{\mathbf{s}}_i, \widetilde{g}_i(\mathbf{x}))$ in the preimage sampling equation, namely,

  $$\mathbf{R}_i^* \leftarrow \mathsf{SampPre}(\mathbf{A}, \mathbf{T}, (\mathsf{hct}_{f_i} + \mathsf{hct}_{0,i} \boxminus \widetilde{f}'_i(\mathsf{hct}, \mathsf{hct}_{0,i}, \mathbf{A}, \mathbf{D}_i, \mathsf{dct}, \mathsf{dct}_{\mathbf{x}}, \mathbf{c}_{\mathbf{s},i}, \tilde{\mathbf{s}}_i, \widetilde{g}_i(\mathbf{x}))), \sigma_0)$$

  All other elements are sampled the same as $\mathcal{H}_{10}$. Hybrid $\mathcal{H}_{11}$ is statistically close to $\mathcal{H}_{10}$, following from the fact that $\widehat{\mathbf{c}}_i^\intercal \mathbf{G}^{-1}(\mathbf{A}) = \mathbf{c}_{\mathbf{s},i}^\intercal \mathbf{G}^{-1}(\mathbf{A}) - \tilde{\mathbf{s}}_i^\intercal \mathbf{A} + \mathsf{dct}_{\widetilde{g}_i}^\intercal - \widetilde{g}_i(\mathbf{x})^\intercal - \mathbf{e}_{\widetilde{g}_i}^\intercal$, and the last error term does not affect rounding result with overwhelming probability.

Note that in this hybrid, the intermediate terms $\widehat{\mathbf{c}}_i$ are no longer needed. Therefore, the hybrid does not need to compute the secret $\mathbf{w}_{\widetilde{g}_i}$ and the ciphertext $\mathsf{dct}_{\mathbf{x}}$ is the only term directly dependent on the secret $\mathbf{W}$.

- $\mathcal{H}_{12}$: Same as $\mathcal{H}_{11}$, except now $\mathsf{dct}_{\mathbf{x}}$ is sampled at random. Hybrid $\mathcal{H}_{12}$ is computationally indistinguishable to $\mathcal{H}_{11}$ following the LWE assumption, since $\mathsf{dct}_{\mathbf{x}}$ is a fresh dual-GSW encryption of $\mathbf{x}$ using secret $\mathbf{W}$ that is not used for the rest of sampling of $\mathcal{H}_{11}$.

  $\mathcal{H}_{12}$ is the RHS of the SIM-security definition in definition 10, where every term is efficiently simulatable given g_i and $g_i(\mathbf{x})$. For full exposure, we describe the simulator as follows.

  $\mathsf{Sim}(1^\lambda, \{g_i, g_i(\mathbf{x})\}_{i\in[Q]})$:

  - Set parameters pp as in $\mathsf{Gen}(1^\lambda, 1^Q, 1^K, 1^L, 1^d)$.
  - Sample public key $(\mathsf{hpk} = \overline{\mathbf{B}}, \mathbf{T}) \leftarrow \mathsf{TrapGen}(1^{n+1}, 1^m)$.
  - Sample elements $(\tilde{\mathbf{s}}_i, \tilde{\mathbf{e}}_i, \mathsf{hct}, \mathsf{hct}_{0,i}, \mathsf{dct}, \mathsf{dct}_{\mathbf{x}}, \mathbf{c}_{\mathbf{s},i}) \leftarrow \$$.
  - Compute $\mathbf{c}_i^\top = (\tilde{\mathbf{s}}_i^\top + \tilde{\mathbf{e}}_i^\top - \mathsf{dct}_{f_i} - \underline{\mathsf{hct}}_i^* - \mathsf{dct}_{\widetilde{g}_i} + \widetilde{g}_i(\mathbf{x})^\top \bmod \Delta)$.
  - Sample $\mathbf{R}_i^* \leftarrow \mathsf{SampPre}(\mathbf{A}, \mathbf{T}, (\mathsf{hct}_{f_i} + \mathsf{hct}_{0,i} \boxminus \widetilde{f}_i'(\mathsf{hct}, \mathsf{hct}_{0,i}, \mathbf{A}, \mathbf{D}_i, \mathsf{dct}, \mathsf{dct}_{\mathbf{x}}, \mathbf{c}_{\mathbf{s},i}, \widetilde{g}_i(\mathbf{x}))), \sigma_0)$.
  - Output $\mathsf{crs} = (\mathsf{pp}, \mathbf{R}^*)$, $\mathsf{ct} = (\mathsf{hpk}, \mathsf{hct}, \mathbf{D}, \{\mathbf{c}_i, \mathbf{c}_{\mathbf{s}_i}\}_{i\in[Q]}, \mathbf{A}, \mathsf{dct}, \mathsf{dct}_{\mathbf{x}})$, and $\rho_i = \tilde{\mathbf{s}}_i$ for all $i \in [Q]$.

Next, we formally argue the indistinguishability between neighboring hybrids.

Lemma 7. *The statistical distance between $\mathcal{H}_0$ and $\mathcal{H}_1$ is bounded by $2^{-\Omega(\lambda)}$.*

Proof. This follows directly from Lemma 2, which states that each sample from $\overline{\mathcal{D}}$ is $2^{-\lambda}$-close to sample $\mathcal{D}$. □

Lemma 8. *If $\sigma_0 \geq m^{O(\log n)}$ and $\sigma_{\mathbf{e}} = 2^\lambda \cdot \sigma\sigma_0\ell^{\Omega(\log n + d)}$, the statistical distance between $\mathcal{H}_1$ and $\mathcal{H}_2$ is bounded by $2^{-\Omega(\lambda)}$.*

Proof. It is easy to observe that the distribution of $\mathbf{s}_i$ is uniformly random in both hybrids. For the distribution of $\mathbf{e}_i$, observe that with probability $2^{-\Omega(\lambda)}$,

$$\left\|\mathbf{e}_{\mathbf{B}}^\top \widetilde{\mathbf{R}}_i - \mathbf{e}_i'^\top - \mathbf{e}_{\widetilde{g}_i}^\top\right\| \leq \sigma\sqrt{\lambda} \cdot m \cdot (m^{O(\log n)} + \sigma_0\sqrt{\lambda}) + \ell n \sigma\sqrt{\lambda} + \ell^{O(d)}\sigma\sqrt{\lambda} \leq \mathsf{poly}(\lambda)\sigma\sigma_0\ell^{O(\log n + d)},$$

where the analysis is identical to the proof of Theorem 5. Thus, following the smudging lemma (lemma 3), the statistical distance between $\mathbf{e}_i \leftarrow \mathcal{D}_{\sigma_{\mathbf{e}}}^\ell$ and $\mathbf{e}_i \leftarrow \mathcal{D}_{\sigma_{\mathbf{e}}}^\ell + (\mathbf{e}_{\mathbf{B}}^\top \widetilde{\mathbf{R}}_i - \mathbf{e}_i'^\top - \mathbf{e}_{\widetilde{g}_i}^\top)^\top$ is bounded by $2^{-\Omega(\lambda)}$, which completes the proof. □

Lemma 9. *$\mathcal{H}_2$ is identical to $\mathcal{H}_3$.*

Proof. In $\mathcal{H}_3$, $\mathbf{c}_i$ is computed by

$$\begin{aligned}
\mathbf{c}_i^\top &= (\tilde{\mathbf{s}}_i^\top \mathbf{A} + \tilde{\mathbf{e}}_i^\top - \mathsf{dct}_{f_i} - \underline{\mathsf{hct}}_i' - \mathsf{dct}_{\widetilde{g}_i} + \widetilde{g}_i(\mathbf{x})^\top \bmod \Delta) \\
&= \left(\begin{array}{rr} (\mathbf{s}_i + \mathbf{u}_i + \mathbf{w}_{\widetilde{g}_i})^\top \mathbf{A} + (\mathbf{e}_i^\top + \mathbf{e}_{\mathbf{B}}^\top \widetilde{\mathbf{R}}_i + \mathbf{e}_i'^\top + \mathbf{e}_{\widetilde{g}_i}^\top) - (\mathbf{u}_i^\top \mathbf{A} + \mathbf{e}_i'^\top - \mathbf{r}^\top(\mathbf{B}\widetilde{\mathbf{R}}_i)) & \\ -((\mathbf{r}^\top \mathbf{B} + \mathbf{e}_{\mathbf{B}}^\top)\widetilde{\mathbf{R}}_i - f_i(\mathbf{r})) - (\mathbf{w}_{\widetilde{g}_i}^\top \mathbf{A} + \mathbf{e}_{\widetilde{g}_i}^\top + \widetilde{g}_i(\mathbf{x})^\top) + \widetilde{g}_i(\mathbf{x})^\top & \bmod \Delta \end{array}\right) \\
&= (\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top + f_i(\mathbf{r})^\top \bmod \Delta) = (\mathbf{s}_i^\top \mathbf{A} + \mathbf{e}_i^\top \bmod \Delta),
\end{aligned}$$

which is identical to $\mathcal{H}_2$. Notice that the last equation follows from the observation that $\Delta | f_i(\mathbf{r})$.

We note that one can similarly obtain the relation through the correctness equation (14) described in theorem 5. Assuming the required conditions for theorem 5 holds, by reordering the variables in the correctness equation (14) we get

$$\mathbf{c}_i^\top = \rho_i \mathbf{A} + \tilde{\mathbf{e}}_i^\top - \widetilde{g}(\mathbf{x})^\top - \underline{\mathsf{hct}}'_i - \mathsf{dct}_i - \mathsf{dct}_{\widetilde{g}} - \Delta \left\lfloor \frac{\mathbf{c}_{\mathbf{s},i}^\top \mathbf{G}^{-1}(\mathbf{A}) - \mathbf{c}_i^\top}{\Delta} \right\rceil .$$

By taking module Δ on both sides, this immediately gives

$$\mathbf{c}_i^\top = (\tilde{\mathbf{s}}_i^\top \mathbf{A} + \tilde{\mathbf{e}}_i^\top - \mathsf{dct}_{f_i} - \underline{\mathsf{hct}}'_i - \mathsf{dct}_{\widetilde{g}_i} + \widetilde{g}_i(\mathbf{x})^\top \bmod \Delta)$$

□

Lemma 10. *$\mathcal{H}_3$ is identical to $\mathcal{H}_4$.*

Proof. This follows from the simple observation that, for every distribution $\mathcal{D}$ and every deterministic function h,

$$\{x \leftarrow \mathcal{D}\} \equiv \left\{ x \leftarrow \mathcal{D}|_{h(x)=y} \;\middle|\; \begin{matrix} x_0 \leftarrow \mathcal{D} \\ y = h(x_0) \end{matrix} \right\}. \tag{16}$$

Therefore the distribution of $\mathbf{R}_i^*$ is identical in both hybrids. The remaining difference between the two hybrids is just a change of variables. □

Lemma 11. *If $q < 2^n$ and $\sigma_0 = 2^\lambda m^{\Omega(\log n)}$, the statistical distance between $\mathcal{H}_4$ and $\mathcal{H}_5$ is bounded by $2^{-\Omega(\lambda)}$.*

Proof. The only difference between the two hybrids is that in the computation of the rerandomized ciphertext hct'_i,

$$\mathcal{H}_4 \;:\; \mathsf{hct}'_i = \overline{\mathbf{B}}(\mathbf{R}_{0,i} - \mathbf{R}_{f_i}) \boxminus f_i(\mathbf{r}), \quad \mathcal{H}_5 \;:\; \mathsf{hct}'_i = \overline{\mathbf{B}}\mathbf{R}_{0,i} \boxminus f_i(\mathbf{r}),$$

where $\left\|\mathbf{R}_{f_i}\right\| \leq m^{O(\log n)}$ since f_i is computable in depth $O(\log(n \log q)) = O(\log n)$. Therefore, by the smudging lemma (lemma 3), the distribution of $\mathbf{R}_{0,i} \leftarrow \mathcal{D}_{\sigma_0}^{m \times \ell}$ is $2^{-\Omega(\lambda)}$-close to $\mathbf{R}_{0,i} \leftarrow \mathcal{D}_{\sigma_0}^{m \times \ell} + \mathbf{R}_{f_i}$, which completes the proof. □

Lemma 12. *If $\Delta \geq 2^\lambda \cdot \sigma \sigma_0$, the statistical distance between $\mathcal{H}_5$ and $\mathcal{H}_6$ is bounded by $2^{-\Omega(\lambda)}$.*

Proof. In $\mathcal{H}_6$, the output of $\widetilde{f}_i$ is

$$\begin{aligned}
&\widetilde{f}_i(\mathsf{hct}, \mathsf{hct}_{0,i}, \mathbf{A}, \mathbf{D}_i, \mathsf{dct}, \widehat{\mathbf{c}}_i)^\top \\
&= \Delta \left\lfloor \frac{\widehat{\mathbf{c}}_i^\top \mathbf{G}^{-1}(\mathbf{A}) + \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\top \cdot \mathsf{dct} + \underline{\mathsf{hct}}_{0,i}}{\Delta} \right\rceil \\
&= \Delta \left\lfloor \frac{(\mathbf{r}^\top \mathbf{D} \mathbf{G}^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{s},i}^\top \mathbf{G}^{-1}(\mathbf{A}) - \mathbf{u}_i^\top \mathbf{A}) + (\mathbf{u}_i^\top \mathbf{A} + \mathbf{e}_i'^\top - \mathbf{r}^\top(\mathbf{B}\mathbf{R}_{0,i})) + (\mathbf{r}^\top \mathbf{B}\mathbf{R}_{0,i} + \mathbf{e}_{\mathbf{B}}^\top \mathbf{R}_{0,i})}{\Delta} \right\rceil \\
&= \Delta \left\lfloor \frac{\mathbf{r}^\top \mathbf{D} \mathbf{G}^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{s},i}^\top + \mathbf{e}_i'^\top + \mathbf{e}_{\mathbf{B}}^\top \mathbf{R}_{0,i}}{\Delta} \right\rceil .
\end{aligned}$$

Notice that the term $\mathbf{r}^\top \mathbf{D}\mathbf{G}^{-1}(\mathbf{A})$ is marginally random given that $\mathbf{r}$, $\mathbf{A}$, $\mathbf{D}$ are all sampled uniform randomly, and that with probability $2^{-\Omega(\lambda)}$, the error vector has norm bounded by

$$\left\| \mathbf{e}_{\mathbf{s},i}^\top + \mathbf{e}_i'^\top + \mathbf{e}_{\mathbf{B}}^\top \mathbf{R}_{0,i} \right\| \leq \sigma\sqrt{\lambda} + \sigma\sqrt{\lambda} \cdot \ell n + \sigma\sqrt{\lambda} \cdot m \cdot \sigma_0\sqrt{\lambda} \leq \mathsf{poly}(\lambda) \cdot \sigma\sigma_0.$$

Therefore by lemma 4, we have

$$\widetilde{f}_i(\mathsf{hct}, \mathsf{hct}_{0,i}, \mathbf{A}, \mathbf{D}_i, \mathsf{dct}, \widehat{\mathbf{c}}_i)^\top = \Delta \left\lfloor \frac{\mathbf{r}^\top \mathbf{D}\mathbf{G}^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{s},i}^\top + \mathbf{e}_i'^\top + \mathbf{e}_{\mathbf{B}}^\top \mathbf{R}_{0,i}}{\Delta} \right\rceil = \Delta \left\lfloor \frac{\mathbf{r}^\top \mathbf{D}\mathbf{G}^{-1}(\mathbf{A})}{\Delta} \right\rceil = f_i(\mathbf{r})^\top$$

with probability at least $(\mathsf{poly}(\lambda) \cdot \sigma\sigma_0)/\Delta + 2^{-\Omega(\lambda)} = 2^{-\Omega(\lambda)}$, which completes the proof.

As a remark, the relation between f_i and $\widetilde{f}_i$ arises from the correctness equation (14), following similar arguments given in claim 1 in section 4. □

Lemma 13. *If $\Delta = 2^{\Omega(\lambda)} \cdot \sigma\sigma_0$, then $\mathcal{H}_6$ and $\mathcal{H}_7$ are ϵ-computationally indistinguishable for all polynomial sized adversaries assuming $\epsilon(\lambda)$ security for the* CRO *assumption with parameters (n, q, Δ, σ) (assumption 1).*

Proof. Let $(f^{\mathrm{circ}}, f^{\mathsf{CRO}}, \widetilde{f}^{\mathsf{CRO}})$ be the function tuple defined in assumption 1 with respect to LWE parameters (n, q, Δ, σ), for polynomial parameters (Q, ℓ) of the functional encoding. We start by arguing that one can perfectly simulate the distribution $\mathcal{H}_6$ with the real distribution $\mathcal{D}_0$ of the $(f^{\mathrm{circ}}, f_{\mathsf{CRO}}, \widetilde{f}_{\mathsf{CRO}})$-CRO assumption. For $(\mathsf{hpk}, \mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2, \mathbf{R}^*) \leftarrow \mathcal{D}_0$, observe that

- $\mathsf{hpk} = \overline{\mathbf{B}}$, $\mathsf{hct} = \overline{\mathbf{B}}\mathbf{R} + \mathsf{bits}(\mathbf{r})^\top \otimes \mathbf{G}$, $\mathsf{hct}_0 = \overline{\mathbf{B}}\mathbf{R}_0$ are computed identically as $\mathcal{H}_6$
- $\mathbf{A}$, $\mathbf{D}$ are sampled at random, identical to $\mathcal{H}_6$.
- $\mathsf{ct}_1 = \mathbf{U}^\top \mathbf{A} + \mathbf{E}_{\mathbf{A}} + \mathbf{I}_\ell \otimes \mathbf{G}_n^\top \mathbf{r}$, identical to the component dct in $\mathcal{H}_6$
- $\mathsf{ct}_{2,i} = \mathbf{r}^\top \mathbf{D}_i + \mathbf{e}_{\mathbf{D},i}^\top + f_i^{\mathrm{circ}}(\mathbf{U}, \mathsf{hct}_0) = \mathbf{r}^\top \mathbf{D}_i + \mathbf{e}_{\mathbf{D},i}^\top - \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i})) \cdot \mathbf{U}^\top \mathbf{G}$, which is identical to the computation of $\widehat{\mathbf{c}}_i$ in $\mathcal{H}_6$ since $\overline{\mathsf{hct}}_i' = \overline{\mathsf{hct}}_{0,i}$.
- The opening $\mathbf{R}^*$ is sampled from discrete Gaussian conditioning on $\mathbf{B}\mathbf{R}_i^* = \mathsf{hct}_{f_i} \boxplus \mathsf{hct}_{0,i} \boxminus \widetilde{f}_i^{\mathsf{CRO}}(\mathsf{enc})$. Note that the function $\widetilde{f}^{\mathsf{CRO}}$ is of the form

$$\widetilde{f}_i^{\mathsf{CRO}}(\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)^\top = \Delta \left\lfloor \frac{\mathsf{ct}_{2,i} \cdot \mathbf{G}^{-1}(\mathbf{A}) + \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\top \mathsf{ct}_1 + \underline{\mathsf{hct}}_{0,i}}{\Delta} \right\rceil,$$

  which is identical to the function $\widetilde{f}_i$ defined in $\mathcal{H}_6$ after establishing the equality $\mathsf{ct}_1 = \mathsf{dct}$ and $\mathsf{ct}_{2,i} = \widehat{\mathbf{c}}_i$. Therefore $\mathbf{R}^*$ sampled from the CRO real distribution distributes identically with $\mathbf{R}^*$ sampled $\mathcal{H}_6$

Furthermore, the remaining elements in $\mathcal{H}_6$, including $\mathbf{c}_i$, $\mathsf{dct}_{\mathbf{x}}$, and $\tilde{\mathbf{s}}$, can be computed/sampled given the samples listed in the above comparison. This directly gives a simulator of $\mathcal{H}_6$ given $\mathcal{D}_0$ from CRO.

Now, following the CRO assumption the samples given by the real distribution $\mathcal{D}_0$ is indistinguishable from the ideal distribution $\mathcal{D}_1$, where $(\mathsf{hpk}, \mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1 = \mathsf{dct}, \mathsf{ct}_2 = \{\widehat{\mathbf{c}}_i\}_{i \in [Q]})$ are all sampled at random, and $\mathbf{R}^*$ is sampled with respect to the random elements. This is exactly the output distribution of $\mathcal{H}_7$. Therefore, $\mathcal{H}_6$ and $\mathcal{H}_7$ are $\epsilon(\lambda)$-computationally indistinguishable assuming ϵ security for the CRO assumption. □

Lemma 14. *If $m \geq 3n \log q$, the statistical distance between $\mathcal{H}_7$ and $\mathcal{H}_8$ is bounded by $2^{-\Omega(\lambda)}$.*

Proof. This follows directly from the statistical randomness property of lattice trapdoors (lemma 5). □

Lemma 15. *If $m \geq 3n \log q$ and $\sigma_0 > m \log n$, the statistical distance between $\mathcal{H}_8$ and $\mathcal{H}_9$ is bounded by $2^{-\Omega(\lambda)}$.*

Proof. This follows directly from the preimage sampling property of lattice trapdoors (lemma 5). □

Lemma 16. *$\mathcal{H}_9$ is identical to $\mathcal{H}_{10}$.*

Proof. It is easy to observe that the distribution of $\widehat{\mathbf{c}}_i$ is uniformly random in both hybrids. □

Lemma 17. *If $\Delta = 2^\lambda \cdot \ell^{\Omega(d)}\sigma$, the statistical distance between $\mathcal{H}_{10}$ and $\mathcal{H}_{11}$ is bounded by $2^{-\Omega(\lambda)}$.*

Proof. In $\mathcal{H}_{11}$, the output of $\widetilde{f}'$ is

$$
\begin{aligned}
&\widetilde{f}_i'(\mathsf{hct}, \mathsf{hct}_{0,i}, \mathbf{A}, \mathbf{D}_i, \mathsf{dct}, \mathsf{dct}_{\mathbf{x}}, \mathbf{c}_{\mathbf{s},i}, \tilde{\mathbf{s}}_i, \widetilde{g}_i(\mathbf{x}))^\intercal \\
=&\Delta \left\lfloor \frac{\mathbf{c}_{\mathbf{s},i}^\intercal \mathbf{G}^{-1}(\mathbf{A}) - \tilde{\mathbf{s}}_i^\intercal \mathbf{A} + \mathsf{dct}_{\widetilde{g}_i} - \widetilde{g}_i(\mathbf{x})^\intercal + \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\intercal \cdot \mathsf{dct} + \underline{\mathsf{hct}}_{0,i}}{\Delta} \right\rceil \\
=&\Delta \left\lfloor \frac{(\widehat{\mathbf{c}}_i^\intercal \mathbf{G}^{-1}(\mathbf{A}) + \tilde{\mathbf{s}}_i^\intercal \mathbf{A} - \mathbf{w}_{\widetilde{g}_i}^\intercal \mathbf{A}) - \tilde{\mathbf{s}}_i^\intercal \mathbf{A} + (\mathbf{w}_{\widetilde{g}_i}^\intercal \mathbf{A} + \mathbf{e}_{\widetilde{g}_i}^\intercal + \widetilde{g}_i(\mathbf{x}))^\intercal - \widetilde{g}_i(\mathbf{x})^\intercal + \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i})) \cdot \mathsf{dct} + \underline{\mathsf{hct}}_{0,i}}{\Delta} \right\rceil \\
=&\Delta \left\lfloor \frac{(\widehat{\mathbf{c}}_i^\intercal \mathbf{G}^{-1}(\mathbf{A}) + \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\intercal \cdot \mathsf{dct} + \underline{\mathsf{hct}}_{0,i}) + \mathbf{e}_{\widetilde{g}_i}^\intercal}{\Delta} \right\rceil,
\end{aligned}
$$

where the major terms $\mathbf{t}^\intercal = \left(\widehat{\mathbf{c}}_i^\intercal \mathbf{G}^{-1}(\mathbf{A}) + \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\intercal \cdot \mathsf{dct} + \underline{\mathsf{hct}}_{0,i}\right)$ has uniformly random marginal distribution given that $\underline{\mathsf{hct}}_{0,i}$ is random and independent to the remaining terms, and the error term has norm bounded by $\left\|\mathbf{e}_{\widetilde{g}_i}\right\| \leq \ell^{O(d)}\sigma\sqrt{\lambda}$ with $1 - 2^{-\Omega(\lambda)}$ probability. Therefore, again by lemma 4, we have with probability $1 - 2^{-\Omega(\lambda)}$,

$$
\widetilde{f}_i'(\mathsf{hct}, \mathsf{hct}_{0,i}, \mathbf{A}, \mathbf{D}_i, \mathsf{dct}, \mathsf{dct}_{\mathbf{x}}, \mathbf{c}_{\mathbf{s},i}, \tilde{\mathbf{s}}_i, \widetilde{g}_i(\mathbf{x}))^\intercal = \Delta \left\lfloor \frac{\mathbf{t}^\intercal + \mathbf{e}_{\widetilde{g}_i}^\intercal}{\Delta} \right\rceil = \Delta \left\lfloor \frac{\mathbf{t}^\intercal}{\Delta} \right\rceil = \widetilde{f}_i(\mathsf{hct}, \mathsf{hct}_{0,i}, \mathbf{A}, \mathbf{D}_i, \mathsf{dct}, \widehat{\mathbf{c}}_i)^\intercal,
$$

which completes the proof. □

Lemma 18. *Assuming $\epsilon(\lambda)$ security for $\mathsf{LWE}_{n,q,\sigma}$ (Assumption 1), then $\mathcal{H}_{11}$ and $\mathcal{H}_{12}$ are $\epsilon(\lambda)$-computationally indistinguishable for all polynomial sized adversaries.*

Proof. Assuming $\epsilon(\lambda)$ security for $\mathsf{LWE}_{n,q,\sigma}$ we immediately have that the following two ensembles are $\epsilon(\lambda)$-computationally indistinguishable

$$
\left\{ \mathsf{dct}_{\mathbf{x}} = \mathbf{W}^\intercal \mathbf{A} + \mathbf{E}_{\mathbf{x}} + \mathbf{x} \otimes \mathbf{G}^\intercal \;\middle|\; \begin{array}{cl} \mathbf{A} & \leftarrow \mathbb{Z}_q^{n \times \ell} \\ \mathbf{W} & \leftarrow \mathbb{Z}_q^{n \times K\ell\lceil\log q\rceil} \\ \mathbf{E} & \leftarrow \mathcal{D}_\sigma^{K\ell\lceil\log q\rceil \times \ell} \end{array} \right\} \approx_c^\epsilon \left\{ \mathsf{dct}_{\mathbf{x}} = \mathbf{V} + \mathbf{x} \otimes \mathbf{G}^\intercal \;\middle|\; \mathbf{V} \leftarrow \mathbb{Z}_q^{K\ell\lceil\log q\rceil \times \ell} \right\}.
$$

The right distribution is identical to sampling $\mathsf{dct}_{\mathbf{x}}$ at random, matching the sampling procedure in $\mathcal{H}_{12}$. Therefore, $\mathcal{H}_{11}$ and $\mathcal{H}_{12}$ is also $\epsilon(\lambda)$-indistinguishable. □

Assuming (subexponential) security for CRO [9], $n = \mathsf{poly}(\lambda)$, $q < 2^n$, $m = \Omega(n \log q)$, $\sigma_{\mathbf{e}} \geq 2^{\lambda}\sigma\sigma_0\ell^{\Omega(\log n + d)}$, $\sigma_0 \geq 2^{\lambda} m^{\Omega(\log n)}\sigma$, and $\Delta \geq 2^{\lambda}\ell^{\Omega(d)}\sigma\sigma_0$, where $m, \sigma_0, \sigma_{\mathbf{e}}, \Delta$ are sufficiently large, then all conditions required by the above lemmas are satisfied. Therefore, combining lemmas 7 to 18, we immediately know that $\mathcal{H}_0$ and $\mathcal{H}_{12}$ are sub-exponentially indistinguishable, which completes the proof.

□

Parameter instantiation We now give a set of plausible parameters toward a sub-exponentially secure functional encoding scheme. In the following, we use the notation $\widetilde{O}(\cdot)$ for the asymptotic notation suppressing all terms of order $\mathsf{poly}\log(\lambda, k, L, d)$.

Let $\delta \in (0, 1)$ be the LWE modulus-to-noise ratio parameter. We set our variables as follow.

- $\kappa = L^{0.1\delta}$, $\ell = L^{1-0.1\delta}$,
- $n = \widetilde{O}(\lambda L^{0.1\delta} d)^{1/\delta}$, $m = O(n \cdot \mathsf{poly}\log n)$,
- $\sigma = \mathsf{poly}(\lambda)$,
- $\sigma_0 = 2^{\lambda} m^{O(\log n)}\sigma = 2^{O(\lambda)} n^{O(\log n)}$,
- $\sigma_{\mathbf{e}} = 2^{\lambda}\sigma\sigma_0\ell^{O(d+\log n)} \leq 2^{O(\lambda)} L^{O(d+\log n)}$,
- $\Delta = 2^{\lambda}\ell^{O(d)}\sigma\sigma_0 \leq 2^{O(\lambda)} L^{O(d+\log n)}$,
- $p = O(\lambda(\ell + m)^{O(d+\log n)}\sigma\sigma_{\mathbf{e}} + \Delta) \leq 2^{O(\lambda)}(n + L)^{O(d+\log n)}$ where $\Delta | p$.
- $q = 2^{\kappa} p = 2^{O(\lambda)}(n + L)^{O(d+\log n)} \cdot 2^{L^{0.1\delta}}$, where $\log q = O(\lambda) + (d + \log n)\log(n + L)L^{0.1\delta} = \widetilde{O}(\lambda L^{0.1\delta} d) \leq n^{\delta}$.

Under the parameter setting, construction construction 1 is correct, succinct, and subexponentially SIM-secure. Therefore, we immediately get the following theorem.

Theorem 8. *Assuming the subexponential* CRO *assumption (assumption 1), there exists a subexponentially secure functional encoding scheme for all polynomial-sized circuits.*

6 Oblivious LWE Sampling from IND-CRO Assumption

In this section, we formulate a weaker indistinguishability version of the CRO assumption (IND-CRO), and show that IND-CRO assumption implies an Oblivious LWE Sampler [WW21], which also implies functional encodings.

[9] Note that CRO readily implies LWE with the same parameters (n, q, σ).

6.1 The IND-CRO Assumption

The $\mathsf{IND\text{-}CRO}$ assumption shares the same structure as the CRO assumption, but instead of stating the pseudorandomness of the ciphertexts and LWE samples, the $\mathsf{IND\text{-}CRO}$ assumption states that the samples jointly hide a secret bit β. The left and right distributions $\mathcal{D}_\beta$ of IND-CRO both augment the real distributions in CRO with an additional LWE sample ct_0 w.r.t. public matrix $\mathbf{A}$ that uses independent secrets $\mathbf{W}$, and hides the secret bit β in the form $\beta\mathbf{G}_\ell^\intercal$. Additionally, the LWE sample ct_2 that used to hide just a circular function $f^{\mathrm{circ}}(\mathbf{U}, \mathsf{hct}_0)$ of the secret $\mathbf{U}$ and hct_0, also hides (the concatenation of) linear functions of $\mathbf{W}$ of form $-\mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\intercal\mathbf{W}^\intercal\mathbf{G}_n$ and functions of β of form $-\beta\hat{\mathbf{s}}_i^\intercal\mathbf{G}_n$, where $\mathbf{v}_i^\intercal = \hat{\mathbf{s}}_i^\intercal\mathbf{A} + \widehat{\mathbf{e}}_i^\intercal$ are fresh LWE samples w.r.t. public matrix $\mathbf{A}$.

We now formally define the $(f^{\mathrm{circ}}, f, \widetilde{f})$-$\mathsf{IND\text{-}CRO}$ assumptions.

Definition 11 ($(f^{\mathrm{circ}}, f, \widetilde{f})$-Indistinguishability Circular Security with Random Opening (CRO) Assumption)**.** *Let λ be the security parameter. Let $n, m, d, k', \ell, M, \sigma$ be integer parameters that are polynomial in λ, and q, σ_0 be (potentially superpolynomial) integer parameters where $m = \Omega(n \log q)$ and $\sigma_0 = 2^\lambda m^{\Omega(d)}$ are sufficiently large. Let $f \in \mathcal{F}_{d,M}$ be a bounded depth packed circuit (definition 5) which parses its input as bits and have depth bound d and output length M, where M w.l.o.g. is a multiple of $(n+1)\lceil \log q \rceil$ and f^{circ} and $\widetilde{f}$ be efficiently computable functions with domain/codomain implicitly defined in Figure 9.*

We say that the (subexponential) $(f^{\mathrm{circ}}, f, \widetilde{f})$-CRO assumption holds if $\mathcal{D}_0$ and $\mathcal{D}_1$ in Figure 9 are (sub-exponentially) indistinguishable to all polynomial time attackers.

$$\left\{\left(\mathsf{hpk}, \mathsf{enc} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \{\mathbf{v}_i\}_{i\in[k']}, \mathsf{ct}_0, \mathsf{ct}_1, \mathsf{ct}_2)\right) \,\middle|\, (\mathsf{hpk}, \mathsf{enc}, \mathsf{hint}) \leftarrow \mathcal{D}_0(\mathbf{v})\right\}_\lambda$$
$$\approx \left\{\left(\mathsf{hpk}, \mathsf{enc} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \{\mathbf{v}_i\}_{i\in[k']}, \mathsf{ct}_0, \mathsf{ct}_1, \mathsf{ct}_2)\right) \,\middle|\, (\mathsf{hpk}, \mathsf{enc}, \mathsf{hint}) \leftarrow \mathcal{D}_1(\mathbf{v})\right\}_\lambda$$

Similar to the case of the CRO assumption, to construct the oblivious LWE sampler, it suffices to assume the $\mathsf{IND\text{-}CRO}$ assumption for specific tuples of functions. By default, the $\mathsf{IND\text{-}CRO}$ assumption refers to this version.

Assumption 3 (Indistinguishability Circular Security with Random Opening (IND-CRO) Assumption)**.** *Let λ be the security parameter, and n, q, σ be LWE parameters dependent on λ, where $\sigma = \mathsf{poly}(\lambda)$, $q \leq 2^{n^\delta}$ for some constant $\delta \in (0,1)$, q is a multiple of Δ such that $q/\Delta \geq 2^\lambda$, and $\Delta \geq (2n \log q)^\lambda$. The (subexponential) IND-CRO assumption with parameters (n, q, σ, Δ) states that for an appropriate $m = \Theta(n \log q)$, $\sigma_0 = \Delta/2^{\Theta(\lambda)}$, and every efficiently computable polynomials $Q = Q(\lambda)$ and $\ell = \ell(\lambda)$, the (subexponential) $(f^{\mathrm{circ}}, f, \widetilde{f})$-IND-CRO assumption holds for the following function tuple, where (f^{circ}, f) are identical to assumption 1:*

$$\mathsf{hct}_0 = \begin{pmatrix} \overline{\mathsf{hct}}_{0,i} \in \mathbb{Z}_q^{n\times\ell} \\ \underline{\mathsf{hct}}_{0,i} \in \mathbb{Z}_q^{1\times\ell} \end{pmatrix}_{i\in[Q]} \qquad \mathbf{D} = \left(\mathbf{D}_i \in \mathbb{Z}_q^{n\times n\lceil \log q\rceil}\right)_{i\in[Q]}$$

$$\mathsf{ct}_2 = \left(\mathsf{ct}_{2,i} = \mathbf{r}^\intercal\mathbf{D}_i + \mathbf{e}_{\mathbf{D},i} + f_i^{\mathrm{circ}}(\mathbf{U}, \mathsf{hct}_{0,i}) - \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\intercal\mathbf{W}^\intercal\mathbf{G}_n - \beta\hat{\mathbf{s}}_i^\intercal\mathbf{G}_n\right)_{i\in[Q]}, \text{where } \mathbf{e}_\mathbf{D} = \{\mathbf{e}_{\mathbf{D},i}\}_{i\in[Q]}$$

$$f^{\mathrm{circ}} = \left(f_i^{\mathrm{circ}}\right)_{i\in[Q]} \qquad f = \left(f_i\right)_{i\in[Q]} \qquad \widetilde{f} = \left(\widetilde{f_i}\right)_{i\in[Q]}$$

$$f_i^{\mathrm{circ}}(\mathbf{U}, \mathsf{hct}_0) = -\mathsf{vec}(\mathbf{G}_n^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\intercal \cdot (\mathbf{U}^\intercal\mathbf{G}_n)$$

$$f_i(\mathbf{r}, \mathbf{A}, \mathbf{D})^\intercal = \Delta\left\lfloor \frac{\mathbf{r}^\intercal\mathbf{D}_i \cdot \mathbf{G}_n^{-1}(\mathbf{A})}{\Delta} \right\rceil$$

$$\widetilde{f_i}(\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \{\mathbf{v}_i\}_{i\in Q}, \mathsf{ct}_0, \mathsf{ct}_1, \mathsf{ct}_2)^\intercal = \Delta\left\lfloor \frac{\mathsf{ct}_{2,i} \cdot \mathbf{G}_n^{-1}(\mathbf{A}) + \mathsf{vec}(\mathbf{G}_n^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\intercal\mathsf{ct}_1 + \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\intercal\mathsf{ct}_0 + \underline{\mathsf{hct}}_{0,i}}{\Delta} \right\rceil$$

$(f^{\mathrm{circ}}, f, \widetilde{f})$-IND-CRO Assumption

Distribution $\mathcal{D}_\beta$

HE (GSW) Components:

- $\mathbf{r} \leftarrow \mathbb{Z}_q^n$.
- $\mathsf{hpk} = \overline{\mathbf{B}} = \begin{pmatrix} \mathbf{B} \\ \mathbf{r}^\intercal \mathbf{B} + \mathbf{e}^\intercal \end{pmatrix}$, $\mathbf{B} \leftarrow \mathbb{Z}_q^{n \times m}$, $\mathbf{e} \leftarrow \mathcal{D}_\sigma^m$.
- $\mathsf{hct} = \overline{\mathbf{B}}\mathbf{R} + \mathsf{bits}(\mathbf{r})^\intercal \otimes \mathbf{G}_{n+1}$, $\mathbf{R} \leftarrow \{0,1\}^{m \times n(n+1)\lceil \log q \rceil^2}$.
- $\mathsf{hct}_0 = \overline{\mathbf{B}}\mathbf{R}_0$, $\mathbf{R}_0 \leftarrow \mathcal{D}_{\sigma_0}^{m \times M}$.

LWE Components:

- $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times \ell}$, $\mathbf{D} \leftarrow \mathbb{Z}_q^{n \times k' n \lceil \log q \rceil}$.
- $\forall i \in [k']$, $\mathbf{v}_i = \hat{\mathbf{s}}_i \mathbf{A} + \widehat{\mathbf{e}}_i$, $\hat{\mathbf{s}}_i \leftarrow \mathbb{Z}_q^n$, $\widehat{\mathbf{e}}_i \leftarrow \mathcal{D}_\sigma^\ell$.
- $\mathsf{ct}_0 = \mathbf{W}^\intercal \mathbf{A} + \mathbf{E}_0 + \beta \mathbf{G}_\ell^\intercal$, $\mathbf{W} \leftarrow \mathbb{Z}_q^{n \times \ell \lceil \log q \rceil}$, $\mathbf{E}_0 \leftarrow \mathcal{D}_\sigma^{\ell \lceil \log q \rceil \times \ell}$.
- $\mathsf{ct}_1 = \mathbf{U}^\intercal \mathbf{A} + \mathbf{E}_\mathbf{A} + \mathbf{I}_\ell \otimes \mathbf{G}_n^\intercal \mathbf{r}$, $\mathbf{U} \leftarrow \mathbb{Z}_q^{n \times \ell n \lceil \log q \rceil}$, $\mathbf{E}_\mathbf{A} \leftarrow \mathcal{D}_\sigma^{\ell n \lceil \log q \rceil \times \ell}$.
- $\mathsf{ct}_2 = \mathbf{r}^\intercal \mathbf{D} + \mathbf{e}_\mathbf{D}^\intercal + f^{\mathrm{circ}}(\mathbf{U}, \mathsf{hct}_0)$ $- \left(\ldots \middle| \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\intercal \mathbf{W}^\intercal \mathbf{G}_n + \beta \hat{\mathbf{s}}_i^\intercal \mathbf{G}_n \middle| \ldots \right)$, $\mathbf{e}_\mathbf{D} \leftarrow \mathcal{D}_\sigma^{k' n \lceil \log q \rceil}$

$\mathsf{Open}(f, \widetilde{f}, (\mathsf{hpk}, \mathsf{enc} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \{\mathbf{v}_i\}_{i \in [k']}, \mathsf{ct}_0, \mathsf{ct}_1, \mathsf{ct}_2)))$: Functions $(f, \widetilde{f})$ satisfies the safety constraint (1) in both $\mathcal{D}_0$ and $\mathcal{D}_1$, i.e., with overwhelming probability over the sampling of $(\mathsf{hpk}, \mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \{\mathbf{v}_i\}_{i \in [k']}, \mathsf{ct}_0, \mathsf{ct}_1, \mathsf{ct}_2)$ according to either $\mathcal{D}_0$ or $\mathcal{D}_1$, it holds that $f(\mathbf{r}, \mathbf{A}, \mathbf{D}) = \widetilde{f}(\mathsf{enc})$.

1. $\mathsf{hct}_f = \mathsf{HE.Eval}(\mathsf{hct}, f_{\mathbf{A},\mathbf{D}}) = \overline{\mathbf{B}}\mathbf{R}_f \boxplus f(\mathbf{r}, \mathbf{A}, \mathbf{D})$, where function $f_{\mathbf{A},\mathbf{D}}(\cdot) = f(\cdot, \mathbf{A}, \mathbf{D})$.
2. $\mathsf{hct}_f^* = \mathsf{hct}_f \boxplus (-\widetilde{f}(\mathsf{enc})) \boxplus \mathsf{hct}_0 \approx_s \overline{\mathbf{B}}(\mathbf{R}_f + \mathbf{R}_0)$.
3. $\mathbf{R}^* \leftarrow \mathcal{D}_{\sigma_0}^{m \times M} \Big|_{\mathsf{hct}_f^* = \overline{\mathbf{B}}\mathbf{R}^*}$.

Output: $(\mathsf{hpk}, \mathsf{enc} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \{\mathbf{v}_i\}_{i \in [k']}, \mathsf{ct}_0, \mathsf{ct}_1, \mathsf{ct}_2), \mathbf{R}^*)$.

Figure 9: The difference between the CRO assumption and IND-CRO assumption are highlighted. $n, m, q, d, k', \ell, M, \sigma, \sigma_0$ are λ-dependent parameters where $n, m, d, k, k', \ell, M, \sigma$ are polynomials in λ, while q, σ_0 may be superpolynomial in λ satisfying $(n+1)\lceil \log q \rceil | M$, $m = \Omega(n \log q)$, and $\sigma_0 = 2^\lambda m^{\Omega(d)}$, where m, σ_0 are sufficiently large. Circuit $f \in \mathcal{F}_{d,M}$ is a bounded depth packed circuit (definition 5) with depth bound d and output length M. We assume that f parses its input as bits.

The corresponding IND-CRO *distribution $\mathcal{D}_0, \mathcal{D}_1$ in Figure 9 have parameters $(n, m = \Theta(n \log q), q, d, k' = Q, \ell, M = Q\ell, \sigma, \sigma_0)$.*

Similar to lemma 6, we can verify that $f, \widetilde{f}$ considered in the IND-CRO assumption indeed satisfies the safety constraint:

Lemma 19. *For all $\beta \in \{0,1\}$, the following safety constraint holds w.r.t. the distribution $\mathcal{D}_\beta$ and functions $(f^{\mathrm{circ}}, f, \widetilde{f})$ specified in Assumption 3.*

$$\textbf{safety constraint:} \quad \Pr[f(\mathbf{r}, \mathbf{A}, \mathbf{D}) = \widetilde{f}(\mathsf{enc})] \geq 1 - 2^{-\Omega(\lambda)}, \tag{17}$$

where the probability is taken over the sampling of $(\mathbf{r}, \mathbf{A}, \mathbf{D}, \mathsf{enc})$ *according to* $\mathcal{D}_\beta$.

Proof. For each $i \in [Q]$ we have

$$\widetilde{f}_i(\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \{\mathbf{v}_i\}_{i\in Q}, \mathsf{ct}_0, \mathsf{ct}_1, \mathsf{ct}_2)^\top = \Delta \left\lfloor \frac{\mathsf{ct}_{2,i} \cdot \mathbf{G}_n^{-1}(\mathbf{A}) + \mathsf{vec}(\mathbf{G}_n^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\top \mathsf{ct}_1 + \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\top \mathsf{ct}_0 + \underline{\mathsf{hct}}_{0,i}}{\Delta} \right\rceil$$

Using the same variable in the description of $\mathcal{D}_\beta$ and in Assumption 3, with the additional notation that $\mathbf{R}_0 = (\mathbf{R}_{0,i})_{i\in[Q]}$ such that $\mathsf{hct}_{0,i} = \overline{\mathbf{B}}\mathbf{R}_{0,i}$, we can expand each term in $\widetilde{f}_i$ by

$$\begin{aligned}
\mathsf{ct}_{2,i} \cdot \mathbf{G}^{-1}(\mathbf{A}) &= \big(\mathbf{r}^\top \mathbf{D}_i + \mathbf{e}_{\mathbf{D},i}^\top - \mathsf{vec}(\mathbf{G}_n^{-1}(-\mathbf{B}\mathbf{R}_{0,i}))^\top \mathbf{U}^\top \mathbf{G}_n - \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\top \mathbf{W}^\top \mathbf{G}_n - \beta \hat{\mathbf{s}}_i^\top \mathbf{G}_n\big) \cdot \mathbf{G}_n^{-1}(\mathbf{A}) \\
&= \mathbf{r}^\top \mathbf{D}_i \mathbf{G}_n^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{D},i}^\top \mathbf{G}_n^{-1}(\mathbf{A}) - \mathsf{vec}(\mathbf{G}_n^{-1}(-\mathbf{B}\mathbf{R}_{0,i}))^\top \mathbf{U}^\top \mathbf{A} - \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\top \mathbf{W}^\top \mathbf{A} - \beta \hat{\mathbf{s}}_i^\top \mathbf{A} \\
\mathsf{vec}(\mathbf{G}_n^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\top \mathsf{ct}_1 &= \mathsf{vec}(\mathbf{G}_n^{-1}(-\mathbf{B}\mathbf{R}_{0,i}))^\top \big(\mathbf{U}^\top \mathbf{A} + \mathbf{E}_\mathbf{A} + \mathbf{I}_\ell \otimes \mathbf{G}_n^\top \mathbf{r}\big) \\
&= \mathsf{vec}(\mathbf{G}_n^{-1}(-\mathbf{B}\mathbf{R}_{0,i}))^\top \mathbf{U}^\top \mathbf{A} + \mathsf{vec}(\mathbf{G}_n^{-1}(-\mathbf{B}\mathbf{R}_{0,i}))^\top \mathbf{E}_\mathbf{A} - \mathbf{r}^\top \mathbf{B}\mathbf{R}_{0,i} \\
\mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\top \mathsf{ct}_0 &= \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\top \big(\mathbf{W}^\top \mathbf{A} + \mathbf{E}_0 + \beta \mathbf{G}_\ell^\top\big) \\
&= \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\top \mathbf{W}^\top \mathbf{A} + \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\top \mathbf{E}_0 + \beta \mathbf{v}_i^\top \\
&= \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\top \mathbf{W}^\top \mathbf{A} + \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\top \mathbf{E}_0 + \beta \hat{\mathbf{s}}_i^\top \mathbf{A} + \beta \widehat{\mathbf{e}}_i^\top \\
\underline{\mathsf{hct}}_{0,i} &= (\mathbf{r}^\top \mathbf{B} + \mathbf{e}^\top)\mathbf{R}_{0,i} \\
&= \mathbf{r}^\top \mathbf{B}\mathbf{R}_{0,i} + \mathbf{e}^\top \mathbf{R}_{0,i}
\end{aligned}$$

Therefore,

$$\begin{aligned}
&\widetilde{f}_i(\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \{\mathbf{v}_i\}_{i\in Q}, \mathsf{ct}_0, \mathsf{ct}_1, \mathsf{ct}_2)^\top \\
&\qquad = \Delta \left\lfloor \frac{\mathbf{r}^\top \mathbf{D}_i \mathbf{G}_n^{-1}(\mathbf{A}) + \mathbf{e}_{\mathbf{D},i}^\top \mathbf{G}_n^{-1}(\mathbf{A}) + \mathsf{vec}(\mathbf{G}_n^{-1}(-\mathbf{B}\mathbf{R}_{0,i}))^\top \mathbf{E}_\mathbf{A} + \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\top \mathbf{E}_0 + \beta \widehat{\mathbf{e}}_i^\top + \mathbf{e}^\top \mathbf{R}_{0,i}}{\Delta} \right\rceil
\end{aligned}$$

Note that $\mathbf{r}, \mathbf{A}, \mathbf{D}$ are all sampled at random, therefore the marginal distribution of $\mathbf{r}^\top \mathbf{D}_i \mathbf{G}^{-1}(\mathbf{A})$ is random in $\mathbb{Z}_q^{1\times\ell}$. Furthermore, by the Gaussian tail bound (lemma 2), with probability $1 - 2^{-\Omega(\lambda)}$ the noise term would have norm bounded by

$$\begin{aligned}
&\left\| \mathbf{e}_{\mathbf{D},i}^\top \mathbf{G}_n^{-1}(\mathbf{A}) + \mathsf{vec}(\mathbf{G}_n^{-1}(-\mathbf{B}\mathbf{R}_{0,i}))^\top \mathbf{E}_\mathbf{A} + \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\top \mathbf{E}_0 + \beta \widehat{\mathbf{e}}_i^\top + \mathbf{e}^\top \mathbf{R}_{0,i} \right\| \\
&\qquad\qquad \le \sqrt{\lambda}\ell\sigma + 2\sqrt{\lambda}\ell n \lceil \log q \rceil \sigma + \sigma + \sqrt{\lambda}\sigma\sigma_0 = \mathsf{poly}(\lambda)\sigma_0.
\end{aligned}$$

Combined with the setup that $\sigma_0 = \Delta/2^{\Theta(\lambda)}$, we can apply the rounding lemma 4 to conclude that

$$\widetilde{f}_i(\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \{\mathbf{v}_i\}_{i\in Q}, \mathsf{ct}_0, \mathsf{ct}_1, \mathsf{ct}_2)^\top \overset{\text{w.h.p.}}{=} \Delta \left\lfloor \frac{\mathbf{r}^\top \mathbf{D}_i \mathbf{G}_n^{-1}(\mathbf{A})}{\Delta} \right\rceil = f_i(\mathbf{r}, \mathbf{A}, \mathbf{D})^\top$$

where the first equality holds with probability $1 - 2^{-\Omega(\lambda)}$. Finally, with a union bound over $i \in [Q]$, we conclude that the safety constraint holds with probability $1 - 2^{-\Omega(\lambda)}$. □

We show that the IND-CRO assumption is implied by the CRO assumption.

Lemma 20. *The (subexponential) IND-CRO assumption (Assumption 3) with parameters* (n, q, σ, Δ) *holds if the (subexponential) CRO assumption (Assumption 1) holds with the same parameters.*

Proof. Fix parameters (n, q, σ, Δ) and any polynomial Q, ℓ, it suffice to show that the $(f^{\mathrm{circ}}, f, \widetilde{f}_{\mathsf{IND\text{-}CRO}})$-$\mathsf{IND\text{-}CRO}$ assumption holds for the function tuple defined in assumption 3 is implied by the $(f^{\mathrm{circ}}, f, \widetilde{f}_{\mathsf{CRO}})$ for the function tuple defined in assumption 1. Note that (f^{circ}, f) are defined identically in both assumptions. Fixing the tuple $(f^{\mathrm{circ}}, f, \widetilde{f}_{\mathsf{IND\text{-}CRO}}, f^{\mathrm{circ}}_{\mathsf{CRO}})$, we start by showing that for all $\beta \in \{0, 1\}$, the IND-CRO distribution $\mathcal{D}_{\beta}^{\mathsf{IND\text{-}CRO}}$ can be statistically simulated given a sample from the real CRO distribution $\mathcal{D}_0^{\mathsf{CRO}}$. In particular, we can give the simulator as follows.

$\underline{\mathsf{Sim}_0(\beta)}$:

- Sample $\left(\mathsf{hpk}, \mathsf{enc}^{\mathsf{CRO}} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2^{\mathsf{CRO}}), \mathsf{hint} = \mathbf{R}^*\right) \leftarrow \mathcal{D}_0^{\mathsf{CRO}}$
- For all $i \in [Q]$, sample $\mathbf{v}_i = \hat{\mathbf{s}}_i^{\mathsf{T}} \mathbf{A} + \widehat{\mathbf{e}}_i^{\mathsf{T}}$, where $\hat{\mathbf{s}}_i \leftarrow \mathbb{Z}_q^n$, $\widehat{\mathbf{e}}_i \leftarrow \mathcal{D}_{\sigma}^{\ell}$.
- Compute $\mathsf{ct}_0 = \mathbf{W}^{\mathsf{T}} \mathbf{A} + \mathbf{E}_0 + \beta \mathbf{G}_{\ell}^{\mathsf{T}}$, where $\mathbf{W} \leftarrow \mathbb{Z}_q^{n \times \ell \lceil \log q \rceil}$, $\mathbf{E}_0 \leftarrow \mathcal{D}_{\sigma}^{\ell \lceil \log q \rceil \times \ell}$.
- For all $i \in [Q]$, compute $\mathsf{ct}_{2,i}^{\mathsf{IND\text{-}CRO}} = \mathsf{ct}_{2,i}^{\mathsf{CRO}} - \mathbf{G}_{\ell}^{-1}(\mathbf{v}_i)^{\mathsf{T}} \mathbf{W}^{\mathsf{T}} \mathbf{G}_n - \beta \hat{\mathbf{s}}_i^{\mathsf{T}} \mathbf{G}_n$.
- Output $\left(\mathsf{hpk}, \mathsf{enc}^{\mathsf{IND\text{-}CRO}} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \{\mathbf{v}_i\}_{i \in [Q]}, \mathsf{ct}_0, \mathsf{ct}_1, \mathsf{ct}_2^{\mathsf{IND\text{-}CRO}}), \mathsf{hint} = \mathbf{R}^*\right)$.

By inspection, it is easy to observe $\mathsf{hpk}, \mathsf{enc}^{\mathsf{IND\text{-}CRO}}$ is sampled identically to $\mathcal{D}_{\beta}^{\mathsf{IND\text{-}CRO}}$. For the hint $\mathbf{R}^*$, observe that it is sampled in $\mathcal{D}_0^{\mathsf{CRO}}$ following

$$\mathbf{R}^* \leftarrow \mathcal{D}_{\sigma_0}^{m \times M}\Big|_{\mathsf{hct}_f \boxplus (-\widetilde{f}^{\mathsf{CRO}}(\mathsf{enc}^{\mathsf{CRO}})) \boxplus \mathsf{hct}_0 = \overline{\mathbf{B}} \mathbf{R}^*}$$

The safety constraint for both CRO and IND-CRO (lemmas 6 and 19) guarantee that with probability $1 - 2^{\Omega(\lambda)}$ over the sampling randomness, it holds that

$$\widetilde{f}^{\mathsf{CRO}}(\mathsf{enc}^{\mathsf{CRO}}) = f(\mathbf{r}, \mathbf{A}, \mathbf{D}) = \widetilde{f}^{\mathsf{IND\text{-}CRO}}(\mathsf{enc}^{\mathsf{IND\text{-}CRO}}).$$

Therefore $\mathsf{Sim}_0(\beta)$ statistically simulates $\mathcal{D}_{\beta}^{\mathsf{IND\text{-}CRO}}$.

We define $\mathsf{Sim}_1(\beta)$ which is identical to $\mathsf{Sim}_0(\beta)$ except that it samples the CRO distribution $(\mathsf{hpk}, \mathsf{enc}^{\mathsf{CRO}}, \mathsf{hint})$ from the ideal distribution $\mathcal{D}_1^{\mathsf{CRO}}$. By the CRO assumption, $\mathsf{Sim}_0(\beta)$ and $\mathsf{Sim}_1(\beta)$ are computationally indistinguishable.

Next, we define the efficient simulator $\mathsf{Sim}_2(\beta)$, which samples the distribution $(\mathsf{hpk}, \mathsf{enc}^{\mathsf{CRO}}, \mathsf{hint})$ by

- $\mathsf{hpk} = \overline{\mathbf{B}}$ is sampled with trapdoor, $(\overline{\mathbf{B}}, \mathbf{T}) \leftarrow \mathsf{TrapGen}(1^{n+1}, q, m)$.
- $\mathsf{enc}^{\mathsf{CRO}} \leftarrow \$$, identical to $\mathcal{D}_1^{\mathsf{CRO}}$.
- $\mathsf{hint} = \mathbf{R}^*$ is computed efficiently, $\mathbf{R}^* = \mathsf{SampPre}(\overline{\mathbf{B}}, \mathbf{T}, \mathsf{hct}_f \boxplus \mathsf{hct}_0 \boxplus (-\widetilde{f}^{\mathsf{CRO}}(\mathsf{enc}^{\mathsf{CRO}})), \sigma_0)$.

By the properties of lattice trapdoor (lemma 5), $\mathsf{Sim}_1(\beta)$ and $\mathsf{Sim}_2(\beta)$ are $2^{-\Omega(n)}$-close.

Now, observe that the computation of $\widetilde{f}^{\mathsf{CRO}}$ can be altered through the following equation:

$$\begin{aligned}
&\widetilde{f}_i^{\mathsf{CRO}}(\mathsf{enc}^{\mathsf{CRO}})^\intercal \\
&= \Delta \left\lfloor \frac{\mathsf{ct}_{2,i}^{\mathsf{CRO}} \cdot \mathbf{G}^{-1}(\mathbf{A}) + \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\intercal \mathsf{ct}_1 + \underline{\mathsf{hct}}_{0,i}}{\Delta} \right\rceil \\
&= \Delta \left\lfloor \frac{\mathsf{ct}_{2,i}^{\mathsf{IND\text{-}CRO}} \cdot \mathbf{G}^{-1}(\mathbf{A}) + \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\intercal \mathbf{W}^\intercal \mathbf{A} + \beta \hat{\mathbf{s}}_i^\intercal \mathbf{A} + \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\intercal \mathsf{ct}_1 + \underline{\mathsf{hct}}_{0,i}}{\Delta} \right\rceil \\
&= \Delta \left\lfloor \frac{\mathsf{ct}_{2,i}^{\mathsf{IND\text{-}CRO}} \cdot \mathbf{G}^{-1}(\mathbf{A}) + \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\intercal \mathsf{ct}_0 - \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\intercal \mathbf{E}_0 - \beta \widehat{\mathbf{e}}_i^\intercal + \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\intercal \mathsf{ct}_1 + \underline{\mathsf{hct}}_{0,i}}{\Delta} \right\rceil \\
&\overset{\text{w.h.p.}}{=} \Delta \left\lfloor \frac{\mathsf{ct}_{2,i}^{\mathsf{IND\text{-}CRO}} \cdot \mathbf{G}^{-1}(\mathbf{A}) + \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\intercal \mathsf{ct}_0 + \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\intercal \mathsf{ct}_1 + \underline{\mathsf{hct}}_{0,i}}{\Delta} \right\rceil
\end{aligned}$$

The last equality holds with probability $1 - 2^{-\Omega(\lambda)}$ following lemma 4 since $\mathsf{ct}_{2,i}^{\mathsf{IND\text{-}CRO}} \cdot \mathbf{G}^{-1}(\mathbf{A})$ is marginally random and $\mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\intercal \mathbf{E}_0$ has norm bound $\sqrt{\lambda}\ell\lceil \log q \rceil \sigma \ll \Delta$ with probability $1 - 2^{-\Omega(\lambda)}$.

With the alternation, the hint $\mathbf{R}^*$ can be computed without depending on $\mathsf{ct}_2^{\mathsf{CRO}}$. Therefore, the β-dependent terms computed in Sim_3 has the following format

$$\left(\mathsf{ct}_0, \{\mathsf{ct}_{2,i}^{\mathsf{IND\text{-}CRO}}\}_{i \in [Q]}\right) = \left(\mathbf{W}^\intercal \mathbf{A} + \mathbf{E}_0 + \beta \mathbf{G}_\ell^\intercal, \{\mathsf{ct}_{2,i}^{\mathsf{CRO}} - \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\intercal \mathbf{W}^\intercal \mathbf{G}_n - \beta \hat{\mathbf{s}}_i^\intercal \mathbf{G}_n\}_{i \in [Q]}\right).$$

Assuming LWE, the components computationally hide β. Therefore $\mathsf{Sim}_3(0)$ and $\mathsf{Sim}_3(1)$ are indistinguishable, thereby proving the IND-CRO assumption. □

We also show that the underlying circular assumption of the IND-CRO is implied by the underlying circular assumption of CRO

Lemma 21. *Let $\lambda, n, m, d, k, \ell, M, \sigma, q, \sigma_0$ and f^{circ} be parameters and a function as specified in Definition 11. We say that the (subexponential) augmented-f^{circ}-circular security assumption holds if $\mathcal{D}_0$ and $\mathcal{D}_1$ without $\mathbf{R}^*$ in Figure 9 are (sub-exponentially) indistinguishable to all polynomial time attackers.*

$$\begin{aligned}
&\{(\mathsf{hpk}, \mathsf{enc} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)) \mid (\mathsf{hpk}, \mathsf{enc}, \mathsf{hint}) \leftarrow \mathcal{D}_0\}_\lambda \\
\approx &\{(\mathsf{hpk}, \mathsf{enc} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \mathsf{ct}_1, \mathsf{ct}_2)) \mid (\mathsf{hpk}, \mathsf{enc}, \mathsf{hint}) \leftarrow \mathcal{D}_1\}_\lambda
\end{aligned}$$

Assuming the (subexponential) f^{circ}-circular security assumption (assumption 2) holds, the (subexponential) augmented-f^{circ}-circular security assumption also holds.

Proof. The lemma follows from the same set of observations from the proof of lemma 20. It is easy to observe that the simulator $\mathsf{Sim}_0'(\beta)$, which is identical to $\mathsf{Sim}_0(\beta)$ except that it does not sample nor output the hint $\mathbf{R}^*$, perfectly samples the augmented-f^{circ}-circular security distributions. By the f^{circ}-circular security assumption, the simulator is indistinguishable to $\mathsf{Sim}_1'(\beta)$, which is again identical to $\mathsf{Sim}_1(\beta)$ except not sampling nor outputting the hint. Now, since Sim_1' is already efficient, by LWE $\mathsf{Sim}_1'(0)$ is indistinguishable to $\mathsf{Sim}_1'(1)$, which completes the proof. □

6.2 Oblivious LWE Sampling

We recall the definition of oblivious LWE sampler from [WW21]. Just as definition 10 of functional encodings, we slightly relax the succinctness requirement while ensuring the outcome is non-trivially succinct, which suffices for applications.

Definition 12 (Oblivious LWE Sampler)**.** *Let λ be the security parameter. All other parameters implicitly depend on λ. Let $n, \ell, q, \sigma, B_{\mathbf{e}}$ be lattice parameters. An oblivious LWE sampler consists of the following algorithms*

- $\mathsf{genCRS}(1^\lambda, 1^Q) \to \mathsf{crs}$
- $\mathsf{init}(1^Q, \mathbf{A} \in \mathbb{Z}_q^{n\times\ell}) \to \mathsf{pub}$, *where* $|\mathsf{pub}| = Q\ell^{1-\epsilon}\log q + \mathsf{poly}(\lambda)$ *for some constant* $\epsilon < 1$, *i.e.,* pub *is smaller than* Q *LWE samples in* $\mathbb{Z}_q^\ell$*.*
- $\mathsf{Sample}(\mathsf{crs}, \mathsf{pub}, i) = \mathbf{b}_i \in \mathbb{Z}_q^\ell$ *is deterministic.*
- $\mathsf{Sim}(1^\lambda, 1^Q, \mathbf{A}, \{\hat{\mathbf{b}}_i\}_{i\in[Q]}) \to (\mathsf{crs}, \mathsf{pub}, \{\tilde{\mathbf{s}}_i\}_{i\in[Q]})$

An oblivious LWE sampler should satisfy the following properties

Correctness *Let* $(\mathbf{A}, \mathbf{T}) \leftarrow \mathsf{TrapGen}(1^n, q, \ell)$, $\mathsf{crs} \leftarrow \mathsf{genCRS}(1^\lambda, 1^Q)$, $\mathsf{pub} \leftarrow \mathsf{init}(1^Q, \mathbf{A})$, $\mathbf{b} \leftarrow \mathsf{Sample}(\mathsf{crs}, \mathsf{pub}, i)$, *then with overwhelming probability there exists* $\tilde{\mathbf{s}} \in \mathbb{Z}_q^n, \tilde{\mathbf{e}} \in [-B_{\mathbf{e}}, B_{\mathbf{e}}]^\ell$ *such that* $\mathbf{b}^\intercal = \tilde{\mathbf{s}}^\intercal\mathbf{A} + \tilde{\mathbf{e}}^\intercal$. *In other words, the output of* Sample *should be LWE samples with respect to* $\mathbf{A}$*.*

Security *The following distributions are computationally indistinguishable.*

- **Real distribution:** $(\mathbf{A}, \mathbf{T}) \leftarrow \mathsf{TrapGen}(1^n, q, \ell)$, $\mathsf{crs} \leftarrow \mathsf{genCRS}(1^\lambda, 1^Q)$, $\mathsf{pub} \leftarrow \mathsf{init}(1^Q, \mathbf{A})$, *for all* $i \in [Q]$, $\mathbf{b}_i \leftarrow \mathsf{Sample}(\mathsf{crs}, \mathsf{pub}, i)$, $\tilde{\mathbf{s}}_i = \mathsf{LWE}_{\mathbf{T}}^{-1}(\mathbf{b})$, *where* $\mathsf{LWE}_{\mathbf{T}}^{-1}$ *is the efficient LWE secret solving procedure given the trapdoor. Output* $(\mathsf{crs}, \mathbf{A}, \mathsf{pub}, \{\tilde{\mathbf{s}}_i\}_{i\in[Q]})$
- **Simulated distribution:** $(\mathbf{A}, \mathbf{T}) \leftarrow \mathsf{TrapGen}(1^n, q, \ell)$. *Sample* $\hat{\mathbf{s}} \leftarrow \mathbb{Z}_q^n$, $\widehat{\mathbf{e}} \leftarrow \mathcal{D}_\sigma^\ell$ *and set* $\hat{\mathbf{b}}^\intercal = \hat{\mathbf{s}}^\intercal\mathbf{A} + \widehat{\mathbf{e}}^\intercal$. *Simulate* $(\mathsf{crs}, \mathsf{pub}, \{\tilde{\mathbf{s}}_i\}_{i\in[Q]}) \leftarrow \mathsf{Sim}(1^\lambda, 1^Q, \mathbf{A}, \{\hat{\mathbf{b}}_i\}_{i\in[Q]})$ *and output* $(\mathsf{crs}, \mathbf{A}, \mathsf{pub}, \{\tilde{\mathbf{s}}_i - \hat{\mathbf{s}}_i\}_{i\in[Q]})$.

In other words, the output distribution of oblivious LWE sampling is indistinguishable to the simulated distribution which hides an additional LWE shift.

6.3 Construction

We can construct an oblivious LWE sampler from IND-CRO following the exact same recipe in the construction 1 of functional encoding. Instead of encoding an input $\mathbf{x}$, it encodes a bit β, which is zero in the real scheme, and instead of evaluating arbitrary functions g_i, it can be viewed as evaluating a specific function $-\beta \cdot \hat{\mathbf{b}}_i$, where $\hat{\mathbf{b}}$ is an additional string in the CRS. These ideas come from the construction of oblivious LWE sampler in [WW21].

Construction 2 (Oblivious LWE Sampler (Sketched))**.** *The construction is described as follows.*

- $\mathsf{genCRS}(1^\lambda, 1^Q)$ *output* $\mathsf{crs} = (\mathsf{pp}, \mathbf{A}, \{\mathbf{R}_i^*, \hat{\mathbf{b}}_i\}_{i\in[Q]})$, *where* $\mathsf{pp}, \mathbf{A}, \mathbf{R}_i^*$ *are defined identically as construction 1, while* $\hat{\mathbf{b}}_i \leftarrow \mathbb{Z}_q^n$ *are sampled at random.*
- $\mathsf{init}(1^Q, \mathbf{A})$ *outputs* pub *which consists of every element in the output of the encoding algorithm* Enc *in construction 1 except the message term* $\mathsf{dct}_{\mathbf{x}}$ *(since there is no message). The message term is replaced with a dual GSW ciphertext of* 0*:*

$$\mathsf{dct}_0 = \mathbf{W}^\intercal\mathbf{A} + \mathbf{E}_0, \quad \text{where } \mathbf{W} \leftarrow \mathbb{Z}_q^{n\times\ell\lceil\log q\rceil}, \mathbf{E}_0 \leftarrow \mathcal{D}_\sigma^{\ell\lceil\log q\rceil\times\ell}.$$

- $\mathsf{Sample}(\mathsf{crs}, \mathsf{pub}, i)$ *largely follows the decryption algorithm* Dec *in construction* 1*, with the difference that the message homomorphism is replaced by*

  $$\mathsf{dct}_{\hat{\mathbf{b}}_i} = \mathbf{G}^{-1}(\hat{\mathbf{b}}_i)^\intercal \mathsf{dct}_0 = \mathbf{w}_{\hat{\mathbf{b}}_i}^\intercal \mathbf{A} + \mathbf{e}_{\hat{\mathbf{b}}_i}^\intercal .$$

  Finally, the sampling algorithm outputs LWE sample induced by the correctness equation (equation (13)*) from the decryption algorithm*

  $$\tilde{\mathbf{s}}_i^\intercal \mathbf{A} + \tilde{\mathbf{e}}_i^\intercal = (\mathbf{s}_i + \mathbf{u}_i + \mathbf{w}_{\hat{\mathbf{b}}_i})^\intercal \mathbf{A} + \tilde{\mathbf{e}}_i^\intercal = \underline{\mathsf{hct}}_i' + \mathsf{dct}_i + \mathsf{dct}_{\hat{\mathbf{b}}_i} + \Delta \left\lfloor \frac{\mathbf{c}_{\mathbf{s},i}^\intercal \mathbf{G}^{-1}(\mathbf{A}) - \mathbf{c}_i^\intercal}{\Delta} \right\rceil + \mathbf{c}_i^\intercal . \quad (18)$$

- $\mathsf{Sim}(1^\lambda, 1^Q, \mathbf{A}, \{\hat{\mathbf{b}}_i\}_{i \in [Q]})$ *executes* genCRS *to obtain* crs*, but programs* $\hat{\mathbf{b}}_i$ *using its input, it then runs* $\mathsf{init}(1^Q, \mathbf{A})$ *for* pub*, but replaces* dct_0 *as a dual GSW ciphertext of* 1*,*

  $$\mathsf{dct}_0 = \mathbf{W}^\intercal \mathbf{A} + \mathbf{E}_0 + \mathbf{G}_\ell^\intercal, \quad \text{where } \mathbf{W} \leftarrow \mathbb{Z}_q^{n \times \ell \lceil \log q \rceil}, \mathbf{E}_0 \leftarrow \mathcal{D}_\sigma^{\ell \lceil \log q \rceil \times \ell}.$$

  Finally, it execute the $\mathsf{Sample}(\mathsf{crs}, \mathsf{pub}, i)$ *algorithm for each i (with all randomness used to generate* pub *known) and output* $(\mathsf{crs}, \mathsf{pub}, \tilde{\mathbf{s}}_i = (\mathbf{s}_i + \mathbf{u}_i + \mathbf{w}_{\hat{\mathbf{b}}_i}))$*. Note that the components sampled by the simulator admit the correctness equation*

  $$\tilde{\mathbf{s}}_i^\intercal \mathbf{A} + \tilde{\mathbf{e}}_i^\intercal = (\mathbf{s}_i + \mathbf{u}_i + \mathbf{w}_{\hat{\mathbf{b}}_i})^\intercal \mathbf{A} + \tilde{\mathbf{e}}_i^\intercal = \underline{\mathsf{hct}}_i' + \mathsf{dct}_i + \mathsf{dct}_{\hat{\mathbf{b}}_i} + \Delta \left\lfloor \frac{\mathbf{c}_{\mathbf{s},i}^\intercal \mathbf{G}^{-1}(\mathbf{A}) - \mathbf{c}_i^\intercal}{\Delta} \right\rceil + \mathbf{c}_i^\intercal + \hat{\mathbf{b}}_i . \quad (19)$$

The correctness of the oblivious LWE sampler follows directly from the correctness equation 18. The security follows from the same observation used for proving security of construction 1, that all elements in the construction can be simulated by the IND-CRO distribution. In particular, we can simulate all elements in the construction by the sample $(\mathsf{hpk}, \mathsf{enc} = (\mathsf{hct}, \mathsf{hct}_0, \mathbf{A}, \mathbf{D}, \{\mathbf{v}_i\}_{i \in [k']}, \mathsf{ct}_0, \mathsf{ct}_1, \mathsf{ct}_2), \mathbf{R}^*)$ from the IND-CRO distribution $\mathcal{D}_\beta$ with the function tuple defined in assumption 3 as follows:

- $\mathbf{b}_i = \hat{\mathbf{s}}_i^\intercal \mathbf{A} + \widehat{\mathbf{e}}_i = \mathbf{v}_i$.
- $\tilde{\mathbf{s}}_i' \leftarrow \mathbb{Z}_q^n$, $\tilde{\mathbf{e}}_i' \leftarrow \mathcal{D}_{\sigma_\mathbf{e}}^\ell$, where implicitly $\tilde{\mathbf{s}}_i = \tilde{\mathbf{s}}_i' + \beta \hat{\mathbf{s}}_i$, $\tilde{\mathbf{e}}_i = \tilde{\mathbf{e}}_i' + \beta \widehat{\mathbf{e}}_i$.
- $\mathsf{dct}_0 = \mathsf{ct}_0 = \mathbf{W}^\intercal \mathbf{A} + \mathbf{E}_0 + \beta \mathbf{G}_\ell^\intercal$, $\mathsf{dct} = \mathsf{ct}_1$.
- $\mathbf{c}_i = ((\tilde{\mathbf{s}}_i')^\intercal \mathbf{A} + (\tilde{\mathbf{e}}_i')^\intercal - \mathsf{dct}_{f_i} - \underline{\mathsf{hct}}_i' - \mathsf{dct}_{\widetilde{g}_i} \bmod \Delta)$
- $\mathbf{c}_{\mathbf{s},i}^\intercal = \mathsf{ct}_{2,i} + \tilde{\mathbf{s}}_i' \mathbf{G}$, where $\mathsf{ct}_{2,i} = \mathbf{r}^\intercal \mathbf{D}_i + \mathbf{e}_{\mathbf{D},i}^\intercal - \mathsf{vec}(\mathbf{G}^{-1}(-\overline{\mathsf{hct}}_{0,i}))^\intercal \mathbf{U}^\intercal \mathbf{G} - \mathbf{G}_\ell^{-1}(\mathbf{v}_i)^\intercal \mathbf{W}^\intercal \mathbf{G}_n - \beta \hat{\mathbf{s}}_i^\intercal \mathbf{G}_n$ is the i-th segment of ct_2.

When $\beta = 0$, the simulation statistically simulates the real distribution $(\mathsf{crs}, \mathbf{A}, \mathsf{pub}, \{\tilde{\mathbf{s}}_i' = \tilde{\mathbf{s}}_i\}_{i \in [Q]})$, while when $\beta = 1$, the simulation statistically simulates the simulated distribution $(\mathsf{crs}, \mathbf{A}, \mathsf{pub}, \{\tilde{\mathbf{s}}_i' = \tilde{\mathbf{s}}_i - \hat{\mathbf{s}}_i\}_{i \in [Q]})$. Therefore, the security reduces directly to the indistinguishability of the IND-CRO output components, which is provided by the assumption.

7 Counterexample for Private-Coin Evasive LWE

In this section, we introduce some counterexamples for the private-coin evasive LWE assumptions. Along the way, we would highlight the problematic heuristic behind current evasive LWE formulations, and provide a concrete attack strategy against the heuristic. Unlike previous counterexamples [VWW22, BÜW24], which provided counterexamples based on obfuscation or in the non-standard case where the precondition does not have the target matrix, our attack is a new zeroization attack inspired by techniques from [HJL21]. This is a new class of attacks that were not known for any formulation of Evasive LWE.

More specifically, we provide a simple counterexample to the version of the assumption called *private-coin binding evasive LWE* (named by [BÜW24]), which underpins the security of many recent obfuscation and advanced encryption schemes [VWW22, WWW22, HLL23, ARYY23, HLL24, MPV24, CM24, BDJ+24, AKY24]. Our counterexample is based on the standard LWE assumption, and involves distributions that are highly similar to existing schemes, particularly [AKY24].

7.1 Evasive LWE Definitions

We start by introducing the private-coin evasive LWE assumption [Tsa22, VWW22]. In the following, we adopt the private-coin binding evasive LWE formulation by Brzuska et.al. [BÜW24], but we also note that all existing variants of evasive LWE assumptions share similar heuristics, and we believe that our attack strategy likely applies to other variants of private-coin evasive LWE assumptions.

Assumption 4 (Private-coin Binding Evasive LWE [BÜW24])**.** *Let the parameters $n, m, m_P, t, q, \sigma_B, \sigma_P, \sigma_T$ be parameterized by λ. Let* Samp *be an efficient algorithm taking input 1^λ and outputs*

$$\mathbf{S} \in \mathbb{Z}_q^{n\times t}, \quad \mathbf{P} \in \mathbb{Z}_q^{n\times m_P}, \quad \mathsf{aux} \in \{0,1\}^*$$

The **precondition** *of evasive LWE states that the following two distributions are computationally indistinguishable:*

$$(\mathbf{B}, \mathbf{P}, \mathbf{S}^\intercal\mathbf{B} + \mathbf{E_B}, \mathbf{S}^\intercal\mathbf{P} + \mathbf{E_P}, \mathsf{aux}) \approx (\mathbf{B}, \mathbf{P}, \mathbf{C_B}, \mathbf{C_P}, \mathsf{aux})$$

where $(\mathbf{S}, \mathbf{P}, \mathsf{aux}) \leftarrow \mathsf{Samp}(1^\lambda)$, $\mathbf{B} \leftarrow \mathbb{Z}_q^{n\times m}$, $\mathbf{E_B} \leftarrow \mathcal{D}_{\sigma_B}^{t\times m}$, $\mathbf{E_P} \leftarrow \mathcal{D}_{\sigma_P}^{t\times m_P}$, $\mathbf{C_B} \leftarrow \mathbb{Z}_q^{t\times m}$, $\mathbf{C_P} \leftarrow \mathbb{Z}_q^{t\times m_P}$.

The **postcondition** *of evasive LWE states that the following two distributions are computationally indistinguishable:*

$$(\mathbf{B}, \mathbf{P}, \mathbf{S}^\intercal\mathbf{B} + \mathbf{E_B}, \mathbf{T} = \mathbf{B}^{-1}(\mathbf{P}), \mathsf{aux}) \approx (\mathbf{B}, \mathbf{P}, \mathbf{C_B}, \mathbf{T} = \mathbf{B}^{-1}(\mathbf{P}), \mathsf{aux})$$

where $(\mathbf{S}, \mathbf{P}, \mathsf{aux}) \leftarrow \mathsf{Samp}(1^\lambda)$, $\mathbf{B} \leftarrow \mathbb{Z}_q^{n\times m}$, $\mathbf{E_B} \leftarrow \mathcal{D}_{\sigma_B}^{t\times m}$, $\mathbf{C_B} \leftarrow \mathbb{Z}_q^{t\times m}$, *and the short preimage* $\mathbf{T} = \mathbf{B}^{-1}(\mathbf{P})$ *is sampled from the conditional discrete Gaussian distribution* $\mathbf{T} \leftarrow \mathcal{D}_{\sigma_T}^{m\times m_P}|_{\mathbf{BT}=\mathbf{P}}$.

The evasive LWE assumption states that, for all efficient sampler Samp*, if the precondition holds, then the postcondition holds.*

Informally speaking, the Evasive LWE assumption is built on the following two heuristics:

- **Heuristic 1: No non-trivial use of trapdoors.** The first heuristic is that in the post-condition, the most "effective" way in which an attacker can make use of the trapdoor matrix $\mathbf{T}$ is to multiply it with the corresponding LWE sample $\mathbf{S}^\intercal\mathbf{B} + \mathbf{E_B}$. This multiplication yields the product $(\mathbf{S}^\intercal\mathbf{B} + \mathbf{E_B})\mathbf{T} = \mathbf{S}^\intercal\mathbf{P} + \mathbf{E_B}\mathbf{T}$. Under this heuristic, if somehow $\mathbf{S}^\intercal\mathbf{P} + \mathbf{E_B}\mathbf{T}$ turned out to be pseudorandom, then $\mathbf{T}$ is useless.

In other words,

$$(\mathbf{S}^{\top}\mathbf{B} + \mathbf{E_B}, \mathbf{S}^{\top}\mathbf{P} + \mathbf{E_B}\mathbf{T}, \mathsf{aux}) \approx (\$, \$, \mathsf{aux}) \implies (\mathbf{S}^{\top}\mathbf{B} + \mathbf{E_B}, \mathbf{T}, \mathsf{aux}) \approx (\$, \mathbf{T}, \mathsf{aux})$$

- **Heuristic 2: No attack on structured error.** The second heuristic says that the resulting LWE sample $\mathbf{S}^{\top}\mathbf{P} + \mathbf{E_B}\mathbf{T}$ with a structured error vector $\mathbf{E_B}\mathbf{T}$ produced in the post-condition, is essentially as immune against attacks as a sample with fresh and independent Gaussian error from the correct LWE error distribution. In other words,

$$(\mathbf{S}^{\top}\mathbf{B} + \mathbf{E_B}, \mathbf{S}^{\top}\mathbf{P} + \mathbf{E_P}, \mathsf{aux}) \approx (\$, \$, \mathsf{aux}) \implies (\mathbf{S}^{\top}\mathbf{B} + \mathbf{E_B}, \mathbf{S}^{\top}\mathbf{P} + \mathbf{E_B}\mathbf{T}, \mathsf{aux}) \approx (\$, \$, \mathsf{aux})$$

New Dimension of Attacks. We note that all the current known counterexample including [VWW22, BÜW24] attack the first heuristic. The attacks proceed by setting up a contrived auxiliary information aux that contains an obfuscated program expecting short preimages as input. The rationale behind the ongoing research agenda of constructing schemes based on this variant of Evasive LWE is that "natural" schemes typically do not make use of such contrived auxiliary information. Therefore, one might believe that the schemes are *plausibly* secure.

Our attack on the other hand exploits vulnerabilities in the second heuristic, for which no contradictions were known. Moreover, it is extremely simple, and involves similar structures underlying existing schemes.

7.2 Overview on Our Attack Strategy

In the following, we will stick to evasive LWE assumptions where $t = 1$, i.e., the secret $\mathbf{s}$ and errors $\mathbf{e}$ are vectors.

Our main technical inspiration for the counterexample comes from the recent application of private-coin-evasive LWE to designing pseudorandom obfuscation [BDJ+24, AKY24] and other related primitives [VWW22, WWW22, HLL23, ARYY23, HLL24, MPV24, CM24, BDJ+24, AKY24]. Notable feature of many of these constructions is a novel usage of predicate encryption [GVW15] encodings to allow a "secure" decryption (under evasive LWE, in fact we show that this step is problematic) of pseudorandom function evaluations.

We start by recalling the overall approach. We assume that the reader is familiar with the predicate encryption scheme of Gorbunov et. al. [GVW15] and its predecessor ABE scheme by Boneh et. al. [BGG+14] (henceforth referred to as GVW and BGG). The idea in these schemes is that one considers a distribution of the form:

$$\mathsf{hpk}, \mathsf{hct}(\mathbf{k}), \mathbf{A}, \mathbf{C}, \ \overline{\mathsf{ct}} = (\mathbf{s}^{\top}[\mathbf{A} - \mathsf{bits}(\mathsf{hct}(\mathbf{k}))^{\top} \otimes \mathbf{G} \mid \mathbf{C} - \mathbf{r}^{\top} \otimes \mathbf{G}] + \mathbf{e}^{\top})\,.$$

Above hpk is a public key for the GSW encryption scheme with secret key $\mathbf{r}$, $\mathsf{hct}(\mathbf{k})$ is a GSW encryption of a PRF key $\mathbf{k}$ which needs to be evaluated. $\overline{\mathsf{ct}}$ is the predicate encoding of the PRF key under matrices $\mathbf{A}, \mathbf{C}$.

The main advantage of the GVW encodings is that, by appropriate homomorphic evaluation procedure, one can compute samples of the form:

$$\mathbf{s}^{\top}\mathbf{A}_f + f(\mathbf{k})^{\top} + \widehat{\mathbf{e}}^{\top},$$

where $f : \{0,1\}^k \to \mathbb{Z}_q^{\ell}$ is the required PRF computation. Importantly, $\mathbf{A}_f$ is efficiently computable from only the public matrices $\mathbf{A}, \mathbf{C}$ and the computation f.

One approach to securely learn $f(\mathbf{k})$ could be to provide a sample of the form:

$$\mathbf{s}^{\top}\mathbf{A}_f + \mathbf{e}'^{\top},$$

for an independent wide-enough Gaussian error $\mathbf{e}'$. One can show that this only reveals $f(\mathbf{k})$, although the decryption is noisy where the lower bits of the output might be incorrect. This intuition is not new, and has already been useful in many prior works, such as in the context of laconic function evaluation [QWW18] and indistinguishability obfuscation [GJLS21].

Unfortunately, the schemes cannot afford to provide samples of the form $\mathbf{s}^{\top}\mathbf{A}_f + \mathbf{e}^{\top}$ as the number of samples will depend linearly on the number of output bits and can't be re-used for different secrets. This approach will violate the succinctness needed for iO applications. This is where Evasive LWE comes in.

Compression via Evasive LWE. Instead of releasing samples of the form $\mathbf{s}^{\top}\mathbf{A}_f + \mathbf{e}^{\top}$, one can release succinct LWE samples $(\mathbf{B}, \mathbf{s}^{\top}\mathbf{B} + \mathbf{e}_{\mathbf{B}}^{\top})$ with dimension of $\mathbf{B}$ much smaller than $\mathbf{A}_f$, along with a trapdoor matrix $\mathbf{T} = \mathbf{B}^{-1}(\mathbf{A}_f)$. This will let us generate samples of the form $\mathbf{s}^{\top}\mathbf{A}_f + \mathbf{e}_{\mathbf{B}}^{\top}\mathbf{T}$ where the error $\mathbf{e}_{\mathbf{B}}^{\top}\mathbf{T}$ is not random, but nevertheless it is small in norm. Therefore the trapdoor serves as a succinct proxy for the fresh sample $\mathbf{s}^{\top}\mathbf{A}_f + \mathbf{e}'^{\top}$ where $\mathbf{e}'$ is an independent Gaussian error. The resulting distribution is:

$$\textbf{Postcondition Distribution:}\qquad \begin{aligned} &\mathbf{B}, \mathbf{s}^{\top}\mathbf{B} + \mathbf{e}_{\mathbf{B}}^{\top}, \mathbf{T} = \mathbf{B}^{-1}(\mathbf{A}_f), \\ &\mathsf{hpk}, \mathsf{hct}(\mathbf{k}), \mathbf{A}, \mathbf{C} \\ &\overline{\mathsf{ct}} = (\mathbf{s}^{\top}[\mathbf{A} - \mathsf{bits}(\mathsf{hct}(\mathbf{k}))^{\top} \otimes \mathbf{G} | \mathbf{C} - \mathbf{r}^{\top} \otimes \mathbf{G}] + \mathbf{e}^{\top}). \end{aligned} \tag{20}$$

We show that this heuristic can be proven secure under evasive LWE and LWE, assuming the security of the PRF, whereas the distribution is not pseudorandom. This argument requires the predicate encryption and GSW encryption to have some natural structural properties that can be satisfied under LWE and were shown to exist by [HJL21].

Security via Evasive LWE. To prove the trapdoor heuristic described above is secure, we analyze the precondition statement required by evasive LWE. The precondition requires us to show that $(\mathbf{B}, \mathbf{s}^{\top}\mathbf{B} + \mathbf{e}^{\top}, \mathbf{s}^{\top}\mathbf{A}_f + \widehat{\mathbf{e}}^{\top})$ with *fresh* $\widehat{\mathbf{e}}$ in the following distribution is indistinguishable to random, given $(\mathsf{hpk}, \mathsf{hct}, \mathbf{A}, \mathbf{C}, \overline{\mathsf{ct}})$.

$$\textbf{Precondition Distribution:}\qquad (\mathbf{B}, \mathbf{s}^{\top}\mathbf{B} + \mathbf{e}^{\top}, \mathbf{s}^{\top}\mathbf{A}_f + \widehat{\mathbf{e}}^{\top}, \mathsf{hpk}, \mathsf{hct}, \mathbf{A}, \mathbf{C}, \overline{\mathsf{ct}}).$$

This can be more or less proven similarly as in prior works related to laconic function evaluation [QWW18], the only notable requirement that we need is that the function $f(\star)$ is pseudorandom over the entire domain $\mathbb{Z}_q$. This is also proven in lemma 25.

Once the precondition can be proven, the evasive LWE assumption asserts that the postcondition holds, which requires that in the distribution in Equation (20), the LWE sample $\mathbf{s}^{\top}\mathbf{B} + \mathbf{e}_{\mathbf{B}}^{\top}$ is indistinguishable to random. Once we replace the sample by random, we can appeal to LWE to argue that even $(\mathsf{hct}(\mathbf{k}), \overline{\mathsf{ct}})$ is indistinguishable to random and independent of the pseudorandom function key. This would let us complete a *security proof*. Unfortunately, we show counterexamples on the post-condition.

Vulnerability in the Post-Condition. Our counterexample examines the noisy output learned upon decryption. Namely, if one inspects the decryption equation for the function output f, we get a noisy outcome of the form

$$(\mathbf{s}^{\top}\mathbf{A}_f + f(\mathbf{k})^{\top} + \widehat{\mathbf{e}}^{\top}) - (\mathbf{s}^{\mathbf{T}}\mathbf{B} + \mathbf{e}_{\mathbf{B}}^{\top})\mathbf{T} = f(\mathbf{k})^{\top} + \widehat{\mathbf{e}}^{\top} - \mathbf{e}_{\mathbf{B}}^{\top}\mathbf{T}$$

Moreover, $\widehat{\mathbf{e}}^{\intercal}$ is a noise that is a sum of two noises $\widehat{\mathbf{e}}_{\mathsf{HE}} + \widehat{\mathbf{e}}_{\mathsf{PE}}$ where $\widehat{\mathbf{e}}_{\mathsf{HE}}$ is the noise that arises from the GSW homomorphic evaluation[10], and $\widehat{\mathbf{e}}_{\mathsf{PE}}$ is the noise from the GVW homomorphic evaluation. Moreover, it turns out that $\widehat{\mathbf{e}}_{\mathsf{PE}}^{\intercal} = \mathbf{e}^{\intercal}\mathbf{H}$ where $\mathbf{H}$ is known low-norm integer matrix and is an efficient deterministic function of the public matrices, hct and the function f.

Thus, now examining the structure decrypted output:

$$f(\mathbf{k})^{\intercal} + \widehat{\mathbf{e}}_{\mathsf{HE}}^{\intercal} + \underbrace{\mathbf{e}^{\intercal}\mathbf{H}}_{\widehat{\mathbf{e}}_{\mathsf{PE}}^{\intercal}} - \mathbf{e}_{\mathbf{B}}^{\intercal}\mathbf{T}$$

The next step is to observe the following claims applying to the colored components. Recall that given an arbitrary function f represented by a Boolean circuit C_f, to perform GSW and GVW homomorphic evaluation, one needs to convert it into an arithmetic circuit C_f' computing the same function, consisting of ADD and MULT gates between two wire values, or between one wire value and a constant. Inspired by [HJL21], we show that, instead of using the canonical conversion from a Boolean circuit to arithmetic circuit, by using a specific arithmetic circuit implementation, one can control the parity bits of the noises in the output ciphertexts resulting from GSW/GVW homomorphic evaluation.

- By choosing a special arithmetic circuit implementation of f, the noise term $\mathbf{e}^{\intercal}\mathbf{H}$ arose from the predicate encryption scheme is always even. The ideas for this part already existed in the work of [HJL21]. Note that this modification only affects the structure of the noises and has no bearings on the security of the predicate encryption scheme, when the modulus q is odd. This is proven in Lemma 22.

- Leveraging the special arithmetic circuit implementation of f, and by setting the noises in the GSW public key to be even (for which LWE holds when q is odd), we have that $f(\mathbf{k})^{\intercal} + \widehat{\mathbf{e}}_{\mathsf{HE}}^{\intercal} \mod q$ is also even. This modification also has no bearing to the security of the GSW encryption. The ideas for such a scheme already existed in the work of [HJL21]. This is proven in Lemma 23

- The final missing piece is the component $\mathbf{e}_{\mathbf{B}}^{\intercal}\mathbf{T}$. Note that both $\mathbf{T}$ and $\mathbf{e}_{\mathbf{B}}$ are small norm integer vectors. Furthermore, because $\mathbf{T}$ is wide, modulo 2, the error $\mathbf{e}_{\mathbf{B}}^{\intercal}\mathbf{T}$ lives in a low-dimensional vector space mod 2.

As a consequence, with these observations, one can simply compute the result of:

$$\mathbf{y}^{\intercal} = f(\mathbf{k})^{\intercal} + \widehat{\mathbf{e}}_{\mathsf{HE}}^{\intercal} + \widehat{\mathbf{e}}_{\mathsf{PE}}^{\intercal} - \mathbf{e}_{\mathbf{B}}^{\intercal}\mathbf{T} \mod 2. \tag{21}$$

Since the sum of the first three terms modulo 2 yields zero, we can then check if the result lives inside a low-dimensional vector space. In Lemma 24, we show that this strategy will not work if the encodings were random. This concludes a high-level overview of the counterexample.

Before proceeding to details of our counterexamples, we make two remarks. First, our counterexamples attack the output plus noise term, and hence is zeroizing. The fact that the output $f(\mathbf{k})$ is large and marginally pseudorandom does not help since the noises are correlated. Second, our manipulation of the structure of GVW/GSW noises highlights that it is easy to find bad examples of arithmetic circuit implementation of f. While it is easy to avoid these specific bad examples, it is unclear how to characterize good circuit implementation.

[10] Recall that the predicate encryption encodes GSW encryption of $\mathbf{k}$ as an attribute. This noise corresponds to the inner-product of the GSW secret key with a GSW evaluation of this ciphertext.

7.3 Counterexample

In this section, we give a fully specified description of our counterexample along with all the required lemmas. The construction fully follows the overview given in section 7.2

The GVW Encodings We start by recalling the algebraic structure of the GVW predicate encryption.

Theorem 9 (GVW Encodings [GVW15])**.** *Let n, q be integers and $m = n\lceil \log q \rceil$. There exist efficient algorithms* Evalf, Evalfx *such that for all input $\mathbf{x} \in \{0,1\}^k$, all function $f : \{0,1\}^k \to \mathbb{Z}_q^{(n+1)\times\ell}$ where every bit of (the bitwise representation of) the output can be computed by a depth d circuit, and all matrices $\mathbf{A} \in \mathbb{Z}_q^{n\times km}, \mathbf{C} \in \mathbb{Z}_q^{n\times nm}$, it holds that*

- $\mathsf{Evalf}(\mathbf{A}, \mathbf{C}, f) = \mathbf{A}_f \in \mathbb{Z}_q^{n\times\ell}$ *outputs a decryption pad that only depends on f and public matrices.*
- $\mathsf{Evalfx}(\mathbf{A}, \mathbf{C}, f, \mathbf{x}) = \mathbf{H}_{f,\mathbf{x}} \in \mathbb{Z}^{(n+k)m\times\ell}$ *outputs a evaluation matrix $\mathbf{H}_{f,\mathbf{x}}$ with bounded norm, $\|\mathbf{H}_{f,\mathbf{x}}\| = m^{O(d)}$.*
- *The output of* Evalf *and* Evalfx *satisfies the following decryption equation.*

$$[\mathbf{A} - \mathbf{x}^\intercal \otimes \mathbf{G} | \mathbf{C} - \mathbf{r}^\intercal \otimes \mathbf{G}] \cdot \mathbf{H}_{f,\mathbf{x}} = \mathbf{A}_f - \mathbf{Y}, \quad \text{where } \mathbf{Y} = \begin{pmatrix} \mathbf{0}^{n\times\ell} \\ (\mathbf{r}^\intercal, -1) \cdot f(\mathbf{x}) \end{pmatrix}$$

The idea of GVW predicate encryption is to combine the above encoding with a GSW homomorphic encryption scheme with secret key $\mathbf{r}$, where the format of $\mathbf{Y}$ corresponds to the linear decryption. In particular, by plugging in $\mathbf{x}$ as the (bitwise representation of) a GSW ciphertext $\mathbf{x} = \mathsf{hct}(\mathbf{k}) = \begin{pmatrix} \mathbf{B}_{\mathsf{HE}} \\ \mathbf{r}^\intercal \mathbf{B}_{\mathsf{HE}} + \mathbf{e}_{\mathsf{HE}}^\intercal \end{pmatrix} \mathbf{R} + \mathbf{k}^\intercal \otimes \mathbf{G}$ of some random seed $\mathbf{k}$, and the function f as the GSW homomorphic evaluation for a pseudorandom function $f = \mathsf{GSW.Eval}(\cdot, \mathsf{PRF})$, one can obtain the equation

$$[\mathbf{A} - \mathbf{x}^\intercal \otimes \mathbf{G} | \mathbf{C} - \mathbf{r}^\intercal \otimes \mathbf{G}] \cdot \mathbf{H}_{f,\mathbf{x}} = \mathbf{A}_f - \begin{pmatrix} \mathbf{0}^{n\times\ell} \\ -\mathsf{PRF}(\mathbf{k}) - \mathbf{e}_{\mathsf{HE}}^\intercal \mathbf{R}_{\mathsf{PRF}} \end{pmatrix}. \tag{22}$$

Subsequently, with GVW secret $\mathbf{s}^\intercal = (\mathbf{t}^\intercal, 1)$ and GVW error $\mathbf{e}_{\mathsf{PE}}$, we have the following homomorphic relation over LWE samples.

$$(\mathbf{s}^\intercal [\mathbf{A} - \mathbf{x}^\intercal \otimes \mathbf{G} | \mathbf{C} - \mathbf{r}^\intercal \otimes \mathbf{G}] + \mathbf{e}_{\mathsf{PE}}^\intercal) \cdot \mathbf{H}_{f,\mathbf{x}} = \mathbf{s}^\intercal \mathbf{A}_f + \mathsf{PRF}(\mathbf{k})^\intercal + \mathbf{e}_{\mathsf{HE}}^\intercal \mathbf{R}_{\mathsf{PRF}} + \mathbf{e}_{\mathsf{PE}}^\intercal \mathbf{H}_{f,\mathbf{x}}. \tag{23}$$

As described in the overview (section 7.2), we aim to provide modified GVW and GSW homomorphic evaluations, such that the error terms in equation (23) admits the correlations that $\mathsf{PRF}(\mathbf{k})^\intercal + \mathbf{e}_{\mathsf{HE}}^\intercal \mathbf{R}_{\mathsf{PRF}}$ is even and $\mathbf{e}_{\mathsf{PE}}^\intercal \mathbf{H}_{f,\mathbf{x}}$.

Correlation-inducing gates Our approach toward the desired correlation follows the same framework in [HJL21]. We introduce the so-called correlation-inducing gates, which are special identity gates that introduce correlation to errors without affecting the homomorphic evaluation procedure. This allows us to transform any circuit into a "bad" implementation (by injecting correlation-inducing gates appropriately to the original circuit) such that the homomorphic evaluation of the bad circuit implementation comes with correlated decryption error.

Notations In the following, we always use odd prime modulus q, and we write 2^{-1} to denote the multiplicative inverse of 2 under mod q. For integer matrix $\mathbf{T}$, we abuse the notation and write $\mathbf{T} = b \pmod 2$ for bit $b \in \{0,1\}$ to denote the condition where every entry $\mathbf{T}[i,j]$ of $\mathbf{T}$ satisfies $\mathbf{T}[i,j] = b \pmod 2$. We say $\mathbf{T}$ is even when $\mathbf{T} = 0 \pmod 2$, and $\mathbf{T}$ is odd when $\mathbf{T} = 1 \pmod 2$

We start by introducing the correlation-inducing gates for the GVW encodings such that we can argue $\mathbf{e}_{\mathsf{PE}}^\intercal \mathbf{H} = 0 \pmod 2$.

Observation 1 (Even-error gate for GVW)**.** *For any GVW encoding* $[\mathbf{A}-\mathbf{x}^\top\otimes\mathbf{G}|\mathbf{C}-\mathbf{r}^\top\otimes\mathbf{G}]$*, homomorphically evaluating the function* $h(b)=2^{-1}\cdot b+2^{-1}\cdot b$ *on the input wires result in the even-error evaluation matrix* $\mathbf{H}_h=2\mathbf{G}^{-1}(2^{-1}\mathbf{G})$*, which admits the relation*

$$[\mathbf{A}-\mathbf{x}^\top\otimes\mathbf{G}|\mathbf{C}-\mathbf{r}^\top\otimes\mathbf{G}]\mathbf{H}_0=[\mathbf{A}\mathbf{H}_0-\mathbf{x}^\top\otimes\mathbf{G}|\mathbf{C}\mathbf{H}_0-\mathbf{r}^\top\otimes\mathbf{G}]$$

where the term on the right is still a GVW encoding for the same input.

By adding the even-error gate on each input wire of the homomorphic evaluated circuit, we obtain a new Evalfx algorithm which always outputs evaluation matrix of form $\mathbf{H}_0\mathbf{H}$, with the guarantee that $\mathbf{H}_0\mathbf{H}=0 \pmod 2$.

For technical requirements of our evasive LWE postcondition distinguisher, we also need $\mathbf{A}_f$ to be marginally random. This can be achieved by introducing additional encodings of 0, and appending a dummy addition-by-0 gate on the output wires.

Observation 2 (Addition by 0 gate for GVW)**.** *Let* $\mathbf{A}_0\leftarrow\mathbb{Z}_q^{n\times\ell}$ *be a random matrix, parsed as a GVW encoding of a zero vector. By extending the GVW encoding to* $[\mathbf{A}_0|\mathbf{A}-\mathbf{x}^\top\otimes\mathbf{G}|\mathbf{C}-\mathbf{r}^\top\otimes\mathbf{G}]$*, and evaluating a circuit (with evaluation matrix* $\mathbf{H}_{f,\mathbf{x}}$*) an additional addition-by-0 gate computing the function* $h(v)=v+2\cdot 0$*, we obtain a new evaluation matrix* $\mathbf{H}'=\begin{pmatrix}2\mathbf{I}\\ \mathbf{H}_{f,\mathbf{x}}\end{pmatrix}$ *that admits the following decryption equation.*

$$[\mathbf{A}_0|\mathbf{A}-\mathbf{x}^\top\otimes\mathbf{G}|\mathbf{C}-\mathbf{r}^\top\otimes\mathbf{G}]\cdot\begin{pmatrix}2\mathbf{I}\\ \mathbf{H}_{f,\mathbf{x}}\end{pmatrix}=(\mathbf{A}_f+2\mathbf{A}_0)-\begin{pmatrix}\mathbf{0}^{n\times\ell}\\ (\mathbf{r}^\top,-1)\cdot f(\mathbf{x})\end{pmatrix}$$

The decryption pad $\mathbf{A}'_f=\mathbf{A}_f+2\mathbf{A}_0$ *is uniformly random in* $\mathbb{Z}_q^{(n+1)\times\ell}$*, and the evaluation matrix* $\mathbf{H}'=\begin{pmatrix}2\mathbf{I}\\ \mathbf{H}_{f,\mathbf{x}}\end{pmatrix}$ *is even whenever* $\mathbf{H}_{f,\mathbf{x}}$ *is.*

Combining the above two observations, we get the following lemma for GVW homomorphic encodings.

Lemma 22 (GVW homomorphic encodings with bad circuit implementation)**.** *For every circuit* $f\colon\{0,1\}^k\to\mathbb{Z}_q^{(n+1)\times\ell}$ *where each output bit can be computed by a depth d circuit, there exist a functionally equivalent circuit implementation* f' *(containing additional correlation-inducing gates) with constant blowups on circuit depth, such that the GVW homomorphic evaluation for circuit* f' *admits the following conditions.*

- $\mathsf{Evalf}(\mathbf{A}_0,\mathbf{A},\mathbf{C},f')=\mathbf{A}_{f'}\in\mathbb{Z}_q^{n\times\ell}$*, where the distribution of* $\mathbf{A}_{f'}$ *is marginally random given random input* $\mathbf{A}_0\leftarrow\mathbb{Z}_q^{n\times\ell}$*.*
- $\mathsf{Evalfx}(\mathbf{A}_0,\mathbf{A},\mathbf{C},f',\mathbf{x})=\mathbf{H}_{f',\mathbf{x}}\in\mathbb{Z}^{(n+k)m\times\ell}$*, where* $\mathbf{H}_{f',\mathbf{x}}$ *has bounded norm* $\|\mathbf{H}_{f',\mathbf{x}}\|=m^{O(d)}$*, and* $\mathbf{H}_{f',\mathbf{x}}=0 \pmod 2$*.*
- $[\mathbf{A}_0|\mathbf{A}-\mathbf{x}^\top\otimes\mathbf{G}|\mathbf{C}-\mathbf{r}^\top\otimes\mathbf{G}]\cdot\mathbf{H}_{f',\mathbf{x}}=\mathbf{A}_{f'}-\mathbf{Y}$*, where* $\mathbf{Y}=\begin{pmatrix}\mathbf{0}^{n\times\ell}\\ (\mathbf{r}^\top,-1)\cdot f(\mathbf{x})\end{pmatrix}$*.*

We now move to the correlation-inducing gates for the GSW encryption scheme, intending to achieve the correlation $\mathsf{PRF}(\mathbf{k})=\mathbf{e}_{\mathsf{HE}}^\top\mathbf{R}_{\mathsf{PRF}} \pmod 2$.

Observation 3 (Correlation-friendly GSW settings)**.** *We focus on GSW encryptions with settings friendly to our correlation inducing gates. We require the dimension n and modulus q to both be odd integers. For the public key, we require the error* $\mathbf{e}_{\mathsf{HE}}$ *to be odd, i.e.,*

$$\mathsf{hpk}=\overline{\mathbf{B}}=\begin{pmatrix}\mathbf{B}_{\mathsf{HE}}\\ \mathbf{r}^\top\mathbf{B}_{\mathsf{HE}}+\mathbf{e}_{\mathsf{HE}}^\top\end{pmatrix},\quad\text{where }\mathbf{B}\leftarrow\mathbb{Z}_q^{n\times m},\mathbf{r}\leftarrow\mathbb{Z}_q^n,\mathbf{e}_{\mathsf{HE}}\leftarrow\mathcal{D}_\sigma^m|_{\mathbf{e}_{\mathsf{HE}}=0\bmod 2}.$$

Note that the conditional discrete Gaussian distribution $e \leftarrow \mathcal{D}_\sigma|_{e=0 \bmod 2}$ is efficiently sampleable by rejection sampling.

For (bit) encryptions, we require the encryption randomness $\mathbf{R}$ *to also be odd. Namely*

$$\mathsf{hct}(b) = \overline{\mathbf{B}}\mathbf{R} + b\mathbf{G}, \quad \text{where } \mathbf{R} \leftarrow \{\pm 1\}^{m\times(n+1)\lceil \log q\rceil}$$

The correlation-friendly setting ensures $\mathbf{e}^\intercal \mathbf{R}_f$ *is even/odd whenever* $\mathbf{R}_f$ *is. Note that the GSW HE scheme with this setting is provably secure from standard LWE.*

Using the ideas from [HJL21], we can provide correlation-inducing gates which generate $\mathbf{R}_f$ that is always even or has parity consistent with the encrypted value.

Observation 4 (Even-error gate for GSW). *For any (not necessarily fresh) bit ciphertext* $\mathsf{hct}(b) = \overline{\mathbf{B}}\mathbf{R} + b\mathbf{G}$, *homomorphically evaluating the function* $h(b) = 2^{-1} \cdot b + 2^{-1} \cdot b$ *result in ciphertext* $\mathsf{hct}'(b) = \overline{\mathbf{B}}\mathbf{R}_h + b\mathbf{G}$, *where* $\|\mathbf{R}_h\| = \left\|\mathbf{R} \cdot 2\mathbf{G}^{-1}(2^{-1}\mathbf{G})\right\| \le 2m\|\mathbf{R}\|$, *and* $\mathbf{R}_h$ *is even.*

Observation 5 (Multiplication by 1 gate for GSW). *For any (not necessarily fresh) bit ciphertext* $\mathsf{hct}(b) = \overline{\mathbf{B}}\mathbf{R} + b\mathbf{G}$ *with even randomness* $\mathbf{R} = 0 \pmod 2$, *homomorphically evaluating the function* $h(b) = b \cdot 1$ *with a fresh ciphertext of 1* $\mathsf{hct}_1 = \overline{\mathbf{B}}\mathbf{R}_1 + b\mathbf{G}$ *result in ciphertext* $\mathsf{hct}'(b) = \overline{\mathbf{B}}\mathbf{R}_h + b\mathbf{G}$ *where* $\|\mathbf{R}_h\| = \left\|\mathbf{R} \cdot \mathbf{G}^{-1}(\mathsf{hct}_1) + b\mathbf{R}_1\right\| \le m\|\mathbf{R}\| + 1$, *and* $\mathbf{R}_h = b \pmod 2$.

Finally, we show that the correlation can be preserved through the packing operation.

Observation 6 (GSW packing). *For* $v \in \mathbb{Z}_q$ *and its bitwise representation* $v = \sum_i 2^i v_i$, *given ciphertexts*

- $\mathsf{hct}_0 = \overline{\mathbf{B}}\mathbf{R}_0 + v_0\mathbf{G}$, *where* $\mathbf{R}_0 = v_0 \pmod 2$.
- *For all* $i \in [\lceil \log q\rceil - 1]$, $\mathsf{hct}_i = \overline{\mathbf{B}}\mathbf{R}_i + v_i\mathbf{G}$, *where* $\mathbf{R}_i = 0 \pmod 2$.

The packing operation admits

$$\mathsf{Pack}(\{\mathsf{hct}_i\}) = \sum_i \mathsf{hct}_i \mathbf{G}^{-1}(\mathbf{1}_{n+1}) = \overline{\mathbf{B}}\mathbf{r}_{\mathsf{Pack}} + \begin{pmatrix}\mathbf{0}^n \\ v\end{pmatrix}, \quad \text{where } \mathbf{r}_{\mathsf{Pack}} = (\sum \mathbf{R}_i)\mathbf{G}^{-1}(\mathbf{1}_{n+1}).$$

Note that $\sum \mathbf{R}_i = v_0 \pmod 2$, *and that* $\mathbf{G}^{-1}(\mathbf{1}_{n+1})$ *is a unit vector. Therefore* $\mathbf{r}_{\mathsf{Pack}} = v_0 \pmod 2$, *implying that the decryption error* $\mathbf{e}_{\mathsf{HE}}^\intercal \mathbf{r}_{\mathsf{Pack}} = v \pmod 2$. *Since the packing operation for vector* $\mathbf{v} \in \mathbb{Z}_q^\ell$ *is a columnwise concatenation of packing operations of each of its entries, the parity relation directly extends to packing vectors.*

Combining the above observations, we get the following lemma for GSW homomorphic evaluations over packed circuits.

Lemma 23 (GSW homomorphic evaluations with bad circuit implementation). *Under GSW parameter settings following observation 3, for every packed-circuit* $f : \{0,1\}^k \to \mathbb{Z}_q^\ell \in \mathcal{F}_{d,\ell}$, *there exists an equivalent circuit implementation* f' *with constant blow up in depth such that the homomorphic evaluation satisfies*

$$\mathsf{Eval}(\mathsf{hct}(\mathbf{x}), f') = \overline{\mathbf{B}}\mathbf{R}_{f'} \boxplus (f'(\mathbf{x})) = \begin{pmatrix}\mathbf{B}\mathbf{R}_{f'} \\ \mathbf{r}^\intercal\mathbf{B}\mathbf{R}_{f'} + \mathbf{e}_{\mathsf{HE}}^\intercal\mathbf{R}_{f'} + f'(\mathbf{x})^\intercal\end{pmatrix}, \quad \text{where } \mathbf{e}_{\mathsf{HE}}^\intercal\mathbf{R}_{f'} = f'(\mathbf{x})^\intercal \pmod 2$$

Finally, we give the following lemma which shows that the subspace structure we detect from the decryption outcome

$$\begin{aligned}\mathbf{y}^{\intercal} &= (\mathbf{s}^{\intercal}\mathbf{A}_f + \mathsf{PRF}(\mathbf{k})^{\intercal} + \widehat{\mathbf{e}}^{\intercal}_{\mathsf{HE}} + \widehat{\mathbf{e}}^{\intercal}_{\mathsf{PE}}) - (\mathbf{s}^{\mathbf{T}}\mathbf{B} + \mathbf{e}^{\intercal}_{\mathbf{B}})\mathbf{T} \bmod 2 \\ &= \mathsf{PRF}(\mathbf{k})^{\intercal} + \mathbf{e}^{\intercal}_{\mathsf{HE}}\mathbf{R}_{\mathsf{PRF}} + \mathbf{e}^{\intercal}_{\mathsf{PE}}\mathbf{H}_{f,\mathbf{x}} - \mathbf{e}^{\intercal}_{\mathbf{B}}\mathbf{T} \bmod 2 \\ &= \mathbf{e}^{\intercal}_{\mathbf{B}}\mathbf{T} \bmod 2 \in \mathsf{span}(\mathbf{T} \bmod 2)\end{aligned}$$

no longer exists when the LWE sample $(\mathbf{s}^{\mathbf{T}}\mathbf{B} + \mathbf{e}^{\intercal}_{\mathbf{B}})$ is replaced with a uniformly random vector. This gives a concrete distinguisher for the evasive LWE postcondition.

Lemma 24. *Let q, m, m_T, σ_P be integers, where q is odd, $m_T \geq \lambda m$, and $q \geq 2^{\lambda}\sigma_P$. Let $(\cdot)_q$ denote the operation of parsing a vector in $\mathbb{Z}_q$ into an integer vector with entries in range $[0, q-1]$. Let $\mathcal{V}$ be a distribution over vectors $\mathbf{v} \in \mathbb{Z}_q^{m_T}$ such that*

- $\mathbf{v}$ *is even with overwhelming probability, i.e.,*

$$\Pr[\mathbf{v} = 0 \bmod 2 | \mathbf{v} \leftarrow \mathcal{V}] \geq 1 - \mathsf{negl}(\lambda).$$

- $\mathbf{v}$ *is pseudorandom when added with a small error. Formally,*

$$(\mathbf{v} + \mathbf{e}|\ \mathbf{v} \leftarrow \mathcal{V}, \mathbf{e} \leftarrow \mathcal{D}_{\sigma_P}^{m_T}) \approx_c (\mathbf{u} \leftarrow \mathbb{Z}_q^{m_T})$$

Then, it holds that

$$\Pr\left[(\mathbf{c}^{\intercal}\mathbf{T} - \mathbf{v}^{\intercal})_q \bmod 2 \in \mathsf{span}(\mathbf{T} \bmod 2) \,\middle|\, \mathbf{c} \leftarrow \mathbb{Z}_q^m, \mathbf{T} \leftarrow \mathcal{D}_{\sigma_T}^{m \times m_T}, \mathbf{v} \leftarrow \mathcal{V}\right] \leq \mathsf{negl}(\lambda)$$

Proof. Observe that

$$(\mathbf{c}^{\intercal}\mathbf{T} - \mathbf{v}^{\intercal})_q \bmod 2 = (\mathbf{c}^{\intercal}\mathbf{T})_q - \mathbf{v}^{\intercal} + q\left\lfloor \frac{(\mathbf{c}^{\intercal}\mathbf{T})_q - \mathbf{v}^{\intercal}}{q} \right\rfloor \bmod 2 = (\mathbf{c}^{\intercal}\mathbf{T})_q + \left\lfloor \frac{(\mathbf{c}^{\intercal}\mathbf{T})_q - \mathbf{v}^{\intercal}}{q} \right\rfloor \bmod 2 \quad (24)$$

Since $q \geq 2^{\kappa}\sigma_{\mathbf{P}}$ and the distribution of $(\mathbf{c}^{\intercal}\mathbf{T})_q$ is marginally random in $\mathbb{Z}_q$, with overwhelming probability it holds that

$$\left\lfloor \frac{(\mathbf{c}^{\intercal}\mathbf{T})_q - \mathbf{v}^{\intercal}}{q} \right\rfloor = \left\lfloor \frac{(\mathbf{c}^{\intercal}\mathbf{T})_q - \mathbf{v}^{\intercal} - \mathbf{e}^{\intercal}}{q} \right\rfloor, \quad \text{where } \mathbf{e} \leftarrow \mathcal{D}_{\sigma_P}^{m}. \quad (25)$$

Now, observe that whether or not a vector falls in the span of ($\mathbf{T}$ mod 2) is efficiently checkable. Therefore, by the pseudorandomness of $\mathbf{v} + \mathbf{e}$ we have

$$\left| \begin{array}{l} \Pr\left[(\mathbf{c}^{\intercal}\mathbf{T})_q + \left\lfloor \frac{(\mathbf{c}^{\intercal}\mathbf{T})_q - \mathbf{v}^{\intercal} - \mathbf{e}^{\intercal}}{q} \right\rfloor \in \mathsf{span}((\mathbf{T} \bmod 2))\right] \\ - \Pr_{\mathbf{u} \leftarrow \mathbb{Z}_q^{m_P}}\left[(\mathbf{c}^{\intercal}\mathbf{T})_q + \left\lfloor \frac{(\mathbf{c}^{\intercal}\mathbf{T})_q - \mathbf{u}^{\intercal}}{q} \right\rfloor \in \mathsf{span}((\mathbf{T} \bmod 2))\right] \end{array} \right| \leq \mathsf{negl}(\lambda). \quad (26)$$

where we can bound the second probability for most $\mathbf{t}^{\intercal} = (\mathbf{c}^{\intercal}\mathbf{T})_q$ by observing that

- Since $\mathbf{t}$ is (marginally) random in $[0, q-1]$, with overwhelming probability over $\mathbf{t}$, there exists 1/3 fraction of indices such that $t_i \in (\frac{1}{4}q, \frac{3}{4}q)$. Denote the index set as S_t

- For every binary vector $\mathbf{w} \in \{0,1\}^{m_P}$,

$$\Pr_{\mathbf{u}}\left[\forall i \in S_t,\ \left(t_i + \left\lfloor \frac{t_i - u_i}{q} \right\rfloor\right)_i = w_i\right] \leq (3/4)^{|S_t|}.$$

Therefore, by a union bound over the 2^m vectors in the row span of ($\mathbf{T}$ mod 2), we have

$$\Pr_{\mathbf{u}\leftarrow\mathbb{Z}_q^{m_P}}\left[(\mathbf{c}^\intercal\mathbf{T})_q + \left\lfloor \frac{(\mathbf{c}^\intercal\mathbf{T})_q - \mathbf{u}^\intercal}{q} \right\rfloor \in \mathsf{span}((\mathbf{T} \bmod 2))\right] \leq 2^m \cdot (3/4)^{1/3m_T} + \mathsf{negl} = \mathsf{negl}. \tag{27}$$

Combining equation (24) to (27), we conclude that

$$\Pr\left[(\mathbf{c}^\intercal\mathbf{T} - \mathbf{v}^\intercal)_q \bmod 2 \in \mathsf{span}(\mathbf{T} \bmod 2)\right] \leq \mathsf{negl}(\lambda)$$

□

Counterexample Sampler We now formally describe the sampler corresponding to the counterexample construction.

$\underline{\mathsf{Samp}(1^\lambda)}$:

Let λ be the security parameter, all other parameters depend on λ. Let n, q be odd integers, m, m_P be integers such that $m = \Theta(n \log q)$ and $m_P \geq \lambda m$. Let $\mathsf{PRF} : \{0,1\}^k \times \{0,1\}^K \to \mathbb{Z}_q^\ell$ be a matrix-value pseudorandom function where each output bit of the function can be computed by depth $d_0 = O(\log\lambda)$ circuits, and let $d = \mathsf{poly}\log\lambda$ be the circuit depth of the GSW homomorphic evaluation circuit $\mathsf{GSW.Eval}(\cdot, \mathsf{PRF}(\cdot, 0^K))$. Let $\sigma, \sigma_B, \sigma_P, \sigma_T$ be Gaussian width parameter such that LWE with parameter (n, q, σ) is secure, $\sigma = \sigma_B = \sigma_T = \mathsf{poly}(\lambda)$, $\sigma_P = 2^\lambda m^{\Omega(d)}\sigma$, and $q \geq 2^\lambda \sigma_P$. The sampler computes the following components.

- Sample PRF seed $\mathbf{k} \leftarrow \{0,1\}^\lambda$.
- Sample correlation-friendly GSW public key $\mathsf{hpk} = \overline{\mathbf{B}} = \begin{pmatrix} \mathbf{B}_{\mathsf{HE}} \\ \mathbf{r}^\intercal\mathbf{B}_{\mathsf{HE}} + \mathbf{e}_{\mathsf{HE}}^\intercal \end{pmatrix}$, where $\mathbf{B}_{\mathsf{HE}} \leftarrow \mathbb{Z}_q^{n\times m}$, $\mathbf{r} \leftarrow \mathbb{Z}_q^n$, and $\mathbf{e}_{\mathsf{HE}} \leftarrow \mathcal{D}_\sigma^m|_{\mathbf{e}_{\mathsf{HE}}=0 \bmod 2}$ is sampled as odd error.
- Compute correlation-friendly GSW ciphertext of seed $\mathbf{k}$ by as $\mathsf{ct} = \mathbf{DR} + \mathbf{k} \otimes \mathbf{G}$.

$$\mathsf{hct} = \overline{\mathbf{B}}\mathbf{R} + \mathbf{k}^\intercal \otimes \mathbf{G}, \quad \text{where } \mathbf{R} \leftarrow \{\pm1\}^{m\times\lambda(n+1)\lceil\log q\rceil}$$

- Set the GVW attribute $\mathbf{x} = \mathsf{bits}(\mathsf{hct}) \in \{0,1\}^{\lambda(n+1)^2\lceil\log q\rceil^2}$ to be the bitwise representation of the GSW ciphertext of the PRF seed, and policy $f = \mathsf{GSW.Eval}(\cdot, \mathsf{PRF}(\cdot, 0^K))$ be the GSW homomorphic evaluation function, where the circuit implementations of PRF and GSW.Eval are chosen as the correlation-inducing implementation introduced in lemma 22 and 23.
- Sample GVW public matrices $\mathbf{A}_0 \leftarrow \mathbb{Z}_q^{(n+1)\times\ell}$,$\mathbf{A} \leftarrow \mathbb{Z}_q^{(n+1)\times|\mathbf{x}|(n+1)\lceil\log q\rceil}$, $\mathbf{C} \leftarrow \mathbb{Z}_q^{n\times n(n+1)\lceil\log q\rceil}$.
- Compute GVW encodings $\overline{\mathsf{ct}} = \mathbf{s}^\intercal[\mathbf{A}_0|\mathbf{A} - \mathbf{x}^\intercal \otimes \mathbf{G}|\mathbf{C} - \mathbf{r}^\intercal \otimes \mathbf{G}] + \mathbf{e}_{\mathsf{PE}}^\intercal$, where $\mathbf{s}^\intercal = (\mathbf{t}^\intercal, 1)$, $\mathbf{t} \leftarrow \mathbb{Z}_q^n$, and $\mathbf{e}_{\mathsf{PE}} \leftarrow \mathcal{D}_\sigma^{m_{\mathsf{PE}}}$ with dimension $m_{\mathsf{PE}} = \ell + |\mathbf{x}|(n+1)\lceil\log q\rceil + n(n+1)\lceil\log q\rceil$.
- Set $\mathbf{P} = \mathbf{A}_f = \mathsf{Evalf}(\mathbf{A}_0, \mathbf{A}, \mathbf{C}, f)$ to be the decryption pad.
- Set $\mathsf{aux} = (\mathsf{hpk}, \mathsf{hct}, \mathbf{A}_0, \mathbf{A}, \mathbf{C}, \overline{\mathsf{ct}})$

- Output $(\mathbf{s}, \mathbf{P}, \mathsf{aux})$

We show that the above sampler is indeed an evasive LWE counterexample by proving that the precondition holds while the postcondition does not hold.

Lemma 25 (Precondition)**.** *For the set of parameters specified by* Samp*, assuming the LWE assumption, it holds that*

$$(\mathbf{B}, \mathbf{P}, \mathbf{s}^\intercal\mathbf{B} + \mathbf{e}_\mathbf{B}^\intercal, \mathbf{s}^\intercal\mathbf{P} + \mathbf{e}_\mathbf{P}^\intercal, \mathsf{aux}) \approx (\mathbf{B}, \mathbf{P}, \mathbf{c}_\mathbf{B}^\intercal, \mathbf{c}_\mathbf{P}^\intercal, \mathsf{aux})$$

where $(\mathbf{s}, \mathbf{P}, \mathsf{aux}) \leftarrow \mathsf{Samp}(1^\lambda)$, $\mathbf{B} \leftarrow \mathbb{Z}_q^{n\times m}$, $\mathbf{e_B} \leftarrow \mathcal{D}_{\sigma_B}^m$, $\mathbf{e_P} \leftarrow \mathcal{D}_{\sigma_P}^{m_P}$, $\mathbf{c_B} \leftarrow \mathbb{Z}_q^m$, $\mathbf{c_P} \leftarrow \mathbb{Z}_q^{m_P}$.

Proof. The proof follows the standard techniques from existing works. We sketch the sequence of hybrids as follows.

- $\mathcal{H}_0$ is the left distribution.
- $\mathcal{H}_1$ replaces $\mathbf{s}^\intercal\mathbf{P} + \mathbf{e}_\mathbf{P}^\intercal$ by

  $$\overline{\mathsf{ct}} \cdot \mathbf{H}_{f,\mathbf{x}} + \mathbf{e}_\mathbf{P}^\intercal - \mathsf{PRF}(\mathbf{k}, 0^K),$$

  where $\mathbf{H}_{f,\mathbf{x}} \leftarrow \mathsf{Evalfx}(\mathbf{A}_0, \mathbf{A}, \mathbf{C}, f, \mathbf{x} = \mathsf{bits}(\mathsf{hct}))$. Observe that the replacement satisfies

  $$\overline{\mathsf{ct}} \cdot \mathbf{H}_{f,\mathbf{x}} + \mathbf{e}_\mathbf{P}^\intercal - \mathsf{PRF}(\mathbf{k}, 0^K) = \mathbf{s}^\intercal\mathbf{P} + \mathbf{e}_\mathbf{P}^\intercal + \mathbf{e}_{\mathsf{HE}}^\intercal\mathbf{R}_{\mathsf{PRF}} + \mathbf{e}_{\mathsf{PE}}^\intercal\mathbf{H}_{f,\mathbf{x}},$$

  where $\left\|\mathbf{e}_{\mathsf{HE}}^\intercal\mathbf{R}_{\mathsf{PRF}} + \mathbf{e}_{\mathsf{PE}}^\intercal\mathbf{H}_{f,\mathbf{x}}\right\| \leq m^{O(d)}\sigma < \sigma_\mathbf{P}/2^\lambda$. Therefore, $\mathcal{H}_0 \approx_s \mathcal{H}_1$ follows from the smudging argument (lemma 3).
- $\mathcal{H}_2$ samples matrix $\mathbf{A}' \leftarrow \mathbb{Z}_q^{(n+1)\times|\mathbf{x}|(n+1)\lceil\log q\rceil}$,$\mathbf{C}' \leftarrow \mathbb{Z}_q^{n\times n(n+1)\lceil\log q\rceil}$, and program $[\mathbf{A}|\mathbf{C}] = [\mathbf{A}' + \mathbf{x}^\intercal \otimes \mathbf{G} | \mathbf{C} + \mathbf{r}^\intercal \otimes \mathbf{G}]$. The GVW encoding is then computed by $\overline{\mathsf{ct}} = \mathbf{s}^\intercal[\mathbf{A}_0|\mathbf{A}'|\mathbf{C}'] + \mathbf{e}_{\mathsf{PE}}^\intercal$. $\mathcal{H}_1$ and $\mathcal{H}_2$ are identical.
- $\mathcal{H}_3$ switches the LWE samples $\mathbf{s}^\intercal\mathbf{B} + \mathbf{e}_\mathbf{B}^\intercal$ and $\overline{\mathsf{ct}} = \mathbf{s}^\intercal[\mathbf{A}_0|\mathbf{A}'|\mathbf{C}'] + \mathbf{e}_{\mathsf{PE}}^\intercal$ to random. Note that these two components are the only components that (directly) depend on $\mathbf{s}$, thus by LWE, $\mathcal{H}_2 \approx_c \mathcal{H}_3$.
- $\mathcal{H}_4$ switches back to sampling $[\mathbf{A}|\mathbf{C}]$ by random. $\mathcal{H}_3$ and $\mathcal{H}_4$ are identical.
- $\mathcal{H}_5$ samples $\mathsf{hpk}, \mathsf{hct}$ by random. Assuming LWE, the pseudorandom public key and ciphertext property for the GSW HE scheme implies $\mathcal{H}_4 \approx_c \mathcal{H}_5$.
- $\mathcal{H}_6$ replaces the term $\overline{\mathsf{ct}} \cdot \mathbf{H}_{f,\mathbf{x}} + \mathbf{e}_\mathbf{P}^\intercal - \mathsf{PRF}(\mathbf{k}, 0^K)$ by random. Since hct no longer depends on $\mathbf{k}$, $\mathcal{H}_5 \approx \mathcal{H}_6$ follows from the PRF security.
- $\mathcal{H}_7$ switch back the sampling of aux according to Samp. $\mathcal{H}_6 \approx_c \mathcal{H}_7$ follows from reverting the changes made by $\mathcal{H}_5$ to $\mathcal{H}_2$, and the indistinguishability argument is identical. $\mathcal{H}_7$ is the right distribution.

□

Lemma 26 (Postcondition)**.** *For the set of parameters specified by* Samp*, there exists an efficient distinguisher* D *with non-negligible advantage* ϵ *such that*

$$\left|\Pr[D(\mathbf{B}, \mathbf{P}, \mathbf{s}^\intercal\mathbf{B} + \mathbf{e}_\mathbf{B}^\intercal, \mathbf{T} = \mathbf{B}^{-1}(\mathbf{P}), \mathsf{aux}) = 1] - \Pr[D(\mathbf{B}, \mathbf{P}, \mathbf{c}_\mathbf{B}^\intercal, \mathbf{T} = \mathbf{B}^{-1}(\mathbf{P}), \mathsf{aux}) = 1]\right| \geq \epsilon(\lambda).$$

where $(\mathbf{s}, \mathbf{P}, \mathsf{aux}) \leftarrow \mathsf{Samp}(1^\lambda)$, $\mathbf{B} \leftarrow \mathbb{Z}_q^{n\times m}$, $\mathbf{e_B} \leftarrow \mathcal{D}_{\sigma_B}^m$, $\mathbf{e_P} \leftarrow \mathcal{D}_{\sigma_P}^{m_P}$, $\mathbf{c_B} \leftarrow \mathbb{Z}_q^m$, $\mathbf{c_P} \leftarrow \mathbb{Z}_q^{m_P}$.

Proof. We start by constructing the following distinguisher D as follows.
$\underline{D(\mathbf{B}, \mathbf{P}, \mathbf{c}_{\mathbf{B}}, \mathbf{T}, \mathsf{aux} = (\mathsf{hpk}, \mathsf{hct}, \mathbf{A}_0, \mathbf{A}, \mathbf{C}, \overline{\mathsf{ct}}))\ :}$

- Compute $\mathbf{y}^\top = \overline{\mathsf{ct}}\mathbf{H}_{f,\mathbf{x}} - \mathbf{c}_{\mathbf{B}}^\top \mathbf{T}$, where $\mathbf{H}_{f,\mathbf{x}} \leftarrow \mathsf{Evalfx}(\mathbf{A}_0, \mathbf{A}, \mathbf{C}, f, \mathbf{x})$.
- Test whether $(\mathbf{y}^\top)_q \bmod 2 \in \mathsf{span}(\mathbf{T} \bmod 2)$. Output 1 if true and 0 otherwise.

On input the left distribution, where $\mathbf{c}_{\mathbf{B}}^\top = \mathbf{s}^\top\mathbf{B} + \mathbf{e}_{\mathbf{B}}^\top$, by the correctness of the GVW encodings,

$$\begin{aligned}\mathbf{y}^\top &= \overline{\mathsf{ct}}\mathbf{H}_{f,\mathbf{x}} - (\mathbf{s}^\top\mathbf{B} + \mathbf{e}_{\mathbf{B}}^\top)\mathbf{T} \\ &= \mathbf{s}^\top\mathbf{A}_f + \mathsf{PRF}(\mathbf{k})^\top + \mathbf{e}_{\mathsf{HE}}^\top\mathbf{R}_{\mathsf{PRF}} + \mathbf{e}_{\mathsf{PE}}^\top\mathbf{H}_{f,\mathbf{x}} - \mathbf{s}^\top\mathbf{A}_f - \mathbf{e}_{\mathbf{B}}^\top\mathbf{T} \\ &= \mathsf{PRF}(\mathbf{k})^\top + \mathbf{e}_{\mathsf{HE}}^\top\mathbf{R}_{\mathsf{PRF}} + \mathbf{e}_{\mathsf{PE}}^\top\mathbf{H}_{f,\mathbf{x}}\mathbf{e}_{\mathbf{B}}^\top\mathbf{T}.\end{aligned}$$

Therefore,

$$\begin{aligned}(\mathbf{y}^\top)_q \bmod 2 &= (\mathsf{PRF}(\mathbf{k})^\top + \mathbf{e}_{\mathsf{HE}}^\top\mathbf{R}_{\mathsf{PRF}} + \mathbf{e}_{\mathsf{PE}}^\top\mathbf{H}_{f,\mathbf{x}}\mathbf{e}_{\mathbf{B}}^\top\mathbf{T})_q \bmod 2 \\ &\overset{\text{w.h.p.}}{=} \mathsf{PRF}(\mathbf{k})^\top + \mathbf{e}_{\mathsf{HE}}^\top\mathbf{R}_{\mathsf{PRF}} + \mathbf{e}_{\mathsf{PE}}^\top\mathbf{H}_{f,\mathbf{x}}\mathbf{e}_{\mathbf{B}}^\top\mathbf{T} \bmod 2 \\ &= \mathbf{e}_{\mathbf{B}}^\top\mathbf{T} \bmod 2 \in \mathsf{span}(\mathbf{T} \bmod 2).\end{aligned}$$

The second equality follows from the fact that $\mathsf{PRF}(\mathbf{k})$ is pseudorandom in $\mathbb{Z}_q$, therefore it is far from the rounding boundary of q with overwhelming probability. The last equality follows from the error correlation demonstrated by lemma 22 and 23. Therefore the distinguisher outputs 1 with overwhelming probability.

On input the right distribution, where $\mathbf{c}_{\mathbf{B}}$ is sampled by random, by definition

$$\mathbf{y}^\top = \overline{\mathsf{ct}}\mathbf{H}_{f,\mathbf{x}} - \mathbf{c}_{\mathbf{B}}\mathbf{T} = (\mathbf{s}^\top\mathbf{B} - \mathbf{c}_{\mathbf{B}}^\top)\mathbf{T} + \mathsf{PRF}(\mathbf{k})^\top + \mathbf{e}_{\mathsf{HE}}^\top\mathbf{R}_{\mathsf{PRF}} + \mathbf{e}_{\mathsf{PE}}^\top\mathbf{H}_{f,\mathbf{x}}.$$

Observe that $\mathbf{c}^\top = (\mathbf{s}^\top\mathbf{B} - \mathbf{c}_{\mathbf{B}}^\top)$ is random in $\mathbb{Z}_q^m$, $\mathbf{T} \leftarrow \mathbf{B}^{-1}(\mathbf{A}_f)$ marginally distributes statistically close to discrete Gaussian of width σ_T since $\mathbf{A}_f$ is marginally random, and $\mathbf{v}^\top = \mathsf{PRF}(\mathbf{k})^\top + \mathbf{e}_{\mathsf{HE}}^\top\mathbf{R}_{\mathsf{PRF}} + \mathbf{e}_{\mathsf{PE}}^\top\mathbf{H}_{f,\mathbf{x}}$ is pseudorandom when added with a small error $\mathbf{e}_{\mathbf{P}} \leftarrow \mathcal{D}_{\sigma_P}^{m_P}$, as observed in the the proof of lemma 25. Therefore immediately by lemma 24, we know that the probability that $((\mathbf{y}^\top)_q \bmod 2)$ falls in the row span of $(\mathbf{T} \bmod 2)$ is negligible[11]. Therefore the distinguisher outputs 1 with negligible probability.

Combining all the above, we conclude that D has a overwhelming advantage on distinguishing the postcondition of the evasive LWE assumption. □

Remark 4. *We note that the given sampler is a* private-coin *sampler, where the PRF seed* $\mathbf{k}$ *and the GSW secret/randomness* $\mathbf{r}, \mathbf{e}_{\mathsf{HE}}, \mathbf{R}$ *needs to be kept private even given the auxiliary information* $\mathsf{aux} = (\mathsf{hpk}, \mathsf{hct}, \mathbf{A}_0, \mathbf{A}, \mathbf{C}, \overline{\mathsf{ct}})$. *It is an interesting open question whether or not one can construct a public-coin sampler, where all random coins of its sampling procedure are revealed in* aux, *such that it still admits a counterexample for evasive LWE.*

Acknowledgment The authors would like to thank Hoeteck Wee for many insightful discussions about evasive LWE assumptions, and attacks on assumptions underlying lattice-based iO and PrO candidates.

Yao Ching Hsieh and Huijia Lin were supported by NSF grant CNS-2026774, and a Simons Collaboration on the Theory of Algorithmic Fairness.

Aayush Jain was supported by a Google Faculty Research Scholar 2023, a Stellar Foundation Grant, CYLAB of CMU, and an NSF CAREER CNS-2441647.

[11] Technically, the lemma cannot be directly applied since the distribution of $\mathbf{v}$ contains errors that are not independent of $\mathbf{T}$. Nevertheless, this can be easily solved by either by having the distinguisher add a (even) smudging noise, or by observing that the rounding-style analysis in the proof of lemma 24 is invariant to any small errors.

References

[AGIS14] Prabhanjan Vijendra Ananth, Divya Gupta, Yuval Ishai, and Amit Sahai. Optimizing obfuscation: Avoiding Barrington's theorem. In Gail-Joon Ahn, Moti Yung, and Ninghui Li, editors, *ACM CCS 2014*, pages 646–658. ACM Press, November 2014.

[Agr19] Shweta Agrawal. Indistinguishability obfuscation without multilinear maps: New methods for bootstrapping and instantiation. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019*, volume 11476 of *LNCS*, pages 191–225, May 2019.

[AJL$^+$19] Prabhanjan Ananth, Aayush Jain, Huijia Lin, Christian Matt, and Amit Sahai. Indistinguishability obfuscation without multilinear maps: New paradigms via low degree weak pseudorandomness and security amplification. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019*, volume 11694 of *LNCS*, pages 284–332, August 2019.

[AJS18] Prabhanjan Ananth, Aayush Jain, and Amit Sahai. Indistinguishability obfuscation without multilinear maps: iO from LWE, bilinear maps, and weak pseudorandomness. Cryptology ePrint Archive, Report 2018/615, 2018.

[AKY24] Shweta Agrawal, Simran Kumari, and Shota Yamada. Pseudorandom multi-input functional encryption and applications. Cryptology ePrint Archive, Paper 2024/1720, 2024.

[AP20] Shweta Agrawal and Alice Pellet-Mary. Indistinguishability obfuscation without maps: Attacks and fixes for noisy linear FE. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020*, volume 12105 of *LNCS*, pages 110–140, May 2020.

[ARYY23] Shweta Agrawal, Mélissa Rossi, Anshu Yadav, and Shota Yamada. Constant input attribute based (and predicate) encryption from evasive and tensor LWE. In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023*, volume 14084 of *LNCS*, pages 532–564, August 2023.

[AS17] Prabhanjan Ananth and Amit Sahai. Projective arithmetic functional encryption and indistinguishability obfuscation from degree-5 multilinear maps. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT 2017*, volume 10210 of *LNCS*, pages 152–181, April / May 2017.

[BDGM20] Zvika Brakerski, Nico Döttling, Sanjam Garg, and Giulio Malavolta. Candidate iO from homomorphic encryption schemes. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020*, volume 12105 of *LNCS*, pages 79–109, May 2020.

[BDGM22] Zvika Brakerski, Nico Döttling, Sanjam Garg, and Giulio Malavolta. Factoring and pairings are not necessary for IO: Circular-secure LWE suffices. In Mikolaj Bojanczyk, Emanuela Merelli, and David P. Woodruff, editors, *ICALP 2022*, volume 229 of *LIPIcs*, pages 28:1–28:20. Schloss Dagstuhl, July 2022.

[BDJ$^+$24] Pedro Branco, Nico Döttling, Abhishek Jain, Giulio Malavolta, Surya Mathialagan, Spencer Peters, and Vinod Vaikuntanathan. Pseudorandom obfuscation and applications. Cryptology ePrint Archive, Paper 2024/1742, 2024.

[BFM14] Christina Brzuska, Pooya Farshim, and Arno Mittelbach. Indistinguishability obfuscation and UCEs: The case of computationally unpredictable sources. In Juan A. Garay and Rosario Gennaro, editors, *CRYPTO 2014*, volume 8616 of *LNCS*, pages 188–205, August 2014.

[BGG+14] Dan Boneh, Craig Gentry, Sergey Gorbunov, Shai Halevi, Valeria Nikolaenko, Gil Segev, Vinod Vaikuntanathan, and Dhinakaran Vinayagamurthy. Fully key-homomorphic encryption, arithmetic circuit ABE and compact garbled circuits. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 533–556, May 2014.

[BGH+15] Zvika Brakerski, Craig Gentry, Shai Halevi, Tancrède Lepoint, Amit Sahai, and Mehdi Tibouchi. Cryptanalysis of the quadratic zero-testing of GGH. Cryptology ePrint Archive, Report 2015/845, 2015.

[BGI+01] Boaz Barak, Oded Goldreich, Russell Impagliazzo, Steven Rudich, Amit Sahai, Salil P. Vadhan, and Ke Yang. On the (im)possibility of obfuscating programs. In Joe Kilian, editor, *CRYPTO 2001*, volume 2139 of *LNCS*, pages 1–18, August 2001.

[BGK+14] Boaz Barak, Sanjam Garg, Yael Tauman Kalai, Omer Paneth, and Amit Sahai. Protecting obfuscation against algebraic attacks. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 221–238, May 2014.

[BHJ+19] Boaz Barak, Samuel B. Hopkins, Aayush Jain, Pravesh Kothari, and Amit Sahai. Sum-of-squares meets program obfuscation, revisited. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019*, volume 11476 of *LNCS*, pages 226–250, May 2019.

[BIJ+20] James Bartusek, Yuval Ishai, Aayush Jain, Fermi Ma, Amit Sahai, and Mark Zhandry. Affine determinant programs: A framework for obfuscation and witness encryption. In Thomas Vidick, editor, *ITCS 2020*, volume 151, pages 82:1–82:39. LIPIcs, January 2020.

[BLP+13] Zvika Brakerski, Adeline Langlois, Chris Peikert, Oded Regev, and Damien Stehlé. Classical hardness of learning with errors. In Dan Boneh, Tim Roughgarden, and Joan Feigenbaum, editors, *45th ACM STOC*, pages 575–584. ACM Press, June 2013.

[BMSZ16] Saikrishna Badrinarayanan, Eric Miles, Amit Sahai, and Mark Zhandry. Post-zeroizing obfuscation: New mathematical tools, and the case of evasive circuits. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016*, volume 9666 of *LNCS*, pages 764–791, May 2016.

[BPR15] Nir Bitansky, Omer Paneth, and Alon Rosen. On the cryptographic hardness of finding a Nash equilibrium. In Venkatesan Guruswami, editor, *56th FOCS*, pages 1480–1498. IEEE Computer Society Press, October 2015.

[BR14] Zvika Brakerski and Guy N. Rothblum. Virtual black-box obfuscation for all circuits via generic graded encoding. In Yehuda Lindell, editor, *TCC 2014*, volume 8349 of *LNCS*, pages 1–25, February 2014.

[BÜW24] Chris Brzuska, Akin Ünal, and Ivy K. Y. Woo. Evasive LWE assumptions: Definitions, classes, and counterexamples. In *ASIACRYPT 2024*, LNCS, pages 418–449, December 2024.

[BV11] Zvika Brakerski and Vinod Vaikuntanathan. Efficient fully homomorphic encryption from (standard) LWE. In Rafail Ostrovsky, editor, *52nd FOCS*, pages 97–106. IEEE Computer Society Press, October 2011.

[BWZ14] Dan Boneh, David J. Wu, and Joe Zimmerman. Immunizing multilinear maps against zeroizing attacks. Cryptology ePrint Archive, Report 2014/930, 2014.

[CCMR24] Ran Canetti, Claudio Chamon, Eduardo R. Mucciolo, and Andrei E. Ruckenstein. Towards general-purpose program obfuscation via local mixing. LNCS, pages 37–70, November 2024.

[CGH$^+$15] Jean-Sébastien Coron, Craig Gentry, Shai Halevi, Tancrède Lepoint, Hemanta K. Maji, Eric Miles, Mariana Raykova, Amit Sahai, and Mehdi Tibouchi. Zeroizing without low-level zeroes: New MMAP attacks and their limitations. In Rosario Gennaro and Matthew J. B. Robshaw, editors, *CRYPTO 2015*, volume 9215 of *LNCS*, pages 247–266, August 2015.

[CHL$^+$15] Jung Hee Cheon, Kyoohyung Han, Changmin Lee, Hansol Ryu, and Damien Stehlé. Cryptanalysis of the multilinear map over the integers. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015*, volume 9056 of *LNCS*, pages 3–12, April 2015.

[CHN$^+$16] Aloni Cohen, Justin Holmgren, Ryo Nishimaki, Vinod Vaikuntanathan, and Daniel Wichs. Watermarking cryptographic capabilities. In Daniel Wichs and Yishay Mansour, editors, *48th ACM STOC*, pages 1115–1127. ACM Press, June 2016.

[CLR15] Jung Hee Cheon, Changmin Lee, and Hansol Ryu. Cryptanalysis of the new CLT multilinear maps. Cryptology ePrint Archive, Report 2015/934, 2015.

[CLT13] Jean-Sébastien Coron, Tancrède Lepoint, and Mehdi Tibouchi. Practical multilinear maps over the integers. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013*, volume 8042 of *LNCS*, pages 476–493, August 2013.

[CLT15] Jean-Sébastien Coron, Tancrède Lepoint, and Mehdi Tibouchi. New multilinear maps over the integers. In Rosario Gennaro and Matthew J. B. Robshaw, editors, *CRYPTO 2015*, volume 9215 of *LNCS*, pages 267–286, August 2015.

[CM24] Yilei Chen and Xinyu Mao. Universal computational extractors from lattice assumptions. Cryptology ePrint Archive, Report 2024/225, 2024.

[DGG$^+$18] Nico Döttling, Sanjam Garg, Divya Gupta, Peihan Miao, and Pratyay Mukherjee. Obfuscation from low noise multilinear maps. In Debrup Chakraborty and Tetsu Iwata, editors, *INDOCRYPT 2018*, volume 11356 of *LNCS*, pages 329–352, December 2018.

[DQV$^+$21] Lalita Devadas, Willy Quach, Vinod Vaikuntanathan, Hoeteck Wee, and Daniel Wichs. Succinct LWE sampling, random polynomials, and obfuscation. In Kobbi Nissim and Brent Waters, editors, *TCC 2021*, volume 13043 of *LNCS*, pages 256–287, November 2021.

[Gen09] Craig Gentry. Fully homomorphic encryption using ideal lattices. In Michael Mitzenmacher, editor, *41st ACM STOC*, pages 169–178. ACM Press, May / June 2009.

[GGG$^+$14] Shafi Goldwasser, S. Dov Gordon, Vipul Goyal, Abhishek Jain, Jonathan Katz, Feng-Hao Liu, Amit Sahai, Elaine Shi, and Hong-Sheng Zhou. Multi-input functional encryption. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 578–602, May 2014.

[GGH13a] Sanjam Garg, Craig Gentry, and Shai Halevi. Candidate multilinear maps from ideal lattices. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 1–17, May 2013.

[GGH$^+$13b] Sanjam Garg, Craig Gentry, Shai Halevi, Mariana Raykova, Amit Sahai, and Brent Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *54th FOCS*, pages 40–49. IEEE Computer Society Press, October 2013.

[GGH15] Craig Gentry, Sergey Gorbunov, and Shai Halevi. Graph-induced multilinear maps from lattices. In Yevgeniy Dodis and Jesper Buus Nielsen, editors, *TCC 2015*, volume 9015 of *LNCS*, pages 498–527, March 2015.

[GJK18] Craig Gentry, Charanjit S. Jutla, and Daniel Kane. Obfuscation using tensor products. Cryptology ePrint Archive, Report 2018/756, 2018.

[GJLS21] Romain Gay, Aayush Jain, Huijia Lin, and Amit Sahai. Indistinguishability obfuscation from simple-to-state hard problems: New assumptions, new techniques, and simplification. In Anne Canteaut and François-Xavier Standaert, editors, *EUROCRYPT 2021*, volume 12698 of *LNCS*, pages 97–126, October 2021.

[GKR08] Shafi Goldwasser, Yael Tauman Kalai, and Guy N. Rothblum. One-time programs. In David Wagner, editor, *CRYPTO 2008*, volume 5157 of *LNCS*, pages 39–56, August 2008.

[GKW17a] Rishab Goyal, Venkata Koppula, and Brent Waters. Lockable obfuscation. In Chris Umans, editor, *58th FOCS*, pages 612–621. IEEE Computer Society Press, October 2017.

[GKW17b] Rishab Goyal, Venkata Koppula, and Brent Waters. Separating semantic and circular security for symmetric-key bit encryption from the learning with errors assumption. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT 2017*, volume 10211 of *LNCS*, pages 528–557, April / May 2017.

[GP21] Romain Gay and Rafael Pass. Indistinguishability obfuscation from circular security. In Samir Khuller and Virginia Vassilevska Williams, editors, *53rd ACM STOC*, pages 736–749. ACM Press, June 2021.

[GPS16] Sanjam Garg, Omkant Pandey, and Akshayaram Srinivasan. Revisiting the cryptographic hardness of finding a nash equilibrium. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016*, volume 9815 of *LNCS*, pages 579–604, August 2016.

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In Richard E. Ladner and Cynthia Dwork, editors, *40th ACM STOC*, pages 197–206. ACM Press, May 2008.

[GSW13] Craig Gentry, Amit Sahai, and Brent Waters. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013*, volume 8042 of *LNCS*, pages 75–92, August 2013.

[GVW15] Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee. Predicate encryption for circuits from LWE. In Rosario Gennaro and Matthew J. B. Robshaw, editors, *CRYPTO 2015*, volume 9216 of *LNCS*, pages 503–523, August 2015.

[Hal15] Shai Halevi. Graded encoding, variations on a scheme. Cryptology ePrint Archive, Report 2015/866, 2015.

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.

[HJ16] Yupu Hu and Huiwen Jia. Cryptanalysis of GGH map. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016*, volume 9665 of *LNCS*, pages 537–565, May 2016.

[HJK$^+$16] Dennis Hofheinz, Tibor Jager, Dakshita Khurana, Amit Sahai, Brent Waters, and Mark Zhandry. How to generate and use universal samplers. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *ASIACRYPT 2016*, volume 10032 of *LNCS*, pages 715–744, December 2016.

[HJL21] Samuel B. Hopkins, Aayush Jain, and Huijia Lin. Counterexamples to new circular security assumptions underlying iO. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021*, volume 12826 of *LNCS*, pages 673–700, Virtual Event, August 2021.

[HLL23] Yao-Ching Hsieh, Huijia Lin, and Ji Luo. Attribute-based encryption for circuits of unbounded depth from lattices. In *64th FOCS*, pages 415–434. IEEE Computer Society Press, October 2023.

[HLL24] Yao-Ching Hsieh, Huijia Lin, and Ji Luo. A general framework for lattice-based ABE using evasive inner-product functional encryption. In *EUROCRYPT 2024*, LNCS, pages 433–464, June 2024.

[HSW13] Susan Hohenberger, Amit Sahai, and Brent Waters. Full domain hash from (leveled) multilinear maps and identity-based aggregate signatures. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013*, volume 8042 of *LNCS*, pages 494–512, August 2013.

[JLLS23] Aayush Jain, Huijia Lin, Paul Lou, and Amit Sahai. Polynomial-time cryptanalysis of the subspace flooding assumption for post-quantum $i\mathcal{O}$. In Carmit Hazay and Martijn Stam, editors, *EUROCRYPT 2023*, volume 14004 of *LNCS*, pages 205–235, April 2023.

[JLMS19] Aayush Jain, Huijia Lin, Christian Matt, and Amit Sahai. How to leverage hardness of constant-degree expanding polynomials over $\mathbb{R}$ to build $i\mathcal{O}$. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019*, volume 11476 of *LNCS*, pages 251–281, May 2019.

[JLS21] Aayush Jain, Huijia Lin, and Amit Sahai. Indistinguishability obfuscation from well-founded assumptions. In Samir Khuller and Virginia Vassilevska Williams, editors, *53rd ACM STOC*, pages 60–73. ACM Press, June 2021.

[JLS22] Aayush Jain, Huijia Lin, and Amit Sahai. Indistinguishability obfuscation from LPN over $\mathbb{F}_p$, DLIN, and PRGs in NC^0. In Orr Dunkelman and Stefan Dziembowski, editors, *EUROCRYPT 2022*, volume 13275 of *LNCS*, pages 670–699, May / June 2022.

[KLW15] Venkata Koppula, Allison Bishop Lewko, and Brent Waters. Indistinguishability obfuscation for Turing machines with unbounded memory. In Rocco A. Servedio and Ronitt Rubinfeld, editors, *47th ACM STOC*, pages 419–428. ACM Press, June 2015.

[Lin16] Huijia Lin. Indistinguishability obfuscation from constant-degree graded encoding schemes. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016*, volume 9665 of *LNCS*, pages 28–57, May 2016.

[Lin17] Huijia Lin. Indistinguishability obfuscation from SXDH on 5-linear maps and locality-5 PRGs. In Jonathan Katz and Hovav Shacham, editors, *CRYPTO 2017*, volume 10401 of *LNCS*, pages 599–629, August 2017.

[LM18] Huijia Lin and Christian Matt. Pseudo flawed-smudging generators and their application to indistinguishability obfuscation. Cryptology ePrint Archive, Report 2018/646, 2018.

[LPST16] Huijia Lin, Rafael Pass, Karn Seth, and Sidharth Telang. Indistinguishability obfuscation with non-trivial efficiency. In Chen-Mou Cheng, Kai-Min Chung, Giuseppe Persiano, and Bo-Yin Yang, editors, *PKC 2016*, volume 9615 of *LNCS*, pages 447–462, March 2016.

[LT17] Huijia Lin and Stefano Tessaro. Indistinguishability obfuscation from trilinear maps and block-wise local PRGs. In Jonathan Katz and Hovav Shacham, editors, *CRYPTO 2017*, volume 10401 of *LNCS*, pages 630–660, August 2017.

[LV16] Huijia Lin and Vinod Vaikuntanathan. Indistinguishability obfuscation from DDH-like assumptions on constant-degree graded encodings. In Irit Dinur, editor, *57th FOCS*, pages 11–20. IEEE Computer Society Press, October 2016.

[MF15] Brice Minaud and Pierre-Alain Fouque. Cryptanalysis of the new multilinear map over the integers. Cryptology ePrint Archive, Report 2015/941, 2015.

[MM11] Daniele Micciancio and Petros Mol. Pseudorandom knapsacks and the sample complexity of LWE search-to-decision reductions. In Phillip Rogaway, editor, *CRYPTO 2011*, volume 6841 of *LNCS*, pages 465–484, August 2011.

[MP12] Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 700–718, April 2012.

[MPV24] Surya Mathialagan, Spencer Peters, and Vinod Vaikuntanathan. Adaptively sound zero-knowledge SNARKs for UP. LNCS, pages 38–71, August 2024.

[MSZ16] Eric Miles, Amit Sahai, and Mark Zhandry. Annihilation attacks for multilinear maps: Cryptanalysis of indistinguishability obfuscation over GGH13. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016*, volume 9815 of *LNCS*, pages 629–658, August 2016.

[PST14] Rafael Pass, Karn Seth, and Sidharth Telang. Indistinguishability obfuscation from semantically-secure multilinear encodings. In Juan A. Garay and Rosario Gennaro, editors, *CRYPTO 2014*, volume 8616 of *LNCS*, pages 500–517, August 2014.

[QWW18] Willy Quach, Hoeteck Wee, and Daniel Wichs. Laconic function evaluation and applications. In Mikkel Thorup, editor, *59th FOCS*, pages 859–870. IEEE Computer Society Press, October 2018.

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In Harold N. Gabow and Ronald Fagin, editors, *37th ACM STOC*, pages 84–93. ACM Press, May 2005.

[RVV24] Seyoon Ragavan, Neekon Vafa, and Vinod Vaikuntanathan. Indistinguishability obfuscation from bilinear maps and LPN variants. LNCS, pages 3–36, November 2024.

[SW14] Amit Sahai and Brent Waters. How to use indistinguishability obfuscation: deniable encryption, and more. In David B. Shmoys, editor, *46th ACM STOC*, pages 475–484. ACM Press, May / June 2014.

[Tsa22] Rotem Tsabary. Candidate witness encryption from lattice techniques. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022*, volume 13507 of *LNCS*, pages 535–559, August 2022.

[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*, volume 13791 of *LNCS*, pages 195–221, December 2022.

[Wee22] Hoeteck Wee. Optimal broadcast encryption and CP-ABE from evasive lattice assumptions. In Orr Dunkelman and Stefan Dziembowski, editors, *EUROCRYPT 2022*, volume 13276 of *LNCS*, pages 217–241, May / June 2022.

[WW21] Hoeteck Wee and Daniel Wichs. Candidate obfuscation via oblivious LWE sampling. In Anne Canteaut and François-Xavier Standaert, editors, *EUROCRYPT 2021*, volume 12698 of *LNCS*, pages 127–156, October 2021.

[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*, volume 13747 of *LNCS*, pages 651–679, November 2022.

[WZ17] Daniel Wichs and Giorgos Zirdelis. Obfuscating compute-and-compare programs under LWE. In Chris Umans, editor, *58th FOCS*, pages 600–611. IEEE Computer Society Press, October 2017.