Candidate Obfuscation via Oblivious LWE Sampling

Hoeteck Wee[1] and Daniel Wichs[1,2(✉)]

[1] NTT Research Inc., San Francisco, USA
wichs@ccs.neu.edu
[2] Northeastern University, Boston, USA

Abstract. We present a new, simple candidate construction of indistinguishability obfuscation (iO). Our scheme is inspired by lattices and learning-with-errors (LWE) techniques, but we are unable to prove security under a standard assumption. Instead, we formulate a new falsifiable assumption under which the scheme is secure. Furthermore, the scheme plausibly achieves post-quantum security.

Our construction is based on the recent "split FHE" framework of Brakerski, Döttling, Garg, and Malavolta (EUROCRYPT '20), and we provide a new instantiation of this framework. As a first step, we construct an iO scheme that is provably secure assuming that LWE holds *and* that it is possible to obliviously generate LWE samples without knowing the corresponding secrets. We define a precise notion of oblivious LWE sampling that suffices for the construction. It is known how to obliviously sample from any distribution (in a very strong sense) using iO, and our result provides a converse, showing that the ability to obliviously sample from the specific LWE distribution (in a much weaker sense) already also implies iO. As a second step, we give a heuristic contraction of oblivious LWE sampling. On a very high level, we do this by homomorphically generating pseudorandom LWE samples using an encrypted pseudorandom function.

1 Introduction

Indistinguishability obfuscation (iO) [BGI+01, GR07] is a probabilistic polynomial-time algorithm $\mathcal{O}$ that takes as input a circuit C and outputs an (obfuscated) circuit $C' = \mathcal{O}(C)$ satisfying two properties: (a) *functionality*: C and C' compute the same function; and (b) *security*: for any two circuits C_1 and C_2 that compute the same function (and have the same size), $\mathcal{O}(C_1)$ and $\mathcal{O}(C_2)$ are computationally indistinguishable. Since the first candidate for iO was introduced in [GGH+13b], a series of works have shown that iO would have a huge impact on cryptography.

The state-of-the-art iO candidates with concrete instantiations may be broadly classified as follows:

– First, we have fairly simple and direct candidates based on graded "multilinear" encodings [GGH+13b, GGH13a, GGH15, FRS17, CVW18, BGMZ18,

A. Canteaut and F.-X. Standaert (Eds.): EUROCRYPT 2021, LNCS 12698, pp. 127–156, 2021.
https://doi.org/10.1007/978-3-030-77883-5_5

CHVW19] and that achieve plausible post-quantum security. These candidates have survived fairly intense scrutiny from cryptanalysts, [CHL+15, MSZ16, CLLT16, ADGM17, CLLT17, CGH17, Pel18, CVW18, CCH+19], and several of them are also provably secure in restricted adversarial models that capture a large class of known attacks. However, none of these candidates have a security reduction to a simple, falsifiable assumption.

- Next, we have a beautiful and remarkable line of works that aims to base iO on a conjunction of simple and well-founded assumptions, starting from [Lin16, LV16, Lin17, LT17], through [AJL+19, Agr19, JLMS19, GJLS20], and culminating in the very recent (and independent) work of Jain, Lin and Sahai [JLS20] basing iO on pairings, LWE, LPN and PRG in NC0. These constructions rely on the prior constructions of iO from functional encryption (FE) [BV15, AJ15], and proceed to build FE via a series of delicate and complex reductions, drawing upon techniques from a large body of works, including pairing-based FE for quadratic functions, lattice-based fully-homomorphic and attribute-based encryption, homomorphic secret-sharing, as well as hardness amplification.
- A number of more recent and incomparable candidates, including a direct candidate based on tensor products [GJK18] and another based on affine determinant programs (with noise) [BIJ+20]; the BDGM candidate based on an intriguing interplay between a LWE-based and a DCR-based cryptosystems [BDGM20a]; the plausibly post-quantum secure candidates in [Agr19, AP20] that replace the use of pairings in the second line of works with direct candidates for FE for inner product plus noise. All of these candidates, as with the first line of work, do not present a security reduction to a simple, falsifiable assumption.[1]

To the best of our knowledge, none of these existing approaches yields a lattice-inspired iO candidate that is plausibly post-quantum secure and enjoys a security reduction under a simple, falsifiable assumption referring solely to lattice-based cryptosystems, which is the focus of this work. We further believe that there is a certain aesthetic and minimalistic appeal to having an iO candidate whose hardness distills to a single source of computational hardness (as opposed to lattice plus pairing/number-theoretic hardness). Such a candidate is also potentially more amenable to crypto-analytic efforts as well as further research to reduce security to more standard lattice problems.

1.1 Our Contributions

Our main contribution is a new candidate construction of iO that relies on techniques from lattices and learning-with-errors (LWE). We formulate a new falsifiable assumption on the indistinguishability of two distributions, and show that our construction is secure under this assumption. While we are unable to

[1] We defer a comparison with the independent and concurrent works [GP20, BDGM20b] to Sect. 1.3.

prove security under a standard assumption such as LWE, we view our construction as a hopeful step in that direction. To our knowledge, this is the first iO candidate that is simultaneously based on a clearly stated falsifiable assumption and plausibly post-quantum secure. Perhaps more importantly, we open up a new avenue towards iO by showing that, under the LWE assumption, the ability to "obliviously sample from the LWE distribution" (see below) provably implies iO. Unlike prior constructions of iO from simpler primitives (e.g., functional encryption [AJ15,BV15], succinct randomized encodings [LPST16b], XiO [LPST16a], etc.), oblivious LWE sampling does not inherently involve any "computation" and appears to be fundamentally different. Lastly, we believe our construction is conceptually simpler and more self-contained (relying on fewer disjoint components) than many of the prior candidates.

Our main building block is an "oblivious LWE sampler", which takes as input a matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ and allows us to generate LWE samples $\mathbf{A} \cdot \mathbf{s} + \mathbf{e}$ with some small error $\mathbf{e} \in \mathbb{Z}^m$ without knowing the secrets $\mathbf{s}, \mathbf{e}$. We discuss the notion in more detail below (see the "Our Techniques" section), and provide a formal definition that suffices for our construction. Our notion can be seen as a significant relaxation of "invertible sampling" (in the common reference string model) [IKOS10,DKR15], and the equivalent notion of "pseudorandom encodings" [ACI+20]. The work of [DKR15] showed that, assuming iO, it is possible to invertibly sample from all distributions, and [ACI+20] asked whether it may be possible to do so under simpler assumptions that do not imply iO. As a side result of independent interest, we settle this question by showing that, under LWE, even our relaxed form of invertible sampling for the specific LWE distribution already implies iO.

Overall, our candidate iO construction consists of two steps. The first step is a provably secure construction of iO assuming we have an oblivious LWE sampler and that the LWE assumption holds (both with sub-exponential security). The second step is a candidate heuristic instantiation of an oblivious LWE sampler. On a very high level, our heuristic sampler performs a homomorphic computation that outputs a pseudorandom LWE sample generated using some pseudorandom function (PRF). Security boils down to a clearly stated falsifiable assumption that two distributions, both of which output LWE samples, are indistinguishable even if we give out the corresponding LWE secrets. Our assumption implicitly relies on some form of circular security: we assume that the error term in the pseudorandom LWE sample "drowns out" any error that comes out of the homomorphic computation over the PRF key that was used to generate it. We also discuss how our construction/assumption avoids some simple crypto-analytic attacks.

1.2 Technical Overview

Our iO construction is loosely inspired by the "split fully-homomorphic encryption (split FHE)" framework of Brakerski, Döttling, Garg, and Malavolta [BDGM20a] (henceforth BDGM). They defined a new cryptographic primitive called split FHE, which they showed to provably imply iO (under the LWE

Approach	Falsifiable	Circuit-Independent	Non-Interactive	Post-Quantum
mmaps-based iO, cf. [GGH+13b]				✓
NLFE candidates [Agr19, AP20]		✓		✓
split-FHE, DCR, LWE [BDGM20a]				
LWE, SXDH, LPN, PRG in NC0 [JLS20]	✓	✓	✓	
circular-SRL [GP20, BDGM20b]	✓			✓
this work (Conjecture 1 HPLS)	✓	✓	✓	✓

Fig. 1. Summary of the main approaches and assumptions used for IO. The column "falsifiable" refers to whether there is a reduction to a clearly stated falsifiable assumption (we don't count just assuming the scheme is secure). The term "circuit-independent" means that the assumption does not refer to computation for general circuits (which is closely related to the notion of instance-independent assumptions [GLW14]). We consider assumptions that quantify over worst-case inputs/parameters to be interactive, since the adversary chooses them in the first step.

assumption). They then gave a candidate instantiation of split FHE by heuristically combining decisional composite residue (DCR) and LWE-based techniques, together with the use of a random oracle. We rely on a slight adaptation of their framework by replacing split-FHE with a variant that we call *functional encodings*. Our main contribution is a new instantiation of this framework via "oblivious LWE sampling", relying only on LWE-based techniques (Fig. 1).

We first describe what functional encodings are and how to construct iO from functional encodings. Then we describe our instantiation of functional encodings via oblivious LWE sampling. We defer a detailed comparison to BDGM to Sect. 1.3.

iO from Functional Encodings

As in BDGM, instead of constructing iO directly, we construct a simpler a primitive called "exponentially efficient iO" XiO, which is known to imply iO under the LWE assumption [LPST16a]. We first describe what XiO is, and then discuss how to construct it from Functional Encodings via the BDGM framework.

XiO. An XiO scheme [LPST16a], has the same syntax, correctness and security requirements as iO, but relaxes the efficiency requirement. To obfuscate a circuit C with input length n, the obfuscator can run in exponential time $2^{O(n)}$ and the size of the obfuscated circuit can be as large as $2^{n(1-\varepsilon)}$ for some $\varepsilon > 0$. Such a scheme is useful when n is logarithmic in the security parameter, so that 2^n is some large polynomial. Note that there is always a trivial obfuscator that outputs the entire truth table of the circuit C, which is of size 2^n. Therefore, XiO is only required to do slightly better than the trivial construction, in that the size of the obfuscated circuit must be non-trivially smaller than the truth table. The work of [LPST16a] showed that XiO together with the LWE assumption (assuming both satisfy sub-exponential security) imply full iO.

Functional Encodings. We define a variant of the "split FHE" primitive from BDGM, which we call "functional encodings". A functional encoding can be

used to encode a value $x \in \{0,1\}^\ell$ to get an encoding $c = \mathsf{Enc}(x; r)$, where r is the randomness of the encoding process. Later, for any function $f : \{0,1\}^\ell \to \{0,1\}^m$, we can create an opening $d = \mathsf{Open}(f, x, r)$ for f, which can be decoded to recover the function output $\mathsf{Dec}(f, c, d) = f(x)$. We require many-opening simulation based security: the encoding $c = \mathsf{Enc}(x; r)$ together with the many openings $d_1 = \mathsf{Open}(f_1, x, r), \ldots, d_Q = \mathsf{Open}(f_Q, x, r)$ can be simulated given only the functions $f_1, \ldots, f_Q$ and the outputs $f_1(x), \ldots, f_Q(x)$. In other words, nothing about the encoded value x is revealed beyond the function outputs $f_i(x)$ for which openings are given. So far, we can achieve this by simply setting the opening d to be the function output $f(x)$. The notion is made non-trivial, by additionally requiring succinctness: the size of the opening d is bounded by $|d| = O(m^{1-\varepsilon})$ for some $\varepsilon > 0$, and therefore the opening must be non-trivially smaller than the output size of the function. We do not impose any restrictions on the size of the encoding c, which may depend polynomially on m. Unfortunately, this definition is unachievable in the plain model, as can be shown via a simple incompressibility argument. Therefore, we consider functional encodings in the common reference string (CRS) model and only require many-opening simulation security for some a-priori bound Q on the number of opening (i.e., Q-opening security). We allow the CRS size, (but *not* the encoding size or the opening size) to grow polynomially with the bound Q.

XiO from Functional Encodings. We construct XiO from functional encodings. As a first step, we construct XiO in the CRS model. Let $C : \{0,1\}^n \to \{0,1\}$ be a circuit of size ℓ that we want to obfuscate. We can partition the input domain $\{0,1\}^n$ of the circuit into $Q = 2^n/m$ subsets S_i, each containing $|S_i| = m$ inputs. We then define Q functions $f_i : \{0,1\}^\ell \to \{0,1\}^m$ such that $f_i(C) = (C(x_1), \ldots, C(x_m))$ outputs the evaluations of C on all m inputs $x_j \in S_i$. Finally, we set the obfuscation of the circuit C to be

$$(\mathsf{Enc}(C; r), \mathsf{Open}(f_1, C, r), \ldots, \mathsf{Open}(f_Q, C, r)),$$

which is sufficient to recover the value of the circuit at all $Q \cdot m = 2^n$ possible inputs. By carefully balancing between m and $Q = 2^n/m$, we can ensure that the obfuscated circuit size is $O(2^{n(1-\varepsilon)})$ for some constant $\varepsilon > 0$, and therefore satisfies the non-triviality requirement of XiO. On a high level, we amortize the large size of the encoding across sufficiently many openings to ensure that the total size of the encoding and all the openings together is smaller than the total output size.[2] The above gives us XiO with a strong form of simulation-based security (the obfuscated circuit can be simulated given the truth table) in the CRS model, which also implies the standard indistinguishability-based security in the CRS model.

[2] In detail, assume we start with a functional encoding where the encoding size is $O(m^a)$ and the opening size is $O(m^{1-\delta})$ for some constants $a, \delta > 0$, ignoring any other polynomial factors in the security parameter or the input size. The size of the obfuscated circuit above is then bounded by $O(m^a + Qm^{1-\delta})$. By choosing $m = 2^{n/(a+\delta)}$ and recalling $Q = 2^n/m$, the bound becomes $O(2^{n(1-\varepsilon)})$ for $\varepsilon = \delta/(a+\delta)$.

So far, we only got XiO in the CRS model, where the CRS size can be as large as $\mathsf{poly}(Q \cdot m) = 2^{O(n)}$. As the second step, we show that XiO in the CRS model generically implies XiO in the plain model. A naive idea would be to simply make the CRS a part of the obfuscated program, but then we would lose succinctness, since the CRS is large. Instead, we repeat a variant of the previous trick to amortize the cost of the CRS. To obfuscate a circuit $C : \{0,1\}^n \to \{0,1\}$, we partition the domain $\{0,1\}^n$ into $Q = 2^n/m$ subsets containing $m = 2^{n'}$ inputs each, and we define Q sub-circuits $C_i : \{0,1\}^{n'} \to \{0,1\}$, each of which evaluates C on the $m = 2^{n'}$ inputs in the i'th subset. We then choose a single CRS for input size n' and obfuscate all Q sub-circuits separately under this CRS; the final obfuscated circuit consists of the CRS and all the Q obfuscated sub-circuits. By carefully balancing between $m = 2^{n'}$ and $Q = 2^n/m$, in the same manner as previously, we can ensure that the total size of the final obfuscated circuit is $O(2^{n(1-\varepsilon)})$ for some constant $\varepsilon > 0$, and therefore the scheme satisfies the non-triviality requirement of XiO.

Constructing Functional Encodings

We now outline our construction of a functional encoding scheme. We start with a base scheme, which is insecure but serves as the basis of our eventual construction. We show that we can easily make it one-opening simulation secure under the LWE assumption, meaning that security holds in the special case where only a single opening is ever provided (i.e., $Q = 1$). Then we show how to make it many-opening secure via oblivious LWE sampling. Concretely, we obtain a Q-opening secure functional encoding candidate for bounded-depth circuits $f : \{0,1\}^\ell \to \{0,1\}^m$ with CRS size $O(Q \cdot m)$, encoding size $O(m^2)$ and opening size $O(1)$, and where $O(\cdot)$ hides factors polynomial in the security parameter, input size ℓ, and circuit depth.

Base Scheme. Our construction of functional encodings is based on a variant of the homomorphic encryption/commitment schemes of [GSW13,GVW15]. Given a commitment to an input $\mathbf{x} = (x_1, \ldots, x_\ell) \in \{0,1\}^\ell$, along with a circuit $f : \{0,1\}^\ell \to \{0,1\}^m$, this scheme allows us to homomorphically compute a commitment to the output $y = f(\mathbf{x})$. Our variant is designed to ensure that the opening for the output commitment is smaller than the output size m.

Given a public random matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ where $m \gg n$, we define a commitment $\mathbf{C}$ to an input $\mathbf{x}$ via

$$\mathbf{C} = (\mathbf{A}\mathbf{R}_1 + x_1\mathbf{G} + \mathbf{E}_1, \ldots, \mathbf{A}\mathbf{R}_\ell + x_\ell\mathbf{G} + \mathbf{E}_\ell)$$

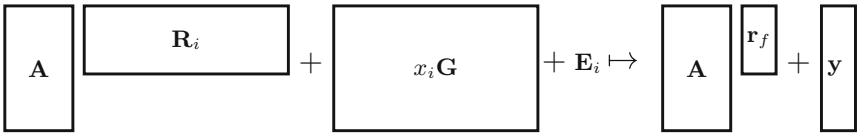
where $\mathbf{R}_i \leftarrow \mathbb{Z}_q^{n \times m \log q}$, $\mathbf{E}_i \leftarrow \chi^{m \times m \log q}$ has its entries chosen from the error distribution χ, and $\mathbf{G} \in \mathbb{Z}_q^{m \times m \log q}$ is the gadget matrix of [MP12]. Although this looks similar to [GSW13,GVW15], we stress that the parameters are different. Namely, in our scheme $\mathbf{A}$ is a tall/thin matrix while in the prior schemes it is a short/fat matrix, we allow $\mathbf{R}_i$ to be uniformly random over the entire space while in the prior schemes it had small entries, and we need to add some error $\mathbf{E}_i$ that was not needed in the prior schemes. The commitment scheme is hiding by the LWE assumption. We can define the functional encoding $\mathsf{Enc}(\mathbf{x}; r) = (\mathbf{A}, \mathbf{C})$

to consist of the matrix $\mathbf{A}$ and the homomorphic commitment $\mathbf{C}$, where r is all the randomness used to sample the above values.

Although we modified several key parameters of [GSW13, GVW15], it turns out that the same homomorphic evaluation procedure there still applies to our modified scheme. In particular, given the commitment $\mathbf{C}$ to an input $\mathbf{x}$ and a boolean circuit $f : \{0,1\}^\ell \to \{0,1\}$, we can homomorphically derive a commitment $\mathbf{C}_f = \mathbf{A}\mathbf{R}_f + f(x)\mathbf{G} + \mathbf{E}_f$ to the output $f(x)$. Furthermore, given a circuit $f : \{0,1\}^\ell \to \{0,1\}^m$ with m bit output, we can apply the above procedure to get commitments to each of the output bits and "pack" them together using the techniques of (e.g.,) [MW16, BTVW17, PS19, GH19, BDGM19] to obtain a vector $\mathbf{c}_f \in \mathbb{Z}_q^m$ such that

$$\mathbf{c}_f = \mathbf{A} \cdot \mathbf{r}_f + f(\mathbf{x}) \cdot \tfrac{q}{2} + \mathbf{e}_f \in \mathbb{Z}_q^m$$

where $f(\mathbf{x}) \in \{0,1\}^m$ is a column vector, $\mathbf{r}_f \in \mathbb{Z}_q^n$, and $\mathbf{e}_f \in \mathbb{Z}^m$ is some small error term.



Now, observe that $\mathbf{r}_f$ constitutes a succinct opening to $f(\mathbf{x})$, since $|\mathbf{r}_f| \ll |f(\mathbf{x})|$ and $\mathbf{r}_f$ allows us to easily recover $f(\mathbf{x})$ from $\mathbf{c}_f$ by computing $\mathsf{round}_{q/2}(\mathbf{c}_f - \mathbf{A} \cdot \mathbf{r}_f)$. Furthermore, we can efficiently compute $\mathbf{r}_f$ by applying a homomorphic computation on the opening of the input commitment as in [GVW15], or alternately, we can sample $\mathbf{A}$ with a trapdoor and use the trapdoor to recover $\mathbf{r}_f$. Therefore, we can define the opening procedure of the functional encoding to output the value $\mathbf{r}_f = \mathsf{Open}(f, \mathbf{x}, r)$, and the decoding procedure can recover $f(x) = \mathsf{Dec}(f, (\mathbf{A}, \mathbf{C}), \mathbf{r}_f)$ by homomorphically computing $\mathbf{c}_f$ and using $\mathbf{r}_f$ to recover $f(x)$ as above. This gives us our base scheme (in the plain model), which has the correct syntax and succinctness properties. Unfortunately, the scheme so far does not satisfy even one-opening simulation security, since the opening $\mathbf{r}_f$ (along with the error term $\mathbf{e}_f$ that it implicitly reveals) may leak additional information about $\mathbf{x}$ beyond $f(\mathbf{x})$.

One-Opening Security from LWE. We can modify the base scheme to get one-opening simulation security (still in the plain model). In particular, we augment the encoding by additionally including a single random LWE sample $\mathbf{b} = \mathbf{A} \cdot \mathbf{s} + \mathbf{e}$ inside it. We then add this LWE sample to $\mathbf{c}_f$ to "randomize" it, and release $\mathbf{d}_f := \mathbf{r}_f + \mathbf{s}$ as an opening to $f(\mathbf{x})$. Given the encoding $(\mathbf{A}, \mathbf{C}, \mathbf{b})$ and the opening $\mathbf{d}_f$, we can decode $f(\mathbf{x})$ by homomorphically computing $\mathbf{c}_f$ and outputting $y = \mathsf{round}_{q/2}(\mathbf{c}_f + \mathbf{b} - \mathbf{A} \cdot \mathbf{d}_f)$. Correctness follows from the fact that $\mathbf{c}_f + \mathbf{b} \approx \mathbf{A}(\mathbf{r}_f + \mathbf{s}) + f(\mathbf{x}) \cdot q/2$.

With the above modification, we can simulate an encoding/opening pair given only $f(\mathbf{x})$ without knowing $\mathbf{x}$. Firstly, we can simulate the opening without knowing the randomness of the input commitments or the trapdoor for $\mathbf{A}$. In particular, the simulator samples $\mathbf{d}_f$ uniformly at random from $\mathbb{Z}_q^n$, and then "programs" the value $\mathbf{b}$ as $\mathbf{b} := \mathbf{A} \cdot \mathbf{d}_f - \mathbf{c}_f + f(\mathbf{x}) \cdot \frac{q}{2} + \mathbf{e}$. The only difference in

the distributions is that in the real case the error contained in the LWE sample $\mathbf{b}$ is $\mathbf{e}$, while in the simulated case it is $\mathbf{e}-\mathbf{e}_f$, but we can choose the error $\mathbf{e}$ to be large enough to "smudge out" this difference and ensure that the distributions are statistically close. Once we can simulate the opening without having the randomness of the input commitments or the trapdoor for $\mathbf{A}$, we can rely on LWE to replace the input commitment to $\mathbf{x}$ with a commitment to a dummy value.

Many-Opening Security via Oblivious LWE Sampling. We saw that we can upgrade the base scheme to get one-opening simulation security by adding a random LWE sample $\mathbf{b} = \mathbf{A} \cdot \mathbf{s} + \mathbf{e}$ to the encoding. We could easily extend the same idea to achieve Q-opening simulation security by adding Q samples $\mathbf{b}_i = \mathbf{A} \cdot \mathbf{s}_i + \mathbf{e}_i$ to the encoding. However, this would require the encoding size to grow with Q, which we cannot afford. So far, we have not relied on a CRS, and perhaps the next natural attempt would be to add the Q samples $\mathbf{b}_i$ to the CRS of the scheme. Unfortunately, this also does not work, since the scheme needs to know the corresponding LWE secrets $\mathbf{s}_i$ to generate the openings, and we would not be able to derive them from the CRS.

Imagine that we had an oracle that took as input an arbitrary matrix $\mathbf{A}$ and would output Q random LWE samples $\mathbf{b}_i = \mathbf{A} \cdot \mathbf{s}_i + \mathbf{e}_i$. Such an oracle would allow us to construct Q-opening simulation secure functional encodings. The encoding procedure would choose the matrix $\mathbf{A}$ with a trapdoor, call the oracle to get samples $\mathbf{b}_i$ and use the trapdoor to recover the values $\mathbf{s}_i$ that it would use to generate the openings. The decoding procedure would get $\mathbf{A}$ and call the oracle to recover the samples $\mathbf{b}_i$ needed to decode, but would not learn anything else. The simulator would be able to program the oracle and choose the values $\mathbf{b}_i$ itself, which would allow us to prove security analogously to the one-opening setting. We define a cryptographic primitive called an "oblivious LWE sampler", whose goal is to approximate the functionality of the above oracle in the standard model with a CRS. We can have several flavors of this notion, and we start by describing a strong flavor, which we then relax in various ways to get our actual definition.

Oblivious LWE Sampler (Strong Flavor). A strong form of oblivious LWE sampling would consist of a deterministic sampling algorithm Sam that takes as input a long CRS along with a matrix $\mathbf{A}$ and outputs Q LWE samples $\mathbf{b}_i = \mathsf{Sam}(\mathsf{CRS}, \mathbf{A}, i)$ for $i \in [Q]$. The size of CRS can grow with Q and the CRS can potentially be chosen from some structured distribution, but it must be independent of $\mathbf{A}$. We want to be able to arbitrarily "program" the outputs of the sampler by programming the CRS. In other words, there is a simulator Sim that gets $\mathbf{A}$ and Q random LWE samples $\{\mathbf{b}_i\}$ as targets; it outputs a programmed string $\mathsf{CRS} \leftarrow \mathsf{Sim}(\mathbf{A}, \{\mathbf{b}_i\})$ that causes the sampler to output the target values $\mathbf{b}_i = \mathsf{Sam}(\mathsf{CRS}, \mathbf{A}, i)$. We want the real and the simulated CRS to be indistinguishable, even for a worst-case choice of $\mathbf{A}$ for which an adversary may know a trapdoor that allows it to recover the LWE secrets. This notion would directly plug in to our construction to get a many-opening secure functional encoding scheme in the CRS model. It turns out that this strong form

of oblivious LWE sampling can be seen as a special case of *invertible sampling* (in the CRS model) as proposed by [IKOS10], and can be constructed from iO [DKR15]. Invertible sampling is also equivalent to pseudorandom encodings (with computational security in the CRS model) [ACI+20], and we answer one of the main open problems posed by that work by showing that these notions provably imply iO under the LWE assumption. Unfortunately, we do not know how to heuristically instantiate this strong flavor of oblivious LWE sampling (without already having iO).

Oblivious LWE Sampler (Relaxed). We relax the above strong notion in several ways. Firstly, we allow ourselves to "pre-process" the matrix $\mathbf{A}$ using some secret coins to generate a value $\mathsf{pub} \leftarrow \mathsf{Init}(\mathbf{A})$ that is given as an additional input to the sampler $\mathbf{b}_i = \mathsf{Sam}(\mathsf{CRS}, \mathsf{pub}, i)$. We only require that the size of pub is independent of the number of samples Q that will be generated. The simulator gets to program both $\mathsf{CRS}, \mathsf{pub}$ to produce the desired outcome. Secondly, we relax the requirement that, by programming $\mathsf{CRS}, \mathsf{pub}$, the simulator can cause the sampler output arbitrary target values $\mathbf{b}_i$. Instead, we now give the simulator some target values $\hat{\mathbf{b}}_i$ and the simulator is required to program $(\mathsf{CRS}, \mathsf{pub}) \leftarrow \mathsf{Sim}(\mathbf{A}, \hat{\mathbf{b}}_i)$ to ensure that the sampled values $\mathbf{b}_i = \mathsf{Sam}(\mathsf{CRSpub}, i)$ satisfy $\mathbf{b}_i = \hat{\mathbf{b}}_i + \tilde{\mathbf{b}}_i$ for some LWE sample $\tilde{\mathbf{b}}_i = \mathbf{A} \cdot \tilde{\mathbf{s}}_i + \tilde{\mathbf{e}}_i$ for which the simulator knows the corresponding secrets $\tilde{\mathbf{s}}_i, \tilde{\mathbf{e}}_i$. In other words, the produced samples $\mathbf{b}_i$ need not exactly match the target values $\hat{\mathbf{b}}_i$ given to the simulator, but the difference has to be an LWE sample $\tilde{\mathbf{b}}_i$ for which the simulator can produce the corresponding secrets. Lastly, instead of requiring that the indistinguishability of the real and simulated $(\mathsf{CRS}, \mathsf{pub})$ holds even for a worst-case choice of $\mathbf{A}$ with a known trapdoor, we only require that it holds for a random $\mathbf{A}$, but the adversary is additionally given the LWE secrets $\mathbf{s}_i$ contained in the sampled values $\mathbf{b}_i = \mathbf{A} \cdot \mathbf{s}_i + \mathbf{e}_i$. In other words, we require that real/simulated distributions of $(\mathsf{CRS}, \mathsf{pub}, \{\mathbf{s}_i\})$ are indistinguishable.

We show that this relaxed form of an oblivious LWE sampling suffices in our construction of functional encodings. Namely, we can simply add pub to the encoding of the functional encoding scheme, since it is short. In the proof, we can replace the real $(\mathsf{CRS}, \mathsf{pub})$ with a simulated one, using some random LWE tuples $\hat{\mathbf{b}}_i$ as target values. Indistinguishability holds even given the LWE secrets $\mathbf{s}_i$ for the produced samples $\mathbf{b}_i = \mathsf{Sam}(\mathsf{CRS}, \mathsf{pub}, i)$, which are used to generate the openings of the functional encoding. The $\hat{\mathbf{b}}_i$ component of the produced samples $\mathbf{b}_i = \hat{\mathbf{b}}_i + \tilde{\mathbf{b}}_i$ is sufficient to re-randomizes the output commitment $\mathbf{c}_f$, and the additional LWE sample $\tilde{\mathbf{b}}_i$ that is added in does not hurt security, since we know the corresponding LWE secret $\tilde{\mathbf{s}}_i$ and can use it to adjust the opening accordingly.

Constructing an Oblivious LWE Sampler. We give a heuristic construction of an oblivious LWE sampler, by relying on the same homomorphic commitments that we used to construct our base functional encoding scheme. The high level idea is to give out a commitment to a PRF key $\mathbf{k}$ and let the sampling algorithm homomorphically compute a pseudorandom LWE sample $\mathbf{b}_{\mathsf{prf}} := \mathbf{A} \cdot \mathbf{s}_{\mathsf{prf}} + \mathbf{e}_{\mathsf{prf}}$

where $\mathbf{s}_{\mathsf{prf}}, \mathbf{e}_{\mathsf{prf}}$ are sampled using randomness that comes from the PRF. The overall output of the sampler is a commitment to the above LWE sample, which is itself an LWE sample! While we do not know how to construct a simulator for this basic construction, we conjecture that it may already be sufficient to instantiate functional encodings. To allow the simulator to program the output, we augment the computation to incorporate the CRS. We give a more detailed description below.

The CRS is a uniformly random string, which we interpret as consisting of Q values $\mathsf{CRS}_i \in \mathbb{Z}_q^m$. To generate pub, we sample a random key $\mathbf{k}$ for a pseudorandom function $\mathsf{PRF}(\mathbf{k}, \cdot)$ and set a flag bit $\beta := 0$. We creates a commitment $\mathbf{C}$ to the input $(\mathbf{k}, \beta)$ and we set the public value $\mathsf{pub} = (\mathbf{A}, \mathbf{C})$. The algorithm $\mathbf{b}_i = \mathsf{Sample}(\mathsf{CRS}, \mathsf{pub}, i)$ performs a homomorphic computation of the function g_i over the commitment $\mathbf{C}$, where g_i is defined as follows:

$$g_i(\mathbf{k}, \beta): \quad \text{Use } \mathsf{PRF}(\mathbf{k}, i) \text{ to sample } \mathbf{b}_i^{\mathsf{prf}} := \mathbf{A} \cdot \mathbf{s}_i^{\mathsf{prf}} + \mathbf{e}_i^{\mathsf{prf}} \text{ and output } \mathbf{b}_i^* := \mathbf{b}_i^{\mathsf{prf}} + \beta \cdot \mathsf{CRS}_i.$$

The output of this computation is a homomorphically evaluated commitment to $\mathbf{b}_i^*$ and has the form $\mathbf{b}_i = \mathbf{A} \cdot \mathbf{s}_i^{\mathsf{eval}} + \mathbf{e}_i^{\mathsf{eval}} + \mathbf{b}_i^*$ where $\mathbf{s}_i^{\mathsf{eval}}, \mathbf{e}_i^{\mathsf{eval}}$ come from the homomorphic evaluation.[3] Overall, the generated samples $\mathbf{b}_i = \mathsf{Sample}(\mathsf{CRS}, \mathsf{pub}, i)$ can be written as

$$\mathbf{b}_i = \mathbf{A} \cdot (\mathbf{s}_i^{\mathsf{eval}} + \mathbf{s}_i^{\mathsf{prf}}) + (\mathbf{e}_i^{\mathsf{eval}} + \mathbf{e}_i^{\mathsf{prf}}) + \beta \cdot \mathsf{CRS}_i$$

where $\mathbf{s}_i^{\mathsf{prf}}, \mathbf{e}_i^{\mathsf{prf}}$ come from the PRF output and $\mathbf{s}_i^{\mathsf{eval}}, \mathbf{e}_i^{\mathsf{eval}}$ come from the homomorphic evaluation.

In the real scheme, the flag β is set to 0 and so each output of Sample is an LWE sample $\mathbf{b}_i = \mathbf{A} \cdot (\mathbf{s}_i^{\mathsf{eval}} + \mathbf{s}_i^{\mathsf{prf}}) + (\mathbf{e}_i^{\mathsf{eval}} + \mathbf{e}_i^{\mathsf{prf}})$. In the simulation, the simulator gets some target values $\hat{\mathbf{b}}_i$ and puts them in the CRS as $\mathsf{CRS}_i := \hat{\mathbf{b}}_i$. It sets the flag to $\beta = 1$, which results in the output of Sample being $\mathbf{b}_i = \mathbf{A} \cdot (\mathbf{s}_i^{\mathsf{eval}} + \mathbf{s}_i^{\mathsf{prf}}) + (\mathbf{e}_i^{\mathsf{eval}} + \mathbf{e}_i^{\mathsf{prf}}) + \hat{\mathbf{b}}_i$. Note that the simulator knows the PRF key $\mathbf{k}$ and the randomness of the homomorphic commitment, and therefore knows the values $(\mathbf{s}_i^{\mathsf{eval}} + \mathbf{s}_i^{\mathsf{prf}}), (\mathbf{e}_i^{\mathsf{eval}} + \mathbf{e}_i^{\mathsf{prf}})$. This means that the difference between the target values $\hat{\mathbf{b}}_i$ and the output samples $\mathbf{b}_i$ is an LWE tuple for which the simulator knows the corresponding secrets, as required.

Security Under a New Conjecture. We conjecture that the above construction is secure. In particular, we conjecture that the adversary cannot distinguish between $\beta = 0$ and $\beta = 1$ given the values:

$$(\mathsf{CRS} = \{\mathsf{CRS}_i = \mathbf{A}\hat{\mathbf{s}}_i + \hat{\mathbf{e}}_i\}_{i \in [Q]}, \mathsf{pub} = (\mathbf{A}, \mathbf{C} = \mathsf{Commit}(k, \beta)), \{\mathbf{s}_i = \mathbf{s}_i^{\mathsf{eval}} + \mathbf{s}_i^{\mathsf{prf}} + \beta\hat{\mathbf{s}}_i\}_{i \in [Q]})$$

[3] Recall that previously we relied on a "packed" homomorphic evaluation, where we could evaluate a function $f : \{0,1\}^\ell \to \{0,1\}^m$ on a commitment to $\mathbf{x}$ to get a commitment $\mathbf{c}_f = \mathbf{A} \cdot \mathbf{s}_f + \mathbf{e}_f + f(\mathbf{x}) \cdot \frac{q}{2}$. The above relies on a slight variant that's even further packed and allows us to homomorphically evaluate a function $g : \{0,1\}^\ell \to \mathbb{Z}_q^m$ over a commitment to $\mathbf{x}$ and derive a commitment $\mathbf{c}_g = \mathbf{A} \cdot \mathbf{s}_g + \mathbf{e}_g + g(\mathbf{x})$.

We refer to this as the *homomorphic pseudorandom LWE samples (HPLS)* conjecture (see Conjecture 1 for a precise statement), and we argue heuristically why we believe it to hold. Since $\mathsf{CRS}, \mathsf{pub}$ completely determine the values $\mathbf{b}_i = \mathbf{A} \cdot \mathbf{s}_i + \mathbf{e}_i$, revealing $\mathbf{s}_i = \mathbf{s}_i^{\mathsf{eval}} + \mathbf{s}_i^{\mathsf{prf}} + \beta\hat{\mathbf{s}}_i$ also implicitly reveals $\mathbf{e}_i = \mathbf{e}_i^{\mathsf{eval}} + \mathbf{e}_i^{\mathsf{prf}} + \beta\hat{\mathbf{e}}_i$. We can think of the HPLS conjecture as consisting of two distinct heuristic components. The first component is to argue that the values $\mathbf{s}_i, \mathbf{e}_i$ look pseudorandom and independent of β given only $(\mathsf{CRS}, \mathbf{A})$, but without getting the commitment $\mathbf{C}$. Intuitively, we believe this to hold since $\mathbf{s}_i^{\mathsf{prf}}, \mathbf{e}_i^{\mathsf{prf}}$ are provably pseudorandom (by the security of the PRF). Therefore, as long as we choose the noise $\mathbf{e}_i^{\mathsf{prf}}$ to be large enough to "smudge out" $\hat{\mathbf{e}}_i$, we can provably argue that $\mathbf{s}_i^{\mathsf{prf}} + \beta\hat{\mathbf{s}}_i$ and $\mathbf{e}_i^{\mathsf{prf}} + \beta\hat{\mathbf{e}}_i$ are pseudorandom and independent of β. Unfortunately, this does not suffice – we still need to rely on a heuristic to ague that there are no computationally discernible correlations between these values and $\mathbf{s}_i^{\mathsf{eval}}, \mathbf{e}_i^{\mathsf{eval}}$ respectively. We believe this should hold with most natural PRFs. Although the first component is already heuristic, there is hope to remove the heuristic nature of this component by explicitly analyzing the distributions $\mathbf{s}_i^{\mathsf{eval}} + \mathbf{s}_i^{\mathsf{prf}}, \mathbf{e}_i^{\mathsf{eval}} + \mathbf{e}_i^{\mathsf{eval}}$ for a specific PRF, and leave this as a fascinating open problem for future work. The second heuristic component is to argue that security holds even in the presence of the commitment $\mathbf{C}$. This part implicitly involves a circular security aspect between the pseudorandom function and the commitment. We'd like to argue that the PRF key $\mathbf{k}$ and the bit β are protected by the security of the commitment scheme, but we release $\mathbf{s}_i = \mathbf{s}_i^{\mathsf{eval}} + \mathbf{s}_i^{\mathsf{prf}} + \beta\hat{\mathbf{s}}_i$, where $\mathbf{s}_i^{\mathsf{eval}}$ depends on the commitment randomness; nevertheless we'd like to argue that this does not hurt commitment security since the value $\mathbf{s}_i^{\mathsf{eval}}$ is masked by the PRF output, but this argument is circular since the PRF key is contained in the commitment! This circularity does not easily lend itself to a proof, and we see much less hope in removing the heuristic nature of the second component than the first. Still, this type of circularity also seems difficult to attack: one cannot easily break the security of the commitment without first breaking the security of the PRF and vice versa.

Simplified Construction. In the full version, we also give a simplified direct construction of functional encodings in the plain model that we conjecture to satisfy indistinguishability based security. The simplified construction does not go through the intermediate "oblivious LWE sampler" primitive. In contrast to our main construction, which is secure under a non-interactive assumption that two distributions are indistinguishable, the assumption that our simplified construction is secure and interactive.

1.3 Discussion and Perspectives

Comparison to BDGM

We now give a detailed comparison of our results/techniques with those of Brakerski, Döttling, Garg, and Malavolta [BDGM20a] (BDGM). BDGM defined a

primitive called split FHE, which they show implies iO under the LWE assumption. They then gave a candidate instantiation of split FHE by heuristically combining decisional composite residue (DCR) and LWE-based techniques, together with the use of a random oracle. While they gave compelling intuition for why they believe this construction of split FHE to be secure, they did not attempt to formulate an assumption under which they could prove security. In our work, we define a variant of split FHE that we call functional encodings. We then provide an entirely new instantiation of functional encodings via oblivious LWE sampling. The main advantages of our approach are:

- We get a provably secure construction of iO under the LWE assumption along with an additional assumption that there is an oblivious LWE sampler, where the latter is a clearly abstracted primitive, which we then instantiate heuristically. In particular, we are able to confine the heuristic portion of our construction to a single well defined component.
- We can prove security of our overall construction under a falsifiable, non-interactive assumption that is independent of the function being obfuscated.
- Our construction of iO relies only on LWE-based techniques rather than the additional use of DCR. In our opinion, this makes the construction conceptually simpler and easier to analyze. Furthermore, the construction is plausibly post-quantum secure.
- We avoid any reliance on random oracles.

On a technical level, we lightly adapt the split FHE framework of BDGM. In particular, our notion of functional encodings can be seen as a relaxed form of split FHE, and our result that functional encodings imply iO closely follows BDGM. The main differences between the two works, lie in the our respective instantiations of split-FHE and functional encodings. We explain the differences in the framework and the instantiation in more detail below.

Functional Encodings vs Split FHE. There are two differences between our notion of functional encodings versus the split FHE framework of BDGM. Firstly, our notion of functional encodings has a simplified syntax compared to split FHE (in particular, we do not require any key generation or homomorphic evaluation algorithms and the opening can depend on all of the randomness r used to generate the encoding rather than just a secret key). While we find the simplified syntax conceptually easier, it is not crucial, and our candidate construction of functional encodings can be adapted to also match the syntactic requirements of split FHE. The second difference is that we explicitly allow for a CRS in functional encodings, and show that the CRS can be removed when we go to XiO (in particular, we show that XiO in the CRS model implies XiO in the plain model). In contrast, the work of BDGM considered split FHE in the plain model (with indistinguishability rather than simulation security). Their instantiation relies on a random oracle model and they argued heuristically that the random oracle can be removed. The fact that we explicitly consider the

CRS model allows us to avoid random oracles entirely, and therefore reduce the number of heuristic components in the final construction.[4]

Heuristic Instantiations. Both BDGM and our work provide a heuristic instantiation of the main building block: split FHE and functional encodings, respectively. These instantiations are concretely very different, and rely on different techniques. On a conceptual level, they also differ in the role that heuristic arguments play. BDGM constructs a provably secure instantiation of split FHE under the combination of LWE and DCR assumptions, in some idealized oracle world (essentially, the oracle samples Damgard-Jurik encryptions of small values). They then give a heuristic instantiation of their oracle. However, there is no attempt to define any standard-model notion of security that such an instantiation could satisfy to make the overall scheme secure. In contrast, we construct a provably secure instantiation of functional encodings under the LWE assumption and assuming we have an "oblivious LWE sampler", where the latter is a cryptographic primitive in the standard model (with a CRS) with a well-defined security requirement. We then give a heuristic construction of an oblivious LWE sampler using LWE techniques. Although the security notion of oblivious LWE sampling involves a simulator, our heuristic construction comes with a candidate simulator for it. Therefore, the only heuristic component of our construction is a clearly stated falsifiable assumption that two distributions (real and simulated) are indistinguishable.

We conjecture that the split FHE construction of BDGM could similarly be proven secure under the LWE assumption, DCR assumption, and some type of "oblivious sampler" for Damgard-Jurik encryptions of random small values. Moreover, the heuristic instantiation of the oracle in BDGM could likely be seen as a heuristic candidate for such an oblivious sampler. However, BDGM does not appear to have a plausible candidate simulator for this instantiation and hence security does not appear to follow from any simple falsifiable assumption (other than assuming that the full construction of split FHE is secure).

We note that BDGM (Sect. 4.4) also presents an alternate construction of split FHE based only the LWE assumption (without DCR) in some other idealized oracle world. However, they were not able to heuristically instantiate the oracle for this alternate construction, and hence it did not lead to even a heuristic candidate for post-quantum secure iO in their work.[5] Their construction does yield a one-opening secure split-FHE / functional encoding under LWE, and our one-opening secure scheme is in part inspired by it (and can be seen as simplifying it). The main advantage of our scheme is that we can extend it to many-opening security via oblivious LWE sampling, which we then instantiate heuristically to get a candidate iO.

[4] We believe that this change could also be applied retroactively to remove the use of a random oracles in BDGM.

[5] As stated in BDGM Sect. 4.4: "We stress that, in contrast with the instantiation based on the Damgard-Jurik encryption scheme (Sect. 4.3), this scheme does not satisfy the syntactical requirements to apply the generic transformations (described in Sect. 4.2) to lift the scheme to the plain model.".

Comparison with FE
The line of work on building iO from simple, well-founded assumptions first builds functional encryption (FE). A functional encryption scheme allows us to encrypt a value x and generate secret keys for functions f so that decryption returns $f(x)$ while leaking no additional information about x. We also consider Q-key security, where an adversary given an encryption of x and Q secret keys for functions $f_1, \ldots, f_Q$ should learn nothing about x beyond $f_1(x), \ldots, f_Q(x)$. A functional encoding scheme can be viewed as a relaxation of a secret-key functional encryption where we allow the key for f to depend on x.

The state-of-the-art for functional encryption is analogous to that for functional encoding:

- We have one-key secure public-key FE for bounded-depth circuits $f : \{0,1\}^\ell \to \{0,1\}^m$ from LWE with ciphertext size $O(m)$ and key size $O(1)$ [GKP+13, GVW13, BGG+14].
- A construction of iO from one-key secure public-key FE for bounded-depth circuits $f : \{0,1\}^\ell \to \{0,1\}^m$ with ciphertext size $O(m^{1-\epsilon})$ [BV15, AJ15]. The latter is in turn implied by Q-key secure public-key FE for $f : \{0,1\}^\ell \to \{0,1\}$ with ciphertext size $O(Q^{1-\epsilon})$.
- A construction of iO from Q-key secure secret-key FE bounded-depth circuits $f : \{0,1\}^\ell \to \{0,1\}^m$ with ciphertext size $Q^{1-\epsilon} \cdot \mathsf{poly}(m)$. Our main candidate is essentially the functional encoding analogue of such a secret-key FE scheme (in the CRS model).

This analogue raises two natural open problems: Do the techniques in this work also yield non-trivial FE schemes (that imply iO) with a polynomial security loss, without passing through iO as an intermediate building block? Can we simplify the constructions or assumptions underlying the FE schemes in [AJL+19, Agr19, JLMS19, GJLS20, JLS20] by relaxing the requirements from FE to functional encodings (which would still suffice for iO)?

Comparison with Concurrent Works: [GP20, BDGM20b]
The recent work of [GP20] together with a follow-up to it [BDGM20b] (both of which are concurrent and independent of our work), present new candidate constructions of iO by adapting the BDGM [BDGM20a] framework. Just like our work, they go through the route of constructing XiO in the CRS model, and have instantiations that rely only on LWE-style techniques and are plausibly post-quantum secure. While there are many high-level similarities between these works and our work, the concrete construction and security assumption are different. In terms of construction, the main difference lies in how the works "re-randomize" the opening/hint that allows one to recover the output of the computation. In our case, we do so via an "oblivious LWE sampler", which is instantiated by using an encrypted PRF key to produce an encrypted pseudorandom LWE sample. The two works [GP20, BDGM20b] follow the original construction of [BDGM20a] more closely and rely on homomorphically decrypt-

ing random ciphertexts in the CRS using a key cycle.[6] Our overall construction is arguably somewhat simper than the others since it relies on a single homomorphic cryptosystem (a variant of GSW FHE) rather than switching between two different homomorphic cryptosystems with different properties. In terms of assumptions, both of the works [GP20,BDGM20b] prove security under a new assumption that a certain cryptosystem satisfies a strong form of "circular security" in the presence of some oracle. In the full version, we give a more detailed comparison and our take on the circular security assumptions.

2 Preliminaries

2.1 Notations

We will denote by λ the security parameter. The notation $\mathsf{negl}(\lambda)$ denotes any function f such that $f(\lambda) = \lambda^{-\omega(1)}$, and $\mathsf{poly}(\lambda)$ denotes any function f such that $f(\lambda) = \mathcal{O}(\lambda^c)$ for some $c > 0$. For a probabilistic algorithm $\mathsf{alg}(\mathsf{inputs})$, we might explicit the randomness it uses by writing $\mathsf{alg}(\mathsf{inputs}; \mathsf{coins})$. We will denote vectors by bold lower case letters (e.g. $\mathbf{a}$) and matrices by bold upper cases letters (e.g. $\mathbf{A}$). We will denote by $\mathbf{a}^\top$ and $\mathbf{A}^\top$ the transposes of $\mathbf{a}$ and $\mathbf{A}$, respectively. We will denote by $\lfloor x \rceil$ the nearest integer to x, rounding towards 0 for half-integers. If $\mathbf{x}$ is a vector, $\lfloor \mathbf{x} \rceil$ will denote the rounded value applied component-wise. For integral vectors and matrices (i.e., those over $\mathbb{Z}$), we use the notation $|\mathbf{r}|, |\mathbf{R}|$ to denote the maximum absolute value over all the entries.

We define the statistical distance between two random variables X and Y over some domain Ω as: $\mathbf{SD}(X, Y) = \frac{1}{2} \sum_{w \in \Omega} |X(w) - Y(w)|$. We say that two ensembles of random variables $X = \{X_\lambda\}$, $Y = \{Y_\lambda\}$ are *statistically indistinguishable*, denoted $X \overset{\text{s}}{\approx} Y$, if $\mathbf{SD}(X_\lambda, Y_\lambda) \leq \mathsf{negl}(\lambda)$.

We say that two ensembles of random variables $X = \{X_\lambda\}$, and $Y = \{Y_\lambda\}$ are *computationally indistinguishable*, denoted $X \overset{\text{c}}{\approx} Y$, if, for all (non-uniform) PPT distinguishers Adv, we have $|\Pr[\mathsf{Adv}(X_\lambda) = 1] - \Pr[\mathsf{Adv}(Y_\lambda) = 1]| \leq \mathsf{negl}(\lambda)$. We also refer to sub-exponential security, meaning that there exists some $\varepsilon > 0$ such that the distinguishing advantage is at most $2^{-\lambda^\varepsilon}$.

We assume familiarity with the learning-with errors (LWE) assumption [Reg05], noise smudging (e.g., [AJL+12]), the Gadget Matrix $\mathbf{G}$ [MP12] and lattice trapdoors [Ajt96,MP12]. See the full version for details.

3 Functional Encodings

3.1 Definition of Functional Encodings

A *functional encoding scheme* (in the CRS model) for the family $\mathcal{F}_{\ell,m,t} = \{f : \{0,1\}^\ell \to \{0,1\}^m\}$ of depth-t circuits consists of four PPT algorithms $\mathsf{crsGen}, \mathsf{Enc}, \mathsf{Open}, \mathsf{Dec}$ where Open and Dec are deterministic, satisfying the following properties:

[6] Interestingly, since decrypting random ciphertexts is a (weak-)PRF, the two approaches may be more similar than may appear.

Syntax: The algorithms have the following syntax:

- $\mathsf{CRS} \leftarrow \mathsf{crsGen}(1^\lambda, 1^Q, \mathcal{F}_{\ell,m,t})$ outputs CRS for security parameter 1^λ and a bound Q on the number of openings;
- $C \leftarrow \mathsf{Enc}(\mathsf{CRS}, x \in \{0,1\}^\ell; r)$ encodes x using randomness r;
- $d \leftarrow \mathsf{Open}(\mathsf{CRS}, f : \{0,1\}^\ell \to \{0,1\}^m, i \in [Q], x, r)$ computes the opening corresponding to i'th function f;
- $y \leftarrow \mathsf{Dec}(\mathsf{CRS}, f, i, C, d)$ computes a value y for the encoding C and opening d.

Correctness:

$$\mathsf{Dec}(f, \mathsf{Enc}(x, r), \mathsf{Open}(f, x, r)) = f(x)$$

Q-SIM **Security:** There exists a PPT simulator Sim such that the following distributions for all PPT adversaries $\mathcal{A}$ and all $x, f^1, \ldots, f^Q \leftarrow \mathcal{A}(1^\lambda)$, the following distributions of $(\mathsf{CRS}, C, d_1, \ldots, d_Q)$ are computationally indistinguishable (even given $x, f^1, \ldots, f^Q$):

- Real Distribution: $\mathsf{CRS} \leftarrow \mathsf{crsGen}(1^\lambda, 1^Q), C \leftarrow \mathsf{Enc}(\mathsf{CRS}, x; r), d_i \leftarrow \mathsf{Open}(\mathsf{CRS}, f^i, i, x, r), i \in [Q]$.
- Simulated Distribution: $(\mathsf{CRS}, C, d_1, \ldots, d_Q) \leftarrow \mathsf{Sim}(\{f^i, f^i(x)\}_{i \in Q})$.

Succinctness: There exists a constant $\epsilon > 0$ such that, for $\mathsf{CRS} \leftarrow \mathsf{crsGen}(1^\lambda, 1^Q, \mathcal{F}_{\ell,m,t})$, $C \leftarrow \mathsf{Enc}(\mathsf{CRS}, x; r)$, $d \leftarrow \mathsf{Open}(\mathsf{CRS}, f, i, x, r)$ we have:

$$|\mathsf{CRS}| = \mathsf{poly}(Q, \lambda, \ell, m, t), |C| = \mathsf{poly}(\lambda, \ell, m, t), |d| = m^{1-\varepsilon}\mathsf{poly}(\lambda, \ell, t).$$

In our discussion, we also refer to indistinguishability-based security, a relaxation of Q-SIM security:

Q-IND **Security:** For all PPT adversaries $\mathcal{A}$ and all $\mathbf{x}_0, \mathbf{x}_1, f^1, \ldots, f^Q \leftarrow \mathcal{A}(1^\lambda)$ such that $f^i(\mathbf{x}_0) = f^i(\mathbf{x}_1)$ for all $i \in [Q]$, the following distributions of $(\mathsf{CRS}, C, d_1, \ldots, d_Q)$ are computationally indistinguishable for $\beta = 0$ and $\beta = 1$:

$$\mathsf{CRS} \leftarrow \mathsf{crsGen}(1^\lambda, 1^Q), C \leftarrow \mathsf{Enc}(\mathsf{CRS}, x^\beta; r), d_i \leftarrow \mathsf{Open}(\mathsf{CRS}, f^i, i, x^\beta, r), i \in [Q]$$

Remark 1 (Comparison with split-FHE). One can think of functional encodings as essentially a relaxation of split-FHE, where we remove the explicit requirements for decryption (and secret keys) and for homomorphic evaluation. This simplifies both the syntax and the security definition. In the language of BDGM, Open corresponds to a decryption hint for an encryption of $f(x)$, obtained by applying partial decryption to homomorphic evaluation of f on the encryption of x. Note that in BDGM, the hint should be computable given the decryption key, whereas we allow the hint to depend on the encryption/commitment randomness. Finally, BDGM circumvents the impossibility of simulation-based security for many-time security in the plain model by turning to indistinguishability-based security, whereas we rely on a CRS.

Remark 2 (Comparison with functional encryption). Functional encoding is very similar to (secret-key) functional encryption where given an encryption of x and a secret key for f, we learn $f(x)$ and nothing else about x. A crucial distinction here is that Open also gets x as input.

4 Homomorphic Commitments with Short Openings

In this section, we describe a homomorphic commitment scheme with short openings.

Lemma 1 (Homomorphic computation on matrices [GSW13,BGG+14]). *Fix parameters m, q, ℓ. Given a matrix $\mathbf{C} \in \mathbb{Z}_q^{m \times \ell m \log q}$ and a circuit $f : \{0,1\}^\ell \to \{0,1\}$ of depth t, we can efficiently compute a matrix $\mathbf{C}_f$ such that for all $\mathbf{x} \in \{0,1\}^\ell$, there exists a matrix $\mathbf{H}_{\mathbf{C},f,\mathbf{x}} \in \mathbb{Z}^{\ell m \log q \times m \log q}$ with $|\mathbf{H}_{\mathbf{C},f,\mathbf{x}}| = m^{O(t)}$ such that*[7]

$$(\mathbf{C} - \mathbf{x}^\top \otimes \mathbf{G}) \cdot \mathbf{H}_{\mathbf{C},f,\mathbf{x}} = \mathbf{C}_f - f(\mathbf{x})\mathbf{G} \tag{1}$$

where $\mathbf{G} \in \mathbb{Z}_q^{m \times m \log q}$ is the gadget matrix. Moreover, $\mathbf{H}_{\mathbf{C},f,\mathbf{x}}$ is efficiently computable given $\mathbf{C}, f, \mathbf{x}$.

Using the "packing" techniques in [MW16,BTVW17,PS19], the above relation extends to circuits with m-bit output. Concretely, given a circuit $f : \{0,1\}^\ell \to \{0,1\}^m$ of depth t, we can efficiently compute a vector $\mathbf{c}_f$ such that for all $\mathbf{x} \in \{0,1\}^\ell$, there exists a vector $\mathbf{h}_{\mathbf{C},f,\mathbf{x}} \in \mathbb{Z}^{\ell m \log q}$ with $|\mathbf{h}_{\mathbf{C},f,\mathbf{x}}| = m^{O(t)}$ such that

$$(\mathbf{C} - \mathbf{x}^\top \otimes \mathbf{G}) \cdot \mathbf{h}_{\mathbf{C},f,\mathbf{x}} = \mathbf{c}_f - f(\mathbf{x}) \cdot \tfrac{q}{2} \tag{2}$$

where $f(\mathbf{x}) \in \{0,1\}^m$ is a column vector. Concretely, let $f_1, \ldots, f_m : \{0,1\}^m \to \{0,1\}$ denote the circuits computing the output bits of f. Then, we have:

$$\mathbf{c}_f = \sum_{j=1}^{m} \mathbf{C}_{f_j} \cdot \mathbf{G}^{-1}(\mathbf{1}_j \cdot \tfrac{q}{2}) \tag{3}$$

$$\mathbf{h}_{\mathbf{C},f,\mathbf{x}} = \sum_{j=1}^{m} \mathbf{H}_{\mathbf{C},f_j,\mathbf{x}} \cdot \mathbf{G}^{-1}(\mathbf{1}_j \cdot \tfrac{q}{2})$$

where $\mathbf{1}_j \in \{0,1\}^m$ is the indicator column vector whose j'th entry is 1 and 0 everywhere else, so that $f(\mathbf{x}) = \sum_j f_i(\mathbf{x}) \cdot \mathbf{1}_j$. Here, $\mathbf{h}_{\mathbf{C},f,\mathbf{x}}$ is also efficiently computable given $\mathbf{C}, f, \mathbf{x}$.

Construction 1 (homomorphic commitments pFHC). *The commitment scheme pFHC ("packed fully homomorphic commitment") is parameterized by m, ℓ and n, q, and is defined as follows.*

– Gen *chooses a uniformly random matrix $\mathbf{A} \leftarrow \mathbb{Z}_q^{m \times n}$.*

[7] Note that if we write $\mathbf{C} = [\mathbf{C}_1 \mid \cdots \mid \mathbf{C}_\ell]$ where $\mathbf{C}_1, \ldots, \mathbf{C}_\ell \in \mathbb{Z}_q^{m \times m \log q}$ and $\mathbf{x} = (x_1, \ldots, x_\ell)$, then

$$\mathbf{C} - \mathbf{x}^\top \otimes \mathbf{G} = [\mathbf{C}_1 - x_1\mathbf{G} \mid \ldots \mid \mathbf{C}_\ell - x_\ell\mathbf{G}]$$

.

- $\mathsf{Com}(\mathbf{A} \in \mathbb{Z}_q^{m\times n}, \mathbf{x} \in \{0,1\}^\ell; \mathbf{R} \in \mathbb{Z}_q^{n\times \ell m \log q}, \mathbf{E} \in \mathbb{Z}^{m\times \ell m \log q})$ *outputs a commitment*
$$\mathbf{C} := \mathbf{AR} + \mathbf{x}^\top \otimes \mathbf{G} + \mathbf{E} \in \mathbb{Z}_q^{m\times \ell m \log q}.$$
Here, $\mathbf{R} \leftarrow \mathbb{Z}_q^{n\times \ell m \log q}, \mathbf{E} \leftarrow \chi^{m\times \ell m \log q}$
- $\mathsf{Eval}(f : \{0,1\}^\ell \to \{0,1\}^m, \mathbf{C} \in \mathbb{Z}_q^{m\times \ell m \log q})$ *for a boolean circuit* $f : \{0,1\}^\ell \to \{0,1\}^m$*, deterministically outputs a (column) vector* $\mathbf{c}_f \in \mathbb{Z}_q^m$*. Here,* $\mathbf{c}_f$ *is the same as that given in (2).*
- $\mathsf{Eval}_{\mathsf{open}}(f, \mathbf{A} \in \mathbb{Z}_q^{m\times n}, \mathbf{x} \in \{0,1\}^\ell, \mathbf{R} \in \mathbb{Z}_q^{n\times \ell m \log q}, \mathbf{E} \in \mathbb{Z}^{m\times \ell m \log q})$*: deterministically outputs (column) vectors* $\mathbf{r}_f \in \mathbb{Z}_q^n, \mathbf{e}_f \in \mathbb{Z}_q^m$*.*

Lemma 2. *The above commitment scheme* pFHC *satisfies the following properties:*

- **Correctness.** *For any boolean circuit* $f : \{0,1\}^\ell \to \{0,1\}^m$ *of depth* t*, any* $\mathbf{x} \in \{0,1\}^\ell$*, any* $\mathbf{A} \in \mathbb{Z}_q^{m\times n}, \mathbf{R} \in \mathbb{Z}_q^{n\times \ell m \log q}, \mathbf{E} \in \mathbb{Z}^{m\times \ell m \log q}$*, we have*
$$\mathbf{C} := \mathsf{Com}(\mathbf{A}, \mathbf{x}; \mathbf{R}, \mathbf{E}), \quad \mathbf{c}_f := \mathsf{Eval}(f, \mathbf{C}), \quad (\mathbf{r}_f, \mathbf{e}_f) := \mathsf{Eval}_{\mathsf{open}}(f, \mathcal{A}, \mathbf{x}, \mathbf{R}, \mathbf{E})$$
satisfies
$$\mathbf{c}_f = \mathbf{A}\mathbf{r}_f + f(\mathbf{x}) \cdot \tfrac{q}{2} + \mathbf{e}_f \in \mathbb{Z}_q^m$$
where $f(\mathbf{x}) \in \{0,1\}^m$ *is a column vector and* $|\mathbf{e}_f| = |\mathbf{E}| \cdot m^{O(t)}$*.*
- **Privacy.** *Under the LWE assumption, for all* $\mathbf{x} \in \{0,1\}^\ell$*, we have:*
$$\mathbf{A}, \mathsf{Com}(\mathbf{A}, \mathbf{x}) \approx_c \mathbf{A}, \mathsf{Com}(\mathbf{A}, \mathbf{0})$$

Proof. Correctness follows from substituting $\mathbf{C} = \mathbf{AR} + \mathbf{x}^\top \otimes \mathbf{G} + \mathbf{E}$ into (2), which yields
$$\mathbf{c}_f = (\mathbf{AR} + \mathbf{E}) \cdot \mathbf{h}_{\mathbf{C},f,\mathbf{x}} + f(x) \cdot \tfrac{q}{2} = \mathbf{A} \cdot \underbrace{\mathbf{R} \cdot \mathbf{h}_{\mathbf{C},f,\mathbf{x}}}_{\mathbf{r}_f} + f(x) \cdot \tfrac{q}{2} + \underbrace{\mathbf{E} \cdot \mathbf{h}_{\mathbf{C},f,\mathbf{x}}}_{\mathbf{e}_f}.$$
The bound on $|\mathbf{e}_f|$ follows from $|\mathbf{h}_{\mathbf{C},f,\mathbf{x}}| = m^{O(t)}$. Privacy follows readily from the pseudorandomness of $(\mathbf{A}, \mathbf{AR} + \mathbf{E})$, as implied by the LWE assumption.

Handling. $f : \{0,1\}^\ell \to \mathbb{Z}_q^m$**.** Next, we observe that we can also augment pFHC with a pair of algorithms $\mathsf{Eval}^q, \mathsf{Eval}^q_{\mathsf{open}}$ to support bounded-depth circuits $f : \{0,1\}^\ell \to \mathbb{Z}_q^m$ (following [PS19]). That is,

- **Correctness II.** For any boolean circuit $f : \{0,1\}^\ell \to \mathbb{Z}_q^m$ of depth t, any $\mathbf{x} \in \{0,1\}^\ell$, any $\mathbf{A} \in \mathbb{Z}_q^{m\times n}, \mathbf{R} \in \mathbb{Z}_q^{n\times \ell m \log q}, \mathbf{E} \in \mathbb{Z}^{m\times \ell m \log q}$, we have
$$\mathbf{C} := \mathsf{Com}(\mathbf{A}, \mathbf{x}; \mathbf{R}, \mathbf{E}), \quad \mathbf{c}_f := \mathsf{Eval}^q(f, \mathbf{C}), \quad (\mathbf{r}_f, \mathbf{e}_f) := \mathsf{Eval}^q_{\mathsf{open}}(f, \mathcal{A}, \mathbf{x}, \mathbf{R}, \mathbf{E})$$
satisfies
$$\mathbf{c}_f = \mathbf{A}\mathbf{r}_f + f(\mathbf{x}) + \mathbf{e}_f \in \mathbb{Z}_q^m$$
where $f(\mathbf{x}) \in \mathbb{Z}_q^m$ is a column vector and $|\mathbf{e}_f| = |\mathbf{E}| \cdot m^{O(t)}$.

Concretely, let $f_1, \ldots, f_{m \log q} : \{0,1\}^m \to \{0,1\}$ denote the circuits computing the output of f interpreted as bits. Then, we have:

$$\mathbf{c}_f = \sum_{j=1}^{m \log q} \mathbf{C}_{f_j} \cdot \mathbf{G}^{-1}(\mathbf{1}_j \otimes \mathbf{g}^\top) \tag{4}$$

$$\mathbf{h}_{\mathbf{C},f,\mathbf{x}} = \sum_{j=1}^{m \log q} \mathbf{H}_{\mathbf{C},f_j,\mathbf{x}} \cdot \mathbf{G}^{-1}(\mathbf{1}_j \otimes \mathbf{g}^\top)$$

5 1-SIM Functional Encoding from LWE

We construct a 1-SIM functional encoding scheme for bounded-depth circuits $\mathcal{F}_{\ell,m,t}$ based on the LWE assumption. The scheme does not require a CRS. Such a result is given in [BDGM20, Sect. 4.4], starting from any FHE scheme with "almost linear" decryption; we provide a more direct construction that avoids key-switching.

Construction 2

– $\mathsf{Enc}(\mathbf{x}; \mathbf{A}, \mathbf{R}, \mathbf{E}, \mathbf{s}, \mathbf{e})$. *Sample*

$$\mathbf{A} \leftarrow \mathbb{Z}_q^{m \times n}, \mathbf{R} \leftarrow \mathbb{Z}_q^{n \times \ell m \log q}, \mathbf{E} \leftarrow \chi^{m \times \ell m \log q}, \mathbf{s} \leftarrow \mathbb{Z}_q^n, \mathbf{e} \leftarrow \hat{\chi}^m$$

Compute

$$\mathbf{C} := \mathsf{pFHC.Com}(\mathbf{A}, \mathbf{x}; \mathbf{R}, \mathbf{E}),\ \mathbf{b} := \mathbf{A}\mathbf{s} + \mathbf{e}$$

and output

$$(\mathbf{A}, \mathbf{C}, \mathbf{b})$$

– $\mathsf{Open}(f, \mathbf{x}; \mathbf{A}, \mathbf{R}, \mathbf{E}, \mathbf{s}, \mathbf{e})$*: Compute* $(\mathbf{r}_f, \mathbf{e}_f) := \mathsf{pFHC.Eval}_{\mathsf{open}}(f, \mathbf{A}, \mathbf{x}, \mathbf{R}, \mathbf{E})$ *and output*

$$\mathbf{d} := \mathbf{r}_f + \mathbf{s} \in \mathbb{Z}_q^n$$

– $\mathsf{Dec}(f, (\mathbf{A}, \mathbf{C}, \mathbf{b}), \mathbf{d})$*: Compute* $\mathbf{c}_f := \mathsf{pFHC.Eval}(f, \mathbf{C})$ *and output*

$$\mathsf{round}_{q/2}(\mathbf{c}_f + \mathbf{b} - \mathbf{A}\mathbf{d}) \in \{0,1\}^m$$

where $\mathsf{round}_{q/2} : \mathbb{Z}_q^m \to \{0,1\}^m$ *is coordinate-wise rounding to the nearest multiple of* $q/2$.

Theorem 3. *Under the* $LWE_{n,q,\chi}$ *assumption, the construction above is a* 1-SIM *functional encoding.*

We defer the proof to the full version.

Remark 3 (An attack given many openings.). We describe an attack strategy on our 1-SIM scheme in the Q-SIM setting, namely, when the adversary gets openings $\mathbf{d}_1, \ldots, \mathbf{d}_Q$ corresponding to many functions $f^1, \ldots, f^Q$. (We stress that this does not contradict our preceding security claim.) Observe that we have

$$\mathbf{d}_i = \mathbf{R} \cdot \mathbf{h}_{\mathbf{C},f^i,\mathbf{x}} + \mathbf{s}$$

where $\mathbf{h}_{\mathbf{C},f^i,\mathbf{x}}$ (as defined in (2)) is efficiently computable given $\mathbf{x}, \mathbf{C}, f^i$. In the case of linear functions, $\mathbf{h}_{\mathbf{C},f^i,\mathbf{x}}$ does not even depend on $\mathbf{x}$. This gives us Q linear equations in the unknowns $\mathbf{R}, \mathbf{s}$, and allows us to recover $\mathbf{R}$ and break many-opening security in both the indistinguishability-based and simulation-based settings as long as we can choose f^i's in such a way that the equations are linearly independent.

6 Oblivious Sampling from a Falsifiable Assumption

Oblivious LWE sampling allows us to compute Q seemingly random LWE samples $\mathbf{b}_i = \mathbf{A}\mathbf{s}_i + \mathbf{e}_i$ relative to some LWE matrix $\mathbf{A}$, by applying some deterministic function to a long CRS that is independent of $\mathbf{A}$ along with a short public value pub that can depend on $\mathbf{A}$ but whose length is independent of Q. We require that there is a simulator that can indistinguishably program $\mathsf{CRS}, \mathsf{pub}$ to ensures that the resulting samples $\mathbf{b}_i$ "almost match" some arbitrary LWE samples $\hat{\mathbf{b}}_i$ given to the simulator as inputs. Ideally, the simulator could ensure that $\mathbf{b}_i = \hat{\mathbf{b}}_i$ match exactly. However, we relax this and only require the simulator to ensure that $\mathbf{b}_i = \hat{\mathbf{b}}_i + \tilde{\mathbf{b}}_i$ for some LWE sample $\tilde{\mathbf{b}}_i = \mathbf{A}\tilde{\mathbf{s}}_i + \tilde{\mathbf{e}}_i$ for which the simulator knows the corresponding secret $\tilde{\mathbf{s}}_i$. Note that the simulator does not get the secrets $\hat{\mathbf{s}}_i$ for the target values $\hat{\mathbf{b}}_i = \mathbf{A}\hat{\mathbf{s}}_i + \hat{\mathbf{e}}_i$, but indistinguishability should hold even for a distinguisher that gets the secrets $\mathbf{s}_i$ for the output samples $\mathbf{b}_i = \mathbf{A}\mathbf{s}_i + \mathbf{e}_i$. In the full version, we show that we can construct a strong form of oblivious sampling using the notion of invertible sampling (in the CRS model) from [IKOS10, DKR15, ACI+20], which can be constructed from iO. This highlights that the notion is plausibly achievable. We then give a heuristic constructions of oblivious LWE sampling using LWE-style techniques and heuristically argue that security holds under a new falsifiable assumption.

6.1 Definition of Oblivious Sampling

An oblivious LWE sampler consists of four PPT algorithms: $\mathsf{CRS} \leftarrow \mathsf{crsGen}(1^\lambda, 1^Q), \mathsf{pub} \leftarrow \mathsf{Init}(\mathbf{A}), \mathbf{b}_i = \mathsf{Sample}(\mathsf{CRS}, \mathsf{pub}, i)$ and $(\mathsf{CRS}, \mathsf{pub}, \{\tilde{\mathbf{s}}_i\}_{i\in[Q]}) \leftarrow \mathsf{Sim}(1^\lambda, 1^Q, \mathbf{A}, \{\hat{\mathbf{b}}_i\}_{i\in[Q]})$. The Sample algorithm is required to be deterministic while the others are randomized. Let $(\mathsf{TrapGen}, \mathsf{LWESolve})$ be the lattice trapdoor algorithms for generating $\mathbf{A}$ with a trapdoor and solving LWE using the trapdoor respectively.

Definition 1. *An $(n, m, q, \hat{\chi}, B_{\mathsf{OLWE}})$ oblivious LWE sampler satisfies the following properties:*

Correctness: *Let* $Q = Q(\lambda)$ *be some polynomial. Let* $(\mathbf{A}, \mathsf{td}) \leftarrow \mathsf{TrapGen}(1^n, 1^m, q), \mathsf{CRS} \leftarrow \mathsf{crsGen}(1^\lambda, 1^Q)$, $\mathsf{pub} \leftarrow \mathsf{Init}(\mathbf{A})$. *Then, with overwhelming probability over the above values, for all* $i \in [Q]$ *there exists some* $\mathbf{s}_i \in \mathbb{Z}_q^n$ *and* $\mathbf{e}_i \in \mathbb{Z}_q^m$ *with* $||\mathbf{e}_i||_\infty \leq B_{\mathsf{OLWE}}$ *such that* $\mathbf{b}_i = \mathbf{A}\mathbf{s}_i + \mathbf{e}_i$.

Security: *The following distributions of* $(\mathsf{CRS}, \mathbf{A}, \mathsf{pub}, \{\mathbf{s}_i\}_{i\in[Q]})$ *are computationally indistinguishable:*

- *Real Distribution: Sample* $(\mathbf{A}, \mathsf{td}) \leftarrow \mathsf{TrapGen}(1^n, 1^m, q)$, $\mathsf{CRS} \leftarrow \mathsf{crsGen}(1^\lambda, 1^Q)$, $\mathsf{pub} \leftarrow \mathsf{Init}(\mathbf{A})$. *For* $i \in [Q]$ *set* $\mathbf{b}_i = \mathsf{Sample}(\mathsf{CRS}, \mathsf{pub}, i)$, $\mathbf{s}_i = \mathsf{LWESolve}_{\mathsf{td}}(\mathbf{b}_i)$. *Output* $(\mathsf{CRS}, \mathbf{A}, \mathsf{pub}, \{\mathbf{s}_i\}_{i\in[Q]})$.
- *Simulated Distribution: Sample* $(\mathbf{A}, \mathsf{td}) \leftarrow \mathsf{TrapGen}(1^n, 1^m, q)$, $\hat{\mathbf{s}}_i \leftarrow \mathbb{Z}_q^n, \hat{\mathbf{e}}_i \leftarrow \hat{\chi}^m$ *and let* $\hat{\mathbf{b}}_i = \mathbf{A}\hat{\mathbf{s}}_i + \hat{\mathbf{e}}_i$. *Sample* $(\mathsf{CRS}, \mathsf{pub}, \{\tilde{\mathbf{s}}_i\}_{i\in[Q]}) \leftarrow \mathsf{Sim}(1^\lambda, 1^Q, \mathbf{A}, \{\hat{\mathbf{b}}_i\}_{i\in[Q]})$ *and let* $\mathbf{s}_i = \hat{\mathbf{s}}_i + \tilde{\mathbf{s}}_i$. *Output* $(\mathsf{CRS}, \mathbf{A}, \mathsf{pub}, \{\mathbf{s}_i\}_{i\in[Q]})$.

Observe that the algorithm $\mathsf{pub} \leftarrow \mathsf{Init}(\mathbf{A})$ in the above definition does not get Q as an input and therefore the size of pub is independent of Q. On the other hand, the algorithm $\mathsf{CRS} \leftarrow \mathsf{crsGen}(1^\lambda, 1^Q)$ does not get $\mathbf{A}$ as an input and hence CRS must be independent of $\mathbf{A}$. This is crucial and otherwise there would be a trivial construction where either CRS or pub would consist of Q LWE samples with respect to $\mathbf{A}$.

Note that the security property implicitly also guarantees the following correctness property of the simulated distribution. Assume we simulate the values $(\mathsf{CRS}, \mathbf{A}, \mathsf{pub}, \{\tilde{\mathbf{s}}_i\}_{i\in[Q]}) \leftarrow \mathsf{Sim}(1^\lambda, 1^Q, \mathbf{A}, \{\hat{\mathbf{b}}_i\}_{i\in[Q]})$ where the simulator is given LWE samples $\hat{\mathbf{b}}_i = \mathbf{A}\hat{\mathbf{s}}_i + \hat{\mathbf{e}}_i$ as input. Then the resulting $(\mathsf{CRS}, \mathbf{A}, \mathsf{pub})$ will generate samples $\mathbf{b}_i = \mathsf{Sample}(\mathsf{CRS}, \mathsf{pub}, i)$ of the form $\mathbf{b}_i = \hat{\mathbf{b}}_i + \tilde{\mathbf{b}}_i$ where $\tilde{\mathbf{b}}_i = \mathbf{A}\tilde{\mathbf{s}}_i + \tilde{\mathbf{e}}_i$ some small $\tilde{\mathbf{e}}_i$. This is because, in the simulation, we must have $\mathbf{b}_i = \mathbf{A}\mathbf{s}_i + \mathbf{e}_i$ where $||\mathbf{e}_i||_\infty \leq B$ as otherwise it would be trivial to distinguish the simulation from the real case. But $\mathbf{s}_i = \hat{\mathbf{s}}_i + \tilde{\mathbf{s}}_i$ and so $\mathbf{e}_i = \hat{\mathbf{e}}_i + \tilde{\mathbf{e}}_i$. This implies $\tilde{\mathbf{e}}_i = \mathbf{e}_i - \hat{\mathbf{e}}_i$ will be small.

Remark 4 (Naive construction fails). Consider the naive construction:

$$\mathsf{pub} := (\mathbf{A}\mathbf{S} + \mathbf{E}), \quad \mathsf{CRS} := (\mathbf{r}_1, \ldots, \mathbf{r}_Q), \quad \mathbf{b}_i := (\mathbf{A}\mathbf{S} + \mathbf{E})\mathbf{r}_i$$

where

$$\mathbf{A} \leftarrow \mathbb{Z}_q^{m\times n},\ \mathbf{S} \leftarrow \mathbb{Z}_q^{n\times m\log q},\ \mathbf{E} \leftarrow \chi^{m\times m\log q}\ \mathbf{r}_i \leftarrow \chi^{m\log q}$$

We stress that the simulator receives a random $\mathbf{A}$ but not the corresponding trapdoor. Indeed, under the LWE assumption, there does not exist an efficient simulator for the naive construction. In more detail, the simulator is required on input $(\mathbf{A}, \{\hat{\mathbf{b}}_i\}_{i\in[Q]})$ to output $(\{\mathbf{r}_i\}_{i\in[Q]}, \mathbf{B}, \{\tilde{\mathbf{s}}_i\}_{i\in[Q]})$ such that

$$(\{\mathbf{r}_i\}_{i\in[Q]}, \mathbf{A}\mathbf{S} + \mathbf{E}, \{\mathbf{S}\mathbf{r}_i\}_{i\in[Q]}) \approx_c (\{\mathbf{r}_i\}_{i\in[Q]}, \mathbf{B}, \{\hat{\mathbf{s}}_i + \tilde{\mathbf{s}}_i\}_{i\in[Q]})$$

We claim that checking whether $\mathbf{B}\mathbf{r}_i \approx \hat{\mathbf{b}}_i + \mathbf{A}\tilde{\mathbf{s}}_i$ yields a distinguisher for whether $(\mathbf{A}, \{\hat{\mathbf{b}}_i\}_{i\in[Q]})$ is drawn from LWE versus uniform distribution. The proof relies on the fact that given $(\{\mathbf{r}_i\}_{i\in[Q]}, \{\mathbf{S}\mathbf{r}_i\}_{i\in[Q]})$ for $Q \gg m$, we can solve for $\mathbf{S}$ via Gaussian elimination, which means that the matrix $\mathbf{B}$ must be of the form

$\mathbf{A}\mathbf{S}_0 + \mathbf{E}_0$ and therefore any $\hat{\mathbf{b}}_i$ that passes the check satisfies $\hat{\mathbf{b}}_i \approx \mathbf{A}(\mathbf{S}_0\mathbf{r}_i - \tilde{\mathbf{s}}_i)$. Note that the LWE distinguisher works even if it does not know $\mathbf{S}_0, \mathbf{E}_0$.

6.2 Heuristic Construction

We now give our heuristic construction of an oblivious LWE sampler. Let n, m, q be some parameters and $\chi, \chi_{\mathsf{prf}}, \hat{\chi}$ be distributions over $\mathbb{Z}$ that are $B, B_{\mathsf{prf}}, \hat{B}$ bounded respectively. Let D be an algorithms that samples tuples $(\mathbf{s}, \mathbf{e})$ where $\mathbf{s} \leftarrow \mathbb{Z}_q^n$ and $\mathbf{e} \leftarrow \chi_{\mathsf{prf}}^m$. Assume that D uses v random coins, and for $r \in \{0,1\}^v$ define $(\mathbf{s}, \mathbf{e}) = \mathsf{D}(r)$ to be the output of D with randomness r. Let $\mathsf{PRF} : \{0,1\}^\lambda \times \{0,1\}^* \to \{0,1\}^v$ be a pseudorandom function. We rely on the homomorphic commitment algorithms $\mathsf{Com}, \mathsf{Eval}^q, \mathsf{Eval}^q_{\mathsf{open}}$ with parameters n, m, q, χ from Sect. 4.

Construction 4. *We define the oblivious LWE sampler as follows:*

- $\mathsf{CRS} \leftarrow \mathsf{crsGen}(1^\lambda, 1^Q)$*:* $\mathsf{CRS} := (\mathsf{CRS}_1, \ldots, \mathsf{CRS}_Q)$ *where* $\mathsf{CRS}_i \leftarrow \mathbb{Z}_q^m$.
- $\mathsf{pub} \leftarrow \mathsf{Init}(\mathbf{A})$*: Sample a PRF key* $k \leftarrow \{0,1\}^\lambda$ *and set a flag* $\beta := 0$*. Set* $\mathsf{pub} := (\mathbf{A}, \mathbf{C})$ *where* $\mathbf{C} \leftarrow \mathsf{Com}(\mathbf{A}, (k, \beta))$.
- $\mathbf{b}_i = \mathsf{Sample}(\mathsf{CRS}, \mathsf{pub}, i)$*: Let* $g_{i,\mathsf{CRS}_i,\mathbf{A}} : \{0,1\}^{\lambda+1} \to \mathbb{Z}_q^m$ *be a circuit that contains the values* $(i, \mathbf{A}, \mathsf{CRS}_i)$ *hard-coded and performs the computation:*

$$g_{i,\mathsf{CRS}_i,\mathbf{A}}(k, \beta): \text{ Let } (\mathbf{s}_i^{\mathsf{prf}}, \mathbf{e}_i^{\mathsf{prf}}) = \mathsf{D}(\mathsf{PRF}(k, i)). \text{ Output } \mathbf{A}\mathbf{s}_i^{\mathsf{prf}} + \mathbf{e}_i^{\mathsf{prf}} + \beta \cdot \mathsf{CRS}_i.$$

  Output $\mathbf{b}_i = \mathsf{Eval}^q(g_{i,\mathsf{CRS}_i,\mathbf{A}}, \mathbf{C})$.
- $(\mathsf{CRS}, \mathsf{pub}, \{\tilde{\mathbf{s}}_i\}_{i\in[Q]}) \leftarrow \mathsf{Sim}(1^\lambda, 1^Q, \mathbf{A}, \{\hat{\mathbf{b}}_i\}_{i\in[Q]})$*: Set* $\mathsf{CRS} := (\hat{\mathbf{b}}_1, \ldots, \hat{\mathbf{b}}_Q)$*. Set the flag* $\beta := 1$ *and* $\mathsf{pub} := (\mathbf{A}, \mathbf{C})$ *for* $\mathbf{C} = \mathsf{Com}((k, \beta); \mathbf{R}, \mathbf{E})$ *where* $\mathbf{R}, \mathbf{E}$ *is the randomness of the commitment. Let* $(\mathbf{r}_i^{\mathsf{eval}}, \mathbf{e}_i^{\mathsf{eval}}) = \mathsf{Eval}^q_{\mathsf{open}}(g_{i,\mathsf{CRS}_i,\mathbf{A}}, \mathbf{A}, (k, \beta), \mathbf{R}, \mathbf{E})$ *and* $(\mathbf{s}_i^{\mathsf{prf}}, \mathbf{e}_i^{\mathsf{prf}}) = \mathsf{D}(\mathsf{PRF}(k, i))$*. Set* $\tilde{\mathbf{s}}_i = \mathbf{r}_i^{\mathsf{eval}} + \mathbf{s}_i^{\mathsf{prf}}$.

Form of Samples $\mathbf{b}_i$. Let us examine this construction in more detail and see what the samples $\mathbf{b}_i$ look like.

In the real case, where $\mathsf{pub} \leftarrow \mathsf{Init}(\mathbf{A})$, we have $\mathsf{pub} = (\mathbf{A}, \mathbf{C})$ where $\mathbf{C} = \mathsf{Com}(\mathbf{A}, (k, 0); (\mathbf{R}, \mathbf{E}))$. For $\mathbf{b}_i = \mathsf{Sample}(\mathsf{CRS}, \mathsf{pub}, i)$ we can write

$$\mathbf{b}_i = \mathbf{A} \underbrace{(\mathbf{r}_i^{\mathsf{eval}} + \mathbf{s}_i^{\mathsf{prf}})}_{\mathbf{s}_i} + \underbrace{(\mathbf{e}_i^{\mathsf{eval}} + \mathbf{e}_i^{\mathsf{prf}})}_{\mathbf{e}_i} \tag{5}$$

where $(\mathbf{s}_i^{\mathsf{prf}}, \mathbf{e}_i^{\mathsf{prf}}) = \mathsf{D}(\mathsf{PRF}(k, i))$ are sampled using the PRF and $(\mathbf{r}_i^{\mathsf{eval}}, \mathbf{e}_i^{\mathsf{eval}}) = \mathsf{Eval}^q_{\mathsf{open}}(g_i, \mathbf{A}, (k, 0), \mathbf{R}, \mathbf{E})$ come from the homomorphic evaluation.

In the simulated case, where $\mathsf{CRS}, \mathsf{pub}$ are chosen by the simulator, we have $\mathsf{pub} = (\mathbf{A}, \mathbf{C})$ where $\mathbf{C} = \mathsf{Com}(\mathbf{A}, (k, 1); (\mathbf{R}, \mathbf{E}))$ and $\mathsf{CRS}_i = \hat{\mathbf{b}}_i = \mathbf{A}\hat{\mathbf{s}}_i + \hat{\mathbf{e}}_i$. For $\mathbf{b}_i = \mathsf{Sample}(\mathsf{CRS}, \mathsf{pub}, i)$ we can write

$$\mathbf{b}_i = \mathbf{A}\underbrace{(\overbrace{\mathbf{r}_i^{\mathsf{eval}} + \mathbf{s}_i^{\mathsf{prf}}}^{\tilde{\mathbf{s}}_i} + \hat{\mathbf{s}}_i)}_{\mathbf{s}_i} + \underbrace{(\overbrace{\mathbf{e}_i^{\mathsf{eval}} + \mathbf{e}_i^{\mathsf{prf}}}^{\tilde{\mathbf{e}}_i} + \hat{\mathbf{e}}_i)}_{\mathbf{e}_i} \tag{6}$$

where $(\mathbf{s}_i^{\mathsf{prf}}, \mathbf{e}_i^{\mathsf{prf}}) = \mathsf{D}(\mathsf{PRF}(k, i))$ are sampled using the PRF and $(\mathbf{r}_i^{\mathsf{eval}}, \mathbf{e}_i^{\mathsf{eval}}) = \mathsf{Eval}^q_{\mathsf{open}}(g_{i,\mathsf{CRS}_i,\mathbf{A}}, \mathbf{A}, (k, 0), \mathbf{R}, \mathbf{E})$ come from the homomorphic evaluation.

Correctness. Equation 5 implies that the scheme satisfies the correctness of an $n, m, q, \hat{\chi}, B_{\mathsf{OLWE}}$ oblivious LWE sampler, where B_{OLWE} is a bound $||\mathbf{e}_i||_\infty$. In particular, $B \leq B_{\mathsf{prf}} + B \cdot m^{O(t)}$, where t is the depth of the circuit $g_{i,\mathsf{CRS}_i,\mathbf{A}}$ (which is dominated by the depth of the PRF).

6.3 Security Under a New Conjecture

The security of our heuristic oblivious sampler boils down to the indistinguishability of the real and simulated distributions, which is captured by the following conjecture:

Conjecture 1 (HPLS Conjecture). For $\beta \in \{0, 1\}$, let us define the distribution $\mathsf{DIST}(\beta)$ over

$$(\{\hat{\mathbf{b}}_i = \mathbf{A}\hat{\mathbf{s}}_i + \hat{\mathbf{e}}_i\}_{i\in[Q]}, \mathbf{A}, \mathbf{C}, \{\mathbf{s}_i = \mathbf{r}_i^{\mathsf{eval}} + \mathbf{s}_i^{\mathsf{prf}} + \beta \cdot \hat{\mathbf{s}}_i\}_{i\in[Q]})$$

where

- $\mathbf{A} \leftarrow \mathbb{Z}_q^{m\times n}, \hat{\mathbf{s}}_i \leftarrow \mathbb{Z}_q^n, \hat{\mathbf{e}}_i \leftarrow \chi^m$, $\hat{\mathbf{b}}_i := \mathbf{A}\hat{\mathbf{s}}_i + \hat{\mathbf{e}}_i$.
- $k \leftarrow \{0,1\}^\lambda$, $(\mathbf{C} = \mathbf{A} \cdot \mathbf{R} + \mathbf{E} + (k, \beta) \otimes \mathbf{G}) \leftarrow \mathsf{Com}(\mathbf{A}, (k, \beta); (\mathbf{R}, \mathbf{E}))$
- $(\mathbf{s}_i^{\mathsf{prf}}, \mathbf{e}_i^{\mathsf{prf}}) := \mathsf{D}(\mathsf{PRF}(k, i))$, $(\mathbf{r}_i^{\mathsf{eval}}, \mathbf{e}_i^{\mathsf{eval}}) := \mathsf{Eval}^q_{\mathsf{open}}(g_{i,\hat{\mathbf{b}}_i,\mathbf{A}}, \mathbf{A}, (k, \beta), \mathbf{R}, \mathbf{E})$ where

$$g_{i,\hat{\mathbf{b}}_i,\mathbf{A}}(k, \beta): \text{ Let } (\mathbf{s}_i^{\mathsf{prf}}, \mathbf{e}_i^{\mathsf{prf}}) = \mathsf{D}(\mathsf{PRF}(k, i)). \text{ Output } \mathbf{A}\mathbf{s}_i^{\mathsf{prf}} + \mathbf{e}_i^{\mathsf{prf}} + \beta \cdot \hat{\mathbf{b}}_i.$$

- $\mathbf{s}_i := (\mathbf{r}_i^{\mathsf{eval}} + \mathbf{s}_i^{\mathsf{prf}} + \beta \cdot \hat{\mathbf{s}}_i)$.

The (sub-exponential) *homomorphic pseudorandom LWE samples (HPLS)* conjecture with parameters $(n, m, q, \chi, \hat{\chi}, \chi_{\mathsf{prf}})$ and pseudodrandom function PRF says that the distributions $\mathsf{DIST}(0)$ and $\mathsf{DIST}(1)$ are (sub-exponentially) computationally indistinguishable.

When we do not specify parameters, we assume the conjecture holds for some choice of PRF and any choices of $n, q, \chi, \hat{\chi}$ and any polynomial m, such that $LWE_{n,q,\chi}$ and $LWE_{n,q,\hat{\chi}}$ assumptions hold and χ_{prf} smudges out error of size $\hat{B}+B\cdot m^{O(t)}$, where t is the depth of the circuit $g_{i,\mathsf{CRS}_i,\mathbf{A}}$ (which is dominated by the depth of the PRF).

Observations. We begin with two simple observations about the conjecture:

- The distribution $\mathsf{DIST}(\beta)$ satisfies the following consistency check for both $\beta = 0$ and $\beta = 1$, namely

$$\mathsf{Eval}^q(g_{i,\mathbf{A}\hat{\mathbf{s}}_i+\hat{\mathbf{e}}_i,\mathbf{A}}, \mathbf{C}) \approx \mathbf{A}\mathbf{s}_i$$

  This means that we cannot rely on homomorphic evaluation to distinguish between the two distributions. In addition, note that the distinguisher can compute

$$\mathbf{e}_i := \mathsf{Eval}^q(g_{i,\mathbf{A}\hat{\mathbf{s}}_i+\hat{\mathbf{e}}_i,\mathbf{A}}, \mathbf{C}) - \mathbf{A}\mathbf{s}_i = \mathbf{e}_i^{\mathsf{eval}} + \mathbf{e}_i^{\mathsf{prf}} + \beta \cdot \hat{\mathbf{e}}_i$$

- If we omit $\mathbf{r}_i^{\mathsf{eval}}$ from $\mathbf{s}_i$, then indistinguishability follows from standard assumptions. Concretely, under the LWE assumption and security of PRF, we have:

$$\begin{aligned}&(\{\mathbf{A}\hat{\mathbf{s}}_i + \hat{\mathbf{e}}_i\}_{i\in[Q]}, \mathbf{A}, \mathbf{C}, \{\mathbf{s}_i^{\mathsf{prf}}, \mathbf{e}_i^{\mathsf{prf}}\}_{i\in[Q]})\\ \approx_c\ &(\{\mathbf{A}\hat{\mathbf{s}}_i + \hat{\mathbf{e}}_i\}_{i\in[Q]}, \mathbf{A}, \mathbf{C}, \{\mathbf{s}_i^{\mathsf{prf}} + \hat{\mathbf{s}}_i, \mathbf{e}_i^{\mathsf{prf}} + \hat{\mathbf{e}}\}_{i\in[Q]})\end{aligned}$$

  By privacy of Com, we can replace $\mathbf{C}$ with a commitment to $\mathbf{0}$, and then security follows from PRF security plus noise smudging. In particular $\mathbf{e}_i^{\mathsf{prf}}$ smudges out $\hat{\mathbf{e}}_i$.

That is, the non-standard/heuristic nature of Conjecture 1 arises from (1) the interaction and potential correlations between $\mathbf{r}_i^{\mathsf{eval}}$ and $\mathbf{s}_i^{\mathsf{prf}}$ (and between $\mathbf{e}_i^{\mathsf{eval}}$ and $\mathbf{e}_i^{\mathsf{prf}}$), and (2) the fact that giving out $\mathbf{C} = \mathsf{Com}(\mathbf{A}, (k, \beta))$ introduces circularity between the PRF key and the commitment randomness – commitment security is needed to ensure PRF security by making sure that the PRF key is hidden by the commitment, while at the same time the PRF security is needed to ensure commitment security by making sure that the values $\mathbf{s}_i^{\mathsf{prf}}, \mathbf{e}_i^{\mathsf{prf}}$ mask any information about the commitment randomness contained in $\mathbf{r}_i^{\mathsf{eval}}, \mathbf{e}_i^{\mathsf{eval}}$. We defer further discussion on the conjecture, its plausibility, and analysis of zeroizing attacks to the full version.

Oblivious LWE Sampling from the New Conjecture. We now that the conjecture implies the (sub-exponential security) of our oblivious LWE sampler in Definition 1.

Lemma 3. *Under the homomorphic pseudorandom LWE samples (HPLS) conjecture (Conjecture 1) (with sub-exponential security), the oblivious sampler construction is (sub-exponentially) secure.*

We defer the proof to the full version.

7 Q-SIM Functional Encodings from Oblivious Sampling

We construct a Q-SIM functional encoding scheme $(\mathsf{crsGen}, \mathsf{Enc}, \mathsf{Open}, \mathsf{Dec})$ for bounded-depth circuits $\mathcal{F}_{\ell,m,t}$ from LWE and an oblivious LWE sampler $(\mathsf{OLWE.crsGen}, \mathsf{Init}, \mathsf{Sample})$.

Construction 5

- $\mathsf{crsGen}(1^\lambda, 1^Q, \mathcal{F}_{\ell,m,t})$. *Output* $\mathsf{OLWE.crsGen}(1^\lambda, 1^Q)$.
- $\mathsf{Enc}(\mathsf{CRS}, \mathbf{x})$*: Sample*

$$(\mathbf{A}, \mathsf{td}) \leftarrow \mathsf{TrapGen}(1^n, 1^m, q), \mathsf{pub} \leftarrow \mathsf{Init}(\mathsf{CRS}, \mathbf{A}), \mathbf{R} \leftarrow \mathbb{Z}_q^{n \times \ell m \log q}, \mathbf{E} \leftarrow \chi^{m \times \ell m \log q}$$

  Compute $\mathbf{C} := \mathsf{pFHC.Com}(\mathbf{A}, \mathbf{x}; \mathbf{R}, \mathbf{E})$ *and output* $(\mathsf{pub}, \mathbf{A}, \mathbf{C})$.
- $\mathsf{Open}(f^i, \mathbf{x})$*: Compute*

$$(\mathbf{r}_{f^i}, \mathbf{e}_{f^i}) := \mathsf{pFHC.Eval}_{\mathsf{open}}(f^i, \mathbf{A}, \mathbf{x}, \mathbf{R}, \mathbf{E}), \quad \mathbf{b}_i := \mathsf{Sample}(\mathsf{CRS}, \mathsf{pub}, i), \quad \mathbf{s}_i := \mathsf{LWESolve}_{\mathsf{td}}(\mathbf{b}_i)$$

  and output $\mathbf{d}_i := \mathbf{r}_{f^i} + \mathbf{s}_i \in \mathbb{Z}_q^n$.
- $\mathsf{Dec}(f^i, (\mathsf{pub}, \mathbf{A}, \mathbf{C}), \mathbf{d}_i)$*: Compute*

$$\mathbf{c}_{f^i} := \mathsf{pFHC.Eval}(f, \mathbf{C}), \quad \mathbf{b}_i := \mathsf{Sample}(\mathsf{CRS}, \mathsf{pub}, i)$$

  and output $\mathbf{y}_i := \mathsf{round}_{q/2}(\mathbf{c}_{f^i} + \mathbf{b}_i - \mathbf{A}\mathbf{d}_i) \in \{0,1\}^m$.

Theorem 6. *Under the LWE assumption and the existence of a* (n, m, q, χ, B) *oblivious LWE sampler, the construction above is a* Q-SIM *functional encoding.*

We defer the proof to the full version.

8 IO from Functional Encodings

See the full version for how to construct XiO from functional encodings. We then rely on the work of [LPST16a], which shows that (sub-exponentially secure) XiO + LWE implies iO. Below, we summarize the main results.

Theorem 7. *The existence of (sub-exponentially secure) functional encoding implies (sub-exponenitally secure) XiO. In particular, sub-exponentially secure functional encodings and sub-exponential security of LWE imply the existence of iO.*

Corollary 1. *Assuming that there exists a sub-exponentially secure oblivious LWE sampler and that the sub-exponentially secure LWE assumption holds, there exists iO.*

Corollary 2. *Assuming the sub-exponential security of Conjecture 1 and the sub-exponential security of LWE, there exists iO.*

Acknowledgments. We thank Yilei Chen and Vinod Vaikuntanathan for insightful discussions on cryptanalysis and bootstrapping.

References

[ACI+20] Agrikola, T., Couteau, G., Ishai, Y., Jarecki, S., Sahai, A.: On pseudorandom encodings. In: TCC, Cryptology ePrint Archive, Report 2020/445 (2020). https://eprint.iacr.org/2020/445

[ADGM17] Apon, D., Döttling, N., Garg, S., Mukherjee, P.: Cryptanalysis of indistinguishability obfuscations of circuits over GGH13. In: Chatzigiannakis, I., Indyk, P., Kuhn, F., Muscholl, A. (eds.) ICALP 2017, LIPIcs, Schloss Dagstuhl, vol. 80, pp. 38:1–38:16, July 2017

[Agr19] Agrawal, S.: Indistinguishability obfuscation without multilinear maps: new methods for bootstrapping and instantiation. In: Ishai and Rijmen [IR19], pp. 191–225

[AJ15] Ananth, P., Jain, A.: Indistinguishability obfuscation from compact functional encryption. In: Gennaro, R., Robshaw, M. (eds.) CRYPTO 2015, Part I. LNCS, vol. 9215, pp. 308–326. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-47989-6_15

[AJL+12] Asharov, G., Jain, A., López-Alt, A., Tromer, E., Vaikuntanathan, V., Wichs, D.: Multiparty computation with low communication, computation and interaction via threshold FHE. In: Pointcheval and Johansson [PJ12], pp. 483–501

[AJL+19] Ananth, P., Jain, A., Lin, H., Matt, C., Sahai, A.: Indistinguishability obfuscation without multilinear maps: new paradigms via low degree weak pseudorandomness and security amplification. In: Boldyreva and Micciancio [BM19], pp. 284–332

[Ajt96] Ajtai, M.: Generating hard instances of lattice problems (extended abstract). In: 28th ACM STOC, pp. 99–108. ACM Press, May 1996

[AP20] Agrawal, S., Pellet-Mary, A.: Indistinguishability obfuscation without maps: attacks and fixes for noisy linear FE. In: Canteaut and Ishai [CI20], pp. 110–140

[BDGM19] Brakerski, Z., Döttling, N., Garg, S., Malavolta, G.: Leveraging linear decryption: Rate-1 fully-homomorphic encryption and time-lock puzzles. In: Hofheinz and Rosen [HR19], pp. 407–437

[BDGM20a] Brakerski, Z., Döttling, N., Garg, S., Malavolta, G.: Candidate iO from homomorphic encryption schemes. In: Canteaut and Ishai [CI20], pp. 79–109

[BDGM20b] Brakerski, Z., Döttling, N., Garg, S., Malavolta, G.: Factoring and pairings are not necessary for IO: Circular-secure LWE suffices. Cryptology ePrint Archive, Report 2020/1024 (2020)

[BGG+14] Boneh, D., et al.: Fully key-homomorphic encryption, arithmetic circuit ABE and compact garbled circuits. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 533–556. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-55220-5_30

[BGI+01] Barak, B., et al.: On the (Im)possibility of obfuscating programs. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 1–18. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-44647-8_1

[BGMZ18] Bartusek, J., Guan, J., Ma, F., Zhandry, M.: Return of GGH15: provable security against zeroizing attacks. In: Beimel, A., Dziembowski, S. (eds.) TCC 2018, Part II. LNCS, vol. 11240, pp. 544–574. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-03810-6_20

[BIJ+20] Bartusek, J., Ishai, Y., Jain, A., Ma, F., Sahai, A., Zhandry, M.: Affine determinant programs: a framework for obfuscation and witness encryption. In: Vidick, T. (ed.) ITCS 2020, LIPIcs, vol. 151, pp. 82:1–82:39, January 2020

[BM19] Boldyreva, A., Micciancio, D. (eds.): CRYPTO 2019, Part III. LNCS, vol. 11694. Springer, Heidelberg, August 2019

[BRF13] Boneh, D., Roughgarden, T., Feigenbaum, J. (eds.) 45th ACM STOC. ACM Press, June 2013

[BTVW17] Brakerski, Z., Tsabary, R., Vaikuntanathan, V., Wee, H.: Private constrained PRFs (and More) from LWE. In: Kalai, Y., Reyzin, L. (eds.) TCC 2017, Part I. LNCS, vol. 10677, pp. 264–302. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70500-2_10

[BV15] Bitansky, N., Vaikuntanathan, V.: Indistinguishability obfuscation from functional encryption. In: Guruswami, V. (ed.) 56th FOCS, pp. 171–190. IEEE Computer Society Press, October 2015

[CCH+19] Cheon, J.H., Cho, W., Hhan, M., Kim, J., Lee, C.: Statistical zeroizing attack: cryptanalysis of candidates of BP obfuscation over GGH15 multilinear map. In: Boldyreva and Micciancio [BM19], pp. 253–283

[CGH17] Chen, Y., Gentry, C., Halevi, S.: Cryptanalyses of candidate branching program obfuscators. In: Coron, J.-S., Nielsen, J.B. (eds.) EUROCRYPT 2017, Part III. LNCS, vol. 10212, pp. 278–307. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-56617-7_10

[CHL+15] Cheon, J.H., Han, K., Lee, C., Ryu, H., Stehlé, D.: Cryptanalysis of the multilinear map over the integers. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015, Part I. LNCS, vol. 9056, pp. 3–12. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-46800-5_1

[CHVW19] Chen, Y., Hhan, M., Vaikuntanathan, V., Wee, H.: Matrix PRFs: constructions, attacks, and applications to obfuscation. In: Hofheinz, D., Rosen, A. (eds.) TCC 2019, Part I. LNCS, vol. 11891, pp. 55–80. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-36030-6_3

[CI20] Canteaut, A., Ishai, Y. (eds.): EUROCRYPT 2020, Part I. LNCS, vol. 12105. Springer, Heidelberg, May 2020

[CLLT16] Coron, J.S., Lee, M.S., Lepoint, T., Tibouchi, M.: Cryptanalysis of GGH15 multilinear maps. In: Robshaw and Katz [RK16], pp. 607–628

[CLLT17] Coron, J.-S., Lee, M.S., Lepoint, T., Tibouchi, M.: Zeroizing attacks on indistinguishability obfuscation over CLT13. In: Fehr, S. (ed.) PKC 2017, Part I. LNCS, vol. 10174, pp. 41–58. Springer, Heidelberg (2017). https://doi.org/10.1007/978-3-662-54365-8_3

[CVW18] Chen, Y., Vaikuntanathan, V., Wee, H.: GGH15 beyond permutation branching programs: proofs, attacks, and candidates. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018, Part II. LNCS, vol. 10992, pp. 577–607. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-96881-0_20

[DKR15] Dachman-Soled, D., Katz, J., Rao, V.: Adaptively secure, universally composable, multiparty computation in constant rounds. In: Dodis and Nielsen [DN15], pp. 586–613

[DN15] Dodis, Y., Nielsen, J.B. (eds.): TCC 2015, Part II. LNCS, vol. 9015. Springer, Heidelberg, March 2015

[FRS17] Fernando, R., Rasmussen, P.M.R., Sahai, A.: Preventing CLT attacks on obfuscation with linear overhead. In: Takagi, T., Peyrin, T. (eds.) ASIACRYPT 2017. LNCS, vol. 10626, pp. 242–271. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70700-6_9

[GGH13a] Garg, S., Gentry, C., Halevi, S.: Candidate multilinear maps from ideal lattices. In: Johansson, T., Nguyen, P.Q. (eds.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 1–17. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38348-9_1

[GGH+13b] Garg, S., Gentry, C., Halevi, S., Raykova, M., Sahai, A., Waters, B.: Candidate indistinguishability obfuscation and functional encryption for all circuits. In: 54th FOCS, pp. 40–49. IEEE Computer Society Press, October 2013

[GGH15] Gentry, C., Gorbunov, S., Halevi, S.: Graph-induced multilinear maps from lattices. In: Dodis and Nielsen [DN15], pp. 498–527

[GH19] Gentry, C., Halevi, S.: Compressible FHE with applications to PIR. In: Hofheinz and Rosen [HR19], pp. 438–464

[GJK18] Gentry, C., Jutla, C.S., Kane, D.: Obfuscation using tensor products. Cryptology ePrint Archive, Report 2018/756 (2018). https://eprint.iacr.org/2018/756

[GJLS20] Gay, R., Jain, A., Lin, H., Sahai, A.: Indistinguishability obfuscation from simple-to-state hard problems: new assumptions, new techniques, and simplification. Cryptology ePrint Archive, Report 2020/764 (2020). https://eprint.iacr.org/2020/764

[GKP+13] Goldwasser, S., Kalai, Y., Popa, R.A., Vaikuntanathan, V., Zeldovich, N.: Reusable garbled circuits and succinct functional encryption. In: Boneh et al. [BRF13], pp. 555–564

[GLW14] Gentry, C., Lewko, A., Waters, B.: Witness encryption from instance independent assumptions. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part I. LNCS, vol. 8616, pp. 426–443. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-44371-2_24

[GP20] Gay, R., Pass, R.: Indistinguishability obfuscation from circular security. Cryptology ePrint Archive, Report 2020/1010 (2020)

[GR07] Goldwasser, S., Rothblum, G.N.: On best-possible obfuscation. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 194–213. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-70936-7_11

[GSW13] Gentry, C., Sahai, A., Waters, B.: Homomorphic encryption from learning with errors: conceptually-simpler, asymptotically-faster, attribute-based. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013. LNCS, vol. 8042, pp. 75–92. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40041-4_5

[GVW13] Gorbunov, S., Vaikuntanathan, V., Wee, H.: Attribute-based encryption for circuits. In: Boneh et al. [BRF13], pp. 545–554

[GVW15] Gorbunov, S., Vaikuntanathan, V., Wichs, D.: Leveled fully homomorphic signatures from standard lattices. In: Servedio, R.A., Rubinfeld, R. (eds.) 47th ACM STOC, pp. 469–477. ACM Press, June 2015

[HR19] Hofheinz, D., Rosen, A. (eds.): TCC 2019, Part II. LNCS, vol. 11892. Springer, Heidelberg, December 2019

[IKOS10] Ishai, Y., Kumarasubramanian, A., Orlandi, C., Sahai, A.: On invertible sampling and adaptive security. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 466–482. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-17373-8_27

[IR19] Ishai, Y., Rijmen, V. (eds.): EUROCRYPT 2019, Part I. LNCS, vol. 11476. Springer, Heidelberg, May 2019

[JLMS19] Jain, A., Lin, H., Matt, C., Sahai, A.: How to leverage hardness of constant-degree expanding polynomials overa $\mathbb{R}$ to build $i\mathcal{O}$. In: Ishai and Rijmen [IR19], pp. 251–281

[JLS20] Jain, A., Lin, H., Sahai, A.: Indistinguishability obfuscation from well-founded assumptions. Cryptology ePrint Archive, Report 2020/1003 (2020)

[KS17] Katz, J., Shacham, H. (eds.): CRYPTO 2017, Part I. LNCS, vol. 10401. Springer, Heidelberg, August 2017

[Lin16] Lin, H.: Indistinguishability obfuscation from constant-degree graded encoding schemes. In: Fischlin, M., Coron, J.-S. (eds.) EUROCRYPT 2016, Part I. LNCS, vol. 9665, pp. 28–57. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49890-3_2

[Lin17] Lin, H.: Indistinguishability obfuscation from SXDH on 5-linear maps and locality-5 PRGs. In: Katz and Shacham [KS17], pp. 599–629

[LPST16a] Lin, H., Pass, R., Seth, K., Telang, S.: Indistinguishability obfuscation with non-trivial efficiency. In: Cheng, C.-M., Chung, K.-M., Persiano, G., Yang, B.-Y. (eds.) PKC 2016, Part II. LNCS, vol. 9615, pp. 447–462. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49387-8_17

[LPST16b] Lin, H., Pass, R., Seth, K., Telang, S.: Output-compressing randomized encodings and applications. In: Kushilevitz, E., Malkin, T. (eds.) TCC 2016-A, Part I. LNCS, vol. 9562, pp. 96–124. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49096-9_5

[LT17] Lin, H., Tessaro, S.: Indistinguishability obfuscation from trilinear maps and block-wise local PRGs. In: Katz and Shacham [KS17], pp. 630–660

[LV16] Lin, H., Vaikuntanathan, V.: Indistinguishability obfuscation from DDH-like assumptions on constant-degree graded encodings. In: Dinur, I. (ed.) 57th FOCS, pp. 11–20. IEEE Computer Society Press, October 2016

[MP12] Micciancio, D., Peikert, C.: Trapdoors for lattices: simpler, tighter, faster, smaller. In: Pointcheval and Johansson [PJ12], pp. 700–718

[MSZ16] Miles, E., Sahai, A., Zhandry, M.: Annihilation attacks for multilinear maps: cryptanalysis of indistinguishability obfuscation over GGH13. In: Robshaw and Katz [RK16], pp. 629–658

[MW16] Mukherjee, P., Wichs, D.: Two round multiparty computation via multi-key FHE. In: Fischlin, M., Coron, J.-S. (eds.) EUROCRYPT 2016, Part II. LNCS, vol. 9666, pp. 735–763. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49896-5_26

[Pel18] Pellet-Mary, A.: Quantum attacks against indistinguishablility obfuscators proved secure in the weak multilinear map model. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018, Part III. LNCS, vol. 10993, pp. 153–183. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-96878-0_6

[PJ12] Pointcheval, D., Johansson, T. (eds.): EUROCRYPT 2012. LNCS, vol. 7237. Springer, Heidelberg, April 2012

[PS19] Peikert, C., Shiehian, S.: Noninteractive zero knowledge for NP from (plain) learning with errors. In: Boldyreva, A., Micciancio, D. (eds.) CRYPTO 2019, Part I. LNCS, vol. 11692, pp. 89–114. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-26948-7_4

[Reg05] Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Gabow, H.N., Fagin, R. (eds.) 37th ACM STOC, pp. 84–93. ACM Press, May 2005

[RK16] Robshaw, M., Katz, J. (eds.): CRYPTO 2016, Part II. LNCS, vol. 9815. Springer, Heidelberg, August 2016