Key-Dependent Message Security: Generic Amplification and Completeness

Benny Applebaum*

February 4, 2011

Abstract

Key-dependent message (KDM) secure encryption schemes provide secrecy even when the attacker sees encryptions of messages related to the secret-key sk. Namely, the scheme should remain secure even when messages of the form $f(\mathsf{sk})$ are encrypted, where f is taken from some function class $\mathcal{F}$. A KDM *amplification* procedure takes an encryption scheme which satisfies $\mathcal{F}$-KDM security and boost it into a $\mathcal{G}$-KDM secure scheme, where the function class $\mathcal{G}$ should be richer than $\mathcal{F}$. It was recently shown by Brakerski et al. (TCC 2011) and Barak et al. (EUROCRYPT 2010), that a strong form of amplification is possible, provided that the underlying encryption scheme satisfies some special additional properties.

In this work, we prove the first *generic* KDM amplification theorem which relies solely on the KDM security of the underlying scheme without making any other assumptions. Specifically, we show that an elementary form of KDM security against functions in which each output bit either copies or flips a single bit of the key (aka *projections*) can be amplified into KDM security with respect to any function family that can be computed in arbitrary fixed polynomial-time. Furthermore, our amplification theorem and its proof are insensitive to the exact setting of KDM security, and they hold in the presence of multiple-keys and in the symmetric-key/public-key and the CPA/CCA cases. As a result, we can amplify the security of all known KDM constructions, including ones that could not be amplified before.

Finally, we study the minimal conditions under which full-KDM security (with respect to all functions) can be achieved. We show that under strong notion of KDM security, the existence of cyclic-secure fully-homomorphic encryption is not only sufficient for full-KDM security, as shown by Barak et al., but also necessary. On the other hand, we observe that for standard KDM security, this condition can be relaxed by adopting Gentry's bootstrapping technique (STOC 2009) to the KDM setting.

1 Introduction

The study of secure encryption scheme is perhaps the most central subject in cryptography. Since the discovery of semantic security [GM84] till the formulation of CCA-security [NY90, RS91, DDN91], modern cryptography has successfully developed increasingly stronger notions of security providing secrecy in highly adversarial settings. Still, all these strong notions of security

*School of Electrical Engineering, Tel-Aviv University, benny.applebaum@gmail.com. Work done in part while a postdoc at the Weizmann Institute of Science, supported by Alon and Koshland Fellowships. Preliminary version will appear in Eurocrypt, 2011

guarantee secrecy only as long as the encrypted messages are independent of the secret key. This limitation dates back to the seminal work of Goldwasser and Micali [GM84] who observed that semantic security may not hold if the adversary gets to see an encryption of the secret key. For many years, such usage scenarios were considered as "security bugs" that should be prevented by system designers.

A decade ago, the assumption of independency between the secret key and the encrypted data was challenged by Camenisch and Lysyanskaya [CL01] and Black et al. [BRS02]. Specifically, Camenisch and Lysyanskaya considered schemes that remain secure under a "key cycle" usage, where we have t keys organized in a cycle and each key is encrypted under its left neighbor. A generalization of this notion, called *key-dependent message* (KDM) security, was suggested by Black et al. Informally, an encryption is $\mathsf{KDM}^{(t)}$ secure with respect to a function class $\mathcal{F}$ if security holds even when the adversary can ask for an encryption of the message $M = f(\mathsf{sk}_1, \ldots, \mathsf{sk}_t)$ under the i-th public-key, where $\mathsf{sk}_1, \ldots, \mathsf{sk}_t$ are the secret keys present in the system and f is an arbitrary function in $\mathcal{F}$. This notion of security implies cyclic-security if $\mathcal{F}$ is expressive enough (e.g., contains all "selector" functions), and it becomes strictly stronger when the function class $\mathcal{F}$ grows. Hence, one would like to achieve KDM security while making the function class $\mathcal{F}$ as large as possible.

The notion of KDM security was extensively studied in the past few years in several flavors including the symmetric/public-key and the CPA/CCA settings [CL01, BRS02, HK07, BPS07, BHHO08, CCS09, BDU08, HU08, HH09, ACPS09, BGK11, ABBC10, BHHI10, BG10]. These works were motivated by the fundamental nature of the question as well as by concrete applications including encrypted storage systems (e.g., BitLocker [BHHO08]), anonymous credentials [CL01], and realization of security proofs at the framework of axiomatic security [AR07, BRS02, ABHS09]. (See [BHHO08] for more motivations and details.)

Although much is known today about KDM security both on the positive and negative sides, it is still unclear whether a standard encryption scheme can be transformed into a scheme which provides $\mathsf{KDM}^{(t)}$ security, even with respect to a single key (i.e., $t = 1$) and simple non-trivial function families (e.g., selectors).[1] Hence, it is natural to move forward and explore the possibility of building strong KDM security given a weak form of KDM security as a primitive. This makes sense as today, following the seminal work of Boneh et al. [BHHO08] and its follow-ups [CCS09, ACPS09, BG10], it is known that a basic form of KDM security (with respect to the family of "affine functions") can be achieved in several settings under various concrete cryptographic assumptions. Therefore, following [BGK11] we ask:

> Is there a *generic* transformation which *amplifies* KDM security from a weak family of functions $\mathcal{F}$ to a larger family of functions $\mathcal{G}$?

The two main features of such a procedure are *generality* – the transformation should work with any scheme which satisfies $\mathcal{F}$-KDM security without relying on any other additional property – and large *amplification gap* – ideally, $\mathcal{F}$ is a very simple function class whereas $\mathcal{G}$ is as rich as possible. The question of KDM amplification was recently addressed by Brakerski et al. [BGK11] and Barak et al. [BHHI10], who made an important progress by showing how to amplify the KDM security of several existing schemes. While these works achieve relatively large amplification gap, they fall short of providing full generality as they strongly rely on additional properties of the underlying

[1] It is known that KDM security with respect to sufficiently rich families of functions cannot be based on standard assumptions via fully black-box reductions [HH09]. However, this impossibility result (and its extension in [BHHI10]) does not hold for simple function class (e.g., projections).

scheme (i.e., *simulatable*-KDM security and *entropic*-KDM security – to be defined later). As a concrete example, it is unknown how to use any of these techniques to amplify the KDM-security of the symmetric-key encryption scheme of [ACPS09] which is based on the Learning Parity With Noise (LPN) assumption. (See Section 1.3 for more details about these works and their relation to our approach.)

1.1 Our Results

We give an affirmative answer to the above question by providing the first generic KDM amplification procedure. In particular, we consider the *projection* function class of all functions $f : (\mathsf{sk}_1, \ldots, \mathsf{sk}_t) \mapsto v$ in which each output bit depends on (at most) a single bit of the input. Namely, each output bit v_j is either fixed to a constant or copies/flips an original bit of one of the keys. We show that this elementary function family is *complete* in the following sense:

Theorem 1.1 (Completeness of projections, Informal)**.** *Let $\mathcal{G}$ be any function family which can be computed in some fixed polynomial time. Then, any encryption scheme which satisfies $\mathsf{KDM}^{(t)}$ security with respect to projections can be transformed into a new encryption scheme which is $\mathsf{KDM}^{(t)}$-secure with respect to $\mathcal{G}$.*

Generality. Theorem 1.1 assumes nothing but KDM security regarding the underlying scheme. Furthermore, the theorem (and its surprisingly simple proof) is insensitive to the exact setting of KDM security: it holds for any number of keys (t), and in both symmetric-key/public-key and CPA/CCA settings. In all these cases, the new scheme is proven to be secure exactly in the same setting as the original scheme. This allows us, for example, to amplify the security of the affine-KDM secure scheme of [ACPS09], and obtain the first symmetric-key encryption scheme with strong KDM security based on the LPN assumption.

Large gap. Theorem 1.1 provides a large amplification gap. In fact, this gap can be further expanded as follows. First, we can achieve *length-dependent* KDM security [BHHI10], which means that the target family $\mathcal{G}$ can be taken to be the family of all polynomial-size circuits whose size grows with their input and output lengths via a fixed polynomial rate (e.g., the circuit size is quadratic in the input and output lengths). This family is very powerful and it was shown to be rich enough for most known applications of KDM security [BHHI10].[2] (See Section 3 for details.) In addition, in the case of CPA security (both in the public-key and symmetric-key settings), we can weaken the requirement from the underlying scheme and ask for KDM security with respect to projections with a *single output*: namely, all Boolean functions $f(\mathsf{sk}_1, \ldots, \mathsf{sk}_t) \mapsto b$ which output a single bit of one of the keys, or its negation. This can be extended to the CCA setting via the transformations of [BPS07, CCS09] (though in the public-key setting one has to employ, in addition, non-interactive zero-knowledge proofs).

The relaxation to single-output projections also enables a liberal interface to which we can easily plug previous constructions. For example, one can instantiate our reduction with schemes that enjoy KDM security with respect to affine functions, while almost ignoring technical details

[2] Most of the statements in [BHHI10] refer to the slightly weaker notion of *Bounded KDM security* in which the circuit size grows only as a function of the input via a fixed polynomial rate. However, as observed in [BHHI10, Sec. 6] their construction actually satisfies the stronger definition of *length-dependent* KDM security.

such as the underlying field and its representation. (These details required some effort in previous works. See the appendices in [BGK11, BHHI10, BG10].) This, together with the simple proof of our main theorem, allows to simplify the proofs of [BHHI10, BG10] for the existence of length-dependent KDM secure encryption scheme under the Decisional Diffie-Hellman (DDH) assumption [BHHO08], the Learning With Errors assumptions (LWE) [ACPS09], and the Quadratic Residuosity (QR) assumption [BG10].

Given this completeness theorem, the current status of KDM security resembles the status of other "complete" primitives in cryptography such as one-way functions or oblivious transfer [Rab79, EGL85]: We do not know how to build these primitives from generic weaker assumptions, however, any instantiation of them suffices for an entire world of applications (i.e., all symmetric-key primitives in the case of one-way functions, and generic secure-computation in the case of oblivious transfer, cf. [Gol01, Gol04]).

Beyond length-dependent security. Although length-dependent KDM security seems to suffice for most applications, one can strive for an even stronger notion of security in which the KDM function class contains all functions (or equivalently all functions computable by circuits of *arbitrary* polynomial size). It is somewhat likely that any length-dependent secure scheme actually achieves *full-KDM* security (see the discussion in [BHHI10]). Still, one may want to construct such a scheme in a provably secure way. As a basic feasibility result, it was shown in [BHHI10] that any fully homomorphic encryption scheme [Gen09] which allows to encrypt the secret-key (i.e., "cyclic-secure") is also full-KDM secure. In light of the small number of FHE candidates [Gen09, vDGHV10], and our little understanding of this notion, one may ask whether it is possible to relax this requirement and achieve full-KDM security under weaker assumptions.

We make two simple observations regarding this question. First, we consider the case of simulatable KDM security [BHHI10], in which it should be possible to simulate an encryption of $f(\mathsf{sk})$ given only the corresponding public-key in a way that remains indistinguishable even to someone who knows the secret-key. We show that in this setting the two notions: circular-secure FHE and full-KDM are equivalent. Hence, achieving full-KDM security under a relaxed assumption requires to use non-simulatable constructions.

Our second observation asserts that the bootstrapping technique of Gentry [Gen09] can be used in the KDM setting as well (even for the case of non-simulatable constructions). That is, if one can construct an encryption scheme which guarantees KDM security with respect to circuits whose depth is only slightly larger than the depth of the decryption algorithm, then this scheme is actually fully KDM secure. Unfortunately, all known amplification techniques [BHHI10, BGK11] including the ones in this paper, amplify KDM security at the cost of making the decryption algorithm "deeper". Still, we view this observation as an interesting direction for future research.

1.2 Our Techniques

To formalize the question of KDM amplification, we define the notion of *reduction* between KDM function families $\mathcal{G} \leq_{\mathsf{KDM}} \mathcal{F}$ which means that any scheme that provides KDM security with respect to $\mathcal{F}$ can be transformed (via a fully black-box reduction) to a new scheme that satisfies KDM security with respect to $\mathcal{G}$. We describe a novel way to derive such KDM reductions based on the machinery of *randomized encoding* of functions [IK00, AIK06b]. Before we explain this notion, let us start with the simpler case of *deterministic encoding*.

Say that a function f deterministically encodes a function g if for every x the output of $f(x)$ "encodes" the output of $g(x)$ in the sense that $g(x)$ can be efficiently computed based on $f(x)$ and vice versa. That is, there are two efficiently computable mappings S and R such that $S(g(x)) = f(x)$, and $R(f(x)) = g(x)$. Suppose that we are given a scheme which provides KDM security with respect to the encoding f, and we would like to immunize it against the function g. This can be easily achieved by modifying the encryption scheme as follows: to encrypt a message M we first translate it into the f-encoding by computing $S(M)$, and then encrypt the result under the original encryption scheme. Decryption is done by applying the original decryption algorithm, and then applying the recovery algorithm R to translate the result back to its original form. Observe that an encryption of $g(\mathsf{sk})$ in the new scheme is the same as an encryption of $S(g(\mathsf{sk})) = f(\mathsf{sk})$ under the original scheme. Hence, the KDM security of the new scheme with respect to g reduces to the KDM security of the original scheme with respect to f.

This simple idea provides a direct reduction with very nice structure: any KDM query for the new scheme is translated into a single KDM query for the original scheme. This simple single-query-to-single-query translation leads to high level of generality: the transformation is insensitive to the exact KDM setting (symmetric-key/public-key and CPA/CCA), to the number of keys, and it can be used with respect to large function families $\mathcal{G}$ and $\mathcal{F}$ as long as every function in $\mathcal{G}$ is encoded by some function in $\mathcal{F}$ via a pair of universal mappings S and R. On the down side, one may complain that security was not really *amplified*, as the function g and its encoding f are essentially equivalent. It turns out that this drawback can be easily fixed by letting f be a *randomized* encoding of g.

In the case of randomized encoding (RE), the function $f(x;r)$ depends not only on x but also on an additional random input r. For every fixed x, the output of $f(x;r)$ is now viewed as a distribution (induced by a random choice of r) which should encode the value of $g(x)$. Namely, there are two efficiently computable randomized mappings S and R such that for every x: (1) the distribution $S(g(x))$ is indistinguishable from $f(x;r)$, and (2) with high probability over the choice of r (or even with probability one) $R(f(x;r)) = g(x)$. One can view these conditions as saying that $g(x)$ is encoded by a *collection* of functions $\{f_r(x)\}_r$, where $f_r(x) = f(x;r)$.

Now suppose that our scheme is KDM secure with respect to the family $\{f_r(x)\}_r$, then we can apply the above approach and get a scheme which satisfies KDM security with respect to g. The only difference is that now the message preprocessing step is randomized: To encrypt a message M first encode it by the randomized mapping $S(M)$, and then use the original encryption function. The security reduction is essentially the same except that a KDM query for g in the new scheme is emulated by an old KDM query for a *randomly chosen* function f_r. This idea can be easily extended to the case where all functions in $\mathcal{G}$ are encoded by functions in $\mathcal{F}$:

Theorem 1.2 (Informal)**.** *If $\mathcal{F}$ is an RE of $\mathcal{G}$, then $\mathcal{G} \leq_{\mathsf{KDM}} \mathcal{F}$.*

The crux of this theorem, is that, unlike deterministic encoding, randomized encoding can represent complicated functions by collections of very simple functions [IK00, IK02, AIK06b, AIK06a]. Specifically, by combining the above theorem with the REs of [AIK06a], which, in turn, are based on Yao's garbled circuit [Yao86], we obtain our main results (Thm. 1.1).

1.3 Comparison with BGK and BHHI

Our techniques are inspired by both [BGK11] (BGK) and [BHHI10] (BHHI). We believe that our approach inherits the positive features of each of these works, and sheds new light on the way they

relate to each other. Let us review the main ideas behind these constructions and explain how they compare to our solution.

1.3.1 The BGK reduction

The starting point in [BGK11] is an encryption scheme which satisfies entropic KDM security with respect to $\mathcal{F}$. Roughly speaking, this means that KDM security should hold not only when sk is chosen uniformly from the key space $\mathcal{K} = \{0,1\}^k$ but also when it is chosen uniformly from a smaller domain $\mathcal{K}'$, e.g., $\mathcal{K}' = \{0,1\}^{k^\varepsilon}$. By relying on this notion, BGK shows that for every efficiently computable injective mapping $\alpha : \mathcal{K}' \to \mathcal{K}$, one can amplify security from $\mathcal{F}$ to the class $\mathcal{F} \circ \alpha$, i.e., with respect to functions $f(\alpha(\mathsf{sk}))$ for every $f \in \mathcal{F}$. The idea is to choose the key sk' from $\mathcal{K}'$ and employ the original scheme with the key $\mathsf{sk} = \alpha(\mathsf{sk}')$. This allows to translate a KDM query $f(\alpha(\mathsf{sk}'))$ for the new scheme into an entropic-KDM query $f(\mathsf{sk})$ for the old scheme.

The deterministic encoding (DE) approach is inspired by the BGK approach, and can be seen as a complementary solution. BGK extends a function $f : \mathcal{K} \to \mathcal{M}$ to $f \circ \alpha : \mathcal{K}' \to \mathcal{M}$ by shrinking the key space (from $\mathcal{K}$ to $\mathcal{K}'$), whereas in the DE approach $f : \mathcal{K} \to \mathcal{M}$ is extended to $R \circ f : \mathcal{K} \to \mathcal{M}'$ by padding messages which effectively shrinks the message space (from $\mathcal{M}$ to $\mathcal{M}' = R(\mathcal{M})$).

As a result BGK enjoys a similar attractive security reduction with single-query-to-single-query translation. This leads to flexibility with respect to the KDM *setting*. Indeed, although the BGK approach is not fully general due to its use of entropic-KDM security (a notion which seems stronger than standard KDM security), it immediately generalizes to the CCA and the symmetric-key settings, as long as the underlying scheme provides entropic-KDM security.

It should be mentioned that in our approach the amplification is achieved by modifying the encryption algorithm, rather than the key-generation algorithm as in BGK. This minor difference turns to have a considerable effect on the amplification-gap. First, it allows to use fresh randomness in every application of the encryption algorithm, and so the linkage between functions in $\mathcal{G}$ to functions in $\mathcal{F}$ can be *randomized*. Indeed, this is exactly what allows us to exploit the power of randomized encoding. In contrast, the BGK approach tweaks the key-generation algorithm and so the relation between $\mathcal{G}$ to $\mathcal{F}$ is bounded to be deterministic. In addition, since our modification happens in the encryption (and decryption) phases, we can let the function class $\mathcal{G}$ grow not only with the security parameter but also with the size of the messages. This leads to the strong notion of length-dependent security, and in addition allows to achieve $\mathsf{KDM}^{(t)}$ where the number of keys t grows both with the message length and the security parameter.

In contrast, the family $\mathcal{G}$ of BGK cannot grow with the message length, and it can only contain a polynomial number of functions. This limitation prevents it from being used in applications which require KDM security wrt larger functions classes (e.g., secure realization of symbolic protocols with axiomatic proofs of security). Furthermore, amplification for large number of keys can be achieved only at the expense of putting more restrictions on the underlying scheme (i.e., simulatable KDM security). On the other hand, assuming these additional properties, the BGK approach can get $\mathsf{KDM}^{(t)}$ for arbitrary unbounded t with respect to some concrete function families (e.g., constant degree polynomials), whereas in our approach t is always bounded by some fixed polynomial (in the security parameter and message length).[3] Finally, it is important to mention that the BGK

[3] In fact, we can achieve a slightly stronger notion. Assuming that the underlying scheme satisfies $\mathsf{KDM}^{(t)}$ security for arbitrary t's (as in [BHHO08, ACPS09]), we get a $\mathsf{KDM}^{(t)}$ secure scheme where there exists an unbounded number of keys in the system, but the arity of the KDM functions available to the adversary is polynomially bounded (in the

reduction treats $\mathcal{G}$ in a black-box way, while the randomized encoding approach treats this class in a non-black-box way.

1.3.2 The BHHI reduction

The BHHI approach relies on a novel connection between homomorphic encryptions and KDM security. First, it is observed that in order to obtain KDM security with respect to $\mathcal{G}$ it suffices to construct a scheme which provides both cyclic-security (i.e., KDM security with respect to the identity function) and homomorphism with respect to a function family $\mathcal{G}$, i.e., it should be possible to convert a ciphertext $C = \mathsf{E}_{\mathsf{pk}}(M)$ into $C' = \mathsf{E}_{\mathsf{pk}}(g(M))$ for every $g \in \mathcal{G}$. Indeed, the homomorphism property can be used to convert a ciphertext $\mathsf{E}_{\mathsf{pk}}(\mathsf{sk})$ into the ciphertext $\mathsf{E}_{\mathsf{pk}}(g(\mathsf{sk}))$, and so cyclic-security is amplified to $\mathcal{G}$-KDM security.

BHHI construct such an encryption scheme by combining a two-party secure computation protocol with two messages (i.e., based on Yao's garbled circuit [Yao86]) with a strong version of oblivious transfer which satisfies an additional *cyclic-security* property. The latter primitive is referred to as *targeted encryption* (TE). The basic idea is to view the homomorphic property as a secure-computation task in which the first party holds the message M and the second party holds the function g. The cyclic nature of the TE primitive allows to implement this homomorphism even when the input M is the secret-key. Finally, BHHI show that TE can be constructed based on affine-KDM secure encryption scheme which satisfies a strong notion of simulation: There exists a simulator which given the public-key pk can simulate a ciphertext $\mathsf{E}_{\mathsf{pk}}(g(\mathsf{sk}))$ in a way which is indistinguishable even for someone who holds the secret-key.

The BHHI construction seems conceptually different from our RE approach (i.e., homomorphism vs. encoding). Moreover, the construction itself is not only syntactically different, but also relies on different building blocks (e.g., TE). Still, the RE construction shares an important idea with BHHI: The use of secure-computation techniques. It is well known that REs are closely related to secure multiparty-computation (MPC) protocols[4], and, indeed, the role of REs in our reduction resembles the role of MPC in BHHI. In both solutions at some point the security reduction applies the RE/MPC to the function g in $\mathcal{G}$. Furthermore, both works achieve strong KDM security by instantiating the RE/MPC with Yao's garbled circuit (GC) — a tool which leads to both stand-alone RE construction [AIK06a] and, when equipped with an OT, to a two-party secure-computation protocol.

It should be emphasized, however, that the actual constructions differ in some important aspects. While we essentially encrypt the whole GC-based encoding under the underlying KDM encryption scheme, BHHI tweak the GC protocol with a cyclic-secure OT (i.e., TE). Pictorially, our underlying KDM-secure scheme "wraps" the GC encoding, whereas in BHHI the KDM-secure primitive is "planted inside" the GC protocol. This difference affects both generality and simplicity as follows.

First, BHHI are forced to implement a KDM-secure OT, a primitive which seems much stronger than standard KDM secure encryption schemes. For example, KDM-secure symmetric-key encryption schemes can be constructed at the presence of a random oracle [BRS02] while OT protocols cannot [IR88].[5] Moreover, as we already mentioned, although TE can be based on several known

security parameter and message length). Still, these functions can be applied to arbitrary subsets of the keys.

[4] In fact, REs were originally defined as a strong form of non-interactive reductions for MPC [IK00].

[5] It seems that a similar statement holds even for public-key KDM-secure schemes. See [BRS02, GKM+00].

affine-secure KDM schemes (i.e., ones which enable strong simulation), the LPN assumption (with constant error-rate) is a concrete example under which symmetric-key encryption scheme with KDM-security wrt affine functions exist, yet OT is not known to exist. Furthermore, since BHHI send the garbled circuit in the clear, it is not hard to show that the resulting scheme is not CCA-secure even if the TE provides CCA security. Finally, the modification of the GC protocol leads to a relatively complicated security proof.

2 Preliminaries

For a positive integer $n \in \mathbb{N}$, let $[n]$ denote the set $\{1, \ldots, n\}$, and U_n denote the uniform distribution over $\{0,1\}^n$. A function $\varepsilon(n)$ is *negligible* if it tends to zero faster than $1/n^c$ for every constant $c > 0$. The term *efficient* refers to probabilistic machines that run in polynomial time in the security parameter.

Efficient functions and randomized functions. A *randomized function* $f : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}^*$ is a function whose second input is treated as a random input. We write $f(x;r)$ to denote the evaluation of f on deterministic input x and random input r, and typically assume length regularity and efficient evaluation as follows: there are efficiently computable polynomials $m(n)$ and $\ell(n)$ and an efficiently computable circuit family $\left\{f_n : \{0,1\}^n \times \{0,1\}^{m(n)} \to \{0,1\}^{\ell(n)}\right\}$ which computes the restriction of f to n-bit deterministic inputs. If the function is *not* length regular, we assume that the circuit family is indexed by a pair of input and output parameters (n, ℓ), and require evaluation in time $\mathrm{poly}(n, \ell)$. Finally, a *deterministic* function corresponds to the special case where $m(n) = 0$.

Function ensembles. A *function ensemble* is a collection of functions $\{f_z\}_{z \in Z}$ indexed by an index set $Z \subseteq \{0,1\}^*$, where for each z the function f_z has a finite domain $\{0,1\}^{n(z)}$ and a finite range $\{0,1\}^{\ell(z)}$, where $n, \ell : \{0,1\}^* \to \mathbb{N}$. (This means that different functions may have different domains but each fixed function f_z is regular.) By default, we assume that ensembles are efficiently computable, that is, the functions $n(z), \ell(z)$, as well as the function $F(z,x) = f_z(x)$ are computable in time $\mathrm{poly}(|z|)$. Hence $n(z), \ell(z) < \mathrm{poly}(|z|)$. We also assume that $|z| < \mathrm{poly}(n(z), \ell(z))$.

Randomized encoding of functions. Intuitively, a randomized encoding of a function $g(x)$ is a randomized mapping $f(x;r)$ whose output distribution depends only on the output of g. We formalize this intuition via the notion of *computationally private randomized encoding* of [AIK06a], while adopting the original definition from a non-uniform adversarial setting to the uniform setting (i.e., adversaries are modeled by probabilistic polynomial-time Turing machines). Consider a function $g = \left\{g_n : \{0,1\}^n \to \{0,1\}^{\ell(n)}\right\}$ and a randomized function $f = \left\{f_n : \{0,1\}^n \times \{0,1\}^{m(n)} \to \{0,1\}^{s(n)}\right\}$, which are both efficiently computable. We say that f *encodes* g, if there exist an efficient recovery algorithms Rec and an efficient simulator Sim that satisfy the following:

- **perfect correctness.** For every $x \in \{0,1\}^n$, the error probabilities $\Pr[\mathsf{Rec}(1^n, f(x, \mathsf{U}_{m(n)})) \neq g(x)]$ and $\Pr[\mathsf{Rec}(1^n, \mathsf{Sim}(1^n, g(x))) \neq g(x)]$ are both zero.[6]

[6] Previous definitions require only that the first quantity is zero, however, all known constructions (of perfectly-correct randomized encoding) satisfy the current (stronger) definition.

- **computational privacy.** For every efficient adversary $\mathcal{A}$ we have that

$$\Pr[\mathcal{A}^{f(\cdot;\mathsf{U})}(1^n)=1]-\Pr[\mathcal{A}^{\mathsf{Sim}(g(\cdot))}(1^n)=1]<\mathrm{neg}(n),$$

  where the oracles are defined as follows: Given x the first oracle returns a sample from $f(x;\mathsf{U}_{m(|x|)})$ and the second oracle returns a sample from $\mathsf{Sim}(1^{|x|},g(x))$.

This notion is naturally extended to functions $g_{n,\ell}$ which are not length-regular and are indexed by both input and output lengths. However, we always assume that privacy is parameterized *only* with the input length (i.e., the adversary's running-time/distinguishing-probability should be polynomial/negligible in the input length.) Note that, without loss of generality, we can assume that the relevant output length ℓ is always known to the decoder and simulator (i.e., it can be always encoded as part of the output of $f_{n,\ell}$).

Encryption schemes (syntax). An encryption scheme consists of three efficient algorithms $(\mathsf{KG},\mathsf{E},\mathsf{D})$, where KG is a key generation algorithm which given a security parameter 1^k outputs a pair $(\mathsf{sk},\mathsf{pk})$ of decryption and encryption keys; E is an encryption algorithm that takes a message $M\in\{0,1\}^*$ and an encryption key pk and outputs a ciphertext C; and D is a decryption algorithm that takes a ciphertext C and a decryption key sk and outputs a plaintext M'. We also assume that both algorithms take the security parameter 1^k as an additional input, but typically omit this dependency for simplicity. Correctness requires that the decryption error

$$\max_{M\in\{0,1\}^*}\Pr_{(\mathsf{sk},\mathsf{pk})\xleftarrow{R}\mathsf{KG}(1^k)}[\mathsf{D}_{\mathsf{sk}}(\mathsf{E}_{\mathsf{pk}}(M))\neq M],$$

should be negligible in k, where the probability is taken over the randomness of KG,E and D. For security parameter k, let $\mathcal{K}_k$ denote the space from which decryption keys are chosen. Without loss of generality, we always assume that $\mathcal{K}_k=\{0,1\}^k$.

Following Goldreich [Gol04], we note that the above definition corresponds to both public-key and symmetric-key encryption schemes where the latter correspond to the special case in which the decryption key sk and encryption key pk are equal. As we will see, the difference between the two settings will be part of the security definitions.

3 KDM-Security

Let $\mathcal{E}=(\mathsf{KG},\mathsf{E},\mathsf{D})$ be an encryption scheme with key space $\mathcal{K}=\{\mathcal{K}_k\}$. Let $t:\mathbb{N}\to\mathbb{N}$ be a function. A t-ary KDM function ensemble is an efficient ensemble of functions $\mathcal{F}=\left\{f_{k,z}:\mathcal{K}_k^{t(k)}\to\{0,1\}^*\right\}_{(k,z)}$. We let $\mathcal{F}_k$ denote the set $\left\{f_{k,z}:\mathcal{K}_k^{t(k)}\to\{0,1\}^*\right\}_z$. An $\mathcal{F}$-KDM Chosen-Ciphertext Attack (CCA) in the public-key setting is defined in Fig. 1 as a game that takes place between a challenger and an adversary $\mathcal{A}$. The advantage of $\mathcal{A}$ when attacking a scheme $\mathcal{E}$ is $\alpha(k)=\Pr[\mathcal{A}\text{ wins the KDM game}]-\frac{1}{2}$.

By restricting the power of the adversary in the KDM game (Fig. 1) we get other KDM settings. Specifically, the symmetric-key setting corresponds to adversaries of type sym who do not ask public-key queries, and the CPA setting corresponds to adversaries of type CPA who do not make decryption queries. Hence, we can classify KDM adversaries into one of the following four *types*: $(\mathsf{pub},\mathsf{CCA}),(\mathsf{pub},\mathsf{CPA}),(\mathsf{sym},\mathsf{CCA})$, and $(\mathsf{sym},\mathsf{CPA})$. An adversary of type T that conducts an $\mathcal{F}$-KDM attack is denoted as $(T,\mathcal{F})$-adversary.

- **Initialization.** The challenger randomly chooses a bit $b \xleftarrow{R} \{0,1\}$ and $t = t(k)$ key-pairs $(\mathsf{sk}_1, \mathsf{pk}_1) \ldots, (\mathsf{sk}_t, \mathsf{pk}_t)$ by invoking $\mathsf{KG}(1^k)$ for t times. The adversary $\mathcal{A}$ can send a "public-key" query and get to see all the encryption keys $(\mathsf{pk}_1, \ldots, \mathsf{pk}_t)$.
- **Queries.** The adversary $\mathcal{A}$ may adaptively make polynomially-many queries of the following types:
  - **Encryption queries** of the form (i, M) where $i \in [t]$ and $M \in \{0,1\}^*$. The challenger responds with $C \xleftarrow{R} \mathsf{E}(\mathsf{pk}_i, M)$ if $b = 1$, and $C \xleftarrow{R} \mathsf{E}(\mathsf{pk}_i, 0^{|M|})$ if $b = 0$.
  - **KDM queries** of the form (i, f) where $i \in [t]$ and $f \in \mathcal{F}_k$. The challenger computes $M = f(\mathsf{sk}_1, \ldots, \mathsf{sk}_t)$ and responds with $C \xleftarrow{R} \mathsf{E}(\mathsf{pk}_i, M)$ if $b = 1$, and $C \xleftarrow{R} \mathsf{E}(\mathsf{pk}_i, 0^{|M|})$ if $b = 0$.
  - **Decryption queries** of the form (i, C) where $i \in [t]$ and the string C was not given as an answer of a previous encryption/KDM query. The challenger responds with $M = \mathsf{D}_{\mathsf{sk}_i}(C)$ regardless of the value of b.
- **Final phase.** The adversary outputs a bit $b' \in \{0,1\}$ and wins if $b = b'$.

Figure 1: The $\mathcal{F}$-KDM game is defined with respect to the function ensemble $\mathcal{F} = \{\mathcal{F}_k\}$ and is indexed by the security parameter k. The presence (resp., absence) of public-key query captures the public-key (resp., symmetric-key) setting.

Definition 3.1. (KDM-secure encryption) *Let T be a type, and $\mathcal{F}$ be a function ensemble. An encryption scheme is* $(T, \mathcal{F})$-KDM secure *if every efficient $(T, \mathcal{F})$ adversary has at most negligible advantage when attacking the scheme.*

Interesting KDM functions ensembles. For every $t = t(k)$ and for every type T we consider the following ensembles:

- **Constants, selectors, and projections**. If $\mathcal{F}_k$ contains all constant functions $\{f_M : (\mathsf{sk}_1, \ldots, \mathsf{sk}_t) \mapsto M\}_M$, then, as observed in [BHHO08], KDM queries are equivalent to standard encryption queries and KDM security is nothing but standard security (with respect to the type T). If the ensemble $\mathcal{F}_k$ contains all selector functions $\{f_j : (\mathsf{sk}_1, \ldots, \mathsf{sk}_t) \mapsto \mathsf{sk}_j\}_{j \in [t]}$, we get the notion of *clique* security [BHHO08] (which is stronger than *circular security* [CL01]), that is, the scheme is secure even if the adversary sees encryptions of the form $\mathsf{E}_{\mathsf{pk}_i}(\mathsf{sk}_j)$ for every $i, j \in [t]$. Another elementary class that slightly generalizes the previous ones is the class of all functions $f : (\vec{\mathsf{sk}}) \mapsto v$ in which each output bit depends on (at most) a single bit of the input $\vec{\mathsf{sk}} = (\mathsf{sk}_1, \ldots, \mathsf{sk}_t)$. Namely, the j-th output bit v_j is either fixed to a constant or copies/flips an original bit of one of the keys, i.e., $v_j \in \{0, 1, \mathsf{sk}_{i,q}, 1 - \mathsf{sk}_{i,q}\}$, where $\mathsf{sk}_{i,q}$ is the q-th bit of the i-th secret key. We refer to this class as the class of *projections* and let $\Pi^t_{k,\ell}$ denote the restriction of this class to functions of input length kt and output length $\ell(k)$. Projections is a proper subclass of the class of affine functions $L : \mathbb{F}_2^{kt} \to \mathbb{F}_2^{\ell(k)}$.
- **Polynomial-size circuits [BHHI10].** For polynomials $p(\cdot)$ and $\ell(\cdot)$, let $\mathcal{C}^t_{k,\ell,p}$ denote the

class of all circuits $C : \{0,1\}^{kt} \to \{0,1\}^{\ell(k)}$ of size at most $p(k) + p(\ell)$. Security with respect to this class is denoted by (p, ℓ)-*bounded circuit-size* KDM security. A slightly stronger notion of security is p-*length-dependent* KDM security which means that the scheme is KDM secure with respect to $\mathcal{C}^t_{k,\ell,p}$ for *every* polynomial ℓ. While, ultimately one would like to have KDM security with respect to all polynomial-size circuits (for arbitrary polynomial), it seems that p-length-dependent security, say for quadratic p, may be considered to be almost as powerful since it allows the adversary to use larger circuits by encrypting longer messages. In particular, one can represent essentially any polynomial-time computable function via padding. That is, if a function f is not in the class since its circuit is too large, then a "padded" version f' of f in which the output is padded with zeroes does fall into the ensemble. Furthermore, in [BHHI10] it was shown that if p is sufficiently large (e.g., the quadratic polynomial) then length-dependent security is sufficient for axiomatic-security applications (i.e., it gives the ability to securely instantiate symbolic protocols with axiomatic proofs of security).

The above definitions become stronger when the arity t grows. At one extreme, one may consider a single scheme which satisfies any of the above definitions for an arbitrary polynomial $t(k)$, and at the other extreme one may consider the case of $t = 1$, which is still non-trivial even for projection functions.

Reductions among KDM-ensembles. We say that a KDM function ensemble $\mathcal{G}$ KDM-reduces to another KDM function ensemble $\mathcal{F}$ (in symbols $\mathcal{G} \leq_{\mathsf{KDM}} \mathcal{F}$) if there exists a transformation which converts an encryption scheme $\mathcal{E}$ that is $\mathcal{F}$-KDM secure to an encryption scheme $\widehat{\mathcal{E}}$ which is $\mathcal{G}$-KDM secure. Formally, such a (black-box) reduction is composed of (1) (construction) an encryption scheme $\widehat{\mathcal{E}}$ which is given an oracle access to the scheme $\mathcal{E}$; and (2) (security reduction) an efficient algorithm $\mathcal{B}$ such that for any $\mathcal{F}$-adversary $\mathcal{A}$ which attacks $\mathcal{E}$ with advantage α, the $\mathcal{G}$-adversary $\mathcal{B}^{\mathcal{A},\mathcal{E}}$ attacks the scheme $\widehat{\mathcal{E}}$ with a similar advantage (up to a negligible loss). This definition can be instantiated with respect to all four different types. We say that the reduction is *type-preserving* if $\mathcal{B}^{\mathcal{A},\mathcal{E}}$ is always of the same type as $\mathcal{A}$ (i.e., $\mathcal{B}$ always ask the same type of queries that $\mathcal{A}$ asks in the KDM game.) Type preserving reduction extends KDM-security while being insensitive to the concrete setting which is being used. Formally,

Lemma 3.2 (KDM-reductions)**.** *Suppose that the KDM function ensemble $\mathcal{G}$ KDM-reduces to the ensemble $\mathcal{F}$ via a type-preserving reduction $(\widehat{\mathcal{E}}, \mathcal{B})$. For every $T \in \{\mathsf{pub}, \mathsf{sym}\} \times \{\mathsf{CCA}, \mathsf{CPA}\}$, if the encryption scheme $\mathcal{E}$ is $(T, \mathcal{F})$-KDM secure then the scheme $\widehat{\mathcal{E}}^{\mathcal{E}}$ is $(T, \mathcal{G})$-KDM secure.*

4 Reductions and Completeness results

4.1 KDM reductions via randomized encoding

Let $\mathcal{F} = \{f_{k,z}\}$ and $\mathcal{G} = \{g_{k,w}\}$ be a pair of KDM function ensembles with the same arity $t = t(k)$. We say that $\mathcal{F}$ *encodes* $\mathcal{G}$ if every function $g(x)$ in $\mathcal{G}$ has a randomized encoding $f(x; r)$ such that for every fixing of the random string r, the resulting function $f_r(x)$ is in $\mathcal{F}$. More formally, the evaluation function $G_k(z, x)$ of $\mathcal{G}$ should have a randomized encoding $F_k((z, x); r)$ such that for every fixing of r and index z, the function $F_{k,z,r}(x) = F(k, z, x; r)$ corresponds to a function $f_{k,w}$ in $\mathcal{F}$, where the mapping from (z, r) to w should be efficiently computable in poly(k) time. Note

that this means that the simulator and decoder are *universal* for all indices z, and depend only on the value of k.

Theorem 4.1 (main theorem)**.** *Suppose that the KDM function ensemble $\mathcal{F}$ encodes the KDM function ensemble $\mathcal{G}$. Then, $\mathcal{G}$ KDM-reduces to $\mathcal{F}$ via a type-preserving reduction.*

To prove the theorem we need to describe a construction and a security reduction. From now on, let Sim and Rec be the universal simulator and recovery algorithm which establish the encoding of $\mathcal{G}$ by $\mathcal{F}$.

Construction 4.2. *Given oracle access to the encryption scheme $\mathcal{E} = (\mathsf{KG}, \mathsf{E}, \mathsf{D})$, we define the scheme $\widehat{\mathcal{E}}$ as follows*

$$\widehat{\mathsf{KG}}(1^k) = \mathsf{KG}(1^k) \qquad \widehat{\mathsf{E}}_{\mathsf{pk}}(M) = \mathsf{E}_{\mathsf{pk}}(\mathsf{Sim}(M)) \qquad \widehat{\mathsf{D}}_{\mathsf{sk}}(C) = \mathsf{Rec}(\mathsf{D}_{\mathsf{sk}}(C)),$$

where all algorithms (i.e., encryption, decryption, simulator and recovery) get the security parameter 1^k as an additional input.

It is not hard to show that $\widehat{\mathcal{E}}$ satisfies the syntactic requirements of encryption schemes, namely correctness.

Lemma 4.3 (correctness)**.** *The decryption error of the scheme $\widehat{\mathcal{E}}$ is the same as the decryption error of $\mathcal{E}$, and so it is negligible.*

Proof. The probability that a message M is incorrectly decrypted is bounded by

$$\Pr_{(\mathsf{sk},\mathsf{pk})\xleftarrow{R}\mathsf{KG}(1^k), M'\xleftarrow{R}\mathsf{Sim}(M)}[\mathsf{D}_{\mathsf{sk}}(\mathsf{E}_{\mathsf{pk}}(M')) \neq M'] + \Pr[\mathsf{Rec}(M') \neq M],$$

since the second term is 0, due to the (perfect) correctness of the encoding, we can bound the above by $\max_{M'\in\{0,1\}^*} \Pr[\mathsf{D}_{\mathsf{sk}}(\mathsf{E}_{\mathsf{pk}}(M')) \neq M']$, where $(\mathsf{sk}, \mathsf{pk}) \xleftarrow{R} \mathsf{KG}(1^k)$. □

We show that the security of $\widehat{\mathcal{E}}$ can be based on that of $\mathcal{E}$. Given an oracle access to a $(T, \mathcal{G})$ adversary $\mathcal{A}$ that attacks $\widehat{\mathcal{E}}$, we define a $(T, \mathcal{F})$ adversary $\mathcal{B}$ that attacks $\mathcal{E}$ by randomly choosing one of two strategies $\mathcal{B}_0$ and $\mathcal{B}_1$.

Reduction 4.4 (The adversary $\mathcal{B}^{\mathcal{A},\mathcal{E}}$)**.** *Toss a coin $\sigma \xleftarrow{R} \{0,1\}$. If $\sigma = 1$ invoke the following adversary $\mathcal{B}_1$:*

- ***Initialization:*** *$\mathcal{B}$ invokes $\mathcal{A}$. If $\mathcal{A}$ asks for the encryption keys then $\mathcal{B}$ makes a similar query and passes the answer to $\mathcal{A}$.*
- ***Encryption query:*** *If $\mathcal{A}$ makes an encryption query (i, M), for $i \in [t]$ and $M \in \{0,1\}^*$, then $\mathcal{B}$ samples $M' = \mathsf{Sim}(M)$, sends (i, M') as an encryption query (wrt to $\mathcal{E}$) and passes the answer of the challenger to $\mathcal{A}$.*
- ***KDM query:*** *If $\mathcal{A}$ makes a KDM query (i, g), for $i \in [t]$ and $g \in \mathcal{G}$, then the adversary $\mathcal{B}$ does the following: She uniformly chooses randomness r for the randomized encoding $f(\cdot; r)$ of $g(\cdot)$, and asks the KDM query (i, f_r) where $f_r(\cdot) = f(\cdot; r)$ which, by our assumption, is in $\mathcal{F}$. The answer of the challenger is being sent to $\mathcal{A}$.*

- ***Decryption query:*** *If $\mathcal{A}$ makes a decryption query (i, C), then $\mathcal{B}$ checks that it is legal (by inspecting all previous encryption/KDM queries), and if so, (1) passes the same decryption query to the challenger, (2) applies the recovery algorithm* Rec *to the result, and (3) sends it back to $\mathcal{A}$.*

- ***Termination:*** *$\mathcal{B}$ terminates with the same output of $\mathcal{A}$.*

If $\sigma = 0$ then invoke the adversary $\mathcal{B}_0$. This adversary is similar to $\mathcal{B}_1$ except that encryption and KDM queries of $\mathcal{A}$ are both translated into encryption queries as follows: given an encryption query of $\mathcal{A}$ of the form (i, M) (resp., KDM query of the form (i, g)), the adversary $\mathcal{B}_0$ samples $M' = \mathsf{Sim}(0^\ell)$ and asks for the ciphertext $\mathsf{E}_{\mathsf{pk}_i}(M')$, where ℓ is the length of M (resp., output length of g).[7] *At the end, $\mathcal{B}_0$ flips the output of $\mathcal{A}$ and terminates.*

Note that the above reduction is indeed type-preserving. Let us first focus on the adversary $\mathcal{B}_1$. If the challenge bit b is 1 (i.e., when the challenger is in the "real-mode"), then the difference between the emulated view of $\mathcal{A}$ and the view of $\mathcal{A}$ in the actual KDM game, is only due to the difference in the way KDM queries are answered. In the real game answers to KDM queries are computed properly as $\widehat{\mathsf{E}}_{\mathsf{pk}_i}(g(\vec{\mathsf{sk}})) = \mathsf{E}_{\mathsf{pk}_i}(\mathsf{Sim}(g(\vec{\mathsf{sk}})))$, whereas in the emulated game they are computed by $\mathsf{E}_{\mathsf{pk}_i}(f(\vec{\mathsf{sk}}; \mathsf{U}))$. However, this difference should not be noticeable due to the privacy of the randomized encoding. Formally, let $\alpha_b(k)$ (resp., $\beta_{\sigma,b}(k)$) denote the probability that $\mathcal{A}$ (resp., $\mathcal{B}_\sigma$) guesses the challenge bit when it takes the value b. Then,

Lemma 4.5. $|\beta_{1,1}(k) - \alpha_1(k)| \leq \mathrm{neg}(k)$.

Proof. We define the following distinguisher $\mathcal{D}$ which, given an oracle access to either $F(\cdot; \mathsf{U})$ or to $\mathsf{Sim}(G(\cdot))$, attempts to distinguish between the two. The adversary $\mathcal{D}$ emulates the challenger with challenge bit $b = 1$. It generates a key vector $(\mathsf{sk}_i, \mathsf{pk}_i)_{i\in[t]}$ by executing the key-generation algorithm $\mathsf{KG}(1^k)$ for t times. Then $\mathcal{D}$ invokes $\mathcal{A}$. If $\mathcal{A}$ asks a KDM query (i, g_z) then $\mathcal{D}$ calls its oracle with the value $G(z, \mathsf{sk}_1, \ldots, \mathsf{sk}_t)$. Let M denote the answer of the oracle. The distinguisher computes the ciphertext $C = \mathsf{E}_{\mathsf{pk}_i}(M)$ and sends the ciphertext C to $\mathcal{A}$. If $\mathcal{A}$ asks other types of queries such as public-key queries, encryption queries, and decryption queries, the distinguisher $\mathcal{D}$ answers them properly exactly as the real challenger does when it's in the real mode $b = 1$. (For the case of a decryption query (i, C), the distinguisher checks that it is legal by inspecting all previous KDM/encryption queries, and if so, sends $\mathsf{D}_{\mathsf{sk}_i}(C)$.) The distinguisher halts with output 1 if and only if $\mathcal{A}$ outputs 1.

Note that: (1) If $\mathcal{D}$ gets an oracle access to $\mathsf{Sim}(G(\cdot))$ then the view of $\mathcal{A}$ is distributed exactly as in the real game and so in this case $\mathcal{D}$ outputs 1 with probability $\alpha_1(k)$; (2) If $\mathcal{D}$ gets an oracle access to $F(\cdot; \mathsf{U})$ then the view of $\mathcal{A}$ is distributed exactly as in the above reduction when $\mathcal{B}_1$ emulates the game with $b = 1$, and so in this case $\mathcal{D}$ outputs 1 with probability $\beta_{1,1}(k)$. Hence, by the privacy of the encoding, it follows that $|\beta_{1,1}(k) - \alpha_1(k)| \leq \mathrm{neg}(k)$. □

We would like to argue now that a similar thing happens in the "fake" mode when $b = 0$; namely, that $\beta_{1,0}$ is close to α_0. However, in this case real-game KDM queries are answered with $\widehat{\mathsf{E}}_{\mathsf{pk}_i}(0^\ell) = \mathsf{E}_{\mathsf{pk}_i}(\mathsf{Sim}(0^\ell))$, whereas in the game emulated by $\mathcal{B}_1$ these queries are answered by $\mathsf{E}_{\mathsf{pk}_i}(0^s)$, where $\ell = |g(\mathsf{sk}_1, \ldots, \mathsf{sk}_t)|$ and $s = |f(\mathsf{sk}_1, \ldots, \mathsf{sk}_t; \mathsf{U})|$. Although the privacy of the encoding ensures that the plaintexts are of the same length, i.e., $s = |\mathsf{Sim}(0^\ell)|$, the actual distributions of the plaintexts

[7] Recall that the output length of $g \in \mathcal{G}$ is given as part of its description.

may differ, and so it may be the case that the two views are distinguishable. For this reason we need the adversary $\mathcal{B}_0$ which breaks the standard (non-KDM) security of $\mathcal{E}$ whenever such a gap exists. Formally, we will show that the average success probability of $\mathcal{B}_1$ and $\mathcal{B}_0$ is roughly half the success probability of $\mathcal{A}$. To this aim we prove the following

Lemma 4.6. *$\beta_{0,1}(k) = \alpha_0(k)$ and $\beta_{0,0}(k) + \beta_{1,0}(k) = 1$.*

Proof. First, we note that when the challenge bit $b = 1$, the view of $\mathcal{A}$ as emulated by $\mathcal{B}_0$ is identical to the view of $\mathcal{A}$ in the fake mode of the real game ($b = 0$). Indeed, in both cases a KDM query (i, g) (resp., an encryption query (i, M)) is answered with $\widehat{\mathsf{E}}_{\mathsf{pk}_i}(0^{|\ell|}) = \mathsf{E}_{\mathsf{pk}_i}(\mathsf{Sim}(0^\ell))$ where ℓ is the output length of g (resp., $\ell = |M|$). Hence, $\beta_{0,1}$, the probability that $\mathcal{B}_0$ outputs 1 when the challenger is in the real mode, is exactly the probability that $\mathcal{A}$ outputs 0 in the real game when the challenger is in the fake mode. (Recall that $\mathcal{B}$ flips the output of $\mathcal{A}$). The first equation follows.

To prove the second equality we first claim that when the challenge bit b is 0, the view of $\mathcal{A}$ when emulated by $\mathcal{B}_0$ is identical to the view of $\mathcal{A}$ as emulated by $\mathcal{B}_1$. Indeed, the only difference is that in the first case KDM queries (i, g) are answered by $\mathsf{E}(0^{|\mathsf{Sim}(g(\mathsf{sk}))|})$, while in the second case the answer is $\mathsf{E}(0^{|f(\mathsf{sk};r)|})$. The output lengths of f and $\mathsf{Sim}(g(\cdot))$ are fixed (for any $g \in \mathcal{G}$) and therefore should be equal (otherwise the privacy of the encoding is violated), and so the claim follows. The claim implies that $\beta_{0,0}(k) + \beta_{1,0}(k) = 1$, as $\mathcal{B}_1$ outputs the outcome of $\mathcal{A}$, and $\mathcal{B}_0$ flips it. □

By combining the two lemmas (4.5 and 4.6), it follows that the advantage $\beta = (\beta_{1,1} + \beta_{1,0} + \beta_{0,0} + \beta_{0,1})/4 - \frac{1}{2}$ of $\mathcal{B}$ is at least $\frac{1}{2}\alpha - \mathrm{neg}(k)$ where $\alpha = \frac{1}{2}(\alpha_1 + \alpha_0) - \frac{1}{2}$ is the advantage of $\mathcal{A}$. Hence, we established the correctness of the reduction.

Theorem 4.7. *If $\mathcal{A}$ is an efficient adversary that breaks $\widehat{\mathcal{E}}$ wrt $\mathcal{G}$ with advantage $\alpha(k)$, then the adversary $\mathcal{B}^{\mathcal{A},\mathcal{E}}$ breaks $\mathcal{E}$ wrt $\mathcal{F}$ with advantage $\beta(k) \geq \alpha(k)/2 - \mathrm{neg}(k)$.*

Remark 4.8. *Thm. 4.1 holds even if the encoding itself makes use of the underlying encryption scheme $\mathcal{E}$ as long as this usage is done in a fully black-box way (the same holds for any cryptographic primitive which can be based on $\mathcal{E}$ via a black-box reduction e.g., one-way function). More precisely, our results hold (i.e., lead to black-box KDM reduction/construction) as long as the security of the encoding reduces to the security of the underlying primitive (i.e., $\mathcal{E}$) via a black-box reduction, and as long as the simulator and decoder can be implemented given a black-box access to the underlying primitive. Similarly, such a black box access can be given to the algorithm which maps fixed index/randomness pairs (z, r) to the index w of the function $g_{k,w} = G_{k,z,r}(x)$.*

4.2 Completeness of projections

In [AIK06a] it is shown that Yao's garbled circuit technique allows to encode any efficiently computable function by a decomposable encoding in which every bit depends on at most a single bit of the deterministic input. By combining this fact with Thm. 4.1 we get the following:

Proposition 4.9 (Completeness of projections)**.** *For every polynomials $p(\cdot)$, $t(\cdot)$ and $\ell(\cdot)$, there exists a polynomial $q(\cdot)$ for which*

$$\mathcal{C}^t_{k,\ell,p} \leq_{\mathsf{KDM}} \Pi^t_{k,q}, \qquad\qquad \mathcal{C}^t_{k,p} \leq_{\mathsf{KDM}} \Pi^t_k, \tag{1}$$

where $\mathcal{C}^t_{k,\ell,p}$ is the t-ary ensemble of p-bounded circuits of output length ℓ, $\Pi^t_{k,q}$ is the t-ary ensemble of projections of output length q, $\mathcal{C}^t_{k,p} = \bigcup_{a\in\mathbb{N}} \mathcal{C}^t_{k,k^a,p}$, and $\Pi^t_k = \bigcup_{a\in\mathbb{N}} \Pi^t_{k,k^a}$. Moreover, the reductions are type preserving.

Hence, one can upgrade KDM security from (almost) the weakest KDM function ensemble to the very powerful notion of p-length-dependent KDM security.

Proof. By [AIK06a] any efficiently computable circuit family $\{g_k(x)\}$ of circuit complexity $a(k)$ can be encoded by a uniform computationally-private perfectly-correct encoding $\{\hat{g}_k(x;r)\}$ with the following properties: (1) The simulator and decoder use a black-box access to a symmetric encryption (equivalently, to a one-way function); (2) For every fixed randomness r, the resulting function $\hat{g}_{k,r}(x) = \hat{g}_k(x;r)$ is a projection function of output length $a(k)^{1+\varepsilon}$, where $\varepsilon > 0$ is an arbitrary small constant. (3) The mapping from the circuit of g_k to the circuit of $\hat{g}_{k,r}$ is efficiently computable given a black-box access to the symmetric encryption scheme.

Let $\{F_k\}$ be the universal (and uniform) circuit family for the mapping $(x, z) \mapsto y$ where $x \in (\{0,1\}^k)^t$, the string z is a description of a circuit $C_z : (\{0,1\}^k)^t \to \{0,1\}^{\ell(k)}$ of size $p(k) + p(\ell(k))$, and the string $y \in \{0,1\}^{\ell(k)}$ is $C_z(x)$. By applying the encoding from [AIK06a] to $\{F_k\}$ it follows that $\mathcal{C}^t_{k,\ell,p}$ is encoded by $\Pi^t_{k,q}$ where q is polynomial in the circuit size of F_k. The first part of the proposition now follows from Thm. 4.1.

The second part follows similarly, except that now we consider the (non-regular) function $\{G_{k,\ell}\}$ which computes the same mapping of F_k but for circuits C_z whose output length ℓ is given as an additional index, and not as a fixed polynomial in k. Again, by applying the encoding from [AIK06a] to $\{G_k\}$ it follows that $\mathcal{C}^t_{k,p}$ is encoded by Π^t_k, and the claim follows from Thm. 4.1. □

In the case of CPA KDM security, one can actually derive KDM-security with respect to projections of arbitrary output length (i.e., Π^t_k) from single-output projections $\Pi^t_{k,1}$.

Lemma 4.10 (Completeness of single-output projections for CPA-KDM)**.** *For every polynomial $t(\cdot)$, we have $\Pi^t_k \leq_{\mathsf{KDM}} \Pi^t_{k,1}$, where the reduction holds for both* $(\mathsf{sym}, \mathsf{CPA})$ *and* $(\mathsf{pub}, \mathsf{CPA})$ *types.*

Proof. The proof follows by simple concatenation: the new encryption/decryption algorithms encrypts/decrypts the message/ciphertext by applying the original encryption/decryption algorithm in a bit by bit manner. Hence, a KDM query in Π^t_{k,k^a} for the new scheme can be emulated by k^a KDM queries in $\Pi^t_{k,1}$ for the original scheme. □

As shown in [BPS07], we can use the standard encrypt-then-MAC transformation to upgrade the security of a scheme that satisfies $(\mathsf{sym}, \mathsf{CPA})$-KDM security into a scheme that satisfies $(\mathsf{sym}, \mathsf{CCA})$-security with respect to the same KDM class. A similar result was proven for the public-key setting by [CCS09] via the Naor-Yung double-encryption paradigm (which relies on the existence of NIZK). Hence, by Proposition 4.9 and Lemma 4.10, we have:

Corollary 4.11 (KDM Collapse)**.** *For every polynomials t and p, there exists a $\Pi^t_{k,1}$-KDM secure scheme if and only if there exists a t-ary p-length-dependent KDM secure encryption scheme. This holds unconditionally for the KDM types* $(\mathsf{sym}, \mathsf{CPA})$, $(\mathsf{sym}, \mathsf{CCA})$, *and* $(\mathsf{pub}, \mathsf{CPA})\}$*, and it holds for* $(\mathsf{pub}, \mathsf{CCA})$ *assuming the existence of non-interactive zero-knowledge proof system for* **NP**.

We remark that all the known constructions of affine-KDM secure encryption schemes [BHHO08, ACPS09, BG10] can be adapted to yield KDM security with respect to single-output projections (see Appendix A). Hence, we get p-length-dependent $(\mathsf{pub}, \mathsf{CPA})$-KDM (resp., $(\mathsf{sym}, \mathsf{CCA})$) based on the DDH, LWE, or QR assumptions (resp., LPN assumption), which can be boosted into $(\mathsf{pub}, \mathsf{CCA})\}$-KDM assuming the existence of NIZK for **NP**.

5 On Full KDM Security

In this section, we study the possibility of constructing a scheme which satisfies KDM security for the class of all functions. In [BHHI10] it was shown that such a scheme can be constructed based on the existence of cyclic-secure fully homomorphic encryption (FHE) [Gen09]. We show that a similar assumption is inherently required for full KDM security which is also *simulatable*. For simplicity, we focus on the case of arity $t = 1$ and single-query adversaries.

A public-key encryption scheme $\mathcal{E} = (\mathsf{KG}, \mathsf{E}, \mathsf{D})$ is simulatable $\mathcal{F}$-KDM secure if there exists a polynomial-time simulator S such that for every $(\mathsf{sk}, \mathsf{pk}) \in \mathsf{KG}(1^k)$, and every circuit family $f_k \in \mathcal{F}_k$ of size $\mathrm{poly}(k)$, the ensemble $S(\mathsf{pk}, f_k)$ is indistinguishable from $\mathsf{E}_{\mathsf{pk}}(f_k(\mathsf{sk}))$. (Note that this means that the distinguisher holds the secret-key sk.) The notions of *simulatable circular-security* and *simulatable full-KDM security* correspond to the two extreme cases where $\mathcal{F}$ contains only the identity function, and $\mathcal{F}$ contains all functions.

An FHE allows to translate encryptions of a message M into an encryption of a related message $h(M)$ for any polynomial-size circuit h. More formally, we say that $\mathcal{E}$ is *fully homomorphic* if there exists an efficient algorithm Eval such that for every $(\mathsf{sk}, \mathsf{pk}) \in \mathsf{KG}(1^k)$, every circuit family $\{h_k\}$ of size $\mathrm{poly}(k)$, and every sequence of messages $M_k \in \{0,1\}^{\mathrm{poly}(k)}$, the ensemble $\mathsf{Eval}(\mathsf{pk}, h_k, \mathsf{E}_{\mathsf{pk}}(M_k))$ is computationally indistinguishable from the ensemble $\mathsf{E}_{\mathsf{pk}}(h_k(M_k))$.

In [BHHI10], it was shown that if an encryption scheme is both simulatable circular-secure and fully-homomorphic then it is also simulatable fully-KDM secure. We show that the other direction holds as well, and so the two notions are equivalent.

Proposition 5.1. *Any simulatable fully-KDM secure encryption scheme is also fully-homomorphic circular-secure.*

Proof. Given a simulatable fully-KDM secure encryption scheme $(\mathsf{KG}, \mathsf{E}, \mathsf{D})$ with simulator S, we define $\mathsf{Eval}(\mathsf{pk}, h, C)$ by invoking S on the pair $(\mathsf{pk}, f_{h,C})$ where $f_{h,C}$ is the mapping $\mathsf{sk} \mapsto h(\mathsf{D}_{\mathsf{sk}}(C))$. Note that the circuit size of $f_{h,C}$ is polynomial in the circuit size of h (since D is efficient). Also, by definition, we have for every $(\mathsf{sk}, \mathsf{pk}) \in \mathsf{KG}(1^k)$, sequence $\{M_k\}$ and sequence $\{h_k\}$,

$$\begin{aligned} \mathsf{Eval}(\mathsf{pk}, h_k, \mathsf{E}_{\mathsf{pk}}(M_k)) &\equiv S(\mathsf{pk}, f_{h_k, \mathsf{E}_{\mathsf{pk}}(M_k)}) \\ &\stackrel{c}{\equiv} \mathsf{E}_{\mathsf{pk}}(h_k(\mathsf{D}_{\mathsf{sk}}(\mathsf{E}_{\mathsf{pk}}(M_k)))) \\ &\equiv \mathsf{E}_{\mathsf{pk}}(h_k(M_k)), \end{aligned}$$

where $\equiv$ ($\stackrel{c}{\equiv}$) denotes statistical (computational) indistinguishability. □

Next, we show that if one removes the simultability requirement then any encryption scheme $(\mathsf{KG}, \mathsf{E}, \mathsf{D})$ which provides KDM security with respect to a function which is slightly stronger than its decryption algorithm D, is actually fully-KDM secure. This is done by observing that Gentry's "bootstrapping technique" can be adapted to the KDM setting.

Proposition 5.2. *Let $T \in \{(\mathsf{pub}, \mathsf{CPA}), (\mathsf{sym}, \mathsf{CPA})\}$, and let $\mathcal{E} = (\mathsf{KG}, \mathsf{E}, \mathsf{D})$ be T-KDM secure encryption with respect to single-output projections and with respect to the function family $\mathcal{F}_k = \{f_{C_1,C_2} : \mathsf{sk} \mapsto \mathrm{NAND}(\mathsf{D}_{\mathsf{sk}}(C_1), \mathsf{D}_{\mathsf{sk}}(C_2))\}$, where C_1, C_2 ranges over $\{0,1\}^{p(k)}$ and $p(k)$ is the length of an encryption of one-bit message under secret-key of length k. Then, $\mathcal{E}$ is fully KDM secure of type T.*

Sketch. In the CPA setting it suffices to prove full KDM security with respect to all circuits of single output. We show how to convert an attacker which sends arbitrary KDM queries into one which uses only queries from $\mathcal{F}_k$. Let h be a circuit of size t, which is wlog composed of NAND gates, and let h_i denote the function computed by the i-th gate of h, where gates are ordered under some topological ordering. We translate a KDM query for h into t KDM calls to $\mathcal{F}_k$ by traversing the circuit from bottom to top in a gate by gate manner preserving the following invariant: The i-th query will be answered by a ciphertext C_i such that, if the oracle is in the real mode $C_i = \mathsf{E}_{\mathsf{pk}}(h_i(\mathsf{sk}))$ and if it is in the fake mode $C_i = \mathsf{E}_{\mathsf{pk}}(0)$. For an input gate, this can be achieved directly by making a single KDM query with a single-output projection. To do this for an internal gate h_ℓ whose input wires are connected to h_i and h_j for some $i, j < \ell$, we use a KDM query to f_{C_i,C_j}. □

Acknowledgement. We thank Iftach Haitner, Yuval Ishai, and the anonymous referees for their helpful comments.

References

[ABBC10] Tolga Acar, Mira Belenkiy, Mihir Bellare, and David Cash. Cryptographic agility and its relation to circular encryption. In *Advances in Cryptology – EUROCRYPT 2010*, pages 403–422, 2010.

[ABHS09] Pedro Adão, Gergei Bana, Jonathan Herzog, and Andre Scedrov. Soundness and completeness of formal encryption: The cases of key cycles and partial information leakage. *Journal of Computer Security*, 17(5):737–797, 2009.

[ACPS09] Benny Applebaum, David Cash, Chris Peikert, and Amit Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In *Advances in Cryptology – CRYPTO 2009*, pages 595–618, 2009.

[AIK06a] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. Computationally private randomizing polynomials and their applications. *Journal of Computional Complexity*, 15(2):115–162, 2006.

[AIK06b] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. Cryptography in NC^0. *SIAM Journal on Computing*, 36(4):845–888, 2006.

[AR07] Martín Abadi and Phillip Rogaway. Reconciling two views of cryptography (the computational soundness of formal encryption). *Journal of Cryptology*, 20(3):395, July 2007.

[BDU08] Michael Backes, Markus Dürmuth, and Dominique Unruh. OAEP is secure under key-dependent messages. In *Advances in Cryptology – ASIACRYPT 2008*, pages 506–523, 2008.

[BG99] Amos Beimel and Anna Gál. On arithmetic branching programs. *Journal of Computer and System Sciences*, 59(2):195–220, 1999.

[BG10] Zvika Brakerski and Shafi Goldwasser. Circular and leakage resilient public-key encryption under subgroup indistinguishability (or: Quadratic residuosity strikes back). In *Advances in Cryptology – CRYPTO 2010*, pages 1–20, 2010.

[BGK11] Zvika Brakerski, Shafi Goldwasser, and Yael Kalai. Circular-secure encryption beyond affine functions. In *TCC 2011: 8th Theory of Cryptography Conference*, 2011.

[BHHI10] Boaz Barak, Iftach Haitner, Dennis Hofheinz, and Yuval Ishai. Bounded key-dependent message security. In *Advances in Cryptology – EUROCRYPT 2010*, pages 423–444, 2010.

[BHHO08] Dan Boneh, Shai Halevi, Michael Hamburg, and Rafail Ostrovsky. Circular-secure encryption from decision Diffie-Hellman. In *Advances in Cryptology – CRYPTO 2008*, pages 108–125, 2008.

[BPS07] Michael Backes, Birgit Pfitzmann, and Andre Scedrov. Key-dependent message security under active attacks - BRSIM/UC-soundness of symbolic encryption with key cycles. In *Proceedings of 20th IEEE Computer Security Foundation Symposium (CSF)*, 2007.

[BRS02] John Black, Phillip Rogaway, and Thomas Shrimpton. Encryption-scheme security in the presence of key-dependent messages. In *SAC 2002: 9th Annual International Workshop on Selected Areas in Cryptography*, pages 62–75, 2002.

[CCS09] Jan Camenisch, Nishanth Chandran, and Victor Shoup. A public key encryption scheme secure against key dependent chosen plaintext and adaptive chosen ciphertext attacks. In *Advances in Cryptology – EUROCRYPT 2009*, pages 351–368, 2009.

[CFIK03] Ronald Cramer, Serge Fehr, Yuval Ishai, and Eyal Kushilevitz. Efficient multi-party computation over rings. In *Advances in Cryptology – EUROCRYPT 2003*, pages 596–613, 2003.

[CL01] Jan Camenisch and Anna Lysyanskaya. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In *Advances in Cryptology – EUROCRYPT 2001*, pages 93–118, 2001.

[DDN91] Danny Dolev, Cynthia Dwork, and Moni Naor. Non-malleable cryptography (extended abstract). In *23rd Annual ACM Symposium on Theory of Computing (STOC)*, pages 542–552, 1991.

[EGL85] Shimon Even, Oded Goldreich, and Abraham Lempel. A randomized protocol for signing contracts. *Communications of the Association for Computing Machinery*, 28, 1985.

[Gen09] Craig Gentry. Fully homomorphic encryption using ideal lattices. In *41st Annual ACM Symposium on Theory of Computing (STOC)*, pages 169–178, 2009.

[GKM+00] Yael Gertner, Sampath Kannan, Tal Malkin, Omer Reingold, and Mahesh Viswanathan. The relationship between public key encryption and oblivious transfer. In *41st Annual Symposium on Foundations of Computer Science (FOCS)*, 2000.

[GM84] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270–299, 1984.

[Gol01] Oded Goldreich. *Foundations of Cryptography: Basic Tools*. Cambridge University Press, 2001.

[Gol04] Oded Goldreich. *Foundations of Cryptography: Basic Applications*. Cambridge University Press, 2004.

[HH09] Iftach Haitner and Thomas Holenstein. On the (im)possibility of key dependent encryption. In *TCC 2009: 6th Theory of Cryptography Conference*, pages 202–219, 2009.

[HK07] Shai Halevi and Hugo Krawczyk. Security under key-dependent inputs. In *ACM CCS 07: 14th Conference on Computer and Communications Security*, pages 466–475, 2007.

[HU08] Dennis Hofheinz and Dominique Unruh. Towards key-dependent message security in the standard model. In *Advances in Cryptology – EUROCRYPT 2008*, pages 108–126, 2008.

[IK00] Yuval Ishai and Eyal Kushilevitz. Randomizing polynomials: A new representation with applications to round-efficient secure computation. In *41st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 294–304, 2000.

[IK02] Yuval Ishai and Eyal Kushilevitz. Perfect constant-round secure computation via perfect randomizing polynomials. In *ICALP 2002: 29th International Colloquium on Automata, Languages and Programming*, pages 244–256, 2002.

[IR88] Russel Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *Advances in Cryptology – CRYPTO'88*, pages 8–26, 1988.

[NY90] Moni Naor and Moti Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *22nd Annual ACM Symposium on Theory of Computing (STOC)*, pages 427–437, 1990.

[Rab79] Michael O. Rabin. Digitalized signatures and public key functions as intractable as factoring. Technical Report 212, LCS, MIT, 1979.

[RS91] Charles Rackoff and Daniel R. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In *Advances in Cryptology – CRYPTO'91*, pages 433–444, 1991.

[vDGHV10] Marten van Dijk, Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. Fully homomorphic encryption over the integers. In *Advances in Cryptology – EUROCRYPT 2010*, pages 24–43, 2010.

[Yao86] A. C. Yao. How to generate and exchange secrets. In *27th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 162–167, 1986.

A From affine functions to projections

Converting affine-security to security under single-output projections is immediate if the affine functions are defined over the binary field $\mathbb{F}_2$ (as in the LPN based scheme of [CCS09]), but can also be established in more general cases, which capture all known schemes, as follows.

Suppose that we have KDM security for affine functions over a ring $\mathcal{R}$ (which may, in general, be a ring family whose size depends on the security parameter). Namely, the scheme encrypts ring elements, the secret-key consists of n ring elements $\mathsf{sk} = (\mathsf{sk}_i)_{i\in[n]}$, and KDM security holds with respect to affine functions: $f_{a,b} : \mathsf{sk} \mapsto (\sum_i a_i \cdot \mathsf{sk}_i) + b$. Let $\langle \mathsf{sk} \rangle = (\langle \mathsf{sk} \rangle_1, \ldots, \langle \mathsf{sk} \rangle_k)$ denote the representation of the secret key sk as a k-bit string. We will show that affine functions can encode projections as long as the bit representation is "meaningful" in terms of the group $\mathcal{R}$, and so, in this case we can apply Thm. 4.1 and get a new scheme with KDM security for projections. Formally, we distinguish between the following two cases.

In the first case, each key element sk_i is represented by a single bit $\langle \mathsf{sk} \rangle_i$. That is, there exists a list of non-zero public elements $g = (g_1, \ldots, g_\ell)$ such that $\mathsf{sk}_i = g_i \cdot \langle \mathsf{sk} \rangle_i$ (where, for the sake of ring arithmetics, we think of a bit β as either the zero element or the one element of the ring). This is the case, for example, in the schemes of [BHHO08] and [BG10]. Let us assume that $\langle \mathsf{sk} \rangle$ is used as the bit-representation of the key. Then we can use the RE approach to amplify affine security (over $\mathcal{R}$) into security against projections by showing that the former encodes the latter. Formally, every projection function $f_{i,\sigma}(\langle \mathsf{sk} \rangle) = \langle \mathsf{sk} \rangle_i \oplus \sigma$ is encoded by $\hat{f}_{i,\sigma}(\langle \mathsf{sk} \rangle; r) = (\langle \mathsf{sk} \rangle_i - \sigma) \cdot g_i \cdot r$ where r is a randomly chosen non-zero element of $\mathcal{R}$. This encoding enjoys perfect correctness via a universal decoder (a zero element is decoded to 0 and any other element is decoded to 1) and perfect privacy via a universal simulator (given an output β of f simulate the corresponding output of $\hat{f}$ by multiplying it with a random non-zero element). Moreover, when the randomness is fixed we get a linear function over $\mathcal{R}$. Hence, by Thm. 4.1, the security of the scheme can be amplified to hold with respect to single-output projections.

We proceed with the second case. Let us assume that the mapping from sk to each bit of the representation $\langle \mathsf{sk} \rangle$ can be computed by a polynomial-size arithmetic branching program (ABP) (see [BG99, CFIK03]) over $\mathcal{R}$. (This is possible in a trivial way whenever the ring is of polynomial size, as in the LWE-based scheme of [ACPS09].) Then, the mappings $f_{i,0} : \mathsf{sk} \mapsto \langle \mathsf{sk} \rangle_i$ and $f_{i,1} : \mathsf{sk} \mapsto 1 - \langle \mathsf{sk} \rangle_i$ can also be computed by a polynomial-size ABP. Hence, by [CFIK03], there exists a perfect (universal) RE $\hat{f}_{i,\sigma}(\mathsf{sk}; r)$ such that for every fixed choice of r, $\hat{f}_{r,i,\sigma}(\mathsf{sk}) = \hat{f}_{i,\sigma}(\mathsf{sk}; r)$ is an affine function over $\mathcal{R}$. Hence, by Thm. 4.1, the security of the scheme can be amplified to hold with respect to single-output projections.