Shift-Type Homomorphic Encryption and Its Application to Fully Homomorphic Encryption

Frederik Armknecht[1], Stefan Katzenbeisser[2], and Andreas Peter[2]

[1] Theoretical Computer Science and IT Security Group
Universität Mannheim, Germany
armknecht@uni-mannheim.de
[2] Security Engineering Group
Technische Universität Darmstadt and CASED, Germany
{stefan.katzenbeisser,andreas.peter}@cased.de

Abstract. This work addresses the characterization of homomorphic encryption schemes both in terms of security and design. In particular, we are interested in currently existing fully homomorphic encryption (FHE) schemes and their common structures and security. Our main contributions can be summarized as follows:

- We define a certain type of homomorphic encryption that we call *shift-type* and identify it as the basic underlying structure of *all existing* homomorphic encryption schemes. It generalizes the already known notion of shift-type *group* homomorphic encryption.
- We give an IND-CPA characterization of all shift-type homomorphic encryption schemes in terms of an abstract subset membership problem.
- We show that this characterization carries over to all leveled FHE schemes that arise by applying Gentry's bootstrapping technique to shift-type homomorphic encryption schemes. Since this is the common structure of all existing schemes, our result actually characterizes the IND-CPA security of all existing bootstrapping-based leveled FHE.
- We prove that the IND-CPA security of FHE schemes that offer a certain type of circuit privacy (for FHE schemes with a binary plaintext space we require circuit privacy for a single AND-gate and, in fact, all existing binary-plaintext FHE schemes offer this) and are based on Gentry's bootstrapping technique is *equivalent* to the circular security of the underlying bootstrappable scheme.

Keywords: Public-Key Cryptography, Homomorphic Encryption, Semantic Security, Circular Security.

1 Introduction

Homomorphic encryption is one of the central topics in public-key cryptography as it allows for the evaluation of certain circuits over encrypted data without the

A. Mitrokotsa and S. Vaudenay (Eds.): AFRICACRYPT 2012, LNCS 7374, pp. 234–251, 2012.

ability to decrypt. Many important applications, such as Outsourcing of Computation [18], Electronic Voting [5, 10, 12, 13], Private Information Retrieval [26], Oblivious Polynomial Evaluation [28], and Multiparty Computation [11] are based on this primitive. In the past decades, a substantial number of homomorphic encryption schemes have been proposed (see survey [17]). The majority of these schemes are *group homomorphic*, i.e., the plaintext and ciphertext spaces are groups and the decryption function is a group homomorphism. In other words, group homomorphic schemes allow the evaluation of circuits, consisting solely of group operations in the plaintext group, over the ciphertexts. Recently, Armknecht et al. [3] gave a comprehensive and complete framework of all currently existing group homomorphic encryption schemes and, in particular, gave characterization both in terms of security and design.

Concerning the construction and characterization of more general homomorphic encryption schemes on the other hand, there is still a lot of work to be done. Much effort has been devoted to the construction of so-called *fully homomorphic encryption* (FHE) schemes [7–9, 15, 19, 21–24, 27, 29, 30], which allow the evaluation of *any* circuit (not just consisting of group operation gates as it is the case for group homomorphic encryption) over the ciphertexts. The first such scheme has been proposed by Gentry [20] which uses a certain technique that subsequently has been the basis of all currently existing FHE schemes. Gentry's technique is called *bootstrapping* and can be summarized in the following 2 steps:

1. Construct a *bootstrappable* homomorphic encryption scheme, i.e., a scheme allowing the evaluation of low-degree polynomials over the ciphertexts and, in particular, the evaluation of its own decryption circuit together with one additional set of gates like AND and NOT.
2. Use the *bootstrapping* technique on this scheme to make it fully homomorphic. This technique refreshes a given ciphertext so that it can further be used for evaluation. Usually, ciphertexts are created by adding noise to a given plaintext and once the noise gets too big, the ciphertexts have to be refreshed to reduce the noise again – this is what bootstrapping achieves.

Essentially, the same bootstrapping technique (with minor differences) can be used to construct so-called *leveled* FHE schemes – a relaxed notion of FHE. Such schemes can evaluate all circuits up to a certain depth.[1]

Concerning security, the resulting FHE schemes can be proven secure in terms of IND-CPA (also known as semantic security) under certain assumptions. For a leveled FHE scheme, IND-CPA security follows from the IND-CPA security of the underlying bootstrappable scheme. For a "pure" FHE scheme, we require the

[1] The recent leveled FHE scheme by Brakerski et al. [7] is built without the bootstrapping technique. It is the only scheme known so far that deviates from Gentry's blueprint. We stress that we focus on schemes that follow the bootstrapping approach.

We also want to point out that we are not concerned with the "squashing of the decryption circuit" step that Gentry originally proposed in his blueprint. The schemes [7–9, 21] circumvent this "squashing" step but still rely on bootstrapping which is the technique we focus on in this paper.

underlying bootstrappable scheme to be *circular secure* which roughly means that the scheme remains secure even if the adversary gets to see the bits of the secret key encrypted under the corresponding public key.

1.1 Contribution and Related Work

In this paper, we address the above mentioned topic of characterizing the security and the design of homomorphic encryption schemes in the context of FHE, thereby extending the work of Armknecht et al. [3] on group homomorphic encryption to these more general homomorphic schemes:

1. We identify and formalize the underlying structure of *all existing* homomorphic schemes and call such schemes *shift-type* homomorphic. It is a natural generalization of the shift-type *group* homomorphic schemes introduced in [3].
2. We give an IND-CPA security characterization of all shift-type encryption schemes in terms of an abstract subset membership problem. In comparison to the proof of the IND-CPA security characterization of group homomorphic schemes in [3] that heavily relies on the group homomorphic property, it is interesting to see that our result shows that it is actually the shift-type structure of the encryption algorithm that gives the IND-CPA characterization (and not the homomorphic property of the decryption).
3. We show that this characterization carries over to all leveled FHE schemes that are based on Gentry's bootstrapping technique applied to shift-type homomorphic schemes. Since all existing schemes are shift-type homomorphic, this gives a characterization of all existing bootstrapping-based schemes. Additionally, our result has the nice application that once an FHE scheme is constructed using Gentry's technique, the underlying hardness assumption yielding IND-CPA security immediately comes out of this characterization.
4. We prove that the IND-CPA security of "pure" FHE schemes that are based on Gentry's bootstrapping technique and that are *circuit-private for a certain small set of circuits* (meaning that a ciphertext that is the evaluation of ciphertexts under one of these circuits does not reveal any information about the used circuit, even when the secret key is known) is *equivalent* to the circular security of the underlying bootstrappable scheme. We note that Gentry [19, Theorem 4.3.2] has already proved one of the directions, namely that *if* the underlying bootstrappable scheme is circular secure, *then* the resulting FHE scheme is IND-CPA secure. Interestingly enough, all existing FHE schemes where the plaintext space is $\{0,1\}$ are circuit-private for this special set of circuits.

   Our characterization result gives another important relation between the notion of circular security and IND-CPA security. Moreover, it shows that when the resulting FHE scheme (using the bootstrapping technique) gives a certain "minimal" circuit privacy, the circular security is not only sufficient but also necessary. Therefore it underlines the importance of Brakerski et al.'s work [9]. Therein, they construct a "somewhat" homomorphic scheme

(i.e., a homomorphic encryption scheme for low-degree polynomials only) that is provably circular secure. However, this scheme is not bootstrappable. By using standard techniques, they turn it into a bootstrappable scheme. Unfortunately, the proof of circular security gets lost in this transformation. We note that, even with Brakerski et al.'s result, we still do not know how to prove circular security for given IND-CPA secure bootstrappable encryption schemes. So currently existing FHE schemes still rely on the assumption that the circular security and the IND-CPA security of their underlying bootstrappable schemes are equivalent.

In regard to circular security, there are two other papers important to mention. First, there is the work by Barak et al. [4]. Therein, they show that any FHE scheme that is circular secure is actually *fully* KDM secure (i.e., the adversary gets evaluations of *arbitrary* functions on the private key). Second, the work by Applebaum [2] shows that any *simulatable* fully KDM secure scheme (a notion which is even stronger than fully KDM security) is also fully homomorphic. Furthermore, it shows that the same bootstrapping technique that Gentry uses to build FHE schemes can be used to construct fully KDM secure encryption schemes.

We stress that in contrast to the just mentioned works, we prove that the IND-CPA security of FHE schemes (that arise by using the bootstrapping technique) which give a certain "minimal" circuit privacy, is equivalent to the circular security of the underlying bootstrappable scheme.

To complete the list of related works on FHE, we want to mention an approach by Aguilar Melchor et al. [1], which uses so-called "chained encryption schemes" and differs from the bootstrapping technique. Although it is likely that our results extend to their method, we do not cover this here, since the computational cost of their solution is exponential in the number of multiplications that the scheme should be able to evaluate over the ciphertexts (formally, they do not achieve leveled FHE but only *constant-bounded* FHE).

1.2 Outline

Throughout the paper, we use standard notation and definitions that are summarized in Section 2. Therein, we also formally define public-key homomorphic encryption, recall its standard security notion, and define a class of subset membership problems. In Section 3, we define *shift-type* homomorphic encryption schemes and characterize their security in terms of these subset membership problems. Finally, Section 4 is entirely devoted to FHE. First, we recall Gentry's bootstrapping technique for leveled FHE schemes and show that our security characterization for shift-type homomorphic encryption carries over to such schemes. Second, we prove the equivalence of a "pure" bootstrapping-based FHE scheme being IND-CPA secure and the underlying bootstrappable scheme being circular secure. Third, we give a brief overview on existing FHE schemes and their underlying shift-type structures, while focusing on the scheme by van Dijk et al. [15] for a better conceptual understanding. We conclude in Section 5.

2 Preliminaries

2.1 Notation

We write $x \longleftarrow X$ if X is a random variable or distribution and x is to be chosen randomly from X according to its distribution. In the case where X is solely a set, $x \xleftarrow{U} X$ denotes that x is chosen uniformly at random from X. For an algorithm $\mathcal{A}$ we write $x \longleftarrow \mathcal{A}(y)$ if $\mathcal{A}$ outputs x on fixed input y according to $\mathcal{A}$'s distribution. If $\mathcal{A}$ has access to an oracle $\mathcal{O}$, we write $\mathcal{A}^{\mathcal{O}}$. Sometimes, we need to specify the randomness of a probabilistic algorithm $\mathcal{A}$ explicitly. To this end, we interpret $\mathcal{A}$ as a deterministic algorithm $\mathcal{A}(y, r)$, which has access to random values r from some randomness space Rnd.

By a *description* of a finite set X we mean an efficient sampling algorithm (according to some distribution) for the set X. If X is a group, a *description* of X additionally includes the neutral element and a set of efficient algorithms that allow us to perform the usual group operation on X and the inversion of group elements. We abuse notation and write X both for the description and for the set itself. If a description of X is given, we denote sampling from X according to the distribution given by the sampling algorithm of the description by $x \longleftarrow X$.

For given probabilistic algorithms $\mathcal{A}$ and Gen that run in time polynomial in a given parameter λ, we describe computational problems P through experiments $\mathbf{Exp}^{\mathrm{P}}_{\mathcal{A},\mathsf{Gen}}(\lambda)$. The output of $\mathbf{Exp}^{\mathrm{P}}_{\mathcal{A},\mathsf{Gen}}(\lambda)$ is always defined to be a single bit. We then say that *problem* P *is hard (relative to* Gen*)* if for all probabilistic polynomial time (PPT) algorithms $\mathcal{A}$ there exists a negligible (in λ) function $\mathtt{negl}$ such that

$$\left| \Pr[\mathbf{Exp}^{\mathrm{P}}_{\mathcal{A},\mathsf{Gen}}(\lambda) = 1] - \frac{1}{2} \right| \leq \mathtt{negl}(\lambda).$$

We recall that a public-key encryption scheme $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ consists of a PPT key generation algorithm KeyGen which generates a pair $(\mathsf{pk}, \mathsf{sk})$ of corresponding public and private keys, a PPT encryption algorithm Enc and a deterministic PT decryption algorithm Dec with the usual correctness condition. We denote the set of plaintexts by $\mathcal{P}$ and the set of ciphertexts by $\widehat{\mathcal{C}}$.

2.2 Public-Key Homomorphic Encryption Schemes

We briefly recall the notion of public-key *homomorphic encryption* (see [25, Definition 5] or [20, Definition 1]).

Definition 1. *A public-key encryption scheme* $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ *is called* homomorphic for a set of circuits $\mathbb{C} = \mathbb{C}[\lambda]$ *(that depends on the security parameter* λ*), if there exists a PPT algorithm* Eval *(that outputs a ciphertext and takes as input public keys* pk *from the output of* KeyGen*, circuits* $C \in \mathbb{C}(\lambda)$ *and ciphertexts* $(c_1, \ldots, c_r)$ *with* $c_i \longleftarrow \mathsf{Enc}_{\mathsf{pk}}(m_i)$ *for some* $m_i \in \mathcal{P}$*,* $i = 1, \ldots, r$*) such that for every output* $(\mathsf{pk}, \mathsf{sk})$ *of* $\mathsf{KeyGen}(\lambda)$ *it holds that (*correctness condition*)*

$$\mathsf{Dec}_{\mathsf{sk}}(\mathsf{Eval}_{\mathsf{pk}}(C, c_1, \ldots, c_r)) = C(m_1, \ldots, m_r),$$

except with negligible (in λ*) probability over the random coins in* Eval*.*

The minimal security property that we require such schemes to have is *semantic security* (or IND-CPA security), which is defined in exactly the same way as for standard public-key encryption schemes and is captured by the following experiment between a challenger and an adversary $\mathcal{A}$:

Experiment $\mathbf{Exp}^{\text{ind-cpa}}_{\mathcal{A},\mathsf{KeyGen}}(\lambda)$:

1. $(\mathsf{pk}, \mathsf{sk}) \longleftarrow \mathsf{KeyGen}(\lambda)$
2. $(m_0, m_1, s) \longleftarrow \mathcal{A}_1(\mathsf{pk})$ where $m_0, m_1 \in \mathcal{P}$ and s a state of $\mathcal{A}_1$
3. Choose $b \xleftarrow{U} \{0,1\}$ and compute $c \longleftarrow \mathsf{Enc}_{\mathsf{pk}}(m_b)$
4. $d \longleftarrow \mathcal{A}_2(m_0, m_1, s, c)$ where $d \in \{0,1\}$
5. The output of the experiment is defined to be 1 if $d = b$ and 0 otherwise.

We say that $\mathcal{E}$ is IND-CPA *secure (relative to* KeyGen*)* if the advantage

$$\left|\Pr[\mathbf{Exp}^{\text{ind-cpa}}_{\mathcal{A},\mathsf{KeyGen}}(\lambda) = 1] - \frac{1}{2}\right| \text{ is negligible for all PPT algorithms } \mathcal{A}.$$

2.3 The Subset Membership Problem

The *Subset Membership Problem* (SMP) was introduced by Cramer and Shoup in [14]: Let Gen be a PPT algorithm that takes a security parameter λ as input and outputs descriptions $(\mathcal{S}, \mathcal{N})$ where $\mathcal{N}$ is a non-trivial, proper subset of a finite set $\mathcal{S}$. Consider the following experiment for a given algorithm Gen, algorithm $\mathcal{A}$ and parameter λ:

Experiment $\mathbf{Exp}^{\mathsf{SMP}}_{\mathcal{A},\mathsf{Gen}}(\lambda)$:

1. $(\mathcal{S}, \mathcal{N}) \longleftarrow \mathsf{Gen}(\lambda)$
2. Choose $b \xleftarrow{U} \{0,1\}$. If $b = 1$: $z \longleftarrow \mathcal{S}$. Otherwise: $z \longleftarrow \mathcal{N}$.
3. $d \longleftarrow \mathcal{A}(\mathcal{S}, \mathcal{N}, z)$ where $d \in \{0,1\}$
4. The output of the experiment is defined to be 1 if $d = b$ and 0 otherwise.

This experiment defines the *Subset Membership Problem* SMP *(relative to* Gen*)* which, informally, states that given $(\mathcal{S}, \mathcal{N}, z)$ where $z \longleftarrow \mathcal{S}$, one has to decide whether $z \in \mathcal{N}$ or not.

3 Shift-Type Homomorphic Encryption

Informally, an encryption scheme is *shift-type homomorphic* if the plaintexts form a non-trivial (say multiplicative) group, encryptions of *known* plaintexts can be transformed (or "shifted") to encryptions of 1, and if the same transformation is applied to a random ciphertext, the resulting ciphertext is still random.

Definition 2. *A public-key encryption scheme* $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ *is called* shift-type homomorphic*, if for every output* $(\mathsf{pk}, \mathsf{sk})$ *of* $\mathsf{KeyGen}(\lambda)$*, the plaintext space* $\mathcal{P}$ *and the ciphertext space* $\widehat{\mathcal{C}}$ *are (multiplicatively written) non-trivial*

groups[2] *such that the public key* pk *contains a description of a subset* $\mathcal{N} \subseteq \widehat{\mathcal{C}}$ *and an efficient injective homomorphism* $\varphi : \mathcal{P} \to \widehat{\mathcal{C}}$ *so that for all plaintexts* $m \in \mathcal{P}$,

$$\mathsf{Enc}_{\mathsf{pk}}(m) \text{ outputs } \varphi(m) \cdot n,$$

where $n \longleftarrow \mathcal{N}$.

We denote the *set of all encryptions* by

$$\mathcal{C} := \{\mathsf{Enc}_{\mathsf{pk}}(m) \mid m \in \mathcal{P}\} \subseteq \widehat{\mathcal{C}}$$

and sometimes call its elements *fresh* ciphertexts/encryptions. Since φ is a homomorphism, we know that $\mathcal{N}$ is actually a subset of $\mathcal{C}$.

Remark 1.

1. The concept of shift-type homomorphic encryption is very similar to the concept of "adding noise" to the plaintext. Here, we are a bit more general, as we allow homomorphic manipulation of the plaintext prior to adding (or multiplying in our case) noise. The "noise" corresponds to the elements of the subset $\mathcal{N}$.
2. The name "shift-type" is due to the fact that we can "shift" encryptions of known plaintexts to encryptions of arbitrary plaintexts under the same noise: Let $c := \varphi(m) \cdot n$ be an encryption of message $m \in \mathcal{P}$. Then, by computing $c' := \varphi(m' \cdot m^{-1}) \cdot c$ for some arbitrary message $m' \in \mathcal{P}$, we receive an encryption $c' = \varphi(m') \cdot n$ of message m' under the same noise n, by using the homomorphic property of φ.
3. Definition 2 is a natural generalization of the notion of shift-type *group* homomorphic encryption as introduced in [3]. For the latter, the decryption procedure is a group homomorphism and the mapping φ is the encryption algorithm under a fixed randomness.
4. We stress that the shift-type structure of the encryption algorithm is not implied by a group homomorphic encryption scheme (recall that this means that the decryption procedure is a group homomorphism, see [3] for details). Although *all existing* IND-CPA secure homomorphic schemes do have this structure, it is easy to construct a group homomorphic scheme (which is insecure in terms of IND-CPA) that does not: Let $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ be an arbitrary IND-CPA secure group homomorphic encryption scheme with randomness space Rnd (e.g., ElGamal's scheme [16]) and let r^* be some fixed value in Rnd. We modify its encryption algorithm as follows and denote it by Enc^*: On input a plaintext m, $\mathsf{Enc}^*(m)$ chooses a random bit $b \in \{0, 1\}$ and some random $r \in \mathsf{Rnd}$. If $b = 1$ or $m = 1$, $\mathsf{Enc}^*(m)$ outputs $\mathsf{Enc}(m, r)$. Otherwise, it outputs $\mathsf{Enc}(m, r^*)$.

   It is easy to see that the modified scheme $\mathsf{Enc}^* = (\mathsf{KeyGen}, \mathsf{Enc}^*, \mathsf{Dec})$ is group homomorphic but *not* IND-CPA secure. On the other hand, it is also not shift-type homomorphic. Interestingly enough, it is an open question

[2] We assume that descriptions of $\mathcal{P}$ and $\widehat{\mathcal{C}}$ are contained in the public key pk. As described in Section 2.1, sampling from $\mathcal{P}$ (resp. $\widehat{\mathcal{C}}$) using the (corresponding) sampling algorithm of the description is denoted by $m \longleftarrow \mathcal{P}$ (resp. $c \longleftarrow \widehat{\mathcal{C}}$).

whether the shift-type structure is implied by the IND-CPA security of a given group homomorphic encryption scheme – meaning that if the output distribution of the encryption algorithm is computationally distinguishable from the shift-type structure, then the given group homomorphic scheme is insecure in terms of IND-CPA.

Next, we will characterize the IND-CPA security of such schemes. We note that by saying that the Subset Membership Problem (SMP) as defined in Section 2.3 is hard relative to KeyGen for a key generator KeyGen of some shift-type homomorphic encryption scheme, we mean that SMP is hard for $(\mathcal{C}, \mathcal{N})$. For a given shift-type homomorphic encryption scheme, we use the notation

$$\mathcal{C}_m := \{c \in \mathcal{C} \mid \mathsf{Dec}_{\mathsf{sk}}(c) = m\}$$

to denote the set of ciphertexts decrypting to $m \in \mathcal{P}$. In particular, we have $\mathcal{N} = \mathcal{C}_1$ in this notation. We are now in a position to prove a characterization of IND-CPA security of such schemes.

Theorem 1 (IND-CPA Security of Shift-Type Schemes). *For a shift-type homomorphic encryption scheme* $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ *we have:*

$$\mathcal{E} \textit{ is } \mathsf{IND\text{-}CPA} \textit{ (rel. to } \mathsf{KeyGen}) \iff \mathsf{SMP} \textit{ is hard (rel. to } \mathsf{KeyGen})$$

Proof. "⇐": Assume that $\mathcal{E}$ is not IND-CPA secure, i.e. there exists a PPT algorithm $\mathcal{A}^{\text{cpa}} = (\mathcal{A}_1^{\text{cpa}}, \mathcal{A}_2^{\text{cpa}})$ that breaks the security with non-negligible advantage $f(\lambda)$. We derive a contradiction by constructing a PPT algorithm $\mathcal{A}^{\text{smp}}$ that successfully solves SMP with advantage $\frac{1}{2}f(\lambda)$.

Since SMP and IND-CPA are both considered relative to KeyGen, $\mathcal{A}^{\text{smp}}$ can simply forward the public key pk of the output of $\mathsf{KeyGen}(\lambda)$ to $\mathcal{A}_1^{\text{cpa}}$. Next, $\mathcal{A}_1^{\text{cpa}}$ outputs two messages $m_0, m_1 \in \mathcal{P}$ to $\mathcal{A}^{\text{smp}}$. The SMP challenger chooses a bit $b \xleftarrow{U} \{0,1\}$ and sends the challenge $c \in \mathcal{C}$ to $\mathcal{A}^{\text{smp}}$, who then chooses a bit $d \xleftarrow{U} \{0,1\}$ and sends the challenge $c_d := \varphi(m_d) \cdot c$ to $\mathcal{A}_2^{\text{cpa}}$. Now, $\mathcal{A}_2^{\text{cpa}}$ outputs a bit d' and sends it back to $\mathcal{A}^{\text{smp}}$ which sends $b' := d \oplus d'$ to the SMP challenger.

We have the following relations: If $b = 0$, then $c \in \mathcal{N} = \mathcal{C}_1$ and $c_d \in \mathcal{C}_{m_d}$ (a fresh encryption of m_d) by definition. Hence, $\mathcal{A}_2^{\text{cpa}}$ makes the right guess with advantage $f(\lambda)$, i.e., $\Pr[b' = b | b = 0] \geq \frac{1}{2} + f(\lambda)$. If $b = 1$, then $c \in \mathcal{C}$, meaning that it is a fresh encryption (by definition of the set $\mathcal{C}$) of some random message m. But φ is a homomorphism and so c_d is a fresh encryption of (the random message) $m_d \cdot m$. Hence, $\mathcal{A}_2^{\text{cpa}}$ guesses d with no advantage, i.e. $\Pr[b' = b | b = 1] = \frac{1}{2}$. We have shown:

$$\begin{aligned}\Pr[\mathbf{Exp}^{\mathsf{SMP}}_{\mathcal{A}^{\text{smp}},\mathsf{Gen}}(\lambda) = 1] &= \sum_{\beta \in \{0,1\}} \Pr[b' = b | b = \beta] \cdot \Pr[b = \beta] \\ &\geq \frac{1}{2} \cdot \left(\frac{1}{2} + f(\lambda) + \frac{1}{2}\right) = \frac{1}{2} + \frac{1}{2} f(\lambda).\end{aligned}$$

"⇒": For the converse, we assume that there is a PPT algorithm $\mathcal{A}^{\text{smp}}$ that solves SMP with advantage $f(\lambda)$. Similarly to what we have done above, we

construct a PPT algorithm $\mathcal{A}^{\text{cpa}} = (\mathcal{A}_1^{\text{cpa}}, \mathcal{A}_2^{\text{cpa}})$ that successfully breaks the IND-CPA security with advantage $f(\lambda)$.

Again as above, $\mathcal{A}_1^{\text{cpa}}$ forwards the output of $\mathsf{KeyGen}(\lambda)$ to $\mathcal{A}^{\text{smp}}$. Next, $\mathcal{A}_1^{\text{cpa}}$ outputs two random messages $m_0, m_1 \in \mathcal{P}$. The IND-CPA challenger chooses a bit $b \xleftarrow{U} \{0,1\}$ and sends the challenge $c_b \longleftarrow \mathsf{Enc}_{\mathsf{pk}}(m_b)$ to $\mathcal{A}_2^{\text{cpa}}$, who then computes $c := \varphi(m_0^{-1}) \cdot c_b \in \mathcal{C}$ and sends the challenge c to $\mathcal{A}^{\text{smp}}$. Now, $\mathcal{A}^{\text{smp}}$ returns a bit d' to $\mathcal{A}_2^{\text{cpa}}$ that then outputs $b' := d'$ to the IND-CPA challenger.

We have the following relations: If $b = 0$, then $c \in \mathcal{C}_1 = \mathcal{N}$ and $\mathcal{A}^{\text{smp}}$ guesses b with advantage $f(\lambda)$, i.e. $\Pr[b' = b | b = 0] \geq \frac{1}{2} + f(\lambda)$. If $b = 1$, then c is a random element in $\mathcal{C}$ and $\mathcal{A}^{\text{smp}}$ guesses b again with advantage $f(\lambda)$, i.e. $\Pr[b' = b | b = 1] \geq \frac{1}{2} + f(\lambda)$. Therefore, we have shown:

$$\begin{aligned}\Pr[\mathbf{Exp}^{\text{ind-cpa}}_{\mathcal{A}^{\text{cpa}},\mathsf{Gen}}(\lambda) = 1] &= \sum_{\beta \in \{0,1\}} \Pr[b' = b | b = \beta] \cdot \Pr[b = \beta] \\ &\geq \frac{1}{2} \cdot (1 + 2f(\lambda)) = \frac{1}{2} + f(\lambda).\end{aligned}$$

□

4 Fully Homomorphic Encryption (FHE)

An encryption scheme $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ that is homomorphic for *all* circuits (in terms of Definition 1) is called *fully homomorphic* (FHE = Fully Homomorphic Encryption). To rule out trivial FHE schemes $\mathcal{E}$, e.g., where Eval simply outputs its input circuit C together with its input ciphertexts and Dec takes circuits C as input as well and simply outputs the evaluation of C on the decryptions of the plugged-in ciphertexts, we require the additional property of *compactness* (cf. [19, Definition 2.1.2]). Informally this means that the size of the output of Eval does not depend on the size of the circuit it evaluates.

We recall this notion in the more general context of encryption schemes that are homomorphic for a given set of circuits.

Definition 3. *Let* $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ *be an encryption scheme that is homomorphic for a set of circuits* $\mathbb{C} = \mathbb{C}[\lambda]$. $\mathcal{E}$ *is called* compact, *if* Dec *can be expressed as a circuit of size at most* $p(\lambda)$ *for some polynomial* p.

With this definition in mind, we can formalize the notion of FHE:

Definition 4. *An encryption scheme* $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ *that is homomorphic for all circuits and compact is called* fully homomorphic.

We note that *all* currently existing FHE schemes in terms of Definition 4 (namely, [7–9, 15, 19, 21–24, 27, 29, 30]) are variants of a scheme proposed by Gentry [20] and do all have the property that decryption Dec is implemented by a circuit that does only depend on the security parameter λ. To achieve this notion of FHE, all these schemes are based on the so-called *bootstrapping* technique by Gentry [20], which we will recall in the next section.

We stress, however, that there is the relaxed notion of *Leveled FHE* that we want to deal with first. Unlike "pure" FHE schemes (as in Definition 4), such schemes can correctly evaluate circuits up to a certain depth only. We will recall this notion in the next section. For such leveled FHE schemes, we remark that except for the scheme by Brakerski, Gentry, and Vaikuntanathan [7], again all existing schemes are based on Gentry's bootstrapping technique.[3] In this paper, we restrict our attention to (leveled) FHE schemes that are based on Gentry's bootstrapping technique.

Our aim is a characterization of the IND-CPA security of all existing (leveled) FHE schemes that are based on the technique of bootstrapping. To do so, we first give a brief summary on Gentry's bootstrapping approach in the next section and prove an IND-CPA characterization of schemes that can be constructed in this way. We will do this both for leveled FHE schemes, as well as for "pure" FHE schemes. For the latter, we need the FHE schemes to have the additional property of *circuit privacy* that we will recall in Section 4.2. Finally, in Section 4.3 we discuss existing schemes, while focusing on a particular scheme by van Dijk et al. [15] for a better conceptual understanding.

4.1 Gentry's Bootstrapping Technique: Leveled FHE Schemes

In this section, we briefly want to recall Gentry's bootstrapping technique [20] on how to construct FHE schemes. Roughly speaking, Gentry constructs a homomorphic encryption scheme for circuits of any depth from an underlying encryption scheme that is homomorphic for "just a little more than its own decryption circuit". We formalize the term in double quotes momentarily (see also [20, Definition 4]), but first need to do some more definitional work. We will prove later that characterizing the IND-CPA security of the underlying schemes is already enough to characterize the IND-CPA security of resulting schemes that are homomorphic for *all circuits up to a certain depth* (such schemes are also known as *Leveled FHE schemes*).

Definition 5. *Let* $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ *be an encryption scheme in which* Dec *is implemented by a circuit that does only depend on the security parameter* λ*. For every output* $(\mathsf{pk}, \mathsf{sk})$ *of* $\mathsf{KeyGen}(\lambda)$*, we let* Γ *be a set of gates with inputs and output in plaintext space* $\mathcal{P}$ *including the identity gate (input and output are the same). For gate* $g \in \Gamma$*, the* g-augmented decryption circuit *consists of a* g*-gate connecting multiple copies of* Dec *(the number of copies equals the number of inputs to* g*), where* Dec *takes the secret key* sk *and a ciphertext as input formatted as elements of* $\mathcal{P}^{\ell(\lambda)}$*, where* $\ell(\lambda)$ *is some polynomial in* λ*. We denote the set of all* g*-augmented decryption circuits,* $g \in \Gamma$*, by* $\mathsf{Dec}(\Gamma)$*.*[4]

[3] Some of the existing schemes, however, deviate from Gentry's original blueprint where one starts with a "somewhat homomorphic scheme", then "squashes" the decryption circuit, and then does the bootstrapping. In the present work, we are not interested in the "squashing" step and will restrict our attention to the bootstrapping step.

[4] Recall that Dec always depends on λ and we sometimes write $\mathsf{Dec}[\lambda]$ to make this dependency obvious.

Recall that from now on, we restrict our attention to encryption schemes in which decryption Dec is implemented by a circuit that does only depend on the security parameter λ. The most important property of an encryption scheme to be of any use in Gentry's approach is that of *bootstrappability*.

Definition 6. *Let $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ be a homomorphic encryption scheme for a set of circuits $\mathbb{C} = \mathbb{C}[\lambda]$. $\mathcal{E}$ is called* bootstrappable for a set of gates Γ, *if $\mathsf{Dec}[\lambda](\Gamma) \subseteq \mathbb{C}[\lambda]$ for all security parameters λ.*

There are two main results in [20] that are of particular interest to us:

Theorem 2 (see Theorem 3 of [20]). *There is an efficient and explicit transformation that for any given bootstrappable scheme $\mathcal{E}$ for a set of gates Γ and parameter $d = d(\lambda)$ outputs another encryption scheme $\mathcal{E}^{(d)}$ that is*

1. *compact and whose decryption circuit is identical to that of $\mathcal{E}$*
2. *homomorphic for all circuits with gates in Γ of depth at most d.*

Theorem 3 (see Theorem 4 of [20]). *Let $\mathcal{E}$ be a bootstrappable scheme for a set of gates Γ. For all parameters $d = d(\lambda)$, we have that the output $\mathcal{E}^{(d)}$ of the transformation from Theorem 2 applied to $\mathcal{E}$ and d is* IND-CPA *secure if $\mathcal{E}$ is.*

We will now prove that the IND-CPA security of $\mathcal{E}^{(d)}$ is actually *equivalent* to that of $\mathcal{E}$. For this we need to recall a few details in Gentry's transformation of Theorem 2. For all remaining details, we refer to [20]. The particular facts, we will need about $\mathcal{E}^{(d)}$ are the following three (cf. [20]):

1. The plaintext space $\mathcal{P}$ of $\mathcal{E}^{(d)}$ is the same as that of $\mathcal{E}$.
2. The key generation algorithm of $\mathcal{E}^{(d)}$ uses the key generator KeyGen of $\mathcal{E}$ $(d+1)$-times to produce $d+1$ public and secret key pairs $(\mathsf{pk}_i, \mathsf{sk}_i)$, $i = 0, \ldots, d$. Let $\mathsf{sk}_{i1}, \ldots, \mathsf{sk}_{i\ell}$ be the representation of sk_i as elements of $\mathcal{P}$ with $\ell = \ell(\lambda)$ as in Definition 5. The key generator of $\mathcal{E}^{(d)}$ then computes $\overline{\mathsf{sk}_{ij}} \longleftarrow \mathsf{Enc}_{\mathsf{pk}_{i-1}}(sk_{ij})$ for $i = 1, \ldots, d$ and $j = 1, \ldots, \ell$, and outputs the secret key $\mathsf{sk}^{(d)} := sk_0$, and public key
$$\mathsf{pk}^{(d)} := \left((\mathsf{pk}_i)_{i=1,\ldots,d}, (\overline{\mathsf{sk}_{ij}})_{\substack{i=1,\ldots,d \\ j=1,\ldots,\ell}} \right).$$
3. Encryption of a message $m \in \mathcal{P}$ in $\mathcal{E}^{(d)}$ is done by computing a ciphertext $c \longleftarrow \mathsf{Enc}_{\mathsf{pk}_d}(m)$, i.e., an encryption of m under pk_d by using the encryption algorithm Enc of $\mathcal{E}$.

We are now in a position to prove the IND-CPA characterization.

Theorem 4. *Let $\mathcal{E}$ be a bootstrappable scheme for a set of gates Γ. For parameter $d = d(\lambda)$, let $\mathcal{E}^{(d)}$ denote the output of the transformation from Theorem 2 applied to $\mathcal{E}$ and d. For all parameters d, it holds:*

$$\mathcal{E}^{(d)} \textit{ is } \mathsf{IND\text{-}CPA} \textit{ secure} \iff \mathcal{E} \textit{ is } \mathsf{IND\text{-}CPA} \textit{ secure.}$$

Proof. "⇐": This is Theorem 3.
"⇒": If $\mathcal{A}$ is a PPT adversary that successfully breaks the IND-CPA security of $\mathcal{E}$, then $\mathcal{A}$ can also be used to break the IND-CPA security of $\mathcal{E}^{(d)}$. By looking at the facts above, we know that in the IND-CPA security game for $\mathcal{E}^{(d)}$, $\mathcal{A}$ receives the public key pk_d, outputs two messages $m_0, m_1 \in \mathcal{P}$ and gets the ciphertext $c \longleftarrow \mathsf{Enc}_{\mathsf{pk}_d}(m_b)$ as the challenge ciphertext, where $b \xleftarrow{U} \{0,1\}$. Due to the initial assumption on $\mathcal{A}$, $\mathcal{A}$ can guess the bit b with non-negligible advantage. □

Unfortunately, the resulting scheme from Theorem 2 after applying the transformation is not yet an FHE scheme as it is only homomorphic for all circuits with gates in Γ of depth at most d (i.e., it is leveled fully homomorphic). However, in [19], Gentry shows how to modify the previously described technique to get "pure" FHE schemes. We will give an IND-CPA security characterization of such schemes (with a certain additional property) in the next section.

4.2 Gentry's Bootstrapping Technique: FHE Schemes

In [19, Section 4.3], Gentry shows that, by changing the transformation as described in the following and by *assuming* that the underlying bootstrappable scheme is *circular secure* (a notion we will recall momentarily), the resulting scheme is indeed fully homomorphic and IND-CPA secure. We start by explaining the modification of the transformation of Theorem 2, whereas we denote the resulting scheme by $\mathcal{E}^*$:

In the key generation step above (this is step 2 right after Theorem 3), $\mathcal{E}^*$ uses the key generator KeyGen of $\mathcal{E}$ only once (instead of $(d+1)$-times) to compute a key pair $(\mathsf{pk}, \mathsf{sk})$ and outputs the secret key $\mathsf{sk}^* := \mathsf{sk}$ and public key $\mathsf{pk}^* := (\mathsf{pk}, \overline{\mathsf{sk}_1}, \ldots, \overline{\mathsf{sk}_\ell})$ where $\overline{\mathsf{sk}_i} \longleftarrow \mathsf{Enc}_{\mathsf{pk}}(\mathsf{sk}_i)$ and $\mathsf{sk}_1, \ldots, \mathsf{sk}_\ell$ is the representation of sk as elements of $\mathcal{P}$. This is the only modification and the rest works exactly as in the transformation of Theorem 2 (see [19, Section 4.3] for details).

Next, we recall the notion of *circular security* for bootstrappable encryption schemes $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$. Consider the following experiment for a given algorithm $\mathcal{A}$ and parameter λ:

Experiment $\mathbf{Exp}^{\mathsf{circular}}_{\mathcal{A},\mathsf{KeyGen}}(\lambda)$:

1. Compute $(\mathsf{pk}, \mathsf{sk}) \longleftarrow \mathsf{KeyGen}(\lambda)$
2. Choose $b \xleftarrow{U} \{0,1\}$. If $b = 0$, then compute $\overline{\mathsf{sk}_j} \longleftarrow \mathsf{Enc}_{\mathsf{pk}}(\mathsf{sk}_j)$ for all $j = 1, \ldots, \ell$ where $\mathsf{sk}_1, \ldots, \mathsf{sk}_\ell$ is the representation of sk as elements of $\mathcal{P}$ with $\ell = \ell(\lambda)$ as in Definition 5. If $b = 1$, then compute $\overline{\mathsf{sk}_j}$ as encryptions of some fixed element $\mathbf{0} \in \mathcal{P}$, unrelated to pk, for all $j = 1, \ldots, \ell$
3. $d \longleftarrow \mathcal{A}\left(\mathsf{pk}, \overline{\mathsf{sk}_1}, \ldots, \overline{\mathsf{sk}_\ell}\right)$ where $d \in \{0,1\}$
4. The output of the experiment is defined to be 1 if $d = b$ and 0 else.

This experiment defines *circular security* for bootstrappable encryption schemes $\mathcal{E}$. We note that, as we consider bootstrappable schemes, this definition is equivalent to the "standard" definition of circular security [6] as originally introduced (this is shown in [19, Chapter 4]).

Before we can state the main result of this section, we need to recall another notion that is related to FHE, namely that of *circuit privacy*. Informally, this notion says that even if the secret key is known, the output of Eval does not reveal any information about the circuit that it evaluates, except for the output value of that circuit. Formally, this idea is captured in the following definition:

Definition 7. *An FHE scheme* $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ *is said to be* (computationally) circuit-private*, if for every keypair* $(\mathsf{pk}, \mathsf{sk}) \longleftarrow \mathsf{KeyGen}(\lambda)$*, any circuit* C*, and any fixed tuple of fresh encryptions* $(c_1, \ldots, c_r)$ *with* $c_i \longleftarrow \mathsf{Enc}_{\mathsf{pk}}(m_i)$ *for plaintexts* $m_i \in \mathcal{P}$ *and* $i = 1, \ldots, r$*, the following distributions (over the random coins in* Enc *and* Eval*) are (computationally) indistinguishable:*

$$\mathsf{Enc}_{\mathsf{pk}}(C(m_1, \ldots, m_r)) \approx_c \mathsf{Eval}_{\mathsf{pk}}(C, c_1, \ldots, c_r).$$

Finally, we can formulate the main result:

Theorem 5. *Let* $\mathcal{E} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ *be a bootstrappable scheme for a* universal[5] *set of gates* Γ*. If the resulting scheme* $\mathcal{E}^*$ *is circuit-private, it holds that*

$$\mathcal{E}^* \textit{ is } \mathsf{IND\text{-}CPA} \textit{ secure} \iff \mathcal{E} \textit{ is circular secure.}$$

Proof (Sketch). "$\Leftarrow$": This is shown in [19, Theorem 4.3.2].
"$\Rightarrow$": We assume that $\mathcal{E}$ is not circular secure, i.e., there exists a PPT algorithm $\mathcal{A}^{\text{circular}}$ that breaks the security of $\mathcal{E}$ with non-negligible advantage $f(\lambda)$. We derive a contradiction by constructing a PPT algorithm $\mathcal{A}^{\text{cpa}} = (\mathcal{A}_1^{\text{cpa}}, \mathcal{A}_2^{\text{cpa}})$ that successfully breaks the $\mathsf{IND\text{-}CPA}$ security of $\mathcal{E}^*$ with advantage $f(\lambda)$.

First, the adversary $\mathcal{A}_1^{\text{cpa}}$ receives the public key $\mathsf{pk}^* = (\mathsf{pk}, \overline{\mathsf{sk}_1}, \ldots, \overline{\mathsf{sk}_\ell})$ where $\overline{\mathsf{sk}_i} \longleftarrow \mathsf{Enc}_{\mathsf{pk}}(\mathsf{sk}_i)$ and $\mathsf{sk}_1, \ldots, \mathsf{sk}_\ell$ is the representation of the secret key sk as elements of $\mathcal{P}$. Then, $\mathcal{A}_1^{\text{cpa}}$ chooses messages $\mathbf{0} \neq m_0 \in \mathcal{P}$ and $m_1 := \mathbf{0}$ together with circuits C_i such that $C_i(m_0, \mathsf{sk}_i) = \mathsf{sk}_i$ and $C_i(m_1, \mathsf{sk}_i) = m_1$ for all $i = 1, \ldots, \ell$. For instance, if we consider all boolean circuits and assume that $\mathcal{P} = \{0, 1\}$, $\mathcal{A}_1^{\text{cpa}}$ could simply choose $m_0 = 1, m_1 = 0$ and C_i as a single AND-gate for all $i = 1, \ldots, \ell$. Now, the $\mathsf{IND\text{-}CPA}$ challenger chooses a random bit $b \overset{U}{\longleftarrow} \{0, 1\}$ and sends the challenge $c \longleftarrow \mathsf{Enc}_{\mathsf{pk}}(m_b)$ to $\mathcal{A}_2^{\text{cpa}}$. Since $\mathcal{E}^*$ is fully homomorphic, $\mathcal{A}_2^{\text{cpa}}$ can compute $\overline{\sigma_i} \longleftarrow \mathsf{Eval}_{\mathsf{pk}}(C_i, c, \overline{\mathsf{sk}_i})$ for all $i = 1, \ldots, \ell$. Due to the correctness condition on $\mathcal{E}^*$, this means for all $i = 1, \ldots, \ell$:

$$\sigma_i := \mathsf{Dec}_{\mathsf{sk}}(\overline{\sigma_i}) = C_i(m_b, \mathsf{sk}_i). \tag{1}$$

Next, $\mathcal{A}_2^{\text{cpa}}$ sends $(\mathsf{pk}, \overline{\sigma_1}, \ldots, \overline{\sigma_\ell})$ to $\mathcal{A}^{\text{circular}}$ that returns a bit $d \in \{0, 1\}$ which in turn is the output b' of $\mathcal{A}_2^{\text{cpa}}$, i.e., $b' = d$.

We have the following relations: If $b = 0$, then $\overline{\sigma_i}$ is computationally indistinguishable (since $\mathcal{E}^*$ was assumed to be circuit-private) from a fresh encryption of sk_i, meaning in particular that $\sigma_i = \mathsf{sk}_i$ for all $i = 1, \ldots, \ell$ due to equation (1). Hence, $\mathcal{A}^{\text{circular}}$ makes the right guess on b with advantage $f(\lambda)$, i.e.,

[5] This is a set of gates by which any circuit can be expressed, e.g., a NAND-gate when considering boolean circuits.

$\Pr[b' = b|b = 0] \geq \frac{1}{2} + f(\lambda)$. If $b = 1$, then $\overline{\sigma_i}$ is computationally indistinguishable from a fresh encryption of $\mathbf{0}$, unrelated to pk, for all $i = 1, \ldots, \ell$. Hence, $\mathcal{A}^{\text{circular}}$ again guesses b with advantage $f(\lambda)$, i.e., $\Pr[b' = b|b = 1] \geq \frac{1}{2} + f(\lambda)$. We have shown:

$$\begin{aligned}\Pr[\mathbf{Exp}^{\text{ind-cpa}}_{\mathcal{A}^{\text{cpa}},\mathsf{Gen}}(\lambda) = 1] &= \sum_{\beta \in \{0,1\}} \Pr[b' = b|b = \beta] \cdot \Pr[b = \beta] \\ &\geq \frac{1}{2} \cdot (1 + 2f(\lambda)) = \frac{1}{2} + f(\lambda).\end{aligned}$$

□

Some remarks on this result are in order:

Remark 2. 1. We would like to stress that Theorem 5 actually holds in a more general context as well. Looking at the proof, one notices that there is no need for $\mathcal{E}^*$ to be circuit-private for all circuits. The circuit privacy is only needed for the special circuits C_i used in the proof. In particular, in the case when only boolean circuits are considered and the plaintext space is $\mathcal{P} = \{0, 1\}$, the circuits C_i are all the same, namely an AND-gate. It is easy to see that all existing FHE schemes that work on the plaintext space $\mathcal{P} = \{0, 1\}$ are circuit-private for a single AND-gate (see also Section 4.3).

2. In Theorem 1, we showed a characterization of the IND-CPA security of shift-type homomorphic encryption schemes. All currently existing FHE schemes rely on the assumption that the IND-CPA security of the underlying scheme already implies its circular security – meaning that for these schemes the two notions of circular security and IND-CPA security are equivalent. So under this assumption, Theorem 5 together with Theorem 1 yield an IND-CPA characterization of *all existing* circuit-private FHE schemes that are based on Gentry's bootstrapping technique.
3. Theorem 5 together with the first item of this remark tell us that the circular security of the underlyling bootstrappable scheme is not only sufficient but also necessary in order to get an FHE scheme. It therefore underlines the importance of Brakerski et al.'s work [9] which actually has the bigger goal of achieving circular secure bootstrappable encryption, instead of only achieving circular security for somewhat homomorphic encryption schemes (cf. Section 1.1).

4.3 Gentry's Bootstrapping Technique: The Existing Schemes

In total, there currently exist 13 FHE schemes that are all based on Gentry's bootstrapping technique (at least concerning the resulting "pure" FHE schemes), namely [7–9, 15, 19–24, 27, 29, 30]. Their underlying schemes are all shift-type homomorphic. This is due to the fact that the concept of shift-type homomorphic encryption is very similar to that of "adding noise" (see Remark 1), which itself is a concept employed in all existing schemes. For the more recently developed schemes [7–9, 21, 23, 24] the shift-type structure of the encryption algorithm can

be immediately seen. A good summary of Gentry's original scheme [20] is given in [30, Section 3.1]. Therein, Gentry's scheme and the variants [19] and [30] are presented in a way such that the shift-type structure is easily seen. Concerning the variants [22, 27, 29], a summary is given in [27, Section 3], again presented in a fashion such that the shift-type structure of the encryption is immediately noticeable.

We will recall (only very briefly due to lack of space) the remaining variant by van Dijk et al. [15] to show that it is shift-type homomorphic. To get rid of a very voluminous and confusing introduction of parameters, we will fix a particular setup of parameters in the key generation phase and note that all of the following can be done in a more general fashion (see [15]). Also, we will focus here on the encryption algorithm only and refer the reader to [15] for details on the remaining algorithms for decryption and evaluation. For the security parameter λ, we fix:

$$\rho := \lambda, \rho' := 2\lambda, \eta \in \rho' \cdot \Theta(\lambda \log^2 \lambda), \gamma \in \omega(\eta^2 \log \lambda) \text{ and } \tau := \gamma + \lambda.$$

The secret key sk of the scheme is $p \xleftarrow{U} (2\mathbb{Z}+1) \cap [2^{\eta-1}, 2^{\eta})$ and we define the following efficiently sampleable distribution

$$\mathcal{D}_{\gamma,\rho}(p) := \left\{ x = pq + r \mid q \xleftarrow{U} \mathbb{Z} \cap [0, 2^{\gamma}/p), r \xleftarrow{U} \mathbb{Z} \cap (-2^{\rho}, 2^{\rho}) \right\}.$$

With this notation, we let $\mathsf{pk} = (x_0, \ldots, x_\tau)$ be the public key with $x_i \xleftarrow{U} \mathcal{D}_{\gamma,\rho}(p)$ for all $i = 0, \ldots, \tau$ whereas the x_i's are relabeled such that x_0 is the largest (if x_0 is even or $x_0 \bmod p$ is odd, then restart). The plaintext space is $\{0,1\}$.

The encryption algorithm takes the public key pk and a plaintext $m \in \{0,1\}$ as input and outputs a ciphertext $c := [(m + 2r + 2\sum_{i \in S} x_i) \bmod x_0]$ whereas S is a random subset of $\{1, \ldots, \tau\}$ and $r \xleftarrow{U} \mathbb{Z} \cap (-2^{\rho'}, 2^{\rho'})$. In the notation of the shift-type homomorphic definition (see Definition 2), we have that $\widehat{\mathcal{C}}$ is the ring $\mathbb{Z}_{x_0}$ and

$$\mathcal{N} = \left\{ 2(r + \sum_{i \in S} x_i) \bmod x_0 \mid r \in \mathbb{Z} \cap (-2^{\rho'}, 2^{\rho'}), S \subseteq \{1, \ldots, \tau\} \right\}.$$

The injective homomorphism φ is given by $m \mapsto m \bmod x_0$, which is even a ring homomorphism. Encryption is then given by $\varphi(m) + n$ where $n \in \mathcal{N}$. Concerning the homomorphic property in Definition 2, we need to make more effort:

It is shown in [15, Lemma 3.3] that the scheme is homomorphic for Boolean circuits with the property that for any $\alpha \geq 1$ and any set of integer inputs all less than $2^{\alpha(\rho'+2)}$ in absolute value, it must hold that the output of the *generalized circuit* (same circuit where the ADD- and MULT-gates are applied to integers instead of bits) has absolute value at most $2^{\alpha(\eta-4)}$. Furthermore, it is shown in [15, Lemma 3.5] that if $f(x_1, \ldots, x_t)$ is the multivariate polynomial of degree d computed by the generalized circuit of a given boolean circuit C with

t inputs, then the scheme is homomorphic for C if $|\bar{f}| \cdot (2^{\rho'+2})^d \leq 2^{\eta-4}$, where $|\bar{f}|$ is the l_1 norm of the coefficient vector of f. In respect of the homomorphic property of Definition 2, it suffices to show that the scheme is homomorphic for the boolean circuit C_{ADD} that consists of a single ADD-gate only. Clearly, the multivariate polynomial that is computed by the generalized circuit of C_{ADD} is $f(x_1, x_2) = x_1 + x_2$ and has degree $d = 1$ with $|\bar{f}| = 2$. Therefore, the scheme is homomorphic for this circuit if we have

$$2^{\rho'+3} \leq 2^{\eta-4}, \text{ which in turn is fulfilled if } \eta \geq \rho' + 7.$$

This final condition holds as $\eta \in \rho' \cdot \Theta(\lambda \log^2 \lambda)$. In total we have shown that the above scheme indeed is shift-type homomorphic.

5 Conclusion

With the identification of shift-type encryption as the most basic structure that all existing homomorphic encryption schemes have in common, we were able to deduce IND-CPA characterizations of all existing bootstrapping-based leveled FHE schemes. This result supports an easier design of such schemes, since new candidates can immediately be checked for IND-CPA security by looking at the corresponding subset membership problem that comes out of our characterization. In regard to [3], it is interesting to see that all existing group homomorphic encryption schemes and the more general homomorphic schemes (in particular, the existing FHE schemes) share the same shift-type structure. Further research in this direction could implicate that a given homomorphic scheme has to have this shift-type structure in order to be IND-CPA secure. We leave this as an open question.

Our result that the IND-CPA security of bootstrapping-based FHE schemes that offer a "minimal" type of circuit privacy is *equivalent* to the circular security of the underlying bootstrappable scheme shows: If we want to construct such IND-CPA secure FHE schemes, we are bound to the design of circular secure bootstrappable schemes. We hope that this fact stimulates the research community to devote even more effort to proving existing schemes circular secure and/or finding a radically new approach to FHE that is not based on Gentry's bootstrapping technique.

References

1. Melchor, C.A., Gaborit, P., Herranz, J.: Additively Homomorphic Encryption with d-Operand Multiplications. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 138–154. Springer, Heidelberg (2010)
2. Applebaum, B.: Key-Dependent Message Security: Generic Amplification and Completeness. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 527–546. Springer, Heidelberg (2011)

3. Armknecht, F., Katzenbeisser, S., Peter, A.: Group homomorphic encryption: Characterizations, impossibility results, and applications. Designs, Codes and Cryptography, 1–24, 10.1007/s10623-011-9601-2, http://dx.doi.org/10.1007/s10623-011-9601-2
4. Barak, B., Haitner, I., Hofheinz, D., Ishai, Y.: Bounded Key-Dependent Message Security. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 423–444. Springer, Heidelberg (2010)
5. Benaloh, J.: Verifiable secret-ballot elections. Ph.D. thesis, Yale University (1987)
6. Boneh, D., Halevi, S., Hamburg, M., Ostrovsky, R.: Circular-Secure Encryption from Decision Diffie-Hellman. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 108–125. Springer, Heidelberg (2008)
7. Brakerski, Z., Gentry, C., Vaikuntanathan, V.: (leveled) fully homomorphic encryption without bootstrapping. In: ITCS, pp. 309–325. ACM (2012)
8. Brakerski, Z., Vaikuntanathan, V.: Efficient fully homomorphic encryption from (standard) LWE. In: FOCS, pp. 97–106. IEEE (2011)
9. Brakerski, Z., Vaikuntanathan, V.: Fully Homomorphic Encryption from Ring-LWE and Security for Key Dependent Messages. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 505–524. Springer, Heidelberg (2011)
10. Cohen, J.D., Fischer, M.J.: A robust and verifiable cryptographically secure election scheme (extended abstract). In: FOCS, pp. 372–382. IEEE (1985)
11. Cramer, R., Damgård, I., Nielsen, J.B.: Multiparty Computation from Threshold Homomorphic Encryption. In: Pfitzmann, B. (ed.) EUROCRYPT 2001. LNCS, vol. 2045, pp. 280–299. Springer, Heidelberg (2001)
12. Cramer, R., Franklin, M.K., Schoenmakers, B., Yung, M.: Multi-authority Secret-Ballot Elections with Linear Work. In: Maurer, U.M. (ed.) EUROCRYPT 1996. LNCS, vol. 1070, pp. 72–83. Springer, Heidelberg (1996)
13. Cramer, R., Gennaro, R., Schoenmakers, B.: A Secure and Optimally Efficient Multi-authority Election Scheme. In: Fumy, W. (ed.) EUROCRYPT 1997. LNCS, vol. 1233, pp. 103–118. Springer, Heidelberg (1997)
14. Cramer, R., Shoup, V.: Universal Hash Proofs and a Paradigm for Adaptive Chosen Ciphertext Secure Public-Key Encryption. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 45–64. Springer, Heidelberg (2002)
15. van Dijk, M., Gentry, C., Halevi, S., Vaikuntanathan, V.: Fully Homomorphic Encryption over the Integers. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 24–43. Springer, Heidelberg (2010)
16. El Gamal, T.: A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. In: Blakely, G.R., Chaum, D. (eds.) CRYPTO 1984. LNCS, vol. 196, pp. 10–18. Springer, Heidelberg (1985)
17. Fontaine, C., Galand, F.: A survey of homomorphic encryption for nonspecialists. EURASIP J. Inf. Secur., 15:1–15:15 (January 2007), http://dx.doi.org/10.1155/2007/13801
18. Gennaro, R., Gentry, C., Parno, B.: Non-interactive Verifiable Computing: Outsourcing Computation to Untrusted Workers. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 465–482. Springer, Heidelberg (2010)
19. Gentry, C.: A fully homomorphic encryption scheme. Ph.D. thesis, Stanford University (2009)
20. Gentry, C.: Fully homomorphic encryption using ideal lattices. In: STOC, pp. 169–178. ACM (2009)
21. Gentry, C., Halevi, S.: Fully homomorphic encryption without squashing using depth-3 arithmetic circuits. In: FOCS, pp. 107–109. IEEE (2011)

22. Gentry, C., Halevi, S.: Implementing Gentry's Fully-Homomorphic Encryption Scheme. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 129–148. Springer, Heidelberg (2011)
23. Gentry, C., Halevi, S., Smart, N.P.: Fully Homomorphic Encryption with Polylog Overhead. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 465–482. Springer, Heidelberg (2012)
24. Gentry, C., Halevi, S., Smart, N.P.: Better bootstrapping in fully homomorphic encryption. Cryptology ePrint Archive, Report 2011/680 (2011)
25. Ishai, Y., Paskin, A.: Evaluating Branching Programs on Encrypted Data. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 575–594. Springer, Heidelberg (2007)
26. Kushilevitz, E., Ostrovsky, R.: Replication is not needed: Single database, computationally-private information retrieval. In: FOCS, pp. 364–373 (1997)
27. Loftus, J., May, A., Smart, N.P., Vercauteren, F.: On CCA-Secure Somewhat Homomorphic Encryption. In: Miri, A., Vaudenay, S. (eds.) SAC 2011. LNCS, vol. 7118, pp. 55–72. Springer, Heidelberg (2012)
28. Naor, M., Pinkas, B.: Oblivious polynomial evaluation. SIAM J. Comput. 35(5), 1254–1281 (2006)
29. Smart, N.P., Vercauteren, F.: Fully Homomorphic Encryption with Relatively Small Key and Ciphertext Sizes. In: Nguyen, P.Q., Pointcheval, D. (eds.) PKC 2010. LNCS, vol. 6056, pp. 420–443. Springer, Heidelberg (2010)
30. Stehlé, D., Steinfeld, R.: Faster Fully Homomorphic Encryption. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 377–394. Springer, Heidelberg (2010)