A Simple Framework for Noise-Free Construction of Fully Homomorphic Encryption from a Special Class of Non-Commutative Groups

Koji Nuida

Research Institute for Secure Systems (RISEC), National Institute of Advanced Industrial Science and Technology (AIST)
k.nuida@aist.go.jp

Abstract

We propose a new and simple framework for constructing fully homomorphic encryption (FHE) which is completely different from the previous work. We use finite non-commutative (a.k.a., non-abelian) groups which are "highly non-commutative" (e.g., the special linear groups of size two) as the underlying structure. We show that, on such groups, the AND and NOT operations on plaintext bits (which are sufficient to realize an arbitrary operation by composing them) can be emulated by a "randomized commutator" (which essentially requires the non-commutativity) and division operations on ciphertext elements, respectively. Then we aim at concealing the "core structure" of ciphertexts by taking conjugation by a secret element (where the non-commutativity is again essential), rather than adding noise to the ciphertext as in the previous FHE schemes. The "noise-freeness" of our framework yields the fully-homomorphic property directly, without the bootstrapping technique used in the previous schemes to remove the noise amplified by the homomorphic operations. This makes the overall structure of the FHE schemes significantly simpler and easier to understand. Although a secure instantiation based on the framework has not been found, we hope that the proposed framework itself is of theoretical value, and that the framework is flexible enough to allow a secure instantiation in the future.

1 Introduction

Until the pioneering work by Gentry [11], construction of *fully homomorphic* (public key) *encryption* (*FHE*) that enables, without revealing any information on encrypted plaintexts, anyone to perform arbitrary operations on the plaintexts through the corresponding "homomorphic operations" on the ciphertexts had been a long-standing open problem. Due to the theoretical and practical high importance, after his first construction of FHE, many research have been done to improve the efficiency and security ([12, 13, 15, 23]), or to give variants of the scheme, possibly under more standard assumptions ([3, 4, 5, 6, 7, 8, 9, 10, 14, 16, 19, 22]; see [21] for a survey). Study of FHE is currently one of the main topics in the research area of cryptography.

To the author's best knowledge, all the known constructions of FHE schemes[1] including the Gentry's original one relied on the bootstrapping technique. Namely, these are two-step construction, where the preliminary scheme uses ciphertexts with noise and has a limitation of the number of successive homomorphic operations caused by the amplification of noise, and then the bootstrapping operation is applied to cancel the noise on the ciphertexts for overcoming the

[1] Here we exclude the "trivial" construction of FHE mentioned in Section 2.1 of [17], where a ciphertext after homomorphic circuit evaluation involves the evaluated circuit itself as well as all the original ciphertexts, since the ciphertext size in the scheme grows rapidly by homomorphic operations and it is not practical at all.

limitation. The bootstrapping is certainly an outstanding idea, but its theoretical and practical treatment is complicated and it makes the overall structure of the FHE scheme not easy to overlook. It is surmised that such a difficulty caused by the bootstrapping would be a hurdle for future practical implementation of FHE schemes. To resolve the problem, a new approach to construct FHE schemes that does not require the bootstrapping operation is really valuable.

1.1 Our contributions

The contribution of the present work is to propose an approach to achieve the fully-homomorphic functionality for public key encryption, completely different from the previous work. Based on our approach, the fully-homomorphic functionality is realized without the bootstrapping.

The main technical difference from the previous work is that, in the proposed framework, we use *non-commutative* (a.k.a., *non-abelian*) (finite) *groups* as the underlying mathematical structure. Roughly speaking, the groups should be "highly non-commutative" in order to ensure that the homomorphic AND operation works correctly; see below for details. The main advantages of using non-commutative groups are as follows:

- In order to increase the security, the well-formed shape of the ciphertext is scrambled by taking a conjugate by a secret element of the group, rather than by adding a noise to the ciphertext as in the previous FHE schemes; the "noise-free" construction avoids the use of bootstrapping. Here we emphasize that taking a conjugate is useless in commutative groups, therefore the non-commutativity is essential in our framework for FHE.
- In our construction, the AND and NOT operations on plaintext bits are emulated by using "randomized commutator" and division operations on the non-commutative group, respectively. We emphasize that the commutator operation provides no information when the underlying group is commutative, hence the non-commutativity is again essential.

Here we explain the idea to emulate the AND and NOT bit operations on a non-commutative group to achieve the fully-homomorphic functionality. A possible strategy to conceal the core structure for security purpose will be explained below.

How to emulate AND operation. Our homomorphic AND operation on a group G is established by using the *commutator* on G, which is an operation defined as follows:

$$[g, h] := g \cdot h \cdot g^{-1} \cdot h^{-1} \text{ for any } g, h \in G \ . \tag{1}$$

Now we can see that the value of the commutator $[g, h]$ becomes an identity element $1 \in G$ if either of the two inputs g, h is 1. The starting point of the idea to our homomorphic AND operation is the following observation: The above-mentioned property $[1, h] = [g, 1] = 1$ looks similar to the property $0 \wedge b_2 = b_1 \wedge 0 = 0$ of the AND operation $\wedge$ for bits[2]. This similarity suggests to associate the identity element $1 \in G$ to plaintext bit 0. Then it is naively expected that it is effective to associate the other elements of G to plaintext bit 1.

However, the naive correspondence above does not work; if the two inputs for the commutator are identical (and not 1), then the property $[g, g] = g \cdot g \cdot g^{-1} \cdot g^{-1} = 1$ of the commutator (which means "$1 \wedge 1 = 0$" under the correspondence) is not consistent to the remaining property $1 \wedge 1 = 1$ of the AND operation. We want to realize that the situation above does not happen. For the purpose, we introduce a re-randomization of the input, which makes the two inputs distinct (except negligible probability). Precisely, we modify the commutator operation as follows:

$$[x, y]_R := [g \cdot x \cdot g^{-1}, y] \text{ for any } x, y \in G \ , \tag{2}$$

[2]One may be reminded of the proof of the Barrington's theorem [1] from the property of commutators.

where g is a uniformly random element of G. We note that the property $[x,1]_R = [1,y]_R = 1$ corresponding to the property $b_1 \wedge 0 = 0 \wedge b_2 = 0$ is preserved by the modification. On the other hand, for the property of the randomized commutator corresponding to the property $1 \wedge 1 = 1$ (that is, if $x, y \in G$ are associated to the bit 1, then $[x,y]_R$ is also associated to the bit 1), there does exist a group G for which the property holds. For example, the special linear group $SL_2(\mathbb{F})$ consisting of the 2×2 matrices over a finite field $\mathbb{F}$ (of exponentially large cardinality) with determinant one satisfies the desired property[3]. Hence, by working on such a group, the homomorphic AND operation can be efficiently realized.

How to emulate NOT operation. In order to realize the homomorphic NOT operation on a group G as above, we switch the consideration from a single element of G to a pair of elements of G. The idea is as follows: For $(x,y) \in G \times G$ for which $y \neq 1$ holds[4], we associate the pair (x,y) to the bit 0 and the bit 1 if $x = 1$ and $x = y$, respectively. Then the division operation $x \mapsto x^{-1} \cdot y$ exchanges the two cases; $x^{-1} \cdot y = 1$ (respectively, $x^{-1} \cdot y = y$) if and only if $x = y$ (respectively, $x = 1$). Hence, the mapping

$$(x,y) \mapsto (x^{-1} \cdot y, y) \tag{3}$$

works as the homomorphic NOT operation under the correspondence of pairs of group elements to plaintext bits above. (Intuitively, the main body of the pair is the first component x, while the second component y plays a role of a "template" to generate the output of the homomorphic NOT operation.)

We note that, since we are now working on pairs of elements instead of single elements, the homomorphic AND operation explained above has to be applied to both of the two components, which should be "synchronized". Namely, the new homomorphic AND operation is given by

$$(x_1,y_1),(x_2,y_2) \mapsto ([g \cdot x_1 \cdot g^{-1}, y_1], [g \cdot x_2 \cdot g^{-1}, y_2]) \tag{4}$$

where $g \in G$ is chosen uniformly at random. Then it is shown that the operation is again consistent to the properties $0 \wedge b_2 = b_1 \wedge 0 = 0$ and $1 \wedge 1 = 1$ via the correspondence above.

Towards secure realization. Although the homomorphic AND and NOT operations can be realized on a non-commutative group G (with certain desirable property mentioned above) by the ideas explained above, the argument above concerned the fully-homomorphic functionality only and did not concern the security property. Indeed, it is trivially easy to detect to which plaintext bit a given pair (x,y) of group elements is associated, by just checking whether $x = 1$ or not. To resolve the problem, a naive hope is that we can make the adversary not able to know whether a given group element is the identity element or not, while keeping anyone being able to compute the group operations.

A strategy which we consider here is as follows. We want to construct a surjective group homomorphism $\varphi\colon \overline{G} \to G$ from an auxiliary group $\overline{G}$ to the main group G, in such a way that it is difficult (without some trapdoor information) to decide whether a given element of $\overline{G}$ is mapped by φ to the identity element of G or not. Once such a mapping $\varphi\colon \overline{G} \to G$ is obtained, we work on the group $\overline{G}$ instead of G for the encryption and homomorphic operations, while the decryption is performed by computing the value of φ from the trapdoor information. Namely, a ciphertext is now a pair $(\overline{x},\overline{y})$ of elements of $\overline{G}$ with $\varphi(\overline{y}) \neq 1$, and its plaintext is 0 and 1 if $\varphi(\overline{x}) = 1$ and $\varphi(\overline{x}) = \varphi(\overline{y})$, respectively[5]. The homomorphic AND and NOT operations are performed in a similar manner on $\overline{G}$ instead of G. See Section 3 for details.

[3] More precisely, now we associate to the bit 1 the elements of an appropriately specified large subset X of $G \setminus \{1\}$, rather than the elements of $G \setminus \{1\}$; see Definition 3 below.

[4] more precisely, $y \in X$ where X is a subset of $G \setminus \{1\}$ mentioned in the previous footnote

[5] Again, the condition $\varphi(\overline{y}) \neq 1$ is actually $\varphi(\overline{y}) \in X$ in the rigorous argument.

Moreover, a possible way to realize such a trapdoor homomorphism φ is the following: We first define a (not necessarily difficult to compute) homomorphism φ to G from a "well-formed" group $\overline{G}$ embedded in a larger group $\widetilde{G}$, and then the "well-formed" structure of $\overline{G}$ is scrambled[6] by taking a conjugation by a secret random element $T \in \widetilde{G}$. Namely, we are working on the secret subgroup $\{T \cdot g \cdot T^{-1} \mid g \in \overline{G}\}$ of $\widetilde{G}$ instead of $\overline{G}$ itself except for the decryption, and the decryption is performed by first moving back to $\overline{G}$ by using the trapdoor element T and then computing φ. As mentioned above, the strategy here of taking the conjugation relies essentially on the non-commutativity of groups, and the "noise-free" construction yields the fully-homomorphic functionality without the bootstrapping technique.

Unfortunately, the author has not found so far a concrete and secure instantiation of FHE schemes based on our proposed framework (more precisely, a secure instantiation of a trapdoor homomorphism φ, though an instantiation of the main group G achieving the fully-homomorphic functionality has been found as mentioned above). To construct an instantiation of the proposed framework which admits a reliable evidence for security (ideally, a security proof under some standard hardness assumptions) is a future research topic. The author hopes that the proposed new framework for constructing FHE schemes is still worthy by itself and it can promote studies of non-commutative group-based cryptography.

1.2 Related work

Our proposed framework for FHE is simple and uses non-commutative groups, which is completely different from the previous FHE schemes. Below we compare our work to the previous proposals of (non-commutative) group-based cryptographic schemes (see e.g., [2] for a survey).

Some major previous proposals of group-based public key encryption [18, 20] are based on ideas analogous to the Diffie–Hellman key exchange. Hence, though their schemes are implemented on non-commutative groups, these essentially use "internal commutativity" of such groups and the non-commutativity of the groups are used only for hiding the internal commutativity. In contrast, our FHE schemes (in particular, the homomorphic AND operation) essentially use the non-commutativity of the platform groups, since the commutator operator provides no information (i.e., its value is always the identity element) when the underlying group is commutative. The essential dependency on non-commutative structures is a remarkable property of our result among the existing group-based schemes.

We also note that, in those previous Diffie–Hellman-type schemes on non-commutative groups, the platform group is required to have normal forms of elements in order to guarantee that the sender and the receiver will obtain identical symmetric keys during the protocol (namely, even if the two parties obtain the same group *element*, the extracted symmetric keys may be different if the *expressions* of the elements by the two parties are not equal). In contrast, in our FHE scheme, it is *not* necessary that normal forms for elements of the platform groups exist; only the necessary condition from the viewpoint is that the receiver can compute the value of the trapdoor homomorphism φ from an arbitrary expression of a given group element. This non-necessity of normal forms would enlarge the potential candidates of the platform groups for our FHE scheme significantly, compared to the previous group-based schemes.

1.3 Organization of the paper

Section 2 summarizes some notations, terminology and basic notions. In Section 3, we describe our proposed framework for constructing FHE schemes, study its functionality and security, and formalizes the conditions for the underlying groups to achieve the functionality and security.

[6] Regarding the design principle, one may feel a flavor similar to the McEliece cryptosystem and several multivariate quadratic public key cryptosystems.

2 Preliminaries

In this section, we summarize some basic definitions and notations used throughout the paper. In the paper, a group is written in multiplicative form and is not commutative unless otherwise specified. Let 1_G denote the identity element of a group G. For any elements g and h of a group, their *commutator* $[g, h]$ is defined by

$$[g, h] := g \cdot h \cdot g^{-1} \cdot h^{-1} \ . \tag{5}$$

Then $[g, h]$ is the identity element if and only if g and h commute (i.e., $gh = hg$). Let "$a \leftarrow_R X$" express that an element a is chosen from a finite set X uniformly at random. Let λ denote the security parameter unless otherwise specified. We say that a quantity $\varepsilon = \varepsilon(\lambda) \geq 0$ depending on λ is *negligible*, if for any integer $n \geq 1$, there exists a $\lambda_0 > 0$ with the property that we have $\varepsilon(\lambda) < \lambda^{-n}$ for every $\lambda > \lambda_0$; and $\varepsilon \in [0, 1]$ is *overwhelming*, if $1 - \varepsilon$ is negligible.

A *public key encryption* (*PKE*) scheme consists of three algorithms KeyGen, Enc and Dec with the following syntax. The key generation algorithm $\mathsf{KeyGen}(1^\lambda)$ outputs a pair $(\mathsf{pk}, \mathsf{sk})$ of a public key pk and a secret key sk. The encryption algorithm $\mathsf{Enc}(\mathsf{pk}, m)$ with plaintext $m \in \mathcal{M}$, where $\mathcal{M}$ denotes the plaintext space, outputs a ciphertext as the encryption result of m. Finally, the decryption algorithm $\mathsf{Dec}(\mathsf{sk}, c)$ with ciphertext c outputs either a plaintext $m \in \mathcal{M}$ as the decryption result of c, or a distinguished symbol $\perp$ indicating decryption failure. In the paper, we only deal with 1-bit plaintexts; i.e., $\mathcal{M} = \{0, 1\}$. The *correctness* of a PKE scheme means that, for any plaintext $m \in \mathcal{M}$, the probability that $\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, m)) \neq m$ is negligible, where the probability is taken over the randomness in the encryption algorithm. We note that non-zero but negligible decryption error probabilities are tolerated in the paper.

A *homomorphic* PKE scheme is a PKE scheme endowed with another algorithm that, for any map of the form $f \colon \mathcal{M}^n \to \mathcal{M}^{n'}$ chosen from some specified class and any ciphertexts $c_i \leftarrow \mathsf{Enc}(\mathsf{pk}, m_i)$ for plaintexts $m_i \in \mathcal{M}$ ($i \in \{1, 2, \ldots, n\}$), efficiently outputs ciphertexts $c'_1, \ldots, c'_{n'}$ satisfying that $(\mathsf{Dec}(\mathsf{sk}, c'_i))_{i=1}^{n'} = f((m_i)_{i=1}^{n})$ with overwhelming probability. In particular, if the f can be any circuit with polynomially many gates, then the scheme is called an FHE scheme.

The security notion for (homomorphic) PKE schemes considered in the paper is the *CPA security* (also known as the *semantic security*). We note that other advanced security notions for FHE such as the circuit privacy (e.g., [17]) are out of the scope of the paper. We recall the definition of the CPA security as follows, where we suppose $\mathcal{M} = \{0, 1\}$ as mentioned above:

Definition 1 (CPA security)**.** We say that a PKE scheme with plaintext space $\mathcal{M} = \{0, 1\}$ is *CPA-secure*, if for any probabilistic polynomial-time (PPT, in short) adversary $\mathcal{A}$, the *advantage* of $\mathcal{A}$ defined by $\mathsf{Adv}^{\mathrm{CPA}}_{\mathcal{A}}(\lambda) := |Pr[b' = b] - 1/2|$ is negligible, where $Pr[b' = b]$ is the probability that $b' = b$ in the following game (called the *CPA game*):

CPA game for PKE (1-bit plaintext case)
$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$, $b \leftarrow_R \{0, 1\}$, $c_* \leftarrow \mathsf{Enc}(\mathsf{pk}, b)$. Then $\mathcal{A}$ outputs $b' \leftarrow \mathcal{A}(1^\lambda, \mathsf{pk}, c_*)$.

3 Our proposed framework for constructing FHE schemes

In this section, we present our proposed framework for construction of FHE schemes (with 1-bit plaintexts) based on non-commutative groups. In Section 3.1, we formalize some requirements for the underlying groups in our proposed framework to achieve the functionality and the security for the resulting scheme. In Section 3.2, we describe our proposed framework. The functionality and security of the schemes based on our framework are discussed in Section 3.3. Finally, in Section 3.4, we give a sufficient condition for the underlying group to satisfy the requirement for the fully-homomorphic functionality, and present examples of groups satisfying the condition.

3.1 Group samplers

Here we formalize some requirements for the underlying groups. First, we introduce the following notion, which is relevant to the key generation and the correctness of the encryption:

Definition 2 (Group samplers)**.** We say that a polynomial-time algorithm GS is a *group sampler*, if it takes the security parameter as input and it outputs the following objects;

- a finite (non-commutative) group $\widetilde{G}$ and its subgroup $\overline{G}$, for which the multiplication and inverse operations are polynomial-time computable;
- a finite group G and a surjective group homomorphism $\varphi\colon \overline{G} \to G$, with kernel denoted by $H := \{g \in \overline{G} \mid \varphi(g) = 1_G\}$;
- two polynomial-time algorithms $\mathsf{Sample}_{\overline{G}}$ and Sample_H that output a uniformly random element of $\overline{G}$ and a uniformly random element of H, respectively;
- a polynomial-time algorithm Ker that for given input $g \in \overline{G}$, output 0 if $g \in H$ and 1 if $g \notin H$.

Secondly, we introduce a condition for group samplers which is relevant to the correctness for our proposed framework (especially for the homomorphic AND operations):

Definition 3 (Commutator-separability)**.** We say that a group sampler GS is *commutator-separable*, if there exists a subset X of $G \setminus \{1_G\}$ associated to each output of GS satisfying that $|X|/|G|$ is overwhelming and the following condition holds: For any $x_1, x_2 \in X$, we have $[gx_1g^{-1}, x_2] \in X$ with overwhelming probability, where $g \leftarrow_R G$.

We also introduce a computational problem associated to group samplers, which is relevant to the security of the schemes based on our framework. Intuitively, the problem is to decide, for a given element g of a group $\overline{G}$ and a (secret) homomorphism $\varphi\colon \overline{G} \to G$, whether $\varphi(g) = 1_G$ or $\varphi(g) = \varphi(g')$, where g' is another known random element of $\overline{G}$. The definition of the problem is as follows:

Definition 4 (Kernel decision problem)**.** Let GS be a group sampler. We define a *kernel decision game* for GS to be the following game, where $\mathcal{A}$ is an adversary for the game:

Kernel decision game for GS

$(\widetilde{G}, \overline{G}, G, \varphi, \mathsf{Sample}_{\overline{G}}, \mathsf{Sample}_H, \mathsf{Ker}) \leftarrow \mathsf{GS}(1^\lambda)$, $s \leftarrow_R \overline{G}$, $b \leftarrow_R \{0,1\}$
If $b = 0$ then
  $g_* \leftarrow_R H$
else
  $h \leftarrow_R H$, $g_* := sh$
end if
Then the adversary outputs $b' \leftarrow \mathcal{A}(1^\lambda, \widetilde{G}, \mathsf{Sample}_{\overline{G}}, \mathsf{Sample}_H, s, g_*)$

We say that *the kernel decision problem for* GS *is hard*, if for any PPT adversary $\mathcal{A}$, the *advantage* of $\mathcal{A}$ defined by $\mathsf{Adv}^{\mathrm{Ker}}_{\mathcal{A}}(\lambda) := |Pr[b' = b] - 1/2|$ is negligible, where $Pr[b' = b]$ is the probability that $b' = b$ in the kernel decision game for GS defined above.

3.2 Description of our framework

Based on the notions in Section 3.1, here we give the description of our proposed FHE scheme $\Pi = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{AND}, \mathsf{NOT})$ determined by a group sampler GS:

$\mathsf{KeyGen}(1^\lambda)$ First, the key generation algorithm generates

$$(\widetilde{G}, \overline{G}, G, \varphi, \mathsf{Sample}_{\overline{G}}, \mathsf{Sample}_H, \mathsf{Ker}) \leftarrow \mathsf{GS}(1^\lambda) \ . \tag{6}$$

Then the algorithm outputs $\mathsf{pk} := (\widetilde{G}, \mathsf{Sample}_{\overline{G}}, \mathsf{Sample}_H)$ as a public key and $\mathsf{sk} := (\mathsf{Ker})$ as a secret key. The plaintext space is $\mathcal{M} := \{0,1\}$.

$\mathsf{Enc}(\mathsf{pk}, m)$ Given a plaintext $m \in \{0,1\}$, first, the encryption algorithm generates $c_{\mathrm{T}} \in \overline{G}$ by $c_{\mathrm{T}} \leftarrow \mathsf{Sample}_{\overline{G}}$, which we call the *template component* of the ciphertext. Secondly, the algorithm generates $c_{\mathrm{B}} \in \overline{G}$, which we call the *body component* of the ciphertext, by

$$c_{\mathrm{B}} \leftarrow \begin{cases} h & \text{if } m = 0 \ , \\ c_{\mathrm{T}} h & \text{if } m = 1 \ , \end{cases} \text{ where } h \leftarrow \mathsf{Sample}_H \ . \tag{7}$$

Then the algorithm outputs $c := (c_{\mathrm{B}}, c_{\mathrm{T}})$ as a ciphertext for the plaintext m.

$\mathsf{Dec}(\mathsf{sk}, c)$ Given a ciphertext $c = (c_{\mathrm{B}}, c_{\mathrm{T}})$, the decryption algorithm outputs $\mathsf{Ker}(c_{\mathrm{B}}) \in \{0,1\}$.

$\mathsf{AND}(\mathsf{pk}, c_1, c_2)$ Given two ciphertexts $c_i = (c_{i,\mathrm{B}}, c_{i,\mathrm{T}})$ $(i \in \{1,2\})$, first, the algorithm generates $g \leftarrow \mathsf{Sample}_{\overline{G}}$. Secondly, the algorithm calculates c'_{B} and c'_{T} by

$$c'_{\mathrm{B}} := [g \cdot c_{1,\mathrm{B}} \cdot g^{-1}, c_{2,\mathrm{B}}] \, , \ c'_{\mathrm{T}} := [g \cdot c_{1,\mathrm{T}} \cdot g^{-1}, c_{2,\mathrm{T}}] \ . \tag{8}$$

Then the algorithm outputs $c' := (c'_{\mathrm{B}}, c'_{\mathrm{T}})$.

$\mathsf{NOT}(\mathsf{pk}, c)$ Given a ciphertext $c = (c_{\mathrm{B}}, c_{\mathrm{T}})$, first, the algorithm calculates c'_{B} and c'_{T} by

$$c'_{\mathrm{T}} := c_{\mathrm{T}} \, , \ c'_{\mathrm{B}} := c_{\mathrm{B}}^{-1} \cdot c'_{\mathrm{T}} \ . \tag{9}$$

Then the algorithm outputs $c' := (c'_{\mathrm{B}}, c'_{\mathrm{T}})$.

We note that the bit-OR operation can be efficiently realized by combining bit-AND and bit-NOT operations, therefore we do not include the homomorphic OR operation into the syntax of our proposed framework above. For simplifying expressions, we often omit the indication of a public key and a secret key in the notations below unless it causes ambiguity.

3.3 Properties of the resulting schemes

Here we study the properties of our proposed scheme Π defined in Section 3.2. First, for the security of Π, we note that the CPA game for Π is identical to the kernel decision game for the group sampler GS used in Π. Therefore, we have the following:

Theorem 1. *Suppose that the kernel decision problem (see Definition 4) for the group sampler* GS *used in our proposed scheme* Π *is hard. Then* Π *is CPA-secure.*

From now, we discuss the correctness and homomorphic properties of Π. Let C be any (possibly empty) circuit with polynomially many (say, n) input bits and polynomially many (say, n') output bits, which consists of polynomially many AND and NOT gates and no OR gates. Let $\widetilde{C}$ denote the algorithm obtained from C by replacing each AND and NOT gates with algorithms AND and NOT in Π, respectively. For $i \in \{1, \ldots, n\}$, let $m_i \in \{0,1\}$ and $c_i \leftarrow \mathsf{Enc}(m_i)$. We put $(c'_1, \ldots, c'_{n'}) := \widetilde{C}(c_1, \ldots, c_n)$, and for $i \in \{1, \ldots, n'\}$, set $m'_i \leftarrow \mathsf{Dec}(c'_i)$. Then we have the following result indicating the homomorphic functionality of Π (when C is an empty circuit, the result means the correctness of Π as a PKE scheme):

Theorem 2. *Suppose that the group sampler* GS *is commutator-separable (see Definition 3). Let* C *be a circuit as above, and let* $m'_1, \ldots, m'_{n'}$ *be the decryption results for ciphertexts generated from ciphertexts of* $m_1, \ldots, m_n$ *by the homomorphic operation* $\widetilde{C}$ *corresponding to* C *(see above). Then we have* $(m'_1, \ldots, m'_{n'}) = C(m_1, \ldots, m_n)$ *with overwhelming probability.*

Since the circuit C above may be an arbitrary circuit of polynomially bounded size (with no OR gates), Theorem 1 and Theorem 2 imply the following main result of the paper:

Theorem 3. *Suppose that the group sampler* GS *above is commutator-separable, and the kernel decision problem for* GS *is hard. Then* Π *is a CPA-secure FHE scheme.*

Now we give a proof of Theorem 2 above.

Proof of Theorem 2. First we introduce some auxiliary definitions: We say that a ciphertext $c = (c_{\mathrm{B}}, c_{\mathrm{T}})$ is of *type* 0, if $\varphi(c_{\mathrm{T}}) \in X$ and $\varphi(c_{\mathrm{B}}) = 1_G$ where X is a subset of $G \setminus \{1_G\}$ specified in the definition of commutator-separability (Definition 3); and c is of *type* 1, if $\varphi(c_{\mathrm{T}}) \in X$ and $\varphi(c_{\mathrm{B}}) = \varphi(c_{\mathrm{T}})$. Then by the condition for the algorithm Ker in Definition 2, it follows that $\mathsf{Dec}(c) = b$ with probability 1 if c is of type $b \in \{0, 1\}$.

By the conditions for $\mathsf{Sample}_{\overline{G}}$ and Sample_H in Definition 2, an output of $\mathsf{Enc}(m)$ for $m \in \{0, 1\}$ is of type m with overwhelming probability.

We check that, if a ciphertext c is of type $b \in \{0, 1\}$, then an output of $\mathsf{NOT}(c)$ is of type $\neg b$ (= NOT b) with probability 1. By the definition of NOT, the claim here follows from the relation $\varphi({c_{\mathrm{B}}}^{-1} \cdot c_{\mathrm{T}}) = \varphi(c_{\mathrm{B}})^{-1}\varphi(c_{\mathrm{T}})$.

On the other hand, we check that, if a ciphertext c_i is of type $b_i \in \{0, 1\}$ for each $i \in \{1, 2\}$, then an output of $\mathsf{AND}(c_1, c_2)$ is of type $b_1 \wedge b_2$ with overwhelming probability. Let $c' \leftarrow \mathsf{AND}(c_1, c_2)$. By the properties $\varphi(c_{1,\mathrm{T}}) \in X$ and $\varphi(c_{2,\mathrm{T}}) \in X$, the commutator-separability of GS (with $x_1 := \varphi(c_{1,\mathrm{T}})$ and $x_2 := \varphi(c_{2,\mathrm{T}})$) implies that $\varphi(c'_{\mathrm{T}}) \in X$ with overwhelming probability (we note that, since φ is surjective, $\varphi(g)$ is uniformly at random on G if g is uniformly at random on $\overline{G}$). From now, we suppose that $\varphi(c'_{\mathrm{T}}) \in X$. Now if $b_1 = b_2 = 1$, then $\varphi(c_{1,\mathrm{B}}) = \varphi(c_{1,\mathrm{T}})$ and $\varphi(c_{2,\mathrm{B}}) = \varphi(c_{2,\mathrm{T}})$, therefore we have $\varphi(c'_{\mathrm{B}}) = \varphi(c'_{\mathrm{T}})$ by the definition of AND. Hence c' is of type $1 = b_1 \wedge b_2$. On the other hand, if $b_1 = 0$, then, since $\varphi(c_{1,\mathrm{B}}) = 1_G$, we have

$$\varphi(g \cdot c_{1,\mathrm{B}} \cdot g^{-1}) = \varphi(g)\varphi(c_{1,\mathrm{B}})\varphi(g)^{-1} = \varphi(g)\varphi(g)^{-1} = 1_G \ , \tag{10}$$

therefore we have

$$\varphi(c'_{\mathrm{B}}) = 1_G\varphi(c_{2,\mathrm{B}}){1_G}^{-1}\varphi(c_{2,\mathrm{B}})^{-1} = 1_G \tag{11}$$

by the definition of AND. Hence c' is of type $0 = b_1 \wedge b_2$. Similarly, if $b_2 = 0$, then, since $\varphi(c_{2,\mathrm{B}}) = 1_G$, we have

$$\varphi(c'_{\mathrm{B}}) = \varphi(g \cdot c_{1,\mathrm{B}} \cdot g^{-1})1_G\varphi(g \cdot c_{1,\mathrm{B}} \cdot g^{-1})^{-1}{1_G}^{-1} = 1_G \tag{12}$$

by the definition of AND. Hence c' is of type $0 = b_1 \wedge b_2$. Therefore, the claim of this paragraph holds.

By the previous three paragraphs, a recursive argument implies that c'_i is a ciphertext of type b_i for every $i \in \{1, \ldots, n'\}$ with overwhelming probability, where $(b_1, \ldots, b_{n'}) := C(m_1, \ldots, m_n)$ (note that C involves only polynomially many gates). Then $\mathsf{Dec}(c'_i) = b_i$ for each index i by the first paragraph of the proof, therefore the proof of Theorem 2 is concluded. □

3.4 Examples of commutator-separable groups

Here we study the condition for commutator-separability in Definition 3 more in detail. First, we give some results in order to give a sufficient condition for a group sampler to be commutator-separable. The key notion in the sufficient condition is the following: For any group G and any $g \in G$, the *centralizer* $Z_G(g)$ of g in G is defined by

$$Z_G(g) := \{h \in G \mid gh = hg\} \ . \tag{13}$$

Then the following lemma provides the above-mentioned sufficient condition:

Lemma 1. *Let G be an arbitrary finite group, and let $X \subset G$. Then for any $x_1, x_2 \in G$, we have*

$$Pr[\,[gx_1g^{-1}, x_2] \notin X] \leq \frac{(|G| - |X|) \cdot |Z_G(x_1)| \cdot |Z_G(x_2)|}{|G|} \ , \tag{14}$$

where the probability is taken over the choice of $g \leftarrow_R G$.

Proof. First, we put $Y := \{y \in G \setminus X \mid [gx_1g^{-1}, x_2] = y \text{ for some } g \in G\}$. Then we have

$$Pr[\,[gx_1g^{-1}, x_2] \notin X] = \frac{\sum_{y \in Y} |\{g \in G \mid [gx_1g^{-1}, x_2] = y\}|}{|G|} \ . \tag{15}$$

For each $y \in Y$, we put $G_y := \{g \in G \mid [gx_1g^{-1}, x_2] = y\}$, and fix an element $g_y \in G_y$. Now for each $g \in G_y$, we have

$$\begin{aligned} (gx_1g^{-1})x_2(gx_1g^{-1})^{-1}x_2{}^{-1} &= [gx_1g^{-1}, x_2] \\ &= [g_yx_1g_y{}^{-1}, x_2] = (g_yx_1g_y{}^{-1})x_2(g_yx_1g_y{}^{-1})^{-1}x_2{}^{-1} \ , \end{aligned} \tag{16}$$

therefore

$$(gx_1g^{-1})x_2(gx_1g^{-1})^{-1} = (g_yx_1g_y{}^{-1})x_2(g_yx_1g_y{}^{-1})^{-1} \ , \tag{17}$$

hence $(g_yx_1g_y{}^{-1})^{-1}(gx_1g^{-1}) \in Z_G(x_2)$. Now for each $h \in Z_G(x_2)$, we put

$$G_{y,h} := \{g \in G_y \mid (g_yx_1g_y{}^{-1})^{-1}(gx_1g^{-1}) = h\} \ . \tag{18}$$

If $G_{y,h} \neq \emptyset$, then we fix an element $g_{y,h} \in G_{y,h}$. Now for any $g \in G_{y,h}$, we have

$$(g_yx_1g_y{}^{-1})^{-1}(gx_1g^{-1}) = (g_yx_1g_y{}^{-1})^{-1}(g_{y,h}x_1g_{y,h}{}^{-1}) \ , \tag{19}$$

therefore $gx_1g^{-1} = g_{y,h}x_1g_{y,h}{}^{-1}$ and $g_{y,h}{}^{-1}g \in Z_G(x_1)$. This implies that $|G_{y,h}| \leq |Z_G(x_1)|$ for any $h \in Z_G(x_2)$, therefore, by the argument above,

$$|G_y| \leq \sum_{h \in Z_G(x_2)} |G_{y,h}| \leq |Z_G(x_1)| \cdot |Z_G(x_2)| \ . \tag{20}$$

Hence we have

$$\begin{aligned} Pr[\,[gx_1g^{-1}, x_2] \notin X] &= \frac{\sum_{y \in Y} |G_y|}{|G|} \\ &\leq \frac{|Y| \cdot |Z_G(x_1)| \cdot |Z_G(x_2)|}{|G|} \leq \frac{(|G| - |X|) \cdot |Z_G(x_1)| \cdot |Z_G(x_2)|}{|G|} \ , \end{aligned} \tag{21}$$

therefore Lemma 1 holds. □

From now, by using Lemma 1 above, we show that the special linear group $SL_2(\mathbb{F})$ of size two over a (sufficiently large) finite field $\mathbb{F}$, which is defined by

$$SL_2(\mathbb{F}) := \{A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \mid a, b, c, d \in \mathbb{F} \text{ and } \det(A) = ad - bc = 1\} \ , \tag{22}$$

satisfies the condition in Definition 3. For the purpose, we give the following evaluation of the cardinality of the centralizers in the group:

Lemma 2. *Let $\mathbb{F}$ be any finite field. For any $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in SL_2(\mathbb{F})$ with $A \neq \pm I$, we have $|Z_{SL_2(\mathbb{F})}(A)| \leq 2|\mathbb{F}|$ if $b \neq 0$ or $c \neq 0$, and $|Z_{SL_2(\mathbb{F})}(A)| = |\mathbb{F}| - 1$ if $b = c = 0$.*

Proof. Let $X = \begin{pmatrix} x & y \\ z & w \end{pmatrix} \in Z_{SL_2(\mathbb{F})}(A)$, i.e., $X \in SL_2(\mathbb{F})$ and $XA = AX$. Then we have

$$\det(X) = 1 \, , \, \begin{pmatrix} ax + cy & bx + dy \\ az + cw & bz + dw \end{pmatrix} = \begin{pmatrix} ax + bz & ay + bw \\ cx + dz & cy + dw \end{pmatrix} \ , \tag{23}$$

therefore

$$xw - yz = 1 \, , \, cy = bz \, , \, bx + dy = ay + bw \, , \, az + cw = cx + dz \ . \tag{24}$$

We consider the case that $b \neq 0$. Then we have $z = b^{-1}cy$ and $w = x + b^{-1}(d-a)y$, therefore $x^2 + b^{-1}(d-a)xy - b^{-1}cy^2 = 1$. Now for each $y \in \mathbb{F}$, the quadratic equation in x has at most two solutions, and z and w are uniquely determined from x and y by the relations above. This implies that the number of the possible X is at most $2|\mathbb{F}|$. The argument for the case $c \neq 0$ is similar; x and y are linear combinations of z and w, and w satisfies a quadratic equation when an element $z \in \mathbb{F}$ is fixed, therefore the number of the possible X is at most $2|\mathbb{F}|$.

On the other hand, we consider the remaining case that $b = c = 0$. By the condition $\det(A) = 1$, we have $ad = 1$, therefore $a \neq 0$ and $d \neq 0$. Now we have $dy = ay$ and $az = dz$, while the assumption $A \neq \pm I$ implies that $a \neq d$. Therefore, we have $y = 0$ and $z = 0$. This implies that $xw = 1$, therefore $w \neq 0$ and $x = w^{-1}$. Hence, the number of the possible X is $|\mathbb{F}| - 1$. This concludes the proof of Lemma 2. □

By combining Lemma 1 and Lemma 2, we have the following result:

Proposition 1. *If the output of a group sampler GS satisfies that $G = SL_2(\mathbb{F})$ and $|\mathbb{F}| = \lambda^{\omega(1)}$, then GS is commutator-separable, where the set X in the definition of commutator-separability can be any subset of $G \setminus \{\pm I\}$ satisfying that $|G| - |X|$ is polynomially bounded in λ (e.g., $X = G \setminus \{\pm I\}$).*

Proof. First, it is known that $|G| = (|\mathbb{F}| + 1)|\mathbb{F}|(|\mathbb{F}| - 1)$, therefore $1/|G|$ is negligible. Hence $|X|/|G|$ is overwhelming by the assumption that $|G| - |X|$ is polynomially bounded. On the other hand, for any $x_1, x_2 \in X$, Lemma 2 implies that $|Z_G(x_1)| \leq 2|\mathbb{F}|$ and $|Z_G(x_2)| \leq 2|\mathbb{F}|$. Therefore, by Lemma 1, we have

$$Pr[\,[gx_1g^{-1}, x_2] \notin X] \leq \frac{|G| - |X|}{(|\mathbb{F}| + 1)|\mathbb{F}|(|\mathbb{F}| - 1)} \cdot 2|\mathbb{F}| \cdot 2|\mathbb{F}| = \frac{4|\mathbb{F}|}{(|\mathbb{F}| + 1)(|\mathbb{F}| - 1)}(|G| - |X|) \tag{25}$$

where $g \leftarrow_R G$. The right-hand side is negligible by the assumption that $|G| - |X|$ is polynomially bounded. Hence Proposition 1 holds. □

Acknowledgments

The author thanks members of Shin-Akarui-Angou-Benkyou-Kai for their helpful comments. In particular, the author thanks Shota Yamada for inspiring him with motivation to the present work; Takashi Yamakawa for pointing out the relation to the proof of the Barrington's theorem; and Takahiro Matsuda, Keita Emura, Yoshikazu Hanatani, Jacob C. N. Schuldt and Goichiro Hanaoka for giving many precious comments on the work. The author also thanks the anonymous referees of a previous conference submission version of the paper for their careful reviews and valuable comments.

References

[1] D. A. Barrington, Bounded-width polynomial-size branching programs recognize exactly those languages in NC^1, in: *Proceedings of STOC 1986*, 1986, pp.1–5

[2] S. R. Blackburn, C. Cid and C. Mullan, Group theory in cryptography, in: *Proceedings of Group St Andrews 2009 in Bath*, LMS Lecture Note Series 387, 2011, pp.133–149

[3] Z. Brakerski, Fully homomorphic encryption without modulus switching from classical GapSVP, in: *Proceedings of CRYPTO 2012*, LNCS 7417, 2012, pp.868–886

[4] Z. Brakerski, C. Gentry and V. Vaikuntanathan, (Leveled) fully homomorphic encryption without bootstrapping, in: *Proceedings of ITCS 2012*, 2012, pp.309–325

[5] Z. Brakerski and V. Vaikuntanathan, Efficient fully homomorphic encryption from (standard) LWE, in: *Proceedings of FOCS 2011*, 2011, pp.97–106

[6] Z. Brakerski and V. Vaikuntanathan, Fully homomorphic encryption from Ring-LWE and security for key dependent messages, in: *Proceedings of CRYPTO 2011*, LNCS 6841, 2011, pp.505–524

[7] J. H. Cheon, J.-S. Coron, J. Kim, M. S. Lee, T. Lepoint, M. Tibouchi and A. Yun, Batch fully homomorphic encryption over the integers, in: *Proceedings of EUROCRYPT 2013*, LNCS 7881, 2013, pp.315–335

[8] J.-S. Coron, A. Mandal, D. Naccache and M. Tibouchi, Fully homomorphic encryption over the integers with shorter public keys, in: *Proceedings of CRYPTO 2011*, LNCS 6841, 2011, pp.487–504

[9] J.-S. Coron, D. Naccache and M. Tibouchi, Public key compression and modulus switching for fully homomorphic encryption over the integers, in: *Proceedings of EUROCRYPT 2012*, LNCS 7237, 2012, pp.446–464

[10] M. Dijk, C. Gentry, S. Halevi and V. Vaikuntanathan, Fully homomorphic encryption over the integers, in: *Proceedings of EUROCRYPT 2010*, LNCS 6110, 2010, pp.24–43

[11] C. Gentry, Fully homomorphic encryption using ideal lattices, in: *Proceedings of STOC 2009*, 2009, pp.169–178

[12] C. Gentry, Toward basing fully homomorphic encryption on worst-case hardness, in: *Proceedings of CRYPTO 2010*, LNCS 6223, 2010, pp.116–137

[13] C. Gentry and S. Halevi, Implementing Gentry's fully-homomorphic encryption scheme, in: *Proceedings of EUROCRYPT 2011*, LNCS 6632, 2011, pp.129–148

[14] C. Gentry and S. Halevi, Fully homomorphic encryption without squashing using depth-3 arithmetic circuits, in: *Proceedings of FOCS 2011*, 2011, pp.107–109

[15] C. Gentry, S. Halevi and N. P. Smart, Better bootstrapping in fully homomorphic encryption, in: *Proceedings of PKC 2012*, LNCS 7293, 2012, pp.1–16

[16] C. Gentry, S. Halevi and N. P. Smart, Fully homomorphic encryption with polylog overhead, in: *Proceedings of EUROCRYPT 2012*, LNCS 7237, 2012, pp.465–482

[17] J. Katz, A. Thiruvengadam and H.-S. Zhou, Feasibility and infeasibility of adaptively secure fully homomorphic encryption, in: *Proceedings of PKC 2013*, LNCS 7778, 2013, pp.14–31

[18] K. H. Ko, S. Lee, J. H. Cheon, J. W. Han, J.-S. Kang and C. Park, New public-key cryptosystem using braid groups, in: *Proceedings of CRYPTO 2000*, LNCS 1880, 2000, pp.166–183

[19] A. López-Alt, E. Tromer and V. Vaikuntanathan, On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption, in: *Proceedings of STOC 2012*, 2012, pp.1219–1234

[20] S.-H. Paeng, K.-C. Ha, J. H. Kim, S. Chee and C. Park, New public key cryptosystem using finite non abelian groups, in: *Proceedings of CRYPTO 2001*, LNCS 2139, 2001, pp.470–485

[21] A. Silverberg, Fully homomorphic encryption for mathematicians, IACR Cryptology ePrint Archive 2013/250, 2013, `http://eprint.iacr.org/2013/250`

[22] N. P. Smart and F. Vercauteren, Fully homomorphic encryption with relatively small key and ciphertext sizes, in: *Proceedings of PKC 2010*, LNCS 6056, 2010, pp.420–443

[23] D. Stehlé and R. Steinfeld, Faster fully homomorphic encryption, in: *Proceedings of ASIACRYPT 2010*, LNCS 6477, 2010, pp.377–394