Can there be efficient and natural FHE schemes?*

Kristian Gjøsteen and Martin Strand

Norwegian University of Science and Technology, Trondheim, Norway
{martin.strand,kristian.gjosteen}@math.ntnu.no

Abstract. In 1978, Rivest, Adleman and Dertouzos asked for algebraic systems for which useful privacy homomorphisms exist. To date, the only acknownledged result is noise based encryption combined with bootstrapping. Before that, there were several failed attempts.
We prove that fully homomorphic schemes are impossible for several algebraic structures. Then we develop a characterisation of all fully homomorphic schemes and use it to analyse three examples. Finally, we propose a conjecture stating that secure FHE schemes must either have a significant ciphertext expansion or use unusual algebraic structures.

Keywords: fully homomorphic encryption, characterisation

1 Introduction

In 1978 Rivest, Adleman and Dertouzos [32] posed two questions about *privacy homomorphisms*:

Q1 Does this approach have enough utility to make it worthwhile in practice?
Q2 For what algebraic systems U does a useful privacy homomorphism exist?

A privacy homomorphism was defined to be an encryption function that would permit direct computations on the encrypted data. Gentry's construction of the first fully homomorphic encryption (FHE) scheme [19] answers this goal, but in a slightly different manner than what Rivest et al. envisioned. The original problem considered clear algebraic structures and mappings between these, while Gentry succeeded using noise-based constructions, putting less emphasis on the mappings.

One could say that both questions are still open. Gentry's breakthrough [19] in creating the first fully homomorphic encryption scheme has been followed by a number of much more efficient schemes [8, 10, 14, 15, 21, 26]. There have been several earlier attempts at finding privacy homomorphisms, but none successful. Instead, there have been some negative results. Ahituv et al. [1] proved that an vector space isomorphism on $\mathbb{F}_2^n$ cannot be secure.

* A previous version of this manuscript was named "Fully homomorphic encryption must be fat or ugly?"

The answer to the first question seems to be positive. Some FHE applications have been demonstrated, but the list of theoretical applications is far more extensive than what anyone has tried to implement yet.

The first part of this work aims at closing some doors for the second question. To do this, we try to analyse the possibility for *natural* schemes based on automorphisms or isomorphisms on various structures. An example of such a scheme is the Pohlig-Hellman blockcipher. For completeness, we also survey previous results in this line of research.

Note that this analysis does not include the modern, noise-based FHE schemes. Although one can define operations on the ciphertext spaces by always performing a bootstrapping operation after an addition or multiplication, the ciphertext spaces do not become rings or any other well-known structure.

A second approach to achieve security is to embed the plaintexts into a larger set, such as for instance in ElGamal. We extend our arguments to show that also this kind of scheme is impossible for some of the structures we study. In this part of the work we do not assume a public *encryption* key, only that the adversary can perform evaluation of ciphertexts.

Furthermore, we extend the characterisation of Armknecht et al. [4] to handle all public key FHE schemes over any algebraic structure. This yields a simple transformation from any scheme to a suitable decision problem in order to analyse whether the scheme can be secure or not. Our results immediately allow us to prove that two proposed schemes are insecure, while our analysis supports the existing work on a third scheme.

The sum of our results make us propose the conjecture that FHE schemes either need to have a rather big expansion, or have a less favourable structure. This conjecture is an initial answer Gentry's challenge to the mathematical community [20, p. 616].

The paper is organised as follows. In Section 3, we analyse the possibility of fully homomorphic schemes on a number of common algebraic structures, and reach negative answers for the most useful structures. This technique does not allow us to consider the current FHE schemes, so we proceed to extend Armknecht et al.'s characterisation in Section 4. A brief recap of the algebra used in this paper is provided in the next section.

Throughout the text, $\mathcal{P}$ will denote the plaintext space and $\mathcal{C}$ will denote the ciphertext space.

1.1 Our contribution

All of the aknowledged FHE schemes in existence today are based on lattices, and usually feature a large ciphertext expansion. While one could say that some schemes are almost practical [13], we should not expect to see them in widespread use just yet. The main reason is probably the communcation cost, which then again influences the computational cost. The big question is to find out if this can be reduced significantly while maintaining usefulness and security.

We are not proposing new schemes, and any discussion of existing schemes is only to demonstrate our new techniques for analysis. Our contribution can be summarised in two points.

- We provide a general tool for analysing new public key FHE schemes, and we believe that the technique will easily distinguish between weak and good constructions.
- We extend existing results by proving that a number of possible FHE schemes must be insecure.

1.2 Related work

Following the original problem, there were a few attemts at creating privacy homomorphisms, usually followed by attacks, see Yu et al. [34] for a brief survey. Earlier impossibility results include Ahituv et al. [1] as mentioned above, and Yu et al. who proved that a FHE scheme cannot achieve IND-CCA2 security.

Boneh and Lipton [7] demonstrated that any deterministic fully homomorphic encryption scheme over $\mathbb{Z}/n\mathbb{Z}$ is breakable in sub-exponential time.

Finally, Armknecht et al. [3] have proven that a group homomorphic scheme will be vulnerable against a quantum adversary. A subset of the same authors also showed that no group homomorphic scheme with a prime order ciphertext group can be IND-CPA secure [5].

Our characterisation extends the construction provided by Armknecht et al. [4].

2 Preliminaries

We assume that the reader has some familiarity to fully homomorphic encryption, so we only give a brief overview. The interested reader should look up the survey by Armknecht et al. [2].

The term FHE has come to mean two things: Either that the scheme can evaluate both addition and multiplication, or that it can evaluate any circuit of *any* multiplicative depth. The scheme is i-hop if it can evaluate i ciruits after each other, or ∞-hop if there is no limit. Note that all *somewhat* homomorphic scheme with a sufficiently small decryption circuit can be made fully homomorphic and ∞-hop using Gentry's original bootstrapping idea [19]. A levelled scheme can compute any circuit with multiplicative depth up to its designated level.

2.1 Algebraic structures

The reader should know the definition of groups, rings, fields and vector spaces. Recall that a division rings a ring where every nonzero element has an inverse, and that all finite division rings are commutative, hence fields. We assume that all rings have identity.

Furthermore, we also need two more concepts, namely *modules* and *algebras*.

A module is a generalization of vectors spaces where the coefficients come from a ring. For a ring R, a (left) R-module is an additive abelian group M together with a scalar multiplication with ring elements on the left. We also have right R-modules, but for commutative rings, left and right R-modules are the same.

It is well known that any vector space of dimension n is isomorphic to n copies of the field. This is not true for modules. In particular, even the word "dimension" is not even well-defined, and not all modules have a basis. Those that have, are called *free* modules, and if every basis has the same number of elements, say n, we say that the module is of rank n.

- If I is an ideal of R, then I is also an R-module. In particular, R itself is an R-module.
- Any vector space over a field or division ring is also a module.
- Matrices over a ring R forms an R-module.

We are interested in mappings between modules. Let M and N be R-modules. An R-homomorphism f is a function $f : M \to N$ such that for all $r \in R$ and $m, m_1, m_2 \in M$,

$$f(rm) = rf(m)$$
$$f(m_1 + m_2) = f(m_1) + f(m_2).$$

For a field k, a *k-algebra* A is a k-vector space which is also equipped with a multiplication operation compatible with the scalar multiplication, such that A is a ring in its own respect. An algebra mapping is a function which is both a linear transformation on A as a vector space and a ring homomorphism on A.

2.2 The Wedderburn-Artin theorem

Recall that the structure theorem for finitely generated abelian groups states that any such group is isomorphic to a direct sum of copies of $\mathbb{Z}$ and cyclic groups of prime order. Rings generally lack a corresponding theorem. However, for certain classes of rings, we know the structure in detail. For instance, every finite field with the same cardinality is isomorphic. In the case of *semisimple* rings, we have the Wedderburn-Artin theorem.

Theorem 1 ([6], p. 382). *Let R be a left (or right) artinian ring with unity and no nonzero nilpotent ideals. Then R is isomorphic to a finite direct sum of matrix rings over division rings.*

The first sentence of the theorem is one of several equivalent definitions of a semisimple ring. An artinian ring is one where any descending chain (under inclusion) of ideals becomes constant after a finite number. Any finite ring is trivially artinian, while the integers $\mathbb{Z}$ are not. Consider this chain of ideals

$$(2) \supseteq (2^2) \supseteq (2^3) \supseteq \cdots$$

to see that it need not stabilise.

Also recall that an ideal I is nilpotent if there exists an integer n such that $I^n = (0)$.

3 Automorphisms on structures

We now consider specific structures. In this section, we treat them in the symmetric case. This is partly for a practical reason – one motivation behind this work was to explore the possibility for very efficient schemes without any expansion. The conclusion seems to be that they are unlikely or at best not practical. We divide our potential schemes into four types.

1. Identical spaces $\mathcal{P} = \mathcal{C}$
2. Isomorphic spaces $\mathcal{P} \simeq \mathcal{C}$ (but possibly with different descriptions)
3. $\mathcal{C}$ is larger than $\mathcal{P}$, but only with a constant expansion
4. $\mathcal{C}$ is larger than $\mathcal{P}$, and the expansion depends on the parameters

In all scenarios, we assume that both $\mathcal{P}$ and $\mathcal{C}$ share the same kind of algebraic structure. For instance, if $\mathcal{P}$ is a vector space over a field k, then $\mathcal{C}$ must also be a k-vector space. For schemes of the second type, we stress that the spaces really must be isomorphic. The next section will deal with the situation where $\mathcal{P}$ is isomorphic to the residue classes of $\mathcal{C}$.

Since we are primarily looking for very efficient schemes, we will not use any energy on schemes of Type 4. Note that there still is a certain jump from that to the state of art today. The ciphertext spaces of modern noise based schemes are not rings, not even if you consider bootstrapping as a part of every multiplication operation. One should rather use the techniques of Section 4 to analyse these schemes.

From now on, we will refer to these cases by their numbers.

It is trivial to observe that any scheme of Type 1 or 2 must be deterministic, and hence not achieve semantical security, while those of Type 3 typically should be randomised, such as ElGamal or Paillier, which always feature a data doubling.

3.1 Achievable security

For security, we use a variant of the standard notion of semantic security, where the adversary is challenged to decide whether we are using an instance of the cryptosystem, or simply a random permutation. Obviously, it can be easy to distinguish any homomorphism from a permutation, so we need to change the game slightly. We assume that we have access to a black box that can efficiently compute any mapping ϵ into $\mathcal{C}$ such that there exists another mapping $\delta : \mathcal{C} \to \mathcal{P}$ such that $\delta \circ \epsilon$ acts as the identity on $\mathcal{P}$. For instance, in the first setting above, ϵ would be any automorphism on $\mathcal{P}$. Essentially, the black box should be able to compute everything that could have been an encryption, whereas the scheme is limited to the automorphisms indexed by the keys.

Definition 1. *We say that a symmetric fully homomorphic scheme of Type 1 or 2 is secure if a polynomial time algorithm is unable to distinguish an encryption from a random automorphism. Furthermore, any automorphism used for encryption must be hard to invert.*

We allow the adversary to query a number of encryptions before giving his answer, but no decryptions.

3.2 Groups

Homomorphic schemes on groups are well known, and there are known secure examples.

Theorem 2. *There exist secure group homomorphic schemes of Type 1, and therefore of Type 2–4 as well.*

The theorem is easily proved by example, for instance the Pohlig-Hellman exponentiation cipher [30]. Let G be a cyclic group of secret order n, and choose e, d such that $ed \equiv 1 \pmod{\phi(n)}$. Encryption and decryption is as with RSA. Then all automorphisms on the group are indexed by e, so the scheme is trivially secure. Reassuringly, it has also withstood cryptanalysis.

It is easy to construct an example to show that this implies the existence of a secure scheme when the groups are isomorphic but with different descriptions, or if we add further copies of the group. However, ElGamal is a better example in that respect.

Textbook RSA is not secure according to our definition. For simplicity, assume that $n = pq = (2p' + 1)(2q' + 1)$ with p, q, p', q' prime, and let G be the group of multiplicative units in $\mathbb{Z}/n\mathbb{Z}$. We then know that, as groups

$$G \simeq \mathbb{Z}/p'\mathbb{Z} \times \mathbb{Z}/q'\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}.$$

This group has $6p'q'$ automorphisms [22], where the factor 6 comes from the permutation of the order 2 elements in the Klein 4 subgroup V. RSA is able to produce $p'q'$ of these automorphisms, all of which leave the four elements of V fixed. The automorphisms outside the RSA subset either leave -1 alone, or swap it with one of the other order 2 elements. In the latter case, it is easy to distinguish. In the first case, a distinguisher must find one of the two other special elements. That will allow the adversary to factor n, using the same idea as in Rabin's oblivious transfer [31].

3.3 Vector spaces

Ahituv, Lativ and Neumann [1] showed that a homomorphism $f : \mathbb{F}_2^n \to \mathbb{F}_2^n$ cannot be secure, by using f on the basis to essentially compute the inverse. Their argument easily extends to a vector space over any field.

For spaces of different sizes, fix a field k, and let $\mathcal{P}$ and $\mathcal{C}$ be k-vector spaces, with $p = \dim \mathcal{P} \leq \dim \mathcal{C} = q$. Get encryptions of at least pq linearly independent vectors. There exists a matrix D such that $Dc_i = m_i$ for each pair (m, c). Solve the resulting system for D, which is essentially the decryption key. Clearly, such a scheme cannot be secure.

An apparent counterexample would be the Goldwasser-Micali cryptosystem. Let n be a product of two primes, let y be a non-quadratic residue modulo n

and let x_i be units modulo n. To encrypt a vector m in $\mathbb{F}_2^\ell$, compute a tuple $(c_1, \cdots, c_\ell)$ where $c_i = y^{m_i} x_i^2$ modulo n. The ciphertext space is then a $\mathbb{Z}/n\mathbb{Z}$-module where addition is given by pointwise multiplication and scalar multiplication is done by exponentiation. Decrypt by checking whether each component is a quadratic residue or not. However, C is not a $\mathbb{F}_2$-vector space, so it does not fit in our system. However, it shows that one doesn't need to go far away from these constructions to find something that is secure, although with a certain expansion factor.

Theorem 3. *Let k be a field and let $\mathcal{P}$ and $\mathcal{C}$ be k-vector spaces. Then there are no secure encryption schemes between $\mathcal{P}$ and $\mathcal{C}$.*

3.4 Fields

We only consider finite fields, and we can separate them into prime fields and extension fields. For prime fields, there are only fields of the kind $\mathbb{Z}/p\mathbb{Z}$, possibly with some strange description, but nonetheless easy to convert into the canonical form. There is only the identity automorphism.

For extension fields, where $\mathcal{P} \simeq \mathcal{C}$, an algorithm by Lenstra [23] demonstrates that it is easy to convert everything into practical descriptions of the fields, and then compute any isomorphisms between those fields. Hence, there can be no secure homomorphic schemes between fields of the same size.

Let us tackle the third type. First note that the fields must have the same characteristic, otherwise there would be no structure preserving mappings between them. We can therefore assume that both $\mathcal{P}$ and $\mathcal{C}$ are extensions of the same prime field $\mathbb{F}_p$. Therefore, we can also view $\mathcal{P}$ and $\mathcal{C}$ as $\mathbb{F}_p$-vector spaces, reducing the problem to the one in the previous subsection.

Alternatively, one can adjoin extra elements to $\mathcal{P}$ to make it isomorphic to $\mathcal{C}$, and then compute an isomorphism based on known plaintext-ciphertext pairs.

If the encryption is merely a deterministic injection into a larger ciphertext space it can still not be secure, as we can view the image as a field in its own right.

Theorem 4. *Let $\mathcal{P}$ and $\mathcal{C}$ be k fields. Then there are no secure encryption schemes between $\mathcal{P}$ and $\mathcal{C}$.*

3.5 Rings

Rings are far harder to tackle than the previous structures, and we are not able to give a definitive answer. The key information is the set of automorphisms on a ring. We start by getting an overview over those for reduced rings, that is, rings with no nonzero nilpotent elements.

Reduced rings Note that since R is finite, it is in particular left (and right) artinian. Assume that R has no nonzero nilpotent ideals, then R is semisimple. By

the Wedderburn-Artin theorem, R is then isomorphic to a finite direct product of matrix rings over division rings,

$$R \simeq M_{n_1}(D_1) \oplus \cdots \oplus M_{n_\ell}(D_\ell).$$

We have assumed that R has no nilpotent ideals. Since R is artinian, nil and nilpotent ideals are the same, so we have simultaneously assumed there are no nonzero nil ideals. In particular, this means that the radical of R, which is precisely all nil elements, is the zero ideal.

Now, if at least one $n_i \geq 2$, then one can create a nilpotent element in that matrix ring, a contradiction. Furthermore, a finite division ring is always a field, so we reach the simplification

$$R \simeq k_1 \oplus \cdots \oplus k_\ell.$$

Assume that there are m distinct fields up to isomorphism, and with ℓ_i fields in each set, $1 \leq i \leq m$. Gather isomorphic fields together. For each i, let d_i be the degree of the field extension. Let ϕ be an automorphism on R. We can write it as $(\phi_{1,1}, \ldots \phi_{1,\ell_1}, \ldots, \phi_{1,\ell_m})$. There are d_i automorphisms on each field, and one can use any permutation on each set of isomorphic field. This yields a total of

$$\prod_{i=1}^{m} (d_i - 1)!\, \ell_i!$$

automorphisms on R, which bounds the size of the key space.

To see why there are no more isomorphisms, consider $R = k_1 + k_2$, where $k_1 = k_2$ are extension fields over k. Define a function $\phi : R \to R$ which is the identity on $k \times k$, and swaps the components otherwise. Take $(x_1, x_2) \in k^2$ and $(x_1', x_2') \notin k^2$. If one applies ϕ on the sum, one can see that it differs from $\phi(x_1, x_2) + \phi(x_1', x_2')$, so ϕ cannot be a ring homomorphism.

Now consider the special case where we let R be a k-algebra. Friedl and Rónyai and others [11,16] have described polynomial time algorithms to compute a Wedderburn-Artin decomposition explicitly. We say "a" rather than "the", since there can be several isomorphic decompositions, given by permutations on isomorphic summands. One can therefore compute the automorphisms for each field separately by the techniques described above. The feasibility of the computation of any automorphism therefore depend only on $\prod_{i=1}^{m} \ell_i!$ being sufficiently large.

Semisimple k-algebras We shift the assumptions slightly. We now allow more nilpotent elements, but only for algebras over a field k. The aforementioned algorithms also work in this setting. It computes orthogonal idempotents in the center of each summand, and next a basis for each matrix ring. Computing the mapping between the algebra and the decomposition is easy: Multiply with the idempotents to decompose the element, and then use linear algebra. We therefore have a canonical description of any ring element, up to permutations of components and bases.

In particular, this setting includes many matrix rings. Note that we cannot automatically use our reasoning about vector spaces here, since linear mappings are different from ring homomorphisms. However, vector transformations form a subset of algebra homomorphisms. Brakerski [9] have proven that whenever decryption is an inner product computation involving the secret key, then the scheme cannot be CPA secure. In particular, this involves all schemes using inner automorphisms on matrix rings, i.e. where a square matrix M (possibly encoding the ciphertext from a subring) is conjugated, $C = A^{-1}MA$. In particular, by the Skolem-Noether theorem, every automorphism of a matrix ring over a field is inner.

Large ciphertext rings Finally, consider the case where both $\mathcal{P}$ and $\mathcal{C}$ are rings, and let $I = \ker \mathsf{Dec} = \mathsf{Dec}^{-1}(0)$. By the reasoning in Section 4 and the first isomorphism theorem, we therefore know that $\mathcal{P} \simeq \mathcal{C}/I$, so we can identify $\mathcal{P}$ as a subring of $\mathcal{C}$. If $\mathcal{P}$ is commutative, then $\mathcal{C}$ trivially becomes an associative $\mathcal{P}$-algebra. When $\mathcal{P}$ is a field, this reduces to the vector space scenario above, hence not secure.

The general challenge is to distinguish elements of I and R.

3.6 Modules

Modules can be quite similar to vector spaces, so in particular for free modules over well behaved rings, one would expect that the same techniques should apply. On the other hand, an abelian group is a $\mathbb{Z}$-module, and $\mathbb{Z}$ is certainly a very well behaved ring, despite its lack of nontrivial units.

4 Characterisation

In this section we treat fully homomorphic schemes with as much generality as we can, by considering various algebraic structures. By that we mean something that consists of a set of elements, is closed under at least one operation and satisfies certain axioms on the operations. Examples include groups, rings, vector spaces and so on. Note in particular that we ignore somewhat and levelled homomorphic schemes, although some of the reasoning can also be applied to those. The short version of this section is that any homomorphic encryption consists of a mapping into the ciphertext space and an addition with a random encryption of zero. For the scheme to be secure, at least one of those operations must protect the message.

We also return to the more conventional security definition IND-CPA for the remainder of the paper.

Fix a structure S with $n \geq 1$ operations $*_1, \ldots, *_n$, where $*_1$ is a binary operation such that for any object O with structure S, we have a neutral element 0 with respect to $*_1$, and that $*_1$ is a bijection when the second coordinate is fixed. This implies that all elements have inverses with respect to the first operation. We stress that we do not put any assumptions on the other operations.

Any structure that contains a group structure satisfies this requirement when $*_1$ is taken to be the group operation, typically addition. For the binary case, it requires either XOR or AND. If the set of binary operators is functionally complete, we can add either to the set and rearrange.

Let $\mathcal{P}$ be an object with structure S, let $\mathcal{C}$ be a set (without any structure) with operations $*'_1, \ldots, *'_n$, and let $1_{\mathcal{P}}$ be the identity map on $\mathcal{P}$. Let $\epsilon : \mathcal{P} \to \mathcal{C}$ and $\delta : \mathcal{C} \to \mathcal{P}$ be maps such that $\delta \circ \epsilon = 1_{\mathcal{P}}$, and for all $1 \leq i \leq n$ and all $c_1, \ldots, c_{a_i}$ where a_i is the number of elements that $*_i$ takes as input, we have

$$\delta(*'_i(c_1, \ldots, c_{a_i})) = *_i(\delta(c_1), \ldots, \delta(c_{a_i})).$$

We call $(\mathcal{P}, \mathcal{C}, \epsilon, \delta)$ a S-homomorphic tuple.

Now assume we have a set (Instance, Epsilon, Sample, Delta, Op) of algorithms such that

Instance	takes in security parameter λ and returns a S-homomorphic tuple
Sample	takes in $(\mathcal{C}, \epsilon)$ and returns a random element from $\delta^{-1}(0)$
Epsilon	takes in $(\mathcal{C}, \epsilon, m)$ and computes $\epsilon(m)$
Delta	takes in $(\mathcal{C}, \delta, c)$ and computes $\delta(c)$
Op	takes $(\mathcal{C}, \epsilon, i, c_1, \ldots, c_{a_i})$ as input and computes $*'_i(c_1, \ldots, c_{a_i})$.

Based on this, we can now construct an abstract encryption scheme.

Definition 2. *The Abstract Homomorphic Encryption Scheme (AHES) is an encryption scheme (*Gen*,* Enc*,* Eval*,* Dec*) defined by*

Gen *Run* Instance(1^λ) *to get* $(\mathcal{C}, \epsilon, \delta)$*. Output* $pk = evk = (\mathcal{C}, \epsilon)$ *and* $sk = (\mathcal{C}, \delta)$*,*
Enc *On input* pk, m*, return*

$$\mathsf{Op}(\mathcal{C}, \epsilon, 1, \mathsf{Epsilon}(pk, m), \mathsf{Sample}(pk)),$$

Dec *On input* sk, c*, return* $\mathsf{Delta}(\mathcal{C}, \delta, c) = \delta(c)$*,*
Eval *On input* $evk, *_i, c_1, \ldots, c_{a_i}$*, output*

$$\mathsf{Op}(\mathcal{C}, \epsilon, i, c_1, \ldots, c_{a_i}).$$

To see why this encryption makes sense, observe that we can induce the structure from $\mathcal{P}$ onto a subset of $\mathcal{C}$ through ϵ and δ. Define a equivalence relation $\sim$ on $\mathcal{C}$ by $c_1 \sim c_2$ if $\delta(c_1) = \delta(c_2)$. Each equivalence class now corresponds to a unique plaintext, so we can identify $\mathcal{P}$ with $\mathcal{C}/\!\sim$. For each m, we mark $\epsilon(m)$ as a distinguished representative of its class, creating the subset. Encryption is therefore to go to the corresponding equivalence class, and then using the operator $*'_1$ with something that decrypts to 0. That amounts to using $*_1$ with the neutral element in the second coordinate, i.e. nothing ("adding encryptions of zero"). The result will be in the same class, but randomly distributed.

Let c_0 be sampled from $\delta^{-1}(0)$ using Sample. Correctness is ensured since

$$\begin{aligned}\mathsf{Dec}(sk, \mathsf{Enc}(pk, m)) &= \delta(\mathsf{Op}(evk, 1, \mathsf{Epsilon}(pk, m), \mathsf{Sample}(pk))) \\ &= \delta(*'_1(\epsilon(m), c_0)) \\ &= *_1(\delta(\epsilon(m)), \delta(c_0)) = m *_1 0 = m.\end{aligned}$$

Evaluation is well-defined by the properties of the S-homomorphic tuple and the same arguments as above.

Theorem 5. *The Abstract Homomorphic Encryption Scheme is a homomorphic ∞-hop scheme. Any homomorphic ∞-hop scheme E over a given algebraic structure S can be expressed in terms of the above.*

Proof. The scheme is clearly fully homomorphic. Express any evaluation in terms of the operations on $\mathcal{P}$. Note that the evaluation and ciphertext spaces are the same, so it is ∞-hop.

Now for a given homomorphic ∞-hop scheme $(E.\mathsf{Gen}, E.\mathsf{Enc}, E.\mathsf{Eval}, E.\mathsf{Dec})$, we need to construct the algorithms for our abstract scheme. Define operations $*_1, \ldots, *_n$ on $\mathcal{P}$ based on the allowed computations, and identify the neutral element 0 with respect to $*_1$.

Instance Run $E.\mathsf{Gen}$ to get $E.\mathcal{C}$ and keys. Let a be a fixed randomness, and define $\epsilon(m) = E.\mathsf{Enc}(E.pk, m)$ using a, and $\delta(c) = E.\mathsf{Dec}(E.sk, c)$. Return $(\mathcal{C}, \epsilon, \delta)$.
Sample Compute a random encryption of 0.
Op Use $E.\mathsf{Eval}$.

The algorithms Epsilon and Delta follows from ϵ and δ.

The homomorphic properties of ϵ and δ are satisfied by definition, and we can use the construction above to create an instance of the AHES. □

The security is based on an assumption that all the equivalence classes are of about the same size. The following lemma ensures that this is the case in certain special cases.

Lemma 1. *If $*'_1$ is a bijection when the first element is fixed, and the whole of $\delta^{-1}(0)$ is samplable, then all equivalence classes as described above in the ciphertext space of the Abstract Homomorphic Encryption Scheme have the same cardinality.*

Proof. Let m be an arbitrary message, $c = \epsilon(m)$ and let c_0 be the element sampled from $\delta^{-1}(0)$ in the encryption algorithm. Recall that the encryption is then defined as

$$\mathsf{Op}(\mathcal{C}, \epsilon, 1, \mathsf{Epsilon}(pk, m), \mathsf{Sample}(pk)) = *'_1(c, c_0).$$

Then the restricted function $f_c : \delta^{-1}(0) \to \delta^{-1}(m)$ given by $f_c(x) = c *'_1 x$ is an injection (as it is a bijection on all of $\mathcal{C}$), so it is clear that $|\delta^{-1}(0)| \leq |\delta^{-1}(m)|$.

Now let m' be such that $m' *_1 m = 0$, which we know exists. Consider the images $A = f_c(\delta^{-1}(0))$, $B = \delta^{-1}(m)$ and $C = f_{\epsilon(m')}(B)$. By the homomorphic property, $f_{\epsilon(m')}$ maps elements of $\delta^{-1}(m)$ to $\delta^{-1}(0)$, so $C \subseteq \delta^{-1}(0)$.

Since $f_{\epsilon(m')}$ is an injection, then $|\delta^{-1}(m)| = |B| = |f_{\epsilon(m')}(B)| = |C| \leq |\delta^{-1}(0)|$, but then $|\delta^{-1}(0)| = |\delta^{-1}(m)|$ for any m. □

The scheme is semantically secure if it is hard to decide if an element is in a given equivalence class.

Definition 3 (Subset membership problem (SMP)). *Let $\mathcal{C}$ be an efficiently samplable set, and let R be a subset of $\mathcal{C}$. The challenger selects a random bit b. If $b = 0$, c is sampled uniformly from R, else from $\mathcal{C}$, and sent to the adversary. The adversary wins if he outputs the correct b.*

We modify the standard problem slightly. Replace the subset R with an efficient structure preserving mapping $\delta : \mathcal{C} \to \mathcal{P}$ as above. We now sample from the subset $\delta^{-1}(0)$.

The following theorem should not come as a surprise, as it is to a large extent a reformulation of IND-CPA security. However, its proof contains a transformation that can be used on any FHE scheme, and that should return a suitable problem to study. As the following examples will show, this enables us to quickly analyse schemes proposed in good faith, and may as such be a valuable tool.

Theorem 6. *Assume that the equivalence classes in $\mathcal{C}$ are of about the same size, that is, for all $m_1, m_2 \in \mathcal{P}$ the inequality $|1 - \frac{|\delta^{-1}(m_1)|}{|\delta^{-1}(m_2)|}| \leq \varepsilon$ holds where ε is negligible. The scheme is then secure if and only if the subset membership problem is hard.*

Proof. First we show that an adversary $\mathcal{A}_{\text{AHES}}$ with non-negligible advantage ε against a real-or-random game implies a distinguisher for the subset membership problem SMP. Our adversary $\mathcal{A}_{\text{SMP}}$ receives the challenge $(\mathcal{C}, \delta^{-1}, x)$ from SMP. We create an instance of AHES with $\mathcal{C}$ and δ^{-1} as a part of the public key. Note that we can construct ϵ by selecting an element from $\delta^{-1}(m)$ for each m. It is clear that $\delta \circ \epsilon = 1_{\mathcal{P}}$, and this will induce the required structure on the equivalence classes $\delta^{-1}(m)$ on $\mathcal{C}$. The homomorphic property holds since δ is defined to be structure preserving.

Transmit the public key. Upon receiving the plaintext m from the adversary, we encrypt it as $c \leftarrow \mathsf{Op}(evk, 1, \mathsf{Epsilon}(pk, m), x)$, and return the challenge ciphertext. When $\mathcal{A}_{\text{AHES}}$ returns a d with $d = 0$ if c is an encryption of m, we simply set $b' = d$ and replies with b' to the SMP instance.

If $b = 0$, then x is an encryption of 0, and the encryption is as usual. Otherwise, x can be interpreted as a random ciphertext, so c encrypts a random element, and the distribution is near uniform since the equivalence classes are of the approximately same size. It is clear that $\mathcal{A}_{\text{SMP}}$ has the same advantage against SMP as $\mathcal{A}_{\text{AHES}}$ has against Abstract Homomorphic Encryption Scheme.

For the opposite direction we again play a real-or-random game with a similar idea as AKP are using. Set up an Abstract Homomorphic Encryption Scheme instance (AHES), and send the public key to the adversary $\mathcal{A}_{\text{AHES}}$. He selects a message m. Note that the message will be invertible with respect to $*_1$. AHES responds with a ciphertext c, and $\mathcal{A}_{\text{AHES}}$ computes $x \leftarrow c *_1' \mathsf{Epsilon}(pk, m^{-1})$ and submits x to $\mathcal{A}_{\text{SMP}}$ along with $\mathcal{C}$ and ϵ. Upon receiving d from $\mathcal{A}_{\text{SMP}}$, $\mathcal{A}_{\text{AHES}}$ forwards $b' = d$ to the AHES instance.

If $b = 0$, then the ciphertext was real, which means that x encrypts 0, and hence selected from the inverse image of δ. Otherwise, c will be a random encryption, and it will just be shifted by $\mathcal{A}_{\text{AHES}}$, so it will still be a random

element selected from $\mathcal{C}$. This means that $\mathcal{A}_{\text{AHES}}$ will have the same advantage as $\mathcal{A}_{\text{SMP}}$. □

Remark 1. We can define the SOAP problem (*Splitting Oracle-Assisted Subgroup Membership Problem*) in the same way as in AKP [5], replacing "subgroup" with "subset". Informally, the SOAP problem is the same as the SMP problem, except that before receiving the challenge, the adversary has access to an oracle that on input c outputs $(\epsilon(m), c_0)$ where $c = \epsilon(m) *_1' c_0$, thus "splitting" a ciphertext into the distinguished representative of its equivalence class and its corresponding encryption of 0. Theorem 3 from AKP will then hold with only the obvious changes to the proof. To find out if the SOAP problem is hard outside a generic group remains an interesting problem.

We give the theorem with the few relevant words changed.

Theorem 7 (Characterization of IND-CCA Security, Theorem 3, [5]). *Let* $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Eval}, \mathsf{Dec})$ *be an* ∞*-hop fully homomorphic encryption scheme. Then:*

$$\mathcal{E} \textit{ is IND-CCA1 secure} \Leftrightarrow \textit{SOAP is hard}$$

4.1 Examples

We now apply our construction to a successful and two less successful scheme to prove the power of our approach. Our attacks spring directly from our characterisation. For the Nuida-Kurosawa scheme [29], it does not provide new insight, as the scheme is already quite close to this construction.

FHE over integers with non-binary space Nuida and Kurosawa have suggsted a simple scheme over the integers [29]. Let Q be a fixed prime, and let p and q_0 be large, secret primes such that $N = pq_0$ is hard to factor. A base encryption of m is of the form $m + pk + Qr$, and decryption is given by taking modulo p and then Q. Let x' be an encryption of 1 and $\{x_i\}$ be a set of encryptions of 0. The public key is $(N, x', \{x_i\})$. To encrypt using the public key, transform the message m from $\mathbb{Z}_Q$ into $\mathbb{Z}_N$ by multiplying with x', and add a random sum of encryptions of 0. See the original paper for details on the bootstrapping.

Note that this scheme by design is similar to our general description. The base problem that needs to be hard is therefore easily isolated, namely distinguishing the following two distributions,

$$\{m + pk + Qr \mid m < Q \ll p\} = \mathbb{Z}_{pq_0}$$

and

$$\{pk + Qr \mid Q \ll p\},$$

along with additional data available for bootstrapping. This is essentially the same problem studied by Cheon et al. [12].

Liu's scheme In May 2015, Liu published a candidate scheme on IACR's ePrint archive [25]. Although it was quickly proven insecure [33], it provides a series of valuable lessons. We refer to the original paper for details about the scheme.

The instance algorithms outputs $\mathcal{P} = \mathbb{Z}/q\mathbb{Z}$ as a field and $\mathcal{C} = (\mathbb{Z}/q\mathbb{Z})^{n+1}$ as a $\mathcal{P}$-algebra with ordinary addition and scalar multiplication, but with a key-dependent multiplication operation. Let $*_1$ be addition.

The public key is the tuple $(\Theta, \Phi, \{\mathsf{Enc}(sk_i sk_j)\}_{i,j=1,\dots n+1})$, and the private key is the vector $sk = (sk_1, \dots, sk_{n+1})$.

Define ϵ as encryption using 0 for all randomisers. Now, δ is just the linear mapping $\mathcal{P}^{n+1} \to \mathcal{P}$ given by $x \mapsto \langle sk, x \rangle$, and the problem is to decide whether the given element is in the kernel. It is clear that the kernel is an n-dimensional subspace of $\mathcal{P}^{n+1}$. Sample $m \geq n$ vectors from the subspace, and append x to form a matrix, and compute the rank. If $x \notin \delta^{-1}(0)$, then the rank will be $n+1$ with high probability, giving the adversary an advantage.

The scheme can therefore not be secure.

Li-Wang scheme Wang and Li proposed a new scheme based on multiplication of matrices over noncommutative rings [24]. As key, select a secret invertible matrix H and compute H^{-1}. To encrypt, place the message m in the top left corner of a upper triangular matrix M. The remaining places are filled with random values. Then compute the ciphertext as $C = HMH^{-1}$. Addition and multiplication works in the natural way.

The authors speculate that the scheme may be IND-CCA1 secure. However, we believe that it is insecure. Note that the scheme is symmetric, but the same reasoning as above implies that we only need to be able to distinguish encryptions of 0 from a random encryption.

Observe that the diagonal of M completely determines the invertibility of C, and that an encryption of 0 cannot be invertible. However, there is not an equivalence, since it can also be non-invertible if any element of the diagonal is a non-unit. To improve the probability, we can ask the encryption oracle for additional encryptions of 0, and add them to C, and checking the invertibility for each.

With high probability, one can then distinguish encryptions of units from encryptions of non-units, which is already sufficient to win a left-or-right game. The advantage against a real-or-random game will depend on the number of non-units in the ring. If the underlying ring is a division ring, then there are no other non-units than 0.

One can efficiently compute inverses by using a variant of LU decomposition suited for noncommutative rings.

Another tool for deciding invertibility, and which could give further information on the linear dependencies of the rows and columns, is the notion of quasideterminants, introduced by Gelfand and Retakh in 1991 [17]. In contrast to determinants for matrices over commutative rings is there not only one quasideterminant for an $n \times n$ matrix, but n^2. They may not always be computable, but provide useful information whenever they are.

Proposition ([18], Proposition 1.4.6)**.** *If the quasideterminant $|A|_{ij}$ is defined, then the following statements are equivalent.*

1. $|A|_{ij} = 0$
2. *the i-th row of the matrix A is a left linear combination of the other rows of A.*
3. *the j-th column of the matrix A is a right linear combination of the other columns of A*

The multiplicative identities for quasideterminants could also provide additional equations in order to perform a message recovery attack.

4.2 Can we do better?

The characterisation could still be more powerful by modelling noise-based encryption schemes in a precise way. Our work has not resulted in a concrete result for this case, but this section contains the arguments that have been tried. The main finding is that noise-based systems may depend on an embedding in some infinite space. This section can be skipped without any loss of continuity.

Remark 2 (Topology and metrics). This is a short introduction to the main tools in this section. We feel that a long introduction to these topics would interupt the reading too much, and rather refer to other sources for those who do not already feel comfortable with this branch of mathematics.

A topology is a selection of subsets that are defined to be open, and they – roughly speaking – generalise the open intervals from the real line and the interior of circles in the plane, that is, all points strictly within some distance from a set centre. Open sets in a topology can be used to measure concepts such as nearness and connectedness, although one could claim that the main objective is to study what continuous functions can do.

A metric is real-valued symmetric function which is positive for all pairs of points x and y unless $x = y$. Furthermore, we also require that the triangle inequality holds.

We are unable to model the concept of growing noise using our algebraic characterisation. Hence, it cannot be used to give a natural description of somewhat homomorphic schemes. This could be solved if we were able to describe metrics or perhaps a topology on the ciphertext space. Any metric will describe a topology, but the converse is not true. The most general approach is therefore to search for a topology, and hope that it has a corresponding metric.

The motivation for this thought is Gentry's construction of a metric in his dissertation [19]. Define the distance $\mathrm{d}(L, t)$ from the lattice L to a vector t by taking the minimum distance in $\mathbb{R}^n$ from a lattice vector to t. Each lattice vector is an encryption of 0, so we can measure how far a ciphertext is from being the canonical 0 encryption.

Forgetting Gentry's distance function, consider the following construction.

To equip $\mathcal{C}$ and $\mathcal{P}$ with a reasonable topology, we make a few assumptions. All operations should be continuous (this corresponds with the definition of topological groups and rings), and encryption and decryption should be continuous functions between the spaces. It is also reasonable to require that each equivalence class is an open set, i.e. having a neighbourhood around any ciphertext inside it. If encryption can reach the whole corresponding equivalence class, then $\mathcal{P}$ has the discrete topology, since the preimage of an open set must be open. The discrete topology hardly gives any information.

This also proves that $\mathcal{C}$ is disconnected. We get $|\mathcal{P}|$ clopen sets in $\mathcal{C}$, since the complement (a closed set) of an equivalence class is the union of the other classes, hence open. Then every class is both open and closed. For the moment being, we don't know anything more about the structure inside these sets.

For the sake of argument, assume that $\mathcal{C}$ is finite, as it is in classical group based constructions, or would be if we were working in some finite extension of $\mathbb{Z}/n\mathbb{Z}$.

It is a fact that a finite topological space is metrisable if and only if it is discrete. Hence, we can only measure how far a ciphertext is from its distinguished representative if we sacrifice all other topological structure on the space.

To relax the requirements, consider a pseudometric instead. A pseudometric does not require non-degeneracy, i.e. the distance between two points may be 0 even if they are different points. One can then define a pseudometric on $\mathcal{C}$ by $\mathrm{d}(x, y) = 1$ only if the points have neighbourhoods not contained in each other, otherwise 0. This, however, means that the distance between all points in the same equivalence classes have distance 0, so the topology will not give any information about the noise.

These arguments together seem to suggest that it is too restrictive to assume that $\mathcal{C}$ is finite. Further noise-based constructions of FHE schemes should therefore continue to focus on ciphertexts inside some infinite set. This is precisely what happens when existing schemes use rounding, as well as describing secret keys as rational numbers instead of inverses in a finite ring.

We ignore the topology for now, and restrict our attention to metric spaces. Assume that our scheme has a function f that can embed the noise into $\mathbb{R}^n$ for some n. In the case of ElGamal, this would be to output the discrete log of the first coordinate, whereas for the DGHV scheme, it could be to identify the small random noise, by reducing modulo the key and subtracting the plaintext. Generally, it is (part of) the random input used to sample the random encryption of 0.

Now define $\mathrm{d} : \mathcal{C} \times \mathcal{C} \to [0, \infty]$ as follows

$$\mathrm{d}(c_1, c_2) = \begin{cases} \infty & \text{if } \delta(c_1) \neq \delta(c_2) \\ \|f(c_1) - f(c_2)\| & \text{otherwise} \end{cases}$$

A metric that can also output infinite distances is sometimes called an extended metric, and does not alter any of the usual properties of ordinary metrics. As it is becoming more common, we will simply call it a metric. Notice that d

is a metric if any only if f is injective. That may not be the case, and then d is a pseudometric, which is sufficient, albeit not optimal.

One could circumvent this problem by stating that f should output all randomness that went info the ciphertext. Then it is injective. However, then we might lose our main goal, which is to measure the noise. In the case of the symmetric DGHV scheme, it would be a tuple with the coefficient to the key along with the small random noise. For the ordinary 2-norm, the first component would dominate the second, leaving the distance function useless.

For the special case of RLWE, we have the canonical embedding into $\mathbb{C}^n$ [27], in which one can compute distances properly.

5 Conclusion

Table 1. ×: No secure schemes. ?: No final conclusion. ✓: Examples of secure schemes exist.

Structure	Abelian groups	Vector spaces	Fields	Rings
Identical sets	✓	×	×	?
Non-trivially isomorphic	✓	×	×	?
Constant expansion	✓	×	×	?

We summarise the findings of Section 3 in Table 1. Clearly, we cannot give a definitive answer to Rivest et al.'s questions, but the trend seems to be that nice structures also make it easy to break schemes. This is unfortunate, as we would like to be able to perform our computations over nicely behaved structures. The contemporary solution to this is to allow a nice plaintext space – even fields – but compensate by having ciphertext spaces with very little structure. In particular, they are not closed under addition and multiplication until the expensive bootstrapping procedure is introduced.

Nuida [28] has proposed a framework for noise-free FHE schemes, but he also notes that the schemes look "somewhat 'artificial' ", and continues: "more 'natural' constructions of the underlying groups [...] would be more desirable [...]. Here we note, however, that such a natural instantiation of our schemes seems not easy to find." [28, p. 3]

We propose the following conjecture.

Conjecture. *The security of a fully homomorphic encryption scehme either depends on a massive ciphertext expansion or a weak or strange algebraic structure, limiting the applicability of the scheme.*

The results from Section 4 can contribute to the understanding of new schemes.

Acknowledgements The authors wish to thank Frederik Armknecht, Angela Jäschke, Colin Boyd, Christopher Carr and Christian Reuter for valuable discussion. Parts of this work was done while the second author visited Armknecht in Mannheim, and the work was partially financed by the Norwegian Research Council grant no. 233977 and the German Academic Exchange Service (DAAD), project number 57068907.

References

1. Niv Ahituv, Yeheskel Lapid, and Seev Neumann. Processing encrypted data. *Commun. ACM*, 30(9):777–780, 1987.
2. Frederik Armknecht, Colin Boyd, Christopher Carr, Kristian Gjøsteen, Angela Jäschke, Christian A. Reuter, and Martin Strand. A guide to fully homomorphic encryption. Cryptology ePrint Archive, Report 2015/1192, 2015. `http://eprint.iacr.org/`.
3. Frederik Armknecht, Tommaso Gagliardoni, Stefan Katzenbeisser, and Andreas Peter. General impossibility of group homomorphic encryption in the quantum world. In Hugo Krawczyk, editor, *Public-Key Cryptography - PKC 2014*, volume 8383 of *Lecture Notes in Computer Science*, pages 556–573. Springer, 2014.
4. Frederik Armknecht, Stefan Katzenbeisser, and Andreas Peter. Shift-type homomorphic encryption and its application to fully homomorphic encryption. In Aikaterini Mitrokotsa and Serge Vaudenay, editors, *Progress in Cryptology – AFRICACRYPT 2012*, volume 7374 of *Lecture Notes in Computer Science*, pages 234–251. Springer, 2012.
5. Frederik Armknecht, Stefan Katzenbeisser, and Andreas Peter. Group homomorphic encryption: characterizations, impossibility results, and applications. *Des. Codes Cryptography*, 67(2):209–232, 2013.
6. P. B. Bhattacharya, S. K. Jain, and S. R. Nagpaul. *Basic abstract algebra*. Cambridge University Press, Cambridge, second edition, 1994.
7. Dan Boneh and Richard J. Lipton. Algorithms for black-box fields and their application to cryptography (extended abstract). In Neal Koblitz, editor, *Advances in Cryptology - CRYPTO '96*, volume 1109 of *Lecture Notes in Computer Science*, pages 283–297. Springer, 1996.
8. Joppe W. Bos, Kristin E. Lauter, Jake Loftus, and Michael Naehrig. Improved security for a ring-based fully homomorphic encryption scheme. In Martijn Stam, editor, *Cryptography and Coding - 14th IMA International Conference, IMACC 2013, Oxford, UK, December 17-19, 2013. Proceedings*, volume 8308 of *Lecture Notes in Computer Science*, pages 45–64. Springer, 2013.
9. Zvika Brakerski. When homomorphism becomes a liability. In *TCC*, pages 143–161, 2013.
10. Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. Fully homomorphic encryption without bootstrapping. *Electronic Colloquium on Computational Complexity (ECCC)*, 18:111, 2011.
11. Murray R. Bremner. How to compute the wedderburn decomposition of a finite-dimensional associative algebra. *Groups Complexity Cryptology*, 3(1):47–66, 2011.
12. Jung Hee Cheon, Jean-Sébastien Coron, Jinsu Kim, Moon Sung Lee, Tancrède Lepoint, Mehdi Tibouchi, and Aaram Yun. Batch fully homomorphic encryption over the integers. In Thomas Johansson and Phong Q. Nguyen, editors, *Advances in Cryptology - EUROCRYPT 2013*, volume 7881 of *Lecture Notes in Computer Science*, pages 315–335. Springer, 2013.

13. Ana Costache and Nigel P. Smart. Which ring based somewhat homomorphic encryption scheme is best? In Kazue Sako, editor, *Topics in Cryptology - CT-RSA 2016*, volume 9610 of *Lecture Notes in Computer Science*, pages 325–340. Springer, 2016.
14. Yarkın Doröz, Yin Hu, and Berk Sunar. Homomorphic aes evaluation using the modified ltv scheme. *Designs, Codes and Cryptography*, pages 1–26, 2015.
15. Junfeng Fan and Frederik Vercauteren. Somewhat practical fully homomorphic encryption. *IACR Cryptology ePrint Archive*, 2012:144, 2012.
16. Katalin Friedl and Lajos Rónyai. Polynomial time solutions of some problems in computational algebra. In Robert Sedgewick, editor, *Proceedings of the 17th Annual ACM Symposium on Theory of Computing*, pages 153–162. ACM, 1985.
17. I.M. Gel'fand and V.S. Retakh. Determinants of matrices over noncommutative rings. *Functional Analysis and Its Applications*, 25(2):91–102, 1991.
18. Israel Gelfand, Sergei Gelfand, Vladimir Retakh, and Robert Lee Wilson. Quasideterminants. *Advances in Mathematics*, 193(1):56 – 141, 2005.
19. Craig Gentry. *A fully homomorphic encryption scheme*. PhD thesis, Stanford University, 2009. `crypto.stanford.edu/craig`.
20. Craig Gentry. Computing on the edge of chaos: Structure and randomness in encrypted computation. *Electronic Colloquium on Computational Complexity (ECCC)*, 21:106, 2014.
21. Craig Gentry, Amit Sahai, and Brent Waters. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In Ran Canetti and Juan A. Garay, editors, *Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I*, volume 8042 of *Lecture Notes in Computer Science*, pages 75–92. Springer, 2013.
22. Christopher J. Hillar and Darren L. Rhea. Automorphisms of finite abelian groups. *The American Mathematical Monthly*, 114(10):917–923, 2007.
23. H. W. Lenstra, Jr. Finding isomorphisms between finite fields. *Math. Comp.*, 56(193):329–347, 1991.
24. Jing Li and Licheng Wang. Noise-free symmetric fully homomorphic encryption based on noncommutative rings. Cryptology ePrint Archive, Report 2015/641, 2015. `http://eprint.iacr.org/`.
25. Dongxi Liu. Practical fully homomorphic encryption without noise reduction. Cryptology ePrint Archive, Report 2015/468, 2015. `http://eprint.iacr.org/`.
26. Adriana López-Alt, Eran Tromer, and Vinod Vaikuntanathan. On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption. In Howard J. Karloff and Toniann Pitassi, editors, *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, New York, NY, USA, May 19 - 22, 2012*, pages 1219–1234. ACM, 2012.
27. Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. *J. ACM*, 60(6):43, 2013.
28. Koji Nuida. Candidate constructions of fully homomorphic encryption on finite simple groups without ciphertext noise. Cryptology ePrint Archive, Report 2014/097, 2014. `http://eprint.iacr.org/`.
29. Koji Nuida and Kaoru Kurosawa. (Batch) fully homomorphic encryption over integers for non-binary message spaces. In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015*, volume 9056 of *Lecture Notes in Computer Science*, pages 537–555. Springer, 2015.

30. Stephen C. Pohlig and Martin E. Hellman. An improved algorithm for computing logarithms over gf(p) and its cryptographic significance (corresp.). *IEEE Transactions on Information Theory*, 24(1):106–110, 1978.
31. Michael O. Rabin. How to exchange secrets with oblivious transfer. *IACR Cryptology ePrint Archive*, 2005:187, 2005.
32. Ronald Rivest, Leonard Adleman, and Michael Dertouzos. On data banks and privacy homomorphisms. *Foundations of Secure Computation, Academia Press*, pages 169–179, 1978.
33. Yongge Wang. Notes on two fully homomorphic encryption schemes without bootstrapping. Cryptology ePrint Archive, Report 2015/519, 2015. `http://eprint.iacr.org/`.
34. Yu Yu, Jussipekka Leiwo, and A. Benjamin Premkumar. A study on the security of privacy homomorphism. *I. J. Network Security*, 6(1):33–39, 2008.