FULLY HOMOMORPHIC ENCRYPTION FOR MATHEMATICIANS

ALICE SILVERBERG

Abstract. We give an introduction to Fully Homomorphic Encryption for mathematicians. Fully Homomorphic Encryption allows untrusted parties to take encrypted data $\mathrm{Enc}(m_1), \ldots, \mathrm{Enc}(m_t)$ and any efficiently computable function f, and compute an encryption of $f(m_1, \ldots, m_t)$, without knowing or learning the decryption key or the raw data $m_1, \ldots, m_t$. The problem of how to do this was recently solved by Craig Gentry, using ideas from algebraic number theory and the geometry of numbers. In this paper we discuss some of the history and background, give examples of Fully Homomorphic Encryption schemes, and discuss the hard mathematical problems on which the cryptographic security is based.

1. Introduction

Fully Homomorphic Encryption (FHE) has been referred to as a "holy grail" of cryptography. Craig Gentry's recent solution to the problem, while not efficient enough to be practical, was considered to be a major breakthrough. Since then, much progress has been made in the direction of finding efficient Fully Homomorphic Encryption schemes.

In this paper we will give a brief introduction to FHE for mathematicians. We will give some of the history and major ideas, we will present some examples of FHE schemes, and we will mention a variety of security assumptions on which FHE schemes have been based. The intended audience is mathematicians at the graduate level or beyond (especially number theorists) who do not necessarily have any background in cryptography. The paper is mostly a survey, though §4.3 gives a number theory proof that does not seem to be in the cryptography literature.

In encryption schemes, Bob encrypts a plaintext message to obtain a ciphertext. Alice decrypts the ciphertext to recover the plaintext. In Fully Homomorphic Encryption, parties that do not know the plaintext data can perform computations on it by performing computations on the corresponding ciphertexts.

A major application of FHE is to cloud computing. Alice can store her data in "the cloud", e.g., on remote servers that she accesses via the Internet. The cloud has more storage capabilities and computing power than does Alice, so when Alice needs computations to be done on her data, she would like those computations to be done by the cloud. However, Alice doesn't trust the cloud. Her data might be sensitive (for example, Alice might be a hospital and the data might be patients' medical records), and Alice would like the cloud to know as little as possible about her data, and about the results of the computations. So Alice sends encrypted data to the cloud, which can perform arithmetic operations on it without learning anything about the original raw data, by performing operations on the encrypted data.

This material is based on research sponsored by DARPA under agreement numbers FA8750-11-1-0248 and FA8750-13-2-0054. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the author and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA or the U.S. Government.

The work was also supported by the National Science Foundation under grant CNS-0831004.

Thanks go to Hendrik Lenstra for helpful conversations about Fully Homomorphic Encryption, and to Lily Khadjavi, Zvika Brakerski, and Chris Peikert for very helpful comments on earlier versions of the paper.

Fully Homomorphic Encryption can be used to query a search engine, without revealing what is being searched for (here, the search engine is doing the computations on encryptions of information that it doesn't know).

More precisely, FHE has the following property. Say that ciphertexts c_i decrypt to plaintexts m_i, i.e., $\text{Decrypt}(c_i) = m_i$, where the m_i's and c_i's are elements of some ring (with two operations, addition and multiplication). In FHE one has

$$\text{Decrypt}(c_1 + c_2) = m_1 + m_2, \qquad \text{Decrypt}(c_1 \cdot c_2) = m_1 \cdot m_2.$$

In other words, decryption is doubly homomorphic, i.e., homomorphic with respect to the two operations addition and multiplication.

Being fully homomorphic means that whenever f is a function composed of (finitely many) additions and multiplications in the ring, then

$$\text{Decrypt}(f(c_1, \ldots, c_t)) = f(m_1, \ldots, m_t).$$

If the cloud (or an adversary) can efficiently compute $f(c_1, \ldots, c_t)$ from ciphertexts $c_1, \ldots, c_t$, without learning any information about the corresponding plaintexts $m_1, \ldots, m_t$, then the system is efficient and secure.

Another requirement for FHE is that the ciphertext sizes remain bounded, independent of the function f; this is known as the "compact ciphertexts" requirement.

Fully Homomorphic Encryption schemes can be either public key (where the encryptor knows the decryptor's public key but not her private key) or symmetric key (where the encryptor and decryptor share a key that is used for both encryption and decryption).

In Section 2 we briefly give some history and background. In Sections 3, 4, and 5 we give some (somewhat) homomorphic encryption schemes, to illustrate a variety of techniques and security assumptions.

As usual, $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$, and $\mathbb{C}$ denote the integers, rational numbers, real numbers, and complex numbers, respectively, and $\mathbb{F}_q$ denotes the finite field with q elements.

2. Some history and background

2.1. **Early history.** In 1978, shortly after the invention of the RSA Cryptosystem, Rivest, Adleman, and Dertouzos [RAD] came up with the idea of fully homomorphic encryption, which they called "privacy homomorphisms". Their paper states, "although there are some truly inherent limitations on what can be accomplished, we shall see that it appears likely that there exist encryption functions which permit encrypted data to be operated on without preliminary decryption of the operands, for many sets of interesting operations. These special encryption functions we call 'privacy homomorphisms'; they form an interesting subset of arbitrary encryption schemes". Despite the optimism of Rivest, Adleman, and Dertouzos, fully homomorphic encryption remained out of reach for many years.

A number of cryptosystems are homomorphic with respect to one operation. For example, RSA and ElGamal encryption are homomorphic with respect to multiplication.

We recall that in (basic) RSA, Alice's public key is (N, e) and private key is d, where N is a product of two large primes and where $de \equiv 1 \bmod \varphi(N)$. If $m \in \mathbb{Z}/N\mathbb{Z}$ is the plaintext, then the ciphertext is $c = m^e \bmod N$. To decrypt, Alice computes $c^d \bmod N = m$. If Bob encrypts messages m_1 and m_2 using Alice's public key (N, e), then the product of the resulting ciphertexts is the ciphertext of the product of the plaintexts m_1 and m_2, i.e., $(m_1^e \bmod N)(m_2^e \bmod N) = (m_1 m_2)^e \bmod N$. Thus, $\text{Decrypt}(c_1 \cdot c_2) = \text{Decrypt}(c_1) \cdot \text{Decrypt}(c_2)$, where $c_i = m_i^e \bmod N$ is the ciphertext corresponding to the plaintext m_i.

For ElGamal, suppose the private key is $x \in \{1, \ldots, n-1\}$ and the public key is $h = g^x \in G$, where G is a cyclic group of order n generated by g. If $m_1, m_2 \in G$ are plaintext messages, then the corresponding ciphertexts are of the form $c_i = (a_i, b_i) = (g^{r_i}, m_i h^{r_i}) \in G \times G$ for $i = 1$ and 2, where the r_i are chosen by the encryptor(s) at random in $\{1, \dots, n-1\}$. Then

$$\begin{aligned}\text{Decrypt}(c_1 \cdot c_2) = \text{Decrypt}(a_1 a_2, b_1 b_2) &= ((a_1 a_2)^x)^{-1} b_1 b_2 \\ &= (a_1^x)^{-1} b_1 \cdot (a_2^x)^{-1} b_2 = \text{Decrypt}(c_1) \cdot \text{Decrypt}(c_2).\end{aligned}$$

There have been other encryption schemes with homomorphic properties. For example, the Goldwasser-Micali cryptosystem [GM] and its generalization the Paillier cryptosystem [Pa] are homomorphic with respect to addition of plaintexts in the sense that

$$\text{Decrypt}(c_1 \cdot c_2) = m_1 + m_2,$$

but are not homomorphic with respect to multiplication of plaintexts.

In [BoGN], Boneh, Goh, and Nissim gave a partially homomorphic encryption scheme that can do one multiplication and any number of additions.

2.2. **Gentry's FHE scheme and beyond.** Craig Gentry solved the problem of how to do Fully Homomorphic Encryption in his Stanford PhD thesis [G1, G2, G3]. For the first time, there was now a scheme that could (inefficiently) do an arbitrary number of additions and multiplications.

Gentry's solution used ideal lattices, i.e., ideals in algebraic number fields. Given that one requires a homomorphic property with respect to two operations, it is natural that rings come into play. In [G1] and [G2], the rings Gentry used were of the form

$$R := \mathbb{Z}[x]/\langle x^N + 1\rangle \quad \text{and} \quad R_d := (\mathbb{Z}/d\mathbb{Z})[x]/\langle x^N + 1\rangle$$

where $N = 2^n$ (see §4 below). It was later realized that one can use the rings $\mathbb{Z}$ and $\mathbb{Z}/d\mathbb{Z}$ to construct schemes parallel to those that use the rings R and R_d (see §3 below).

There have been a number of improvements, implementations, and new schemes. See for example [SmV, DGHV, G4, SS, GH1, LaNV, GH2, BV2, BV1, CorMNT, LMSV, BrGV, GHS1, GHS2, CorNT, Br]. The NTRU encryption scheme [HofPS], which was developed in the late 1990's, turned out to be "somewhat homomorphic", and has been turned into an FHE scheme [LTV].

2.3. **Security.** The primary known attacks on FHE schemes are variants of the LLL lattice basis reduction algorithm [LLL]. The security of almost all currently known schemes is based on the presumed difficulty of some lattice problem, such as finding an approximately shortest (non-zero) vector in a high dimensional lattice.

A number of FHE schemes use ideal lattices rather than arbitrary lattices. These are very special lattices, and it might turn out to be the case that lattice attacks are easier for ideal lattices than for generic lattices. This is an open question. At the moment, special attacks that work better for ideal lattices than for general lattices are not yet known.

2.4. **Somewhat Homomorphic Encryption (SHE).** Somewhat Homomorphic Encryption (SHE) schemes are encryption schemes that have some homomorphic properties but are not fully homomorphic. With Somewhat Homomorphic Encryption one can generally do a limited number of additions and multiplications, but each time one does an operation, it contributes "noise" to the ciphertext (see §3 for an example). Eventually the noise is so great that it is not possible to decrypt. Also, in SHE schemes the ciphertexts could get larger (message expansion), i.e., the compact ciphertexts requirement might be violated. In Gentry's initial work he started with an SHE scheme and then "bootstrapped" it to obtain an FHE scheme.

2.5. **Bootstrapping.** Gentry's original FHE papers and thesis introduced the idea of bootstrapping. One "bootstraps" to go from a (bootstrapable) somewhat homomorphic encryption scheme to a fully homomorphic encryption scheme.

To make an SHE scheme fully homomorphic, one can include as part of the public key an encryption of the private key. When a ciphertext gets too large or too noisy, the encryptor can then use the somewhat homomorphic encryption scheme to evaluate the decryption function applied to the ciphertext, using the encrypted private key. This re-encryption process produces a new encryption of the original plaintext, that is more compact and less noisy. For this to work, it is necessary for the somewhat homomorphic scheme to be "circular secure" (i.e., it must be able to securely encrypt its own private key) and capable of evaluating the function $f = \text{Decrypt}$ and "a little more" (enough to allow homomorphic encryptions with respect to addition and multiplication; see the "augmented decryption circuits" in Definition 4 of [G1] or [DGHV]).

Gentry also uses what he calls "squashing" of the decryption circuit in order to simplify decryption enough so that it is among the functions that the somewhat homomorphic scheme can homomorphically evaluate correctly. Squashing converts an SHE scheme into a bootstrappable SHE scheme. In [BV2], Brakerski and Vaikuntanathan use "dimension-modulus reduction" to simplify the decryption circuit and avoid squashing. Another way to remove squashing is given in [GH2].

In [BrGV], Brakerski, Gentry, and Vaikuntanathan use "modulus switching" to reduce noise and lessen the need for bootstrapping. Modulus switching replaces a ciphertext mod p_1 with a ciphertext modulo a smaller modulus p_2 that decrypts to the same plaintext.

See [G3] for a nice analogy ("Alice's jewelry store", with jewelry fabricated in nested secure gloveboxes) that gives the idea of FHE and bootstrapping. See also [H] for a good explanation of FHE for a general audience. See Vaikuntanathan's survey article [V] for a good description of modulus switching and other concepts from FHE.

2.6. **Malleability.** We remark that FHE schemes are always "malleable". In cryptography, malleability means that a ciphertext can be perturbed to create a new ciphertext that decrypts to a perturbation (in a known way) of the original plaintext. In a non-malleable encryption scheme, perturbing a ciphertext a little will generally produce an invalid ciphertext, i.e., one that does not decrypt to a valid plaintext. Malleability is often an undesirable property in cryptography. For example, if an auction uses encrypted bids, and (an adversary) Mallory sees the encryption of Bob's bid, one wants it to be the case that Mallory cannot construct a new ciphertext that decrypts to a bid that is one more than Bob's bid, i.e., one wants non-malleable encrypted bids.

There has been some work on obtaining partial or "targeted" non-malleability along with some limited homomorphic ability; see for example [PR, BoSW]. There are interesting open questions in this area.

3. Somewhat Homomorphic Encryption over the integers

We begin with a warm-up example from the introduction to [DGHV]. This example of a somewhat homomorphic encryption scheme comes in two flavors, symmetric key and public key. To keep it short, we will be very imprecise about parameter choices and other details.

We first give the symmetric key version. The shared key is an odd positive integer k. The message is a bit $m \in \{0, 1\}$. The encryptor chooses random integers q and r in a certain range, and so that $|2r| < k/2$, and computes the ciphertext

$$c = m + kq + 2r.$$

To decrypt, the decryptor computes $(c \bmod k) \bmod 2 = m$ where $a \bmod w$ means that one takes the representative of $a \bmod w$ in the range $(-w/2, w/2]$.

If $c_i = m_i + kq_i + 2r_i$ for $i = 1, 2$, then

$$c_1 + c_2 = (m_1 + m_2) + k(q_1 + q_2) + 2(r_1 + r_2),$$

$$c_1 \cdot c_2 = m_1 \cdot m_2 + k(m_1q_2 + m_2q_1 + kq_1q_2 + 2q_1r_2 + 2r_1q_2) + 2(m_1r_2 + r_1m_2 + 4r_1r_2).$$

Thus the noise grows, and after one does too many multiplications or additions, the decryption function no longer outputs the correct plaintext. The ciphertexts also blow up in size. This Somewhat Homomorphic Encryption scheme is not fully homomorphic, but in [DGHV] van Dijk et al. use Gentry's bootstrapping techniques to turn it into a Fully Homomorphic Encryption scheme.

A public key version, as in §3.1 of [DGHV], is as follows. The secret key is again an odd positive integer k. The public key now consists of the integers $x_i = kq_i + 2r_i$ for $i = 0, 1, \ldots, t$, where the q_i and r_i are as before, so each x_i can be viewed as an encryption of 0 under the symmetric key scheme. The x_i are taken so that x_0 is the largest, x_0 is odd, and $x_0 \bmod k$ is even, where again $x \bmod k$ is in the interval $(-k/2, k/2]$.

To encrypt a message bit $m \in \{0, 1\}$, the encryptor chooses a random subset S of $\{1, \ldots, t\}$ and a random integer r in a certain range. The ciphertext is

$$c = m + 2\sum_{i \in S} x_i + 2r \mod x_0.$$

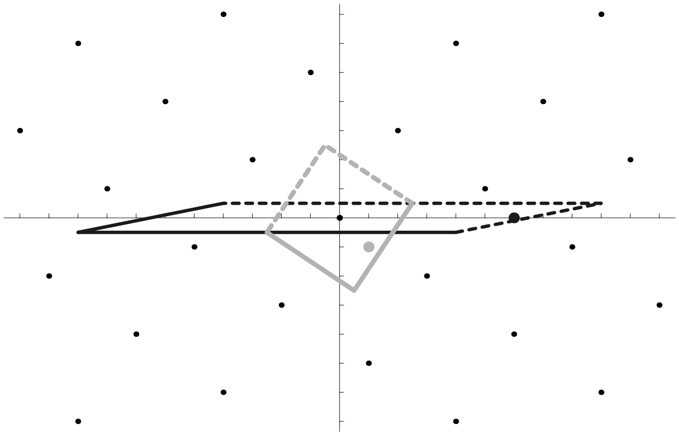
The decryptor computes $(c \bmod k) \bmod 2 = m$.

The security is based on the difficulty of the Approximate Common Divisor Problem, which is the problem of finding k, given a collection of integers of the form $\{kq_i + r_i\}_{i=0}^{t}$ with r_i "small". Approximate Common Divisor Problems were introduced in [How] and have been studied in [CN, CoH].

4. The Gentry, Smart-Vercauteren, and Gentry-Halevi SHE schemes

We next give a version of the Somewhat Homomorphic Encryption schemes that were introduced by Gentry in [G1, G2] and improved on by Smart and Vercauteren in [SmV] and by Gentry and Halevi in [GH1] (see also [LMSV]). In these schemes, the public key corresponds to a "bad" (skewed) basis for a lattice, while the private key is a "good" (more orthogonal) basis for the same lattice. The (N-dimensional) lattices are ideals in the ring of integers of the cyclotomic field of $2N$-th roots of unity. The plaintext is encoded as a (suitable) point in the ambient space $\mathbb{R}^N$. Encryption translates that point into the fundamental parallelepiped associated to the bad (public) basis. Decryption translates the ciphertext point into the fundamental parallelepiped associated to the good (private) basis. (See Figure 1 and the description near the end of §4.1.) The security relies partly on the fact that it is difficult to find a good, nearly orthogonal basis for a given lattice.

Figure 1. Encryption and Decryption



4.1. **The scheme.** We next give some of the details of a version of the scheme. Let

$$F(x) = x^N + 1 \in \mathbb{Z}[x]$$

with $N = 2^n$. Let θ be a root of $F(x)$; then θ is a primitive $2N$-th root of unity. Let

$$K = \mathbb{Q}[x]/\langle F(x)\rangle \cong \mathbb{Q}(\theta),$$

a CM-field of degree N over $\mathbb{Q}$. Let

$$v(x) = \sum_{i=0}^{N-1} v_i x^i \in \mathbb{Z}[x]$$

be a degree $N-1$ polynomial whose coefficients v_i are random t-bit integers for a suitably chosen t, and

$$V := \begin{pmatrix} v_0 & v_1 & \cdots & v_{N-1} \\ -v_{N-1} & v_0 & \cdots & v_{N-2} \\ & & \cdots & \\ -v_1 & -v_2 & \cdots & v_0 \end{pmatrix} \in \mathrm{M}_N(\mathbb{Z}).$$

The rows of V are the coefficients of $x^i v(x)$ mod $F(x)$ for $i = 0, \ldots, N-1$. Let L denote the lattice in $\mathbb{Z}^N$ generated by the rows of V, let $\gamma = v(\theta) \in K$, let $\mathrm{N}_{K/\mathbb{Q}} : K \to \mathbb{Q}$ denote the norm map, and let

$$d := \mathrm{N}_{K/\mathbb{Q}}(\gamma) = \det(V) = \det(L) = \mathrm{resultant}(F, v).$$

Replace the random polynomial $v(x)$ if necessary, until you have found one for which d is odd and square-free. (In [SmV], they start with $v(x) \equiv 1 \bmod 2\mathbb{Z}[x]$ to ensure that d is odd, and they replace $v(x)$, if necessary, until they find one for which d is prime. In [GH1] they show that it is not necessary for d to be prime; it suffices to have d odd and square-free.)

Whenever A is a matrix whose rows $\{\mathbf{a}_1, \ldots, \mathbf{a}_N\}$ form a $\mathbb{Z}$-basis for a lattice $L \subset \mathbb{R}^N$, define

$$\mathcal{P}(A) := \{\sum_{i=1}^{N} \alpha_i \mathbf{a}_i : \alpha_i \in [-0.5, 0.5)\},$$

a (half-open) parallelepiped. This is the "fundamental parallelepiped" associated to A. Every element of $\mathbb{R}^N/L$ has a unique representative in $\mathcal{P}(A)$.

All reductions mod d will be taken in the range $[-d/2, d/2)$. Let $r \in [-d/2, d/2)$ denote the unique common root of $F(x)$ and $v(x)$ mod d. Let $r_i = r^i \pmod{d}$ and let

$$B := \begin{pmatrix} d & 0 & 0 & \cdots & 0 \\ -r_1 & 1 & 0 & \cdots & 0 \\ & \cdots & & & \\ -r_{N-1} & 0 & 0 & \cdots & 1 \end{pmatrix} \in \mathrm{M}_N(\mathbb{Z}).$$

Since d is odd and square-free, it follows that B is the Hermite Normal Form of the matrix V.

The public key now consists of d and r (or equivalently the matrix B), and the secret key is $v(x)$ (or the matrix V). To encrypt a bit $m \in \{0, 1\}$, choose a random noise polynomial $u(x) = \sum_{i=0}^{N-1} u_i x^i$ with each coefficient $u_i \in \{0, \pm 1\}$ taking values 1 and -1 with equal probability. Let $a(x) = m + 2u(x)$ and let

$$\mathbf{a} := (2u_0 + m, 2u_1, \ldots, 2u_{N-1})$$

be the vector of coefficients of $a(x)$. Let $\lceil \cdot \rfloor$ denote rounding to the nearest integer.

Let the ciphertext be

$$\mathbf{c} := \mathbf{a} - (\lceil \mathbf{a}B^{-1} \rfloor B) = (m + 2u(r) \mod d, 0, \ldots, 0),$$

which is the translation of $\mathbf{a}$ to the parallelepiped $\mathcal{P}(B)$ (where translation means that one subtracts lattice vectors until one lands in the fundamental parallelepiped).

To decrypt a ciphertext $\mathbf{c}$, let

$$\mathbf{a}_1 := \mathbf{c} - (\lceil \mathbf{c}V^{-1} \rfloor V) = (a_0, \ldots, a_{N-1}),$$

which is the translation of $\mathbf{c}$ to the parallelepiped $\mathcal{P}(V)$, and compute $m = a_0 \pmod 2$. As shown on p. 145 of [GH1], decryption works (i.e., $\mathbf{a}_1 = \mathbf{a}$) as long as the absolute value of every entry in $\mathbf{a}V^{-1}$ is less than $\frac{1}{2}$.

In Figure 1, the small dots are the lattice. The light gray point represents the plaintext, the (inside of the) light gray diamond represents the fundamental parallelepiped $\mathcal{P}(V)$, the (inside of the) dark parallelogram represents the fundamental parallelepiped $\mathcal{P}(B)$, and the large dark point, which is the ciphertext, is the translation to $\mathcal{P}(B)$ of the light gray point.

The rows of the matrix B are a "bad", i.e., skewed basis for the lattice L, while the rows of V are a "good" (secret) basis for L. If the rows of V are sufficiently orthogonal, and if the plaintext point is chosen in a suitable way, then decryption yields the original plaintext point.

The scheme is homomorphic because its multiplication and addition are just multiplication and addition in the ring of integers of the cyclotomic field K.

4.2. **Security.** The security of the above scheme is based on the simultaneous difficulty of the following problems.

The **Small Principal Ideal Problem (SPIP)** is the problem, given a principal ideal in either Hermite Normal Form (i.e., the matrix B) or two element representation (i.e., $\langle d, \theta - r\rangle$), of finding a "small" generator (e.g., $v(\theta)$) for it. If the SPIP is sufficiently hard, that would thwart a key recovery attack, wherein an adversary who knows the public key (B or (d, r)) tries to find the secret key ($v(x)$).

Security against an attack where the adversary tries to find the plaintext, given a ciphertext, is closely related to the difficulty of the **Closest Vector Problem (CVP)** for ideal lattices. This is the problem of finding a closest lattice point to a given point in the ambient space.

Another type of security is "semantic security". The requirement for semantic security is that an adversary, who is presented with a ciphertext that is either an encryption of 0 or an encryption of 1, cannot distinguish which it is with probability greater than $\frac{1}{2} + \varepsilon$ of getting the correct answer. The semantic security of the scheme is related to a new problem, that Smart and Vercauteren call the **Polynomial Coset Problem (PCP)**. The Polynomial Coset Problem is the problem of distinguishing between a random element of $\mathbb{Z}/d\mathbb{Z}$ and an element of the form $f(r) \bmod d$, where $f(x) \in \mathbb{Z}[x]$ is random (and unknown) with small coefficients and r is the common root of $F(x)$ and $v(x) \bmod d$. The paper [SmV] states that the Polynomial Coset Problem is akin to Gentry's Ideal Coset Problem from [G1]. These problems can be viewed as versions of the Bounded Distance Decoding problem from coding theory.

Gentry, Smart-Vercauteren and Gentry-Halevi "bootstrap" their somewhat homomorphic encryption schemes into fully homomorphic encryption schemes using a re-encryption algorithm. Making this cryptographically secure requires an additional security assumption, namely the difficulty of a decisional version of the **Sparse Subset-Sum Problem (SSSP)**, i.e., it should be difficult to distinguish between random subsets of $\mathbb{Z}/d\mathbb{Z}$ and those that have sparse subsets that sum to 0. Here, bootstrapping augments the public key with a "hint" about the secret key, namely, with a large set of vectors that has a very sparse subset that sums to the secret key.

4.3. **Why F and v have exactly one common root mod d.** Since it is not in the FHE literature, we give a proof that $F(x)$ and $v(x)$ have a unique common root mod d. This shows the use of some algebraic number theory in FHE. The next result allows for a more general polynomial $F(x)$.

Lemma 1. *Suppose $F(x), v(x) \in \mathbb{Z}[x]$. Suppose that $F(x)$ is monic and irreducible, and $\theta \in \bar{\mathbb{Q}}$ is a root of F. Let $K = \mathbb{Q}[x]/\langle F(x)\rangle \cong \mathbb{Q}(\theta)$ and suppose $K/\mathbb{Q}$ is a Galois extension. Let $\gamma =$*

$v(\theta)$ and suppose that $\mathrm{N}_{K/\mathbb{Q}}(\gamma)$ is square-free and relatively prime to the discriminant of K. Then $F(x) \bmod \langle\gamma\rangle$ and $v(x) \bmod \langle\gamma\rangle$ have exactly one common root in $\mathcal{O}_K/\langle\gamma\rangle$, namely $\theta \bmod \langle\gamma\rangle$.

Proof. Since $v(\theta) = \gamma$ and $F(\theta) = 0$ both map to 0 under the projection map $\mathcal{O}_K \to \mathcal{O}_K/\langle\gamma\rangle$, it follows that θ is a common root of $F(x) \bmod \langle\gamma\rangle$ and $v(x) \bmod \langle\gamma\rangle$. Since $K/\mathbb{Q}$ is Galois, $F(x)$ splits completely in $K[x]$, so the reductions mod $\langle\gamma\rangle$ of the roots of $F(x)$ are the roots of $F(x) \bmod \langle\gamma\rangle$. Thus any other common root is the reduction mod $\langle\gamma\rangle$ of a root of $F(x)$, so it is $\sigma(\theta)$ for some non-identity $\sigma \in \mathrm{Gal}(K/\mathbb{Q})$. But $v(\sigma(\theta)) = \sigma(v(\theta)) = \sigma(\gamma)$, which cannot be 0 mod $\langle\gamma\rangle$, since $\gcd(\sigma(\gamma), \gamma) = 1$, as follows.

Factor $\gamma\mathcal{O}_K = \prod_i \mathfrak{p}_i$ with prime ideals $\mathfrak{p}_i$ of $\mathcal{O}_K$. Since $\mathrm{N}_{K/\mathbb{Q}}(\gamma)$ is square-free and relatively prime to the discriminant of K, it follows that:

(a) each $\mathfrak{p}_i$ has degree one (i.e., its norm is a prime in $\mathbb{Z}$),
(b) the different $\mathfrak{p}_i$'s have distinct residue characteristics, and
(c) $\sigma(\mathfrak{p}_i) \neq \mathfrak{p}_j$ for all i and j.

To obtain (c), note that if $\sigma(\mathfrak{p}_i) = \mathfrak{p}_i$, then σ would be in the decomposition group for $\mathfrak{p}_i$, whose order is the degree of $\mathfrak{p}_i$, which is 1 by (a). Part (c) now follows from (b). Since $\sigma(\gamma)\mathcal{O}_K = \prod_i \sigma(\mathfrak{p}_i)$, it now follows that $\gcd(\sigma(\gamma), \gamma) = 1$. □

5. LWE and Ring-LWE

A promising recent development is to create Fully Homomorphic Encryption schemes whose security is based on the difficulty of the LWE Problem (introduced in [R]) or the Ring-LWE Problem (introduced in [LyPR]). These FHE schemes are more efficient than earlier schemes, with short ciphertexts.

LWE stands for Learning with Errors. A version of the LWE problem is as follows. If F is a field and $v = (v_1, \ldots, v_n), w = (w_1, \ldots, w_n) \in F^n$, let $\langle v, w\rangle$ denote the usual inner product $\sum_{j=1}^n v_j w_j$. Take p prime, of size polynomial in a parameter n. For uniformly random $a_i \in \mathbb{F}_p^n$, and "noise" $e_i \in \mathbb{Z}$ chosen via a probability distribution (usually Gaussian) that outputs e_i with $|e_i|$ much smaller than p, given polynomially (in n) many pairs $(a_i, b_i = \langle a_i, s\rangle + e_i \mod p)$, find $s \in \mathbb{F}_p^n$. Here, the e_i's are the errors, and the problem is to learn the secret s, even in the presence of errors. If there are no errors, i.e., all $e_i = 0$, then one can easily recover s using linear algebra, given enough pairs (a_i, b_i). When $p = 2$ the Learning with Errors Problem is known as the Learning Parity with Noise Problem.

In the decisional version one needs to distinguish such ordered pairs (a_i, b_i) from uniformly random pairs $(a_i, u_i) \in \mathbb{F}_p^n \times \mathbb{F}_p$. By [R, Pe], this problem is at least as hard as (variants of) the problem of finding short vectors in lattices.

Next, following [BV2], we give a simplification of a symmetric key somewhat homomorphic encryption scheme whose security is based on the decisional version of LWE. The secret key is a random $s \in \mathbb{F}_p^n$. To encrypt a plaintext bit $m \in \{0, 1\}$, choose a random $a \in \mathbb{F}_p^n$ and a "noise" e. Compute $b := \langle a, s\rangle + 2e + m \in \mathbb{F}_p$. The ciphertext is $(a, b) \in \mathbb{F}_p^n \times \mathbb{F}_p$. To decrypt, compute $b - \langle a, s\rangle \equiv 2e + m \pmod{p}$ and reduce mod p to get $2e + m$ (since $|e| \ll p$). Now reduce mod 2 to obtain m (note that the "masking" terms $\langle a, s\rangle$ and $2e$ do not interfere with each other since p is odd). The scheme is homomorphic with respect to addition, and can do one homomorphic multiplication but with a large ciphertext expansion. In [BV2] it is shown how to turn this into a fully homomorphic encryption scheme (without the need for squashing).

In Ring-LWE, R is a ring. The Ring-LWE problem is to find s, given polynomially many $(a_i, b_i) \in R \times R$ with $b_i = a_i s + e_i$ where the a_i's are uniformly random in R, s is random in R, and the e_i's are "small" in R.

In the decisional version of Ring-LWE, one needs to distinguish such ordered pairs (a_i, b_i) from uniformly random $(a_i, u_i) \in R \times R$.

Next, taken from [BV1], is a simplified symmetric key somewhat homomorphic encryption scheme whose security is based on the decisional version of Ring-LWE. Fix an odd prime p and let R_p denote the ring $\mathbb{F}_p[x]/\langle x^N+1\rangle$ where $N=2^n$. The secret key is a random $s\in R_p$. To encrypt $m\in\mathbb{F}_2[x]/\langle x^N+1\rangle$, lift m to a polynomial $\{0,1\}[x]\subset\mathbb{Z}[x]$ of degree $< N$ and (reduce mod p and x^N+1 to) view it in as an element $\hat{m}$ of R_p. Then choose a random $a\in R_p$ and a "noise" e, and compute $b := as+2e+\hat{m}\in R_p$. The ciphertext is $(a,b)\in R_p\times R_p$. To decrypt, compute $b-as \pmod 2 = m$. Security follows from decisional Ring-LWE for R_p, since under that assumption and the fact that p is odd, pairs $(a, as+2e)$ are indistinguishable from pairs (a,u) where u is uniformly random in R_p. Again, this can be turned into a fully homomorphic encryption scheme (see [BV1]).

Fully homomorphic encryption schemes based on Ring-LWE are more efficient than those based on standard LWE. However, Ring-LWE uses lattices coming from ideals in algebraic number fields. As mentioned earlier, it is not known whether cryptosystems based on ideal lattices are more vulnerable to attack than those based on general lattices.

References

[BoGN] D. Boneh, E-J. Goh, and K. Nissim, *Evaluating 2-DNF formulas on ciphertexts*, in Theory of Cryptography—TCC'05, Lect. Notes in Comp. Sci. **3378** (2005), Springer, 325-341.

[BoSW] D. Boneh, G. Segev, and B. Waters, *Targeted malleability: homomorphic encryption for restricted computations*, Innovations in Theoretical Computer Science 2012 (ITCS 2012), ACM (2012), 350–366.

[Br] Z. Brakerski, *Fully Homomorphic Encryption without Modulus Switching from Classical GapSVP*, in Advances in Cryptology—CRYPTO 2012, Lect. Notes in Comp. Sci. **7417** (2012), Springer, 868–886.

[BrGV] Z. Brakerski, C. Gentry, and V. Vaikuntanathan, *(Leveled) fully homomorphic encryption without bootstrapping*, in Innovations in Theoretical Computer Science (ITCS) 2012, ACM, 309–325.

[BV1] Z. Brakerski and V. Vaikuntanathan, *Fully Homomorphic Encryption from Ring-LWE and Security for Key Dependent Messages*, in Advances in Cryptology—CRYPTO 2011, Lect. Notes in Comp. Sci. **6841** (2011), Springer, 505–524.

[BV2] Z. Brakerski and V. Vaikuntanathan, *Efficient Fully Homomorphic Encryption from (Standard) LWE*, FOCS 2011, IEEE 52nd Annual Symposium on Foundations of Computer Science, IEEE (2011), 97–106.

[CN] Y. Chen and P. Q. Nguyen, *Faster Algorithms for Approximate Common Divisors: Breaking Fully-Homomorphic-Encryption Challenges over the Integers*, in Advances in Cryptology—EUROCRYPT 2012, Lect. Notes in Comp. Sci. **7237** (2012), Springer, 502–519.

[CoH] H. Cohn and N. Heninger, *Approximate common divisors via lattices*, to appear in to appear in Algorithmic Number Theory (ANTS X), Mathematical Sciences Publishers; `http://arxiv.org/abs/1108.2714`.

[CorMNT] J-S. Coron, A. Mandal, D. Naccache, and M. Tibouchi, *Fully homomorphic encryption over the integers with shorter public keys*, in Advances in Cryptology—CRYPTO 2011, Lect. Notes in Comp. Sci. **6841** (2011), Springer, 487-504.

[CorNT] J-S. Coron, D. Naccache, and M. Tibouchi, *Public Key Compression and Modulus Switching for Fully Homomorphic Encryption over the Integers*, in Advances in Cryptology—EUROCRYPT 2012, Lect. Notes in Comp. Sci. **7237** (2012), Springer, 446–464.

[DGHV] M. van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan, *Fully Homomorphic Encryption over the Integers*, in Advances in Cryptology—EUROCRYPT 2010, Lect. Notes in Comp. Sci. **6110** (2010), Springer, 24–43.

[G1] C. Gentry, *Fully homomorphic encryption using ideal lattices*, in Proceedings of the 41st ACM Symposium on Theory of Computing—STOC 2009, ACM, New York (2009), 169–178.

[G2] C. Gentry, *A fully homomorphic encryption scheme*, Stanford University PhD thesis, 2009, `http://crypto.stanford.edu/craig/craig-thesis.pdf`.

[G3] C. Gentry, *Computing arbitrary functions of encrypted data*, Communications of the ACM **53** (2010), 97–105.

[G4] C. Gentry, *Toward Basing Fully Homomorphic Encryption on Worst-Case Hardness*, in Advances in Cryptology—CRYPTO 2010, Lect. Notes in Comp. Sci. **6223** (2010), Springer, 116–137.

[GH1] C. Gentry and S. Halevi, *Implementing Gentry's Fully-Homomorphic Encryption Scheme*, in Advances in Cryptology—EUROCRYPT 2011, Lect. Notes in Comp. Sci. **6632**, (2011), Springer, 129–148.

[GH2] C. Gentry and S. Halevi, *Fully Homomorphic Encryption without Squashing Using Depth-3 Arithmetic Circuits*, `http://eprint.iacr.org/2011/279.pdf`; extended abstract appeared in FOCS 2011.

[GHS1] C. Gentry, S. Halevi, and N. P. Smart, *Better Bootstrapping in Fully Homomorphic Encryption*, Public Key Cryptography 2012, 1–16.

[GHS2] C. Gentry, S. Halevi, and N. P. Smart, *Fully Homomorphic Encryption with Polylog Overhead*, in Advances in Cryptology—EUROCRYPT 2012, Lect. Notes in Comp. Sci. **7237** (2012), Springer, 465–482.

[GM] S. Goldwasser and S. Micali, *Probabilistic encryption and how to play mental poker keeping secret all partial information*, Proceedings of the 14th ACM Symposium on Theory of Computing—STOC 1982, ACM (1982), 365-377.

[H] B. Hayes, *Alice and Bob in Cipherspace*, American Scientist **100** (2012), 362–367.

[HofPS] J. Hoffstein, J. Pipher, and J. H. Silverman, *NTRU: A Ring-Based Public Key Cryptosystem*, in Proceedings of ANTS-III—Algorithmic Number Theory Third International Symposium, Lect. Notes in Comp. Sci. **1423**, (1998), Springer, 267–288.

[How] N. Howgrave-Graham, *Approximate integer common divisors*, in Cryptography and Lattices, International Conference, CaLC 2001, Lect. Notes in Comp. Sci. **2146** (2001), Springer, 51-66.

[LaNV] K. Lauter, M. Naehrig, and V. Vaikuntanathan, *Can homomorphic encryption be practical?*, in Proceedings of the 3rd ACM Cloud Computing Security Workshop, CCSW 2011, ACM, New York, 113–124.

[LLL] A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász, *Factoring polynomials with rational coefficients*, Math. Ann. **261** (1982), 515-534.

[LMSV] J. Loftus, A. May, N. P. Smart, F. Vercauteren, *On CCA-Secure Somewhat Homomorphic Encryption*, in Selected Areas in Cryptography 2011, Lect. Notes in Comp. Sci. **7118** (2012), Springer, 55–72.

[LTV] A. Lopez-Alt, E. Tromer, and V. Vaikuntanathan, *On-the-Fly Multiparty Computation on the Cloud via Multikey Fully Homomorphic Encryption*, STOC 2012, Proceedings of the 44th symposium on Theory of Computing (2012), ACM, New York, 1219–1234.

[LyPR] V. Lyubashevsky, C. Peikert, and O. Regev, *On Ideal Lattices and Learning with Errors over Rings*, in EUROCRYPT 2010, Lect. Notes in Comp. Sci. **6110** (2010), Springer, 1–23.

[Pa] P. Paillier, *Public-Key Cryptosystems Based on Composite Degree Residuosity Classes*, in Advances in Cryptology—EUROCRYPT '99, Lect. Notes in Comp. Sci. **1592** (1999), Springer, 223–238.

[Pe] C. Peikert, *Public-key cryptosystems from the worst-case shortest vector problem*, in STOC 2009, ACM, 333-342.

[PR] M. Prabhakaran and M. Rosulek, *Homomorphic Encryption with CCA Security*, `http://eprint.iacr.org/2008/079`.

[R] O. Regev, *On lattices, learning with errors, random linear codes, and cryptography*, in STOC 2005, ACM, 84-93; full version in J. ACM **56** (2009).

[RAD] R. Rivest, L. Adleman, and M. Dertouzos, *On Data Banks and Privacy Homomorphisms*, in Foundations of Secure Computation, R. DeMillo, D. Dobkin, A. Jones, and R. Lipton (eds.), Academic Press, New York (1978), 169–180.

[SmV] N. P. Smart and F. Vercauteren, *Fully Homomorphic Encryption with Relatively Small Key and Ciphertext Sizes*, in Public Key Cryptography—PKC 2010, Lect. Notes in Comp. Sci. **6056** (2010), Springer, 420–443.

[SS] D. Stehlé and R. Steinfeld, *Faster fully homomorphic encryption*, in Advances in Cryptology—ASIACRYPT 2010, Lect. Notes in Comp. Sci. **6477** (2010), Springer, 377-394.

[V] V. Vaikuntanathan, *Computing Blindfolded: New Developments in Fully Homomorphic Encryption*, FOCS 2011, 5–16.

Department of Mathematics, University of California, Irvine, CA 92697
E-mail address: `asilverb@math.uci.edu`