Hindawi
Security and Communication Networks
Volume 2023, Article ID 3106927, 10 pages
https://doi.org/10.1155/2023/3106927

Research Article

CRT-Based Homomorphic Encryption over the Fraction

De Zhao,[1] **Haiyang Ding**,[1] **Zhenzhen Li**,[1] **Zhenzhen Zhang**,[1] **Zichen Li**,[1] **and Jing Gao**[2]

[1]*College of Information Engineering, Beijing Institute of Graphic Communication, Beijing 102600, China*
[2]*State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China*

Correspondence should be addressed to Zichen Li; lizc2020@163.com

Received 11 October 2021; Revised 30 January 2022; Accepted 3 March 2022; Published 8 May 2023

Academic Editor: Leo Yu Zhang

Homomorphic encryption technology is the holy grail of cryptography and has a wide range of applications in practice. This paper proposes a homomorphic encryption scheme over the fraction based on the Chinese remainder theorem (CRT) Dayan qiuyi rule. This homomorphic scheme performs encryption and decryption operations by forming congruence groups and has homomorphism. The solution in this paper first combines the traditional CRT algorithm with the Dayan qiuyi rule to obtain the CRTF algorithm that can be operated on the fraction field. Finally, in the decryption process, modulo arithmetic is used twice to obtain the correct plaintext components, restored to plaintext by CRTF. The scheme's security is related to a decisional version of an approximate GCD problem. The proof of theoretical derivation shows that this paper's homomorphic encryption scheme can realize the homomorphic addition operation in the fraction field. Compared with the CKKS scheme, efficiency is improved.

1. Introduction

With the birth and development of the Internet and cloud computing concepts, people's demands for data processing and search are constantly increasing, making homomorphic encryption (HE) more critical. HE is also the focus and hot issue of international cryptography research in recent years. The concept of HE appeared in the paper [1] jointly published by Rivest, Adleman, and Detourzos in 1978. It first proposed the concept of calculating encrypted data without decrypting the encrypted data. The advantage of HE is that users can still analyze and retrieve encrypted data when the data are encrypted [2], which ensures the security of data transmission and prevents the plaintext from being exposed or leaked when the data are processed in the cloud.

Furthermore, the correct encrypted data can also get the correct decryption result [3]. HE has a significant application value, and it has many applications in cloud computing and electronic voting. After the idea of homomorphism was proposed, many scholars tried to construct a fully homomorphic encryption (FHE) scheme, but none of the proposed schemes possessed the characteristic of full homomorphism [3–9]. On this basis, a homomorphic cipher that can satisfy finite times of multiplication and addition at the same time is also proposed [8–22]. It is called somewhat homomorphic encryption (SWHE). In 2009, Gentry was the first to construct an FHE scheme [10] based on the ideal lattice concept [11]. Since then, Gentry has successively constructed some other FHE schemes [12]. In addition, in order to promote the idea of "bootstrapping," Gentry used a simple algebraic structure to construct a DGHV10 scheme [23] over the integer in 2010. In the follow-up, many scholars not only carried out a lot of improvement and advancement work but also expanded the plaintext domain, increased efficiency, and solved the problem of ciphertext expansion.

After Gentry's breakthrough, homomorphic cryptography is known as a hot topic again. In 2011, Brakerski proposed a lattice-based encryption hypothesis learning with errors (LWEs) [24]. In the same year, Brakerski, Gentry, and Vaikuntanathan completed the system together and officially published it. It is called BGV [12]. The BGV system is a homomorphic encryption system with a finite number of stages, but it can be turned into a fully homomorphic system through Bootstrapping. The BGV system is the second-generation FHE system.

In 2013, Gentry, Sanai, and Waters launched the new GSW scheme [25]. The GSW system is similar to BGV and has a finite series of fully homomorphic properties. GSW is called the third-generation FHE system.

After 2013, based on original third-generation FHE, various new designs have emerged, dedicated to optimizing and accelerating the operating efficiency of the BGV and GSW systems. IBM developed an open-source fully homomorphic computing library (HElib) based on the BGV system and successfully transplanted it to major mobile platforms.

However, the above homomorphic encryption schemes (whether SWHE or FHE schemes) are mostly applied to the integer field [23], and there is still a gap in the use of homomorphic encryption over fractions.

In 1978, Rivest mentioned an example in his paper [1], which was based on CRT. It is homomorphic, but it is insecure and challenging to resist known plaintext attacks [26].

DGHV15 [27] solves the security problem by adding random information to ciphertext, and at the same time, links security to a decisional version of approximate GCD problems. Nevertheless, it did not solve the homomorphic operation of fractions.

This paper first reviews DGHV15 [27], combines CRT with the Dayan qiuyi rule based on it to obtain CRTF, and applies CRTF to the encryption and decryption processes, expanding the scheme's calculation range from integers to fractions. Furthermore, the scheme also has a homomorphic nature. The safety proof of the scheme proposed in this paper is equivalent to the safety analysis of DGHV15 [27].

In Section 2, some basic concepts are introduced. In Section 3, the scheme of this article is explained in detail, including parameters and structure. Section 3.3 proves correctness and homomorphism. Finally, safety and efficiency are compared and analyzed in Sections 3.4 and 3.5.

2. Preliminaries

2.1. Chinese Remainder Theorem. CRT (Chinese remainder theorem) first appeared in a book on mathematics during the Southern and Northern dynasties of China. *Sunzi Suanjing* (Problem26, Volume 3) reads "there are certain things whose number is unknown. A number is repeatedly divided by 3, the remainder is 2; divided by 5, the remainder is 3; and by 7, the remainder is 2. What will the number be?" This problem can be expressed as

$$x \equiv 2 \pmod 3 \equiv 3 \pmod 5 \equiv 5 \pmod 7. \tag{1}$$

Generally, the target object of the CRT is one-variable congruence equations. $p_1, p_2, .., p_n$ is a positive integer that is prime to each other [15]. c and $r_i (i = 1, 2, .., n)$ are both positive integers:

$$\begin{cases} c \equiv r_1 \pmod{p_1}, \\ c \equiv r_2 \pmod{p_2}, \\ \vdots \\ c \equiv r_n \pmod{p_n}. \end{cases} \tag{2}$$

We can find the positive integer solution of the unitary congruence group. The above process is called the Chinese reminder theorem (CRT) or *Sunzi* theorem [16]. It has many applications in various fields [17]. Its specific form can be expressed as follows.

The solution c is as follows:

$$c = \sum_{i=1}^{n} P_i F_i r_i \pmod P, \tag{3}$$

where F_i is multiplicative inverse and must meet the conditions of $P_i F_i \equiv 1 \pmod{p_i}$.

$$\begin{aligned} P &= \prod_{i=1}^{n} p_i, \\ P_i &= \frac{P}{p_i}. \end{aligned} \tag{4}$$

2.2. Dayan Qiuyi Rule. The *Dayan qiuyi* rule originated from the mathematics book *Shushu Jiuzhang* written by Qin Jiushao in 1247 AD in the Song dynasty. Some of the problems are expressed by the congruence system [27]. The modulus and remainder of the congruence equations formed by these practical problems have different situations, including decimals, integers, and fractions. The rest of the numbers in the fraction field provides theoretical feasibility and ideas for the scheme of this article.

To solve the remainder's situation in the fraction field, the remainder and the modulus must be multiplied by the least common multiple of two denominators. The modulus set no longer satisfies any two elements in the modulus set mutually prime in the unary congruence equation. The set of modulus needs to be transformed into an equivalent form on the unary congruence equation.

The modulus set can be divided into the following four categories according to the relative prime of the elements in its own set.

Yuanshu: there is no greatest common factor in the set of modules

Tongshu: there are elements in the set of modules that exist in the fraction field

Fushu: there is the greatest common factor in the modulus set

Dingshu: any two elements in the set of modulus are relatively prime

Step 1. We convert the remainder existing in the fraction field into an equivalent integer form. $\{x_i\}, \{y_i\}, z$ are all integers:

$$\begin{aligned} c &\equiv \frac{x_i}{z} \left(\bmod \frac{y_i}{z} \right) (i = 1, 2, \cdots, n), \\ zc &\equiv x_i \pmod{y_i} (i = 1, 2, \cdots, n). \end{aligned} \tag{5}$$

If there is the greatest common factor in $\{y_i\}(i = 1, 2, \cdots, n)$, we go to Step 2. Otherwise, we enter Step 3.

Step 2. We convert *Fushu* to *Yuanshu*.

Essentially, it solves the situation where there are common factors in the set of modules. We suppose that there is a set of modulus $a, b, .., c$, where $a = a_1 d, b = b_1 d, \cdots, c = c_1 d$. $a, b, .., c$ can be transformed into equivalent $a, b_1, \cdots, c_1$ if $d = \gcd(a, b, \cdots, c)$ and a_1 is the exponent of a common factor, which is the highest.

Step 3. The process of transforming *Yuanshu* into *Dingshu* is as follows.

For the modulus set $y_1, y_2, \cdots, y_n$, we can convert it to *Dingshu* by the following calculation method. We get first $d_1 = \gcd(y_{n-1}, y_n)$. y_{n-1} and y_n are the *Dingshu* of y_{n-1} and y_n if $d_2 = \gcd(y_{n-1}/d_1, y_n)$. y_{n-1} and y_n/d_1 are the *Dingshu* of y_{n-1} and y_n if $d_2 > 1$ and $d_2' = \gcd(y_{n-1}/d_1, y_n)$.

We continue to iterate and calculate along with the above rules.

$p_{n-1}' = y_{n-1}/d_2' d_3 \cdots d_k$ and $p_n' = y_{n-1} d_2' d_3 \cdots d_k/d_1$ as the *Dingshu* of y_{n-1} and y_n if $d_{k+1}' = \gcd(p_{n-1'}, p_n) = 1$. Considering the length of this paper, the detailed derivation and calculation process can be obtained in Kangsheng [28]. Performing circular operations on the subsequent modulus can get the converted modulus set. Afterwards, CRT can be used to perform substitution operations on the fraction field.

The *Dayan qiuyi* rule transforms the remainder in the congruence equation from the fractional form to the equivalent integer form on the congruence equation. Furthermore, through certain arithmetic rules, the modulus is transformed into a modulus set of any two elements that are relatively prime so that it conforms to the construction form of unary congruence. The final integer solution answer can be obtained by using the CRT method.

2.3. Operation over the Fraction. In the fraction field, we must first define the operating rules of modular arithmetic. k is an integer:

$$\frac{r}{R} \bmod p = \frac{r_0}{R}, \tag{6}$$

where $0 < r_0 < Rp$ and $r_0 = r - kRp$.

The modular addition operation of the fractional field is as follows:

$$\frac{r_1}{R_1} \bmod p + \frac{r_2}{R_2} \bmod p = \left(\frac{r_1}{R_1} + \frac{r_2}{R_2}\right) \bmod p. \tag{7}$$

The proof process, where k_1 and k_2 are an integer, is as follows:

$$\begin{cases} \dfrac{r_1}{R_1} \bmod p = \dfrac{r_1 - k_1 R_1 p}{R_1}, \\ \dfrac{r_2}{R_2} \bmod p = \dfrac{r_2 - k_2 R_2 p}{R_2}, \end{cases} \tag{8}$$

$$\frac{r_1}{R_1} \bmod p + \frac{r_2}{R_2} \bmod p = \frac{r_1 R_2 + r_2 R_1 - (k_1 + k_2) R_1 R_2 p}{R_1 R_2},$$

$k_1 + k_2 = k$. Because both k_1 and k_2 are integers, k is also an integer:

$$\frac{r_1 R_2 + r_2 R_1 - k R_1 R_2 p}{R_1 R_2} = \left(\frac{r_1}{R_1} + \frac{r_2}{R_2}\right) \bmod p. \tag{9}$$

Example 1. Suppose there is a set of modulus $\tau = \{54, 57, 75, 72\}$, we transform it into a set of modulus cc with any two elements that are relatively prime by the *Dayan qiuyi* rule:

Step 1: We convert *Fushu* to *Yuanshu* that is to remove the greatest common divisor in the modulus set τ. Due to $3 = \gcd(54, 57, 75, 72)$, each element in the set τ can be written as $54 = 3^3 \times 2$, $57 = 3^1 \times 19$, $75 = 3^1 \times 25$, and $72 = 3^2 \times 8$. The power of factor 3 in number 54 is 3, and it is the highest value. So $\tau' = \{54, 57/3, 75/3, 72/3\} = \{54, 19, 25, 24\}$.

Step 2: We convert *Yuanshu* to *Dingshu* to convert the modulus set τ' that does not contain common factors but may have two elements that are not mutually prime to any two modulus set τ'' that is relatively prime to any two elements. We name the elements in the τ' collection. The name from front to back is τ_1, τ_2, τ_3, and τ_4. So $\tau_1 = 54$, $\tau_2 = 19$, $\tau_3 = 25$, and $\tau_4 = 24$. The converted modulus set is $\tau'' = \{27, 19, 25, 8\}$.

$$\begin{cases} \gcd(\tau_3, \tau_4) = d_1 = 1, \\ \gcd\left(\dfrac{\tau_3}{d_1}, \tau_4\right) = d_2 = 1, \end{cases} \Rightarrow \tau_3 \longrightarrow \tau'',$$

$$\begin{cases} \gcd(\tau_2, \tau_4) = d_1 = 1, \\ \gcd\left(\dfrac{\tau_2}{d_1}, \tau_4\right) = d_2 = 1, \end{cases} \Rightarrow \tau_2 \longrightarrow \tau'',$$

$$\begin{cases} \gcd(\tau_1, \tau_4) = d_1 = 6, \\ \gcd\left(\frac{\tau_1}{d_1}, \tau_4\right) = d_2 = 3 > 1, \\ \gcd\left(\tau_1, \frac{\tau_4}{d_1}\right) = d_2^{'} = 2 > 1, \\ \gcd\left(\frac{\tau_1}{d_2^{'}}, \tau_4 \frac{d_2}{d_1}\right) = d_3 = 1. \end{cases} \Rightarrow \frac{\tau_1}{d_2^{'}}, \frac{\tau_4 d_2^{'}}{d_1} \longrightarrow \tau'', \quad (10)$$

This process can be clearly and intuitively demonstrated (see Figure 1).

3. Our Homomorphic Encryption Scheme

This paper proposes a homomorphic encryption scheme that can process data in the fraction field based on the above theoretical discussion. The method obtained by improving the CRT based on the *Dayan qiuyi* rule is called CRTF. By using CRTF in the encryption and decryption process can encrypt and decrypt data in the fraction field.

In Section 2.1 of this paper, r_i and p_i are positive integers (see equation (2) or equation (3)). Based on the basic requirements of CRT and congruence groups, any two elements in $p_i (i = 1, 2, \cdots, n)$ should be relatively prime. However, in the CRTF constructed, the restrictions on the modulus and remainder are relaxed so that the remainder can exist on the fraction field and have stronger ability to solve practical problems:

$$m = CRTF_{(p_1, p_2, \cdots, p_n)}\left(\frac{r_1}{R_1}, \frac{r_2}{R_2}, \cdots, \frac{r_n}{R_n}\right),$$

$$= \begin{cases} m \equiv \frac{r_1}{R_1} \pmod{p_1}, \\ m \equiv \frac{r_2}{R_2} \pmod{p_2}, \\ \vdots \\ m \equiv \frac{r_n}{R_n} \pmod{p_n}, \end{cases} \quad (11)$$

where r_i, R_i, and p_i are all positive integers.

Suppose there is a group of unary congruences in the form $m \equiv r_i / R_i \bmod p_i$, where r_i, R_i, and p_i are all integers, we use the following steps:

Step 1: The unary congruence group is expressed in the following form:

$$\begin{cases} Qm \equiv r_1 R_1' \bmod (Qp_1), \\ Qm \equiv r_2 R_2' \bmod (Qp_2), \\ \vdots \\ Qm \equiv r_n R_n' \bmod (Qp_n), \end{cases} \quad (12)$$

where $R_{k'} = Q / R_k$ and $Q = lcm(R_1, R_2, \cdots, R_n)$.

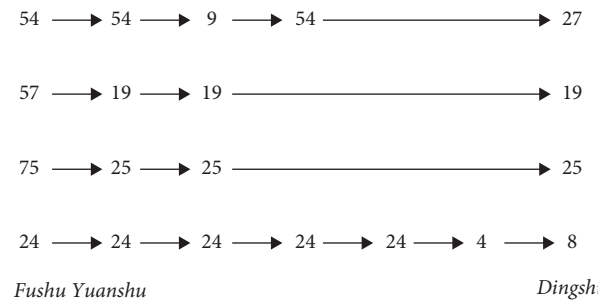

Figure 1: *Fushu* into *Yuanshu* and then into *Dingshu*.

Step 2: The modulus set $\{Qp_1, Qp_2, \cdots, Qp_n\}$ is transformed into a modulus set $\{(Qp_1)', (Qp_2)', \cdots, (Qp_n)'\}$ that meets the requirement of pairwise coprime through the method in Section 2.2 of this article. To form a new congruence group, we get the following:

$$\begin{cases} Qm \equiv r_1 R_1' \bmod (Qp_1)', \\ Qm \equiv r_2 R_2' \bmod (Qp_2)', \\ \vdots \\ Qm \equiv r_n R_n' \bmod (Qp_n)', \end{cases} \quad (13)$$

$$P = \prod_{i=1}^{n} (Qp_i)', P_i = \frac{P}{(Qp_i)'},$$

where $Qm \equiv \sum_{i=1}^{k} P_i F_i r_i R_i' \pmod{P}$ and $F_i P_i \equiv 1 \bmod (Qp_i)'$.

Example 2. We solve the answer to a system of unary congruence equations $m = CRTF_{(37,41)}(1/4, 27/12)$ based on the fractional domain:

$$\begin{cases} m \equiv \frac{1}{4} \bmod 37, \\ m \equiv \frac{27}{12} \bmod 41. \end{cases} \quad (14)$$

The specific process of the solution is as follows:

Step 1: By multiplying both ends of each congruence in the unary congruence group by $lcm(4, 12) = 12$ at the same time, we can get

$$\begin{cases} 12m \equiv 3 \bmod 444, \\ 12m \equiv 27 \bmod 492. \end{cases} \quad (15)$$

Step 2: The modulus set $\{444, 492\}$ is transformed into a modulus set that is equivalent and conforms to the $\{37, 492\}$ pairwise prime through the Dayan qiuyi rule:

$$\begin{cases} 12m \equiv 3 \bmod 37, \\ 12m \equiv 27 \bmod 492. \end{cases} \quad (16)$$

Step 3: We bring various parameters into the solution formula abovementioned. We can get $12m = 8883 \Leftrightarrow m = 8883/12$.

3.1. Parameters. Many schemes require a constant (the number is 2 in DGHV10 [23]) or parameters to determine their plaintext domain; constructing an array $\{Q_i\}_{i=1}^{k}$ is necessary to clarify the plaintext domain. Q_i must be a prime number, and l_Q is the bit length of Q_i. k is vital because it determines the number of prime elements in sk and the size of the plaintext space to a certain extent. We set a parameter $U = \prod_{i=1}^{k} Q_i$, and the plaintext space is Q_U^+ where Q stands for a rational number.

3.2. Construction. In this part, we mainly discuss the four structures of key generation, encryption processes, decryption processes, and addition homomorphic operation. For the convenience of expression, we define $CRT_{(q_0,p_1,\cdots,p_k)}$ as CRT. Similarly, $CRTF$ can also be used to express $CRTF_{(q_0,p_1,\cdots,p_k)}$.

KeyGen (k, $\{Q_i\}$): A set of η-bit prime numbers $\{p_i\}_{i=1}^{k}$ is selected. $q_0 \leftarrow [0, 2^{\gamma}/\prod p_i)$. γ is bit length of the ciphertext. Setting parameter $x_0 = q_0 \prod_{i=1}^{k} p_i$ is used to reduce ciphertext expansion, and it should meet condition $\gcd(Q_i, x_0) = 1$ for each value of i:

$$X \coloneqq \{x_i = CRT(e_{i0}, e_{i1}Q_1, \ldots, e_{ik}Q_k)\}_{i=1}^{\tau} \tag{17}$$

where $e_{i0} \leftarrow Z \cap [0, q_0)$ and $e_{i1}, \ldots, e_{ik} \leftarrow Z \cap (-2^{\rho}, 2^{\rho})$. Obviously, ρ is the bit length of the random error. We output the public key $PK = (X, x_0)$ and secret key $SK = (\{p_i\}_{i=1}^{k})$.

Enc (PK, m): The output is $c = c_0 \mod x_0$:

$$c_0 = \sum_{i=1}^{k} m_i CRTF(e, e_1Q_1, \ldots, 1 + e_iQ_i, \ldots, e_kQ_k) + \mathrm{Sum}_r(PK), \tag{18}$$

where $m_i = m \mod Q_i (i = 1, 2, \ldots, k)$. Similarly, $e \leftarrow Z \cap [0, q_0)$ and $e_i \leftarrow Z \cap (-2^{\rho}, 2^{\rho})$ for $i = 1, .., k$. $\mathrm{Sum}_r(PK) = \sum_{i \in r} x_i$ where r is a random subset of $\{1, \ldots, \tau\}$.

Dec (SK, c): the output is $m = CRTF_{(Q_1,Q_2,\ldots,Q_k)}(d_1, d_2, \ldots, d_k)$ where $d_i = (c \mod p_i) \mod Q_i$

Add (PK, c_1,..., c_n): the output is $\sum_{i=1}^{n} c_i \mod x_0$

Mul (PK, c_1,..., c_n): the output is $\prod_{i=1}^{n} c_i \mod x_0$

First, the plaintext space of the structure can be limited to Z_2^k if $Q_1 = Q_2 = Q_K = 2$ is met. Furthermore, the structure mentioned above is no different from DGHV10 [23] if $k = 1$. Second, x_0 also limits the expansion of the ciphertext and plays a role in reducing the bit length of the ciphertext. Finally, the public key PK can be understood as a set of 0 ciphertexts, with τ elements in total. We pick a random number of x_i to sum and append it to the ciphertext. $\mathrm{Sum}_r(PK)$ after two modulo operations in the decryption process is 0, and the next part of the proof may ignore this part.

3.3. Additive Homomorphism. In this section, we demonstrate the homomorphism and correctness of our construction in this paper. We denote $CRTF(e, e_1Q_1, \cdots, 1 + e_iQ_i, \cdots, e_kQ_k)$ by $CRTF(i)$. Assuming that M is plaintext, there are some random numbers c whose bit length is ρ. According to Theorem 1, $\alpha_i|_{i=1}^{k}$ also exists and meets the requirements:

$$\begin{cases} m_1 = M \pmod{p_1}, \\ m_2 = M \pmod{p_2}, \\ \vdots \\ m_k = M \pmod{p_k}. \end{cases} \tag{19}$$

Part of the encryption process can be expressed by the following equation:

$$\begin{aligned} m_i CRTF(i) &= m_i(e_{i1}Q_1 + \alpha_{i1}p_1) \\ &= m_i(e_{ii}Q_i + m_i\alpha_{ii}p_i) + m_i \\ &= m_i(e_{ik}Q_k + \alpha_{ik}p_k). \end{aligned} \tag{20}$$

Putting equation (20) into the encryption process can get

$$c = m_i + \left(\sum_{j=1}^{k} m_j e_{ij}\right) Q_i + \left(\sum_{j=1}^{k} m_j \alpha_{ij}\right) p_i. \tag{21}$$

We decrypt the ciphertext c to get

$$\begin{aligned} d_1 &= (c \mod p_1) \bmod Q_1 = m_1, \\ d_2 &= (c \mod p_2) \bmod Q_2 = m_2, \\ &\vdots \\ d_k &= (c \mod p_n) \bmod Q_n = m_k. \end{aligned} \tag{22}$$

We can get $M = CRTF_{(Q_1,Q_2,\cdots,Q_k)}(d_1, d_2, \cdots, d_k)$, which prove that construction can correctly encrypt and decrypt data.

Theorem 1. *We can get $\alpha_i|_{i=1}^{k}$ from $x = CRTF_{(p_1,p_2,..,p_k)}(r_1, r_2, .., r_k)$, and we can also represent x in $x = \alpha_i p_i + r_i$ for $i \in \{1, 2, .., k\}$.*

Proof. When c, $x \equiv r \mod p$ can be written as $x - r = \alpha p$. We can get $x = \alpha p + r$, and we can get our conclusions by promotion.

Verifying homomorphism requires us to assume ciphertexts c and c', derived from the encryption of m and m' sequentially. The two ciphertexts have k components on $\{Q_i\}_{i=1}^{k}$ each:

$$\begin{aligned} c &= m_i + \left(\sum_{j=1}^{k} m_j e_{ij}\right) Q_i + \left(\sum_{j=1}^{k} m_j \alpha_{ij}\right) p_i, \\ c' &= m_i' + \left(\sum_{j=1}^{k} m_j' e_{ij}'\right) Q_i + \left(\sum_{j=1}^{k} m_j' \alpha_{ij}'\right) p_i. \end{aligned} \tag{23}$$

We set h to be the sum of ciphertexts of m and m':

$$\begin{aligned} h &= c + c' \\ &= m_1 + m_1' + A_1 p_1 + B_1 Q_1 \\ &\vdots \\ &= m_k + m_k' + A_k p_k + B_k Q_k, \end{aligned} \tag{24}$$

where $A_i = (\sum_{j=1}^{k} m_j e_{ij}) + (\sum_{j=1}^{k} m_j' e_{ij}')$ and $B_i = (\sum_{j=1}^{k} m_j \alpha_{ij}) + (\sum_{j=1}^{k} m_j' \alpha_{ij}')$.

We bring h into the decryption process:

$$\begin{aligned} d_1 &= (h \bmod p_1) \bmod Q_1 = m_1 + m_1', \\ d_2 &= (h \bmod p_2) \bmod Q_2 = m_2 + m_2', \\ &\vdots \\ d_k &= (h \bmod p_k) \bmod Q_k = m_k + m_k'. \end{aligned} \tag{25}$$

We continue to decrypt h:

$$\mathrm{Dec}(h) = CRTF_{(Q_1, Q_2, \cdots, Q_k)}\left(m_1 + m_1', m_2 + m_2', \cdots, m_k + m_k'\right). \tag{26}$$

We expand the CRTF to get

$$\begin{cases} \mathrm{Dec}(h) \equiv \left(m_1 + m_1'\right) \bmod Q_1, \\ \mathrm{Dec}(h) \equiv \left(m_2 + m_2'\right) \bmod Q_2, \\ \vdots \\ \mathrm{Dec}(h) \equiv \left(m_k + m_k'\right) \bmod Q_k. \end{cases} \tag{27}$$

According to Theorem 2, $\mathrm{Dec}(h) = (m + m')$ or $\mathrm{Dec}(h) + \prod_{i=1}^{k} Q_i = (m + m')$ can be obtained. In the plaintext domain Q_q^+, there is a relationship of $(m + m') < \prod_{i=1}^{k} Q_i$. In the end, we can get $\mathrm{Dec}(\mathrm{Enc}(m) + \mathrm{Enc}(m')) = (m + m')$.

Theorem 2. *We can get $c = m$ or $m = c + \prod_{i=1}^{n} p_i$ if c, m, and $p_i (i = 1, 2, \cdots, n)$ are a positive real number, and the set has the following conditions:*

$$\begin{cases} c \equiv m \bmod p_1, \\ c \equiv m \bmod p_2, \\ \vdots \\ c \equiv m \bmod p_n. \end{cases} \tag{28}$$

Proof.

$$\begin{cases} k_1 p_1 + c = m, \\ k_2 p_2 + c = m, \\ \vdots \\ k_n p_n + c = m, \end{cases} \tag{29}$$

where $k_i \in Z (i = 1, 2, \cdots, n)$.

The final answer of equation (28) is $k_i \in Z (i = 1, 2, \cdots, n)$ or $m = c + \prod_{i=1}^{n} p_i$:

$$\mathrm{Dec}\left(\mathrm{Enc}\left(M_1\right) + \mathrm{Enc}\left(M_2\right)\right) = M_1 + M_2. \tag{30}$$

So we can get equation (30), and the scheme in this paper is satisfied with homomorphism.

3.4. Security. In this section, discussing the security of the construction is the main content. The definition of the approximate GCD problem will also appear. The security of the construction in this paper is equivalent to the security proof of DGHV15 [27].

They all depend on a decisional version of the approximate GCD problem. In DGHV10 [23], the approximate GCD problem is also taken as a security guarantee.

Definition 1 (approximate GCD problem, AGCD). For η-bit prime p, we give some samples from $\Psi_{\gamma,\rho}(p)$ and find p.

Definition 2 (partial approximate GCD problem, ACD). For η-bit prime p, we give a γ-bit integer $x_0 = pq_0$ and some samples from $\Psi_{\gamma,\rho}(p)$ and find p.

$$\begin{aligned} \Psi_{\gamma,\rho}(p) := \{&\text{choose } q \leftarrow Z \cap [0, 2^{\gamma}), e \leftarrow Z \cap (-2^{\rho}, 2^{\rho}): \\ &\text{output } x = pq + e\}. \end{aligned} \tag{31}$$

The security of our scheme is based on a modified decisional ACD assumption. It [29] is shown that this assumption is equivalent to the ACD assumption. In order to resist the existing attacks, the parameters also need to have a certain range and equation. According to DGHV15 [27], we have the following:

$\gamma = \eta^2 \omega(\log \lambda)$: to resist Cohn and Heninger's attack [30] and the attack using the Lagarias algorithm [31] on the approximate GCD problem

$\eta = \widetilde{\Omega}(\lambda^2 + \lambda \cdot \rho)$: to resist the factoring attack using the elliptic curve method [32] and to permit enough multiplicative depth

$\rho = \widetilde{O}(\lambda)$: to be secure against Chen–Nguyen's attack [26] and Howgrave–Graham's

In addition, we choose $\gamma = \widetilde{O}(\lambda^5)$, $\eta = \widetilde{O}(\lambda^2)$, $\rho = 2\lambda$, and $\tau = \lambda + \gamma$, which is similar to DGHV10 [23] and DGHV15 [27].

Then, we introduce another decisional version of the ACD problem.

Definition 3 (decisional partial approximate GCD problem, DACD). For η-bit prime p, we give a γ-bit integer $x_0 = pq_0$ and some samples from $\Psi_{\rho}(p; q_0)$ and determine $b \in \{0, 1\}$ from $z = x + b * r \bmod x_0$, where $x \leftarrow \Psi_{\rho}(p; q_0)$ and $r \leftarrow Z \cap [0, x_0)$.

$$\Psi_\rho(p;q_0) \coloneqq \left\{\text{choose } e \leftarrow Z \cap [0,q_0), e' \leftarrow Z \cap (-2^\rho, 2^\rho): \text{output } CRT_{(q_0,p)}\left(e,e'\right)\right\}. \tag{32}$$

DACD says that, for given distribution $\Psi_\rho(p\colon q_0)$ and some integer z, it is hard to determine whether z is chosen from $\Psi_\rho(p\colon q_0)$ or not. Our scheme is semantically secure based on the DACD assumption. The DACD problem is hard for any polynomial time distinguisher. We define several definitions below in order to build a bridge between our scheme and DACD assumption.

Definition 4 (decisional partial approximate GCD_Q problem, DACD_Q). For η-bit prime p and an l_Q-bit integer Q, we give a γ-bit integer $x_0 = pq_0$ with $\gcd(x_0, Q) = 1$ and some samples from $\Psi_\rho(p;Q;q_0)$ and determine $b \in \{0,1\}$ from $z = x + b * r \bmod x_0$, where $x \leftarrow \Psi_\rho(p;Q;q_0)$ and $r \leftarrow Z \cap [0,x_0)$.

Definition 5 (k-decisional partial approximate GCD_Q problem, $k - DACD_Q$). For η-bit distinct prime $\{p_i\}_{i=1}^k$ and l_Q-bit integers $\{Q_i\}_{i=1}^k$, we give a γ-bit integer $x_0 = q_0\prod_{i=1}^k p_i$ with $\gcd(x_0, Q_i) = 1$ for $i \in \{1,2,..,k\}$ and some samples from $\Psi_\rho(\{p_i\}_{i=1}^k;\{Q_i\}_{i=1}^k;q_0)$ and determine $b \in \{0,1\}$ from $z = x + b * r \bmod x_0$, where $x \leftarrow \Psi_\rho(\{p_i\}_{i=1}^k;\{Q_i\}_{i=1}^k;q_0)$ and $r \leftarrow Z \cap [0,x_0)$.

$$\Psi_\rho\left(\{p_i\}_{i=1}^k;\{Q_i\}_{i=1}^k;q_0\right) \coloneqq \left\{\text{choose } e \leftarrow Z \cap [0,q_0), e_i \leftarrow Z \cap (-2^\rho, 2^\rho) \text{for} \forall i \in \{1,2,\ldots,k\}\text{: output} CRT_{(q_0,p_1,..,P_k)}(e,e_1Q_1,..,e_kQ_k)\right\} \tag{33}$$

We say that the DACD assumption holds if no polynomial time distinguisher can solve the DACD problem with non-negligible advantage. The k- DACD_Q assumption is defined similarly.

Due to three steps, our homomorphic encryption scheme is semantically secure under the DACD assumption:

Step 1: DACD $\longrightarrow$ DACD_Q (c)

Step 2: DACD_Q $\longrightarrow$ k- DACD_Q (Lemma 2)

Step 3: k- DACD_Q $\longrightarrow$ our construction

Lemma 1 (see [27]). *The DACD problem is reducible to the* DACD_Q *problem.*

Lemma 2 (see [27]). *The* DACD_Q *problem is reducible to the* k-DACD_Q *problem with the advantage of the latter k times that of the former on average.*

In order to complete the semantic security proof of the scheme, we also need to quote the two lemmas of DGHV15 [27].

Lemma 3 (see [27]). *We suppose that there is an attack algorithm* **A**. *The distribution of the pseudopublic key generated by it is indistinguishable from the standard public key generated by the scheme in Section 3.2.*

Lemma 4 (see [27]). *We suppose that there is an attack algorithm* **A**, *and the ciphertext generated by it is correct for the encryption process of the scheme in Section 3.2.*

Now, we prove the semantic security of our scheme.

Theorem 3. *The cryptosystem given in Section 3 is semantically secure if the* k-DACD_Q *assumption holds.*

Proof. We suppose that a polynomial time algorithm **B** breaks the semantic security of the scheme with nonnegligible advantage. There must be a polynomial time algorithm **A** that solves the k-DACD_Q problem with nonnegligible advantage. For η-bit distinct prime $\{p_i\}_{i=1}^k$ and l_Q-bit integers $\{Q_i\}_{i=1}^k$, the input of **A** is $(x_0, \{Q_i\}_{i=1}^k, \Psi_\rho(\{p_i\}_{i=1}^k;\{Q_i\}_{i=1}^k;q_0), CRTF(i)||_{i=1}^k, z)$, where $x_0 = q_0\prod_{i=1}^k p_i$ is a γ-bit integer. The algorithm **A** do as follows:

(1) **A** gives $(x_0, \{Q_i\}_{i=1}^k, \Psi_\rho(\{p_i\}_{i=1}^k; X \coloneqq \{x_j \leftarrow \{Q_i\}_{i=1\ j=1}^{k\ \tau}; q_0), CRTF(i)\,|_{i=1}^k)$ to **B** as the public key.

(2) **B** chooses $\{\vec{m}_0 = (m_{01},..,m_{0k}), \vec{m}_1 = (m_{11},..,m_{1k})\}$ and sends it to **A**.

(3) **A** computes $c' = \sum_{i=1}^k m_{bi} CRTF(i) + \text{Sum}_r(X) \bmod x_0$ for randomly chosen $b \in \{0,1\}$, where r is a random subset of $\{1,..,\tau\}$, and gives c' to **B**.

(4) **B** outputs $b \in \{0,1\}$.

(5) If $b = b'$, then **A** outputs 0. Otherwise, it outputs 1.

The public key given to **B** is correctly formed and distributed. We see that c' is uniform in Z_{x_0} when z is randomly chosen in Z_{x_0}. Hence, in this case, the advantage of **A** is zero since c' does not reveal any information of $\vec{m}_b$ and **B**'s probability of correct guessing is exactly 1/2. Thus, in this case, the probability of correct answer for **B** is at most negligibly different from that of **B**. This shows that the advantage of **A** is nonnegligible, violating the k-DACD_Q assumption. Therefore, there is not a polynomial time algorithm **B** that could break the semantic security of our scheme with nonnegligible advantage. The cryptosystem given in Section 3 is semantically secure. □

3.5. Efficiency Comparative Analysis. There are elements in $sk = (p_1, p_2, \cdots, p_k)$, and the value of k affects the operating efficiency of the encryption/decryption algorithm. The data

Table 1: The time of the homomorphic scheme in this paper. The time unit in the table is second (s).

k	5	15	25	35	40
Encryption time	0.0087	0.019	0.0293	0.0412	0.0485
Decryption time	0.0022	0.0064	0.0106	0.0149	0.0186

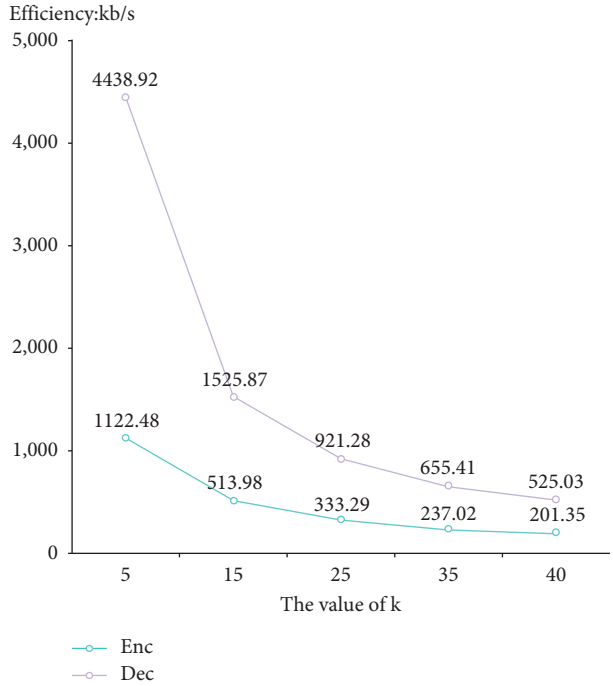

Figure 2: The impact of different k values on the efficiency of encryption and decryption.

Table 2: A comparison of the scheme based on CRTF and several other schemes in several aspects.

	SIMD	Key size	Ciphertext expansion	Complexity	Hard problems
DGHV10 [23]	N	$O(\lambda^{10})$	$O(\lambda^5)$	$O(\lambda^{12})$	AGCD
CMNT [33]	N	$O(\lambda^7)$	$O(\lambda^5)$	$O(\lambda^{15})$	AGCD
CLT13 [34]	Y	$O(\lambda^7)$	$O(\lambda^3)$	$O(\lambda^8)$	AGCD
Scheme based on CRTF	Y	$O(\lambda^{10})$	$O(\lambda^2)$	$O(\lambda^{10})$	PAGCD

in the table show the time of encrypting and decrypting 1-bit plaintext 10,000 times when k takes a typical value. The processor of the test equipment is Intel(R) Core(TM) i5-8250U @1.60 GHz.

From the data in Table 1, it can be seen that when the number of elements in the key is small, it has high efficiency. We can also use efficiency as the numerical value, which can more intuitively observe the influence of the k value on the encryption and decryption processes (see Figure 2). We compare the schemes in this paper with others for some theoretical complexity because it has many similarities with DGHV10 [23] and its derivative works [33, 34] (see Table 2).

We will show that the CKKS [35] in the SEAL library written by Microsoft to encrypt and decrypt one-bit plaintext 1,000 times. The result of running the CKKS [35] is as follows: the encryption operation takes 0.809 s, and the decryption operation takes 0.131 s. It is not comparable with the data in Table 1. The primary reason is that CKKS [35] is an FHE scheme, which uses more extensive parameters for subsequent homomorphic operations and noise control. However, by observing data, there are apparent advantages in our scheme within a specific parameter range under similar or consistent application scenarios, especially when the fractional domain homomorphic encryption scheme is not mature.

4. Conclusion

Most of the existing homomorphism encryption schemes are over integers. This paper proposes and implements a homomorphic encryption scheme over fractions. Compared with the homomorphic scheme over the integer field, this homomorphic encryption scheme on the fraction field has a broader range of applications and more practical application scenarios, such as banking and interest rate calculation. Similarly, the homomorphic scheme over the fraction also provides a theoretical basis and feasibility for the emergence of new operating modes for cloud computing or federated machine learning application scenarios. Furthermore, it will be of progressive significance if an FHE

scheme can be constructed based on CRT that can perform any form of operation on the ciphertext in the fractional domain like CKKS [36]. However, this paper does not make a detailed analysis of the noise problem or the possibility of transforming into an FHE scheme. The main content of the next step is to conduct a detailed analysis of the noise control problem, and at the same time, try to combine "bootstrapping" to transform it into an FHE scheme in the fractional domain and improve the operational efficiency of the program.

Data Availability

One part of data is from the SEAL (it was developed by Microsoft). The remaining parts of the experimental data about efficiency data used to support the findings of this study have not been made available because they will be used for the experiment and discussion of the next article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61370188), the Beijing Municipal Education Commission Scientific Research Project (KM202010015009, KM202110015004), the Beijing Institute of Graphic Communication Doctoral Funding Project (27170120003/020 and 27170122006), the BIGC Project (Ec202201), the Beijing Institute of Graphic Communication Research Innovation Team Project (Eb202101), the Intramural Discipline Construction Project of Beijing Institute of Graphic Communication (21090121021), and the Key Educational Reform Project of Beijing Institute of Graphic Communication (22150121033/009).

References

[1] R. L. Rivest, L. Adleman, and M. L. Dertouzos, "On data banks and privacy homomorphisms," in *Proceedings of the 17th IEEE Annual Symposium on Foundations of Computer Science*, pp. 169–177, Academe Press, New York, NY, USA, 1978.

[2] C. Gentry, "Computing arbitrary functions of encrypted data," *Communications of the ACM*, vol. 53, no. 3, pp. 97–105, 2010.

[3] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[4] S. Goldwasser and S. Micali, "Probabilistic encryption," *Journal of Computer and System Sciences*, vol. 28, no. 2, pp. 270–299, 1984.

[5] J. D. Cohen and M. J. Fischer, "A robust and verifiable cryptographically secure election scheme," Yale University, New Haven, CT, USA, Department of Computer Science, 1985.

[6] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. 31, no. 4, pp. 469–472, 1985.

[7] M. Ajtai and C. Dwork, "A public-key cryptosystem with worst-case/average-case equivalence," in *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pp. 284–293, Rochester, NY, USA, 1997.

[8] T. Okamoto and S. Uchiyama, "A new public-key cryptosystem as secure as factoring," *International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 308–318, Springer, Berlin, Heidelberg, 1998.

[9] D. Naccache and J. Stern, "A new public key cryptosystem based on higher residues," in *Proceedings of the 5th ACM Conference on Computer and Communications Security*, pp. 59–66, Rochester, NY, USA, 1998.

[10] T. Sander, A. Young, and M. Yung, "Non-interactive cryptocomputing for nc/sup 1," in *Proceedings of the 40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039)*, pp. 554–566, IEEE, New York, NY, USA, October 1999.

[11] D. Boneh, E. J. Goh, and K. Nissim, "Evaluating 2-DNF formulas on ciphertexts," *Theory of Cryptography Conference*, pp. 325–341, Springer, Berlin, Heidelberg, 2005.

[12] Y. Ishai and A. Paskin, "Evaluating branching programs on encrypted data," *Theory of Cryptography Conference*, pp. 575–594, Springer, Berlin, Heidelberg, 2007.

[13] C. Gentry, S. Halevi, and V. Vaikuntanathan, "A simple BGN-type cryptosystem from LWE," *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 506–522, Springer, Berlin, Heidelberg, 2010.

[14] C. A. Melchor, P. Gaborit, and J. Herranz, "Additively homomorphic encryption with d-operand multiplications," *Annual Cryptology Conference*, pp. 138–154, Springer, Berlin, Heidelberg, 2010.

[15] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, pp. 169–178, ACM Press, Rochester, NY, USA, 2009.

[16] C. Gentry, *A Fully Homomorphic Encryption scheme*, Stanford University, Stanford, CA, USA, 2009.

[17] M. Yagisawa, "Fully homomorphic encryption without bootstrapping," *IACR Cryptol. EPrint Arch.*vol. 2015, p. 474, 2015.

[18] M. Van Dijk, C. Gentry, and S. Halevi, "Fully homomorphic encryption over the integers," *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 24–43, Springer, Berlin, Heidelberg, 2010.

[19] O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," *Journal of the ACM*, vol. 56, no. 6, pp. 1–40, 2009.

[20] C. Gentry, A. Sahai, and B. Waters, "Homomorphic encryption from learning with errors: conceptually-simpler, asymptotically-faster, attribute-based," *Annual Cryptology Conference*, pp. 75–92, Springer, Berlin, Heidelberg, 2013.

[21] E. F. Brickell and Y. Yacobi, "On privacy homomorphisms," *Workshop on the Theory and Application of of Cryptographic Techniques*, pp. 117–125, Springer, Berlin, Heidelberg, 1987.

[22] J. H. Cheon, J. Kim, M. S. Lee, and A. Yun, "CRT-based fully homomorphic encryption over the integers," *Information Sciences*, vol. 310, pp. 149–162, 2015.

[23] W. Wang and X. G. Xia, "A closed-form robust Chinese remainder theorem and its performance analysis," *IEEE Transactions on Signal Processing*, vol. 58, no. 11, pp. 5655–5666, 2010.

[24] K. Meng, F. Miao, Y. Xiong, and C. C. Chang, "A reversible extended secret image sharing scheme based on Chinese remainder theorem," *Signal Processing: Image Communication*, vol. 95, Article ID 116221, 2021.

[25] X. Yan, Y. Lu, L. Liu, J. Liu, and G. Yang, "Chinese remainder theorem-based two-in-one image secret sharing with three

decoding options," *Digital Signal Processing*, vol. 82, pp. 80–90, 2018.

[26] Y. Chen and P. Q. Nguyen, "Faster algorithms for approximate common divisors: breaking fully-homomorphic-encryption challenges over the integers," *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 502–519, Springer, Berlin, Heidelberg, 2012.

[27] D. Pei, A. Salomaa, and C. Ding, *Chinese Remainder Theorem: Applications in Computing, Coding, Cryptography*, World Scientific, Singapore, 1996.

[28] S. Kangsheng, "Historical development of the Chinese remainder theorem," *Archive for History of Exact Sciences*, pp. 285–305, Universiti Putro Malaysia, Selongor, Malaysia, 1988.

[29] J. S. Coron, T. Lepoint, and M. Tibouchi, "Scale-invariant fully homomorphic encryption over the integers," *International Workshop on Public Key Cryptography*, pp. 311–328, Springer, Berlin, Heidelberg, 2014.

[30] H. Cohn and N. Heninger, "Approximate common divisors via lattices," *The Open Book Series*, vol. 1, no. 1, pp. 271–293, 2013.

[31] J. C. Lagarias, "The computational complexity of simultaneous diophantine approximation problems," *SIAM Journal on Computing*, vol. 14, no. 1, pp. 196–209, 1985.

[32] H. W. Lenstra, "Factoring integers with elliptic curves," *Annals of Mathematics*, vol. 126, no. 3, pp. 649–673, 1987.

[33] J. S. Coron, A. Mandal, D. Naccache, and T. Mehdi, "Fully homomorphic encryption over the integers with shorter public keys," *Annual Cryptology Conference*, pp. 487–504, Springer, Berlin, Heidelberg, 2011.

[34] J. H. Cheon, J. S. Coron, J. Kim et al., "Batch fully homomorphic encryption over the integers," *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 315–335, Springer, Berlin, Heidelberg, 2013.

[35] J. H. Cheon, A. Kim, M. Kim et al., "Homomorphic encryption for arithmetic of approximate numbers," *International Conference on the Theory and Application of Cryptology and Information Security*, pp. 409–437, Springer, Cham, 2017.

[36] N. Howgrave-Graham, "Approximate integer common divisors," *International Cryptography and Lattices Conference*, pp. 51–66, Springer, Berlin, Heidelberg, 2001.