Diffraction Encryption: A Brand-New Chaotic Encryption Model

Anqi Hu (✉ 3283846941@qq.com)

Chongqing University of Posts and Telecommunications <https://orcid.org/0000-0001-7162-7693>

Xiaoxue Gong

Chongqing University of Posts and Telecommunications <https://orcid.org/0000-0002-7440-4003>

Lei Guo

Chongqing University of Posts and Telecommunications

Research Article

Keywords: Diffraction Encryption model, Plaintext wavification, Chaotic Encryption, Security

Posted Date: February 28th, 2022

DOI: <https://doi.org/10.21203/rs.3.rs-1389312/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Diffractive Encryption: A Brand-New Chaotic Encryption Model

Anqi Hu¹, Xiaoxue Gong^{1*} and Lei Guo¹

^{1*}School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, No.2, Chongwen Road, Nanan District, Chongqing, 400065, Chongqing, China.

*Corresponding author(s). E-mail(s): gongxx@cqupt.edu.cn;
Contributing authors: 3283846941@qq.com; guolei20211102@163.com;

Abstract

In this paper, we propose a brand-new chaotic encryption model that defines operations based on the data, which named Diffractive Encryption. We take a specific encryption algorithm as an example to prove the excellent effect and high security of the model. Unlike the existing encryption methods, the ciphertext generated by our encryption model hardly carries any data-related information, i.e. the plaintext does not directly participate in the encryption operation to form the ciphertext. Finally, we make a more objective security analysis of the encryption algorithm in this paper with the perspective of traditional security analysis. In the field of Chaotic Cryptography, this is a disruptive idea, and may become a new research direction.

Keywords: Diffractive Encryption model, Plaintext wavification, Chaotic Encryption, Security

1 Introduction

With the rapid development of Internet technology, the continuous improvement of computing speed, and the keep upgrading of cryptanalysis, some weaknesses of traditional cryptosystem are gradually exposed [1]. Therefore, focusing on the research and design of new cryptographic technologies to ensure the storage and transmission of information in cloud computing, big data and other new fields has become an important topic in the current cryptography [2–4]. As one of the three new cryptographic technologies, chaotic cryptography has been paid close attention by researchers since it was proposed [5, 6].

Chaotic cryptography is a cross-discipline that combines chaos theory with cryptography. Chaotic cryptography can provide fast and secure data protection [7]. Chaos is a typical nonlinear dynamic system.

It is highly sensitive to the initial state and control parameters of the system, and meets the requirements of key sensitivity in cryptography. Moreover, chaos has the characteristics of ergodicity, good pseudorandom, unpredictability and aperiodic, which is similar to the characteristics of confusion, diffusion and the number of rounds in cryptography [8]. The organic combination of chaos theory and cryptography will achieve complementary advantages, so as to design an encryption system with superior security performance. Therefore, chaotic cryptography has important theoretical significance and practical application value in protecting information security.

The theoretical basis of cryptographic technology is Shannon's "Communication Theory of Secrecy Systems" [9]. In the paper, all the theorems and formulas treat data and information as operands (objects to be manipulated and processed). However, this paper puts

forward a train of thought which is very different from Shannon's theorem for the first time, defines data and information (plaintext) as a kind of operation, and creatively puts forward a new method, plaintext wavification. Then, combining this new method with chaos, we creatively put forward a kind of chaotic encryption: Diffractive Encryption. Finally, our experiments show that this brand-new chaotic encryption method has high security and can resist common attacks.

In this paper, the definition of Diffractive Encryption is defined in Sect. 2. By comparing with the traditional encryption model, we creatively propose the diffractive encryption model for the first time in Sect. 3. In Sect. 4, a practical application example is provided for the Diffractive Encryption model, and the encryption and decryption algorithm of the example is analyzed and explained. In Sect. 5, the experimental results of the Diffractive Encryption model are shown, and the security of the experimental results is analyzed. In Sect. 6, we summarize this innovative Diffractive Encryption model and look forward to the future.

2 Theory of Diffractive Encryption

2.1 Plaintext wavification

Plaintext wavification is the process of mapping plaintext to a certain encryption operation and implementing the operation with corresponding diffusion strategy. The encryption operation mentioned here requires it to be reversible, or in the case of additional information, it is easy to find the inverse (such as trapdoor function, etc). Any operation that have higher security and at same time can be decrypted more conveniently could be used for plaintext wavification. In this paper, we use more intuitive bitwise exclusive OR (XOR) and exclusive NOR (XNOR) operations.

2.2 Diffractive Encryption

In Sect. 2.1, for the first time, a subversive, new type of plaintext wavification that defines operations based on the data itself is proposed. On the basis of plaintext wavification, a new encryption and decryption algorithm based on chaotic sequence is proposed. The object of this algorithm is the random sequence generated by chaos, which has no relationship with the plaintext. The random sequence is waved in plaintext, and the result is a corresponding permutation strategy

to form ciphertext. This kind of encryption method is called "Diffractive Encryption".

Here, we quote the physics term: diffraction, which is used to vividly illustrate the processing and change of data and information (plaintext) in the process of encryption. Physically, the effects of diffraction of light were first carefully observed and characterized by Francesco Maria Grimaldi [10], who also coined the term diffraction, from the Latin *diffingere*, 'to break into pieces', referring to light breaking up into different directions. In the process of our encrypted plaintext processing:

1. The plaintext is first transformed into a defined operation, which is similar to the diffraction effect of light in which the propagation direction of the wave is broken.
2. Implement the operation according to the grouping, and construct the ciphertext through the replacement strategy, so that there is no correspondence between the plaintext and the ciphertext, which is similar to the phenomenon that waves bend and scatter in different directions in the diffraction effect of light.

It is very vivid to use the diffraction phenomenon to describe our plaintext processing in the encryption procedures. Therefore, our encryption method is called "Diffractive Encryption". In Sect. 4, a practical application example is provided for the Diffractive Encryption model, and the encryption and decryption algorithm of the example is analyzed and explained.

In particular, in the process of forming ciphertext, plaintext information is not directly involved in operations as operands as in traditional encryption. In this paper, we define data and information (plaintext) as an operation through plaintext wave, and the operand of this operation is a random sequence that has nothing to do with information. The specific process of encryption and decryption is to define the plaintext as a binocular operation of a specific rule, and perform the operation on a random sequence, and obtain the ciphertext by permuting the result of the operation. The decryptor uses the same rules to inverse the received data to obtain the plaintext. Due to the use of plaintext wave, any ciphertext information obtained by the eavesdropper is only the result of the operation of the random sequence, which has no direct relationship with the plaintext, so the plaintext cannot be analyzed. That is, the eavesdropper cannot obtain any details of the plaintext wavification during the eavesdropping process, so that the plaintext cannot be recovered

from the ciphertext. On the other hand, since the random sequence directly participates in the generation of the ciphertext (as an operand), if an eavesdropper wants to crack the ciphertext, he has to obtain the random sequence itself to explore the details of the plaintext wave. Therefore, the security of random sequences becomes the second barrier to protect Diffractive Encryption. In short, there are two barriers between ciphertext and plaintext, namely plaintext wave and random sequence, to jointly maintain the security of Diffractive Encryption.

3 Encryption model

For a long time, in the field of encryption, information or data has been regarded as the object of operation, that is, to operate and calculate information or data for a certain algorithm. In a broad mathematical sense, information and data are treated as calculation objects, which seems to be a matter of course. We use a simplified model to describe the traditional encryption process, and get Fig. 1.

Fig. 1 Schematic diagram of traditional encryption/decryption model

As shown in Fig. 1, X represents plaintext information, F represents a certain encryption algorithm, S represents secret key of F , and C represents ciphertext. The encryption process is: the encryption algorithm F uses the particularity of the key S to confuse and diffuse the plaintext X , and obtain the ciphertext C with the plaintext information hidden. However, the ciphertext obtained after encryption retains different degrees of statistical characteristics of the plaintext according to different algorithms, and there is still a risk of exposing the plaintext information. The eavesdropper may analyze the plaintext X from the ciphertext C , or obtain the key S through some means, thereby cracking the entire encryption model. On the contrary, the core idea of the Diffractive Encryption model in this paper is to do the opposite thing: treat information and data as some kind of operation, using random sequence as the operand of the operation, and the

result of the operation as the ciphertext of encryption. Similarly, we use a simplified model to describe the Diffractive Encryption, and get Fig. 2.

Fig. 2 Diffractive Encryption model

As shown in Fig. 2, where X represents plaintext information, U represents operands irrelevant to X , F_{x_i} ($x_i \in X$) represents a set of operations defined by plaintext information X through several layers of effective mapping or translating, and C represents ciphertext. In Sect. 4, the example of Diffractive Encryption, we use random sequence as U . In the process of encryption, the most important thing is not to encrypt plaintext X directly, but how to define X as some operation rules F_{x_i} . Then calculate by using rules F_{x_i} to the X -irrelevant operand U , and the result C is obtained as ciphertext. In other words, we map the plaintext to a certain operation, taking the random sequence as its operand, and the result of this operation as the ciphertext. The advantage of this new encryption model is that neither the ciphertext nor the operand contains information related to the plaintext. Therefore, using our Diffractive Encryption model in transmission, even if the channel is eavesdropped, it will never bring the potential hazard of information leakage. This will greatly reduce the risk of plaintext information exposure.

The decryption model of Diffractive Encryption is shown in Fig. 3. It is the inverse process of the encryption. When decrypting, the operand U and the ciphertext C are required. According to the definite effective mappings of encryption, do the inverse mapping or analyzing, and then the plaintext X could be obtained. The largest difference of decryption between the Diffractive Encryption and the traditional encryption is that it is necessary to know both the ciphertext C and the operand U . However, just only getting the ciphertext C , you will never get any information related to the plaintext anyway.

Fig. 3 The decryption model of Diffractive Encryption

4 Diffractive Encryption model application example algorithm analysis

The application example of the Diffractive Encryption model in this paper is edited and tested in the PyCharm Community Edition 2021.1.2 x64 environment. We take advantage of the strong chaotic behavior of the Chebyshev map with variable parameters to generate a chaotic sequence, and use it as our operand.

It can be represented by the Chebyshev map with modified control parameters:

$$X_{n+1} = \cos(w_n \arccos X_n) \quad (1)$$

$$w_{n+1} = w_n + n/T \quad (2)$$

where $X_n \in (-1, 1)$, $w_n \geq 2$, and T is a constant greater than 20000.

Chaos mapping is a pseudo-random process sensitive to initial values. As long as the initial values and rules are set, both the encryption side and the decryption side can obtain the same chaotic sequence. So, chaotic maps naturally have perfect synchronization characteristics, which is the main reason why we choose chaotic sequences. To our Diffractive Encryption model, its security performance would be enhanced by improving the security of operands. Therefore, the Chebyshev chaotic sequence involved in the encryption operation is obtained by the method of changing the initial parameters with different value.

Figure 4 shows this chaotic sequence. The advantage of using variable value as parameter is that even if the length is the same, the sequence obtained each time will be different. What's more, its initial value X_0 also has greater randomness. Figure 5 shows 8 chaotic sequences (each length is 50) generated by changing parameter value as Eq. 1. It can be seen that the chaotic sequence obtained each time is quite different.

Fig. 4 Improved Chebyshev map (ICM) chaotic sequence (length is 200)

Fig. 5 8 chaotic sequences (each length is 50) generated by changing parameter value

4.1 Encryption algorithm

Bitwise exclusive OR (XOR) and bitwise exclusive NOR (XNOR) are common binary operations. The XOR operation is expressed as:

$$(x + y) \bmod 2 \quad (3)$$

where $x, y \in \{0, 1\}$.

The XNOR operation is expressed as:

$$(x + y + 1) \bmod 2 \quad (4)$$

where $x, y \in \{0, 1\}$.

Based on these two bitwise operation, we will redefine the operation rules corresponding to the plaintext. Since the description of redefinition process requires symbols to assist, we will explain them in encryption algorithm steps. Of course, in our Diffractive Encryption model proposed in this paper, the definition content should be regarded as confidential

Fig. 6 Define m_1 and m_2 as operations to realize the operation of the encryption process

content, and its public description here is to provide an example as a proof of model application.

We use the the Chebyshev map with modified control parameters as the pseudo-random sequence generator in this case. Suppose the chaotic sequence generated by the chaotic generator is $K = \{k_1, k_2, k_3, k_4\}$ (divide K into 4 equal-length sub-sequences in turn), the encryption algorithm steps are as follows:

Step 1: Suppose the information sequence to be encrypted is: $M = \{m_1, m_2, m_3, m_4\}$ (divide M into 4 equal-length sub-sequences in turn), and M and K are of equal length. Obviously, the subsequences k_i and m_i are also of the same length.

Step 2: Define operations (m_1) and (m_2). Operation (m_1) means: define bit ‘0’ in the binary sequence m_1 as an XNOR operation, bit ‘1’ is defined as an XOR operation, and the operands are k_2, k_4 ; similarly, operation (m_2) means: define bit ‘0’ in the binary sequence m_2 as an XNOR operation, bit ‘1’ is defined as an XOR operation, and the operands are k_1, k_3 . Combine the results of the two operations into ciphertext C_1 , which is expressed as:

$$C_1 = k_2 (m_1) k_4 + k_1 (m_2) k_3 \quad (5)$$

As shown in Fig. 6, the operation process defined by m_1 is: traverse each bit of m_1 , if the bit of m_1 is 0, the corresponding bits of k_2 and k_4 are XNORed; if the bit of m_1 is 1, the corresponding bits of k_2 and k_4 are XORed. The operation defined by m_2 is the same as m_1 , except that the operands are k_1 and k_3 . ‘+’ represents a merge operation, and the operation results of k_2 and k_4 and the operation results of k_1 and k_3 are connected and merged to obtain the ciphertext C_1 .

Step 3: Define operations [m_3] and [m_4]. Operation [m_3] means: define the ‘0’ in the binary sequence m_3 as an XOR operation, and ‘1’ as the XNOR operation, and the operands are k_1, k_4 ; similarly, the operation [m_4] means: define the ‘0’ in the binary sequence m_4

Fig. 7 Define m_3 and m_4 as operations to realize the operation of the encryption process

as an XOR operation, and ‘1’ as the XNOR operation, and the operands are k_2, k_3 . Combine the results of the two operations into the ciphertext C_2 , which is expressed as:

$$C_2 = k_1 [m_3] k_4 + k_2 [m_4] k_3 \quad (6)$$

As shown in Fig. 7, the calculation process defined by m_3 is: traverse each bit of m_3 , if the bit of m_3 is 0, the corresponding bits of k_1 and k_4 are XORed; if the bit of m_3 is 1, the corresponding bits of k_1 and k_4 are XNORed. The operation defined by m_4 is the same as m_3 , except that the operands are k_2 and k_3 . ‘+’ represents a merge operation, and the operation results of k_1 and k_4 and the operation results of k_2 and k_3 are connected and merged to obtain the ciphertext C_2 .

Step 4: Take the bits in C_1 as the odd bits of the ciphertext C , and the bits in C_2 as the even bits of the ciphertext C to form the ciphertext C .

Through the above four steps of calculation and encryption process, the length of the ciphertext C obtained is the same as the length of the input plaintext M .

4.2 Decryption algorithm

The decryption algorithm in this paper is the inverse process of the encryption algorithm. Similarly, when decrypting, the Chebyshev map with modified control parameters generates a chaotic sequence of $K = \{k_1, k_2, k_3, k_4\}$ (K is divided into 4 subsequences of equal length in turn). The K here is totally the same as the encryption. It should be noted that the chaotic sequence K is very critical, without which it is difficult to parse the ciphertext. Thereby, the Decryption Algorithm steps are as follows:

Step 1: Split the ciphertext C by parity bits to obtain C_1 and C_2 . Knowing from Eqs. 5 and 6, C_1 and C_2 are combined ciphertexts. By splitting in half, the ciphertext sequence C is finally decomposed into: $C = \{c_1, c_2, c_3, c_4\}$, Here, according to Eqs. 5 and 6, c_1

Fig. 8 Use ciphertext c_1 , c_2 and operands to perform reverse operation for decryption

Fig. 9 Use ciphertext c_3 , c_4 and operands to perform reverse operation for decryption

is the result of $k_2 (m_1) k_4$, the same to c_2 and $k_1 (m_2) k_3$, c_3 and $k_1 [m_3] k_4$, c_4 and $k_2 [m_4] k_3$.

Step 2: The ‘0’ in the binary sequence c_1 performs the XNOR operation, the ‘1’ performs the XOR operation, and the operands are k_2 , k_4 ; the ‘0’ in the binary sequence c_2 performs the XNOR operation, and the ‘1’ performs the XOR operation, and the operands are k_1 , k_3 . Combine the results of the two inverse operations into plaintext M_1 ($M_1 = \{m_1, m_2\}$), expressed as:

$$M_1 = k_2 (c_1) k_4 + k_1 (c_2) k_3 \quad (7)$$

As shown in Fig. 8, the decryption process of M_1 is: traverse each bit of c_1 , if the bit of c_1 is 0, the corresponding bits of k_2 and k_4 are XNORed; if the bit of c_1 is 1, the corresponding bits of k_2 and k_4 are XORed. The inverse process of c_2 is the same as that of c_1 , except that the operands are k_1 and k_3 . ‘+’ represents a merge operation, and the inverse operation results of k_2 and k_4 and the inverse operation results of k_1 and k_3 are connected and merged to obtain the plaintext M_1 .

Step 3: The ‘0’ in the binary sequence c_3 performs the XOR operation, the ‘1’ performs the XNOR operation, and the operands are k_1 , k_4 ; the ‘0’ in the binary sequence c_4 performs the XOR operation, the ‘1’ performs the XNOR operation, and the operands are k_2 , k_3 . Combine the results of the two inverse operations into plaintext M_2 ($M_2 = \{m_3, m_4\}$), which is expressed as:

$$M_2 = k_1 [c_3] k_4 + k_2 [c_4] k_3 \quad (8)$$

Fig. 10 Time overhead of plaintext encryption

As shown in Fig. 9, the decryption process of M_2 is: traverse each bit of c_3 , if the bit of c_3 is 0, the corresponding bits of k_1 and k_4 are XORed; if the bit of c_3 is 1, the corresponding bits of k_1 and k_4 are XNORed. The inverse process of c_4 is the same as that of c_3 , except that the operands are k_2 and k_3 . ‘+’ represents a merge operation, and the inverse operation results of k_1 and k_4 and the inverse operation results of k_2 and k_3 are connected and merged to obtain the plaintext M_2 .

Step 4: Combine plaintext M_1 and M_2 to obtain plaintext M .

Through the decryption process of the inverse operation of the above four steps, the ciphertext C is successfully restored to the correct plaintext M .

5 Experimental results and safety analysis

5.1 Display of experimental results

5.1.1 Time cost comparison

We select the first 10,000 bytes of the e-book “DESCRIPTION OF FARMER OAK - AN INCIDENT” for encryption. It runs on Windows 10 Home Edition. The type of the CPU is Intel(R) Core(TM) i5-10300H @ 2.50GHz. The RAM size is 16GB. The software platform is PyCharm 2021 (Community Edition). Except that the code of bit operation is optimized in C language, the rest is Python code. Block Size is defined as the length of the message m_i ($i = 1, 2, 3, 4$) that is, 1/4 of the message M length (in Byte). When Block Size takes different values, the time cost of encryption is shown in Fig. 10.

Fig. 11 Time overhead of plaintext encryption

The corresponding decryption time cost is shown in Fig. 11. Obviously, the time of decryption is almost the same as that of encryption.

From Fig. 10 and Fig. 11, when the Block Size is greater than or equal to 2, as the Block Size increases, the encryption and decryption speed becomes faster and faster. After the Block Size is larger than 40 bytes, the encryption time of 10,000 bytes is no more than 1 second, which shows that the encryption and decryption process of this paper is highly efficient.

5.1.2 Comparison before and after encryption of the same plaintext with different times

Since each encryption uses a different chaotic sequence as the operand, the ciphertext is also different for the same plaintext, as shown in Table 1.

As shown by the experimental results in Table 1, the application example using the Diffractive Encryption model has realized the effect of one-time pad, reflecting the high security of the Diffractive Encryption model.

5.2 Security analysis

5.2.1 Comparison with Shannon's theorem

Shannon's theorem [9] in "Communication Theory of Secure System" is as follows:

Theorem 1 Suppose that the plaintext, ciphertext, and key are respectively M , $M, C, K = \{0, 1\}^n$, the encryption algorithm for one-time pad is E , and a set of keys $k =$

$(k_1, k_2, k_3, k_4) \in K$, use exclusive OR operation to encrypt a set of plaintexts $m = (m_1, m_2, m_3, m_4) \in M$, m and k are independent of each other [9]:

$$E(m) = m \oplus k = (m_1 \oplus k_1, m_2 \oplus k_2, \dots, m_n \oplus k_n) \quad (9)$$

E is unconditionally safe, if and only if it is evenly distributed on K .

"Communication Theory of Secrecy Systems" laid a theoretical foundation for the study of cryptography, and cryptography became a science. Theorem 1 is an important theoretical basis for modern encryption algorithms. By comparison, the encryption algorithm proposed in this paper differs from Shannon's description in Theorem 1 in two points:

1. We define operations differently: we define data as operations, and use random numbers generated by chaos as operands of operations. Due to the randomness of the operands and the non-uniqueness of the operation method, the operation we define belongs to the one-time pad encryption algorithm. In Sect. 5.1.2, we encrypt the same plaintext several times, and the result can be a good indication that our encryption is a one-time pad. Also, there is no dependency between our data and the operand sequence, they are completely independent of each other.
2. In our encryption operation, the plaintext is not involved, that is, the operands that participate in the operation are the operands unrelated to the plaintext. This is our most obvious difference from Shannon's theorem. From a security point of view, the plaintext does not directly participate in the operation. Even if the attacker obtains the ciphertext, he only obtains a bunch of operation results unrelated to the plaintext. This will be meaningless and useless for attackers to speculate and decipher the plaintext. Through our encryption method, the plaintext information is highly concealed and secure.

For decades, people have been keen to follow the pattern of Shannon, even taking it for granted. Almost all modern encryption algorithms are built under Shannon's system and framework. However, for the first time in decades, this paper puts forward ideas and ideas that are very different from Shannon's theorem, which is the value of this paper. The ideas proposed in this paper may change some of the status quo, and may also open up a new chapter in Chaotic Cryptography.

Table 1 Encryption results of the same plaintext with different times

Plaintext:“ T. HARDY><T Madding Crowd(Penguin 1978)”	
Encryption times	Ciphertext
1	\xc4\xc2\x9b\x07\xf8<\xff\xbbM\xbc\xa7\xca\x80\t3xX!\x9cq\x83 8\xfc\xci\x01\xb7 \xfd\xd1‘\xbb)\xde\xb5\x0fTr`‘\x9a\xcc
2	(Z\xd5v\xffP\x0b\xf1\x04/\x0f\x9c\xea\x13\x88\xa7\x06\xca\x8e\x09`‘\xef\x06 \x84RD\x05\xb2\x9e\xae\xaf\x8bo\xf2M\xe4\xf2\x03\xdf
3	\xc2\xa5\x88`‘\x85*\x95\xcd‘\xf3\xd0\xe8>\x05*&\x9e\x80\x81\x04\xf7S\x012F \x87h\x05\x0c\x04\x8d\xcc\x197\xa3\x01`‘]
4	\x17\xdc\x88\x14+S\x10\x9c\x11\x93-\xf7~\x05vvQ\x84\xa9\x80\xbc \xa1\x86 \xc7\xae<\xe7\xd9\t.\xfe\xac\xfdh\xe7o\xf0v
5	]5?\x0f\xdf\xf9\$\xb1g\xda\x9cJ\xfe;\x9b\x86\xb5\r\xfd\x94\x95\xe3\xeaW\x08 \x8a\x16\x84J\x83\xb9B\x84\xeb\xaa\xbb\x80TS\x8b
6	\xbb3\x04\xab\xb2\xb7\x86\xe2\xacJ\x0c\x99\xdcB#\x85\xce\xe9\xb1\x82\xf7x ~O^\x90Nw\x8a\xa6\x96\xa0\xcb\xf1\x85\x01\x92\x15\x90\xe4
7	\x0f\xab\x9d*\x1d\x99\x8b\xaa\x0f\x9b\xce F\xcdY=\xca\x04F\x07\xaf\x891G \xfb\x0b\xf4\xd4\xe28\x07?,\xfa\x9e\xaa\x0b\xee\xe2(
8	\xaf\xbb\x09b\x03\xeb\x9e\xca>a\x8a\x04\x932\x94\xaa\x03\xbf/676\xfa\x0f \x8f\xa6\x92\x8bcEpo>\x80s\x1d\xa0\x08\xe3
9	\xb7\xe0p\x00\xe9H\xe1\xda\x04\xab\x09\xd6\x84E6\xeeP~\x18\xafj&\xb2\x15 \x85g\x8fr\xeb}d\xf2v\x19 \x07!\x02”q

5.2.2 Security analysis of the Diffractive Encryption model

The security performance of traditional encryption mode mainly depends on key space, key sensitivity, plaintext sensitivity, randomness evaluation, data probability distribution, computational complexity, etc. These evaluation factors can evaluate well the degree of proliferation and confusion of the plaintext of the encryption algorithm in the encryption process. However, in our new Diffractive Encryption model proposed, the plaintext does not directly participate in diffusion and confusion, but is defined as a certain operation. The operation process to operands is the process of encryption. Thereby the ciphertext generated in this process is different from the traditional ciphertext. Since the ciphertext generated in this paper does not contain the statistical characteristics of the plaintext, the method of using statistics to analyze the ciphertext is basically unsuitable and invalid. In this respect, the risk of information leakage will be greatly reduced or even disappeared.

Another important security guarantee of the Diffractive Encryption model comes from the flexibility of operation. Operations defined based on plaintext information are not limited to numeric operations. They can be list mapping, address mapping, or non-linear relationship mapping, etc. In order to describe the encryption process of the algorithm concisely, the encryption case proposed in Sect. 4 only uses the most common XOR and XNOR operations as a set of defined operations for plaintext. In practical applications, we should define plaintext as more complex, and novel multi-object operations. This defined operation is an important “key” and never made public. In this way, we’ll get not only the complexity of the digital key, but also the complexity of the operation logic and the diversity of mapping types. The multi-dimensional encryption feature makes the computer totally impossible to enumerate and crack from a single direction.

Besides the flexibility and complexity of the operations defining, the numbers of encryption operations terms can also improve the security of encryption.

The more operands involved in the calculation, the more complicated the calculation process will be. This makes the new encryption model that uses calculation as an encryption method have higher security performance.

In addition, the traditional encryption model can improve the security performance of the Diffractive Encryption model. From a theoretical analysis, as long as the defined operation is complex enough, the operand can be a pseudo-random chaotic sequence, a random constant sequence and even the public data. However, the disclosure of the operands may make it possible for the eavesdroppers to infer the calculation process by comparing with the ciphertext. Therefore, the traditional encryption method is still effective in protecting the operands. When the definition of calculation is less complicated, the traditional encryption method still could be used as an auxiliary algorithm to improve the security.

To sum up, by proposing the Diffractive Encryption model in this paper, we concealed the plaintext information effectively from a new perspective and eventually achieved excellent security.

5.2.3 Key space analysis

Key space refers to the range of encryption key sizes. Unique keys are usually counted in bits, that is, the number of bits. For conventional encryption algorithms, the longer the bits of the key, the larger the key space.

For Diffractive Encryption, the operation defined by the plaintext has nothing to do with the plaintext operand sequence, which together constitute the coding space. Although the operation and operand sequence are equivalent to the key of the conventional encryption algorithm, in the process of enumerating the ciphertext, it is not necessary to simulate the operand sequence and operation, but the encoding space is used. The encoding space is mainly determined by two factors. The first factor is the operation defined by plaintext, because the operation not only specifies the number and length of the operands, but also specifies the range set of the operation result (denoted as A). The number of elements of A directly determines the encryption result of each plaintext. One part here is the minimum unit that specifies that the text is mapped to the operation, which can be 1 bit or several bits. For the convenience of description, we denote this minimum unit as i . The second factor is the length L of the original plaintext, because the longer

the plaintext, the more copies that can be mapped into operations, and the more complex the ciphertext generated. Therefore, the encoding space of Diffractive Encryption is:

$$S = (card A)^{\frac{L}{i}} \quad (10)$$

Taking the encryption algorithm given in this article as an example, although it is only a very simple example, it is enough to highlight its advantages. The final value range set size is $card A = 2$, which is mapped to the smallest unit of operation $i = 1$. When the length of the plaintext reaches 1MB, the encoding space $S = 2^{8388608}$, which is enough to resist exhaustive attacks.

5.2.4 Key sensitivity analysis

For an encryption algorithm, key sensitivity is one of the important evaluation criteria, that is, a small change in the key can cause sufficient changes in the ciphertext. From a statistical point of view, the rate of change of the ciphertext must close to 50% [11]. The Diffractive Encryption proposed in this paper, the intuitive embodiment of the change of the ciphertext, is the change of the result of each operation. For the same piece of data, there are mainly two factors that can cause changes in the calculation results. The first factor is the difference in operation, which will lead to changes in the results. That is, the same data can be mapped to different operations. This method is also reflected in the examples given in this paper: the same data, when it appears in the m_1 subsequence, use the operation (m_1) ; when it appears in the m_3 subsequence, use the operation $[m_3]$. The second factor is the difference in the number of operands, which can also lead to differences in the results. In the example given in this paper, the random sequence generated by the chaotic map is used as the operand, and the operand used in each calculation is random, which satisfies the requirement of key sensitivity well.

In order to test the key sensitivity of the algorithm in this paper, we encrypt each file in the Calgary Corpus 9 times. Except for the first time, the average bit-change rate of ciphertext in last 8 times is calculated. The change rate of initial values and parameters of chaotic map is 10^{-15} , which can fully deal with Brute-Force Attack [11]. The bit-change rate of each file is shown in Table 2.

5.2.5 NIST testing and safety assessment

Randomness is an important evaluation factor for the security of encryption algorithms [12]. Since the

Table 2 The average bit-change rate of the last 8 encrypted ciphertexts

File	ΔX_0	ΔW_n	Average bit-change rate
bib	0	1×10^{-15}	0.4986427436440678
	1×10^{-15}	0	0.4996226165254238
book1	0	1×10^{-15}	0.500249743852459
	1×10^{-15}	0	0.5004610655737705
book2	0	1×10^{-15}	0.49896900614754097
	1×10^{-15}	0	0.4992635758196722
news	0	1×10^{-15}	0.49962218237704914
	1×10^{-15}	0	0.5007876536885245
paper1	0	1×10^{-15}	0.5001888020833334
	1×10^{-15}	0	0.49767578125
paper2	0	1×10^{-15}	0.4987825520833334
	1×10^{-15}	0	0.49930338541666663
paper3	0	1×10^{-15}	0.5011002604166667
	1×10^{-15}	0	0.5003515625
paper4	0	1×10^{-15}	0.4996029713114754
	1×10^{-15}	0	0.49991675204918035
paper5	0	1×10^{-15}	0.50125
	1×10^{-15}	0	0.4995768229166667
paper6	0	1×10^{-15}	0.4977213541666667
	1×10^{-15}	0	0.5013997395833334
pic	0	1×10^{-15}	0.500359122983871
	1×10^{-15}	0	0.4998235887096774
progc	0	1×10^{-15}	0.4995247395833333
	1×10^{-15}	0	0.5007486979166667
progl	0	1×10^{-15}	0.49986328124999996
	1×10^{-15}	0	0.50046875
progp	0	1×10^{-15}	0.49727886652542375
	1×10^{-15}	0	0.5007613877118644
trans	0	1×10^{-15}	0.49878906250000005
	1×10^{-15}	0	0.49990234375

chaotic sequence is a pseudo-random sequence, some chaotic systems with poor performance may even have obvious short-period phenomena, which easily leave security loopholes. Therefore, many research institutions make standard analysis tools based on statistical analysis methods commonly used in statistics to make a confidence evaluation of the degree of encryption randomness. NIST SP800-22 [13] is the NIST statistical test suite provided by the National Institute of Standards and Technology of America. The kit contains 15 test contents. The test result of each content has a p-value. In this paper, we set p to 0.01 for the example of the Diffractive Encryption model, and use the default parameters of the suite to perform statistical tests on three sets of encrypted data from different plaintext, as shown in Table 3. All the p-values of the three groups of ciphertexts in Table 3 are greater than 0.01, and they have successfully passed all NIST test items.

It can be seen from the NIST test results in Table 3 that the ciphertext according to Diffractive Encryption model has strong randomness. That means the data will be safe during transmission.

6 Conclusion

This paper proposes a brand-new chaotic encryption model for the first time, which named Diffractive Encryption. The specific example of the proposed model defines information as a bitwise operation, using random sequences as operands, and the result of the operation as ciphertext, thus obtaining a perfect encryption effect. In traditional encryption, plaintext participates in the encryption operation directly. The essence of Diffractive Encryption is that the plaintext does not participate in the operation directly, but is mapped to an encryption rule and algorithm. In the field of Chaotic Cryptography, this is a disruptive idea, and may become a new research direction.

In the example provided for the Diffractive Encryption model of this paper, the encryption process based on bit operation is relatively simple. In the follow-up work of future, we can define more complex mathematical operations for encryption based on the information and data itself. At the same time, we also need to expand the operation itself to a multi-objective operation with more operands, and there can even be hundreds or thousands of operands, not just this two-operand Binocular Operation, so as to ensure the security of our Diffractive Encryption model.

Acknowledgments. This research was supported by the National Natural Science Foundation of China under Grant No.62075014.

Declarations

Funding

This work was supported by the National Natural Science Foundation of China (Grant numbers 62075014).

Competing interests

The authors have no relevant financial or non-financial interests to disclose.

Authors' contributions

Conceptualization: Anqi Hu; Funding acquisition: Xiaoxue Gong, Lei Guo; Investigation: Anqi Hu;

Table 3 NIST test results

Statistical test	P-value1	P-value2	P-value3
Monobit	0.807385	0.390655	0.743878
Frequency within block	0.762062	0.801969	0.614161
Runs	0.268225	0.643343	0.353265
Longest run ones in a block	0.172359	0.477006	0.871262
Binary matrix rank	0.685433	0.854232	0.984959
Dft	0.110031	0.680968	0.321882
Non overlapping template matching	0.999998	0.999999	0.999999
Overlapping template matching	0.599492	0.832776	0.745841
Maurers universal	0.308000	0.025960	0.019893
Linear complexity	0.022162	0.700292	0.971343
Serial	0.871983	0.505585	0.227909
Approximate entropy	0.871965	0.669163	0.227380
Cumulative sums	0.714159	0.430759	0.532511
Random excursion	0.091986	0.121274	0.082175
Random excursion variant	0.023376	0.283544	0.211629
Pass rate	15/15	15/15	15/15

Methodology: Anqi Hu; Project administration: Xiaoxue Gong, Lei Guo; Software: Anqi Hu; Supervision: Xiaoxue Gong, Lei Guo; Validation: Anqi Hu; Writing – original draft: Anqi Hu; Writing –review & editing: Anqi Hu, Xiaoxue Gong, Lei Guo.

Data availability

Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Code availability

All code of this paper will be made available on reasonable request.

References

- [1] Lingfeng, Q., *et al.*: Security analysis of multiple permutation encryption adopt in reversible data hiding. *Multimed. Tools Appl.* **79**(39), 29451–29471 (2020)
- [2] Varri, U., Pasupuleti, S., Kadambari, K.: A scoping review of searchable encryption schemes in cloud computing: taxonomy, methods, and recent developments. *J. Supercomput.* **76**(4), 3013–3042 (2020)
- [3] Sajay, K., Babu, S.S., Vijayalakshmi, Y.: Enhancing the security of cloud data using hybrid encryption algorithm. *J. Ambient Intell. Human. Comput.*, 1–10 (2019)
- [4] Raza, S.F., Satpute, V.: A novel bit permutation-based image encryption algorithm. *Nonlinear Dyn.* **95**(2), 859–873 (2019)
- [5] Kaur, M., Kumar, V.: A comprehensive review on image encryption techniques. *Arch. Computat. Methods Eng.* **27**(1), 15–43 (2020)
- [6] Özkaynak, F.: Brief review on application of nonlinear dynamics in image encryption. *Nonlinear Dyn.* **92**(2), 305–313 (2018)
- [7] Senyange, B., Manda, B.M., Skokos, C.: Characteristics of chaos evolution in one-dimensional disordered nonlinear lattices. *Phys. Rev. E* **98**(5), 052229 (2018)
- [8] Baptista, M.: Cryptography with chaos. *Phys. Lett. A* **240**(1-2), 50–54 (1998)

- [9] Shannon, C.E.: Communication theory of secrecy systems. *Bell Syst. Tech. J.* **28**(4), 656–715 (1949)
- [10] Cecchini, R., Pelosi, G.: Diffraction: the first recorded observation. *IEEE Antennas Propag. Mag.* **32**(2), 27–30 (1990)
- [11] Mishra, M., Mankar, V.: A chaotic encryption algorithm: Robustness against brute-force attack. In: *Advances in Computer Science, Engineering & Applications*, pp. 169–179. Springer, Heidelberg (2012)
- [12] Avaroğlu, E., Tuncer, T., Özer, A.B., Ergen, B., Türk, M.: A novel chaos-based post-processing for trng. *Nonlinear Dyn.* **81**(1), 189–199 (2015)
- [13] Bassham III, L.E., Rukhin, A.L., Soto, J., Nechvatal, J.R., Smid, M.E., Barker, E.B., Leigh, S.D., Levenson, M., Vangel, M., Banks, D.L., et al.: Sp 800-22 rev. 1a. a statistical test suite for random and pseudorandom number generators for cryptographic applications. National Institute of Standards & Technology (2010)