High Security Sequence Design for Differential Frequency Hopping Systems

Rui Chen, Jia Shi, Lie-Liang Yang, *Fellow, IEEE,* Zan Li, *Senior Member, IEEE,* and Lei Guan

***Abstract*—Differential frequency hopping (DFH) technique is widely used in wireless communications by exploiting its capabilities of mitigating tracking interference and confidentiality. However, electronic attacks in wireless systems become more and more rigorous, which imposes a lot of challenges on the DFH sequences designed based on the linear congruence theory, fuzzy and chaotic theory, etc. In this paper, we investigate the sequence design in DFH systems by exploiting the equivalence principle between the G-function algorithm and the encryption algorithm, in order to achieve high security. In more details, first, the novel G-function is proposed with the aid of the Government Standard (GOST) algorithm and the Rivest-Shamir-Adleman (RSA) algorithm. Then, two sequence design algorithms are proposed, namely, the G-function assisted sequence generation (GF-SG) algorithm, which takes the full advantages of the symmetric and asymmetric encryption algorithms, and the high order G-function aided sequence generation (HGF-SG) algorithm, which is capable of enhancing the correlation of the elements in a DFH sequence. Moreover, the security and ergodicity performance of the proposed algorithms are analyzed. Our studies and results show that the DFH sequences generated by the proposed algorithms significantly outperform the sequences generated by the reversible hash algorithm and affine transformation in terms of the uniformity, randomness, complexity and the security.**

***Index Terms*—Differential frequency hopping, G-function, encryption algorithm, hybrid encryption, security, complexity.**

I. Introduction

With the rapid development of information technology, wireless communication has brought great convenience to people's lives. However, due to the openness of wireless channels, data transmission is facing more and more security risks. There are possibly various attacks, such as information disclosure of mobile users, eavesdropping and network criminals, etc., which impose serious threats to the future wireless communications. Therefore, it is imperative and vital to study the data and information security in wireless systems.

It is well known that the conventional encryption technology is only for data encryption and mainly used for data protection in the network layer and layers above, and it is in general independent of the physical layer. What's more, the existing studies mainly assume that the channel between encryption and decryption is perfect for error-free transmission. On the other side, based on the wiretap channel model and considering the wide applications of wireless communications, there are a lot of references having investigated the security issues from the perspective of the physical layer, where the uniqueness and reciprocity of physical channels are exploited to implement information encryption and identification of legitimate users. Furthermore, physical layer security has been exploited as a supplement to the upper level security, so as to enhance the security performance of the whole wireless systems.

In literature, various physical layer security techniques, have been proposed and investigated [1], [2], including spread spectrum communication [3], channel coding assisted the encryption technology [4], modulation based encryption technology [5], etc. Among this, spread spectrum and frequency hopping (FH) techniques have been widely used in the various kinds of communication systems owing to their good anti-interference ability [6], [7]. However, conventional FH techniques are prone to the tracking jamming, which becomes severer nowadays due to the fact that the eavesdropper's ability to capture signals becomes stronger and stronger. In order to mitigate this problem, the Sanders company developed a correlated hopping enhanced spread spectrum (CHESS) radio in 1996 [8], [9], which is capable of improving data rate and solving the problems of tracking jamming and multipath interference [10]–[14]. A core technique in CHESS is the differential frequency hopping (DFH), which significantly enhances the security [15], owing to the so-called G-function algorithm is employed for data modulation and demodulation. With the employment of the G-function, even if an eavesdropper can capture a large number of DFH signals, it is still difficult to demodulate the transmitted information, when without the knowledge about the correlation between the previous and current frequency. Hence, the design of a powerful G-function algorithm is crucial to the security performance of DFH systems.

So far, a range of methods have been considered for the constructions of the G-function. Chen et al. [16] have developed a G-function based on the linear congruence theory [17], which has a simple structure and is also easy to control, but the continuity and randomness of generated sequences are generally poor. Zhou et al. [18] have presented a G-function designed based on the fuzzy and chaotic theory, which has a large linear complexity, but its continuity is not desirable. Zhu et al. [19] have proposed a time domain and frequency domain perturbation G-function. It is able to

This work was supported in part by the Key Project of National Natural Science Foundation of China under Grant 61631015 and 61501354, in part by the National Natural Science Foundation for Distinguished Young Scholar of China under Grant 61825104, in part by National Natural Science Foundation of China under Grant 61941105, 61901327 and 61801518. (Corresponding authors: Jia Shi and Zan Li.)

R. Chen, J. Shi, Z. Li and L. Guan are with the State Key Laboratory of Integrated Service Networks, School of Telecommunications Engineering, Xidian University, Xi'an 710071, China. (E-mail: ruichen@stu.xidian.edu.cn, jiashi@xidian.edu.cn, zanli@xidian.edu.cn, lguan@xidian.edu.cn).

L-L. Yang is with the School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, U.K. (E-mail: lly@ecs.soton.ac.uk).

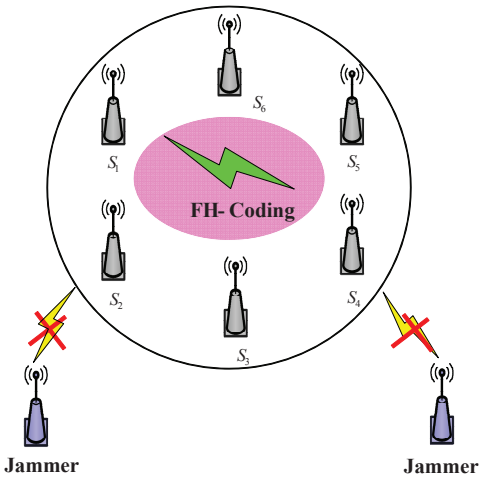

Fig. 1: A possible communication network supported by FH techniques.

improve the performance against partial-band noise jamming over additive white Gaussian channels. However, this method requires synchronization between transmitter and receiver. In [20], a G-function designed based on the cryptography theory have been proposed. The generated sequences have good uniformity and randomness. However, the security analysis of the algorithm has not been given. Furthermore, all the above mentioned algorithms for G-function are symmetric encryption algorithms. They may not be suitable for operations in the wide area networks. Moreover, as shown in Fig. 1, which depicts a FH supported communication system with multiple users as well as some jammers. The security of this system is relied on the keys of a symmetric cryptographic algorithm. It can be shown that for the number of users S_n, the number of keys needing to be managed is $n(n-1)/2$. Hence, the secret key management in this kind of system has become a problem and there is a big risk of key.

Due to the above mentioned issues and by taking the advantages of asymmetric encryption algorithm [21], in this paper, we propose a novel G-function assisted sequence generation (GF-SG) algorithm. The basic idea behind our method is to encrypt plaintext with the Government Standard (GOST) algorithm, while the encrypt keys are protected with the Rivest-Shamir-Adleman (RSA) algorithm [22]. Furthermore, in order to solve the problem of poor correlation among the sequences generated by the G-function and improve the decoding reliability at receiver, we propose a high order G-function aided sequence generation (HGF-SG) algorithm designed on the basis of the GF-SG algorithm. It can be shown that our proposed methods can not only make full use of the advantages of the different types of cryptographic algorithms, but also employ the excellent performance of the G-function.

To summary, our main contributions can be stated as follows.

1) The equivalence between cryptosystem and DFH system is investigated, from which we show the equivalence between the G-function and the corresponding cryptosystem.

2) Based on the equivalence between cryptosystem and DFH systems, we propose two sequence generation algorithms, namely the GF-SG algorithm and the HGF-SG algorithm, which are designed on the basis of the GOST algorithm and the RSA algorithm, respectively.

3) As the two most important performance metrics, we analyze the security and the ergodicity of the proposed GF-SG and HGF-SG algorithms. In our analysis, we mainly consider the security of the keys, which are protected by the RSA algorithm. Then, the exhaustive attack and factorization of RSA algorithm are considered. Furthermore, the ergodicity is analyzed on the basis of Markov model.

4) The performance of the proposed algorithm is investigated based on a range of numerical results in terms of the uniformity, randomness, complexity and security. Our studies show that the DFH sequences generated by our proposed G-function algorithms have good performance without having to assume the synchronization between both sides.

The rest of this paper is organized as follows. Section II introduces the main assumptions and system model. Section III discusses the general theory of DFH sequence design. Section IV proposes the novel GF-SG algorithm and HGF-SG algorithm. Section V analyzes the security and ergodicity performance of the proposed algorithms. Section VI presents the simulation and testing results. Finally, conclusions are summarized in Section VII.

Notation: The following notations are used. $||$ denotes the join operation, $\varphi(.)$ denotes the Euler function. Given the elements a and b in set $\mathcal{N}$, we denote that for any a belonging to $\mathcal{N}$, there is a $b \neq a$ by $\forall a \in \mathcal{N}, \exists b, a \neq b$. Given any two numbers x and y, we use $x \equiv y$ to denote that x equals y in any case.

II. System Model

In this section, we introduce the system model and principles of DFH systems, as well as the main assumptions. Let us first address the system model and main assumptions.

A. System Model and Main Assumptions

Due to the invulnerability of the ionosphere, high frequency (HF) communication techniques have drawn intensive interest in the development of reliable high-speed data communication systems [23], [24]. In this paper, we conceive a DFH based secure wireless system, which has the structure as depicted in Fig. 2. As shown in the figure, our DFH system consists of a certification center (CA) and a cluster of users, which may also experience eavesdropping. Each user may act as both transmitter and receiver. For a communication link, the two users are authenticated by the trusted center CA. The authentication process can be divided into five phases, as follows:

Phase 1: Each of the two users sends its own identity I_i to the CA.

Phase 2: CA signs the identity of the receiver with its private key, and then distributes the public key of the receiver user to the transmitter.

Phase 3: The transmitter user encrypts its identity as well as the identity of the receiver with the public key of the receiver, and sends the encrypted identity to the receiver.

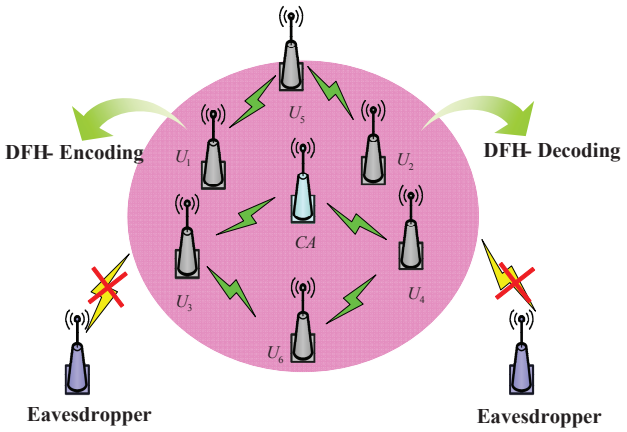

Fig. 2: DFH secure communication systems.

Phase 4: The receiver sends the identity of itself as well as that of the transmitter to the CA.

Phase 5: CA signs the identities of both the transmitter and receiver, as well as the public key of the transmitter. Then it sends all the signed information to the receiver.

From the above process, it can be seen that the transmitter obtains information about the legitimate receiver with the aid of the CA and sends identity to the receiver, who sends the identity of itself and the transmitter to the CA to verify their identities. Hence the transmitter and receiver are effectively authenticated. In order to prevent eavesdroppers from attacking the authentication process, at phase 3, a randomly generated number R_i is sent associated with the encrypted identity. Let U_i denote user i, I_i and I_j are the identities of the transmitter and receiver, respectively, PK_{U_i} and SK_{U_i} are the public and private keys of U_i, SK_{CA} is the private key of CA and the public key PK_{CA} of CA is known to all the users. Then, the above process can be described using these definitions, the encryption E and decryption D functions as follows.

$$U_i \rightarrow CA : I_i, \tag{1}$$

$$CA \rightarrow U_i : E_{SK_{CA}}(PK_{U_j}, I_j), \tag{2}$$

$$U_i \rightarrow U_j : E_{PK_{U_j}}(I_i, I_j, R_i), \tag{3}$$

$$U_j \rightarrow CA : E_{PK_{CA}}(R_i, I_i, I_j), \tag{4}$$

$$CA \rightarrow U_j : E_{PK_{U_j}}(E_{SK_{CA}}(I_i, I_j, R_i))||E_{SK_{CA}}(PK_{U_i}, I_i). \tag{5}$$

After authentication but prior to communication, the transmitter and receiver exchange the session keys and initial frequency in advance. When DFH systems are considered, let us assume that the transmitter and receiver are U_i and U_j, respectively. The process of key distribution can be divided into three steps, as follows:

Step 1: The transmitter encrypts its identity, the session keys as well as the initial frequency with the public key of CA.

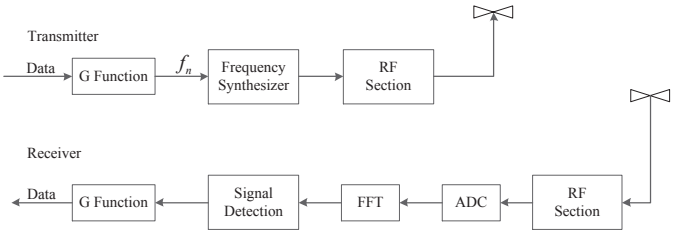

Fig. 3: Schematic diagram of DFH systems.

Step 2: CA signs the identity of the transmitter with its private key. In addition, CA encrypts the session keys and the initial frequency with the public key of the receiver.

Step 3: The receiver decrypts the information with its private key.

Meanwhile, let the initial frequency be f_0. Hence, the above process is expressed as follows.

$$C = E_{PK_{U_j}}(f_0, key), \tag{6}$$

$$(f_0, key) = D_{SK_{U_j}}(C). \tag{7}$$

From above we can be inferred that the secret information cannot be obtained by an eavesdropper even if it is able to intercept the DFH signal. This because the eavesdropper does not know the private key SK_{U_j} of U_j. During data transmission, U_i can transmit information to U_j using the session key, to ensure that secrecy communication in our DFH systems can be achieved.

B. Principle of DFH Communications

Fig. 3 shows the schematic diagram of DFH systems. Specifically, the transmitter of DFH systems has the components of a frequency transfer function referred to as the G-function, and a frequency synthesizer, at the receiver side, the DFH signal is detected based on the FFT assisted signal processing, as to be detailed in our forthcoming discourses.

With the aid of the G-function, given the information symbols $X_1, X_2, ...$, the DFH sequence is generated as

$$f_n = G(f_{n-1}, X_n) \quad n = 1, 2, ..., \tag{8}$$

where f_n is referred to as the nth frequency. At the receiver, the detector has to be designed to achieve the operation of

$$X_n = G^{-1}(f_{n-1}, f_n). \tag{9}$$

From (8) and (9), we can know that at the transmitter, the G-function uses the frequency of the previous slot and the information to be transmitted X_n at the current slot as inputs, to generate the current frequency. At the receiver, the inverse G-function is used to recover the information based on the frequencies received during the previous and current slots. Moreover, the requirements and characteristics of the G-function are summarized as [25].

From the properties of the G-function, each individual frequency can be viewed as a node or state. Correspondingly, the generation of a DFH pattern can be regarded as a state

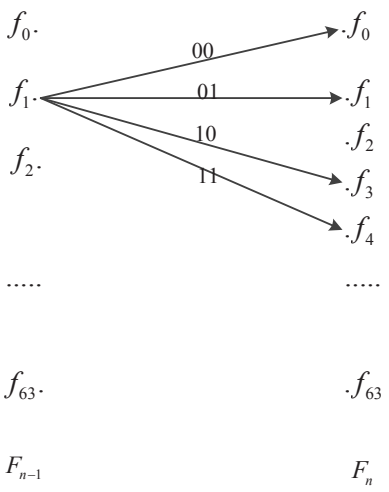

Fig. 4: An example to show the directed graph of a G-function.

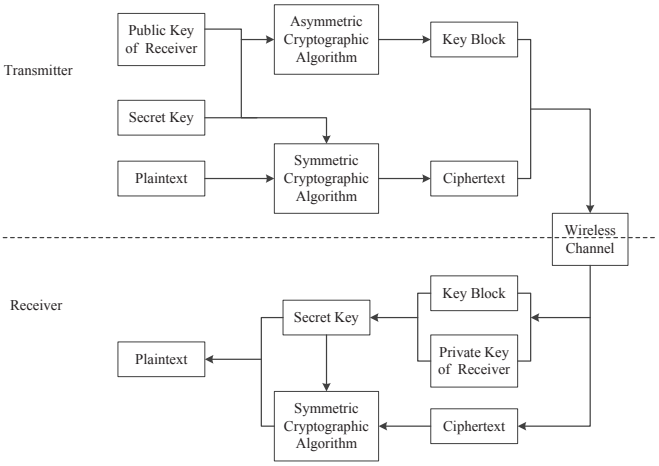

Fig. 5: Flow chart of hybrid cryptosystem.

transition process. Assume that the number of branches outputing each state is $f = 2^B$ and the number of bit transmitted per hop symbol is B. Similarly, if the DFH pattern is regularly designed to make all the frequencies uniformly activated, each state will also have the same $f = 2^B$ number of branches. As an example shown in Fig. 4, where $B = 2$, each state has four branches, activated respectively by the bit sequences of (00,01,10,11), yielding a frequency shown at the right hand side of Fig. 4, which is transmitted during the current slot.

From the above discussion, we can know that the design of the G-function and the DFH sequence is critical for the DFH system. It directly affects the performance of the DFH system. In the next section, we will consider the general theory for the design of DFH sequences.

III. General Theory for Design of DFH Sequences

In the design of the security systems for the networks supporting a large number of users, hybrid encryption is a commonly used encryption technique used for, such as, sending secure email, visiting secure sites, secure online document transmitting, etc. Therefore, we introduce the hybrid technique in the design of DFH sequences. Below, we first introduce the principles of hybrid encryption, and then analyze the equivalence between the design of the hybrid encryption and that of the DFH sequences.

A. Principle of Hybrid Encryption

It is well-known that encryption algorithms are divided into symmetric encryption algorithms and asymmetric encryption algorithms. Symmetric encryption algorithms have the advantages of low cost and fast operation, but they require relatively higher resources and cost to manage the keys. By contrast, in asymmetric encryption systems, both public keys and private keys are used, and users only need to keep their private keys. Hence, it is easy to manage and distribute the secret keys in asymmetric cryptograph systems. However, the data rate of asymmetric cryptograph systems is usually low. Due to the above-mentioned advantages and disadvantages, in practice, the combination of the symmetric and asymmetric encryption algorithms is often used to attain a good trade-off between security and efficiency.

The basic principle of hybrid encryption is that during a session, the key to be used by a symmetric encryption algorithm is first conveyed from a transmitter and a receiver with the aid of an asymmetric encryption algorithm. Then, the plaintext information is transmitted based on a symmetric encryption algorithm. Fig. 5 shows the structure of hybrid encryption and decryption implemented at the transmitter and receiver, respectively. At the transmitter side, the key used by the symmetric encryption algorithm is encrypted using the public key of the receiver. This encrypted key is transmitted along with the ciphertext, which is obtained by encrypting the plaintext using the key of the symmetric encryption algorithm. At the receiver side, after receiving the contents for recovering the symmetric key, this key is recovered using the private key of the receiver. Having obtained the key for the symmetric encryption algorithm, the plaintext can be recovered.

From the above process, we can know that the key can be transmitted over open communication channels, instead of having to use secret channels to transmit the key to the receiver. In this way, keys can be relatively easily distributed and secrecy data rate may be improved.

B. Equivalence between Design of Encryption Algorithms and that of G-Function

According to [26], there exists an equivalence between the design of cryptosystem and that of FH communication system. To meet the requirement of high security in DFH systems, we propose an approach for generating DFH sequence. As shown in our forthcoming discourse, our DFH sequence is generated in the principles of the hybrid encryption algorithm, which can be shown to have high security. We should note that, in contrast to the conventional FH sequence, DFH sequence is independent of the TOD, as mentioned previously, it is determined by the G-function. In order to demonstrate the equivalence between the design of DFH sequence and that of a cryptograph algorithm, let us first introduce a theorem.

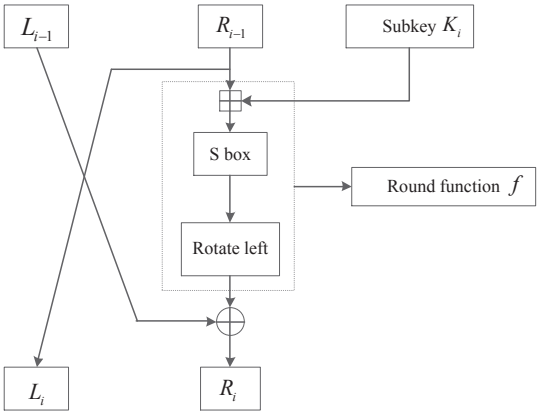

Fig. 6: Operation of one iteration in the GOST algorithm.

Theorem 1: Let g_1 be a G-function of a DFH system, which generates a DFH sequence of $F = \{f_i | i = 1, 2.., N\}$, according to the operation of $f_i = g_1(f_{i-1}, x_i)$, where f_i is the current frequency and x_i is the current information symbol to be transmitted. Let g_2 be a cryptograph algorithm, which generates a ciphertext sequence $C = \{c_i | i = 1, 2.., M\}$ by the operation of $c_i = g_2(p_i, k_i)$, where p_i is a plaintext and k_i is a subkey. Then, the design of the G-function g_1 is equivalent to the design of cryptograph algorithm g_2.

Proof 1: See Appendix A.

IV. Schemes for Generation of DFH Sequences

In this section, we propose two algorithms for DFH sequence generation, namely the GF-SG and HGF-SG schemes, which are designed based on the rationales of the symmetric cryptographic algorithm of GOST and the asymmetric cryptographic algorithm of RSA, respectively. Before introducing the proposed DFH sequence generation schemes, in the following two subsection, a brief review of the principles of the GOST and RSA algorithms is provided.

A. Overview of Principles of the GOST and RSA Algorithms

GOST algorithm is a block cipher algorithm processing 64-bit data blocks using a 256-bit key. Fig. 6 shows the operations occurred during one iteration of the GOST algorithm. The details of the iterative process and methodology are provided in [27], not discussed here. In summary, given the inputs L_{i-1}, R_{i-1} and K_i, the operations during the ith iteration can be represented as

$$L_i = R_{i-1}, \tag{10a}$$

$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i). \tag{10b}$$

In cryptograph, RSA algorithm is one of the most popular public key encryption algorithms. Built on the fact that it is extremely difficult to derive the private key from the public key, the RSA algorithm is able to resist the vast majority of attacks known today. The operational principles of the RSA algorithm is shown in Fig. 7. It consists of encryption algorithm E, decryption algorithm D and the key pair generator, which generates a public key for the transmitter and a private key for the receiver. Similarity, the specific key distribution

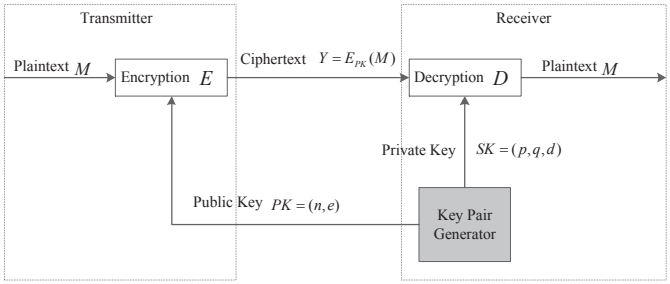

Fig. 7: Illustration of the operation of the RSA algorithm.

process of RSA algorithm is detailed in [28], not discussed here. As the result, (n, e) is the public key for the transmitter, and (p, q, d) is the private key kept by the receiver. During data transmission, the transmitter uses the public key of the receiver to encrypt plaintexts m, while the receiver uses its own private key to decrypt the received ciphertexts c. Mathematically, the encryption and decryption operations can be represented by

$$\text{Encryption} : c = m^e \text{mod}(n), \tag{11}$$

$$\text{Decryption} : c^d \text{mod}(n) = m^{ed} \text{mod}(n) = m. \tag{12}$$

B. GF-SG Algorithm

According to the equivalence design stated in Theorem 1, the GF-SG algorithm can be designed to have the flow chart as shown in Fig. 8, which shows that the GF-SG algorithm can be regarded as the encryption phase of a cryptographic algorithm. As shown in Fig. 8, the frequency f_{n-1} generated from a previous iteration acts like the plaintext and is input to the encryption unit, while the ciphertext is the resultant frequency f_n generated in the current iteration. A dotted rectangle represents the G-function. Let us explain one iteration of the GF-SG algorithm in more detail. Specifically, during the nth iteration, the frequency f_{n-1} generated by the $(n-1)$th iteration is input to the GOST algorithm. Then, a frequency f_n of the current iteration is generated by the XOR operation between the output of the GOST algorithm and the information X_n to be transmitted in the current iteration. Furthermore, the secret key to be used by the GOST algorithm is provided via the RSA algorithm. In summary, the GF-SG algorithm can be described as the following three steps.

Step 1 The initial frequency f_0 and the keys of the GOST algorithm are encrypted by the RSA algorithm with the public key of the receiver.

Step 2 Taking the frequency generated from the last iteration and the encrypted keys as the inputs to the GOST algorithm.

Step 3 Finally, a frequency of the current iteration is generated by the XOR operation between the output of the GOST algorithm and the information to be transmitted.

At the receiver, the initial frequency f_0 and the keys used by the GOST algorithm are recovered by the RSA decryption using the receiver's private key. Then, the receiver utilizes the keys to recover the information based on the frequency observed by following the operation as shown in (9).

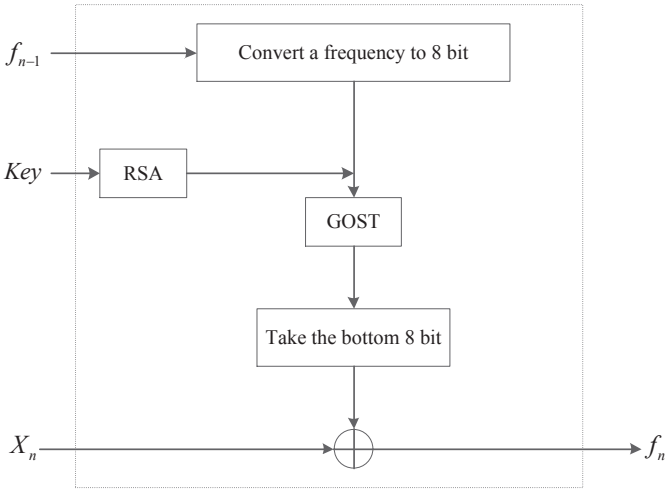

Fig. 8: G-function generated by the GF-SG algorithm.

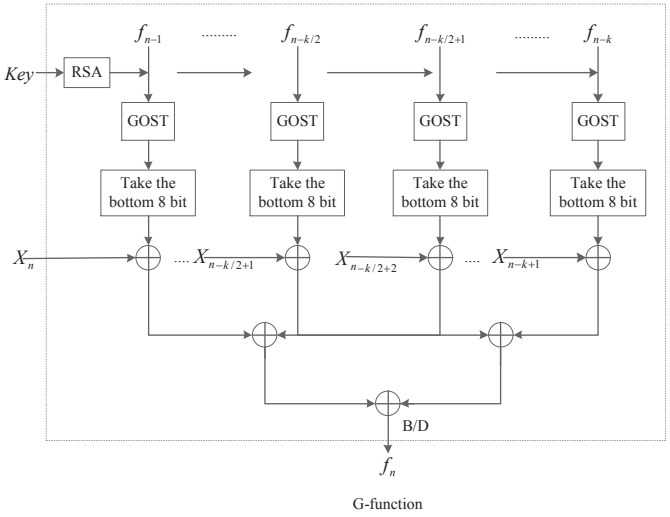

Fig. 9: G-function generated by the HGF-SG algorithm.

The GF-SG algorithm has the following characteristics. First, a small amount of information, (f_0 and secret key) is encrypted using a RSA algorithm, which ensures the security of the initial f_0 and the keys used by the GOST algorithm. This allows the receiver to achieve the synchronization with the transmitter, and the keys to be transmitted in open channel. In the following transmission, a faster encryption algorithm is used, which can improve the efficiency of data transmission. According to the principle of the GOST algorithm, the block length of plaintext is 64 bit. Hence, the DFH can generate different up to 2^{64} frequencies, which are enough for many applications in reality. For the convenience of calculation, we represent frequency using 8 bit and with 56 zeros prefixing the front. As a result, a balance between the efficiency and security of the GF-SG algorithm can be achieved.

C. HGF-SG Algorithm

In the existing literature, such as in [29], [30], the design of G-function follows the traditional method, which only establishes a mapping relationship between two adjacent frequencies. This design cannot result in a DFH sequence with high complexity, which thereby is lack of security. By contrast, our proposed HGF-SG algorithm exploits the key features of the RSA algorithm, which generates a frequency that is depended on all the past frequencies and information symbols. Hence, it is much more secure.

Considering the equivalence between the GF-SG algorithm and a cryptographic algorithm, at time n, the frequency f_{n-1} and the information symbol X_n can be regarded as the plaintext, while the new frequency f_n can be viewed as the ciphertext output. In order to enhance the complexity and security of the DFH sequence, we may let a newly generated frequency be depended on k past frequencies and k information symbols, expressed as

$$f_n = G(f_{n-1}, f_{n-2}, ..., f_{n-k}, X_n, X_{n-1}, ..., X_{n-k+1}). \tag{13}$$

$$(X_n, X_{n-1}, ..., X_{n-k+1}) = G^{-1}(f_{n-1}, f_{n-2}, ..., f_{n-k}, f_n). \tag{14}$$

In this way, a frequency in a DFH sequence, such as f_n, should be directly related to the consecutive k hops. Corresponding, The DFH sequences are classified as the high order DFH (HO-DFH) sequences.

Fig. 9 shows the operations of the HGF-SG algorithm, which can be described in three steps.

Step 1 Initially, k consecutive frequencies $(f_0, f_1, ..., f_{k-1})$, k consecutive information symbols $(X_1, ..., X_k)$ as well as keys of the GOST algorithm are encrypted using the RSA algorithm with the public key of the receiver.

Step 2 Taking the frequency generated from the last k iterations and the encrypted keys as input to the GOST algorithm.

Step 3 Let $f^{'}_{n-1}$, $f^{'}_{n-2}$, ..., $f^{'}_{n-k}$ be the generated frequencies by XOR the output of the GOST algorithm with the information symbols $X_n, X_{n-1}, ..., X_{n-k+1}$. At the same time, we assume that k is an even number, then $f^{'}_{n-1} \oplus f^{'}_{n-k/2+1}$, $f^{'}_{n-2} \oplus f^{'}_{n-k/2+2}$, ..., $f^{'}_{n-k/2} \oplus f^{'}_{n-k}$. Next, $k/2$ frequencies continue to be divided into two equal parts and repeat the above XOR, until the XOR of the last two frequencies, yielding the new frequency f_n.

At the receiver, the initial k consecutive frequencies, k consecutive information symbols as well as the keys of the GOST algorithm are recovered by the RSA decryption using the receiver's private key. Then, the receiver can utilize the keys to recover the information based on the frequency observed by following the operation as shown in (14). In comparison to the GF-SG algorithm, the HGF-SG algorithm has a higher complexity and security. Meanwhile, the efficiency of data transmission in the HGF-SG algorithm is lower than that of the GF-SG algorithm. The other processes for sending and receiving are the same as the GF-SG algorithm.

From the above, in order to ensure sufficient security, the parameter n of RSA is generally chosen to be large. Hence, the proposed algorithm generates DFH sequence slowly. Meanwhile, the ultimate goal of this paper is to design high security

DFH sequences. As a result, we sacrifice efficiency to improve security and complexity, which makes the practical application limited. In order to solve the problem of efficiency, we can choose the appropriate length of the key or short key frequently replaced method to achieve the balance between efficiency and security in practice.

V. Performance Analysis

The performance of DFH sequences can be measured by their security and ergodicity. When the number of carriers is large enough, ergodicity can make each carrier be selected, thus improving the anti-interference ability. Security determines the ability of the system to resist various analytical attacks. Hence, in this section, the performance of the proposed GF-SG and the HGF-SG algorithms is analyzed in terms of security and ergodicity.

A. Analysis of Security Performance

According to the previous sections, DFH sequences are designed in the principles of the cryptographic algorithms, and they are equivalent to each other. Hence, the security performance of the DFH sequences can be analyzed in the similar way as that of the corresponding cryptographic algorithms. It can be seen from the principle of Kerckhoffs that the security of a cryptographic algorithm is depended on the secrecy of secret key [31]. Conventionally, a dedicated secret channel is employed to transmit the secret key, which demands additional overhead bandwidth and computational burden. With the advent of the public key cryptography, this problem can be effectively solved by exchanging the secret keys with the aid of public keys. In these type of systems, the security of the concerned cryptographic algorithms mainly depends on the security of the public key algorithm. Therefore, below we analyze the security of the RSA algorithm embedded in our proposed algorithm, which directly determines the security performance of the proposed DFH sequences. As suggested in [32], the two most important metrics, namely the exhaustive search and decomposition, are analyzed in order to analyze the security performance of the DFH sequences.

a) : **Exhaustive Search**

The values of the parameters p and q used in RSA algorithms are typically in the range of $2^{511}<p<2^{512}$ and of $2^{511}<q<2^{512}$. Hence, if the exhaustive search approach is used to test for p, the average number of tries required is about

$$\frac{(2^{512}-2^{511})}{2}=2^{510}. \tag{15}$$

which is a huge number that is beyond the capability of any existing computer. As a result, it is safe enough for us to employ the RSA algorithm with a key of 1024-bit length.

b) : **Decomposition of** n

In RSA algorithm, $n=pq$. Hence, when p and q are known to an attacker, the value of

$$\varphi(n)=(p-1)(q-1), \tag{16}$$

can be readily computed. Since the pubic key e is known, the attacker can easily derive the secret key d using the relationship of

$$ed\equiv 1 \bmod(\varphi(n)). \tag{17}$$

Consequently, the RSA algorithm is broken.

From above we can conceive that deciphering a RSA is no more difficult than factorizing a large integer. However, factorizing a large integer has been recognized to be a hard mathematical problem, as there are no efficient factorization algorithms available for the time being. Instead, the exhaustive search has to be used to attack the system. Therefore, the DFH systems designed based on the RSA algorithm have strong resistance to the factorization relied attack.

B. Analysis of Ergodicity Performance

It is well known that the frequency generation process in DFH systems can be regarded as a Markov process. Below, we analyze the ergodicity of the proposed DFH systems.

1) GF-SG algorithm: From the GF-SG algorithm shown in Section IV, we can see that the frequency generated at a certain iteration is only related to the frequency generated at the previous iteration. Hence, the frequency generation process of the GF-SG algorithm can be regarded as a first order Markov process [33]. Then, the DFH sequences generated by the GF-SG algorithm has the properties summarized by Theorem 2.

Theorem 2: Let the frequency set used by the GF-SG algorithm be expressed as F=$\{f_i|0\leq i\leq N-1, N=2^m\}$, where m is an integer. Let the information symbol set be X=$\{0,1,2...,2^B-1\}$, where B is the number of bit transmitted per hop. Assume that $B\geq m$, the frequency generation process of the GF-SG algorithm is ergodic.

Proof 2: See Appendix B.

2) HGF-SG algorithm: In the conventional first order Markov chain, the future state is only related to the present state (when it is given) and has nothing to do with the other past states. However, in some scenarios, a state may be related to more than one past states. In this case, using the first order Markov model may be very inaccurate. For this sake, the concept of high order Markov process was proposed [34], by generalizing the first order Markov process. Based on the principles of the generalized Markov process, as seen in (13), the frequency transfer process in the HGF-SG algorithm can be regarded as a kth order discrete time Markov process. Furthermore, it can be shown that the frequency transfer process in the HGF-SG algorithm follows the ergodicity summarized by Theorem 3.

Theorem 3: Let the frequency set be F=$\{f_i|0\leq i\leq N-1, N=2^m, m=1,2,...\}$ and the information symbol set be X=$\{0,1,2...,2^B-1\}$, where B is the number of bit transmitted per hop. When $B\geq m$, the HGF-SG algorithm guarantees that all the frequencies are activated.

Proof 3: See Appendix C.

VI. Numerical results

The performance evaluation for the DFH sequences is carried out from the perspectives of uniformity, randomness, complexity and security, which can directly reflect the performance of the DFH system. Note that, our simulation results are obtained according to the principle of statistical hypothesis

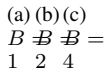

Fig. 10: Test results for uniformity.

Fig. 11: Test results for randomness.

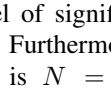

Fig. 12: LZ complexity of the sequences generated by different approaches.

[35]. Specifically, the level of significance in the simulation tests is set to $\alpha = 0.05$. Furthermore, we assume that the number of frequency set is $N = 256$ and the length of sequences is $L = 10000$. In our simulations, the initial frequency is set to $f_0 = 67$, and the number of bit per symbol is $B = 1, 2, 4$. Moreover, the size of the GOST and RSA key is 32 bit. Meanwhile, for comparison, four algorithms are considered, which are the affine transformation, the reversible hash algorithm, the GF-SG algorithm and the HGF-SG algorithm.

A. Uniformity

The standard chi-squared test is used to measure the degree of uniformity for the DFH sequences generated, where the hypotheses H_0 and H_1 represent that the sample data is uniformly distributed or not, respectively. Specifically, we use P value to determine the results of hypothesis test. If the probability that the P value is greater than α, we accept H_0, otherwise reject H_0.

Fig. 10 shows the test results statistics for the uniformity, when $B = 1$ (Fig. 10(a)), $B = 2$ (Fig. 10(b)) and $B = 4$ (Fig. 10(c)). Furthermore, in each case, four data streams are used. As shown in Fig. 10(a) and (b), when $B = 1$ or $B = 2$, the affine transformation operated on the second data stream gives $P \leq 0.05$. For the case of $B = 1$, the reversible hash algorithm attains the most stable result with $0.5 \leq P \leq 0.8$. For the case of $B = 2$, the GF-SG algorithm achieves the most stable result with $0.6 \leq P \leq 0.9$. By contrast, for the case of $B = 4$, the HGF-SG algorithm achieves the most stable result with $0.8 \leq P \leq 0.9$. From the above observations, we can draw the conclusion that both our proposed DFH sequences as well as the sequences generated by the reversible hash algorithm have uniform distribution, while the sequences derived from the affine transformation are not uniformly distributed. Furthermore, the DFH sequences generated by the HGF-SG algorithm have the best uniformity, when the number of bit transmitted per hop is relatively big, such as, $B = 4$.

B. Randomness

According to [36], the randomness of DFH sequences can be tested by the Z statistics. Specifically, let the null hypothesis H_0 and the alternative hypothesis H_1 be the binary decisions that infer whether the sample data is random or not. Then, if the P value is larger than 0.05, we accept H_0, otherwise reject H_0.

Fig. 11 shows the test statistics for the randomness, the case of $B = 1$ (Fig. 11(a)), $B = 2$ (Fig. 11(b)) and $B = 4$ (Fig. 11(c)), respectively. As shown in Fig. 11, for the case of $B = 1$, the reversible hash approach achieves the most stable result with $0.7 \leq P \leq 0.9$. For the case of $B = 2$, the HGF-SG algorithm achieves the most stable result with $0.8 \leq P \leq 1.0$. For the case of $B = 4$, the GF-SG algorithm achieves the most stable result with $0.8 \leq P \leq 0.85$. From the above observations, we can conclude that the proposed DFH sequences and the sequences generated by the affine transformation have relatively high randomness. By contrast, the sequences derived form the reversible hash algorithm are not random. Furthermore, the HGF-SG algorithm generates the DFH sequences with the best randomness, if the number of bit transmitted per hop is relatively high.

C. Complexity

Lempel-Ziv (LZ) complexity is used to measure the complexity of the DFH sequences. As defined in [37], LZ complexity can characterize the complexity of a sequence by measuring the rate that a new pattern appears in a single sequence. Compared with the other complexity metrics, LZ complexity is computable and the computation speed is very fast.

Let us denote the complexity of a given sequence $S = (s_1 s_2 ... s_n)$ by $c(n)$. According to the definition, we have

$$\lim_{n \to 0} c(n) = b(n) = \frac{n}{log_2(n)}, \tag{18}$$

where $b(n)$ is the progressive behavior of random sequences. Then, the normalized complexity of S is defined as

$$C_{LZN}(n) = \frac{c(n)}{b(n)}. \tag{19}$$

With the aid of (18), we can readily know that the complexity of a random sequence S tends to 1 and the complexity of a nonrandom sequence tends to 0, where a higher C_{LZN} means a higher complexity of a sequence.

Fig. 12 gives the LZ complexity of the sequences generated by the four approaches considered. It can be seen that the sequences obtained by the HGF-SG algorithm have the highest LZ complexity in all the three cases, especially, when $B = 1$. According to the construction of the HGF-SG algorithm, a frequency that is depended on the previous k consecutive frequencies. As a result, the bigger the k , the higher the complexity of the HGF-SG algorithm, this is consistent with the results in Fig. 12. Therefore, the HGF-SG algorithm results in the best performance in terms of complexity.

TABLE I Decryption Time (L=10000)

Sequence / B	Affine Transformation	Reversible Hash Algorithm	GF-SG algorithm	HGF-SG algorithm
$B=1$	0.896 s	8.513 s	1345.863 s	43763.224 s
$B=2$	0.910 s	7.689 s	1353.352 s	43249.624 s
$B=4$	0.925 s	7.767 s	1350.122 s	43173.880 s

TABLE II Decryption Time (L=100000)

Sequence / B	Affine Transformation	Reversible Hash Algorithm	GF-SG algorithm	HGF-SG algorithm
$B=1$	8.953 s	77.089 s	13383.647 s	435193.908 s
$B=2$	9.011 s	76.939 s	13370.029 s	427271.365 s
$B=4$	9.067 s	77.599 s	13276.193 s	424532.823 s

D. Security

The security performance of a sequence can be measured by the time required to decrypt the information symbols embedded in the sequence [38]. Note that, as transmitted information in the GOST algorithm is encrypted by the symmetric cipher algorithm, the decryption time is equal to the time required to decrypt the secret key plus the time required to decrypt the transmitted data. Let us assume that each algorithm encrypts the same length of information symbols using a key of 32 bit. Furthermore, we consider two cases with the sequence length of $L = 10000$ and $L = 100000$, respectively.

Correspondingly, Table I and Table II show the time required to decrypt the transmitted data. From the tables, we can see that the decryption time of both the GF-SG and HGF-SG algorithm are much higher than that of the other two algorithms. Furthermore, the HGF-SG algorithm requires the longest time to decrypt the transmitted data. From the security results, due to high security of the RSA algorithm, this is mainly reflected in the difficulty of decomposing the key of the RSA algorithm. Moreover, because of high complexity of the HGF-SG algorithm in this paper, hence, it takes more time to decipher the proposed algorithms than the other algorithms, which is consistent with the results in Table I and Table II. Therefore, if the size of the RSA key is relatively large, such as 1024 bit, the time required to decrypt the transmitted data becomes negligible in comparison with the time required to decrypt the secret key. As a result, the proposed algorithm is able to achieve high security performance.

VII. Conclusions

In this paper, the G-function algorithm for DFH systems has been deeply analyzed. Based on the equivalence proof between the DFH system and the cryptosystem, the advantages of symmetric and asymmetric algorithm have been fully utilized to design the G-function. Hence the GF-SG and HGF-SG algorithms have been proposed. Then the security and ergodicity performance have been deeply analyzed. At the same time, the statistical test results have shown that the proposed DFH sequences have better performance than other two algorithms in terms of the uniformity, randomness, complexity and security. As a result, it is of great significance to improve the security of wireless communications.

Appendix A
Proof of Theorem 1

Let us show the equivalence introduced in Theorem 1 from the respective of the operations at both transmitter and receiver.

a) : **Operations at Transmitter**

Let us assume a DFH system, which is at the initial state f_0. Then, when one symbol x_A or x_B is transmitted, we have the DFH sequence of

$$f_A = g_1(f_0, x_A),\ f_B = g_1(f_0, x_B). \tag{20}$$

Ideally, when $N \to \infty$, we should have

$$x_A \neq x_B \Rightarrow f_A \neq f_B, \tag{21}$$

Meanwhile, $\forall f_i \in F$, $\exists f_j$ and x_j, such that $f_i = g_1(f_j, x_j)$. Hence, when given x_i, g_1 can be seen as a one-to-one mapping.

Corresponding, in cryptosystems, let the plaintext space be $P = \{p_i | i = 0, 1, 2.., M\}$. Then, given the plaintext of $p_A \in P$ and $p_B \in P$, as well as the key k_0, we have

$$c_A = g_2(p_A, k_0),\ c_B = g_2(p_B, k_0). \tag{22}$$

In the design of cryptograph algorithms, when the key is determined, it is required that

$$c_A = c_B \Rightarrow p_A = p_B, \tag{23}$$

At the same time, $\forall c_i \in C$, $\exists p_i$ and k_i such that $c_i = g_2(p_i, k_i)$. In other words, when given k_i, g_2 can also be regarded as a one-to-one mapping.

From the above description, we can know that g_1 is a mapping from F to F, while g_2 is a mapping from P to C. According to the theory of finite set [39], if $N = M$, the sets of F, P and C are equivalent. As the result, the input parameters and output parameters of the two systems are equivalent, i.e., the design of the G-function is equivalent to that of the cryptograph algorithm.

b) : **Operations at Receiver**

First, In a DFH system, the G-function must be invertible so that the receiver can demodulate the received symbols from the DFH sequence, which can be formulated as

$$x_i = g_1^{-1}(f_{i-1}, f_i), \tag{24}$$

Hence, the operation in G-function and its inverse function should be invertible. By contrast, in the cryptosystem, the decryption operation can also be explained by an inverse operation of the encryption operation, which achieves

$$p = g_2^{-1}(c, k_1). \tag{25}$$

where k_1 is the key for decryption, which may be the same as the transmit key or a private key of the receiver, if asymmetric cryptograph system is implemented. Therefore, when considering the operations at both transmitter and receiver, we can see that the operation rules for the DFH system and that for a cryptograph system are equivalent.

In other words, the design of the G-function g_1 is equivalent to that of a cryptograph algorithm g_2.

APPENDIX B
PROOF OF THEOREM 2

Let the probability of activating a frequency of the current slot f_i be $p(f_i)$ and the information symbol being sent be X_i. According to the principles of the GF-SG algorithm, as seen in Section IV, f_i is determined jointly by f_{i-1} and X_i. Hence, when considering N adjacent slots, the joint probability distribution of the N frequency slots can be expressed with the aid of the property of Markov chain as

$$p(s_1, s_2, ..., s_N) = p(s_1) \prod_{n=2}^{N} p(s_n|s_{n-1}), \tag{26}$$

where $S = (s_1, s_2, .., s_N)$ is a pseudorandom state sequence determining the N frequency slots.

Let the one-step transition matrix of DFH system be expressed as

$$Q = \begin{bmatrix} q_{00} & q_{01} & \cdots & q_{0(N-1)} \\ q_{10} & q_{11} & \cdots & q_{1(N-1)} \\ \vdots & \vdots & \cdots & \vdots \\ q_{(N-1)0} & q_{(N-1)1} & \cdots & q_{(N-1)(N-1)} \end{bmatrix}. \tag{27}$$

where q_{ij} $(0 \le i, j \le N-1)$ denotes the transition probability from the frequency f_i to the frequency f_j. Since the sum of the probabilities from one frequency to another is 1. Hence, we have $0 \le q_{ij} \le 1$, $\sum_{j=0}^{N-1} q_{ij} = 1$ $(i = 0, 1, ..., N-1)$. Considering that the number of bit transmitted per hop is B, i.e., there are 2^B possible information symbols, 2^B different frequencies can be activated from a given frequency. Then, each row and column has 2^B non-zero elements, while the other elements are zero. According to the Chapman-Kolmogorov (CK) equation, the k-step transition matrix is given by

$$Q^k = \begin{bmatrix} q_{00} & q_{01} & \cdots & q_{0(N-1)} \\ q_{10} & q_{11} & \cdots & q_{1(N-1)} \\ \vdots & \vdots & \cdots & \vdots \\ q_{(N-1)0} & q_{(N-1)1} & \cdots & q_{(N-1)(N-1)} \end{bmatrix}^k, \tag{28}$$

According to the properties of Q, we can show that Q can be expressed as

$$Q = \begin{bmatrix} (E-A) & A \\ B & (E-B) \end{bmatrix}, \tag{29}$$

where

$$A = \begin{bmatrix} q'_{0(N/2)} & q'_{0(N/2+1)} & \cdots & q'_{0(N-1)} \\ q'_{1(N/2)} & q'_{1(N/2+1)} & \cdots & q_{1(N-1)} \\ \vdots & \vdots & \cdots & \vdots \\ q'_{(N/2-1)(N/2)} & q'_{(N/2-1)(N/2+1)} & \cdots & q'_{(N/2-1)(N-1)} \end{bmatrix}, \tag{30}$$

$$B = \begin{bmatrix} q'_{(N/2)0} & q'_{(N/2)1} & \cdots & q'_{(N/2)(N/2-1)} \\ q'_{(N/2+1)0} & q'_{(N/2+1)1} & \cdots & q'_{(N/2+1)(N/2-1)} \\ \vdots & \vdots & \cdots & \vdots \\ q'_{(N-1)0} & q'_{(N-1)1} & \cdots & q'_{(N-1)(N/2-1)} \end{bmatrix}, \tag{31}$$

and E represents an $(N/2) \times (N/2)$ identity matrix. Furthermore, it can be shown that when the number of bit transmitted per hop is $B \ge m$, i.e., the number of information bit per symbol is not less than that of the available frequencies, all the frequencies can be activated. In this case, Q is full rank, provide that the probabilities of activating different frequencies are different. When Q is full rank, A and B are invertible matrices, and Q can be transformed into a diagonal matrix, which can be expressed as $Q = HDH^{-1}$, O represents a $(N/2) \times (N/2)$ zero matrix, where

$$H = \begin{bmatrix} E & (-A) \\ E & B \end{bmatrix},\ D = \begin{bmatrix} E & O \\ O & (E-A-B) \end{bmatrix}, \tag{32}$$

Then, we have

$$\begin{aligned} Q^n &= HD^nH^{-1} \\ &= \begin{bmatrix} E & (-A) \\ E & B \end{bmatrix} \begin{bmatrix} E & O \\ O & E-A-B \end{bmatrix}^n \begin{bmatrix} \frac{B}{(A+B)} & \frac{A}{(A+B)} \\ \frac{(-E)}{(A+B)} & \frac{E}{(A+B)} \end{bmatrix} \\ &= \begin{bmatrix} \frac{(B+A(E-A-B)^n)}{(A+B)} & \frac{(A-A(E-A-B)^n)}{(A+B)} \\ \frac{(B-B(E-A-B)^n)}{(A+B)} & \frac{(A+B(E-A-B)^n)}{(A+B)} \end{bmatrix}. \end{aligned} \tag{33}$$

It can be shown that when $n \to \infty$, $(E-A-B)^n \to O_{(N/2)\times(N/2)}$. Applying this result to (38), we obtain

$$\lim_{n\to\infty} Q^n = \begin{bmatrix} B(A+B)^{-1} & A(A+B)^{-1} \\ B(A+B)^{-1} & A(A+B)^{-1} \end{bmatrix} = \Pi. \tag{34}$$

To this point, we can assign suitable values to the matrix A and B, then the steady-state probability Π of the DFH sequence is obtained. Then, accounting to the properties of the steady-state Markov process, we have

$$Q \times \Pi = \Pi, \tag{35}$$

where $\Pi = (\pi_0, ..., \pi_{N-1})$ is the vector. Besides $\sum_{i=0}^{N-1} \pi_i = 1$. Since Q is full rank, the probabilities of activating the frequencies from a given one are different. This guarantees that all the steady-state probabilities non-zero. Hence, $\pi_i \ne 0$.

Consequently, the GF-SG algorithm is capable of activating each frequency.

APPENDIX C
PROOF OF THEOREM 3

As can be seen from the definition of the kth order Markov process model, when N adjacent slots are considered, the joint probability distribution of the N frequencies can be expressed as

$$\begin{aligned} p(s_1, s_2, ..., s_N) =& p(s_1)p(s_2|s_1)...p(s_k|s_{k-1}, s_{k-2}, ..., s_1) \\ &\times \prod_{n=k+1}^{N} p(s_n|s_{n-1}, ..., s_{n-k}), \end{aligned} \tag{36}$$

where $S = (s_1, s_2, .., s_N)$ is a pseudorandom sequence determining the frequencies of the N slots.

According to the limit theorem [40], we can know that there is a steady-state probability vector $\pi = (\pi_0, ..., \pi_{N-1})$, which satisfies

$$\sum_{i=0}^{N-1} \pi_i = 1,\ P\pi = \pi. \tag{37}$$

Therefore, when the number of bit transmitted per hop is $B \geq m$, P is full rank provide that the probabilities of activating the frequencies from a given one are different. This guarantees that all the steady-state probabilities non-zero. Then, we have

$$\lim_{t\to\infty} p[S_t = m_j | S_k = m_n, ..., S_1 = m_s] = \pi_{m_j}. \quad (38)$$

It can also be obtained from the ergodicity analysis of the GF-SG algorithm that each frequency of the HGF-SG algorithm is also utilized.

REFERENCES

[1] J. Zhang, E. Bj?rnson, M. Matthaiou, D. W. K. Ng, H.Yang, and D. J. Love, "Prospective multiple antenna technologies for beyond 5g," *IEEE J. Sel. Areas Commun.*, pp. 1–1, 2020.

[2] Z. Wei, L. Yang, D. W. K. Ng, J. Yuan, and L. Hanzo, "On the performance gain of noma over oma in uplink communication systems," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 536–568, 2020.

[3] R. Pickholtz, D. Schilling, and L. Milstein, "Theory of spread-spectrum communications - a tutorial," *IEEE Trans. Commun.*, vol. 30, no. 5, pp. 855–884, May 1982.

[4] M. C. Davey and D. MacKay, "Low-density parity check codes over gf(q)," *IEEE Commun. Lett.*, vol. 2, no. 6, pp. 165–167, June 1998.

[5] V. Annovazzi-Lodi, M. Benedetti, S. Merlo, T. Perez, P. Colet, and C. R. Mirasso, "Message encryption by phase modulation of a chaotic optical carrier," *IEEE Photonics Technol. Lett.*, vol. 19, no. 2, pp. 76–78, Jan 2007.

[6] M. Hannon, S. Feng, H. Kwon, and K. Pham, "Jamming statistics-dependent frequency hopping," in *MILCOM 2016 - 2016 IEEE Military Communications Conference*, Nov 2016, pp. 138–143.

[7] Z. Li, Y. Chang, L. Jin, and J. Cai, "Analysis of fhma performance on block cipher based frequency-hopping sequences," *IEEE Commun.Lett.*, vol. 8, no. 7, pp. 434–436, July 2004.

[8] D. L. Herrick and P. K. Lee, "Chess a new reliable high speed hf radio," in *Military Communications Conference, 1996. MILCOM '96, Conference Proceedings, IEEE*, vol. 3, Oct 1996, pp. 684–690 vol.3.

[9] D. L. Herrick, P. K. Lee, and L. L. Ledlow, "Correlated frequency hopping-an improved approach to hf spread spectrum communications," in *Proceedings of the 1996 Tactical Communications Conference. Ensuring Joint Force Superiority in the Information Age*, Apr 1996, pp. 319–324.

[10] S. Xie and B. Qian, "Performance analysis of differential frequency hopping communication system over rician channel," in *2018 IEEE 4th Information Technology and Mechatronics Engineering Conference (ITOEC)*, 2018, pp. 1015–1019.

[11] Q. Yingchao and Y. Feng, "Design and implementation of differential frequency hopping communication system based on fpga," in *2018 IEEE 4th Information Technology and Mechatronics Engineering Conference (ITOEC)*, 2018, pp. 1006–1010.

[12] W. Zhu, B. Yi, L. Gan, and Q. Xie, "Anti-jamming performance of fountain coded differential frequency hopping systems in awgn," *China Communications*, vol. 11, no. 14, pp. 53–60, Supplement 2014.

[13] ——, "The application of fountain code in differential frequency hopping systems," in *International Conference on Cyberspace Technology (CCT 2014)*, 2014, pp. 1–4.

[14] Z. Chen, Y. Song, and B. Dong, "Performance of a compressed spectrum differential frequency hopping system over rayleigh fading channels," in *MILCOM 2013 - 2013 IEEE Military Communications Conference*, Nov 2013, pp. 781–785.

[15] S. E. El-Khamy, S. H. ElFar, and M. R. M.Rizk, "A novel secure signaling technique for underwater communication channels based on differential frequency hopping and spinal codes," in *2017 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2017, pp. 599–604.

[16] C. Yong and Z. Hangsheng, "Study of dhf g function algorithm used for dhf systems," *Journal on Communications.*, vol. 27, no. 10, pp. 100–105, Oct 2006, (in Chinese).

[17] Y. Yuliang, H. Zunwen, and K. Jingming, "Research on the transition function of differential frequency hopping," *Journal on Communications.*, vol. 23, no. 4, pp. 103–108, April 2002, (in Chinese).

[18] Z. Zhou, S. Li, and Y. Cheng, "Designing frequency transition function of differential frequency hopping system," in *2010 International Conference on Communications and Mobile Computing*, vol. 2, April 2010, pp. 296–300.

[19] Z. Wenjie, Y. Benshun, and G. Liangcai, "Performance research of fountain-dfh concatenated coding systems over awgn with partial-band noise jamming," *Systems Engineering and Electronics*, vol. 38, no. 3, pp. 665–671, Mar 2016, (in Chinese).

[20] B. Zhiqiang, W. Bo, and G. Fan, "A new differential frequency hopping scheme based on encryption algorithm," *STUDY ON OPTICAL COMMUNICATIONS*, no. 4, pp. 74–78, Aug 2017, (in Chinese).

[21] W.Diffie and M.Hellman, "New directions in cryptography," *IEEE Trans. Inf. Theory*, vol. 22, no. 6, pp. 644–654, Nov 1976.

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

[23] L. Zhang, T. Xiao, J. Hao, and X. Xiang, "Regression forest for interference assessment in real ultra short-wave communication jamming system," in *2016 12th World Congress on Intelligent Control and Automation (WCICA)*, June 2016, pp. 1459–1462.

[24] Z. Cheng, S. Wang, X. Qu, S. Yan, F. Hu, and A. Li, "Cogdfh- a cognitive-based differential frequency hopping network," in *MILCOM 2009 - 2009 IEEE Military Communications Conference*, Oct 2009, pp. 1–7.

[25] Y. Fuqiang, "Analysis of some problems about the system and techniques in hf differential frequency hopping," *RESEARCH and DEVELOPMENT.*, vol. 6, no. 5, pp. 114–118, 2004.

[26] L. Guan, Z. Li, J. Si, and R. Gao, "Generation and characteristics analysis of cognitive-based high-performance wide-gap fh sequences," *IEEE Trans.Veh.Technol.*, vol. 64, no. 11, pp. 5056–5069, Nov 2015.

[27] B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*. John Wiley New York, 1996.

[28] T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*. Massachusetts Institute of Technology, 2001.

[29] G. Liangcai and W. Shuangyuan, "A kind of shortwave frequency hopping code based on dfh transform function," *Journal of Electronics and Information Technology.*, vol. 27, no. 2, pp. 218–220, Feb 2005, (in Chinese).

[30] Y. Li, F. Yao, B. Xu, and L. Zhao, "Equalization algorithm in frequency domain for broadband dfh systems," in *2015 4th International Conference on Computer Science and Network Technology (ICCSNT)*, vol. 01, Dec 2015, pp. 1154–1159.

[31] A. Kerckhoffs, "La cryptographie militaire," *Journal des sciences militaires*, vol. IX, pp. 5–83, Jan 1883.

[32] T. Kleinjung, K. Aoki, and J. Franke, "Factorization of a 768-bit rsa modulus," in *2010 Cryptology Conference(CRYPTO)*, 2010, pp. 333–350.

[33] G. Sponsler, "First-order markov process representation of binary radar data sequences," *IEEE Trans. Inf. Theory*, vol. 3, no. 1, pp. 56–64, March 1957.

[34] M.S.Bartlett., "The frequency goodness of fit test for probability chains," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 47, no. 1, pp. 86–95, 1951.

[35] E. L.Lehmann and J. P.Romano, *Testing Statistical Hypotheses*. Springer-Verlag New York, 2005.

[36] C.Georgescu, E.Simion, A.Nita, and A.Toma, "A view on nist randomness tests (in)dependence," in *2017 9th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, June 2017, pp. 1–4.

[37] A. Lempel and J. Ziv, "On the complexity of finite sequences," *IEEE Trans.Inf.Theory*, vol. 22, no. 1, pp. 75–81, Jan 1976.

[38] G. Hu, "Study of file encryption and decryption system using security key," in *2010 2nd International Conference on Computer Engineering and Technology*, vol. 7, April 2010.

[39] H. Simmons, *An Introduction to Category Theory*. Cambridge University, 2011.

[40] A. E. Raftery, "A model for high-order markov chains," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 47, no. 3, pp. 528–539, 1985.