Intelligent Automation & Soft Computing
DOI:10.32604/iasc.2022.025076
Article

Tech Science Press

Stream Cipher Based on Game Theory and DNA Coding

Khaled Suwais[*]

Faculty of Computer Studies, Arab Open University, Riyadh, Saudi Arabia
*Corresponding Author: Khaled Suwais. Email: Khaled.suwais@arabou.edu.sa

Received: 10 November 2021; Accepted: 28 December 2021

Abstract: Securing communication over public communication channels is one of the challenging issues in the field of cryptography and information security. A stream cipher is presented as an approach for securing exchanged data between different parties through encryption. The core of stream cipher relies on its keystream generator, that is responsible for generating random and secure keystream of encrypting streaming data. Thus, the security of the keystream is measured by its randomness and its resistance to statistical and cryptanalytic attacks. As there is always a trade-off between the security and performance while designing new cryptographic primitives, we introduce a game theory-based stream cipher that performs fast and utilises the unpredictable behaviour of prisoners' dilemma players to generate random sequences that serve as keystreams. To assure further security, the ciphertext is coded by DNA bases that are randomly scrambled at the beginning of every iteration of generating new keystream. The proposed stream cipher's performance and security are thoroughly examined in terms of randomness, performance, and resistance to cryptanalytic attacks. The experiments show that our stream cipher can encrypt around 1230 Mbit/s. Statistically, our proposed stream cipher passed the NIST statistical tests successfully.

Keywords: DNA cryptography; game theory; prisoners' dilemma; stream cipher; information security

1 Introduction

Cryptography plays a significant role in many theoretical and practical fields, including: information security, computer networks, cybersecurity, etc. Cryptography aims at providing both sender and receiver a reliable and considerable level of protection to communicate over insecure public communication channels. Cryptographic primitives are classified into symmetric and asymmetric encryption algorithms. Both categories aim to transform the readable input data (plain-text) into non-readable form (ciphertext), they differ in the techniques used to carry out the encryption and decryption processes. In symmetric-key encryption, one key is shared between sender and receiver and used for encryption and decryption. While, in asymmetric key encryption, two different keys (private, public) are used for encryption and decryption. The private key is kept secret, while the public key is available to be used by other users.

Conventional stream ciphers are designed based on various structures such as linear feedback shift registers [1], nonlinear feedback shift registers [2], T-functions [3], linear finite state machines [4], and chaos theory [5]. However, modern cryptographic primitives are developed with the intersection of other disciplines of mathematics [6], computer science, physics [7], biology [8] and business theories [9], etc. The aim of having such an intersection is to secure encryption algorithms against cryptanalysis attacks. Hence, various encryption algorithms are developed on the basis of intersection with other disciplines to harden the tasks of cryptanalysts in breaking or even detecting some behavioural patterns of cipher algorithms.

Game theory is a field of study concerned with studying the interactions between mutually distrusting parties. This theory is well-known in business, where different firms aim to increase their market's share against their competitors. Katz [9] assesses the use of game theory in the context of encryption algorithms in order to bridge the gap between game theory and cryptography. Accordingly, he found that game theory allows a higher level of sophistication and efficiency in cryptography as it resolves the issues and security flaws present in conventional cryptographic techniques. Recently, several researchers have shown interest in developing security models and protocols that combine the approaches of cryptography and game theory, as found in [10–14]. These approaches show promising results in terms of performance and security. However, none of the existing researches presented stream cipher based on game theory.

The other field of interest is the intersection between biology and cryptography, which results in Deoxyribonucleic Acid (DNA) cryptography. In this regard, DNA bases are used for encoding purposes that satisfy the security requirements and are compatible with DNA computations. Various schemes are proposed based on DNA cryptography to improve data security and exhibit security issues. However, some schemes are found time-consuming as they take excessive time for keystream generation. Hence, the time taken for encryption and decryption is also not satisfactory [15].

For this purpose, our research presents an alternative approach for developing stream ciphers based on game theory and DNA coding to resolve the performance and security limitations of some of the existing ciphers. Our objective is to develop a new stream cipher algorithm with a non-predictable keystream generator, which can generate random and secure keystreams for encryption purposes. The main contributions of this research is firstly to propose a novel stream cipher is proposed based on Iterated Prisoners' Dilemma game to generate random and secure keystream. Secondly, utilize DNA coding to help our algorithm support computing environments of low power consumption and provide higher performance.

The sections to follow in the paper are the preliminaries in Section 2, followed by a discussion on the related works in Section 3. The proposed scheme is given in Section 4 followed by the security analysis in Section 5. The complexity analysis and performance analysis are presented in Sections 6 and 7, respectively. The performance analysis of the proposed scheme is discussed in Section 7. Finally, concluding remarks are highlighted in Section 8.

2 Preliminaries

2.1 Stream Ciphers

In stream ciphers, both the sender and receiver have previously set up secret information in which they use this information for encryption and decryption. A stream cipher consists of a pseudorandom generator that works as a keystream generator (KSG). Two attributes seed this generator: a secret key and an initial vector (IV) to produce a sequence of keys known as keystream. The keystream is then used to encrypt/decrypt (E/D) a plain-text with an XOR operation. The general structure of the stream cipher is demonstrated in Fig. 1:

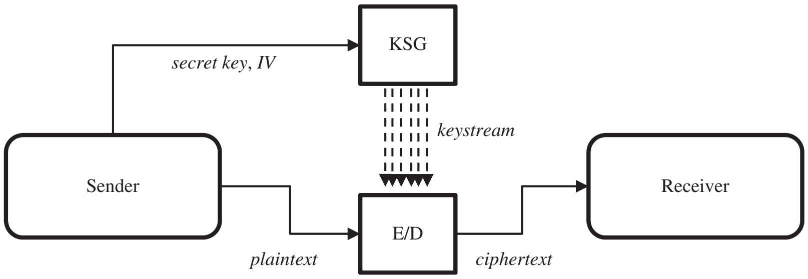

Figure 1: Stream cipher structure

The core of any stream cipher depends on the security of its keystream generator. Cryptanalysts target this generator to find any statistical or behavioural patterns that lead to any potential cryptographic weaknesses that attackers can exploit. In our proposed scheme, game theory is utilised in KSG to generate random keystream, while the DNA coding is applied in the E/D processes. The complete implementation is demonstrated in Section 4.

2.2 Iterated Prisoner's Dilemma in Game Theory

Prisoner's Dilemma (PD) is a symmetric matrix game with a transparent payoff matrix, where both players can simultaneously act without knowledge of the other's moves. PD is a strategic game between two players, and it models the story of two prisoners held suspect of a serious crime. If prisoner 1 testifies to the other, prisoner 1 will go free, while prisoner 2 will serve a long prison sentence. If both testify, their punishment will be relatively less. However, if they both cooperate with each other by not testifying, they will only be imprisoned for a short term. What makes up the dilemma is that the defect strategy dominates the cooperate strategy. At the same time, rational players will not choose to play the dominated strategy as both players will lose more if they continue adopting this strategy [16]. However, there is a risk for a prisoner cooperating while the other testifies, or if the player testifies and the other does not.

Tab. 1 shows the payoff matrix of the PD game for two players (P1, P2). Each player can choose one of the two strategies: cooperate or defect. The payoff rewarded for each player depends on the players' strategy and its opponent's strategy. The payoff R, S, T, P should satisfy S < P < R < T before starting any game. However, in the 2-players Iterative Prisoners' Dilemma (2IPD), the game is repeated *n* times. The winning player is the player who can achieve a higher payoff after completing the *n* games.

Table 1: General prisoner's dilemma payoff matrix

		P2	
		Cooperate	*Defect*

P1	*Cooperate*	P1 = R
		P2 = R
			P1 = S
			P2 = T

	Defect	P1 = T
		P2 = S
			P1 = P
			P2 = P

2.3 DNA Coding

DNA is composed of molecules known as nucleotides. The nucleotide may have one single base from a set of four kinds of bases, as illustrated in Fig. 2. These bases are: Adenine (A), Thymine (T), Cytosine (C), and Guanine (G). In DNA cryptography, plain-text is encrypted, and the ciphertext is converted into DNA bases. During the encryption process, the sender performs a mapping between the ciphertext binary bits with the combinations of DNA bases for encrypting the data, as this helps to enhance the level of security. Each combination of DNA bases results in a DNA sequence. Accordingly, the resulting ciphertext is a set of DNA sequences.

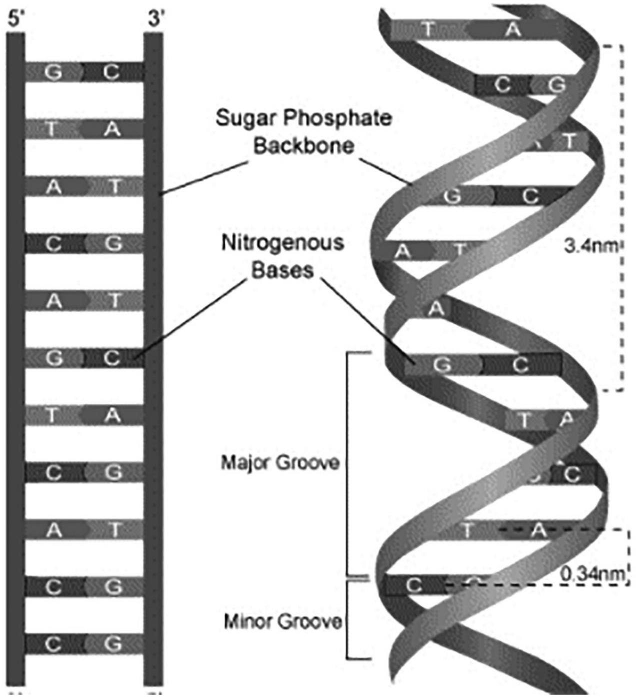

Figure 2: DNA structure [17]

Example 1. Assume that the following encoding scheme ($\mathcal{M}$) is used for mapping each two adjacent ciphertext binary bits into DNA sequence:

$$\mathcal{M} : 00 \rightarrow \mathrm{C}, 01 \rightarrow \mathrm{A}, 10 \rightarrow \mathrm{G}\,\text{and}\,11 \rightarrow \mathrm{T} \tag{1}$$

Hence, the encrypted letter ciphertext “01001011” is converted to the DNA sequence “ACGT”.

3 Related Works

3.1 Application of Game Theory in Cryptography

The concept of the game theory has extensively strengthened the scope and capabilities of the cryptography paradigm. Katz [9] assesses the use of game theory in the context of cryptography in order to bridge the gap with game theory. According to the researcher, the game theory allows a higher level of sophistication and efficiency in cryptography as it resolves the issues and security flaws present in the previously used conventional cryptography techniques. For this purpose, Katz argues that almost all of the traditional cryptographic models work under the assumption that all parties interacting in the process of data transmission are honest parties who will follow the protocol of data transmission faithfully and reliably. This is no longer an assumption that can be made in the present state of internet communication. Hence, game theory enables protection against such attacks because it changes the assumption of all parties being honest to assume that all parties involved in data transmission have their own self-interests.

Katz also states that game-theory-based cryptography is more secure than conventional techniques, which would make it the standard in the near future.

One of the core issues that come from the nature of the encryption is the disruption that it can cause in ensuring optimal compression of data getting transmitted by users. As bandwidth issues often persist in internet connections in many scenarios, compression quality is an important objective without compromising the security aspect. A method for image compression within the context of encrypted files using game theory is proposed in [14]. According to the researchers, there is an abundance in the prominence of digital imaging devices for consumers in the era of smartphones and the sharing of images using the internet, making it necessary to develop better compressions tactics that do not lower the privacy provided by encryption methods. After carrying out multiple possible approaches to the solution, Liu et al. agreed on the notion that a block-based compression technique for image file compression is potentially very effective in the domain of encrypted network traffic. This is achieved using formalisation of the compression issue in the form of a theoretical game condition and the utility function that was primarily defined on the basis of both the parameters set for the file size and block characteristics. The results are found satisfying compared to other compressions methods.

As internet technology becomes more widely adopted and used across all industries, its lucrativeness for a potential attack also increases rapidly. Intrusions and malicious attacks are at an all-time peak that takes advantage of the networking flaws and weaknesses. Wang et al. [18] explored the potential use of game theory in creating active defence mechanisms that could allow for effective intrusion detection. The focus of their study is exclusively on the cyber-physical embedded systems (CPES), which are often deployed in situations that require the highest level of security. Their study recognised that the most critical aspect of the security vulnerability in CPES comes from the security parameters of embedded sensor networks (ESNs). The challenge in implementing proper security in this area is driven by the fact that ESNs have very limited power, processing power, and storage capabilities. Hence, the researchers focused on identifying successful attacks early through an optimal intrusion detection technique. The research proposed an attack-defence game model that relies on a repeated game technique to continuously identify and deal with the potentially new attack scenarios for accurate detection. Their proposed model was able to work more efficiently while lowering the energy consumption by half compared to the existing monitoring systems used for ESNs.

In [13], the researchers made an assessment of a cryptographic approach to securing crypto-cloud computing from the use of game theory. Their research focused on not just the security level of the algorithm, but also on the encryption method being economical to deploy. The aim of that study is to present an efficient encryption system built based on XTR through game theory deployment to develop a sense of semantic security. According to their findings, developing and constructing encryption tools with the game theory concepts allow the system to operate in a manner that is the tools following a natural optimisation process to create encryption. Based on their findings, it would be possible for the various firms offering financial services to meet the various technical challenges faced by cloud computing adoption. In this manner, the use of game theory in cloud computing for crypto-cloud enables an optimum cloud strategy for scalable and comprehensive deployment.

The use of game theory is extensively popular in promoting unconventional solutions to encryption and other computing problems. One of the main factors of contributions of the game theory comes in its excellence in allowing true randomness in a system. The researchers in [19] argue that in the present-day world that is extremely connected and open, true randomness is at risk despite being extensively important in computing paradigms. According to the authors, if there is no randomness to achieve, the world may soon turn into an era where privacy on the internet will be a thing of the past. In their article, Henno et al. introduce a definition of randomness related to computerised systems and in devices that have a limited amount of memory, which is known as *k*-randomness. According to the study, there are ample necessary applications of *k*-randomness ranging from video games to randomised trials. The research argues that even though the *k*-randomness in games creates new ways of achieving randomness, it essentially works on par with other methods of computer-generated randomness. However, there is even further potential of randomness using game theory and observation of human actions. For instance, the erratic movement of the mouse pointer or player moves in a video game can be used to carry out new randomness in computing. An encryption method devised based on this approach of human interaction would not even require a public key or use the concept of a shared secret key.

The work presented by [11] aims to identify various applications that crypto cloud computing has in the context of social networks. For this purpose, the researchers used cooperative game theory to deal with the uncertainty aspect of the cloud network. To evaluate the application of crypto cloud computing, the researchers developed a cooperative interval game model, which was then applied to social networks. The data for the application was gathered from Amazon Web Services databases. The motivation of the research comes from the belief that it is possible to design crypto cloud computing in a manner that the system meets the needs of most users in the cloud infrastructure if game theory is adopted. There is a current difficulty in ensuring the same level of access and scalability in the existing cloud services offered by many prominent cloud services vendors as Google App Engine, Eucalyptus, Amazon S3, and Microsoft Azure. The focus on social media websites and cloud networks in a collection was done due to the high need for security on such systems, which is not being met. The main contribution of their research is to develop an encryption algorithm based on elliptic curves to protect cloud-based systems. The algorithm achieved a high level of security but with lower performance than other well-known ciphers (e.g., RC4, AES-128, etc.).

In the industry of healthcare as well, the need for digital imagery and the distribution of digital imagery through internet mediums has become rapidly more common for convenience and speed of consultation. However, due to the nature of the profession, the privacy of patients in relation to their personal medical images is highly important. The authors in [12] assessed the case of transmitting medical images using the internet and identified that due to the need for physicians to look after even a slight anomaly in the images, the only feasible method of medical image transmission needs to be lossless in order to retain as much data and image fidelity as possible. To overcome the issue of privacy, Zhou et al. proposed a novel lossless medical image encryption scheme to facilitate an easier and effective encryption message that explicitly fulfils the need for lossless image encryption in the healthcare industry. For the purpose of encryption, the researchers rely on the use of game theory, which is further optimised with a hidden region of interest (ROI) position and other regions of interest parameters. In order to generate random sequences to scramble the file block sequences in the image, they made use of the Quantum Cell Neural Network (QCNN). The position of ROI is set to hidden in order to avoid leakage of the information, which could have made decryption feasible.

One of the developing new challenges in the field of cryptography is the operations of IoT, where the sheer number of devices is too high, and all of these devices seemingly remain connected to all the other IoT devices. Hence, encryption and privacy become difficult with a multi-party data transfer setup [20]. This paradigm is studied extensively in [21] and made use of game theory in order to create a 3-way repeated game model that could ensure the privacy of data transmitted in an IoT infrastructure in the specific context of Mobile Edge Crowd Sensing (MECS). The key characteristic of the MECS is the very low latency in request-response delay, which ensures that different devices are able to interact and communicate quickly. However, as the technology of IoT gets expanded very rapidly without proper standards of privacy or communication prerequisites, there are many dishonest nodes in the system that are deployed to collect private information of the users. To overcome this issue, Zhao et al. proposed a three-party repeated game model that is able to intelligently sense data and observe the rationality of the data transmission patterns from the normal protocols and detect deviations. Through simulation of the same system, these researchers established that the proposed system model is feasible and beneficial in protecting the privacy of data communication.

In the age of cloud computing, one of the core ideas is the need to reduce the redundancy of the files if all the files' characteristics are the same. While the presence of redundancy on a local storage can have some benefits, it is not true in the context of cloud computing, where the entire cloud architecture uses different means for a thorough redundancy. Liang et al. [10] studied the case of data deduplication in the cloud systems for encrypted data. Their research study investigated the adoption of hybrid encrypted cloud data deduplication (H-DEDU) by means of using the game theory paradigm. The results of their study showed that the solution proposed was feasible and was able to achieve the objective of Nash equilibrium in the game theory of the Stackelberg game.

3.2 DNA Cryptography

Li et al. [22] proposed an alternative approach for image encryption based on the randomness of DNA coding for better diffusion along with a chaotic map system. For an improved chaos mapping process, the researchers presented a new type of chaotic spatiotemporal system. This chaotic spatiotemporal system is constructed using two different approaches integrated into one, coupled map lattice (CML) and Tent-Sine system (TSS). For the purpose of getting the initial parameters, the TSS, logistic map, and Lorenz map are utilised. The research experiments show that the image encryption algorithm offers high resistance against some of the most common types of attacks.

Sohal et al. [23] proposed a symmetric key cryptography scheme with a focus on securing a cloud computing paradigm with a high level of accuracy and reliability. The research argues that there are many encryption algorithms that are designed to keep the data secure in the cloud setting; however, due to the relatively new architecture of cloud computing, many of these encryption technologies fall short of optimum results in varying case scenarios. To address the issue of encryption in a cloud setting, the authors propose an encryption scheme that relies on the method of client-side encryption of the data so that it is encrypted and secured even before it is transmitted to the cloud server using the internet channels. The symmetric-key encryption scheme is based on DNA coding cryptography. The researchers comparatively assessed the algorithm against many of the existing standardised encryption techniques like DES, AES, and DNA. The limited experiment results showed that the proposed encryption algorithm yielded more promising outcomes than the conventionally used algorithms in terms of encryption time and overall throughput.

As computer-generated software tools and methods are poor in designing true randomness, there is a new trend of relying on naturally occurring processes and bio-inspired activities to create true randomness. Basu et al. [24] argue that bio-inspired cryptography systems are the fundamentals for modern-day cryptography in which the tools of machine learning and bio-inspired algorithms are effectively used to secure the data in the transmission phase. Their research study proposed a new cryptosystem based on Central Dogma for Molecular Biology (CDMB). These methods are used in order to accurately and reliably simulate the conditions of a normal, natural genetic coding, transcription, and translation processes. The input for the system is considered in the form of 16-bits dataset, and the resulting outcome is ciphertext presented as a protein base form. For the purpose of key generation, the concept of Bi-Directional Memory Neural Network (BAMNN) is adopted. The experiment results show that their scheme provides efficient encryption times in both small and large input files.

Qiu-yu et al. [25] put the focus of their research study on the concept of proposing a more comprehensive and effective encryption technique. According to their research, the existing encryption technologies and algorithms prominently used in image file encryption are not suitable for widespread use due to security concerns. The researchers especially point towards different types of statistical analysis such as noise attacks, cropping attacks, exhaustive attacks, and differential attacks. As a solution to the problem of encryption in the existing scenarios, they propose an algorithm for image encryption that relies on three core components, DNA coding, enhanced chaotic mapping, and image hashing. For the improvement of the chaotic sequence, Chen's chaotic sequence is used for the initial parameter of the chaotic map. According to their results, their algorithm shows a higher level of security, increased key space, and improved key sensitivity.

Audio files dominate the transmission on the internet servers due to the very high popularity of the media type with an association in video formats. While the size of the audio files can be very low compared to other file formats, security of the same is often an issue as audio files can have different parameters that can lead to the disclosure of unwanted information. To overcome this issue, Wang et al. [8] argued that there is a high need for different encryption technologies to focus on the audio transmission on the internet in order to raise the security level. They proposed a scheme for audio file encryption that theoretically and in limited test runs, provided much higher security for audio files than many conventional means. The strength of their scheme is derived from its use of a chaotic system alongside DNA coding to continuously perform the task of confusing and diffusing the audio file. The researchers claimed their success in achieving higher security when using a hash value of the audio file as the initial value of the chaos system.

In [26], the research recognises that in the age of the internet, the massive number of images getting shared online are all coloured, and the colour space for the images has also been widened extensively as the digital imaging devices today offer HDR images that offer a larger colour space. Hence, encryption using effective methods without significant compromise on the quality of the decrypted file is difficult. As a potential solution to this problem, the researchers propose the use of a new approach of encryption that integrates both a double chaos system and DNA coding in order to perform the encryption process on a bit level for each image. As part of their research and construction of a new encryption technique, they rely on the Arnold algorithm to scram the various components present in an image file. Afterwards, Lorenz chaotic mapping approach is used to achieve an effective double-chaos system. Using DNA coding principles, the researchers transformed the chaotic element of the images and the sequence of the sections integrated into DNA coding. Their study concluded that their approach allows more effective image encryption with less computing overhead.

In a manner similar to [26], Zhu et al. [27] also focused their research study on the encryption of image files for higher security than provided by the conventional all-purpose encryption techniques. The researchers achieve their goal of image encryption by means of developing a 5-D continuous hyperchaotic system. In addition to this, in order to add better diffusion and scrambling of the image files for the encryption process, a dynamic DNA coding approach is used. A multi-round diffusion of the image files is performed through DNA encoding and subsequent decoding on the basis of the pixel value using the plain-text version of the image. The study suggests that the proposed encryption algorithm provides many key benefits in the context of large key spaces.

Kang et al. [28] proposed a symmetric encryption algorithm designed specifically for image encryption. Their proposed algorithm was devised on the basis of the peculiarity of plain-text in the process of DNA coding (PPDC). Their design is built on the basis of the inefficiency and insecurity of many of the existing image encryption algorithms that rely on chaotic systems. To overcome these shortcomings, this study proposes an image encryption algorithm through PPDC, and the same is also tested with symmetric image encryption. Chaotic sequences are used to scramble the image data, and hence Lorenz's hyper-chaotic system is implemented. The resulting encrypted image gets a higher level of security as the permutation stage for encryption of PPDC uses both the image data and secret keys. The resulting encryption algorithm shows decent resistance against popular attacks like differential attacks, statistical attacks, and exhaustive attacks. In addition, the algorithm shows a higher level of efficiency and privacy.

Meftah et al. [29] proposed a symmetric encryption algorithm based on DNA and Huffman coding. The proposed encryption algorithm works by first codifying the secondary DNA key that gets extracted from the primary DNA key by implementing the Huffman coding on the DNA key. In the next sequence of operation, the encryption is carried out by XOR'ing the plain-text image file and the codified DNA sequence. To gain further security with the proposed encryption algorithm, the algorithm makes use of diffusion of the plaintext in ciphertext with the help of a permutation box system. The algorithm is tested thoroughly, and the performance and security results show that their encryption algorithm can be considered an alternative to other existing ciphers.

DNA coding has enabled major improvements and a fundamental shift in the use of encryption with a higher level of efficiency. Privthran et al. [15] argue that the sheer amount of data getting generated and transmitted using the internet has been increasing at a rapid rate, and as such, the amount of sensitive and confidential data is also increasing. Hence, security of all of this data is necessary for a more secure internet. In their research study, these researchers proposed a novel cryptosystem that is based on DNA cryptography that can help keep the data transmissions secure on the internet. In addition to the use of DNA cryptography, the researchers make use of finite automata theory. The entire cryptosystem is composed of three key components, key generator, data sender, and data receiver. In the usual data transmission process, the sender encrypts the data using 256-but DNA-based secret key. A Mealy machine is also placed in between the sender and the receiver to encode the DNA sequence. The results show that their encryption scheme achieves a higher security level and better performance than other existing DNA-based cryptosystems.

In a conclusion, the fields of DNA and game theory is found to have a positive contribution to the field of cryptography. However, our extensive research has not found any stream cipher that adopt a game theory model. This motivates us to bridge the gap between stream ciphers and game theory by adopting the prisoners' dilemma model as a core model in our design.

4 The Proposed Scheme

4.1 Structure Overview

Our proposed stream cipher requires a 256-bit secret key SK and a set of four 8-bit (V1, V2, V3, V4) IVs to produce 512-bit keystream in each iteration. Updating the states of most of the cipher's components depends on the IV values to maximise randomness and avoid any possible correlations. The keystream generator consists of a payoff matrix that will allow the multiple bits of different sequences to play against each other under the prisoner's dilemma paradigm. We will consider the following mathematical notations through the description of the proposed cipher:

- $\bigoplus$: Bitwise XOR operation
- $||$: Bitwise concatenation
- $m \gg n$: Cyclic right shift of m by n bits
- $m \ll n$: Cyclic left shift of m by n bits

The main component of our stream cipher is divided into: key/IV setup, keystream generation, encryption, and DNA coding. The overall design of the proposed stream cipher is illustrated in Fig. 3.

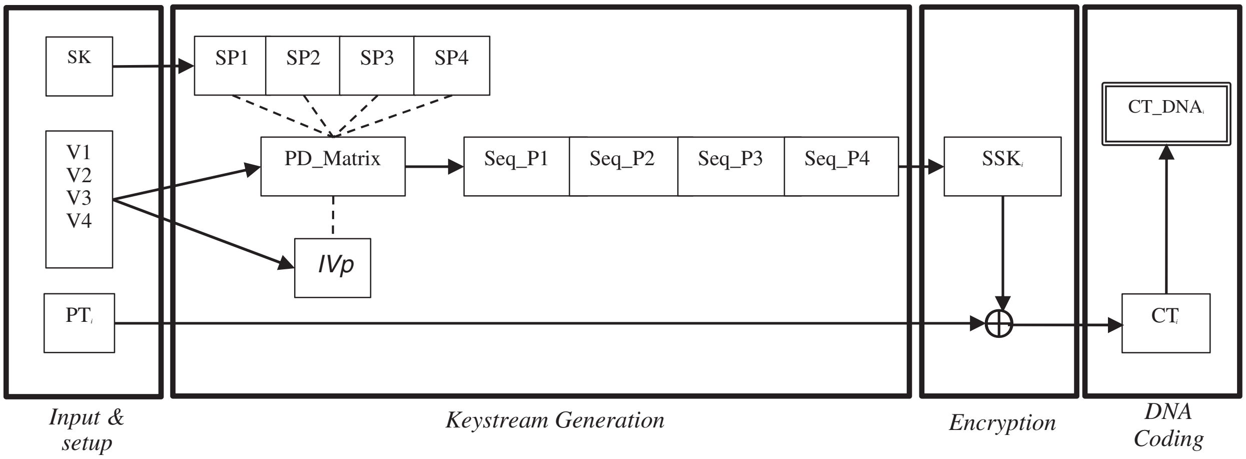

Figure 3: The general structure of the proposed stream cipher

The core concept of our keystream generator relies on a prisoner's dilemma and its matrix that regulates the payoff (rewards) achieved by each player in a game of 2-players. The payoff achieved by a player does not depend only on the player's own action. Instead, it also depends on the action taken by the opponent player. At the beginning of the game, each player chooses either 1 to cooperate or 0 to defect. Predicting the opponent's behaviour is considered a dilemma in the field of game theory, where each player plays with a strategy to achieve the highest possible payoff. In our proposed cipher, the IV values (V1, V2, V3, V4) are updated at every round of generating a new keystream. At each round, two players will only play the game. The first player is always IVp, while the second player is one of the SP1, SP2, SP3, or SP4.

4.2 Key/IV Initialisation

The initialisation stage of our stream cipher starts by setting up the secrete key (SK) and the initial vector (IV). The secret key should be of 256-bit length to ensure the highest level of security against a brute-force attack, and any other attacks related to the length of the key. The second parameter is the IV, which is divided into four 8-bit values denoted by V1, V2, V3, and V4. In the Key/IV setup stage, we use the four IV values to initialise the PD matrix, as shown in Tab. 2.

Table 2: Initialising PD matrix

		P2	
		Cooperate	*Defect*

P1	*Cooperate*	P1 = V1
		P2 = V1
			P1 = V2
			P2 = V3

	Defect	P1 = V3
		P2 = V2
			P1 = V4
			P2 = V4

The four IV values are also used to generate a general IV value denoted by IVp, which results from concatenating (denoted by ||) the four IV values as per Eq. 2. The value of IVp plays a pivot rule in the stage of keystream generation. The Key/IV stage is detailed in Algorithm 1.

$$IVp = V3 \ \| \ \mathrm{V1} \ \| \ \mathrm{V4} \ \| \ \mathrm{V2} \ \| \ \mathrm{V3} \ \| \ \mathrm{V4} \ \| \ \mathrm{V1} \ \| \ \mathrm{V2} \qquad (2)$$

Algorithm 1: Key/IV Setup

```
Input:  256-bit value SK
        8-bit value V1, V2, V3, V4
Output: PD_Matrix[ ][ ],
        64-bit values SP1, SP2, SP3, SP4
        64-bit value IVp
IV[ ] = sort(V1,V2,V3,V4)
// initialize the payoff values of PD_Matrix
PD_Matrix [0] [0] = IV [2], IV [2]
PD_Matrix [0] [1] = IV [0], IV [3]
PD_Matrix [1] [0] = IV [3], IV [0]
PD_Matrix [1] [1] = IV [1], IV [1]
// concatenate Vi to initialize IVp
IVp[ ] = V3 || V1 || V4 || V2 || V3 || V4 || V1 || V2
// convert SK to binary
SK_bin = to_binary(SK)
// split SK_bin into 4 segments
SP1[ ] = SK_bin [0–63]
SP2[ ] = SK_bin [64–127]
SP3[ ] = SK_bin [128–191]
SP4[ ] = SK_bin [192–255]
```

4.3 Keystream Generation

The keystream generation stage involves presenting the SK as a set of four players that play against the opponent player IVp in an iterative PD game. The SK is divided into four players denoted by SP1, SP2, SP3, and SP4 according to Eqs. (3)–(6). Each player has 64-bit to use as a strategy in the game to play against the opponent, where bit 0 represent defection and 1 represent cooperation. Note that IVp is considered the opponent of all SP$_i$ players in all iterations. In the first iteration, IVp plays against SP1 such that IVp[*i*] plays against SP[*i*]. With reference to the PD matrix, SP1 will receive a payoff *x*. This payoff value is appended to the sequence of SP1 until reaching the 64 iterations. In addition, player SP1 will also have an accumulative sum of all its achievements in the previous 64 iterations. The result of the first game between IVp and SP1 is a sequence of 512-bit $(8 \times 64 - bit)$. The same procedure is applied on SP2, SP3, and SP4 when they play against IVp.

$$SP1[\,] = SK[0\text{–}63] \tag{3}$$

$$SP2[\,] = SK[64\text{–}127] \tag{4}$$

$$SP3[\,] = SK[128\text{–}191] \tag{5}$$

$$SP4[\,] = SK[192\text{–}255] \tag{6}$$

Upon completing the four games, the player with the highest payoff is labelled by H0 down to the player with the lowest payoff labelled by H3. The total number of bits generated by the four players is 2048-bit ($4 \times 512 - bit$). Hence, the bits from the four sequences are scrambled to generate 64 keystreams of 32-bit/each, as illustrated in Fig. 4.

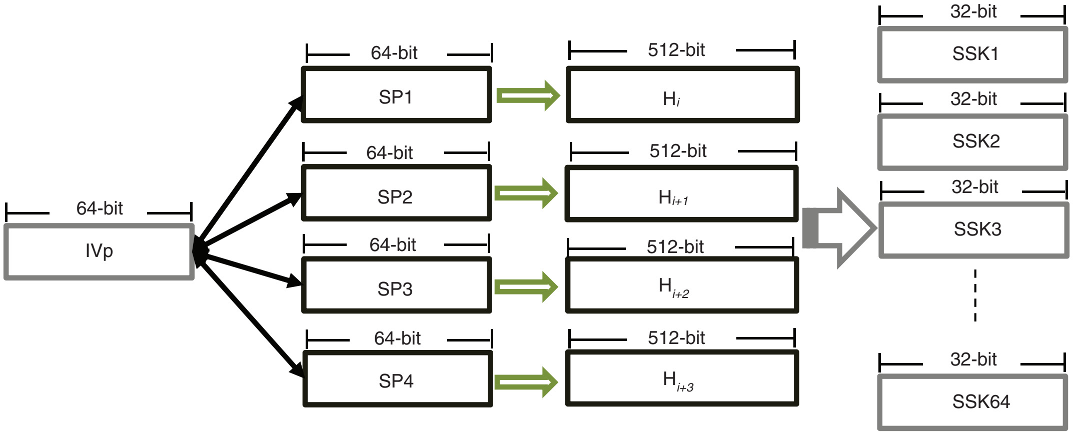

Figure 4: Keystream generation stage

Upon generating the first set of the keystream, a set of operations are applied to re-initialise the PD matrix with the new IVs values (Eqs. (7)–(10)), and apply right and left circular rotation on the initial secrete key SK as formulated in Eqs. (11), (12).

$$\mathrm{V1} = (\mathrm{V1} + \mathrm{V2})mod255 \tag{7}$$

$$\mathrm{V2} = (\mathrm{V2} + \mathrm{V3})mod255 \tag{8}$$

$$\mathrm{V3} = (\mathrm{V3} + \mathrm{V4})mod255 \tag{9}$$

$$\mathrm{V4} = (\mathrm{V4} + \mathrm{V1})mod255 \tag{10}$$

$$\mathrm{SK} >> \mathrm{V1} \oplus \mathrm{V2} \tag{11}$$

$$\mathrm{SK} << \mathrm{V3} \oplus \mathrm{V4} \tag{12}$$

Consequently, the Key/IV setup algorithm is re-called to prepare all the parameters needed to generate a new set of keystreams. The complete keystream generation stage is depicted in Algorithm 2.

Algorithm 2: Keystream Generation

```
Input: 64-bit values SP1, SP2, SP3, SP4
Output: 64 keystream (SSK) of 32-bit value each
// initialize P_sum and Seq_P values of 4 players
for $i = 0$ to $i = 3$ do
    P_sum[$i$] = 0
    Seq_P[$i$] = 0
end for
```

(Continued)

Algorithm 2 (continued)

```
// generate 4 sequences for the 4 players of 512 bit each
for j = 0 to j = 3 do
  for m = 0 to m = 63 do
// the game will be played between IVp and one of the SP players
    payoff = Play(IVp[m], SP[j][m], PD_Matrix)
    P_sum[j] += payoff
    Seq_P[j] = Seq_P[j].append(payoff)
  end for
end for
// descending sort for Seq_P[ ] of the 4 players based on P_sum[ ] of each player
// store Seq_P[ ] of highest P_sum[ ] in H [0] [0] down to the lowest in H [3] [0]
H[ ][ ] = sort(Seq_P[ ], P_sum[ ])
// generate 64 SSK keystream by concatenating segments of H[ ][ ] array
c = 0
for n = 0 to n = 63 do
  i = 0
  SSK[n] = H[i][c:c+7] || H[i+1][c:c+7] || H[i+2][c:c+7] || H[i+3][c:c+7]
  c:c+8
end for
//re-setting the PD payoff V1, V2, V3, V4 values and SK_bin
V1 = (V1 + V2) mod 255
V2 = (V2 + V3) mod 255
V3 = (V3 + V4) mod 255
V4 = (V4 + V1) mod 255
SK_bin >> V1 ⊕ V2
SK_bin << V3 ⊕ V4
//re-call the key setup algorithm
Key_setup (V1, V2, V3, V4, SK_bin)
```

4.4 Encryption and DNA Coding

Reaching the last stage of our stream cipher, which includes both encryption and DNA coding. In encryption, the plain-text bits (PT) are XORed with the previously generated keystream (SSK). The XOR operation continues as long as more streams of PT is received, and the 64 SSK are not completely used.

Once we reach the point where the 64 SSK are used, the keystream generation algorithm is re-called to generate a new set of SSKs, as depicted in Algorithm 3.

Algorithm 3: Encryption

```
Input:  32-bit values PT, SSK
Output: 32-bit values CT
//use the 64 SSK to be exclusively or'ed with PT (bitwise)
while PT != null do
    for $n$ = 0 to $n$ = 63 do
        for bit = 0 to bit = 31 do
            CT[bit] = PT[bit] $\oplus$ SSk[bit]
        end for
    end for
//call keystream generation algorithm to generate new 64 SSK
Keystream_Generation()
end while
```

Upon generating the first 32-bit of Ciphertext (CT), the DNA coding algorithm transfers the CT from its binary representation to a new DNA code with four bases (C, T, G, A). The coding is carried out as per the depicted Algorithm 4. Note that every two adjacent bits are coded to one of the 4 bases. For instance, assume that bits 00 = 'C', 01 = 'T', 10 = 'G, and 11 = 'A', then the DNA code of the CT: '10011100' is 'GTAC'. To ensure higher level of security, the mapping between DNA bases and the binary representation is randomly scrambled at the end of each coding of 32-bit.

Algorithm 4: DNA Coding

```
Input:  32-bit values CT
Output: 32-bit value DNA
//DNA coding is applied on every two adjacent bits of CT
DNA_base[ ] = [C,T,G,A]
for bit = 0 to bit = 31 Step 2 do
    if (CT[bit] = = 0 && CT[bit+1] = = 0) then
        DNA[bit] = DNA_base [0]
    elseif (CT[bit] = = 0 && CT[bit+1] = = 1) then
        DNA[bit] = DNA_base [1]
    elseif (CT[bit] = = 1 && CT[bit+1] = = 0) then
        DNA[bit] = DNA_base [2]
    elseif (CT[bit] = = 1 && CT[bit+1] = = 1) then
        DNA[bit] = DNA_base [3]
```

(Continued)

Algorithm 4 (continued)

```
        end if
    end for
    //Circular right shift is applied on DNA_base[ ]
    DNA_base >> V1 mode 4
```

As for the decryption process, the DNA codes are transferred back to their binary representation according to the generated DNA mapping. Consequently, the plaintext is generated by applying the XOR operation between the ciphertext bits and the corresponding keystream.

5 Security Analysis

5.1 Statistical Tests and Balance

As statistical tests are critical tests for detecting statistical flaws in a given set of numbers, the standard statistical test suites NIST is used to evaluate our stream cipher [30]. The sample size used for the test is 1000 1-Mbit as in [5]. Tab. 3 shows that our stream cipher passed the NIST test successfully.

Table 3: NIST statistical test

Test	p-value	Passing rate	Results
Runs	0. 192011	0.992	*pass*
Cumulative sums	0. 348852	0.988	*pass*
Non-overlapping templates	0. 201441	0.987	*pass*
Overlapping templates	0. 505301	0.987	*pass*
Random excursion variant	0. 312548	0. 981	*pass*
Rank	0. 011180	0.987	*pass*
Linear complexity	0. 356098	0.988	*pass*
Longest run	0. 129104	0.988	*pass*
FFT	0. 021019	0.987	*pass*
Universal	0. 598913	0.985	*pass*
Approximate entropy	0. 830014	0.987	*pass*
Random excursion	0. 219814	0.985	*pass*
Block frequency	0. 242145	0.987	*pass*
Serial	0. 547174	0.985	*pass*
Frequency	0.012556	0.993	*pass*

Passing the NIST tests proofs also that the number of zeros and ones in each keystream is balanced. To visualise this balance, we plot the differences between 500 generated keystreams, as shown in Fig. 5. The figure shows a balanced number of zeros and ones in the generated 500 keystreams, and the differences between zeros and ones in each keystream ranged between 0–2 bits.

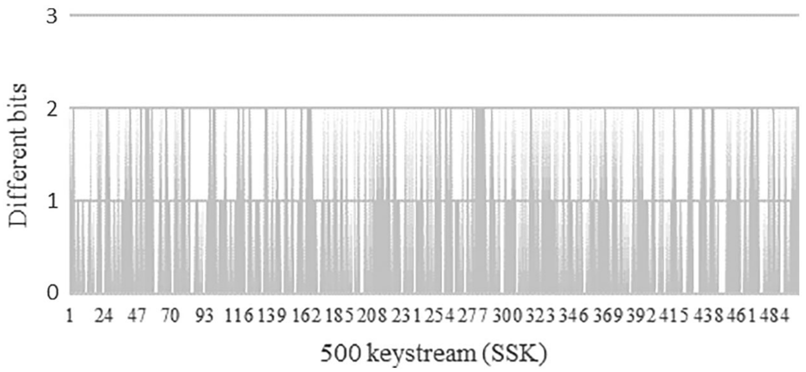

Figure 5: Balance analysis of the keystream

On the other hand, statistical analysis is carried out over the DNA codes results as a ciphertext. Our experiment aims to examine whether the generated DNA codes are statistically biased toward specific DNA bases. Results presented in Tab. 4 shows that the DNA bases are well-balanced when tested over different sets of ciphertext.

Table 4: Balance analysis of DNA bases

# of CT characters	% C	% G	% T	% A
100	25.0%	24.0%	25.0%	26.0%
200	25.5%	24.5%	26.0%	24.0%
500	25.2%	25.4%	24.8%	24.6%
1000	25.3%	25.1%	24.9%	24.7%
5000	25.2%	24.9%	25.1%	24.8%
10000	25.1%	25.0%	25.0%	24.9%

5.2 Avalanche Effect

The Avalanche effect is a cryptographic property that measures the impact of changing one bit of the input parameter on the output resulting from different stream cipher stages. In our case, we measure the avalanche effect of changing one bit in the SK to generate SSKs. We also measure the avalanche effect on the ciphertext. Tab. 5 shows an example of measuring the avalanche effect of changing 1 bit in SK and its impact on the generated SSK. According to this case, around 72% of the bits are changed after flipping only 1 bit in SK.

Similar to the example discussed above, extensive analysis is performed to measure the avalanche effect on a big group of keystream and ciphertext. 1 bit of 256-bit secret key is flipped, and the impact is measured on both keystreams and ciphertext. The results of these experiments are tabulated in Tab. 6.

5.3 Cryptanalysis Attacks

5.3.1 Brute Force Attacks

A brute-force attack relies on guessing the secret key used for encryption. In our case, our stream cipher uses 256-bit SK and four 8-bit IVs to generate a set of SSK in each iteration. Hence, the brute-force attack should make $2^{256} + 2^{32}$ guesses to reveal the key and IV, which is impossible with available computational resources.

Table 5: Example on avalanche effect on keystreams

Original scenario	
Initial 256-bit SK	1000010011011110111001101001100111010000111010111110000110101000000000 101010001000000011000000001100011011011100001101111010010111110111000 010010011110110011110001011100011110011111110100111010110101011110111 00111000110111000000001110001001110011100100010001
Generated 512-bit SSK	A9657AD75299B416B6BFB888F0261783DE88F8F17ACC2F0D7B5E6D11DFBB 7475C85B5B4E4A84CC6BF46C58CD581E9271ED0EB17CFB3FDEABFD57A2 41DDD06AE7
1-bit modification scenario	
1-bit Flipped in SK	1**0**0001001101111011100110100110011101000011101011111000011010100000000 010101000100000001100000000110001101101110000110111101001011111101110 000100100111101100111100010111000111100111111101001110101101010111101 1100111000110111000000001110001001110011100100010001
Generated 512-bit SSK	A9159986C5B2A762CF1B92B74F9B058936580B9CACAD2208264AD4116351 DD144CB1A6D1862A6F60AED93DEFD51E9271ED88B17CFB3FDA9AFD14 2241DDD06AE7

Table 6: Evaluating avalanche effect on SSK and CT

SK *vs*. SSK		SK *vs*. CT	
No. of generated SSKs	300	Size of CT (bits)	5000
Lowest % of changes	69%	Lowest % of changes	68%
Highest % of changes	76%	Highest % of changes	74%
Average % of changes on SSKs	72.49%	Average % of changes on CT	71%

5.3.2 Known-Plaintext Attack

In known-plaintext attack, the attacker seeks to access both the ciphertext and its corresponding plain-text to reveal the secret keys. This will enable the attacker to decrypt consequent messages. In our proposed cipher, the ciphertext is generated through an XOR operation between the plain-text bits and the secret keys (SSK) to generate a completely new binary sequence. A random DNA bases then replace the binary sequence according to the rules described in Algorithm 4. The randomness of selecting the DNA bases guarantee that two similar plain-texts will generate completely different ciphertexts. Hence, our cipher is secure against known-plaintext attacks.

5.3.3 Ciphertext-Only Attack

In ciphertext-only attack, attacker seeks to access only ciphertexts sequences to recover many plain-text sequences from guessing the secret keys. Once the secret key is guessed, the attacker will try to decrypt all other ciphertext sequences using that secret key. However, our stream cipher depends on a PD matrix to generate random numbers that is later formulated as encryption keys. Therefore, given two identical ciphertexts, the generated DNA sequences for these sequences will be entirely different. In conclusion, our cipher is secure against ciphertext-only attacks.

5.3.4 *Differential Attack*

In a differential attack, the attacker aims to reveal the key or at least reduce the time needed to reveal the key by conducting intensive analysis onset of plaintext-ciphertext pairs. Our stream cipher is considered secure against this kind of attack since the SSK is not repeated during the SSK generation process. Every time a new SSK is generated, the PD matrix and the IVs values are all re-initialised by new random numbers. Hence, the same plain-text will be converted into a completely different ciphertext. Thus, our cipher is secure against differential attacks.

6 Complexity Analysis

In this section, we analyse the complexity of our four algorithms: Key/IV setup, keystream generation, encryption and DNA coding. The analysis results tabulated in Tab. 7 show that our stream cipher has an efficient design. The Key/IV setup has the lowest complexity of O(1), and the encryption algorithm of O($n$log$n$).

Table 7: Complexity analysis

	Key/IV setup	Keystream generation	Encryption	DNA coding
Complexity	O(1)	O(n)	O($n$log$n$)	O(log$n$)

7 Performance Analysis

Our stream cipher is implemented in Python environment installed on a Lenovo with Intel Core i7® machine with 6 GB RAM, 500 GB HDD, 128 GB SSD, and Microsoft Windows 10 as the operating system. The dataset used in our experiment is the 20 Newsgroup dataset, which is used in similar research papers [31,15].

The performance of our stream cipher is measured and compared against the well-known stream ciphers: AES-128, RC4 ciphers [32,5]. Our comparative analysis also considers the DNA-based cipher proposed by Pavithran et al. [15]. The proposed cipher was found to have an average throughput of 1230.77 Mbit/s compared to AES-128, an average throughput of 1888.89 Mbit/s, and RC4, with an average throughput of 1879.3 Mbit/s. As for the DNA-based cipher in [15], the throughput is found to be very low at 0.007 Mbit/s as tabulated in Tab. 8. However, our algorithm's encryption throughput has outperformed the DNA-based cipher and offers a secure alternative stream cipher design against cryptanalysis attacks.

Table 8: Performance analysis

	AES-128	RC4	Pavithran et. al [15]	Our cipher
Throughput (Mbit/s)	1888.89	1879.3	0.008	1230.77

8 Conclusion

In this paper, we introduced a novel stream cipher based on a game theory model known as prisoners' dilemma. The main goal of our research is to design an encryption algorithm that is fast and secure against cryptanalytic attacks. The keystream generator of our cipher operates by allowing different bits to act as players to play a PD game iteratively. The player's behaviour (strategy) is unpredictable, and hence, the generated DNA sequences are random and unpredictable. The stream cipher is composed of different components, including: the key/IV initialization, keystream generation, encryption, and DNA coding. Extensive statistical, security, and performance tests were carried out. The results showed that the proposed stream cipher is secure and fast to secure exchanged data between multiple parties communicating over public communication channels. The cipher is found secure against statistical and cryptanalytic attacks, and it can achieve a throughput of about 1230 Mbit/s.

Acknowledgement: The author would like to thank Oracle-Saudi Arabia and the Arab Open University-Saudi Arabia for supporting this study.

Funding Statement: The author received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] P. Ekdahl and T. Johansson, “A new version of the stream SNOW,” in *Selected Areas in Cryptography*, Berlin, Heidelberg: Springer, pp. 47–61, 2003.

[2] C. De Cannière, O. Dunkelman and M. Knežević, “KATAN and KTANTAN—A family of small and efficient hardware-oriented block ciphers,” in *Cryptographic Hardware and Embedded Systems-CHES 2009*, Vol. 5747, Berlin, Heidelberg: Springer, pp. 272–288, 2009.

[3] D. Moon, D. Kwon, D. Han, J. Lee, G. Ryu *et al.,* “T-function based stream cipher TSC-4,” *ECRYPT Stream Cipher Project*, 2006. [Online]. Available: https://www.ecrypt.eu.org/stream/papersdir/2006/024.pdf (Accessed 18 October 2021).

[4] C. Jansen, T. Helleseth and A. Kholosha, “Cascade jump controlled sequence generator and pomaranch stream cipher,” in *New Stream Cipher Designs*, Vol. 4986, Berlin, Heidelberg: Springer, pp. 224–243, 2008.

[5] J. Teh and A. Samsudin, “A stream cipher based on spatiotemporal chaos,” *IETE Journal of Research*, vol. 63, no. 3, pp. 346–357, 2017.

[6] A. Kosek, *An exploration of mathematical applications in cryptography*, Ohio: Ohio State University, 2015.

[7] F. Cavaliere, J. Mattsson and B. Smeets, “The security implications of quantum cryptography and quantum computing,” *Network Security*, vol. 2020, no. 9, pp. 9–15, 2020.

[8] X. Wang and Y. Su, “An audio encryption algorithm based on DNA coding and chaotic system,” *IEEE Access*, vol. 8, pp. 9260–9270, 2020.

[9] J. Katz, “Bridging game theory and cryptography: Recent results and future directions,” *Theory of Cryptography*, vol. 4948, pp. 251–272, 2008.

[10] X. Liang, Z. Yan, R. Deng and Q. Zheng, “Investigating the adoption of hybrid encrypted cloud data deduplication with game theory,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 32, no. 3, pp. 587–600, 2021.

[11] S. Ergün, B. Kırlar, S. Gök and G. Weber, “An application of crypto cloud computing in social networks by cooperative game theory,” *Journal of Industrial & Management Optimization*, vol. 16, no. 4, pp. 1927, 2020.

[12] J. Zhou, J. Li and X. Di, “A novel lossless medical image encryption scheme based on game theory with optimized ROI parameters and hidden ROI position,” *IEEE Access*, vol. 8, pp. 122210–122228, 2020.

[13] B. Kırlar, S. Ergün, S. Gök and G. Weber, “A game-theoretical and cryptographical approach to crypto-cloud computing and its economical and financial aspects,” *Annals of Operations Research*, vol. 260, no. 1, pp. 217–231, 2018.

[14] S. Liu, A. Paul, G. Zhang and G. Jeon, “A game theory-based block image compression method in encryption domain,” *The Journal of Supercomputing*, vol. 71, no. 9, pp. 3353–3372, 2015.

[15] P. Pavithran, S. Mathew, S. Namasudra and P. Lorenz, “A novel cryptosystem based on DNA cryptography and randomly generated mealy machine,” *Computers & Security*, vol. 104, no. 1, pp. 102160, 2021.

[16] S. Almanasra and K. Suwais, “3D model for optimising the communication topologies of iterated N-players prisoners’ dilemma,” *International Journal of Applied Decision Sciences*, vol. 11, no. 4, pp. 420–439, 2018.

[17] DNA structure,” 2012. [Online]. Available: https://sites.google.com/site/imlovingmygenes/dna-structure (Accessed 14 October 2021).

[18] K. Wang, M. Du, D. Yang, C. Zhu, J. Shen *et al.,* “Game-theory-based active defense for intrusion detection in cyber-physical embedded systems,” *ACM Transactions on Embedded Computing Systems*, vol. 16, no. 1, pp. 1–18:21, 2016.

[19] J. Henno, H. Jaakkola and J. Mäkelä, “Creating randomness with games,” in *2019 IEEE 23rd Int. Conf. on Intelligent Engineering Systems*, Gödöllő, Hungary, 2019.

[20] A. Hnaif and M. Alia, “Mobile payment method based on public-key cryptography,” *International Journal of computer networks and communications*, vol. 7, no. 2, pp. 81–92, 2015.

[21] M. Zhao, L. Chen, J. Xiong and Y. Tian, “A three-party repeated game model for data privacy in mobile edge crowdsensing of IoT,” in *13th EAI Int. Conf. on Mobile Multimedia Communications*, Harbin, 2020.

[22] X. Li, C. Zhou and N. Xu, “A secure and efficient image encryption algorithm based on DNA coding and spatiotemporal chaos,” *International Journal of Network Security*, vol. 20, pp. 110–120, 2018.

[23] M. Sohal and S. Sharma, “BDNA-A DNA inspired symmetric key cryptographic technique to secure cloud computing,” *Journal of King Saud University- Computer and Information Sciences*, vol. 34, no. 1, pp. 1417–1425, 2018.

[24] S. Basu, M. Karuppiah, M. Nasipuri, A. Halder and N. Radhakrishnan, “Bio-inspired cryptosystem with DNA cryptography and neural networks,” *Journal of Systems Architecture*, vol. 94, no. 4–5, pp. 24–31, 2019.

[25] Z. Qiu-yu, J. Han and Y. Ye, “An image encryption algorithm based on image hashing, improved chaotic mapping and DNA coding,” *IET Image Processing*, vol. 13, no. 6, pp. 2905–2915, 2019.

[26] Q. Liu and L. Liu, “Color image encryption algorithm based on DNA coding and double chaos system,” *IEEE Access*, vol. 8, pp. 83596–83610, 2020.

[27] S. Zhu and C. Zhu, “Secure image encryption algorithm based on hyperchaos and dynamic DNA coding,” *Entropy*, vol. 22, no. 7, pp. 772, 2020.

[28] Y. Kang, L. Huang, Y. He, X. Xiong, S. Cai *et al.,* “On a symmetric image encryption algorithm based on the peculiarity of plaintext DNA coding,” *Symmetry*, vol. 12, no. 9, pp. 1393, 2020.

[29] M. Meftah, A. Pacha and N. Hadj-Said, “DNA encryption algorithm based on Huffman coding,” *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 13, no. 10, pp. 1–14, 2020.

[30] A. Rukhin, J. Soto and J. Nechvatal, *A statistical test suite for random and pseudorandom number generators for cryptographic applications*, USA: National Institute of Standards and Technology, 2010.

[31] K. Lang, “20 Newsgroups,” 2008. [Online]. Available: http://qwone.com/~jason/20Newsgroups/ (Accessed 10 October 2021).

[32] ECRYPT stream cipher project,” 2012. [Online]. Available: https://www.ecrypt.eu.org/stream/ (Accessed 20 September 2021).