Nat Comput (2013) 12:101–107
DOI 10.1007/s11047-012-9334-9

A novel text and image encryption method based on chaos theory and DNA computing

Majid Babaei

Published online: 1 August 2012

Abstract In today's world, the security of information is associated with valid and reliable encryption algorithms that we have used in our systems. Today, the latest methods for data encryption are based on DNA computing. In this paper, we consider a reliable data encryption algorithm (OTP) which is theoretically unbreakable, but it experiences some disadvantages in its algorithm. These drawbacks have prevented the common use of its scheme in modern cryptosystems. In this research, we include a logistic chaotic map as an input of OTP algorithm. So, the obtained result of 'Matlab Simulation' could prove the efficiency of proposed algorithm in image encryption. In addition to the cryptography of text files, we can propose an interesting encryption algorithm based on a chaotic selection between original message DNA strands and OTP DNA strands. Finally, the empirical results of our proposed algorithm will be compared with AES Open SSl algorithm.

Keywords DNA computing · One-Time-Pad algorithm · Logistic chaotic map · Image encryption · Text encryption

1 Introduction

Recent study shows that DNA Computing can be used as a new efficient method to solve difficult mathematical problems (Adleman 1994). Adleman, in 1994 solved Hamiltonian path problem (HPP) by using DNA computing and its advantages such as vast parallelism and extraordinary information density.

Deoxyribonucleic acid (DNA) is a kind of molecule that encodes genetic information by cellular function. A single strand DNA (Leier et al. 2000; Dove 1999) consists of four different base nucleotides, adenine (A), thymine (T), cytosine (C) and guanine (G). Those nucleotides are able to be bound together in the long sequence. One of the Important DNA roles was presented by Watson–Crick which is described in (Hegedüs et al. 2011). Actually, DNA computing mostly includes three main steps (Cui et al. 2009):

Step 1. Encoding of all candidate's solutions against computational problems,
Step 2. Reaction Control by enzymes and generating all types of data pools that include possible solution to the computational problem,
Step 3. Problem solution's mining by a Polymerase Chain Reaction (PCR).

Until now several cryptographic methods have been proposed and most of them are based on complex mathematical equations. In order to make them more secure, scientists are working to increase their complexity by changing their mathematical equations. Thus, an intruder could not find a quick solution to predict secret keys and break the cryptosystems. While complex equations have been used in the heart of traditional cryptosystems for several years, today, DNA computing breaks these cryptosystems by using its exclusive characteristics (i.e. parallel processing in molecular level). For example, RSA data encryption method has some computational disadvantages and DNA computing is able to attack into different parts of RSA algorithm simultaneously and breaks it in a short period of time. These computational disadvantages are described in the following (Xiao et al. 2012):

M. Babaei (✉)
Department of Computer Engineering, Shahrood University of Technology, Shahrood, Iran
e-mail: babaee.majid@gmail.com

Problem 1: Reliability of RSA algorithm is based on produced factoring large numbers.
Problem 2: Breaking RSA cryptosystem is infeasible on the assumption.

Another example, DES is one of the most popular cryptographic systems. It produces a 64-bit cipher text from a 64-bit plaintext under the control of 56-bit secret key. However, considering the unique characteristics of DNA computing, the security of DES algorithm was questioned. Finally, Dan Boneh, could broke it within a day using an specific secret key and efficient DNA method (Cui et al. 2009).

Although, traditional cryptosystems could be broken using some new methods based on DNA computing, there is a private key encryption algorithm which is reaming secure even against DNA computing. This method is One-Time-Pad (OTP) (Hirabayashi et al. 2009; Tantau 2011) that is absolutely secure in theory. But it has some difficulty in key distribution and generation practically.

In the next section we will describe chaos theory briefly and show some advantages of this theory for using in data encryption algorithms.

2 Chaos theory and logistic map

Chaotic systems are really disordered and chaos theory tries to find the underlying order in a sequence of random data. Nowadays, many researchers are observing several interesting relationships between chaotic behavior and Random number generators (RNGs). As a matter of fact, many properties of chaotic systems like their sensitivity to initial conditions (Tantau 2011) can be considered as a fundamental part of a secret key generator in cryptographic algorithms (Rahimov et al. 2011).

Pseudorandom number generator (PRNG) is used to produce secret key in several cryptosystems (Babaei and Farhadi 2011). Thus, the use of an efficient PRNG with more unpredictability can be useful for preparing a reliable random system as an input of OTP algorithm.

In this paper, we choose logistic map as a chaotic system, because of its advantages to generate several efficient random sequences. The simple mathematical form of the logistic map is given as follows (Babaei and Ramyar 2011):

$$f(x_n) = x_{n+1} = rx_n(1 - x_n) \tag{1}$$

where x_n is the state variable being in the interval [0, 1] and r is the system parameter which might have any value between 1 and 4 (Rahimov et al. 2011). We decided to use $x_0 = 0.8934, 0.9272, 0.9523, 0.9774$ and $r = 3.9996$.

The Bifurcation plot of the Logistic map in different intervals was shown in Rahimov et al. (2011).

In the next section, after introducing OTP method, we will use this chaotic system to provide pad sequence as a random input for OTP algorithm.

3 One-Time-Pad

3.1 OTP structure

One-Time-Pad (OTP), is one of the best data encryption algorithms and theoretically, it is unbreakable (Hirabayashi et al. 2009; Tantau 2011). Each bit or character of a plain text is encrypted by a modular addition which gets a bit or a character from a random key generator. If the secret key is truly random, it is impossible that cipher text breaks theoretically. It was proven that in order to produce a reliable cipher text, each secret key must be used just one time in OTP algorithm. As an example for encryption method based on OTP algorithm, assume that user-A and user-B argue on a string of bit as *pad* (Tantau 2011):

$$pad = k_1 k_2 \ldots k_n \quad (where\ k_i \in \{0, 1\}) \tag{2}$$

It means that pads' bits are chosen in $\{0, 1\}^n$ with uniform probability and also they are used as a common secret key for message encryption. The following sequence is original message that will be encrypted by the generated pad:

$$M = m_1 m_2 \ldots m_n \quad (where\ m_i \in \{0, 1\}) \tag{3}$$

The result of the following function (i.e. *Encrypt(X, Y)*) is encrypted message (i.e. cipher text).

$$c = Encrypt(pad, m) = m \oplus pad \tag{4}$$

To decrypt the cipher text in the receiver side (i.e. user-B), the following function (i.e. Decrypt(X, Y)) is used:

$$Decrypt(pad, c) = (c \oplus pad) \oplus pad \tag{5}$$

Therefore, OTP for data encryption is a reliable method when you pass a secret message through some insecure channels.

3.2 Difficulties and disadvantage

Although, OTP algorithm benefits from its perfect security properties, yet, it has not been used widely in security applications. The OTP algorithm experiences serious drawbacks, mentioned as follows:

First: Each *Pad sequence* should be used only once.
Second: The length of *Pad sequence* for encryption a message with *m* bits is at least *m* bits.

Third: Each *Pad sequence* that is used for data encryption has to be truly random and unpredictable.

If these requirements be satisfied, OTP will really be a secured cryptosystem.

4 DNA cryptography

One of the new attractive fields in cryptography is DNA cryptography. As a matter of fact, the vast parallelism and extraordinary information density are exclusive characteristics of DNA molecule that can be used for cryptographic objectives such as data encryption, authentication, and digital signature (Gehani et al. 1999). In this section, we briefly introduce principle of DNA computing and summarize some disadvantages about this method.

In DNA cryptography, computational processes can be done by some chemical reaction. These processes are controlled accurately and then produce sequence of nucleotides (i.e. A, T, C, and G) as an output for data encryption. It is worked out by specific hybridization between the DNA molecules. Actually, it is formed by specific double helix structure of complementary base pair for encoding data and puts them into operations.

The recently study about molecular computation shows that DNA cryptography is mainly confronted with the following problems:

4.1 Theoretical problems

In 1949, Shannon, in his famous paper described a fundamental model and developmental direction for modern privacy communication. It introduced that a complex mathematical theory should be used as a powerful tool for generating public and private keys in encryption algorithms. So, several cryptosystems based on mathematical equations have been invented over the last decades such as RSA, EIGamal (Gamal 1985), DES and AES. In the contrary, DNA cryptography does not have any mature mathematical background which provides strong support of its theory. This problem is still open as a fundamental subject of DNA cryptography.

4.2 Difficult implementation

Many biological experiments need to be performed to propose a DNA cryptosystem as a reliable method for data encryption (Sakamoto et al. 2000). Scientists perform these experiments in well-equipped labs and by many expensive materials (Ouyang et al. 1997; Bancroft et al. 2001; Ignatova et al. 2008). For these reasons, there are several difficulties to implement a DNA cryptosystem practically.

5 Proposed method for image encryption

In this section, we will present a procedure for image encryption.

5.1 Encryption and decryption algorithm

Step 1. Preparing an original gray scale image as a $m \times n$ matrix (i.e. $A = (m, n)$, where m is the number of rows and n is the number of columns.

Step 2. The matrix elements are converted to the corresponding binary sequence.

Step 3. The binary sequence is encoded into a matrix of nucleotides (i.e. DNA matrix) by the following rules:

$A \leftrightarrow 00, T \leftrightarrow 01, C \leftrightarrow 10, G \leftrightarrow 11$

For example, Original matrix is defined as below with $a_{11} = (200, 100)$:

$$A = \begin{bmatrix} (200, 100) & . & . \\ . & . & . \\ . & . & . \end{bmatrix}_{m \times n}$$

Algorithm converts a_{11} to a binary sequence as below:

$a_{11} = (200, 100) \rightarrow (11001000, 01100100)$

Finally, according to the roles, DNA matrix will be yielded as below:

$$\text{DNA matrix} = \begin{bmatrix} (GACA, TCTA) & . & . \\ . & . & . \\ . & . & . \end{bmatrix}_{m \times n}$$

Step 4. Algorithm divides the DNA matrix into four bits-planes. So after step 3, we will have four DNA sub-matrixes which were extracted from the main DNA matrix

Main DNA matrix = DNA sub-matrix1 & DNA sub-matrix2 & DNA sub-matrix3 & DNA sub-matrix1.

$$\text{DNA sub-matrix1} = \begin{bmatrix} GA & . & . \\ . & . & . \\ . & . & . \end{bmatrix}_{m \times n}$$

$$\text{DNA sub-matrix2} = \begin{bmatrix} CA & . & . \\ . & . & . \\ . & . & . \end{bmatrix}_{m \times n}$$

$$\text{DNA sub-matrix3} = \begin{bmatrix} TC & . & . \\ . & . & . \\ . & . & . \end{bmatrix}_{m \times n}$$

$$\text{DNA sub-matrix4} = \begin{bmatrix} TA & . & . \\ . & . & . \\ . & . & . \end{bmatrix}_{m \times n}$$

Step 5. According to the size of each DNA, the sub-matrix logistic chaotic map generates four different bit sequences with four different seeds.

Step 6. Four random-bit sequences are converted to the corresponding binary matrixes.
Step 7. The binary matrixes are converted to the corresponding DNA matrixes (i.e. logistic chaotic map with seed$_i$ generates sequence$_i$ where $1 \leq i \leq 4$).
Step 8. Two operators (i.e. "ADD" operator for addition and "SUB" operator for subtraction) are defined in Table 1.
Step 9. Four DNA sub-matrixes which were extracted from main DNA matrix (produced in step 4) are added to the chaotic matrixes (generated in step 7). Thus, the result of this operation is four new sub-matrixes.
Step 10. The new DNA sub-matrixes (i.e. the results of previous step) are integrated into a sequence and form an encoded DNA sequence for the original image.
Step 11. The DNA sequence is converted to the corresponding binary sequence by the roles (mentioned in step 3). The result of this step is a secure pad as an input for OTP algorithm.
Step 12. According to the OTP algorithm, if we use a high secure pad for data encryption only one time, it won't be broken theoretically. So in this step the initial binary matrix in step2 convert to a binary sequence. Then "XOR" operation will be performed between the pad and the binary sequence. The result of this step is a new binary sequence that contains the encrypted data.
Step 13. In the final step of the image encryption algorithm, the binary sequence will be transformed into a binary matrix and then the encrypted image will be formed as the result of our proposed algorithm.

Decryption process simply uses only "SUB" instated of "ADD" operator that was used in step 9.

5.2 Result analysis

After a complete explanation of proposed method, we should prove that it is secure enough and efficient as an image encryption method.

Table 1 Addition operation and subtraction operation (Sadeg 2010)

ADD	A	T	C	G
A	A	T	C	G
T	T	C	G	A
C	C	G	A	T
G	G	A	T	C
SUB	A	T	C	G
A	A	G	C	T
T	T	A	G	C
C	C	T	A	G
G	G	C	T	A

5.2.1 *Key space analysis*

One of the main concerns in designing modern crypto systems is preparing an efficient key space for using in high security applications. Our proposed algorithm includes four system parameters for generating a high secure pad (Step 1 to Step 11) and a system parameter for performing "XOR" operation between pad and initial binary sequence. Hence, key space of the proposed algorithm includes five groups of system parameters, so it could be calculated by the following equation:

$$\text{Key Space } = K_1 \times K_2 \times K_3 \times K_4 \times K_5 \tag{6}$$

where $K_i, 1 \leq i \leq 4$ refers to the four different initial values for logistic chaotic maps and K_5 refers to the "XOR" operation in the algorithm.

5.2.2 *Correlation analysis in image encryption*

Efficiency of correlation coefficient is very important feature in new encryption algorithm. It is calculated based on correlation between the encrypted pixels and the original pixels. There are some important points about correlation coefficient that are mentioned below:

- When it is close to 1, it suggests that there is a positive linear relationship between the data columns.
- When it is close to −1, it suggests that a column of data has a negative linear relationship to another column.
- When it is close to 0, it suggests that there is no linear relationship between the data columns.

So when the correlation coefficient is near to 0, intruders are not able to break encrypted image easily. While, if it is near to 1 or −1, they could break it in reasonable time duration. According to these facts, we run our proposed algorithm on a simple gray scale image. And the results are shown in the Fig. 1a–d.

For analyzing the correlation coefficient in the encrypted image, following equations are used:

$$\mathrm{cov}(x, y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)) \tag{7}$$

$$r_{xy} = \frac{\mathrm{cov(x, y)}}{\sqrt{D(x)} \times \sqrt{D(y)}} \tag{8}$$

where x and y are grey value of two adjacent pixels in the image. *E(x)* is the expectation of variable x, *D(y)* is the variance of variable y, and *cov (x, y)* is the covariance between x and y.

In this paper, the correlation coefficients of some randomly sample parts in original image and encrypted image are calculated and shown in Table 5 and Fig. 2. Also pixel grey values and frequency of original image and encrypted

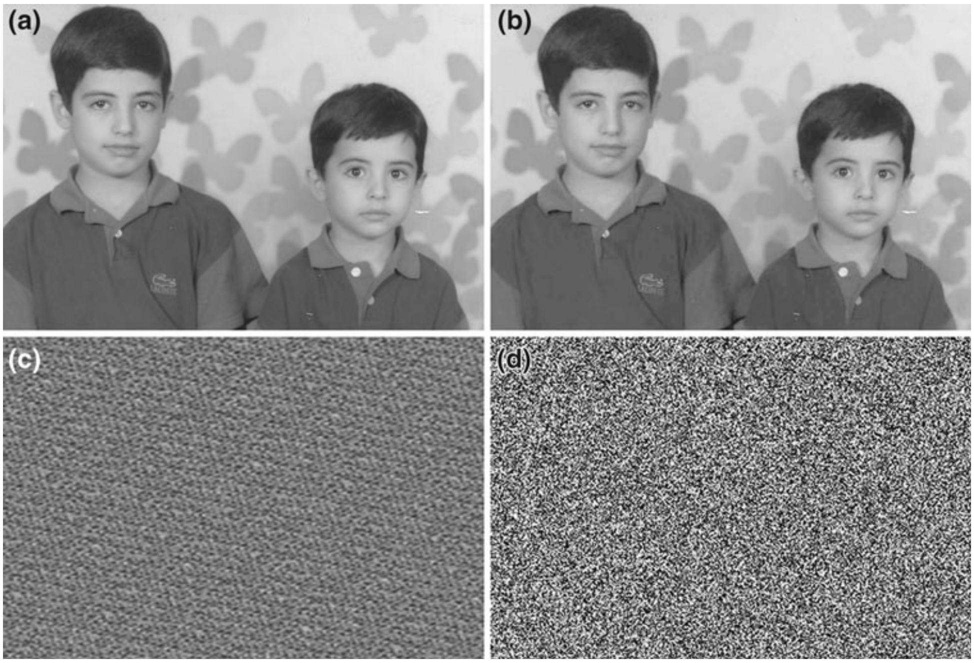

Fig. 1 **a** Original image. **b** Decrypted image with the correct key. **c** Encrypted image. **d** Decrypted image with the wrong key

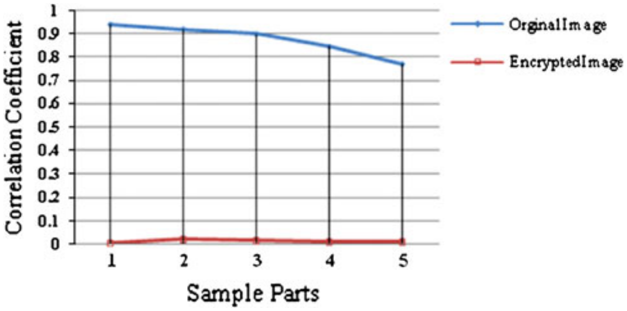

Fig. 2 Correlation coefficients of original image and encrypted image

image are presented in two histograms that are mentioned in Fig. 3a, b.

6 Proposed method for text encryption

In this section we will present a reliable cryptosystem as a text encryption method. Then results are compared with AES open SSL algorithm.

6.1 Encryption and decryption algorithm

Binary data text or image is visualized like ASCII code or brightness levels. In our proposed method, in order to encrypt a text file, we will use advantages of DNA computing, chaos theory and OTP algorithm to achieve maximum security. Therefore, some steps should be taken as the following:

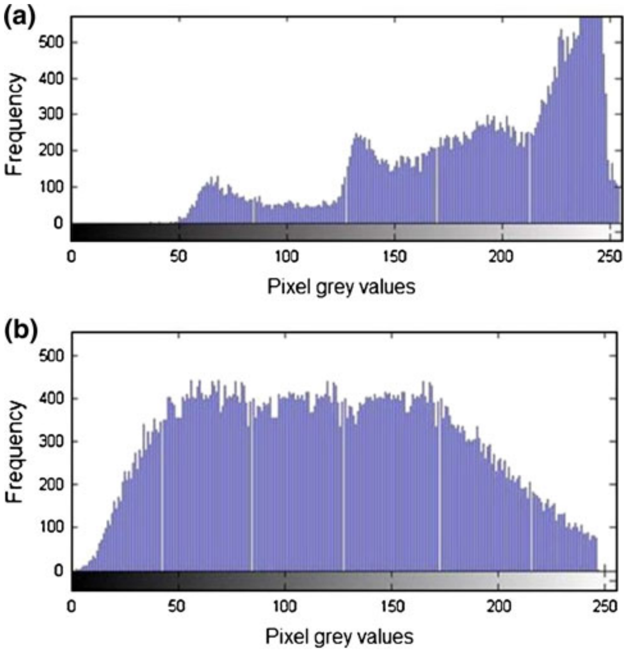

Fig. 3 **a** The grey histogram of the original image. **b** The grey histogram of the encrypted image

Step 1. In the first step, original messages are converted to the equivalent ASCII codes.

Step 2. ASCII codes are converted to the corresponding binary codes which are called Binary (msg), as follows:

Original Message: “Majid”

ASCII: &#8230; &#169; &#251; &#8240;t

Binary (msg):

10000101101010011111101110001001011110100

Table 2 Generated sequences by OTP's algorithm

By using of these parameters, generated 10 stream of alphabetic:
Number of keys = 10, Line length = 48
Key length = 4, Group length = 4
Seed = from CPU's clock (832557801)

1	FsGp	2	Hxnt	3	uJqF	4	sMyM	5	zvDr
6	tAHL	7	tuoq	8	pAsz	9	ABED	10	qpHB

Table 3 Comparison between run times of proposed algorithm and AES method for various file size using 10 rounds and a key of 128 bits (Sadeg 2010)

Length of the file in bytes	Proposed algorithm		AES Open SSL	
	Encryption time (s)	Decryption time (s)	Encryption time (s)	Decryption time (s)
6915116	1.560	1.630	0.348	0.439
22845992	1.141	1.389	0.592	0.866
171132790	11.695	28.338	4.177	11.564
556002942	21.634	20.679	22.408	19.706

Table 4 Comparison between run times of proposed algorithm and AES method for various file size using 10 rounds and a key of 256 bits (Sadeg 2010)

Length of the file in bytes	Proposed algorithm		AES OpenSSL	
	Encryption time (s)	Decryption time (s)	Encryption time (s)	Decryption time (s)
6915116	1.027	1.501	11.657	17.239
22845992	2.867	3.001	38.575	56.658
171132790	20.982	30.786	288.243	273.245
556002942	89.012	100.678	946.444	1383.754

Step 3. Each bit of Binary (msg), can be encoded into a single DNA strand that is called DNA str (msg). As it was shown in (Ignatova et al. 2008), two different groups of nucleotides determine the existing difference between zero-bits and one-bits.

Step 4. In this step, OTP algorithm generates some character sequences based on the specific parameters that are presented in Table 2.

Step 5. OTP sequences are converted to the equivalent binary sequences which are Binary (otp).

Step 6. Each bit of Binary (otp) is mapped to the corresponding DNA strand that is called DNA str (otp).

Step 7. In this step, logistic chaotic map acts as a random selector which binds DNA str (msg) into DNA str(otp) and vice versa. Thus, proposed algorithm generates a long sequence of DNA strands.

Step 8. The long sequence of DNA strands (i.e. result of step 7) is converted to the corresponding binary sequence as cipher text.

The decryption process is the contrary process of encryption.

6.2 Results comparison

In this sub-section we will perform some empirical test on proposed method by a computer system with 2.0 GHz Intel CPU and 3G RAM. So execution time will be evaluated according to the size of the original text file. Actually, a lot of improvements in execution time by our proposed method are shown in Tables 3, 4, 5. In addition, the results were compared with AES Open SSL (Secure Socket Layer). All algorithms were implemented in UBUNTU 8.10 distribution of Linux operating system.

In the proposed algorithm for file encryption we have three main levels. The first level is converting the original text message into corresponding DNA strands. This level includes three sub-levels (i.e. step 1–3). The second level is the generating of OTP sequence and then converting it into corresponding DNA strands. This level like the previous one includes three sub-levels (i.e. step 4–6). The third level

Table 5 Correlation coefficients of original image and encrypted image in the five random sample parts with 2000 pixels

No.	The original image	The encrypted image
Part 1	0.9363	0.0023
Part 2	0.9167	0.0190
Part 3	0.8956	0.0118
Part 4	0.8423	0.0091
Part 5	0.7678	0.0074
Average	0.8717	0.0099

is binding chaotically the results of level 1 into the results of level 2. As it is clear, we have no chance to use DNA characteristics such as parallel processing in first and second level. So these steps (i.e. step 1–6) should be done in the sequential procedure. But when the algorithm comes to the third level, all the DNA strands could be bound together in parallel. Thus, if the key length increases, the more number of DNA strands will be bound together in the third level. So the encryption and decryption time decreases.

7 Conclusion

In this paper, we proposed an efficient method for text and image encryption based on a novel combination of chaos theory and DNA computing. We pointed out that DNA computing has several advantages such as its ability to store an enormous amount of data and perform massive parallel reaction. In addition, chaos theory is one of the great concerns in modern crypto systems and it is very sensitive to initial condition. Moreover, although OTP is a reliable data encryption algorithm which is unbreakable theoretically, it has some disadvantages generally. In this paper, we presented a mix data encryption algorithm composed of chaos theory and DNA computing and then improved performance of OTP algorithm in text and image encryption. Finally, Security analysis and simulation experiment results show that our proposed algorithm could be used as a reliable data encryption method.

References

Adleman LM (1994) Molecular computation of solutions to combinatorial problems. Science 266:1021–1024

Babaei M, Farhadi M (2011) Introduction to Secure PRNGs. Int'l J Communi Network Sys Sci 4(10):616–621. doi:10.4236/ijcns.2011.410074

Babaei M, Ramyar M (2011) Improved performance of LFSR's system with discrete chaotic iterations. Word Appl Sci J 13(7):1720–1725

Bancroft C, Bowler T, Bloom B et al (2001) Long-term storage of information in DNA. Science 293:1763–1765

Cui G, Li C, Li H, Li X (2009) DNA computing and its application to information security field. In: International conference on natural computation, pp 148–152. doi:10.1109/ICNC.2009.27

Dove A (1999) DNA cryptography. J Nat Biotechnol 17:625. doi:10.1038/10813

El Gamal T (1985) A public-key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans Inf Theory 31(4):469–472. doi:10.1109/TIT.1985.1057074

Gehani A, LaBean T, Reif J (1999) DNA-based cryptography. In: 5th annual DIMACS meeting on DNA based computers (DNA 5), MIT, Cambridge, MA, June 1999

Hegedüs L, Nagy B, Eğecioğlu Ö (2011) Stateless multicounter $5' \rightarrow 3'$ Watson-Crick automata: the deterministic case. Nat Comput. doi:10.1007/s11047-011-9290-9

Hirabayashi M, Kojima H, Oiwa K (2009) Effective algorithm to encrypt information based on self-assembly of DNA tiles. Oxford University Press, Nucleic Acids Symposium Series No. 53, 27th September 2009, pp 79–80. doi:10.1093/nass/nrp040

Ignatova Z, Martinez I, Zimmermann K (2008) DNA computing models. Springer book. doi:10.1007/978-0-387-73637-2

Leier A, Richter C, Banzhaf W, Rauhe H (2000) Special cryptography with DNA binary strands. J Biosyst 57(1):13–22

Ouyang Q, Kaplan PD, Liu S et al (1997) DNA solution of the maximal clique problem. Science 278:446–449

Rahimov H, Babaei M, Hassanabadi H (2011) Improving middle square method RNG using chaotic map. Int J Appl Math 137–141.doi:10.4236/am.2011.24062

Rahimov H, Babaei M, Farhadi M (2011) Cryptographic PRNG based on combination of LFSR and chaotic logistic map. Appl Math 2:1531–1534. doi:10.4236/am.2011.212217

Sadeg S, Gougache M, Mansouri N, Drias H (2010) An encryption algorithm inspired from DNA., International Conference on Machine and Web Intelligence (ICMWI). doi:10.1109/ICMWI.2010.5648076

Sakamoto K, Gouzu H, Komiya K et al (2000) Molecular computation by DNA hairpin formation. Science 288:1223–1226

Tantau T (2011) The One-Time Pad algorithm—the simplest and most secure way to keep secrets. Book chapter, Algorithms Unplugged, Part 2, pp 141–146. doi:10.1007/978-3-642-15328-0_15

Xiao G, Lu M, Qin L, Lai X (2012) New field of cryptography: DNA cryptography. Chinese Sci Bull 51(12):1413–1420. doi:10.1007/s11434-006-2012-5