International Journal of Computer Applications (0975 - 8887)
Volume 70 - No. 15, May 2013

Hiding Secret Messages using Artificial DNA Sequences Generated by Integer Chaotic Maps

Eihab Bashier
Department of Mathematics,
Faculty of Science and Arts
Baljurashi, Albaha University,
Albaha, P.O. Box: 1988, Saudi Arabia

Ghaidaa Ahmed
Department of Computer Sciences,
Faculty of Mathematical Sciences,
University of Khartoum, P.O. Box: 321,
Khartoum, Sudan

Hussam-Aldeen Othman
Department of Computer Sciences,
Faculty of Mathematical Sciences,
University of Khartoum, P.O. Box: 321,
Khartoum, Sudan

Rayan Shappo
Department of Computer Sciences,
Faculty of Mathematical Sciences,
University of Khartoum, P.O. Box: 321,
Khartoum, Sudan

ABSTRACT

Almost all the existing DNA steganography techniques are based on DNA sequences from the existing databases such as the GenBank. With the current available computational powers, brute-force attacks on those DNA data sequences can easily be carried out.
In this paper, the technique of generating artificial DNA sequences using chaotic maps running on the domain of integer numbers is introduced. Then, two steganography algorithms for hiding a cipher message in artificial DNA sequences, before sending it to the other party are developped; where the natural techniques by which the genes are hidden, accessed in the DNA and translated are simulated in this paper. The purpose of using this type of chaotic maps is to allow the compatibility between devices with different architictures, such that the extraction and decryption of the hidden message can be carried out correctly. The first algorithm has low computational requirements but vulnerable to statistical attacks when the cipher message is divided into large number of sub-messages. This issue is dealt with in the second algorithm.

General Terms:

DNA steganography, Chaotic maps

Keywords:

Chaotic maps, Cryptography, DNA steganography, Genes, Message hiding

1. INTRODUCTION

Transmitting information securely between any communication parities using any transformation channel has filled the mind of man for centuries. Ancient Egyptians, Chinese and Arabs had created and developed elaborated techniques and methods to ensure information security and privacy. Today with the huge development of digital medium and the revolution in the computer world, mathematical algorithms and protocols are used to achieve the same goal under the names of cryptography and steganography.
Steganography is the art and science of Hiding messages with the purpose to prevent the detection of hidden messages [8]. To use this technique a carrier is needed beside the secret message and also a key to ensure that only the communication parities are authorized to extract the covered message.[2]
Steganography is an old technique that was used by the antient Romans, who shaved the heads of the most trusted slaves and tattooed the secret messages in them and waited for the hair to grow again [13]. The ancient Greeks also used to write in tablets and cover them by wax. Nowadays the steganography methods have developed a lot using digital cover medium carrier such as videos, audios, images[5], [15], text files, TCP/IP headers, documents and DNA steganography [1].
The DNA steganography explores and exploits DNA molecular techniques to hide information. It is completely a new field that had been emerged after the disclosure of the computational ability of DNA. One know advantage of the DNA is that one gram of DNA is capable to store 10^8 terabytes of data. Moreover, the randomness of DNA strands generation makes it more suitable to encrypt data and hide sensitive information[7],[12].
In DNA cryptography and steganography, each of the DNA basis elements is represented by a couple of bits. For example $A = 00$, $T = 01$, $C = 10$ and $G = 11$. Then, each symbol of the language is encoded by a triplet of the DNA bases elements in such a way that the natural construction of the amino acids is simulated.
There are three popular methods of steganography that have been extensively used, namely the insertion method, the complementary pair method and the substitution method. The general idea behind such a method is to choose a reference DNA sequence S secretly, where, only the sender and the receiver are aware of, and incorporate the secret message M into it to obtain a new DNA sequence S'. The second party(receiver) will receive S', identify it, extract the hidden message M and ignore all of the other parts of the sequence, using the appropriate inverse algorithm (See [3], [4] and [10]).
Most of the algorithms based on DNA steganography depend on DNA sequences that are publicly available in many web-sites. The Genbank database of DNA sequences contains 162,886,727 (April 2013) record. On the otherhand, Titan (DOE/SC/Oak Ridge National Laboratory) achieved an impressive 17.59 petaflop/s on the Linpack benchmark using 560640 cores) has been classified as the top supercomputer in the world (November 2012). When dividing the number of the GenBank

Fig. 1. DNA double helix Structure. Source [11]

DNA sequences by the number of Titan's cores, then there are approximately 291 DNA sequences per one core. Assuming that each core can process a DNA record in 30 seconds, Titan will take less than three hours to pass through all the GenBank DNA sequences. This gives the rise that brute force attacks are possible and can be carried out with cheap costs.

In the literature, many authors used DNA and/or chaotic maps for image steganography. See for example references [6], [9], [12] and [13]. Also, almost all the chaotic maps used with the steganography, run on the domain of real numbers. The theoretical seen problem with this type of chaotic maps occur when the encrytion and decryption routines are implemented in two devices with different architictures and hence different representation of the floating points. This might cause the decryption process to fail. This latest problem is our motivation to use chaotic maps running in the ring of integers.

To the best of our knowledge, the idea of generating artificial DNA sequences by using integer chaotic maps for information hiding has not appeared in the literature. In this paper, two steganography techniques for hiding short and long messages, respectively, are developped. The core idea is to use a type of chaotic maps that run over the ring of integers to generate artificial DNA sequences. Then, slicing the encrypted message into sub-cipher messages, inserting each sub-cipher message between a promoter and a terminator (sequences of DNAbases) and insert them in random positions of the generated DNA sequence and send the resulting DNA sequence to the other party.

The rest of this paper is organized as follows. In Section 2 we give some necessary backgrounds about how the genes are accessed and expressed in the DNA. In Section 3 we discuss the construction of artificial DNA sequences by using chaotic maps running in the domain of integer numbers. In Section 4 we design the two DNA steganography algorithms. Finally, in Section 5 are the conclusions and remarks.

2. PRELIMINARIES AND RELATED WORK

The Deoxyribo nucleotide acid (DNA) is a composition of two twisted strands. Each strand has a backbone formed of sugar molecules called 2'-deoxyribose attached to a phosphate residue. The two strands are twisted together into a double helix, and linked together through molecules called bases. There are four bases for the DNA, adenene (A), thymine (T), cytosine (C) and guanene (G). In the DNA, the bases are paired such that, A is always paired to T and C is always paired to G. These two pairs are called pairs of complementary basis.

Each cell of an organism has a few very long DNA molecules called chromosomes, and certain contiguous stretches of a chromosome encode information for building proteins. Such a stretch of DNA that contains information for building a protein or RNA molecule is called a gene.

Proteins are long chains of molecules called amino acids. The sequence of amino acids for a given protein is encoded by using triplets of nucleotides called codons to specify each amino acid contained in the protein. For example, the codon CCG encodes the amino acid glycine, AUG encodes methionine but also it encodes the beginning of a gene, while the codons TGA, TAA and TAG encode a stop signal.

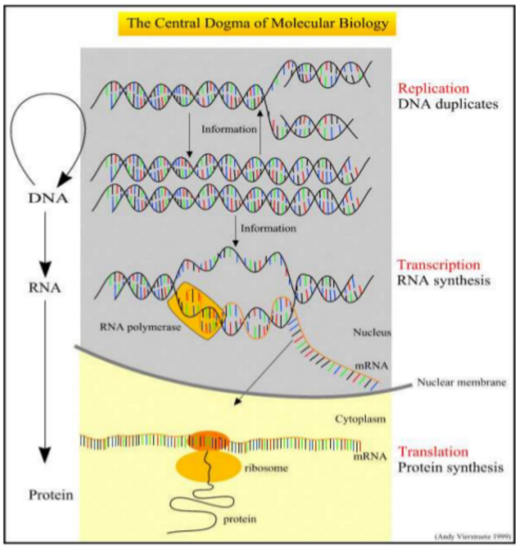

Fig. 2. The central dogma of molecular biology. Source [14]

In Eucaryotic cells, the DNA is found in the nucleus, while proteins are made in the cytoplasm. A messenger molecule is used to carry the information from the nucleus to the ribosomes in the cytoplasm. The flow of genetic information from the DNA to (and from) the RNA to form the protein is described by the central dogma of molecular biology.

The beginning of a gene is recognised due to a promoter (DNA region serves as an indication of a gene). The codon AUG signals the start of a gene. Having recognised the beginning of the gene by the mRNA polymerase, it is copied to the mRNA. This process is called the transcription. The mRNA travels out of the nucleus, into the cytoplasm, where it joins up with it. In the cytoplasm, are amino acids and transfer RNA (tRNA) molecules. Each tRNA molecules has three bases at one end, which represent only one amino acid. The corresponding amino acid, attaches to the other end of the tRNA. The tRNA which has three bases called anti-codons to fit the first three on the mRNA brings it?s amino acid to the ribosome. The next tRNA brings its amino acid, and the two amino acids are chemically joined together. The mRNA moves along by three bases and the first tRNA is released, while a third tRNA arrives with it?s amino acid. The process continue until a whole chain of amino acids (protein) is made according to the code on the mRNA, which was originally copied from the DNA in the nucleolus. This process is called the translation[16].

In the next section, two DNA steganography algorithms based on the concepts of gene expression, transcription and translation are developed.

3. GENERATION OF ARTIFICIAL DNA SEQUENCES

In this section, a type of integer logistic chaotic maps and then use it to generate artificial DNA sequences is first discussed. This section is divided into two subsections. In the first subsection we formulate the chaotic map and show that it exhibit chaotic behaviour.

3.1 Integer chaotic maps

Chaotic maps have been used extensively in cryptography during the past two decades, particulrly for multimedia encryption.

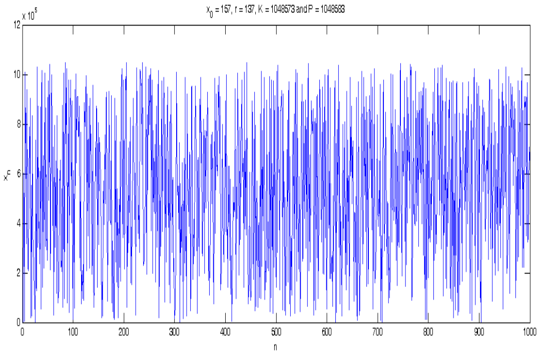

Fig. 3. The chaotic map obtained by setting $x_0 = 157, r = 137$, $K = 1048573$ and $P = 1048583$.

The logistic map is the most popular chaotic map associated with the one dimentional data, whereas the Baker map and the cat map are the most chaotic maps that are used with the images encryption. Chaotic maps are characterized by their sensitivity to changes in initial conditions and the map's parameters. However, the mentioned above chaotic maps run in domains of real intervals. Despite this can be seen as a computational advantage of those chaotic maps, but when ecrypting with chaotic maps of this type, it is expected that embedded devices with different architictures will fail to decrypt the sent messages, because of the difference in representing the floating points in these devices. This has been a motivation to the appearence of some chaotic maps running within the domain of subsets of the integer numbers.

In this section, we discuss a logistic map of the form:

$$x_{n+1} = (rx_n(N - x_n)modP), \qquad (1)$$

where r, N, x_0 and P are positive integers and P is a large prime number with $P > N$. In this chaotic map, r is the interinsic growth rate and N is the carrying capacity. The reason for choosing $P > N$ is to disallow the chaotic map to go to its equilibrium points $x_n = 0$ and $x_n = N$.

```
GENERATECHAOTICINTEGERS $(r, N, P, x_0, L)$
  $k \leftarrow 1$;
  repeat
      $x_{k+1} \leftarrow rx_k(N - x_k)modP$;
      $k \leftarrow k + 1$;
  until $k > L$
  return $x$;
```

In Figure 3, the chaotic behaviour of logistic map 1 is displayed for $x_0 = 157$, $r = 137$, $N = 1048573$ and $P = 1048583$.

3.2 DNA sequences

In this subsection, an integer logistic chaotic map is used to generate random sequences of DNA. To do this, we choose a number $Q < P$ a multiple of 4. Then each one of the four DNA basis is represented by one quarter of the range 0 to $Q - 1$. The random DNA sequence is obtained by running the integer logistic map, obtain $c_n = x_n(modQ)$ and replace c_n by the corresponding DNA basis element.

```
GENERATEARANDOMDNASEQUENCE $(r, N, P, x_0, L, Q)$
  $x \leftarrow$ GENERATECHAOTICINTEGERS$(r, N, P, x_0, L)$;
  $k \leftarrow 1$;
  $J \leftarrow Q/4$;
  repeat
      $s \leftarrow x_k(modQ)$;
  if $s \equiv 0\ (mod\ J)$ then
      $S_k \leftarrow A = 00$;
  else if $s \equiv 1\ (mod\ J)$ then
      $S_k \leftarrow C = 01$;
  else if $s \equiv 2\ (mod\ J)$ then
      $S_k \leftarrow T = 10$;
  else
      $S_k \leftarrow G = 11$;
  endif
  $k \leftarrow k + 1$;
  until $k > L$;
  return $S$;
```

Using values of the parameters: $P = 1048583$, $K = 1048573$, $r = 123$ and $x_0 = 417$, the following DNA sequence is generated by algorithm GENERATEARANDOMDNASEQUENCE:

GCGGGTCGAACAGTTTGAATGATACGAAACTCCGTTCAGGGGACTCAGGCATTGTCTAAAACAAGCACCCTCAACCCCCATTGACCCCTAGGGGAACGCGCGGGGAAATAGCACCGGGAGATAGAGAGAAACTTGGTGCG CCATTCGTGCGGCTTAAAGAAAGTGACACCTACGGTCCTCGGCCGAAACTCGGATACGTTGCCACACCTA CAGTGCCATGTTTGATTTTTTGCTTGCATGGACCGGTACC

By changing the initial condition to $x_0 = 416$ the following DNA sequence is obtained:

ATCGTCAAATTCGCTCACCAATCCACAGCAAGAGACCCCGTCAGTTAACGCCCTGCCTAATAGCCTTCATGAAAGGAATCAAGTTTTTCCATGCATCTGTGAGGGGACGCGTTCTCGTTCGCGGGGTCCATGGTAACAGCGT CTCATGTGGGAGATGGGTCTCGTGTAAAGCGTGGGGGAAAAGTTCTTAAAGCAGCGCGACGTGGCCCAAGG ATAGACTTTGATTATATGGTGTCGGTTAAGTTATGGC

Other DNA sequences are obtained by making slight changes to the initial condition and the parameter r, resulted in completely different DNA sequences.

4. THE TWO DNA STEGANOGRAPHY ALGORITHMS

In this section two DNA steganography algorithms are introduced. The first one can be used to hide massages with small sizes, whereas the second one can be use to hide messages of large sizes.

4.1 Algorithm 1

Within this algorithm, it is assumed that the sender wants to encrypt and hide a message M and send it to the receiver. We also assume that the sender and receiver have agreed about a couple (E, D) of symmetric encryption and decryption algorithms. At the beginning the sender and the receiver use a key agreement protocol to agree on a secret symmetric key $\kappa = K\|Pr\|R$, where K is a secure semmetric key, Pr is a random binary sequence acting as the promoter and T is a random binary sequence acting as the terminator. Both the sender and the receiver can extract the triplet (K, Pr, T) from the secure symmetric key κ.

The idea behind this algorithm is that the sender generates an artificial DNA sequence using the GenerateARandomDNASequence algorithm using chaotic map parameters of his choise. The chaotic map parameters need not be known to the receiver. The length of the generated DNA sequence shall be m multiples of the message size, where m is a positive integer. The sender encrypts the message M with the key K giving a cipher message C. Then, he slices the cipher message C into n sub-cipher messages $C_1, \ldots, C_n$ (acting as genes), not necessarily equal in length and preceed any sub-cipher message C_j, $j = 1, \ldots, n$ by the promoter Pr and follow it by the terminator T giving $\widetilde{C}_j$, $j = 1, \ldots, n$. The sender also slices the DNA sequence into $n + 1$ sub-DNA sequences $S_0, \ldots, S_n$ and inserts $\widetilde{C}_j$ between S_{j-1} and S_j for $j = 1, \ldots, n$. The resulting DNA sequence is $\widetilde{S} = S_0||\widetilde{C}_1||S_1|| \ldots ||\widetilde{C}_n||S_n$. Finally the sender sends the new DNA sequence $\widetilde{S}$ to the other party.

Symbol	Code	Symbol	Code	Symbol	Code	Symbol	Code
A	AGC	B	GCT	C	CTA	D	TAG
E	GAC	F	ATT	G	GCC	H	CAT
I	TAC	J	GGA	K	ACT	L	GGC
M	CGC	N	TAT	O	GAT	P	AAG
Q	GTT	R	CAG	S	TTC	T	CAA
U	AGG	V	GTC	W	CGT	X	TGA
Y	TGG	Z	GCG		GCC	.	CCA

The sender uses the following algorithm encrypt and hide the message.

```
ENCRYPTANDHIDE $(\kappa, M)$
Extract the key $K$, the promoter $Pr$ and the terminator $T$ from $\kappa$;
Choose positive random integers $m, r, N, P, x_0$ and $L$;
$\ell \leftarrow m|M|$;
$\mathbf{D} \leftarrow$ GENERATEARANDOMDNASEQUENCE$(r, N, P, x_0, \widetilde{\ell}, L)$;
$C \leftarrow E_K(M)$;
Slice $C$ into $n$ sub-cipher messages $C_1, C_2, \ldots, C_n$;
$\widetilde{C}_j \leftarrow Pr\|C_j\|T$, for $j = 1, \ldots, n$;
Slice $S$ into $n+1$ sub-sequences $S_0, S_1, S_2, \ldots, S_n$;
Constructs the DNA $\widetilde{S} \leftarrow S_0\|\widetilde{C}_1\|S_1\|\widetilde{C}_2\|\ldots\|;\widetilde{C}_n\|S_n$;
return $\widetilde{S}$;
```

To explain how algorithm 1 works algorithm, suppose that the language symbols are encoded by the DNA codes in Table 4.1. The sender first generates a random DNA sequence S = CTAGATGGCCGCTCCACCCAAAGGACGAAGTGCTAGTTGTTTGCCAGTGATTCAACAC-GAGGCAGCGATCCAAGACGCTAAGT

Then, he encrypts the message "HELLO WORLD." by XORing it with the key "OTHMAN" and uses the codes in Table 4.1 to convert the resulting cipher message into the DNA sequence C = TAAGACGGGTAGGGGCCCTGATATAACTAACGAGCT. Then, the promoter "Ahmed" and terminator "Shappo" (which are assumed to be obtained from the key agreement protocol) are encoded from Table 4.1 as

$$AHMED \equiv AGCCATCGCGACTAG$$

and

$$SHAPPO \equiv TTCCATAGCAAGAAGGAT$$

, respectively. The sender then slices his encrypted message into $C_1 = TAAGACGGGTAGGGGCCC$ and $C_2 = TGATATAACTAACGAGCT$, respectively. Adding the promoter and terminator at the begining and end of each of the two sub-cipher messages C_1 and C_2 gives the DNA sequences $\widetilde{C}_1$ = AGCCATCGCGACTAGTAAGACGGGTAGGGGCCCTTCCATAGCAAGAAGGAT and $\widetilde{C}_2$ = AGCCATCGCGACTAGTGATATAACTAACGAGCTTTCCATAGCAAGAAGGAT

Finally, the sender inserts the two sub-cipher messages in random positions of the DNA sequence S to obtain the new DNA sequence

$\widetilde{S}$ =CTAGATGGCCGCTCCACCCAAAGGAGCCATCGCGACTAGTAAGACGGGTAGGGGCCCTTCCATAGCAAGAAGGAT ACGAAGTGCTAGTTGTTTGCCAGTGATTCAAAGCCATCGCGACTAGTGATATAACTAAC-GAGCTTTCCATAGCAAGAAGGAT CACGAGGCAGCGATCCAAGACGCTAAGT and sends $\widetilde{S}$ to the receiver.

On receiving the DNA message $\widetilde{S}$ from the sender, the receiver extracts the key K, the promoter Pr and the terminator T. Then, he searches the encrypted blocks C_j's using the promoter Pr and the terminator T. Finally, the receiver uses the key K to decrypt the cipher message $C = C_1\|\ldots\|C_n$. The receiver applies the following algorithm to extract the hidden encrypted message.

```
DETECTANDDECRYPT $(\kappa, \widetilde{S})$
$\widetilde{\kappa} \leftarrow D_\kappa(\widetilde{K})$;
extract $K$, $Pr$ and $T$ from $\kappa$;
look for $P_r$ and $T$ in $\widetilde{S}$ to extract $C_1, C_2, \ldots, C_n$;
$\widetilde{C} \leftarrow C_1\|C_2\|\ldots\|C_n$;
$M \leftarrow D_K(\widetilde{C})$;
```

Supposing that the receiver has received the message $\widetilde{S}$ =CTAGATGGCCGCTCCACCCAAAGGAGCCATCGCGACTAGTAAGACGGGTAGGGGCCCTTCCATAGCAAGAAGGAT ACGAAGTGCTAGTTGTTTGCCAGTGATTCAAAGCCATCGCGACTAGTGATATAACTAAC-GAGCTTTCCATAGCAAGAAGGAT CACGAGGCAGCGATCCAAGACGCTAAGT from the sender, he looks for the promoter AGCCATCGCGACTAG and terminator TTCCATAGCAAGAAGGAT in $\widetilde{S}$ which each is located in two positions. He extracted the DNA sub-sequences between, gathers them and decrypt with the symmetric key OTHMAN to obtain the original message "HELLO WORLD.".

4.2 Algorithm 2

In the first algorithm both the promoter and terminator are fixed for all the encrypted submessages. If the number of submessages is large then that may may cause the promoter and terminator to be noticeable for adversaries. The first algorithm can be used to encrypt and hide short messages but is not suitable for long messages.

In this subsection, we make improvements to the first algorithm to allow the promoter and terminator varying for each encrypted submessage, before hiding it in the DNA.

In this algorithm, the sender chooses two sets of the logistic map parameters (r_1, N_1, P_1, x_0^1) and $(r_2, N_2, P_2, x_0^P, x_0^T)$. The first set together with two more parameters ℓ and m is used to generate the artificial DNA sequence, whereas the second sequence is used to generate sequences of promoters Pr_j and terminators T_j, subject to the number of the sub-cipher messages.

The sender chooses two parameters n and ℓ, where n is the number of sub-cipher messages. To generate the promoters Pr_j for $j = 1, \ldots, n$, the sender uses the integer logistic map to generate a chaotic sequence y^P of length $n \cdot \ell$ and chooses $x_j^P = y_{j\ell}^P$. Then, he obtains $Pr_j = (x_j \oplus K) \oplus Pr_{j-1}$ with $Pr_0 = Pr$. A similar procedure is used to obtain the sequence of terminators T_j. In Figure 4 we show the procedure to obtain the sequences of promoters and terminators, where $X \in \{Pr, T\}$ and $Z_j \in \{Pr_j, T_j\}$.

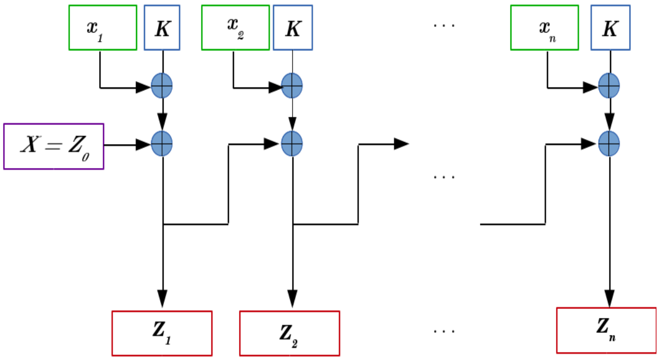

Fig. 4. Generation of sub-promoters and sub-terminators using chaotic maps.

Having generated the sequences of promoters and terminators, the sender continues as in algorithm 1, by constructing $\widetilde{C}_j = Pr_j\|C_j\|T_j$ for $j = 1, \ldots, n$, constructing the new DNA sequence $\widetilde{S} = S_0\|\widetilde{C}_1\|S_1\|\ldots\|\widetilde{C}_n\|S_n$, encrypting the parameters r_2, N_2, P_2, n and ℓ yielding $\widetilde{\kappa}$ and finally sending the couple $(\widetilde{\kappa}, \widetilde{S})$ to the other party.

```
ENCRYPTANDHIDE $(\kappa, M)$
Extract the key $K$, the promoter $Pr$ and the terminator $T$ from $\kappa$;
Choose random positive integers $m$ and $r, N, P, x_0$ and $L$;
$\widetilde{\ell} \leftarrow m|M|$;
$S \leftarrow$ GENERATEARANDOMDNASEQUENCE$(r_1, N_1, P_1, x_0^1, \widetilde{\ell}, L)$;
```

```
$C \leftarrow E_K(M)$;
Choose two random integers $n, \ell \in \mathbb{Z}^+$ and set $L \leftarrow n \cdot \ell$;
$y^P \leftarrow$ GenerateChaoticIntegers$(r_2, N_2, P_2, x_0^P, L)$;
$y^T \leftarrow$ GenerateChaoticIntegers$(r_2, N_2, P_2, x_0^T, L)$;
for $j \leftarrow 1$ to $n$ do
    $x_j^P \leftarrow y_{j\ell}^P$;
    $Pr_j \leftarrow (x_j^P \oplus K) \oplus Z_j (Z_0 \equiv Pr)$;
    $x_j^T \leftarrow y_{j\ell}^T$;
    $T_j \leftarrow (x_j^P \oplus K) \oplus Z_j (Z_0 \equiv T)$
endfor
Slice $C$ into $n$ sub-cipher messages $C_1, C_2, \ldots, C_n$;
$\widetilde{C}_j \leftarrow Pr_j \| C_j \| T_j$, for $j = 1, \ldots, n$;
Slice $S$ into $n+1$ sub-sequences $S_0, S_1, S_2, \ldots, S_n$;
The sender constructs the DNA $\widetilde{S} \leftarrow S_0 \| \widetilde{C}_1 \| S_1 \| \widetilde{C}_2 \| \ldots \|; \widetilde{C}_n \| S_n$;
$\widetilde{\kappa} \leftarrow E_K(r_2 \| N_2 \| P_2 \| n \| \ell)$;
return $(\widetilde{\kappa}, \widetilde{S})$;
```

On the other hand, when the receiver receives the parameters $\widetilde{\kappa}$ and the DNA sequence $\widetilde{S}$, he decrypts $\widetilde{\kappa}$ to get the parameters r_2, N_2, P_2, n and ℓ. Then, he compute the chaotic sequences y^P and y^T from which he can obtain the sequences x_j^P and x_j^T and compute the sequences of the promoters $Pr_j = (x_j^P \oplus K) \oplus Z_{j-1}$ $(Z_0 = Pr)$ and terminators $T_j = (x_j^T \oplus K) \oplus Z_{j-1}$ $(Z_0 = T)$ for $j = 1, \ldots, n$. He then looks for the encrypted blocks C_j's using the promoters Pr_j and the terminators T_j. Finally, the receiver uses the key K to decrypt the cipher message $C = C_1 \| \ldots \| C_n$. The receiver applies the following algorithm to extract the hidden encrypted message.

```
DetectAndDecrypt $(\kappa, \widetilde{\kappa}, \widetilde{S})$
$\widetilde{\kappa} \leftarrow D_\kappa(\widetilde{K})$;
extract $K$, $Pr$ and $T$ from $\kappa$;
extract $r_2$, $N_2$, $P_2$, $n$ and $\ell$ from $D_K(\widetilde{\kappa})$;
$y^P \leftarrow$ GenerateChaoticIntegers$(r_2, N_2, P_2, x_0^P, L)$;
$y^T \leftarrow$ GenerateChaoticIntegers$(r_2, N_2, P_2, x_0^T, L)$;
for $j \leftarrow 1$ to $n$ do
    $x_j^P \leftarrow y_{j\ell}^P$;
    $Pr_j \leftarrow (x_j^P \oplus K) \oplus Z_j (Z_0 \equiv Pr)$;
    $x_j^T \leftarrow y_{j\ell}^T$;
    $T_j \leftarrow (x_j^P \oplus K) \oplus Z_j (Z_0 \equiv T)$
endfor
look for $P_r$ and $T$ in $\widetilde{S}$ to extract $C_1, C_2, \ldots, C_n$;
$\widetilde{C} \leftarrow C_1 \| C_2 \| \ldots \| C_n$;
$M \leftarrow D_K(\widetilde{C})$;
```

5. CONCLUSIONS AND REMARKS

In this paper, the chaotic behaviour of a logistic map running on the domain of integer numbers is discussed. This logistic map is then used to generate artificial DNA sequences that are used to hide encrypted messages using two hiding algorithms. The first algorithm is efficient for hiding short messages, but is not suitable to hide long messages, whereas the second algorithm - which is an improvement to the first one- can be used to hide both short and long encrypted messages.

The advantage of the first algorithm is the low computational requirements to hide and extract the cipher message. But, the noticeable shortage with this algorithm appears when the cipher message is divided into a large number of sub-massages. That makes the algorithm volnerable to statistical attacks, due to the repetition of the sequences of promoters and terminators which indicate the beginnings and endings of the sub-cipher messages. On the other hand, the second algorithm which generates sequences of promoters and terminators for hiding the sub-cipher message blocks, solves the problem of the statistical attacks, but is more complex than the first algorithm. Our tests on the second algorithm showed us that the algorithm is still absolutely ideal for hiding cipher messages with both small and large divisions into sub-messages.

The advantage that is introduced by the two hiding algorithms in this paper is that regardless of the architicture differences of the two communicating devices, the processes of hiding and extracting messages can be carried our correctly.

6. REFERENCES

[1] G. Ahmed, H. Othman, and R. Shappo. Information hiding using dna stegnography, 2012. B.Sc. dissertation report, Faculty of Mathematical Sciences, University of Khartoum.

[2] M. M. Amin, S. Ibrahim, M. Salleh, and M. R. Katmin. Information hiding using stegnography, 2003.

[3] S. T. Amin, Magdy Saeb, and Salah El-Gindi. A dna-based implementation of yaea encryption algorithm, 2011. Availabe online at Accessed at Magdy Saeb homepage: http://www.magdysaeb.net/images/DNAYAEAaminsaeb.pdf, Cited 07/07/2011, Accessed 02/01/2012.

[4] A. Atito, A. Khalifa, and S. Z. Rida. Dna-based data encryption and hiding using playfair and insertion techniques. *Journal of Communications & Computer Engineering*, 2(3):44–49, 2012.

[5] S. Chakraborty and S.K. Bandyopadhyay. Two stages data-image steganography using dna sequence. *International Journal of Engineering Research and Development*, 2(7):69–72, 2012.

[6] R. Enayatifar, F. Mahmoudi, and K. Mirzaei. Using the chaotic map in image steganography. In *Proceedings of the 2009 International Conference on Information Management and Engineering*, ICIME '09, pages 491–495, Washington, DC, USA, 2009. IEEE Computer Society.

[7] A. Gehani, T.H. LaBean, and J. H. Reif. Dna-based cryptography. *DIMACS Series in Discrete Mathematics and Theoretical Computer Sciences*, 54:233–249, 2000.

[8] N.F. Johnson and S. Jajodia. Exploring steganography: Seeing the unseen. *Computer*, 31(2):26–34, 1998.

[9] A. Kanso and H. S. Own. Steganographic algorithm based on a chaotic map. *Communications in Nonlinear Science and Numerical Simulation*, 17(8):3287 – 3302, 2012.

[10] I.K. Maitra, S. Nag, B. Datta, and S.K. Bandyopadhyay. Steganalysis: Review on recent approaches. *Journal of Global Research in Computer Science*, 2(1):1–5, 2011.

[11] T. Mandge and V. Choudhary. A review on emerging cryptography technique: Dna cryptography. In *IJCA Proceedings on International Conference Technology and Computer Science 2012*, volume ICRTITCS, pages 9–13, 2013.

[12] H. Mousa, K. Moustafa, W. Abdel-wahed, and M. Hadhoud. Data hiding based on contrast mapping using dna medium. *The International Arab Journal of Information Technology*, 8(2):147–154, 2011.

[13] M. R. Torkaman, N.S. Kazazi, and A. Rouddini. Innovative approach to improve hybrid cryptography by using dna steganography. *International Journal of New Computer Architectures and Their Applications*, 2(1):225–236, 2012.

[14] A. Vierstraete. The central dogma of mollecular biology. http://users.ugent.be/ avierstr/principles/centraldogma.html, accessed 01/03/2013.

[15] Q. Zhang, X. Xue, and X. Wei. A novel image encryption algorithm based on dna subsequence operation. *The Scientific World Journal*, 2012:10, 2012.