# A Compression Algorithm for DNA Palindrome Compression Technique

Dr.D.Suneetha ,Dr.D.Rathna Kishore,Mr.P.Narendra Babu

Professor, Department of CSE, NRIIT, Vijayawada, A.P, India.
Professor, Department of CSE, NRIIT, Vijayawada, A.P, India.
Associate Professor, Department of CSE, NRIIT, Vijayawada, A.P, India

## Abstract:

Data Compression in Cryptography is one of the interesting research topic. The compression process reduces the amount of transferring data as well as storage space which in turn effects the usage of bandwidth. Further, when a plain text is converted to cipher text, the length of the cipher text becomes large. This adds up to tremendous information storing. It is extremely important to address the storage capacity issue along with the security issues of exponentially developing information. This problem can be resolved by compressing the ciphertext based on a some compression algorithm. In this proposed work used the compression technique called palindrome compression technique. The compression ratio of the proposed method is better than the standard method for both colored and gray scaled images. An experimental result for the proposed methods is better than existing methods for different types of image.

Keywords: DNA Cryptography, Plaindrome, Data Compression, Encryption, Decryption

## Introduction:

The cryptography plays a vital role to provide security in the field of network or any storage media. There are various cryptographic techniques available now a days. Out of which, DNA cryptography is new born field in the field of Cryptography. While encrypting the cipher text using DNA, one will get very long length sequences of ciphertext. In order to provide efficient storage for the ciphertext there is a need to compress the generated DNA sequences[1]. The compression process reduces the amount of transferring data as well as storage space which in turn effects the usage of bandwidth. Reducing the size of data leads to reduction in the transmission time of data in a network. There are two types of data compression techniques available. One is Lossless and Lossy Compression techniques. Examples of Lossless Compression are Runlength Coding, Huffman Coding and LZ77[3]. Examples of Lossy Compression techniques are picture transformation, picture resizing and quantization. This chapter discuss the compression technique which will be used for providing efficient storage for cipher text.

## Related Work:

Amikov proposed compression algorithm based on the tree modelling for color map images. It works with the phenomena of n ary context free model with complex binary tree structures of n color map images. The major thing present in this is it is suitable for the color images only with

72 colors only[5]. Improvement in LZC[7] command of unix operating system proposed a new method in compression algorithm based on the dictionary and it is suitable for textual patterns not for the images.

Cuitex propsed a new compression algorithm with increase in throughput and compression ratio. The design principal of this algorithm is based on the adaptive pre-processor which will reduce the coorelation ratio between the blocks[8]. VLSI architecture is used for reducing compression ratio. Moreover this algorithm have high redundancy of data when compared to other existing algorithms like LZW[9].

Increase the bandwidth speed by reducing the size of the transmitted data on a network. Decrease the size of the data without loss of information is called Lossless Compression. Lossy compression can be done on images[4]. In Lossless Compression, RLE, Huffman Coding, LZW77 and PPM are familiar methods. In RLE, note down the occurrences of same sequence consequently[2].

LZ77 was implemented based on search buffer and look ahead buffer. It is also called as Sliding window Compression. Search buffer is used as dictionary and look ahead buffer is used to find the matches which are search buffer[6].

## Proposed Method:

DESIGN AND IMPLEMENTATION

It is used the compression technique called palindrome compression technique. Palindrome means the original sequence is equal to the reverse of the sequence.

For eg., Consider a sequence S=CGTTTTGC.

The reverse of the sequence is R=CGTTTTGC which is same as the original sequence. The palindrome compression technique includes storing of half of the string instead of storing full length string. In decompression process, the string already stored is reversed and added at the end of the same string to get the final string.

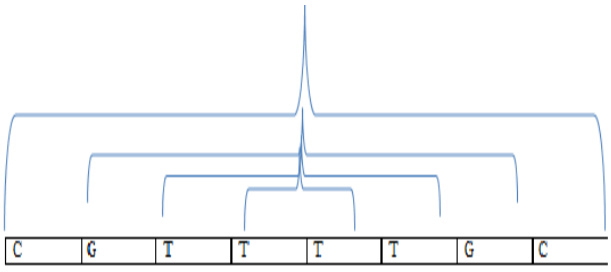


Figure 1.Palindrome Technique

Compression Algorithm:

Algorithm PalinCompress(C, length)

C is the Ciphertext in the form of DNA Sequences and length is used to divide the DNA sequence into number of specified length strings.

BEGIN

1. Split the DNA strand into number of substrings based on the given length
2. Check whether each substring is palindrome or not. If the substring is palindrome then store half of the string.
3. If it is not palindrome, continue with the same string.
4. Repeat step 2 & 3 for all the substrings.
5. Revrse the obtain string and divide into equal parts

6. Repeat step 2&3 for the substrings

END

Decompression Algorithm

Algorithm PalinDeCompress(D)

D is the Decompressed DNA strand.

BEGIN

1. Read the DNA strand until you find the digit.
2. When the digit is found, count the number of characters equal to the digit, that are immediately presented the digit and add the same characters in reverse order from the location where the digit is found that is the digit is replaced by the first character.

END

Empirical Analysis

For example take a DNA Strand

Best Case:

C=AGGTTGGAACGTTGCAAAAAAAAATTGGGGTT and Length=8

Divide the DNA strand into 8 length substrings then

C1=AGGTTGGA

C2=ACGTTGCA

C3=AAAAAAAA

C4=TTGGGGTT

Find each substring is palindrome or not, if yes then store half of the string and the number of leftover characters of the substring. Here, all four strings are palindromes then store half of the strings and the lengths then the compressed string is

C=AGGTACGTAAAATTGG

Compression Ratio=Number of Sequences After Compression/Number of Sequences

Before Compression,

Compression Ratio=16/32=0.5. In this case, the transmitted date rate reduced to half.

C=GGTTAAAATGCATGGA

C1=GGTT

C2=AAAA

C3=TGCA

C4=TGGA

Find each substring is palindrome or not, if yes then store half of the string and the number of leftover characters of the substring.

C=GGTTAATGCATGGA

Compression Ratio=14/32=0.43.

Worst Case:

Consider the Sequence

S=ACGTACGTAAAAAGTAAATTATATTCCTTCCC and take the length as 8

S1=ACGTACGT

S2=AAAAAGTA

S3=AATTATAT

S4= TCCTTCCC

Here no substring is palindrome so compression ratio is 32/32=1. When the compression ratio is one then compression will not be helpful.

Results:

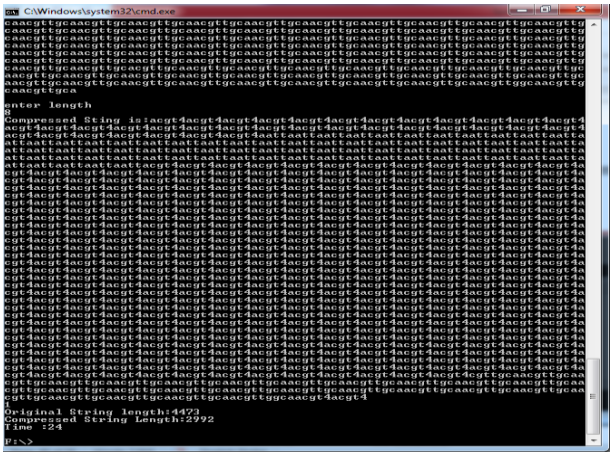


Figure 2 Compression of DNA sequences

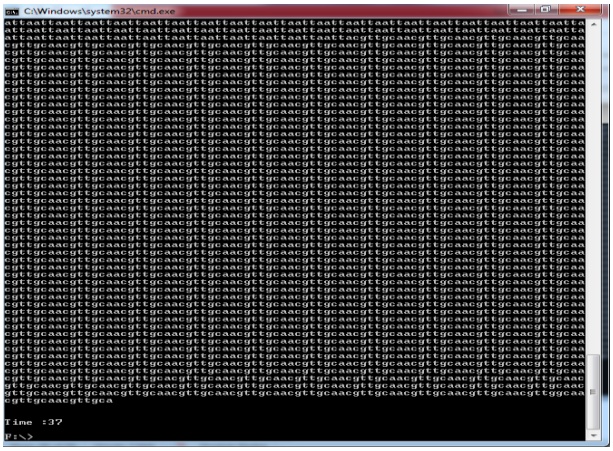


Figure 3 Decompressing DNA sequences

PERFORMANCE ANALYSIS

The analysis of an algorithm measured in Intel i3 processor in jdk1.8.0-131 environment with 4GB RAMS in Window7 Platform for various text documents. The following table showed that the compression and Decompression time taken to compress various DNA Sequences which were stored in various text documents.

Table 1. Performance Measurement for Compression and Decompression of various text documents.

| Text Documents in terms KB | Original DNA Sequence | Compressed DNA Sequence | Compression Ratio | Compression Time(ms) | Decompressior Time(ms) |
|---|---|---|---|---|---|
| 1 | 169 | 109 | 0.644 | 3 | 3 |
| 2 | 417 | 284 | 0.68 | 4 | 4 |
| 3 | 1251 | 1120 | 0.895 | 11 | 10 |
| 4 | 2048 | 1876 | 0.916 | 15 | 17 |
| 5 | 4393 | 2645 | 0.602 | 23 | 22 |
| 6 | 6425 | 4654 | 0.7214 | 41 | 38 |

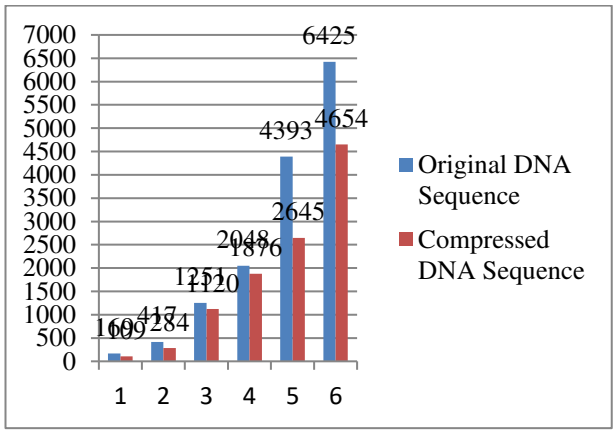


Figure 4 : Comparison of Conversion of Compressed to Original sequences with the time

CONCLUSION

Several methods are available to compress the DNA sequences which are stored in GeneBank. The compression algorithm which is implemented here used the palindrome technique which is meant to reduce the storage space as well as bandwidth while transmitting the data. Thus it gave rise to a lossless compression reducing more than 50% of space. Further it is easy to implement with less computational complexity.

REFERENCES:

1. https://en.wikipedia.org/wiki/Run-length_encoding visited on 12.06.2017.
2. [Ziv, J.1978] Lempel, “Compression of Individual Sequences via Variable-Rate Coding”. IEEE Transactions on Information Theory, Vol.24, Issue 5,pp. 530-536.
3. Xin Chen 1,” A Compression Algorithm for DNA Sequences and Its Applications in Genome Comparison”
4. Nour Bakr,” DNA Lossless Compression Algorithms: Review, Reserch gate Vol 24.
[5] AkimA[1], KolenikovA, FraäntiP.” Lossless compression of color map images by context tree modeling”,IEEE Transction Image Process,2007,Jan;16(1);114-20.
[6] Ammit Makar, 2Gurit Singh, 3Rajneesh Narula,” Improving LZW Compression”, IJCST Vol. 3, Issue 1, Jan. - March 2012.
[7] Jing-Yu Cui,∗ Saurabh Mathur, Michele Covell, Vivek Kwatra, Mei Han,” Example-Based Image Compression”, Google Research, Google Inc., Mountain View CA 94043.
[8] Madhu Sunil Dalal,” Review Paper on Image Compression Using Lossless and Lossy Technique”, International Journal of Advance Research , Ideas and Innovations in Technology, ISSN: 2454-132X Impact factor: 4.295 (Volume3, Issue2).
[9] Pankaj R. Parwe1 , Prof. Nitin N. Mandaogade2,” A Review on Image Compression Techniques”, International Journal of Computer Science and Mobile Computing, Vol. 4, Issue. 2, February2015,pg.198–201