Digital Data Storage on DNA

Raunak Laddha
Smt. Kashibai Navale
College of Engineering,
Pune, India

Kishor Honwadkar
Smt. Kashibai Navale
College of Engineering,
Pune, India

ABSTRACT
Digital data has changed the use and access of information. Everyday lot of data is produced and this requires high-density storage devices which can retain values for a long time[1]. Deoxyribonucleic acid (DNA) can be potentially used for these purposes as it is not much different from the conventional method used in a computer. DNA can be used as a robust and high-density storage device even under unfavourable conditions[2]. Theoretically, one can encode 2 bits per nucleotide in DNA which can store 455 exabytes per gram maximum data in single-stranded DNA (ssDNA)[3]. In this paper, the method described can be used to store text data in DNA by compressing, storing multiple copies along with providing security to data.

General Terms
Storage method and compression.

Keywords
DNA, data storage, encoding, compression, storage mechanism, digital data, secured storage

1. INTRODUCTION
The demand for data storage devices is increasing day by day as more and more data is generated every day. Total information in digital format in the year 2012 was about 2.7 zettabytes[4]. Presently devices such as optical discs, portable hard drives, and flash drives are used to store data. But silicon and the other non-biodegradable materials used in data storage pollute the environment. Also, they are available in limited quantities. Thus, they would be exhausted one day. The linear density of digital storage device is 10 kb per square mm[5]. Hence, newer technology is needed for data storage and archival process. As the data increases, the current data storage technology would not be enough to store data in future as data is growing every day. Even potentially important information can get lost due lack of storage space.

One of the most common cause of data loss is accidental deletion of files without backup. Every day many people lose important data because of deleting files accidently because they do not have proper backup systems. Poor handling of the optical disk can cause data loss in them. Data loss can occur due to damage of hard drive. Mechanical damage to the hard drive is common as it contains a lot of fragile parts moving at very high speed. Hard drives can get damaged due to accidental drop of computers. Hard drives can get damaged if any liquid enters it. Liquids can cause damage to electronic parts of drive making it difficult to recover data. The hard disk can get damaged due to fire. Solid State Drive has a limited number of write cycles. Thus after write cycle limit, it is not possible to write data on them. A printed book has better life expectancy than best of the data storage method.

There are many ways to backup the data. One can use cloud services to store data. But to access data which is stored in a remote cloud, an internet connection is needed all the time. Thus without an internet connection, it is not possible to access the data which is stored in the cloud. Another way is to store data on an external drive. But external drives are prone data loss too.

Scientists and Researchers for over the past decade, have been trying to develop a robust way of storing data on a medium which is dense, robust and ever-lasting. They are sticking to the storage medium which is used by nature that is DNA. There are many reasons to use DNA as the storage medium such as small size and high density. Just 1 gram of dry DNA can store about 455 exabytes of data[3]. Thus, data on DNA can be conveniently stored.

The power usage required while working with DNA is a very little compared to a conventional storage[5]. Even the error rate of DNA storage is much less than normal storage device[5]. DNA is a very robust material and it has a long shelf life. The information stored in DNA can be recovered even after thousands of years[6][7]. As long as the DNA is stored in dry, dark and cold conditions, DNA can be stored for a long time. By using Polymerase Chain Reaction techniques[8], it is possible to get as many copies as required. Thus, copying of data can be done easily and many copies of data can be obtained.

As DNA can retain information for centuries, DNA can be used for long-term storage[2]. Due to high density, the DNA can store a large amount of data in very small space[3]. As in approach ref. 3 and 9, the data is stored in long virtual DNA molecule but encoding is done using synthetically prepared short DNA strand. Short strands will allow to easily manipulate data[9]. It is possible to read simultaneously and randomly read files stored in DNA. Also, compression technique is used to compress data without any loss. The 4 nucleotides of DNA used in the model are Adenine which will be denoted as A, Cytosine as C, Guanine as G and Thymine as T.

2. RELATED WORK
Various methods and techniques have been used for storing data in DNA. Storing data in DNA demonstrated in ref. 3 shows that they read data in binary form and encoded data by using A, C in place of 0 and G, T in place of 1 randomly choosing bases while avoiding 3 or more nucleotide repeats and balancing GC content. In this, 19-bit long stream is used for addressing. By this technique, it is possible to store 5.27e6 bits per cubic millimeter. This technique is very simple and easy to implement. But they have not used any compression technique to reduce the size of the data. Hence, a lot of space is required to store data comparatively.

In model demonstrated as in ref. 9, the data is converted to ASCII format and then encoded using Huffman code given and converted to base-3 format. The index is calculated using length, file ID and parity. This information is converted to nucleotide format using defined table and avoids nucleotide repeats. They synthesized DNA of the data and recovered it

100%. Although they have used good compression technique, the data cannot be read randomly easily. A lot of computations are required to encode and decode data using this method.

Another way to store data in DNA is demonstrated in ref. 10. In this, each letter on the keyboard is mapped to a combination of 4 nucleotides. This method allows storing data in 50% space than a conventional digital storage system. Even though this method is highly optimized, it only considers letters on the keyboard. Other characters are not taken into consideration. There is no indexing in this method. Hence, random access of information is not possible. Thus, one has to scan whole DNA to find information.

3. PROPOSED SYSTEM

In the model proposed in this paper, ssDNA is used to store data. In this, a delimiter is used at the end of each file so that data can be accessed randomly. The data will be encoded using specialized Huffman tree[11][12][13]. If required, each file can be given separate Huffman tree for encoding which will increase data security along with compressing the data. In the case of any error in data while encoding, this error is contained in that file only. As Huffman tree is used for encoding, data compression is achieved. It provides security as anyone cannot decode it without the original tree. For sequencing of DNA strand, a lot of specialized equipment are needed. So without the equipment required, DNA cannot be read. Maximum of 2 nucleotide repeats, except for delimiters, are there in DNA. There are 2 copies of all the data. So in the case of data loss, other copy can be used to retrieve data. This method is flexible and the user can manipulate the method to suit the needs and store all kind of data.

3.1 Encoding

1. Form frequency table of characters of the data.
2. Now Huffman tree of non-repeating nucleotides for encoding is generated as follows:
    a. Each node in the tree will have 3 children.
    b. The weights of branches of children will depend on the incoming weight of parent.
    c. If the weight of incoming branch of a parent is A, then C represents the leftmost child, G represents the middle child and T represents the rightmost child.
    d. If the weight of incoming branch of a parent is C, then G represents the leftmost child, T represents the middle child and A represents the rightmost child.
    e. If the weight of incoming branch of a parent is G, then T represents the leftmost child, A represents the middle child and C represents the rightmost child.
    f. If the weight of incoming branch of a parent is T, then A represents the leftmost child, C represents the middle child and G represents the rightmost child.
    g. T will be considered to be an incoming weight for root.
3. Now split the whole data into overlapping segments of 100 nucleotides with an offset of 50 nucleotides from previous.
4. Form pairs of segments starting from the 1st segment.
5. Index each pair from 0 to 107 and after 107, start from 0 again.
6. Reverse complement 2nd segment in each pair.
7. The index will be of 4 nucleotides long. The index is encoded by a combination of nucleotides in a sequence of A, C, G, T such that no 2 consecutive nucleotides same. Example: 0=ACAC, 1=ACAG, 2=ACAT.
8. Prepend A and append C to the 1st segment of the pair.
9. Prepend T and append G to the 2nd segment of the pair.
10. Each segment is now synthesized to actual DNA strand of length 106 nucleotides.

If the length of the code of a character is 1, then to avoid repetition of nucleotides, 1 more nucleotide is added in the code of the character.

3.2 Decoding

1. The decoding process is simply the reverse of the encoding process.
2. The 1st nucleotide of DNA will tell whether the DNA is the 1st or 2nd segment of the pair, whether the data is reverse complemented or not and directionality of strand.
3. If 1st nucleotide is A then:
    a. Remove 1st nucleotide.
    b. Next 4 nucleotides will tell us about segment number.
    c. Next 100 nucleotides will be data.
    d. The last nucleotide can be used for confirmation of the type of segment.
4. If 1st nucleotide is C then:
    a. Reverse whole segment.
    b. Remove 1st nucleotide.
    c. Next 4 nucleotides will tell us about segment number.
    d. Next 100 nucleotides will be data.
    e. The last nucleotide can be used for confirmation of the type of segment.
5. If 1st nucleotide is G then:
    a. Reverse whole segment.
    b. Remove 1st nucleotide.
    c. Next 4 nucleotides will tell us about segment number.
    d. Reverse complement next 100 nucleotides.
    e. These 100 nucleotides will now be data.
    f. The last nucleotide can be used for confirmation of the type of segment.
6. If 1st nucleotide is T then:
    a. Remove 1st nucleotide.
    b. Next 4 nucleotides will tell us about segment number.
    c. Reverse compliment next 100 nucleotides.
    d. These 100 nucleotides will now be data.

e. The last nucleotide can be used for confirmation of the type of segment.

7. If TTTT sequence is found, this will denote the end of the file. The new character will start from next nucleotide.

8. Now by using the same Huffman tree, data can convert the data into original characters.

It is possible to generate different Huffman tree for different files or single Huffman tree for whole data. This will compress the data and decoding cannot be done unless one has the original tree. As specific orientation nucleotides have been used in the strands, it is possible to read double number segments in the same number of indexes. The user can read the strand from any direction.

4. POSITIONS AND ARRANGEMENT OF DNA STRANDS

The fig. 1 tells us about how the DNA strands will be arranged. The green part is orientation nucleotide which will tell about the direction or orientation of DNA. The orange part is the data or information encoded which is not reverse complemented. The yellow part is the data which is reverse complemented. The blue part is the index of the strand. 2 strands of DNA always run in opposite direction and the pairs that can form a bond are specific[12]. A can a form with T and C can form a bond with G. Due to these conditions, alternate strands were reverse complemented.

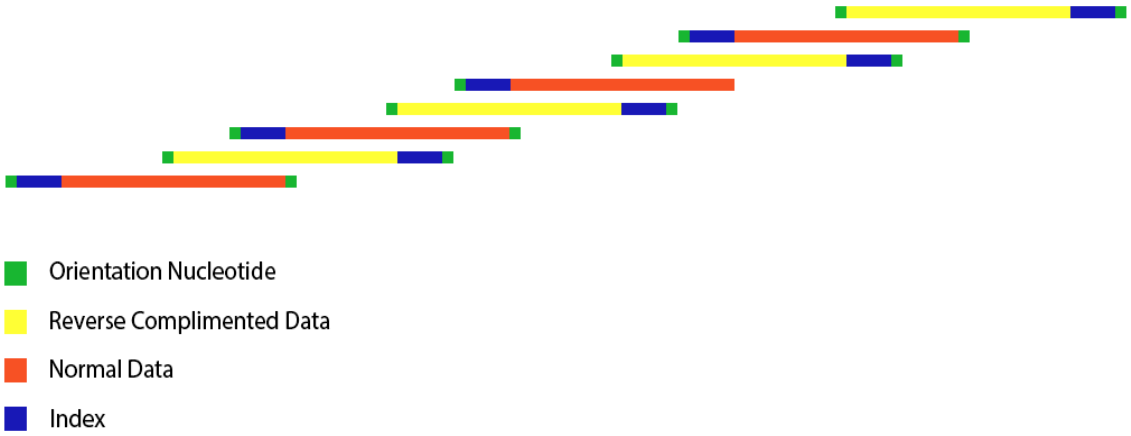

Figure 1: Position and Arrangement of DNA strands

5. COMPARISON

Table 1[15] shows a comparison of the digital storage device and DNA storage.

Table 1: Comparison [15]

Parameter	Digital Storage Device	DNA Storage
Basic Element	Transistor, Magnetic or Optical Domain	Nucleotide
Basic Element Size	1,000 / 10,000 nm^2	1 nm^3
Addressing Scheme and Organization	Discrete, Hierarchical	Sequential, Parallel Copying.
Redundancy	Multi-Layer	Data Replication
Capacity	Unlimited	0.1 GB / genome
Linear Density	10 kilobits / mm^2	1 Mbit / mm
Areal Density	1 gigabit / mm^2	-
Access Time	0.01 millisecond	-
Internal Data Transfer	10 megabits / sec	100 kilobits / sec
Error Rate	10^{+4}	10^{-9}

6. RESULTS AND DISCUSSION

Table 2: Size of files before and after encoding

File name	Size in bits	Length after encoding	Total nucleotides required
Watsoncrick.pdf (from website http://www.nature.com/nature/dna50/archive.html)	2943008	1001699	2123604
pic.png (author's own picture)	930360	305289	647236
Dummy.txt (Text document with only alphabets)	8192	3162	6784
IJCA paper template.docx (from website http://www.ijcaonline.org/)	189944	63070	133772
MLK_excerpt_VBR_45-85.mp3[9]	1348312	454124	962904

Table 2 shows the size of the file and total nucleotides that will be required to store the data of file using the encoding

process. The file name column gives information about the file, size in bits column is size of the file in bits, length after encoding column tells about the length of data after encoding, and total nucleotides column tells about the number of nucleotides required to store the data in the required format along with a copy of data with is reverse complemented. All the data of these files can be stored in a single system by using delimiter after each file. If the length of data is increased in DNA strand, fewer nucleotides will be required to store the data. The length of data on DNA strand must be multiple of the number of copies to be stored.

As there is a repetition of more than 2 nucleotides only in delimiter, it is easy to determine the delimiter from all DNA strands randomly and it is not necessary to rely on data from the previous file. This allows the user to read data randomly. Even if 2 files may share data on a single strand, both files can be read simultaneously as there are 2 copies of all data.

Considering the data which is not reverse complemented, as no character code starts with T, if any code found, the error can be detected easily. As there is only T repeated for more than 2 times, if any other nucleotide repeated for more than 2 times, the error can be detected immediately in the DNA strand which is not reverse complemented. 2nd copy of each data can be used in case of error. The delimiter can be changed to AAAA, CCCC, and GGGG but one must change the root of Huffman tree to A, C, and G respectively so that after delimiter, the data doesn't start with the same nucleotide. Even length of delimiter can be changed. If the size of files is large, different Huffman tree can be generated for each file. This will provide security to each file separately.

The number of copies can be increased by changing the offset value. The length of data in each segment can be increased by increasing the size of DNA strand. Index size can be increased or decreased to fit the need of user but the method to create index should not change to avoid repetition. In this, as there are more outgoing branches in Huffman tree compared to Huffman tree used for binary data, the data is relatively more compressed.

This method can be used to store other types of files too such as multimedia files, document files, etc. To convert other types of files, the files need to be read in hex, binary, etc. types depending on needs. For hex type, by forming pairs of hex numbers, the compression technique can be applied easily. For binary, data need to be read as byte and then the encoding method can be applied.

7. CONCLUSION

Thus, using DNA for data storage, it is possible to store huge amount of data in very less size. As DNA can retain data for millions of years, it is possible to store data for a long time. By using this technique, data is compressed and the security to the data is provided. Parallel reading of files is also possible enabling users to read multiple files at the same time. This technique maintains two copies of data. Hence in case of data damage, its copy can be used to read data. In the case of any errors while encoding the data, the error is restricted to that particular file and no other file is affected due to that error. This technique can be used for all kind of files by making minor changes to adapt to the type of file. This technique can be used to store big data in very small space with little computational overhead. This method is scalable and can be used to store large files too. Also multiple copies can be made easily. This method can be used to store information in archival systems or big data. Instead of using conventional storage devices which have less capacity to store data, DNA-based storage method be used in distant future to store data secured manner and for long time storage and solve the problem of limited space.

8. REFERENCES

[1] J. Gantz, D. Reinsel. Extracting value from chaos. International Data Corporation (IDC), Framingham, MA (2011), www.emc.com/collateral/analyst-reports/idc-extracting-value-fromchaos-ar.pdf.

[2] C. Bancroft, T. Bowler, B. Bloom, C. T. Clelland. Long-Term Storage of Information in DNA. Science, 293, 1763 (2001).

[3] George M. Church, Yuan Gao, Sriram Kosuri. Next-Generation Digital Information Storage in DNA. Science, 337, 1628 (2012).

[4] Siddhant Shrivastava and Rohan Badlani. Data Storage in DNA. International Journal of Electrical Energy, Vol. 2, No. 2, June 2014.

[5] Mohan S., Vinodh S. and Jeevan F. R. Preventing Data Loss by Storing Information in Bacterial DNA. International Journal of Computer Applications (0975 8887) Volume 69 No.19, May 2013.

[6] Green, R. E. et al. A draft sequence of the Neandertal genome. Science 328, 710722 (2010).

[7] Willerslev, E. et al. Ancient biomolecules from deep ice cores reveal a forested southern Greenland. Science 317, 111114 (2007).

[8] Lilit Garibyan and Nidhi Avashia. Polymerase Chain Reaction. Journal of Investigative Dermatology (2013) 133, e6. doi:10.1038/jid.2013.1

[9] Nick Goldman, Paul Bertone1, Siyuan Chen, Christophe Dessimoz, Emily M. LeProust, Botond Sipos & Ewan Birney. Towards practical, high-capacity, low-maintenance information storage in synthesized DNA. Nature, 494, 7780 (2013).

[10] Salunke Avinash N., Shruti Gupta, Varsha Agarwal, and Muhammad Rukunuddin Ghalib. A Novel Digital Information Data Storage Approach in DNA. International Journal of Applied Engineering Research, ISSN 0973-4562, Vol. 8, No. 19 (2013).

[11] Mamta Sharma. Compression Using Huffman Coding. IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.5, May 2010.

[12] Nigam Sangwan. Text Encryption with Huffman Compression. International Journal of Computer Applications (0975 8887), Volume 54 No.6, September 2012.

[13] Ailenberg, M. & Rotstein, O. D. An improved Huffman coding method for archiving text, images, and music characters in DNA. Biotechniques 47, 747754(2009).

[14] Watson, J. D., & Crick, F. H. C. A structure for deoxyribose nucleic acid. Nature 171, 737–738 (1953).

[15] Mohan S, Vinodh S and Jeevan F R. Preventing Data Loss by Storing Information in Bacterial DNA. International Journal of Computer Applications 69(19):53-57, May 2013.