DNA Sequence Compression Technique Based on Nucleotides Occurrence

Bacem Saada, *Member, IAENG*, Jing Zhang

***Abstract*— Nowadays, The development of genome sequencing technologies has become more accessible for the biologists who are now facing a huge amount of genomic data. Storing and analyzing tis huge amount of data is a big challenge for scientists. For that reason, it is suitable to implement compression algorithms for genomic data. These algorithms attempt to reduce the DNA information before storing it in databases. In this paper, we will introduce a two phases compression algorithm based on the binary representation of DNA sequences. In the first phase, we will a new technique to compress the DNA sequence and convert it into binary representation. Thereafter, we will compress the resulting DNA using the Extended-ASCII encoding through which one character represent four nucleotides. The remarkable compression ratio of our algorithm makes its use interesting.**

***Index Terms*— DNA compression, Extended-ASCII code, Genomics, Horizontal compression;**

I. INTRODUCTION

The development of second and the third generation sequencing technologies has led the scientists to sequence huge genomes containing millions and billions nucleotide bases. In 2000 the first human genome was sequenced and it contains three billion nucleotide bases [1]. This project costed three billion dollars. In the last few years, the human whole-genome sequencing cost has dropped to less than $1500 [2]. In addition, many projects are sequencing whole genomes of many species to analyze its coding zone and understand the functions of its proteins and the evolution process of the specie regarding its ancestors.

Nowadays, powerful computers are used to properly analyze and store this genomic data. Consequently, two problems have arisen. First, the encoding of the data and second, the time required to process them. As a result, to reduce data sizes, many data compression techniques have been implemented. Lossless compressors that compress data without any loss of its information are used for text compression techniques and thus for DNA sequences chain.

Technological progress has led to the birth of bioinformatics research area which processes and analyzes different living beings' data. The essential element in achieving these treatments is the Deoxyribonucleic Acid or DNA, which is a biomolecule present in all cells. This biomolecule contains the genetic information required for the functioning and development of all living beings. Each monomer constituting it is a nucleotide, which is composed of a nitrogenous base; adenine (A), cytosine (C), guanine (G) or thymine (T). Many online databases are accessible to store and share the genomic information. GenBank, managed by the International Nucleotide Sequence Database Collaboration, is a free access database that contains a large amount of genomic data which can be analyzed. This huge quantity of genomic information, led us to propose DNA sequences compression algorithms that reduce the size and so thoroughly analyze and choose the data that will be stored as the databases may contain redundant information.

In this article, we will start with a review of existing DNA sequences compression algorithms (Section II). In section III, we will present our approach for DNA sequences compression and explain how it can reduce their sizes. Finally, in section IV, we will illustrate the experimental results and we will draw a comparison of ratio between our algorithm and other existing algorithms.

II. EXISTING DNA SEQUENCES COMPRESSION ALGORITHMS

The compression of DNA sequences is based on text compression algorithms. However, researchers proved that conventional text compression algorithms are not enough for DNA sequences compression as it is composed by only four nucleotides and proposed specific compression algorithms. Based on the standard benchmark of DNA sequences data [3] GZIP tool [4] for example has a compression ratio of 2.217 Bit per Base. However, a compression algorithm provides significant results only if the BpB is lower than two because only then the four nucleotides based on binary representation [5] can be represented by two bits.

There are two major classes of DNA sequences compression algorithms. The algorithms for DNA compression in horizontal mode and the algorithms for DNA Compression in vertical mode. The first class compresses a single sequence based on its genetic information. For example, Biocompress [6] compresses the repetitions and palindromes in a sequence. BIocompress-2[7] uses a Markov model to compress non-repetitive regions of a sequence. By applying these algorithms, the compression ratio is 1.85 BpB for Biocompress and 1.78 BPB for biocompress-2. Therefore, they are better than conventional Lossless compression algorithms since the BpB rate is under two BpB.

Some other DNA sequences compression techniques are based on the binary representation of the nucleotides (e.g. A = 00, C = 01, G = 10, T = 11). GENBIT [8] divides sequences in blocks of 8 bits each and makes a 9th bit. If the block is identical to the above, the 9th bit is equal to 1, otherwise to 0. DNABIT [9] divides the sequence into small blocks and compresses them while taking into consideration if they existed previously or not. Saada, B. and Zhang, J

Manuscript received October 29, 2017; revised November 21, 2017.

Bacem Saada, Ph.D. Student with Harbin Engineering University, College of Computer Science and Technology, Harbin, China, (email:basssoum@gmail.com).

Jing Zhang, Ph.D. Professor with Harbin Engineering University, College of Computer Science and Technology, Harbin, China, (email: zhangjing@hrbeu.edu.cn).

reduced the size of the DNA sequence to less than 25% of its initial size by compressing it using the extended-ASCII representation and applying the RLE technique to compress the similar blocks and keep only one block [10]. They also proposed an algorithm that compressed a DNA sequence with a compression ratio equal to 1.6 BpB[11].

The second major class of DNA sequences compression algorithms analyzes the genetic information of a set of sequences in order that one of them would be representative of the whole set. For exemple, DNAZIP package [12] has a series of algorithms that divide a genome into small blocks and compress them. LZ77 [13] proposes a compression technique for several genomes belonging to the same genus as their genomes contains many common regions. Saada, B. and Zhang, J. use some techniques to convert the DNA sequence to an hexadecimal representation and detect regions of similarities between a set of sequences [14]. They also proposed an algorithm to detect the longest common chain for a set of sequences species belonging to the same genus and use it as representative of the whole set [15].

III. Our proposed algorithm

A. *Description of the algorithm*

Our algorithm is based on the binary representation of nucleotides. First, our algorithm records the total count of each nucleotide in the DNA sequence and compress the DNA sequence based on the frequency of each nucleotide. Thereafter, to reduce the size of the output sequence, the bits will be converted to Extended ASCII coding which one character encode 8-bits.

B. *Presentation of the algorithm*

1. *Counting the nucleotide frequencies*

The algorithm counts the frequency of each nucleotide.

2. *Classification of the nucleotides based on their frequencies*

In this phase, we classify the four nucleotides and we assign the and we assign them as first, second, third and fourth depending on their frequencies.

Example:

Sequence : AACTAAACGTTT

The frequencies are as fellows

Table. 1. Vocabulary table content

Nucleotide	Frequency	Assigned value
A	5	First
T	4	Second
C	2	Third
G	1	Fourth

3. *Compression phase*

In this step, we assign 0 to all the first occurrences. In addition we assign a 0 also to all the occurrences of the second occurrence and save its position in another data structure.

Regarding the third and the fourth nucleotides, we assign to them the value 1 and we save the positions of the fourth nucleotide in another data structure.

For the initial sequence, this step can be illustrated as follows:

Sequence: AACTAAACGTTT

Output: 001000011000

Data structure for the second occurrence:

4 10 11 12

Data structure for the fourth occurrence:

9

The choice to assign the value 0 to the two highest nucleotides is because the more data is represented by the bit 0 the more its size while stored on auxiliary memory is the lowest.

This data will be compressed by zip compressor as it can reduce text file to 35% of its initial size

4. *Compression of the binary output to extended-ASCII representation*

To better reduce the size of the data stored in the databases, we will convert the binary representation to an extended-ASCII representation. The benefit from the use of this technique is that one extended-ASCII character encodes 8 binary digits. The output result will be reduced to 12.5% of the initial binary representation (figure 1.).

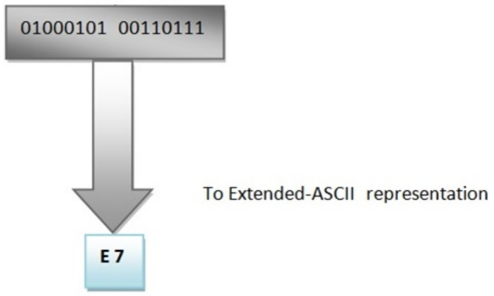

Fig. 1. Conversion to extended-ASCII representation

C. *Decoding Phase*

For the decode phase, the algorithm decompresses the zip file containing the two data structures and the modified binary representation of the DNA sequence.

For each nucleotide equivalent to 1 and represented on the data structure, we change its corresponding positions in the DNA sequence to the fourth nucleotide.

The other nucleotides represented by 1, we change them to the third nucleotide in the DNA sequence.

Regarding the nucleotide represented by 0 and having corresponding positions in the additional data structure, we replace them by the second nucleotide.

The other nucleotides, we replace them by the first nucleotide.

IV EXPERIMENTAL RESULTS

A. *Evaluation Metrics*

To measure the performance of our approach, we use entire genomes in order to calculate the performance of our approach, in terms of compression ratio, to the genomes which have a large number of nucleotides.

B. *Performance in terms of data compression*

To achieve our experimental study, we used the Human Globin Gene (HUMHBB), the Human Sequence of Contig (HUMHDABCD) the Mitochondrial genome (MPOMTCG) and the Vaccinia Virus genome (VACCG) whose size is about 190000 nucleotides.

As indicated in table II, applying our approach helped to reduce the compression ratio of those genomes. The results demonstrate that most of the existing algorithms have a compression ratio higher than 1.6 BpB. Our algorithm provides better results and has a compression ratio equal to 1.41 BpB for the compression of the genome MPOMTCG.

TABLE II. Comparison with other algorithms

Sequence	DNA Compress	DNABIT -2	Our Approach
HUMHBB	1.79	1.6	1.42
HUMHDABCD	1.796	1.6	1.46
MPOMTCG	1.90	1.54	1.41
VACCG	1.75	1.63	1.44

TABLE III. Performance of our algorithm

Sequence Name	Size after applying the compression techniques	Size after the Extended-ASCII Compression
HUMHBB	52048	6507
HUMHDABCD	40318	5040
MPOMTCG	132346	16544
VACCG	133150	16644

C. *Experiments in Time execution*

To measure the execution time of our algorithm, we used a computer with an Intel i3-2375M processor cadenced at 1.5 Ghz and a 4GB Ram memory.

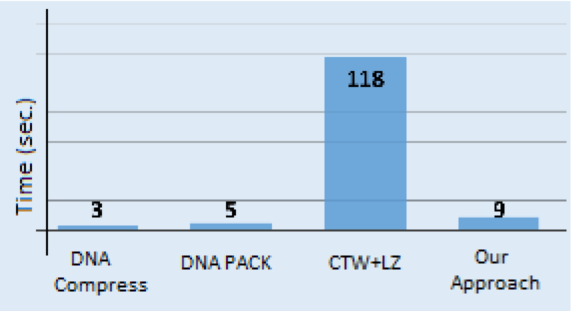

Fig. 2. Execution time comparison between our approach and other algorithms

The previous figure (fig.2) presents the execution time by applying the algorithm on the MPOMTCG genome. It shows that its execution time is less than that of CTW+LZ algorithm and slightly higher than DNA Pack and DNA Compress execution time.

V. CONCLUSION AND FUTURE WORK

The advantage of our technique is that it allows to have a compression ratio lower than 1.5 BpB thus better than other existing compression techniques. The algorithm is also easy to implement and interesting as it uses the Extended-ASCII representation compresses the initial nucleotide representation to less than 10% of its initial representation.

In the future, we will try to associate our algorithm to compression algorithms based Markov models to predict the frequency of each nucleotide depending on the region of the genome (GC region, AT region, etc.) and to better compress the DNA sequences with a rate higher than the rate of the current existing techniques.

ACKNOWLEDGMENT

This paper is funded by the International Exchange Program of Harbin Engineering University for Innovation-oriented Talents Cultivation.

REFERENCES

[1] Lander, E.S., Linton, L.M., Birren, B., Nusbaum, C., Zody, M.C., Baldwin, J., et al.: Initial sequencing and analysis of the human genome. Nature 409, 860–921 (2001)

[2] Caporaso, J. G., Lauber, C. L., Walters, W. A., Berg-Lyons, D., Huntley, J., Fierer, N., ... & Gormley, N. (2012). Ultra-high-throughput microbial community analysis on the Illumina HiSeq and MiSeq platforms. The ISME journal, 6(8), 1621-1624.

[3] S. G rumbach and F. Tahi, "Compression of DNA Sequences," in Proc. of the Data Compression Conf., (DCC '93), 1993, 340–350.
Pierzchala, S. (2004). Compressing Web Content with mod—gzip and
[4] mod—deflate. Linux Journal, 1-10.

[5] Matsumoto, T., Sadakane, K., Imai, H., et al., 2000, Can General-Purpose Compression Schemes Really Compress DNA Sequences?, Computational Molecular Biology, Universal Academy Press, 76–77.

[6] Grumbach S. and Tahi F.: Compression of DNA Sequences. In Data compression conference, pp 340-350. IEEE Computer Society Press, 1993.

[7] Korodi, G., Tabus, I., Rissanen, J., et al., 2007, DNA Sequence Compression Based on the normalized maximum likelihood model, Signal Processing Magazine, IEEE, 24(1), 47-53.

[8] Grumbach, S., Tahi, F.: A new Challenge for compression algorithms: genetic sequences. Journal of Information Processing and Management 30, 866–875 (1994).

[9] A.AppaRao, "DNABIT compress-compression of DNA sequences," in Proc. the Bio medical Informatics, 2011.

[10] Saada, B., & Zhang, J. (2015). DNA Sequences Compression Algorithm Based on Extended-ASCII Representation. In Proceedings of the World Congress on Engineering and Computer Science (Vol. 2).

[11] Saada, B., & Zhang, J. (2016). DNA Sequences Compression Techniques Based on Modified DNABIT Algorithm. In Proceedings of the World Congress on Engineering (Vol. 1).

[12] Ahmed, S., Brickner, D. G., Light, W. H., Cajigas, I., McDonough, M., Froyshteter, A. B., ... & Brickner, J. H. (2010). DNA zip codes control an ancient mechanism for gene targeting to the nuclear periphery. Nature cell biology, 12(2), 111-118.

[13] Ahmed, S., Brickner, D. G., Light, W. H., Cajigas, I., McDonough, M., Froyshteter, A. B., ... & Brickner, J. H. (2010). DNA zip codes control an ancient mechanism for gene targeting to the nuclear periphery. Nature cell biology, 12(2), 111-118.

[14] Saada, B., & Zhang, J. (2015, November). DNA sequences compression algorithms based on the two bits codation method. In Bioinformatics and Biomedicine (BIBM), 2015 IEEE International Conference on (pp. 1684-1686). IEEE.

[15] Bacem Saada, and Jing Zhang, "Vertical DNA Sequences Compression Algorithm Based on Hexadecimal Representation," Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering and Computer Science 2015, 21-23 October, 2015, San Francisco, USA, pp570-574.