Volume 4, Issue 7, July 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

# Developing an approach for DNA packing through dReaM: A Novel Tool

**V. Hari Prasad**
Research Scholar in CSE
Jawaharlal Nehru Technological University (JNTUK),
Andhra Pradesh, India

**Dr. P. V. Kumar**
Professor of CSE
Osmania University Hyderabad,
Andhra Pradesh, India

***Abstract*— Vacuuming will arise when the data base reaches to its threshold. Data can't be expunged, just it can supersede. So desideratum of compression will evade vacuuming in database? Due to the extortionate storage of DNA sequences in a quotidian routine, database becoming plenary. Hence compaction of DNA sequence will solves the quandary of vacuuming. In this connection many classical algorithms are proposed but failed due to the encoded designation of 'text' in a DNA and in integration to that some more algorithms are carried out but the results are not that much bountiful. Our proposed implement dReaM(DNA repetitive encoding analysis methodology) will achieve higher compression rate when compared to the subsisting algorithms. Sure, our technique will be an invaluable loss less implement in bio-informatics era.***

***Keywords*— encoding; decoding; bio compress; Huff bit compress; dnabit compress; LSBD compression**

## I. Introduction

Bio informatics is one of the emerging fields in computer science includes processing and maintenance of biological databases. This is the one of the active area of research which will more helpful in different areas of specialty, including (but in no means limited to) statistics, computer science, physics, biochemistry, genetics, molecular biology and mathematics Computational Biology is the mathematical and algorithmic study of bio informatics allied areas like DNA computing, protein docking and visualization protein information etc.Bio informatics and computational biology are two multidisciplinary fields typically refers to the field concerned with the collection and storage of biological information, where as computational biology refers to the aspect of developing algorithms and statistical models necessary to analyze biological data through the aid of computers.

Defining the terms bioinformatics and computational biology is not necessarily an easy task, as evidenced by multiple definitions available over the web. A recent goggle search for "definition of bioinformatics" returned over 35,000 results! In the past few years, as the areas have grown, a greater confusion into these two terms has prevailed. For some, the terms bioinformatics and computational biology have become completely interchangeable terms, while for others, there is a great distinction. I'll throw my two cents in, based on what my experience has been to the consensus use of these two terms.

In this respect, my understanding of bioinformatics and computational biology follows the

**Bioinformatics:** Research, development, or application of computational tools and approaches for expanding the use of biological, medical, behavioral or health data, including those to acquire, store, organize, archive, analyze, or visualize such numbers.

**Computational Biology:** The development and application of data-analytical and theoretical methods, mathematical modeling and computational simulation techniques to the study of biological, behavioral, and social systems.

## II. Motivation

life is strongly associated with organization and structure [1].With the completion of 1000 genomes project, the project is estimated to generate about 8.2 billion bases per day, with the total sequence to exceed 6 trillion Nucleotide bases. The DNA molecule is made up of a concatenation of four different kinds of nucleotides namely: Adenine, Thymine, cytosine and Guanine (A,T,C,G).Today, more and more DNA sequences are available, due to the excessive surge of genomes storage databases size is two or three times bigger annually. Thus, it becomes very hard to download and process the data in intra and internetworking systems. To maintain it compression is came into the existence .compression can performed in two ways either Loss or Loss- less. Lossy compression is applicable for images because if we remove unnecessary pixels also image doesn't violates its property. But sequences like DNA and RNA encoded information in textual format. So Lossy compression is not advisable to compress such sequences. Text compression is always Loss-less because we have to retain its original property after decoding.

Universal compression algorithms are fails to compress genetic sequences due to specificity of 'text'. Some standard algorithms are worked on it and achieved negative compression rates. General purpose compression algorithms do not perform well with biological sequences. Giancarlo *et al*. **[2]** have provided a review of compression algorithms designed for biological sequences. Finding the characteristics and comparing Genomes is a major task (Koonin 1999**[3]**; Wooley 1999**[4]**). In mathematical point of view, compression implies understanding and comprehension (Li and Vitanyi 1998) **[5]**. Compression is a great tool for Genome comparison and for studying various properties of Genomes. DNA sequences, which encode life should be compressible. It is well known that DNA sequences in higher eukaryotes contain many tandem repeats, and essentials genes (like rRNAs) have many copies. It is also proved that genes duplicate themselves sometimes for evolutionary purposes. All these facts conclude that DNA sequences should be compressible. The compression of DNA sequences is not an easy task. (Grumback and Tahi 1994**[6]**, Rivals *et al*. 1995 **[7]**; Chen *et al*. 2000 **[8]**) DNA sequences consists of only four nucleotides bases {a,c,g,t}. Two bits are enough to store each base. The standard compression software's such as "compress", "gzip", "bzip2", "winzip" expanded the DNA genome file more than compressing it.

Most of the Existing software tools worked well for English text compression (Bell *et al*. 1990**[9]**) but not for DNA Genomes. There are many text compression algorithms available having quite a good compression ratio. But they have not been proved well for compressing DNA sequences as the algorithm does not incorporate the characteristics of DNA sequences even though DNA sequences can be represented in simple text form. DNA sequences are comprised of just four different bases labeled A, T, C, and G (for adenine, thymine, cytosine, and guanine respectively). T pairs with A, and G pairs with C. Each base can be represented in computer code by a two character binary digit, two bits in other words, A (00), C (01), G (10), and T (11). At first glance, one might imagine that this is the most efficient way to store DNA sequences. Like the binary alphabet {0, 1} used in computers, the four-letter alphabet of DNA {A, T,C, G} can encode messages of arbitrary complexity when encoded into long sequences.

## II. BASIC KNOWLEDGE OF GENOME DATA

The consummate set of genetic information for a cell is referred to as its genome. Technically, this includes plasmids as well as the chromosome; however, the term genome is often used interchangeably with chromosome. The genome of all cells is composed of DNA, but some viruses have an RNA genome.

### 3.1 DNA Characteristics

A single strand of DNA is composed of a series of deoxyribonucleotide subunits, more commonly called nucleotides. These are joined in a chain by a covalent bond between the 5„PO4 (5 prime phosphate) group of one nucleotide and the 3„OH (3 prime hydroxyl) group of the next. Note that the designations 5„ and 3„ refer to the numbered carbon atoms of the pentose sugar of the nucleotide (optically discern figure 2.22). Joining of the nucleotides in this manner engenders a series of alternating sugar and phosphate moieties, called the sugar-phosphate backbone. Connected to each sugar is one of the nitrogenous bases, an adenine (A),thymine (T), guanine (G), or cytosine (C). Because of the chemical structure of the nucleotides and how they are joined, a single strand of DNA will always have a 5„PO4 group at one end and a 3„OH group at the other. These ends are often referred to as the 5„ end and the 3„ end and have paramount implicative insinuations in DNA and RNA synthesis that will be discussed later.

The two strands of double-stranded DNA are complementary. Wherever an adenine is in one strand, a thymine is in the other; these two opposing nucleotides are cohered by two hydrogen bonds between them. Similarly, wherever a cytosine is in one strand, a guanine is in the other.

The DNA in a cell customarily occurs as a double-stranded, helical structure. The two strands of double-stranded DNA are complementary. Wherever an adenine is in one strand, a thymine is in the other; these two opposing nucleotides are cohered by two hydrogen bonds between them. Similarly, wherever a cytosine is in one strand, a guanine is in the other. These are cohered by the formation of three hydrogen bonds, a scarcely more vigorous magnetization than that of an A:T pair. The characteristic bonding of A to T and G to C is called base pairing and is fundamental to the remarkable functionality of DNA. Because of the rules of base-pairing, one strand can always be utilized as a template for the synthesis of the complementary opposing strand.

### 3.2 RNA Characteristics

RNA is in many ways commensurable to DNA, but with some consequential exceptions. One difference is that RNA is composed of ribonucleotides rather than deoxynucleotides, albeit in both cases these are customarily referred to simply as nucleotides. Another distinction is that RNA contains the nitrogenous base uracil in lieu of the thymine found in DNA. Like DNA, RNA consists of a sequence of nucleotides, but RNA customarily subsists as a single-stranded linear molecule that is much shorter than DNA. A fragment of RNA, a transcript, is synthesized utilizing a region of one of the two strands of DNA as a template. In making the RNA transcript, the same base-pairing rules of DNA apply except uracil, rather than thymine, base-pairs with adenine. This base-pairing is only transient, however, and the molecule expeditiously leaves the DNA template. Numerous different RNA transcripts can be engendered from a single chromosome utilizing categorical regions as templates. Either strand may accommodate as the template. In a region the size of a single gene, however, only one of the two strands is generally transcribed. As a result, two complementary strands of RNA are not mundanely engendered. Like DNA, RNA consists of a sequence of nucleotides, but RNA customarily subsists as a single-stranded linear molecule that is much shorter than DNA.

DNA can be converted to RNA simply superseding thymine T by uracil U in ribonucleic acid. In the below figure (i) shows how the sample sequence of DNA converted to mRNA.

DNA

ACGT GCGC GATC GCCT GCTA GGCG TACG TCGC AGGC GATC GATG TGCT AGAT CAGA TGAC TCAG TGCA CGAT

mRNA

ACGU GCGC GAUC GCCU GCUA GGCG UACG UCGC AGGC GAUC GAUG UGCU AGAU CAGA UGAC UCAG UGCA CGAU.

The conversion process is much utilizable in central dogma of molecular biology i.e DNA to RNA and RNA to PROTEIN in natural evaluation processes of transcription and translation process which is subsidiary in DNA replication.

### *A. Work flow of the paper*

This paper is organized as follows. Section 4 describes general compression algorithms. Section 5 describes cognate subsisting algorithms to compress genome data. Section 6 describes proposed algorithms analysis how it is more preponderant one than subsisting techniques. Section 7 describes comparative study on a sample sequence. Section 8 is concluding with future work.

## III. GENERAL COMPRESSION ALGORITHMS

Due to the specificity of the particular kind of “text”.

There are several approaches for encoding of texts which are Huffman Encoding, Adaptive Huffman Encoding, Arithmetic coding, Arithmetic adaptive coding, Context Tree weighted method etc. Algorithms designed for DNA Compression like GenCompress, BioCompress, DNA Compress, CTW+LZ, Cfact have achieved an approximate compression ratio of 22% utilize the above mentioned approaches. With BioCompress, at each step, the longest factor beginning at the current position which matches with a factor starting afore is opted for. BioCompress-2 uses arithmetic coding of order 2. Cfact performs in two pass in that it probes for longest exact matching reiterate and utilizes a suffix-tree for finding the longest reiterate. GenCompress works on approximate reiterates in which at each step, it probes for the optimal prefix of the not yet encoded part of the DNA sequence. It uses Hamming distance (v1) and edit distance (v2) for approximate reiterates. CTW+LZ is a Combinaison of GenCompress and CTW uses Context Tree Weighting method. Common components of most of DNA compression algorithms are

- Finding the candidate reiterate segments.
- Considering approximate reiterates.
- Encoding of the reiterate segments.
- Encoding of the non-reiterate segments.

## IV. RELATED EXISTING ALGORITHMS

We can encode every base of DNA by two bits. Compression method mainly categorized into two ways one statistical and other is substitution. In statistical method longer stream are replaced by shorter code and other is dictionary based mechanism. The existing algorithm based on two bits encoding schemes like A (00), C(01), G(10) nd T(11). HUFFBIT[13],GENBIT[14], and DNABIT[15] algorithms are evaluated in Best,Avg and Worst case analysis based on fragments repetitions in the sequences .Suppose in the given sequence more fragments are repeated they achieve Best case if not worst case In this connection our existed techniques achieve 2.25 bits/Bases. But Sequences like AT-rich DNA, which constitutes a distinct fraction of the cellular DNA of the archaebacterium Methanococcus voltae, consist of non-repetitive sequences .So existed algorithms may run in worst case and we will achieve less compression rates. Our proposed algorithms DNASP (DNA Sequence pack) are well suitable for non repetitive DNA sequences and we achieve more compression rates than existing algorithms.

## V. PROPOSED DREAM TOOL

Our technique is developed predicated on the idea of comparative study of subsisting techniques. In this paper we used java language to Performa compression and decompression. In this process we applied of our technique on 8 lakhs bases of dna and observed the results with minimum timing constraints and less recollection storage.Our algorithm is more preponderant suitable for non perpetual sequences and even if repetitive withal we are going to achieve the same compression rate. In this technique compression rate varies with the sequence. By applying different sequences and observed the performance in O(n) in all the cases.

### *B. Basic Idea behind dReaM the Algorithm*

Since the DNA sequence contains only A, C, G, T nucleotides, where each is literal named as BASES. Each four literal are grouped as FRAGMENTS and substituted its equivalent binary bits as follows

A=00, C=01, G=10 and G=11.

Compression Rate is calculated in terms of Encoded bits.

(Compression Ratio) = Encoded bits / Total Number of Bases

### *C. Plan of work*

The input sequence of Dna can be applied to our DREAM tool in FASTA format. Length of the given sequence is divided into four bases i.e A,C,T,G called fragment. In our algorithms the entire sequence is divided into n/16 fragments. The first four figments called predecessor and reaming called successor for first block value and second block value. Consecutively we can calculate main block value for first set by taking the average of Pdv and Sdv by storing it in different indexes. This process can be continued till the end of the sequence

$n = Length\ of\ the\ given\ sequnce$
$Pd = Predecessor\ sequance$
$Sd = successor\ sequance$
*Pdv=Predecessor sequence value*
*Sdv=successor sequence value*
$Fbv = First\ block\ value$
$Sbv = Second\ block\ value$
$Mbv = Main\ block\ values$
$Teb = Total\ number\ of\ encoded\ bits$
$Cr = Compression\ Ratio$

$$Fbv = \sum_{v=0}^{n/16} Pdv$$

$$Sbv = \sum_{v=0}^{n/16} Sdv$$

$$Pdv = \sum_{x=0}^{n/16} Pd$$

$$Sdv = \sum_{x=0}^{n/16} Sd$$

$$Mbv = (Fbv + Sbv) / 2$$

Here Mbv will represent the binary equivalent numeric (nearest to integer) in terms of Bytes storage.

$$T_{eb} = \sum_{v=0}^{n} (Mbv)$$

Finally compression Ratio calculated as follows

$$Cr = (Teb / N)$$

### *D. Analysis*

Let us consider the sequence.

Sequence1:
ACGT GCGC GATC GCCT GCTA GGCG TACG TCGC AGTC GATC GATG TGCT AGAT CAGA TGAC TCAG TGCA CGAT ATCG ACTG CTAG AGAT CAGA TGAC TCAG.

Suppose if we took sample sequence of DNA which contain 96 bases then by applying dReaM techniques it will fragmented into n/32 i.e. 3 main blocks, 3 Fbv and 3 Sb which will contain 24 sub fragment bases and then grouped it into single Mbv partition. This will represent number of encodedv bits in the given sequence.
The first block value can store in pdv and sdv which is together can stored in four bytes and it will repeat for second block value and third and so on. Finally the main block value is summation of different block values. Main block value

will represent total number of bytes required to store the DNA sequance

Fbv=Pdv=Pd1+pd2+pd3
Sbv=sdv=sd1+{sd2+sd3
Mbv=FB1+FB2+FB3
= 4 + 4+ 4
Teb=Mbv=12 bytesi.e 96 bits

Finally, we can calculate the compression ratio in terms of bits per bases.

$$Cr = Compression\ Ratio = \frac{96}{96} = 1.0012$$

Sequence length (no of bases) = 96.
Bytes required to store in a text file = 96 Bytes.

The above sequence dmay contain tandem repeats so existing algorithms like Huff bit compress, Genbit Compress and Dnabit compress may run on best case and require more bits to encode the sequence.

| | |
|---|---|
| Huffbit, GenBit and Dna compress | =204 bits (2.428) |
| Genbit Compress (Tool based) | = 202 bits (2.404) |
| DNASC Compress | = 128 bits (1.523) |
| Splinted Binary compression (SBC) | =96bits (1.142) |
| dReaM tool based approach | =96 bits(1.012) |

***Encoding Algorithm***

```
INS:  input String
OPS: Encoded String
PROCEDURE ENCODE
Begin
•   Group INS into equivalent proper subsets  length as  four bases
•   Generate all possible combinations of DNA and it will contain non- repetitive (our INS assumed as no tandem
    repeats).
•   Assign binary bits(0&1) for every base of DNA like
        A=00, C=01, G=10 and T=11
  Calculate Mbv for every Mb in INS till eof INS
•   Calculate Teb for every Mb till eof INS
•   Repeat  the steps 4 and 5 until the length of the INS
•   Transfer the sequence Eb to the  output string i.e.  OPS String.
End.
```

Decoding algorithm is carried out in the reverse process of encoding because DNA compression is always loss less

***Decoding Algorithm***

```
INS: input String
OPS: Decoded String
PROCEDURE DECODE
Begin
•   Generate all possible combinations of (A,C,G,T)
•   Read the binary data from OPS and assign the two bits by equivalent Base s (00=A,01=C,10=G and 11=T) till eof
•   Repeat step 2 until eof INS is reached and calculate Db and Ds in the reverse process..
•   Transfer the sequence Db to the input String i.e. INS
End.
```

## VI. CONCLUSIONS AND FUTURE WORK

By using of our algorithm we can encode every base by 1.002 bits .By applying of ours we are saving nearer of 8 bytes to encode the given sequence, compression may vary with size of the sequence. So our technique is far better than existing ones and we can apply this technique on non repetitive DNA sequences of genomes .If the given sequence can contain tandem repeats also our technique will achieve same compression rate in an average. In addition to that existing techniques uses dynamic programming to compress the sequence which is complex in implementation and time consuming. Our technique is implemented without dynamic programming approach, so it is simple and fast. The simplicity of this will reduce the complexity in processing. Our algorithm can be extended to any tool based approach.

**References**

[1] E Schrodinger. Cambridge University Press: Cambridge, UK, 1944.[PMID: 15985324]

[2] R Giancarlo et al. A synopsis Bioinformatics 25:1575 (2009) [PMID:19251772]

[3] EV Koonin. Bioinformatics 15: 265 (1999)

[4] JC Wooley. J.Comput.Biol 6: 459 (1999) [PMID: 10582579]

[5] CH Bennett et al. IEEE Trans.Inform.Theory 44: 4 (1998)

[6] S Grumbach & F Tahi. Journal of Information Processing and Management 30(6): 875 (1994)

[7] E Rivals et al. A guaranteed compression scheme for repetitive DNA sequences. LIFL, Lille I University, technical report IT-285 (1995)

[8] X Chen et al. A compression algorithm for DNA sequences and its applications in Genome comparison. In Proceedings of the Fourth Annual International Conference on Computational Molecular Biology, Tokyo, Japan, April 8-11, 2000. [PMID: 11072342]

[9] TC Bell et al. Newyork:Prentice Hall (1990)

[10] J Ziv & A Lempel. IEEE Trans. Inf. Theory 23: 337 (1977)

[11] A Grumbach & F Tahi. In Proceedings of the IEEE Data

[12] X Chen et al. In Proceedings of the Fourth Annual International Conference on Computational Molecular Biology, Tokyo, Japan, April 8-11, 2000.

[13] X Chen et al. Bioinformatics **18**: 1696 (2002) [PMID: 12490460]

[14] An Efficient Horizontal and Vertical Method for Online DNA Sequence Compression in IJCA proceedings 2010 vol.3,Issue 1 June 2010.

[15] Allam AppaRao.In proceedings of the Bio medical Informatics Journal [2011].DNABIT compress-compression of DNA sequences

[16] Loss less segment based compression in IEEE confernece proceedings in ICECT-2011 kanyakumari,India.

[17] Srinivasa K G,Jagadish M, Venugopal K R and L M Patnaik “Efficient compression of non repetitive DNA sequances using Dynamic programming “ pages 569-574 IEEE 2006

[18] National Center for Biotechnology Information, Entrez Nucleotide Query, http://www.ncbi.nlm.nih.gov/htbin-post/Entrez/query?db=n s.

[19] Allam AppaRao in proceedings of the JATIT journal computational biology and Bio informatics:[2011].Huffbit compression of DNA sequances

[20] Allam AppaRao in proceedings of the JATIT journal of computational Biology[2011],Genbit compress fro DNA sequances.

**V Hari Prasad ,A**ssoc.professor, B.Tech CSE from JNTU University,Anantapur,M.Tech CSE from JNTUCEH,HYD and pursuing research in CSE at JNTU KAKINADA, A.P as a Research scholar in CSE .He has 10 years of teaching experience in various Engineering colleges. Presently He is heading the CSE Sphoorthy Engineering college ,Nadergul(V),Hyd. He is a Life Member of MISTE and Member of IEEE.He presented papers at International & National conferences on various domains. His interested areas are Bio Informatics, Databases, and Artificial Intelligence.

**Dr.P.V Kumar** ,Professor of CSE in Osmania University Hyd,Completed M.Tech CSE from Osmania university and PhD (CSE) welding from Osmania university. He has 30 years of Teaching & R&D experience. Many students are working under him for PhD .He has to his credits around 56 papers in various fields of Engineering, Indian and international journals, National and International conferences, He worked as Chairman BOS in OUCE and conducted various staff development programs and workshops. He is Life Member of MISTE, Life Member of CSI..His interested area is temporal databases, Bio Informatics, Data mining and Artificial Intelligence.