

A NEW REVISED DNACRAMP TOOL BASED APPROACH OF CHOPPING DNA REPETITIVE&NON REPETITIVE GENOME SEQUENCES

V.Hari Prasad and Dr.P.V.Kumar²

¹ Research Scholar in Department of CSE
Jawaharlal Nehru Technological University (JNTUK)
Kakinada, Andhra Pradesh,India.

² Professor, Department of Computer Science & Engg
Osmania University
Hyderabad, Andhra Pradesh,India.

Abstract

In vogue tremendous amount of data generated day by day by the living organism of genetic sequences and its accumulation in database, their size is growing in an exponential manner. Due to excessive storage of DNA sequences in public databases like NCBI, EMBL and DDBJ archival maintenance is tedious task. Transmission of information from one place to another place in network management systems is also a critical task. So To improve the efficiency and to reduce the overhead of the database need of compression arises in database optimization. In this connection different techniques were bloomed, but achieved results are not bountiful. Many classical algorithms are fails to compress genetic sequences due to the "specificity of text" "encoded in dna and few of the existing techniques achieved positive results. DNA is repetitive and non repetitive in nature. Our proposed technique DNACRAMP is applicable on repetitive and non repetitive sequences of dna and it yields better compression ratio in terms of bits per bases. This is compared with existing techniques and observed that our one is the optimum technique and compression results are on par with existing techniques.

Keywords- compression; encoding; decoding; bio compress; Huffbit compress;dnabit compress;LSBD compression.

1. 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.

1.1 Motivations

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 essential 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. (Grumbach 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.

2 Basic knowledge of Genome Data

The complete 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.

2.1 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 (see figure 2.22). Joining of the nucleotides in this manner creates 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 important implications 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 held together 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 usually 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 held together by two hydrogen bonds between them. Similarly, wherever a cytosine is in one strand, a guanine is in the other. These are held together by the formation of

three hydrogen bonds, a slightly stronger attraction 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 used as a **template** for the synthesis of the complementary opposing strand.

2.2 RNA Characteristics

RNA is in many ways comparable to DNA, but with some important exceptions. One difference is that RNA is made up of ribonucleotides rather than deoxynucleotides, although in both cases these are usually referred to simply as nucleotides. Another distinction is that RNA contains the nitrogenous base uracil in place of the thymine found in DNA.

Like DNA, RNA consists of a sequence of nucleotides, but RNA usually exists as a single-stranded linear molecule that is much shorter than DNA. A fragment of RNA, a **transcript**, is synthesized using 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 quickly leaves the DNA template. Numerous different RNA transcripts can be generated from a single chromosome using specific regions as templates. Either strand may serve 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 normally generated. Like DNA, RNA consists of a sequence of nucleotides, but RNA usually exists as a single-stranded linear molecule that is much shorter than DNA.

DNA can be converted to RNA simply replacing 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.

Fig.(i)

The conversion process is much useful 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 useful in DNA replication.

2.2.1 Work flow of the paper

This paper is organized as follows. Section 3 describes general compression algorithms. Section 4 describes related existing algorithms to compress genome data. Section 5 describes proposed algorithms analysis how it is better one than existing techniques. Section 6 describes comparative study on a sample sequence. Section 6 is concluding with future work.

3 General Compression Algorithms

The compression of DNA sequences is considered as one of the most challenging tasks in the field of data compression. In this connection the very first DNA compression and its subsequent algorithms BioCompress[10] and BioCompress-2[11] detects exact repeats and complementary palindromes located in the sequence and Encode the factor by the size representation(l, p) where l is the length of the factor and p is the position of its first occurrence .If the size is greater the factor then use two bit encoding . More memory references will require decoding the same, so the performance may degrade. In addition to that some of the lossless algorithms CTW, Gen2, and GenML are also available to compress DNA sequences but they are never achieved higher efficiency for longer sequences.

Gencompress [12] is a lossless compression algorithm for genetic sequences based on searching for approximate repeats of hidden regularities of DNA sequences .This algorithm achieved compression rate of 1.800 bits per bases in an average.

DnaPack[13] which uses hamming distance for the repeats and complementary palindromes and it is implemented by dynamic programming approach. So that it is not simple in design and it will require more time to execute and require more memory requirements also. The algorithm achieves a compression rate in an average of 1.777.

DNASC [14] was developed by applying both substitution and statistical methods based on 128 conditions to preserve loss property of genome sequences. In this technique they showed how transformation can be performed from one location to another location. This technique is inspired by Gen2 and GenML of horizontal and vertical methods of compression data [Karodi and Tahi] and achieved compression rate of 1.501 in an average for both repetitive and non repetitive sequences.

Some more algorithms are proposed exclusively for non repetitive sequences like Srinivasa at el [17] This

algorithm is pair based matrix generation developed in two passes. In two passes every two DNA bases are replaced by single base in i.e. A and G represented by A and G and T represented by T and by doing it in the reverse process they achieved original sequence in decoding. Due to Dynamic programming its implementation is complex and requires more memory references.

The Lossless segment based compression LSB [16] enables part by part decompression by introducing non base character so that it will save memory requirements but it is applicable well on repeating sequences are more and more in the sequence. If such sequences like AT-rich DNA, which constitutes a distinct fraction of the cellular DNA of the archaeobacterium *Methanococcus voltae*, consists of non-repetitive sequences, so part by part decompression is little bit tedious.

3.1 Related Existing Algorithms

Dna compression is always loss less, we have to retain its original property after decoding. Zipping and unzipping is one of the secured mechanism we can use the DNA data transmission from one location to another location. Most of the existing techniques mainly classify into one is substitution and other is statistical. First mechanism is replacement of short code by the longer sequences and other is dictionary based mechanism. With the spirit of substitution and statistical techniques many lossless Genome compression algorithms are strived based on two bits encoding scheme i.e. A.[00], C[01], G[10] and T[11]. Some of the algorithms like DNASC [14] and DNABITcompress [15] will work on approximation of repeats if number of tandem repeats more it saves bits to encode if not discard. Non repeated sequences will be appended to the sequence at the end. This algorithm achieves a compression rate 1.583 bits per base.

Our proposed new algorithm is developed based on comparative study of existing techniques and achieves better compression rate i.e. 1.1428 which is on par of existing ones. Our algorithm is applied on both repetitive and non repetitive sequences of DNA and achieved the same compression rate in best, avg and worst cases. Really it is a first-rate technique it is achieving the same compression in all the cases and it is very simple in design and it will take less time for execution also. In implementation our technique performance is varying in a linear way in all the cases proportional to the length of the sequence even the sequence may contain tandems repeat of

bases in DNA (DNA composed of text bases like alphabet and every alphabet is called as base) .

4. Proposed Algorithm

Our proposed technique DNACRAMP tool is developed based on the idea of comparative study of existing technique. In this technique we used basic procedural language to perform encoding and decoding process with the help of two-stage index bounded array linear data structure. We took different phosphates, viruses and human genome sequences and applied the technique over it and observed better compression rate with minimum timing constraint and less memory storage in a loss less way. Our technique can be applied for both repetitive and non repetitive sequences of dna with same compression rate in best, worst and avg cases. In this technique compression rate varies linearly with the sequence so that the performance analysis is $O(n)$ in all the cases.

4.1 Idea behind the Algorithm

Every DNA sequence contain {A, C, G, T} nucleotides where each literal is named as BASE and encoded in two bits as follows

$$A=00, C=01, G=10 \text{ and } T=11.$$

Compression ratio is calculated encoded bits per Bases.

$$\text{Compression Ratio} = \text{Encoded Bits/Bases}$$

4.2 Plan of work

Here we took a a sample sequence of dna and divided into $n/4$ fragments (where every fragment contain four bases). In the first phase we can group quadrupled fragments into two sub partitions ,the first one is first header end and trailer end i.e FHr and FTr and second one is second header end trailer end i.e SHr and STr which is grouped into a cluster set Cs. The cluster set value can stored in an array index. So every cluster set will contain $n/28$ bases where two sub headers and trailers contain $n/14$ bases. Sub sequent cluster set values can be calculated for an entire sequence and grouped into regional set Rs. we can calculate binary equivalent numeric value of every cluster set and grouped to Rs. Here Rs will represent total number of encoded bits.

Suppose if we took the sample sequence of DNA contain 84 bases, in the first stage is divided into 21 quadrupled fragments, triplet cluster sets, which will include 3 header ends and 3 trailer ends , one regional set and finally grouped to main segment which will represent total encoded bits. Our DNACRAMP will work as follows.

Now the cluster set value can calculate as follows.

$$C_{st} = \sum_{s=0}^n C_s$$

For the first segment of cluster header end and trailer end values and sub sequent values can calculate as follows

$$S_{Hr} = S_{h1} + S_{h2} + S_{h3} + \dots + S_{hn}$$

$$S_{Tr} = S_{t1} + S_{t2} + S_{t3} + \dots + S_{tn}$$

$$F_{Hv} = \sum_{r=0}^{n/14} (F_{Hr} + F_{Tr})$$

$$SHv = \sum_{r=0}^{n/14} (S_{Hr} + S_{Tr})$$

To calculate cluster set values we have to calculate cluster set.

$$C_s = (F_{Hv} + S_{Hv}) / 2$$

$$R_{sv} = \sum_{v=0}^n C_{sv}$$

Here Rsv will represent the binary equivalent numeric (nearest to integer) in terms of Bytes storage. (Suppose if we will implement the technique in C language unsigned long will require 4 bytes of storage). Here $m=n/28$ i.e. length of the given sequence.

Total Number of Encoded group bits are calculated as follows.

$$T_{eb} = \sum_{s=0}^n (R_{sv})$$

Finally compression Ratio calculated as follows

$$Cr = (T_{eb} / N)$$

4.3 Analysis

Sequence1 :(HUMDYSTROP)

TTTT CGAA TTNA CCTC GTTN CCTG CCTA
 ACCT CCGA TGCA ACGT AGTA GCTG GGAC TACA
 GGCG CCTG CCGG CGCA CCGG GCTA

Sequence2:

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

The above sequence is human genome sequence contain 84 bases and we may find tandem repeats in it, and N is considered as unknown nucleotide i.e. blank space. DNA contain 84 bases, in the first stage is divided into 21 quadrupled fragments, triplet cluster sets, which will include 3 header ends and 3 trailer ends, one regional set and finally grouped to main segment which will represent total encoded bits. Now the sample data can be extended to the DNACRAMP for the calculation main partition value i.e. Rsv. Now or the first and subsequent cluster set values i.e. FHv, FTv, SHv and STv for sequence2 calculate as follows.

```
0001101110011001100011011001
1100011011011001001010011000
0010001110001000111000010110
```

```
0111100111001010011011000110
1110111001110010001101000010
0011011000101110010010010111
```

Now we can calculate the binary equivalent numeric and stored in separate bi stage index array and based on the index values we can calculate the cluster set value and regional set value based on the above formula

$$C_{st} = \begin{pmatrix} 28940505 \\ 227711192 \\ 236793414 \\ 127706822 \\ 250028866 \\ 56812695 \end{pmatrix} \quad M_p = \begin{pmatrix} 128325848 \\ 182250118 \\ 153420780 \end{pmatrix}$$

4.4 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 Cs for every Ps in INS till eof INS
- Calculate Eb for every Cs 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)

the two viruses: VACCG and HEHCMVCG and computed results in terms of bits per bases on par with existing algorithms as shown in fig(iii)

6. Conclusion

This technique is implemented without dynamic programming so its not that much complex likes other algorithms in implementation. It is very simple in design by using this technique we encode every base by 1.19 bits, which is on par with existing technique's. DNACRAMP is versatile technique in optimizing the time and space

Fig(iii)DnaCramp comparison with existing algorithms

Sequence	size	CTW	Bio2	Gen2	DnaPac k	Dnabit	DnaSC	DNACRAM P
CHMP XX	121024	1.838	1.684	1.673	1.660	1.517	1.500	1.14295
CHNT XX	155844	1.933	1.617	1.614	1.610	1.584	1.510	1.14286
HEHCMVC G	229354	1.958	1.848	1.847	1.834	1.573	1.800	1.15283
HUMD- YSTROP	33770	1.920	1.926	1.922	1.908	1.572	1.890	1.14359
HUMH BB	73308	1.892	1.88	1.820	1.777	1.606	0.910	1.14279
HUMHDAB CD	58864	1.897	1.877	1.819	1.739	1.606	1.610	1.14325
HUMHPRT B	56737	1.913	1.906	1.846	1.788	1.574	1.710	1.14286
MPOMTCG	186609	1.962	1.937	1.905	1.893	1.565	1.880	1.13778
VACCG	191737	1.857	1.761	1.761	1.758	1.652	1.700	1.14286
AVG		1.907	1.796	1.800	1.774	1.583	1.612	1.14353

4.5 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

5. Example and comparison of Results


Propose tool DNACRAMP applied on different sequences like two mitochondria: MPOMTCG, PANMTPACGA (also called MIPACGA); two chloroplasts: CHNTXX and CHMPXX (also called MPOCPCG) sequences from humans: HUMGHCSA, HUMHBB, HUMHDABCD, HUMDYSTROP, HUMHPRTB, complete genome from

requirements.


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 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, Assoc. 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.