Dynamic Programming Based DNA Compression Algorithm through Substitution Method

Annwesha Banerjee Majumder, Somsubhra Gupta

Abstract— **In this paper, a DNA sequence compression algorithm through substitution mechanism has been proposed. The field of bioinformatics research deals with huge DNA data, which requires to be compressed for proper storage utilization. Different symbols for the different sequence of occurrence of DNA sequence has been chosen and compressed the sequence. Dynamic programming concepts have been explored in the proposed algorithm.**

Index Terms— **Bioinformatics, Compression, DNA, Dynamic programming.**

I. Introduction

Today is the era of Big Data. A very common example of huge database driven area of technology is Biological data computation. Computers play a very important role of biological data analysis. Biological data always require a very vast amount of data analysis like DNA sequencing which also require a very vast storage volume. As a result of which Biological data compression becomes an indispensible part of computational biological research field. Data compression is the technique to reduce the total number of bits. Through compression we can encode the actual message to fewer bits representation so it can take lesser amount of storage and transmission time will also be reduced [1].

DNA contains all hereditary information in form of 4 letters A,T,C and G [2]. DNA sequence have different features where some strings are off repeated, some are palindrome and some are reverse palindrome.[3] which has positive influences in compression. Use of compression in biological data has the advantages of data structure, data modeling and speed [4].

Dynamic programming is an important technique for the solution of problems involving the optimization of a sequence of decisions. The simple idea underlying this technique is to represent the problem by a process which evolves from state to state in response to decisions. A typical

optimization problem then becomes one of the guiding system to a terminal state at minimum cost.

When the cost structure is appropriate, the determination of an optimal policy (sequence of decisions) may be reduced to the solution of a functional equation in which the state appears as an independent variable

Dynamic programming is a very use full model for problem solving in computer science [23]. Actually dynamic programming is based on the work of Bellman [25] and has been applied to problems in operation research, mathematics, computer science and many other fields.

Dynamic programming includes divide-and-conquer, linear and integer programming, and branch-and-bound. Divide-and-conquer actually divide the problem set in two set [25]. Linear and integer programming are applicable to problems with linear optimization functions and linear constraints [26]. Branch-and-bound is commonly used for pruning of large tree searches, and is also closely related to DP [27, 28].

II. Literature Survey

Many biological data compression algorithms have already proposed and having their different advantages.

BioCompress and BioCompressed 2 are two algorithms proposed by Grumbach and Tahi [5], [6] based on Ziv and Lempel data compression method to find repeats, palindrome and reverse palindrome.

E. Rivals et al. proposed cfact [7] which use Suffix Tree data structure to find longest repeats. Chen et al. proposed GenCompression which is a substitution based compression algorithm[8]. CTW[9] is another compression algorithm proposed by Matsumoto et al. that use Context Tree Weighting[10] method which is be good for short repeats. DNA Compress proposed by Chen et. al [11] also used the Lempel and Ziv compression scheme. Dynamic programming technique can also be used to compress DNA sequence and that was proposed by Behshad Behzadi et al. [12] Substitution and statistical combined approach is being used in an algorithm Normalized Maximum Likelihood proposed by Tabus.[13] HUFFBIT COMPRESS[14] is a extended binary tree based compression algorithm and its compression rate is 1.006 bits per phase.

Look Up Table compression algorithm is based on fixed length LUT and LZ77 proposed by Sheng Bao et al, 2005[15,16,17] Raja Rajeswari et al. [16] developed *GenBit Compress that is based on extended binary tree.[18].* Taysir Soliman et al. proposed an algorithm that works on repeats and reverse palindrome sequencs[19]. Kamta Nath Mishra et al porposed another statistical and substitution methods named as DNACS[20] Differential Direct Coding [21] this algorithm accommodates large data sets, consist of multiple sequences and auxiliary data and LSDB METHOD[22] performs gene based compression and reduce memory requirement. A symbol driven DNA Compression algorithm has also been proposed in cloud environment.[29]

III. Proposed Method

The proposed DNA compression algorithm is based on substitution technique. The total string is divided into 4 symbols long sequences and then searched for repeat, reverse repeat and non repeated sequences and encoded to compress form. After compression of first set of 4 symbols the sequence

Mrs. Annwesha Banerjee Majumder Department of Information Technology, JIS College of Engineering, Kalyani, Nadia, West Bengal, India
Dr. Somsubhra Gupta, Department of Information Technology, JIS College of Engineering, Kalyani, Nadia, West Bengal, India

is checked for the equality of the second sets, if consecutive two set of symbols are same apply the previously computed result without further computation that will save computation time.

γ =length of DNA sequence
τ= length of compressed sequence

$$\text{Compression Ratio } (\Omega) = \gamma / \tau \qquad (1)$$

```
COMPRESION(SET[i])
{Save result in a Intermediated variable}
    IF(SET[i]==SET[i-1])
            {Use previously computer result}
    ELSE
            { Compute Result Again}
```

A. Compression

1. Sub divide the DNA sequence into block each having 4 symbols
2. Assign each symbols following code:
   A: 00 T: 01 C: 10 G: 11
3. A. If All the four symbols are same (i.e. AAAA) use only one symbols and put it in new array
   B. Divide each block into two sub block having 2 symbols
   i. If two blocks are same then use first block and then append + after and keep it new array.
   ii. If two blocks are reverse to each other then use first block and then append - after and keep it new array.
   C. If consecutive three symbols are same then use symbol single time and then * and keep it new array.
   D. If above any conditions are not matching then use binary code for each symbols

B. Decompression

1. Read each four blocks
2. A. If the compressed array contain single symbol decompress it by replicate it four times\
   B. If compressed array contain 3 character long sequence along with + at end then decompress it by using first two consecutive symbols two times deleting +
   C. If compressed array contain 3 character long sequence along with - at end then decompress it by using first two symbols followed by reverse sequence of these symbols deleting –
   D. If compressed array contain 3 characters along with '*' then use the previous symbols three times along with remaining symbols in proper position
   E. If any one of the above condition is not matched that means the DNA sequences are compressed by using binary sequences as follows
   A: 00 T: 01 C: 01 G: 11

C. Case Study

Followings are some case studies considering different DNA sequence and analysis the Compression rate.

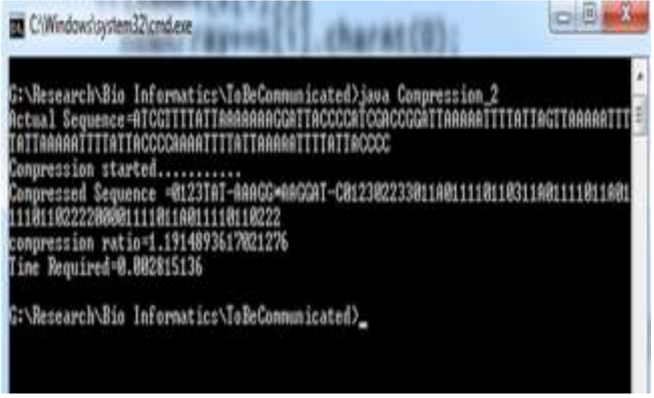
CASE STUDY 1:
AAAA---→A
γ=4
τ=1
Compression ratio (Ω): 4

CASE STUDY 2:
ATAT--→ AT+
γ=4
τ=3
Compression ratio: 1.33

CASE STUDY 3:
GAAG-→GA-
γ=4
τ=3
Compression ratio (Ω): 1.33

CASE STUDY 4:
AAAG-→A*G
γ=4
τ=3
Compression ratio (Ω): 1.33

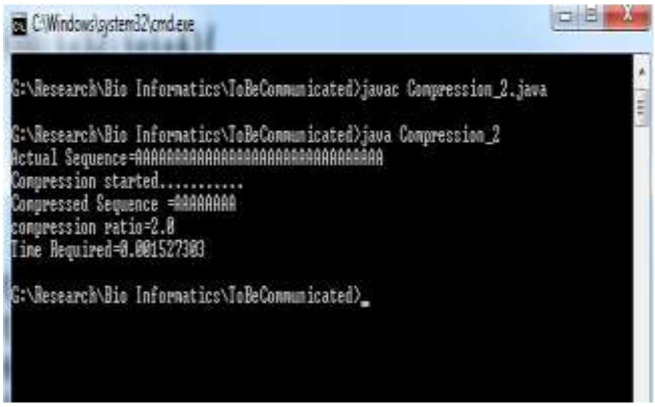
D. Result Analysis



```
G:\Research\Bio Informatics\ToBeCommunicated>java Compression_2
Actual Sequence=ATCGTTTTATTAAAAAAAAGGATTACCCCATCGACCGGATTAAAAATTTTATTAGTTAAAAATTT
TATTAAAAATTTTATTACCCCAAAATTTTATTAAAAATTTTATTACCCC
Compression started..........
Compressed Sequence =0123TAT-AAAGG*AAGGAT-C012302233011A011110110311A01111011A01
1110110222200001111011A011110110222
compression ratio=1.1914893617021276
Time Required=0.002815136

G:\Research\Bio Informatics\ToBeCommunicated>_
```

Fig: Result Analysis-1



```
G:\Research\Bio Informatics\ToBeCommunicated>javac Compression_2.java

G:\Research\Bio Informatics\ToBeCommunicated>java Compression_2
Actual Sequence=AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
Compression started..........
Compressed Sequence =AAAAAAAA
compression ratio=2.0
Time Required=0.001527303

G:\Research\Bio Informatics\ToBeCommunicated>_
```

Fig 2: Result Analysis-2

E. Comparison with Existing Algorithm

Algorithm	Compression Rate
GeneCompression	1.7428
DNACompression	1.7254
CBSTD	1.82

Proposed Algorithm	4(when consecutive 4 symbols are same)
	1.33(In other cases)

(i)

IV. Conclusion

A substitution based DNA compression algorithm has been proposed that is based on dynamic programming.This proposed method has achieved a good compression rate having the added advantages of dynamic programming. Further investigation required for upgrading the compression rate.

Acknowledgment

We are grateful to our institution for providing us support from all aspects.

References

[1] R.S. Brar and B. Singh, "A survey on different compression techniques and bit reduction Algorithm for compression of text data" International Journal of Advanced Research In Computer Science and Software Engineering (IJARCSSE) Volume 3, Issue 3, March 2013

[2] Wong, L., "Some New Results and Tools for Protein Function Prediction, RNA Target Site Prediction, Calling, Environmental Genomics, and More", Journal of Bioinformatics and Computational Biology, Vol. 9, No. 6, 2011.

[3] Wong, L., "Some New Results and Tools for Protein Function Prediction, RNA Target Site Prediction, Genotype Calling, Environmental Genomics, and More", Journal of Bioinformatics and Computational Biology, Vol. 9, No. 6, 2011.

[4] R. Giancarlo, D. Scaturro and F. Utro, "Textual Data Compression in Computational Biology: a synopsis," Bioinformatics, vol. 25, no. 13, pp. 1575 - 1586, 2009.

[5] Grumbach, S. and Tahi, F., "Compression of DNA Sequences", In Proc. IEEE Symp. On Data Compression, pp. 340-350, 1993.

[6] Grumbach, S. and Tahi, F., "A new challenge for compression algorithms: Genetic Sequences", Journal of Information Processing & Management, Vol. 30, pp. 875-886, 1994.

[7] Rivals,E., Jean Paul Delahaye, M., Dauchet and Delgrange,O.,"A Guaranteed Compression Scheme for Repetitive DNA Sequences", In Proc. Data Compression Conf. (DCC-96), Snowbird, UT. p453, 1996.

[8] Chen, X., Kwong, S. and Li, M., "A compression algorithm for DNA sequences and its applications in genome comparison", The 10thworkshop on Genome Informatics (GIW-99), pp.51–61, Tokyo, Japan, 1999.

[9] Williems, Shtarkov and Tjalkens, "The context tree-weighting method: Basic properties", IEEE Trans. Info. Theory, pp.653-664, 1995.

[10] Matsumoto, T., Sadakane, K., Okazaki, T. and Imai, H., "Implementing the context tree weighting method by using conditional probabilities", Proc. of 22ndSymposium on Information Theory and its Applications, pp. 673–676, SITA, December 1999.

[11] Chen, X., Li, M., Ma, B. and Tromp, J., "DNACompress: Fast and effective DNA sequence compression", Bioinformatics, Vol. 18(12), pp. 1696–1698, 2002.

[12] Behzadi, B. and Le Fessant, F., "DNA Compression Challenge Revisited", Symposium on Combinatorial Pattern Matching (CPM2005), pp.190-200, June 2005.

[13] Tabus, Korodi and Rissanen, "DNA sequence compression using the normalized maximum likelihood model for discrete regression", DCC, p253, 2003.

[14] P.Raja Rajeswari Dr. Allam Apparao Dr. R.Kiran Kumar "HUFFBIT COMPRESS – Algorithm to Compress DNA Sequences Using Extended Binary Trees." Journal of Theoretical and Applied Information Technology Page(s): 101-106 2005 – 2010

[15] R.K.Bharti,2011, et al, "A Biological sequence compression Based on Approximate repeat Using Variable length LUT" International Journal of Advances in Science and Technology, Vol. 3, No.3,PP:71-75.

[16] R.K.Bharti, 2011, et al.,"Biological sequence Compression Based on Cross chromosomal properties Using variable length LUT", CSC Journal, Vol 4 Issue 6, PP:217-223.

[17] R.K.Bharti,2011, et al, "Biological sequence Compression Based on properties unique and repeated repeats Using variable length LUT" CiiT journal, Vol 3 Issue, 4, PP: 158 – 162

[18] Raja Rajeswari and Dr.AlamApparao, "GenBit Compress-Algorithm for repetitive and non repetitive DNA sequences", Journal of theoretical and applied information technology, pp. 25-29, 2010.

[19] Soliman, T., "A Lossless Compression Algorithm for DNA sequences", International Journal of Bioinformatics and Applications, Vol. 5(6), pp. 593, 2009.

[20] Kamnath Mishra, Dr.Anupam Agarwal, Dr.EdriesAbdelhadi and Dr. Prakash C. Srivasatava, "An Efficient Horizontal and Vertical Method for Online DNA Sequence Compression", IJCA, Vol. 3(1), pp.39-46, June, 2010.

[21] Bolshoy,A. (2003) DNA sequence analysis linguistic tools: contrast vocabularies,compositional spectra and linguistic complexity. Appl. Bioinform., 2, 103–112

[22] Choi Ping Paula Wu, 2008, et al., " Cross chromosomal similarity for DNA sequence compression", Bioinformatics 2(9): 412-416

[23] Dynamic programming in computer science Kevin Q. Brown Carnegie Mellon University

[24] R. Bellman , Dynamic Programming, Princeton Universit y Press , Princeton , New Jersey , 1957

[25] A.V . Aho , J . Hopcroft, and J.D. Ullman, The Design and Analysis of Computer Algorithms, Addison-Wesley , Reading, Mass., 1974. Constructing optimal binar y searc h tree , pp . 119-123. G.B. Dantzig , Linear Programming and Extensions, Princeton Universit y Press , Princeton , New Jersey , 1963.

[26] T . Ibaraki , Branch-and-bound procedure and state-space representation of combinatorial optimization problems. Information and Control , 36 (1978), pp . 1-27 .

[27] W.H. Kohle r and K. Steiglitz , Enumerative and Iterative Computational Approaches, Compute r and Job-Sho p Scheduling Theory , E.G. Coffman, Jr., ed., Joh n Wile y & Sons , N e w York , 1976, pp . 229-287 .

[28] A.Banerjee Majumder, SS Gupta "CBSTD: A Cloud Based Symbol Table Driven DNA Compression Algorithm" Industry Interactive Innovations in Science, Engineering and Technology 2016 , Lecture Notes in Networks and Systems 11, DOI 10.1007/978-981-10-3953-9_45

Annwesha Banerjee Majumder She has completed her M.Tech and B.Tech in MCNT and Information technology respectively from West Bengal University of Technology. Currently she is working as an Assistant Professor in the department Information Technology in JIS College of Engineering. She is doing her research work since 2013.

Dr. Somsubhra Gupta is presently the Dean of the Academic Affairs and Associate Professor in JIS College of Engineering, (An Autonomous Institution), Kalyani, WB.. He has graduated form **University of Calcutta**, obtained his Master's Degree from **Indian Institute of Technology, Kharagpur** and received the Ph.D. from University of Kalyani. He has two decades of experience, publications of 67 research papers (IEEE Xplore, Science Direct, Springer etc.) 2 Book Chapters, over 70 citations, having **3 Funded Projects** as Principal Investigator / Project Coordinator, (AICTE MODROB/RPS), Editor of 2 Proceeding books, and published **2 Patents** (as Inventor). He received **Bharat Vidya Shiromani Award** by the International Business Council, New Delhi and **Outstanding Faculty Award** by the Centre for Advanced Research and Design (CARD), **VIF** Chennai in July 2015. Couple of his papers won **Best Paper Awards**