IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol-6, Special Issue-4, May 2018 E-ISSN: 2347-2693

Reference Based Genomic Data Compression Using R Programming

M. Mary Shanthi Rani [1*], S. Jegatheesh Chandra Bose [2]

[1*] Dept.of Computer Science and Applications, The Gandhigram Rural Institute (Deemed to be University), Dindigul, India
[2] Dept.of Computer Science and Applications, The Gandhigram Rural Institute (Deemed to be University), Dindigul, India

Corresponding Author: m.maryshanthirani@ruraluniv.ac.in[1], Tel.:+91 9445803558

Available online at: www.ijcseonline.org

***Abstract*—** Genomics has become a hot research area in medical field for diagnosis of monogenetic disorder identification, pharmaco genetics, targeted therapy, genome editing and personalized medicine. Each human genome consists of 3 billion pairs which are to be effectively stored and transmitted for analysis. This process necessitates the development of novel genomic data compression algorithms. In this paper a referential based method for compressing genomes has been proposed. The input and reference genomes are compared for dissimilarities and further entropy coded to achieve high compression ratio.

***Keywords*—** FASTA file, Genomic data compression, R-Programming, Huffman coding, BIG DATA.

I. INTRODUCTION

The exponential development in data produced by next generation sequencing (NGS) platforms poses a big challenge for data storage infrastructures [1]. The modern sequencing platforms create terabytes of genomic data in a single run, and their throughput is expected to increase 3-5 times each year. An effective solution to this challenge is data compression, which has consequently become a major area of interest in genomics research. FASTA and FASTQ are two popular formats for storing genomic data. These formats are based on a plain text representation of the genomic data components, namely the sequences of DNA letters, quality scores, and meta data, as shown in Figure1. These components are typically compressed independently using strategies which best suit the data characteristics. A reference-based approach is often adopted for compressing sequence component, whereby the sequences are encoded as a mapping to a known reference sequence. Given the high similarity of inter-species DNA (over 99.9% for Human DNA), this strategy typically achieves higher compression ratios than non-reference-based strategies. Several reference-based compression tools have been implemented in software, including FASTQZ [2], FQZCOMP [2], LWFQZIP [4], and gdc2 [3]. These tools achieve higher compression ratios than general purpose compression algorithms such as GZIP and bzip2, however the improvement in compression ratio comes at the cost of the compression speed. For example, most of the tools listed would require several hours to compress the data generated by a single NGS platform run.

```
1. >chr22
2. TAGAAGTTCTCTGAGACCTAGGCTTTGTGAATCCAA
3. AGGGATCTTTTTAACGAATAAAATGAATCAGGGCCC 4
4. AATGGGACGTGAGGGTTCCTCAGGCCAGTAGTATGG
```

Fig. 1: Genomic data formats.

(a) FASTA format. Lines with the symbol > in the first position contain meta data, the remaining lines contain sequences. No quality scores are stored.

```
1. @HWI – EAS 209 : 5 : 58 : 5894 : 21141 # ATCACG/1
2. GGGTGATGGCCGCTGCCGATGGCGTCAAATCCCACC
3. +HWI – EAS 209 : 5 : 58 : 5894 : 21141 # ATCACG/1 4
4. I IIIIIIIIIIIIIIIIIIIIIIIIIII 9 I G 9 I C
```

Fig. 2: FASTA formats

(b) FASTQ format. Each record is stored over 4 lines. Lines 1 and 4 contain meta-data, line 2 contains the sequence, and line 4 contains the quality scores as shown in Fig. 1.

The data generated by NGS Platforms in recent years is very high which makes it hard for the existing tools to cope with the storage of massive amounts of genomic data.

The great challenge in reference based compression is the mapping to the reference sequence which accounts for over 70% of the compression time. Field Programmable Gate Arrays (FPGAs) are a promising candidate for accelerating reference-based compression: first, there are several successful works on accelerating sequence alignment [5], [6], [7] which comprises a similar mapping problem; and second, the low operational clock frequencies of FPGAs allow compact and energy efficient solutions appropriate for data centers and clinical settings.

II. LITERATURE SURVEY

Kwang Su Jung *et al.* [8] proposed a method for compressing genome sequence cluster using sequence alignment. The author has defined a sequence cluster as a cluster that is constituted of similar sequences and also, has presented a new compressing technique for sequence clusters using a sequence alignment method.

M.Mary Shanthi Rani [9] proposed a novel referential based method for compressing genomes. The algorithm finds the matching blocks in the reference chromosome and the resulting indices are compressed using delta encoding. Biji CL and Achuth sankar S.Nair [10] presented a whole genome sequence compression using benchmark dataset. The author has discussed about the current state of achievement in DNA compression. Also, has proposed a benchmark dataset using multistage sampling procedure.

RabiaArshad *et al.* [11] described a performance comparison of huffman coding and double huffman coding. The author has proposed a double huffman coding using the code word of the symbol which is compressed on binary basis.

Komal Sharma and Kunal Gupta [12] discussed about the performance of lossless data compression techniques. The author presented a study of various lossless data compression techniques and compares their performance and efficiency using time and space complexity.

Kakoli Banerjee and R. A. Prasad [13] proposed a compression algorithm using reference based inter-chromosomal similarity based DNA sequences. The author has presented a new genetic compression algorithm which has explored inter chromosomal similarities and yield positive compression.

III. PROPOSED METHOD

In this paper, a reference based compression of genomic sequences is proposed. Generally, genomic sequences of similar species show great similarity except for some mismatches. These mismatches are due to insertions/deletions of some sub sequences during genomic sequencing. The basic idea of reference based compression is to find the similar and dissimilar blocks of fixed size between the input and reference genomes. The compressed stream will contain only the block numbers/indices of the similar blocks. The block indices of the similar blocks are further compressed using entropy coding. The dissimilar blocks are split into small sub sequences with a minimum length M_L of three characters and are compressed using Huffman Coding.

The proposed algorithm is listed below

1. Divide the Input genome into fixed size blocks of size BS_I
2. Divide the Reference genome into fixed size blocks of is BS_R.
3. Map the input block with reference block
4. If match found, write the block index and position into the compressed stream
5. If not, split the input block into sub sequences of minimum length M_L and compress using Huffman coding.

The block size of reference genome BS_R is chosen to be greater than the input block size BS_I. The dissimilar blocks are further divided into sub sequences and subjected to Huffman coding based on the probability of occurrences of each subsequence. The motive of using Huffman coding is that there are high chances of repetition of some subsequences which can be effectively compressed by assigning short codes. Infrequent subsequences are assigned longer codes.

IV. RESULTS AND DISCUSSION

The performance the proposed method is evaluated using real genomic data from the UCSC genome browsers. The genomic sequences used in our experiments are HG18 release genome, watson JW genome, korean genomes KOREF20090131, KOREF20090224 and the Han Chinese genome YH. The process of matching genome sequences is done using R- Programming which is an effective analytical tool based on statistical computing techniques. It is a open source software that provides a programming language environment for processing Big Data[8].

R programming code is written in two ways; R-Editor and R-Console. For processing genomic sequences, bio strings package should be installed and declared in the header file. The dataset are processed in line by line and displayed as shown in following Table 1.

The input and reference blocks are matched using vmatch command in R which requires the following packages: library ("s4vectors"), library("stats4"), library("IRanges"), library("GenomicRanges"), library("GenomicInfoDb"), library ("Biostrings"), library("XVector") and library ("rtracklayer"). Fig. 3 lists the block indices of dissimilar blocks. The dissimilar blocks are subdivided into substrings of length M_L and their frequency of occurrences are found for compressing using Huffman Coding.Table 2 shows the matching of given substring or subsequence of length 3 using Vmatch command that reports the block index(group) and the start and end positions in the reference block.

Table.1 Output of reading line by line in Genomes

	row.names	x
1	P46644 cTP 43	MKTTHSSSSSSDRRGARHNSGSDSDNSSYASTSGGTGGSVSHVADGVTV>
2	P46248 cTP 52	MASMSGSTSRNKDKKGTSASNKAKSSRVTMTVAVKSRGTMADGVSAKAD>
3	Q96375 cTP 49	MYASSARDGGKWCNARRKSKDAYHSCKSNGHKKVKGVKATAAATTKSNS>
4	O81360 cTP 50	MASTYNSMNSAAVSRTHNKDSCHTDYHRSRTRSGKKCTVRATVASTVSA>
5	P93092 cTP 52	MASVTGTSSMASKASASRVSNRSVSKGKSARMRSARVVCCAAKTVKVCA>
6	P52411 cTP 56	MASAAAGASCKSASSAGRSSRSVSVSRKSSKSSKSSARVSCAKTVAKVC>
7	P02902 cTP 59	MAHCAAVSSSSAVRRRSSVANVVSSRSSVSHSRMSVSSSRSSRKCCAAM>
8	P08817 cTP 49	MASAAASAVSARVKACVNSVSSARKDNVSRVRSVCCAAKKTVKVCDVKS>
9	P52413 cTP 60	MATAAAGSSCKSASCSNRAVSGSSRSVSSGKSRSSRGSRVCCAKTVTRV>
10	P15543 cTP 49	MASAGSAVSAKVKANTNSSSGARRGNARVMRAVCCSAKDTVKVCVKKAV>
11	P49079 cTP 92	MRSTVASRHGAASTRRRHAAAGRDSTRCWRWKTDSSGSSRTSRRTVHGD>
12	Q05753 cTP 36	MSSTHTSRTSSRSSHSATRRSSYSSTSDAGDDVGDCVYDGVDYDKVTVA>
13	Q42690 cTP 27	MAMMKSSASKAVSAAAASCAGKYDKTAGTVASKGRGAMDSNATCGKRDS>
14	P16096 cTP 46	MASASKTSVDNKGTRSVAGVRTSGSSSTVRASSYADVKTAKTVASGRGA>
15	Q10712 cTP 53	MATRVSSASSSSSHSNSVTKYSSKWASVTCSKRSKRVHCAGDTGTRNSD>
16	Q42876 cTP 48	MNGVCSSSSSHSYSTKSSWSSSVTCSRRAKRMAHSARDTGTHTNSDAKS>
17	P92979 cTP 53	MAMSVNVSSSSSSGNSRGVSKVSGSRDRVHVAVSNSGKRSSSVKNAKTK>
18	P92981 cTP 66	MAAVTSSSTASGSSSRSGASSSKACSRSDRTHSRRYSMKNASHSRSSWV>
19	Q42884 cTP 54	MASVTKVGASSSSDGSSRVSSKSSSNHSRSKRAAGSTGNYRVTTGSHGG>
20	P23981 cTP 74	MASSMGGRTNNSYKTKVSHSGSKKTNSAKSWVSKDSVRVAKSRSASVVT>
21	P05466 cTP 76	MAVSRCNGVNSSNSKSSRKSSVSKTHRAYSSSWGKKSGMTGSRKVMSSV>
22	P57720 cTP 49	MASSSTSKSGSTKGSSSSRRSSAVSRTTRKNATGSSYGTHRVSTGSHGG>
23	P27793 cTP 57	MASSSTKSGSRRRSTTDGSGWSYTSDRSNSVSVRRTAKVVASGSSGKVV>
24	P29976 cTP 52	MASNASSSTRSYGGDSHRSNRSSTHAVNTKKSVNVTAVHAAARNAVSVK>
25	P21357 cTP 74	MASSTSTTNSNRSVNSKNASNNSTKTVRVSAVHSSDSNKVSDKSKSSAA>
26	Q9SJ05 cTP 67	MAATRYSWVDCRKVCKRGSRYSVKVDRRRGSARRRNDRRAVSAVSCSDN>
27	Q00497 cTP 66	MARVSSSSWNSDKVVRKSGRSKWNKRHRVVVSCHRKAAHSDRRVKVSCS>
28	Q01908 cTP 50	MACSNTTMWVSSKSSADSSSSRSVKCTNTSSSRASSVSASRRDRDSVKN>
29	Q01909 cTP 60	MTGSSTSWSSSNSNSASSSSYSATKVRYYSTNRSSRSSRAGRRRDSVKN>
30	Q42687 cTP 33	MAAKSAGRAKASAVRAAKAGRRTVVVMARKNVSSYAKAVADKGKAVHAD>
31	Q02758 cTP 64	MASHTTASHSKHKTTNTRKNSSSTYSKKKKTKTRRSTGGAGARMSSAAG>

Table 2 Output of V-match command in R

	group	group_name	start	end	width
1	1	NA	1	3	3
2	25	NA	203	205	3
3	37	NA	180	182	3
4	41	NA	1	3	3
5	45	NA	170	172	3
6	70	NA	183	185	3
7	80	NA	19	21	3
8	84	NA	70	72	3
9	108	NA	100	102	3
10	109	NA	104	106	3
11	114	NA	268	270	3
12	115	NA	260	262	3
13	141	NA	1	3	3
14	176	NA	31	33	3
15	210	NA	341	343	3
16	212	NA	141	143	3
17	220	NA	303	305	3
18	248	NA	16	18	3
19	251	NA	394	396	3
20	267	NA	387	389	3
21	275	NA	54	56	3
22	353	NA	298	300	3
23	362	NA	136	138	3
24	384	NA	68	70	3
25	385	NA	79	81	3
26	404	NA	128	130	3
27	437	NA	33	35	3
28	438	NA	41	43	3
29	443	NA	253	255	3
30	446	NA	39	41	3

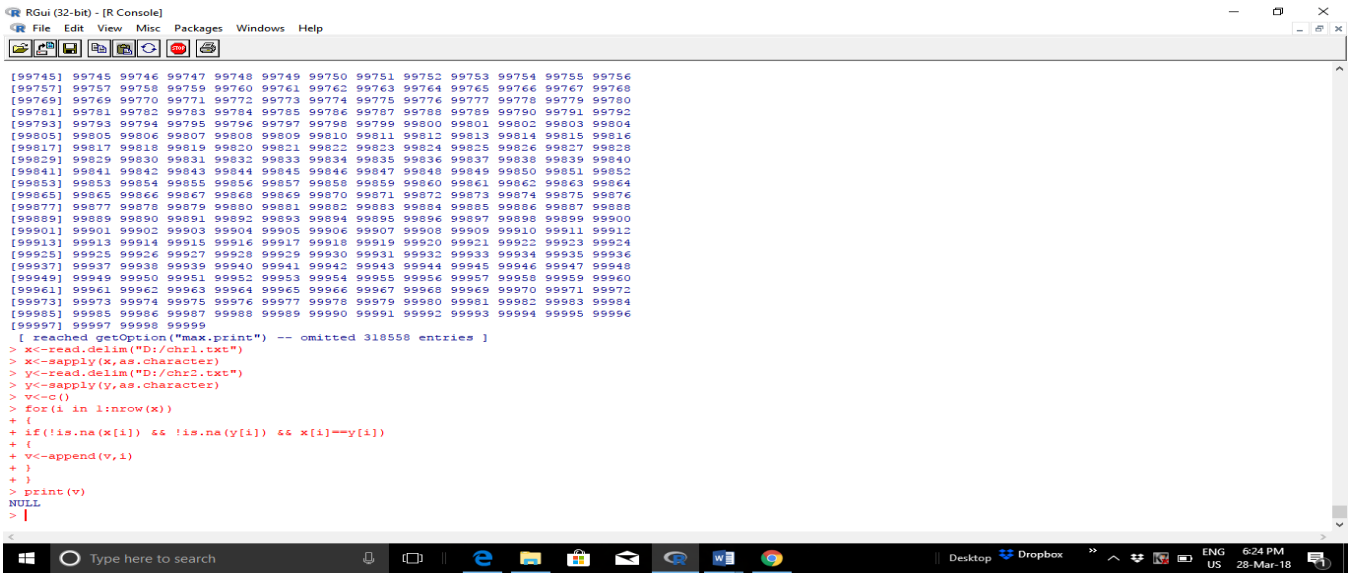

Fig.3: Block Indices of Dissimilar blocks

From Table 2, the frequency of sub sequences (within dissimilar blocks) is found out for applying Huffman coding. The size of compressed genomes is used as the parameter for the evaluating performance of proposed method. The block size of reference chromosome BSR is set to 4 MB for human genomes, 2 MB for TAIR genomes and rice genomes to use optimal number of bits for storing block indices. Otherwise, a big block index will result in poor compression ratio. It has also been observed that a block size of 12 bytes for BS1 speeds up the matching process thereby reducing the computation complexity. Moreover, as the dissimilar blocks are broken down into subsequences of minimum length three, BSI is set to multiple of three bytes.

The compressed stream will contain the matched block index of the reference genome for similar blocks and the Huffman codes for subsequences of dissimilar blocks. The size of each index vector bits is equal to the sum of numbers of bits required for storing the reference block index and the position within the reference blocks. Our experimental results show that the proposed method achieves a compression ratio 4:1 in compressing the human and TAIR genomes which is comparatively good. The use of Huffman coding for compressing subsequence accounts for enhancing the compression ratio. A good choice of BS_I and BS_R are crucial parameters for effectively compressing the genomes using the proposed method.

V. CONCLUSION

This paper proposes a new and simple method of compressing genomes based on referential compression by exploiting the huge similarities that exist between genomes of similar species using R programming tool. In future the compression ratio can be further enhanced by compressing block indices using differential coding schemes.

VI. REFERENCES

[1] S. D. Kahn. "*On the future of genomic data. Science (Washington)",*vol.331,pp.728–729, 2011.

[2] J. K. Bonfield and M. V. Mahoney, "Compression of FASTQ and SAM format sequencing data", PLoS ONE,vol.8, issue.3, 2013.

[3] S. Deorowicz, A. Danek, and M. Niemiec. Gdc , "*Compression of large collections of genomes*", arXiv preprint arXiv:1503.01624, 2015.

[4] Y. Zhang, L. Li, Y. Yang, X. Yang, S. He, and Z. Zhu. "*Light-weight reference-based compression of FASTQ data*", BMC bioinformatics, vol.16, issue.1, pp.188, 2015.

[5] E. S. Lander, et al., "*Initial sequencing and analysis of the human genome*", Nature, vol. 409, pp. 860-921, 2001.

[6] S. Kuruppu, S. J. Puglisi and J. Zobel, "*Optimized relative Lempel-Ziv compression of genomes*", Proceeding of ACSC 2011.

[7] P.SubrahmanyaandT.Berger, "*Asliding window Lempel-Zivalgorithm for differential layer encoding in progressive transmission*", Proc. IEEE Int. Symp. Inf. Theory, Whistler, BC, Canada, pp. 266,995, 1995.

[8] Kwang Su Jung, Nam Hee Yu, Seung Jung Shin, Keun Ho Ryu, "*A Compressing Method for Genome Sequence Cluster using Sequence Alignment*", 2008

[9] M.Mary Shanthi Rani, "A New Referential Method for Compressing Genomes" International Journal of Computational Bioinformatics and In Silico Modeling, Research Article Open Access, Vol. 4, issue.1, pp.592-596 2015.

[10] Biji CL and AchuthsankarS.Nair, "*Benchmark dataset for Whole Genome sequence compression*", pp.1545-5963, 2016.

[11] RabiaArshad,AdeelSaleem and Danista Khan, " *Performance Comparison of Huffman Coding and Double Huffman Coding* ",978-1-pp.5090-2000,2016.

[12] Komal Sharma, Kunal Gupta, "*Losseless Data Comperssion Techniques and Their Performance*", ieee, 2017.

[13] Kakoli Banerjee and A.Prasad, "*Reference Based Inter Chromosomal Similarity based DNA sequence Compression algorithm*", ,ieee, 2017

Authors Profile

Dr. M. Mary Shanthi Rani holds Ph.D in Computer Science and has more than 13 years of teaching experience .She has great passion for teaching and is currently working as Assistant Professor in the Department of Computer Science and Applications, Gandhigram Rural Institute (Deemed University), Gandhigram. She has nearly sixty two publications in International Journals and Conferences .Her research areas of interest are Image Compression, Information Security, Ontology, Biometrics and Computational Biology. She has served in various academic committees in designing the curriculum for B.Sc. and M.C.A courses as well. She has also served as reviewer of Peer-reviewed International Journals and Conferences. She is a Life member of Indian Society for Technical Education. She has the credit of being the Associate Project Director of UGC Indo-US 21st Knowledge Initiative Project.

Jegatheesh Chandra Bose S pursed Bachelor of Computer Science from M.S University College, SankaranKovil, India in 2014, and Master of Computer Science from Pondicherry University, Pondicherry, India in 2017. He is currently pursuing MPhil in Department of Computer Science and Applications, Gandhigram Rural Institute (Deemed to be University), India in 2018. Her main research work focuses on Medical Image Processing.

