Academy of Scientific Research and Technology
Faculty of Electronic Engineering, Menoufia Univ., Menouf. Egypt
27th National Radio Science Conference
16-18 March 2010

Genomic Sequences Differential Compression Model

H. Afify [1], M. Islam [2], M. Abdel Wahed [3], Y. M. Kadah [4]
Systems and Biomedical Engineering Department, Cairo University, Egypt,
[1] *hebaaffify@yahoo.com*
[3] *manalaw2003@yahoo.com*
[4] *ymk@k-space.org*

Abstract

In this paper, we exploit the differential compression model based on alignment of two similarity sequences or more, which compress one sequence comparing with another sequence and the renewal entropy estimation to improve the compression ratio. We determine the representative sequence from set of sequences. On the other hand, through our investigation we have found that the use of differential compression model of the genomic sequences could open new frontiers in quickly identifying unknown sequence related to set of sequences.

1. Introduction

The compression of genomic sequences remains a challenging problem, with profound implications in biology and with important technological impact when the use of genomic data will become a daily practice in health and medicine. As such, it will certainly be investigated further due to several reasons: benefits when storing and transmitting the genome files; possibilities for comparison of entire genomes by similarity metrics approximating Kolmogorov similarity [1], [2]; discovering statistically significant relationships among various sequences.

The amount of DNA being extracted from organisms and sequenced is increasing exponentially [3].This yields two problems: storage and comprehension. The problem of storage, while practical and useful to study, is depending on the size of each base. DNA is composed of four bases (A, T, G, C), and can be coded using two bits per base. According to functionality, DNA manifests different properties from other kinds of data. Compression algorithms for text or image files, exploit small repeated patterns and contextual similarities to achieve compression, these methods cannot be applied successfully to DNA. Since the repeated patterns in DNA sequences are typically much longer and less frequent, it is for this reason that traditional compression algorithm such as; the algorithm in Ziv and Lempel [4] performs poorly on DNA.

Despite the prevalence of broadband network connections — especially between universities and research centres — there still exists a need for compact representation of data to speed up transmission. Transferring a single sequence that is millions of characters long may take ten to fifteen minutes over a dial-up connection. In this area the compression of biological sequences may be useful.

Understanding genomic sequences has wide applications, from the synthesis of medicines to genetic screening and engineering. The structure of a sequence is important knowledge for its comprehension. If a sequence has a particular property that is shared by another sequence, it is possible that they are related in some way, or that knowledge that applies to one may also be useful for the other. Compression can help to show both the structure of a sequence and how it is related to other sequences.

In the last two decades, compression of genomic sequences can be divided into two categories: techniques developed for efficiently compressing sequence data for the sake of reduced resource consumption (disk space or network usage) [5]-[9]; and investigations of the usefulness of compressibility as a measure of information content, for the purpose of making inferences about sequences (such as the relatedness of two sequences) [1], [10].The results of compressing genomic sequences can be applied to the problem of evolution derivation [11]. Compression –based distance measures (CBMs) that depend on probabilistic mismatching [12], are not distinct enough among

different classes. However, researches have been suffering from the poor information to characterize the relatedness between sequences. To fill this gap, the differential compression based on sequence related to another sequence using alignment and entropy theory has been shown to be effective for compression of DNA sequences.

Researchers have worked in entropy estimation for biological sequences, either by computing frequency of *n*-mers for long enough inputs, called Shannon entropy [13], or by adopting compression methods to obtain an upper bound on entropy [5]. Loewenstern [14], [15] introduced a compression method CDNA by considering inexact match in finding patterns. Importantly, Lanctot and Yang [16] improved the compression further by exploiting the reverse complement property of DNA sequences. Also, this latter method produces a good estimation of entropy, e.g., the estimate approaches the actual entropy for long enough input. Badger and Chen [1] proposed a distance function with nice properties for cluster related sequences. While much research has been done on compressing individual DNA sequences, surprisingly little has focused on the compression of entire databases of such sequences. In this study we introduce the sequence database compression depending on the minimum entropy.

One of our goals is to calculate entropy in a useful way regarding two or more sequences and their corresponding species to serve in reduction compression ratio and to be used as criterion for identifying the representative reference sequence to other sequences. Also, the alignment of DNA sequences can give clues to common ancestry for discovering patterns and relationships between sequences that carry high information intervals to improve the compression ratio [17]. The proposed implementations were used to compute the differences between sequences by using Multiple Sequence Alignment MATLAB tool that makes all sequences with the same lengths and then encoding each sequence corresponding to the other sequence using 8 different op-codes that are based on similar, replace and insert operations. Minimum entropy was computed on op-codes files as an indicator to good compression. Finally, The Burrows-Wheeler transform (BWT) and Move-To-Front encoder [18] are used to rearrange op-codes files using a sorting algorithm, the output of that were compressed with arithmetic coding [19].

2. Material and Method

2.1. Data selection

The data set used in our model consists of 22 genomic sequences from two different classes: the human and mouse genome, downloaded from the GenBank database [20], these sequences are shown in Table 1. Sequence similarity is important as a method to infer how two sequences are related that help in sequences compression. The alignment-based comparison of biological sequences plays a fundamental role in most areas of compressed genomics. We have applied the following two methods to the encoded sequences in order to obtain their entropy estimates and compression ratio.

2.2. Compression sequence related to set of sequences

In this section we briefly demonstrate basic steps to determine the representative sequence of a set of sequences on the condition of the same class. We start first by aligning all the sequences using Multiple Sequence Alignment MATLAB tool to align between the 11-sequences of each classe. Some gaps may be inserted to a sequence to make all the sequences with same length. Then produce op-codes files by using the operation presented in Table 2, Table 3. Insert operation means insert the new base intervals of the next character in the reference sequence.

A good choice for a reference sequence is yielded by calculating the entropy of op-codes files that consist of symbols and determine the minimum entropy to recognize the reference sequence. Compression and entropy are inseparable. The entropy of an event *x* can be calculated by:

$$H(x) = -\sum_{i=1}^{n} p(i) \log_2 p(i) \quad (1)$$

Where *H(x)* is the entropy of *x* and *p(i)* is the probability of outcome *i* from a pool of *n* possible outcomes.

Table 2. Operation codes of alignments of DNA sequences

DNA sequences	Operation	Equivalent op-codes
The same base of given sequence	Similar operation	" 0 "
A ↔ T C ↔ G	Replace Complement	" 1 "
A ↔ G C ↔ T	Replace Diagonal	" 2 "
A↔C G ↔T	Replace Vertical	" 3 "
A or T or G or C↔ '-'	Replace with ' - '	" 4 "
A ↔ T C ↔ G	Insert complement	" 5 "
A ↔ G C ↔ T	Insert Diagonal	" 6 "
A↔C G ↔T	Insert Vertical	" 7 "
A ↔ A C ↔ C G ↔ G T ↔ T	Insert similarity = Duplicate	" 8 "

Table 1. DNA test sequences

Complete human mitochondrion genomes		Complete mouse genomes	
Sequence	Accession number	Sequence	Accession number
S1	EU849002	S11	FJ374658
S2	EU849091	S12	FJ374656
S3	EU825949	S13	FJ374665
S4	EU828774	S14	FJ374655
S5	EU828638	S15	FJ374657
S6	EU828637	S16	FJ374659
S7	EU742163	S17	FJ374600
S8	EU742162	S18	FJ374661
S9	EU742161	S19	FJ374662
S10	EU742160	S20	FJ374663
SH	NC_001807	SM	AB042432

Table 3. Example of using operation codes

	Insert operation	Replace operation	Similar operation
Given Sequence	A – T G	A C T G	A – T C
Other sequence	A – T G	A – G G	A C T T
op-codes	0 0 0 0	0 4 3 0	0 7 0 8

2.3. Compression data set based on selection of the representation sequence

The implementation of the proposed model is outlined as follows:

1. Calculate the entropy of each sequence related to other sequence based on op-codes files, then determine the summation entropy of each sequence and select the minimum one that refers to the representative sequence.

2. Determine the nearest sequence to the representative sequence by using the method presented in section 2.2 that makes the other reference. The representative sequence can be changed through the model; therefore, we need to arrange the reference by using the threading.

3. Finally, we have applied the Burrows – Wheeler Transform (BWT) that takes a block of data and rearranges it using a sorting algorithm. The resulting output block contains exactly the same data elements that it started with, differing only in their ordering. The transformation of a string $w = w[0…n-1]$ is defined as follows:

- Create a square matrix $M_{[n\,n]}$ in which the *kth* row contains the *kth* cyclic rotation of w
- Sort rows of M in lexicographic order
- Store the string represented by the last column of M, and the index of row which contains the position of the original string w (i.e., *0th* cyclic rotation)

BWT takes a file of n bytes and creates n permutations of the data by moving the first 1 to n bytes of the data to the end of the remaining data, in effect rotating the data. The n strings of n bytes each are then sorted, this groups similar contexts together, and the last byte of each string is the output data. Since similar contexts are grouped together, this sequence of " last characters" is highly compressible. To compress the data, the output of the transform is run through a Move-To-Front encoder. The choice of Move-To-Front (MTF) coding is important. While contexts are grouped together, there is no per-context statistical information kept, and so the encoder must rapidly adapt from the distribution of one context to the distribution of the next context. The MTF coder has precisely this rapidly adapting

quality. The last step is concerned with applying the arithmetic technique on operation code that converts a string into another representation that represents frequently used characters using fewer bits and infrequently used characters using more bits, with the goal of using fewer bits in total.

3. Results and discussion

We have applied Differential model to 22 Genome sequences encoded op-codes files from alignment method to determine minimum entropy that refers to the representative sequence for given sequence.

When the op-code files are ordered by Burrows – Wheeler Transform (BWT) which groups symbols with a similar context close together, the sorting is done in standard lexicographical order. BWT based on a permutation of the input sequence so the ordering could be saved in a few bits. Attempted improvements on BWT that followed by Move-To-Front (MTF) coding have shown success in our model. The most prominent improvements come from the extensive study done by Peter Fenwick [21] to improve the compression performance.

Table 4 and Table 5 show the entropy of op-codes files to recognize the minimum one. The sequence which has minimum entropy must be close to the given sequence and produces a good estimation of compression and minimum compression ratio (Bites/Base) after applying BWT and arithmetic techniques. The results confirm that S5 is reference to SH for human data and S13, S17, S20 are references to SM for mouse data. Compression ratio of SH depending on S5 equals **0.1527** Bites/Base and compression ratio of SM depending on S13 or S17 or S20 equals **0.0417** Bites/Base.

We have applied the differential model to two classes (10 Human Mitochondria sequences and 10 Mouse sequences) to find the reference sequence of each class. The results of human sequences confirm that S10 is reference where the minimum summation entropy of S10 is 0.1969. We describe these sequences (S8, S9, S1) by S10 because the entropy of S8-S10 equals 0.0017, S9-S10 equals 0.0009, and entropy of S1-S10 equals 0.0295. Sequences (S2, S3) are close to sequence S1 and can be deduced using S1. Similarly sequences (S5, S6) can be deduced using S4. Moreover, S4 close to S8 and S7 close to S9. On the other hand, the results of mouse sequences confirm that S20, S13, S17 are references. So we can select any one from these references to describe S11, S12, S14, S15but we can use S14 as reference to S16 and S18. We also use S11 or S15 as reference to S19. Results of minimum entropy and compression ratio for human and mouse sequences are shown in Table 6 and Table 7 respectively. Reconstructed phylogenic trees based on entropy of each class under study are shown in Fig.1 and Fig.2.

Table 4. The entropy and compression ratio of differential sequence for human genomic sequences

Differential sequences	Entropy	Compression ratio
SH-S1	0.0272	0.1719
SH-S2	0.0278	0.1784
SH-S3	0.0257	0.1675
SH-S4	0.0159	0.1619
SH-S5	**0.0154**	**0.1527**
SH-S6	0.0231	0.1659
SH-S7	0.0278	0.1674
SH-S8	0.0252	0.1652
SH-S9	0.0243	0.1672
SH-S10	0.0249	0.1644

Table 5. The entropy and compression ratio of differential sequence for mouse genomic sequences

Differential sequences	Entropy	Compression ratio
SM-S11	0.0009	0.0421
SM-S12	0.0019	0.0453
SM-S13	**0.0005**	**0.0417**
SM-S14	0.0009	0.0421
SM-S15	0.0009	0.0421
SM-S16	0.0009	0.0421
SM-S17	**0.0005**	**0.0417**
SM-S18	0.0019	0.0453
SM-S19	0.0018	0.0451
SM-S20	**0.0005**	**0.0417**

Table 6. The minimum entropy and compression ratio of differential sequence for human genomic sequences

Differential sequences	Minimum Entropy	Compression ratio
S9-S10	0.0009	0.0421
S8-S10	0.0017	0.0449
S1-S10	0.0295	0.0797
S2-S1	0.0135	0.1534
S3-S1	0.0145	0.1558
S4-S8	0.0266	0.1717
S5-S4	0.0257	0.1752
S6-S4	0.0270	0.1759
S7-S9	0.0050	0.0762
Average		**0.1194**

Table 7. The minimum entropy and compression ratio of differential sequence for mouse genomic sequences

Differential sequences	Minimum Entropy	Compression ratio
S13-S20	0.0009	0.0421
S17-S20	0.0009	0.0421
S11-S20	0.0017	0.0449
S12-S20	0.0017	0.0449
S14-S20	0.0009	0.0421
S15-S20	0.0009	0.0421
S16-S14	0.0009	0.0421
S18-S14	0.0009	0.0421
S19-S11	0.0009	0.0421
Average		**0.0472**

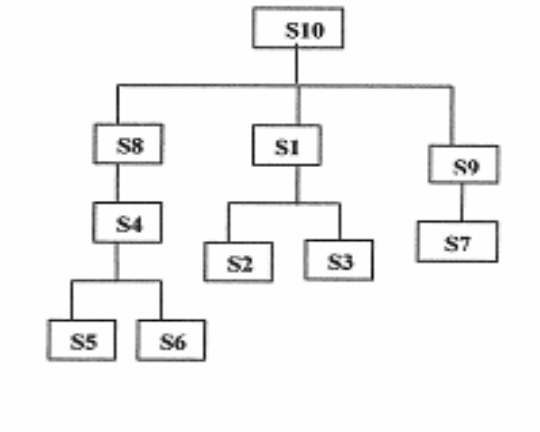

Figure 1: The phylogenic tree of human genomic sequences.

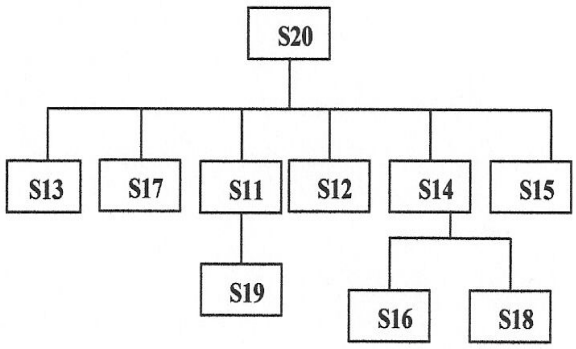

Figure 2: The phylogenic tree of mouse genomic sequences.

By using reference sequence for each model organism where sufficient data is available, we can use this reference sequence to considerably compress the remaining DNA sequences of this model organism.

In this work, we compressed each of ten different DNA sequences of complete human mitochondria genomes by average compression ratio equals to **0.1194** Bites/Base and for mouse genomes by average compression ratio equals to **0.0427** Bites/Base without sending the reference sequence. The execution time of the differential compression is depending on the length of the sequence needed to be compressed. As the sequence length increases the execution time increases.

4. Conclusion

This paper advocates that differential model can be used in DNA compression based on description and recognition of the reference sequence that helps to improve the compression ratio.

We have presented an efficient algorithm for DNA compression based on the relation between two similarity sequences by sequence alignment and minimum entropy. Furthermore, BWT that followed by MTF coding are used to improve the compression performance combined with high speed. For a pair of sequences, if one sequence is used as training data for compressing the other sequence, the rate of compression achieved is some measure of the relatedness between them. In other words, it is possible to measure how much information one sequence gives about the other.

We proved that the suitability of our model can be measured by compression. A model that is able to predict accurately the sequence given the representative would achieve excellent compression. A model that made poor predictions would not do as well. Thus, the compression rate can be used to evaluate a new model. This is of prime importance for genomic sequences; a model of the composition of such sequences would help in their understanding.

In conclusion, the differential compression as presented leaves some avenues explored. Firstly, sequences that belong to the same "class" can be expected to obtain roughly the representative sequence based on the minimum entropy. Secondly, applying the information from the representative sequence can predict the compression of another sequence under our model. Based on efficient algorithm for compression, differential compression should extend to classification of new sequences related to the representative of a set and the similarity between sequences.

.

References

[1] M. Li, J. Badger, X. Chen, S. Kwong, P. Kearney and H. Zhang, "An information based sequence distance and its application to whole mitochondrial genome," *Bioinformatics,* vol. 17, no. 2, pp. 149-154, 2001.

[2] M. Li, X. Chen, X. Li, B. Ma, and P. Vitanyi, "The Similarity Metric," *Proc. 14th Ann. ACM-SIAM Symp. Discrete Algorithms*, pp. 863-872, 2003.

[3] L. Rowen, G. Mahairas and L. Hood, "Sequencing the Human Genome," *Science,* vol. 278, pp. 605-607, Oct. 1997.

[4] J. Ziv and A. Lempel, "A Universal Algorithm for Sequential Data Compression," *IEEE Trans. Information Theory*, vol. IT-23, pp. 337-343, May 1977.

[5] S. Grumbach and F. Tahi, "Compression of DNA Sequences," *Proc. IEEE Symp. Data Compression*, Snowbird, UT, pp. 340-350, 1993, doi:10.1109/DCC.1993.253115.

[6] S. Grumbach and F. Tahi, "A New Challenge for Compression Algorithms: Genetic Sequences," *Information Processing Management,* vol. 30, no. 6, pp. 875-886, 1994.

[7] X. Chen, S. Kwong and M. Li, "A Compression Algorithm for DNA Sequences and its Applications in Genome Comparison," *Proc. 4th Ann. International Conf. Computational Molecular Biology*, pp. 107, 2000.

[8] X. Chen, M. Li, B. Ma and J. Tromp, "DNACompress: Fast and Effective DNA Sequence Compression," *Bioinformatics,* vol. 18, no. 12, pp. 1696-1698, 2002.

[9] M. D. Cao, T. I. Dix and L. Allison, "A Simple Statistical Algorithm for Biological Sequence Compression," *Proc. IEEE Data Compression Conference (DCC '07)*, pp. 43-52, March 2007, doi: 10.1109/DCC.2007.7.

[10] A. Kocsor, A. Kertesz-Farkas, L. Kajan and S. Pongor, "Application of Compression-based Distance Measures to Protein Sequence Classification: A Methodology Study," *Bioinformatics,* vol. 22, pp. 407-412, 2006.

[11] A. Hatengan and I. Tabus, "Protien is Compressible," *Proc. 6th Nordic Signal Processing Symposium*, pp. 192-195, 2004.

[12] R. Cilibrasi and P. M. B. Vitanyi, "Clustring by Compression," *IEEE Trans. Information Theory*, vol. 51, no. 4, pp. 1523-1545, April 2005, doi: 10.1109/TIT.2005.844059.

[13] C. E. Shannon, "A Mathematical Theory of Communications," *Bell System Technology Journal,* vol. 27, pp. 379-423, 1948.

[14] D. M. Loewenstern, H. Hirsh, P. Yianilos and M. Noordewier, "DNA Sequence Classification Using Compression-based Induction," Technical Report 95-04, DIMACS, 1995.

[15] D. M. Loewenstern and P. N. Yianilos, "Significantly Lower Entropy Estimates for Natural DNA Sequences," *Proc. IEEE Data Compression Conference (DCC '97)*, pp. 151-160, 1997, doi:10.1109/DCC.1997.581998.

[16] J. K. Lanctot, M. Li and E. H. Yang, "Estimating DNA Sequence Entropy," *Proc. 11th Annul. ACM-SIAM Symp. Discrete Algorithms*, pp. 409-418, 2000.

[17] S. F. Altschul, W. Gish, W. Miller, E. W. Myers and D. J. Lipman, "Basic Local Alignment Search," *J. Molecular Biology,* 215, pp. 403-410, 1990.

[18] M. Burroes and D. J. Wheeler, "A Block Sorting Lossless Data Compression Algorithm," Technical Report, Digital System Research Center, 1994.

[19] I. H. Witten, R. M. Neal and J. G. Cleary, "Arithmetic Coding for Data Compression," *Communications of the ACM,* vol. 30, no. 6, pp. 520-540, June 1987.

[20] National Center for Biotechnology Information, http://preview.ncbi.nlm.nih.gov/guide. (Sited on 11 Oct. 2009).

[21] P. Fenwick, "Block Sorting Text Compression | Final Report," Technical Report 130, The University of Auckland Department of Computer Science, March 1996.