Genome Sequence compression algorithm based on the Distributed source coding

Jing-Jing Shao[1]

[1]College of science and technology, DianChi College of Yunnan University, Kunming, 652008, China

email: 153818059@qq.com, Contact email:18459423@qq.com

Keywords: Distributed source coding; Genome sequence compression; Side information; Context weighting; Hilbert spacing filling

Abstract. The genome sequence compression algorithm based on the distributed source coding technology purely is proposed in this paper. In order to enhance the compression efficiency, the genome sequence is mapped into two binary sources and then they are transmitted into two bilevel images. After initialization, the distributed source coding based on LDPC is constructed for compressing these two sequences. To compress the side information, the optimized context weighting is suggested. The experiments results indicate that the coding efficiency is better than results from any other compression algorithms for microbial genome sequence compression.

Introduction

With the development of the rapid genome sequencing technology, the number of genome sequence become larger and larger, which leads to the difficulty to store these data. Therefore, the compression strategy can be considered to console this conflict. In the past decades, a lot of types of compression algorithms were proposed to enhance the compression efficiency of genome sequences. Generally speaking, there are three types of compression algorithms which are used to compress various genome sequences. They are Lz77 based algorithm, entropy coding scheme and the reference algorithm.

The first compression algorithm for genome sequence is BioCompress and its improved methos[1,2], which is from the Lz77. Then the similar algorithms BioCompress2 [3,4] are proposed to resolve the method to encode those un-repetitive sequences. However, these Lz77 based algorithms rely on the vast repetitive sequence existing. For these sequences which contain small scale of repetitive fragments, this type algorithm can not obtain better performance. On the other side, the reference algorithms[5,6] attract the interesting of researchers, which reason to the high compression rate of this type algorithm. In [5], the compression rate can be 80 when the human genome sequence are encoded by using the corresponding algorithm. Actually, the radio of the repetitive sequence contained in the human genome sequence can reach to 90%, which ensure the radio that the fragment needed to be encoded can mapping the existing reference can maintain in a high level. However, for these genome sequences with low repetitive fragments , such as the microbial genome sequence, this type algorithm is not suitable. Actually, for these genome sequences, the context based entropy coding scheme[6-10] is more powerful than other types of algorithms.

On the other hands, with the development of the internet of things, some mobile equipment is designed to obtain data more easily. For these equipment, one problem needed to be resolved is that the cost for transmitting data. The distributed source coding is one efficient coding scheme to balance the conflict between coding and cost. Resent years, more and more corresponding researches aiming to this topic. Such as [11,12]. Especially in [12], the distributed source coding is suggested to compress the genome sequence. However, it is just the complementary of those reference type compression algorithms, which can not achieve better performance for compression.

For inspiring, in this paper, the genome sequence compression based on the distributed source coding is proposed. As the improvement, the distributed source coding is employed to compress only one sequence once. In order to obtain the side information, the genome sequence is mapped

into two bilevel images to ensure the distributed source coding can be executed.

Methods

In [12], the distributed source coding is employed to compress the genome sequences, which only the human genome sequences compression is discussed. Meanwhile, its application is limited into the compression for those independent bases. Actually, the correlation among bases is weaken. For microbial genome sequence, due to the little repetitive fragments, whatever the decoder is improved, the total compression efficiency can not be enhanced more.

Let X and Y denote the genome sequence and the reference respectively. For the distributed source coding, the correlative entropy $H(X,Y)$ should be transmitted. There are:

$$H(X,Y) = H(X) + H(Y \mid X) \tag{1}$$

If the reference sequence Y is existing, the cost for coding X can only be $H(X \mid Y)$, but not $H(X)$. It implies that the sequence X is compressed. In this case, the $H(Y)$ is referred to as the side information. If the correlation between X and Y is strong, the compression efficiency can obtain a high rate. However, in the microbial genome sequence compression, each two genome sequence maintain low correlation, which means that the fact that the reference is Y can not be suitable.

In order to resolve the problem of correlation, in this paper, we give a method to map one genome sequence into two 2-D bilevel images. In the first step, four types of bases are mapped into two values. The mapping rule is illustrated in (2)

$$\begin{cases} A \\ T \end{cases} \to A \quad , \quad \begin{cases} G \\ C \end{cases} \to G \tag{2}$$

In this case, there are only two values contained in a genome sequence, A and G. Then another binary sequence is constructed to illustrate the state that one base is itself or being mapped. It is given in (3)

$$a = \begin{cases} 0, & if\ A \to A \\ 1, & if\ T \to A \end{cases} \tag{3}$$

Then the case G and C can also be determined similarly with (2) and (3). By using formula (2) and (3), one genome sequence is mapped into two binary source. Meanwhile, these two sequences contain high correlation. Furthermore, in order to utilize more correlations, these two binary sources are mapped into two 2-D images respectively. The Hilbert space filling curve is suggested to implement this mapping operation. We can use the filling matrix in (4) to calculate the filling results and when each item in these two binary sources is filled into its corresponding location in the image, the mapping operation is finished.

$$\begin{cases} \begin{bmatrix} H_{2^k} & 4^k E_{2^k} + H_{2^k} \\ 4^{(k+1)} E_{2^k} - H_{2^k} & (3 \times 4^k + 1) E_{2^k} - (H_{2^k})^T \end{bmatrix} & k \text{ is even} \\ \begin{bmatrix} H_{2^k} & (4^{k+1} + 1) E_{2^k} - H_{2^k} \\ 4^k E_{2^k} + H_{2^k}^T & (3 \times 4^k + 1) E_{2^k} - (H_{2^k})^T \end{bmatrix} & k \text{ is odd} \end{cases} \tag{4}$$

By using (4) with iterations, two binary images can be obtained. Then the distributed source coding scheme is employed to compress these two images.

In this paper, the distributed source encoder with side information is suggested to implement the coding. The LDPC is used to implement the distributed source coding.

Encoder:

The encoder in the distributed coding scheme is easy, which aims to reduce the coding complexity. Let X and Y denote two images and Y is the side information. For a given block code (n,k), its each corresponding codeword c_n can be determined by calculating the formula (5)

$$C_n = \vec{G} \bullet C_k \tag{5}$$

Where C_n denote the set of all codeword and $\vec{G}$ is the generated matrix. Actually, for genome sequence, each protein is consist of three bases. It implies that there are 64 possible combinations of bases. However, there are not 64 proteins in practice. Let m denote the number of existing proteins. When one genome sequence is mapped into two images, it can be considered as a block code $(3,\ \log_2\sqrt{m})$. Theoretically, its corresponding compression rate η can reach to (6)

$$\eta = \frac{\log_2\sqrt{m}}{3}, \quad m \le 64 \tag{6}$$

For this block code, whatever CRC or LDPC can be used to implement it easily.

Meanwhile, in order to enhance the compression efficiency ulteriorly. The side information is also compressed by using the context based entropy coding scheme. Due to the binary source, or bilevel images, the optimized context weighting entropy coding is employed. Three context models with different orders are constructed. The context templates are given in table 1:

Table 1: The context templates used in this paper

models	order	contexts
Context model 1	5	XXXXX?
Context model 2	8	XXXXXXXX?
Context model 3	3	XOOXX?

The objective of the context weighting is to minimize the description length of the weighted model. The optimized weights can be determined by resolving the equations (7) and (8)

$$L = \sum_{i=1}^{N} w_i L_i \tag{7}$$

And the group equations

$$\begin{cases} \dfrac{\partial f(W)}{\partial w_i} = 0, \quad i = 1,2,\ldots,N \\ \sum\limits_{i=1}^{N} w_i = 1 \end{cases} \tag{8}$$

Where N denotes the number of context models joining in weighting and L_i and w_i denote the description length of the model i and its corresponding weight.

Here, the side information is compressed and the corresponding other information can be encoded by using the proposed algorithm above. In next section, some experiments are given to testify the compression efficiency of the proposed algorithm

Experiments and Results

The proposed algorithm is employed to compress some microbial genome sequences, which come from NCBI[14]. They are:

HEHCMVCG: Human cytomegalovirus strain AD169 complete genome

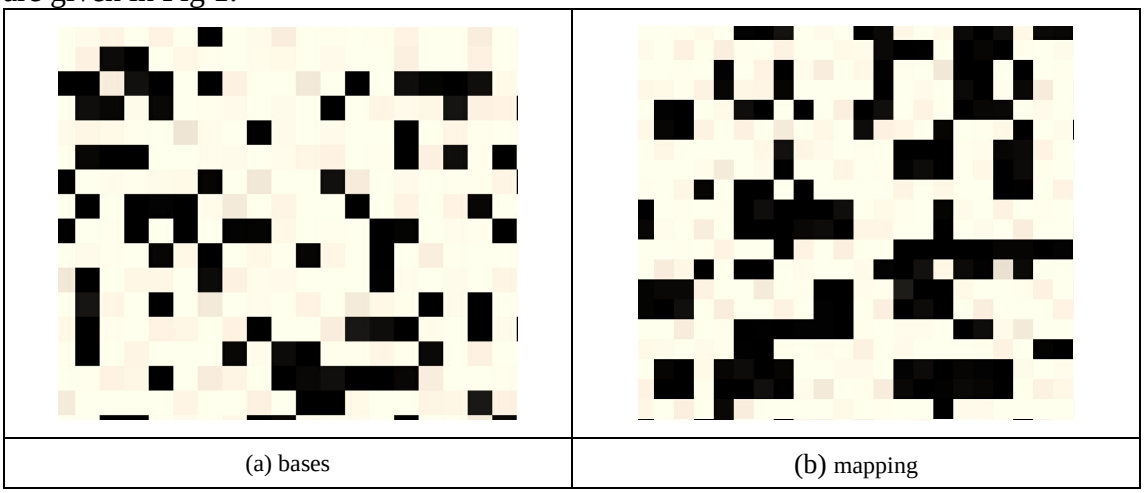
CHMPXX: Marchantia polymorpha chloroplast genome DNA

CHNTXX: Nicotiana tabacum chloroplast genome DNA

VACCG，Vaccinia virus, complete genome.

To illustrate the Hilbert mapping, the sequence VACCG is changed and mapped into 2-D images.

They are given in Fig 1:



(a) bases (b) mapping

Fig 1: The 2-D images of one genome sequence

Then the proposed algorithm is used to compress these two bilevel images. In order to represent the compression rate of out algorithm, the results of us are used to compare with those results from other algorithms. These algorithms are: BioC[2], GenC[3], DNAP[4], GeMNL[5], XM[13]. The compared results are listed in table 2:

Table 2: The comparison of results from various algorithms

Sequences	BioC	GenC	DNAC	DNAP	GeMNL	XM	proposed
HEHCMVCG	1.8480	1.8470	1.8492	1.8346	1.8420	1.8426	1.6233
CHMPXX	1.6848	1.673	1.6716	1.6602	1.6617	1.6577	1.4338
CHNTXX	1.6172	1.6146	1.6127	1.6103	1.6101	1.6068	1.4189
VACCG	1.7614	1.7614	1.7580	1.7583	1.7644	1.7649	1.4241

It is obviously that the proposed algorithm can achieve better results than any other existing algorithms. However, the complexity of our algorithm is also higher than other algorithms, which is on the reason that two aspects. One is that our algorithm should encode two images with different coding scheme. Another is that the decoding cost is higher than other algorithms. However, the improvement on the compression efficiency is more valuable than the increment of cost.

Conclusion

In this paper, the genome sequence compression algorithm based on the distributed source coding technology purely is proposed. In order to enhance the compression efficiency, the genome sequence is mapped into two binary sources and then they are transmitted into two bilevel images. After initialization, the distributed source coding based on LDPC is constructed for compressing these two sequences. To compress the side information, the optimized context weighting is suggested. The experiments results indicate that the coding efficiency is better than results from any other compression algorithms for microbial genome sequence compression.

Acknowledgement

This work is supported by the Foundation of Instruction Science of Yunnan Province under Grant2014y634.

References

[1]GRUMBACH S,TAHI F. Compression of DNA sequences[C] Proc Data Compression Conference．1993:340-350.

[2]GRUMBACH S,TAHI F. A new challenge for compression algorithms:Genetic sequences[J]．Information Processing & Management,1994,30(6):875-866.

[3]RIVALS E,DELAHAYE J P,DAUCHET M,et al．A guaranteed compression scheme for repetitive DNA sequences[C]//Proc Data Compression Conference．1996:453-471.

[4]CHEN X,S. KWONG S,LI M．A compression algorithm for DNA sequences and its applicationsin genome comparison[C]// Proceedings of the fourth annual international conference on Computational molecular biology．New York:NY,2000:107.

[5]CHEN X,LI M,MA B,et al． DNACompress: Fast and effective DNA sequence compression, Bioinformatics[J]．2002,18(2):1696-1698.

[6]BEHZADI B,FESSANT F L．DNA compression challenge revisited: A dynamic programming approach[J]．Combinatorial Pattern Matching,2005,353:190-200.

[7]MATSUMOTO T,SADAKANE K,IMAI H. Biological sequence compression algorithms[J]．GenomeInformatics,2000,11:43-52.

[8]TABUS I,KORODI G,RISSANEN J．DNA sequence compression using the normalized maxi-mum likelihood model for discrete regression [C]//Proc Data Compression Conference．2003:253-263.

[9]KORODI G,TABUS I．An efficient normalized maximum likelihood algorithm for DNA sequence compression[J]．ACM Trans Inf Syst,2005,23(1):3-34.

[10]SOLIMAN T H A．A lossless compression algorithm for DNA sequence[J]．J Bioinformatics Research and Application,2009,5(6):593-602.

[11]PRADHAN S S,RAMCHANDRAN K．Distributed source coding using syndromes (DISCUS): Design and construction[J]．IEEE Trans Inform Theor,2003,49(3):626-643.

[12]WANG Shuang,JIANG Xiao-qian,CHI Li-juan,et al．Genome sequence compression with distributed source coding[J]．//Proc Data Compression Conference．2013:525-572.

[13] CAO M D,DIX T I,ALLISON L,et al．A simple statistical algorithm for biological sequence compression[C]//Proc Data Compression Conference.2007:43-52.

[14]NCBI. Center for biotechnology information [EB/OL]. [2014-03-25]. http://www.ncbi.nih.gov/genomes/Bacteria/