DNA-RESIDUAL: A DNA COMPRESSION ALGORITHM USING FORWARD LINEAR PREDICTION

Rony Ferzli and Lina J. Karam
Department of Electrical Engineering
Arizona State University
Tempe AZ 85287-5706
{rony.ferzli, karam}@asu.edu

ABSTRACT

This paper presents an efficient lossless DNA compression algorithm, DNA-Residual, that significantly decreases the average bit-rate required to losslessly code correlated DNA sequences. The algorithm can be divided into two parts: modeling and coding. The modeling part consists of mapping the DNA bases into a binary representation and, then, a forward linear prediction filter is used to predict the current input from the previous ones. The prediction error is then transformed into a binary error sequence that is coded using an adaptive binary arithmetic coder. Compared to state-of-the-art compressors using benchmark DNA sequences, the proposed algorithm reveals a significantly higher compression ratio whenever correlation between bases is high.

1. INTRODUCTION

Recently, and with the completion of the human genome, compression of DNA sequences gained considerable research interest. DNA sequences can contain millions of nucleotides, thus, the need arises for an efficient lossless DNA compression algorithm. Although the storage media increased in size considerably, there is always a need for compression for lowering the cost of storing and maintaining large databases. In addition, DNA sequences are typically shared among different servers and information is retrieved using the Internet and other bandwith-limited transmission media. This necessitates the use of compression in order to reduce the needed transmission time and bandwidth.

DNA sequences contain only four bases {A, C, T, G}, thus, at most 2 bits are needed to represent each base. However, the standard text compression algorithms cannot compress the base below 2 bits. In fact, the size of the files encoded with standard compression algorithms yield a compression higher than 2 bits/base [1]. Most of the general purpose compressors are tailored towards text compression focusing on the redundancies of exact replicas and on the predictability of the current symbol based on the previous context.

It can be shown that the DNA sequences are not random sequences [2], but sequences where approximate repeats (repeats having one or several mutated bases) and complementary palindromes (reversed repeats where nucleotides are replaced by their complementary bases) can be detected. Based on these characteristics, several algorithms [1, 3-10] have been proposed resulting in a higher compression ratio than standard algorithms. Though these proposed algorithms try to take advantage of DNA sequence characteristics, their compression ratio is not satisfactory giving, on average, a compression ratio around 1.6 bits/base. Since the DNA sequence is a discrete one, information coding and digital signal techniques can be applied. The proposed algorithm, DNA-Residual, uses forward linear prediction and entropy coding to encode the DNA residual or prediction error.

This paper is organized as follows. Section 2 presents an overview of available algorithms for DNA sequence compression. The proposed algorithm based on forward linear prediction is presented in Section 3. Section 4 compares the performance of the state-of-the-art existing DNA compression algorithms with the proposed one.

2. EXISTING DNA COMPRESSION ALGORITHMS

This section provides an overview of existing popular DNA compression algorithms.

- BioCompress [2]: is the first algorithm dedicated exclusively to DNA sequences. It uses a Lempel-Ziv substitution algorithm detecting exact repeats and complementary palindromes. Palindromes matches are as frequent as direct matches and redundancy is exploited. Later on, an improved version was introduced, BioCompress-2 [3], using a context-based arithmetic coder (order-2) whenever repetition is not found.
- GenCompress [4]: introduces the idea of approximate

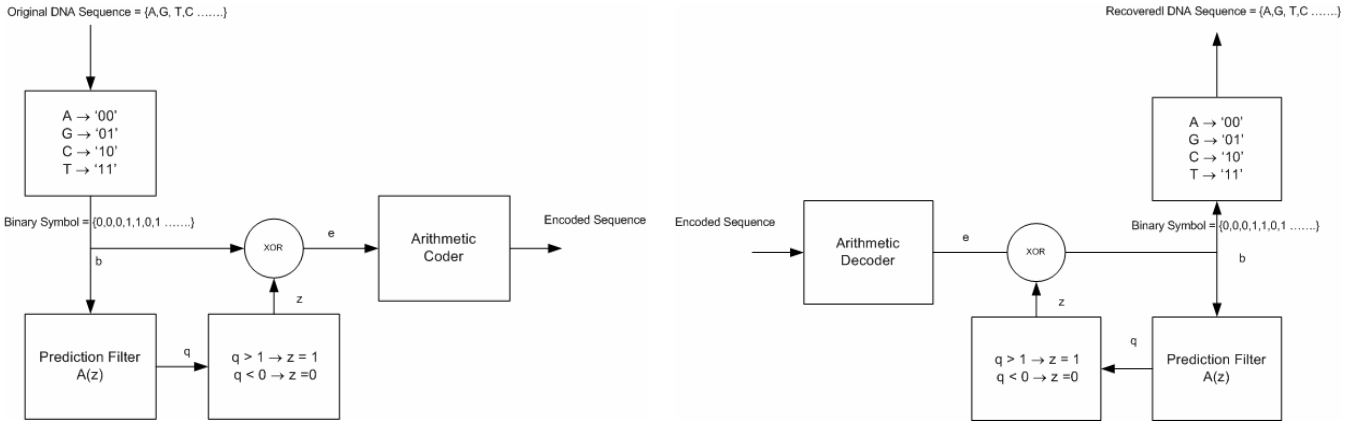

Figure 1. Block Diagram of the proposed DNA-Residual algorithm: (a) Encoder, (b) Decoder.

matches where it searches for approximate repeats with replacement operations using the Hamming distance. The algorithm searches for approximate repeats or approximate reverse complements, and encodes these as length, position and differences using Lempel-Ziv. If an approximate repeat or an approximate reverse complement contains many differences, this algorithm does not provide gain in the encoding. In order to increase the coding gain, GenCompress uses a second-order arithmetic coder. An improved version, GenCompress–2 [5], uses an edition operation for the replacement (deletion, insertion and substitution).

- DNACompress [6]: employs the Lempe-Ziv scheme for exact and approximate repeats. The repeats are detected using a search algorithm, PatternHunter, employing non consecutive symbols as seeds. Non-repeat regions are encoded using a context-based arithmetic coder.
- DNAPack [7]: searches for approximate matches similar to the approach adopted in the preceding algorithms. The difference lies in the detection of the approximate repeats using dynamic programming, increasing thus the probability of detecting the longest approximate matching pattern rather than, for example, detecting the first occurring one.
- CTW-LZ [8]: is a combination of the context-tree weighting method (CTW) and the Lempel-Ziv scheme. The long exact or approximate matches are encoded using CTW, while short sequences are encoded using Lempel-Ziv. Searching for approximate repeats or approximate reverse complements, is performed using dynamic programming.
- GeNML [9]: takes a different approach by using a combination of encoding schemes. The DNA sequence is divided into fixed length blocks. The first scheme uses a reference to a previously encoded segment for conditionally encoding the current block using a Normalized Maximum Likelihood (NML) model. A discrete regression model is obtained by detecting hidden regularities which are considered as the approximate matches. The second scheme uses a context-based arithmetic coder for non-regular block.

Other DNA compression algorithms also exist but are less popular due to their high complexity or low compression ratio [10]. It is worth noting that since the appearance of the first dedicated DNA compressor by Grumbach and Tahi [1] till the latest proposed one by Korodi and Tabus [9], the reduction in bit-rate is less than 0.02 bits/base.

3. PROPOSED ALGORITHM

A block diagram of the proposed coding algorithm is shown in Fig. 1. The encoding process (Fig. 1(a)) can be summarized as follows. To apply DSP techniques to DNA sequences, each DNA base is first assigned a numerical value. The four bases A, G, C, and T are mapped to 00, 01, 10, and 11, respectively, maintaining, thus, the base characteristics of having {A,T} and {G,C} as complements [2]. Note that the resulting binary sequence ‘b’ is regarded as stationary (or at least, quasi-stationary), which is the underlying assumption in the prediction and compression process.

The linear prediction filtering attempts to remove the redundancy from the DNA bit stream ‘b’. Note that the resulting prediction filter output ‘q’ is a multi-bit value and is thresholded, with respect to zero, to obtain a binary sequence ‘z’. As indicated in Fig. 1(a), the error signal ‘e’ is calculated by an exclusive-or (XOR) operation between the original DNA bit-stream, ‘b’, and the thresholded filter output,‘z’. If the prediction is successful (so as to create as many zeroes in ‘e’ as possible), the probability of false prediction P(e = 1) will be low, enabling significant data reductions by entropy encoding. The prediction error ‘e’ is encoded using adaptive binary arithmetic coding. Note that along the encoded bits from

the arithmetic coder, the coefficients of the prediction filter need also to be transmitted for each encoded input sequence. The decoder (Fig. 1(b)) performs the inverse operations of the encoder, recovering the binary sequence 'b' and reverse mapping the bits to DNA bases. More details about the linear prediction filter and the adaptive arithmetic coding blocks are provided below.

3.1. Linear Prediction Filter

Correlation functions have been widely used to study the statistical properties of DNA sequences [2, 11, 12]. Linear prediction and autocorrelation-based techniques were used to extract a feature vector from different regions of the gene identifying coding and non coding regions [2]. In addition, the prediction error was used as a similarity measure between different genes. For example, the coefficients for the prediction filter of gene 1 were obtained, and an analysis was performed using the same model coefficients that are obtained for gene 1, but applied to the other genes. The variance of the residual error was used for similarity check; the smaller the variance, the higher the similarity. These ideas are borrowed from speech processing algorithms where the residual error is used in speech to differentiate between voiced and unvoiced frames. Recently, linear prediction was used by Philips in the Super Audio CD Standard to losslessly compress oversampled speech or audio signals [13]. A similar idea is adopted here to compress DNA sequences. Comparing the audio signal to DNA sequences, the correlation of the oversampled audio signal is much higher giving a better prediction and thus, a higher compression ratio. But due to the continuous increase in the deciphered genes where billions of bases need to be stored, any reduction in bit-rate, as compared to the existing DNA compression algorithms, is significant.

Though there is no causality constraint on DNA sequences, only forward linear prediction filters will be used (depending only on the previous input samples), since the start/stop codons and the transcription of the nucleotide triplets implicitly give directionality to the nucleotide sequences in the genes. The transfer function of the forward linear prediction filter is given by:

$$A(z) = \sum_{i=1}^{P} a_i z^{-i} \tag{1}$$

where $a_1, a_2, \ldots, a_P$ are the linear prediction coefficients. The FIR prediction filter coefficients can be obtained using standard methods. The AR Yule-Walker and Burg algorithms based on the autocorrelation matrix are widely used to compute the filter coefficients. The involved autocorrelation matrix values are typically calculated using the biased estimate version given by [14]:

$$R_{bb}(m) = \frac{1}{N} \sum_{n=0}^{N-|m|-1} b(n+|m|)b(n) \tag{2}$$

where 'b' is the input, 'm' is the distance or shift between the sequence, and 'N' the total number of considered samples. Note for large sequences, framing may be used to reduce computation complexity.

In our case, a prediction filter of order $P = 10$ was found to provide a good performance in modeling the correlation of the DNA sequences. Adaptive linear prediction is used by computing the optimal prediction coefficients for each input DNA sequence to be coded. For each DNA sequence, the computed optimal coefficients are sent to the decoder as part of the bit-stream by using 16 bits to represent each coefficient.

3.2. Adaptive Arithmetic Coding

It is well known, from information theory, that a source with entropy H requires, on average, only H bits to represent each of its symbols losslessly. The entropy is the theoretical limit for any compression algorithm. For a binary source, the first-order entropy can be expressed as:

$$H = -p_0 \log_2 p_0 - p_1 \log_2 p_1 \tag{3}$$

where p_0 and p_1 are the probabilities of binary symbols '0 and '1', respectively. From (3), we can infer that as the probability of one of the symbols increases with respect to the other one, the entropy will decrease. So, a well-designed prediction filter (Section 3.1) should result in a large number of zeros and a small number of ones in the binary error sequence 'e'. Consequently, this would result in a low-entropy error sequence that can be efficiently compressed using entropy coding schemes. In our case, adaptive binary arithmetic coding [15] is adopted for coding the resulting binary error sequence 'e'.

4. SIMULATION RESULTS

In order to test the performance of the proposed lossless DNA compression algorithm, the standard benchmark data used in [3-10] was coded using the proposed algorithm. This standard benchmark includes the complete genomes of two mitochondria: MPOMTCG, PANMTPACGA (also called MIPACGA); two chloroplasts: CHNTXX and CHMPXX (also called MTPACG); five sequences from humans: HUMGHCSA, HUMHBB, HUMHDABCD, HUMDYSTROP, HUMHPRTB; and finally the complete genome from two viruses: VACCG and HEHCMVCG (also called HS5HCMVCG).

Table 1 presents the obtained coding results using the proposed DNA-Residual algorithm. For comparison, Table 1 also shows the coding results obtained by using existing state-of-the-art lossless compression schemes. Note that the compression ratio (number of bits/base), for

Table 1. Lossless coding results (bits/base) using the proposed DNA Residual algorithm and comparison with existing lossless DNA compression algorithms.

Sequence	Size (bytes)	Bio2 [3]	Gen2 [5]	DNA Compress [6]	DNAPack [7]	CTW-LZ [8]	GeNML [9]	Proposed DNA Residual
CHMPXX	121024	1.68	1.67	1.67	1.66	1.67	1.66	**0.01**
CHNTXX	155844	1.62	1.61	1.61	1.61	1.61	1.61	**0.69**
HEHCMVCG	229354	1.85	1.85	1.85	**1.83**	1.84	1.84	2.03
HUMDYSTROP	38770	1.93	1.92	1.91	1.03	1.92	1.91	**0.06**
HUMGHCSA	66495	1.31	1.1	1.03	1.91	1.1	**1.01**	2.14
HUMHDABCD	58864	1.88	1.82	1.8	1.74	1.82	**1.71**	2.13
HUMHPRTB	56737	1.91	1.85	1.82	1.78	1.84	1.76	**1.69**
MPOMTCG	186608	1.94	1.91	1.89	1.74	1.9	1.88	**1.45**
MTPACG	100314	1.88	1.86	1.86	1.86	1.86	1.84	**0.01**
VACCG	191737	1.76	1.76	1.76	1.76	1.76	1.76	**0.10**
AVERAGE	-	1.7706	1.7350	1.7200	1.6920	1.7320	1.6980	**1.031**

the proposed algorithm, is calculated by summing the total number of bits (encoded bits + filter coefficients) and dividing by the total number of bases in the gene. From Table 1, it can be seen that, on average, the DNA-Residual is resulting in a significantly lower average bit-rate (1.031 bit/base) as compared to the existing lossless DNA compression schemes (lowest average bit-rate is 1.698/base using GeNML [9]). The relation between the bases is almost linear for genes 'CHMPXX' and 'MTPACG', resulting in a very high compression ratio (very low bit-rate) with the proposed scheme. On the other hand, for some genes such as 'HEHCMVCG', the results are not satisfactory implying that the bases are decorrelated. To overcome this problem, a combination of GeNML and DNA-Residual may be used. For example, for the high-entropy, less correlated sequences, GeNML is used, while for correlated sequences, the proposed DNA-Residual scheme is selected. Finally, more efficient context-based adaptive binary arithmetic encoders will be investigated to approach the calculated entropy for the less correlated sequences.

5. REFERENCES

[1] Grumbach S. and Tahi F., "A new Challenge for Compression Algorithms", *Journal of Information Processing and Management*, vol. 30, pp. 875-866, 1994.

[2] Chakravarthy N, Spanias A, Iasemidis LD, et al., " Autoregressive Modeling and Feature Analysis of DNA Sequences", *EURASIP Journal on Applied Signal Processing*, Jan. 2003.

[3] Grumbach S. and Tahi F., "Compression of DNA Sequences", *Proceeding of the Data Compression Conference*, pp340-350, 1993.

[4] Chen, X., Kwong, S., Li, M., "A Compression Algorithm for DNA Sequences and its Applications in Genome Comparison", *The 10th Workshop on Genome Informatics*, 1999.

[5] Chen, X., Kwong, S., Li, M., "A Compression Algorithm for DNA Sequences", *IEEE Engineering and Biology Magazine*, vol. 2-, 2001.

[6] Chen, X., Li, M., Ma, B. and Tromp J., " DNACompress: Fast and Effective DNA Sequence Compression". *Bioinformatics*, 18:1696-1698, 2002.

[7] Behzadi B., LE Fessant F., "DNA Compression Challenge Revisited", *16th Annual Symposium on Combinational Pattern Matching*, June 2005.

[8] Matsumuto T., Sadakane K.,Imai H., " Biological sequence compression algorithms", *Genome Inform. Ser. Workshop Genome Inform*. 11:43-52, 2000.

[9] Korodi G. and Tabus I., "An Efficient Normalized Maximum Likelihood Algorithm for DNA Sequence Compression". *ACM Transactions on Information Systems*, Jan, vol.23, No.1, 2005.

[10] Rivals E., Delahaye J.-P., Dauchet M., Delgrange O., "A Guaranteed Compression Scheme for Repetitive DNA Sequences", *Data Compression Conference*, 1996.

[11] Buldyrev S.V.,Goldberger A.L.,Havlin S., et al.,"Long-Range Correlation Properties of Coding and Noncoding DNA Sequences:GenBank Analysis," *Phys.Rev.E*, vol.51,no.5,pp. 5084–5091, 1995.

[12] Galvan P.B., Carpena P., Roldan R.R., and Oliver J.L., "Study of Statistical Correlations in DNA Sequences", *Gene*, vol.300,no.1-2,pp.105–115,2002.

[13] Knapen E., Reefman D., Janssen E., and Bruekers F., "Lossless Compression of One-bit Audio", *Journal of Audio Engineering Society*, vol. 52, no. 2, Feb 2004.

[14] Jackson, L.B., "Digital Filters and Signal Processing", 2nd Edition, Kluwer Academic Publishers, 1989.

[15] Sayood K., "Introduction to Data Compression,", Morgan Kaufmann Series, Feb. 2000.