EXPLORING THREE-BASE PERIODICITY FOR DNA COMPRESSION AND MODELING

Paulo J. S. G. Ferreira, António J. R. Neves, Vera Afreixo, Armando J. Pinho

Signal Processing Lab, DET / IEETA
University of Aveiro, 3810-193 Aveiro, Portugal

ABSTRACT

To explore the three-base periodicity often found in protein-coding DNA regions, we introduce a DNA model based on three deterministic states, where each state implements a finite-context model. The results obtained show compression gains in relation to the single finite-context model counterpart. Additionally, and potentially more interesting than the compression gain on its own, is the observation that the entropy associated to each of the three states differs and that this variation is not the same among the organisms analyzed.

1. INTRODUCTION

There is a steady demand for efficient methods able to reduce the storage space taken by the impressive amount of genomic data that are continuously being generated, and to investigate their structure. Although this is a paper on DNA data compression that describes a new DNA-specific compression method, its goals go beyond compression. We seek to better understand and model specific regions of the DNA data, the protein-coding zones. It is known that these DNA regions possess specific properties, different from those of non-coding parts. Particularly, they are generally more difficult to compress, because the main feature used by most DNA compressors, the occurrence of sequence repeats, is not so frequent in these zones [1]. However, there is a characteristic of protein-coding regions that has not been yet exploited for compression: The three-base periodicity [2].

The aim of this paper is to explore the three-base periodicity property of protein-coding regions in the context of data compression. To achieve this we propose a model composed of three states. Each of the models is selected periodically, according to the three-base period, and each state is implemented using a finite-context model. The comparison of this cyclically varying three-state model with the single finite-context model counterpart shows that it is able to to better capture the statistics of the data. We have also found that the entropy of the three states differs, and that the variation is not the same among the organisms analyzed, a fact that may have independent interest.

2. DNA COMPRESSION METHODS

The first compression method designed specifically for DNA sequences is *Biocompress* [3]. It explores the occurrence of complemented inverted repeats, and switches between LZ compression and transparent mode. *Biocompress-2* introduced a third mode of operation, based on a second order finite-context arithmetic encoder [4].

Rivals *et al.* proposed another compression technique based on exact repetitions, *Cfact*, which relies on a two-pass strategy [5, 6]. Contrarily to *Biocompress*, *Cfact* does not explore any particularity of DNA sequences and hence it can be considered a general purpose compression algorithm.

The idea of using repeating sub-sequences as a means of achieving compression was also exploited by Chen *et al.* [7, 8]. One version of the algorithm, *GenCompress-1*, used only replacement operations. The next version, *GenCompress-2* was also able to perform deletion and insertion operations in the sub-sequence. Both schemes appear to show identical compression performance, which seems to indicate that replacements should be enough.

A further modification of *GenCompress* led to a two-pass algorithm, *DNACompress*, relying on a separated tool for approximate repeat searching, *PatternHunter*, [9]. Besides providing additional compression gains, *DNACompress* is considerably faster than *GenCompress*.

Before the publication of *DNACompress*, a technique based on context tree weighting and LZ compression was proposed [10]. It provided a slight compression gain over *GenCompress* [10], but the computing time needed for large sequences showed to be prohibitive [9].

The paradigm of exact matching was addressed recently by Manzini *et al.* [11]. The aim was a fast, although competitive, DNA encoder. One of the key problems of compression techniques based on sub-sequence matching is the time taken by the search operation. Manzini *et al.* addressed this drawback by proposing a solution based on fingerprints. Basically, in this approach, the possibility of matching small sub-sequences is waived in exchange for increased speed.

Tabus *et al.* proposed a DNA sequence compression method based on normalized maximum likelihood discrete regression for approximate block matching [12].

Contact author: Paulo J. S. G. Ferreira, pjf@det.ua.pt. We thank X. Chen and G. Manzini for providing an implementation of their compression algorithms, and the *Fundação para a Ciência e Tecnologia* (FCT) for funding.

1-4244-0469-X/06/$20.00 ©2006 IEEE

3. THE NEW THREE-STATE MODEL

Most existing DNA compressors emphasize finding good exact/approximate repeats or inverted complements. In our opinion, other potentially important aspects, such as the particular characteristics of protein-coding regions, deserve equal attention.

In this paper we seek to explore the three-base periodicity often found in protein-coding regions [2]. This periodicity is useful to locate the protein-coding regions [13, 14], and motivated the development of fast algorithms for calculating the spectral coefficient of interest [15].

The three-base periodicity suggests an underlying statistical model driven by three different, although related, information sources. To test this conjecture, we use a setup based on three states, each a finite-context model, and where the switching between states follows the three-base periodicity.

Consider an information source that generates symbols, s, from an alphabet $\mathcal{A}$. At time t, the sequence of outcomes generated by the source is $x^t = x_1 x_2 \dots x_t$. A finite-context model (see Fig. 1) of an information source assigns probability estimates to the symbols of the alphabet, according to a conditioning context computed over a finite and fixed number, M, of past outcomes (order-M finite-context model) [16]. At time t, we represent these conditioning outcomes by $c^t = x_t, x_{t-1}, \dots, x_{t-M+1}$. The number of conditioning states of the model is $|\mathcal{A}|^M$, dictating its complexity (or model cost).

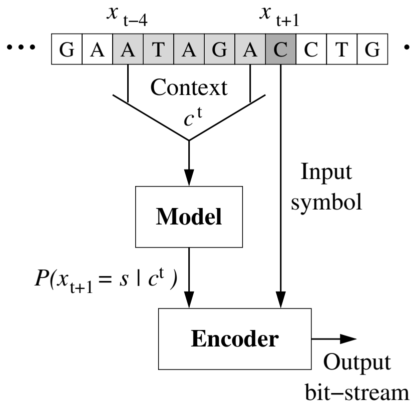

Fig. 1. Finite-context model: The probability of the next outcome, x_{t+1}, is conditioned by the M last outcomes.

In practice, the probability estimate of the next outcome x_{t+1} being $s \in \mathcal{A}$ is obtained from

$$P(x_{t+1} = s|c^t) = \frac{n(s, c^t) + \delta}{\sum_{a \in \mathcal{A}} n(a, c^t) + |\mathcal{A}|\delta}.$$

In our case $\delta = 1$, and $n(s, c^t)$ is the number of times that in the past the information source generated symbol s having c^t as the conditioning context. These counters are updated each time a symbol is encoded. The context template is causal, and the decoder is able to reproduce identical probability estimates without side information.

Figure 2 shows the model used in this paper. It differs from the finite-context model of Fig. 1 by the inclusion of three internal states. Each state is selected periodically, according to a three-base period. Each state comprises a finite-context model, similar to the one in Fig. 1.

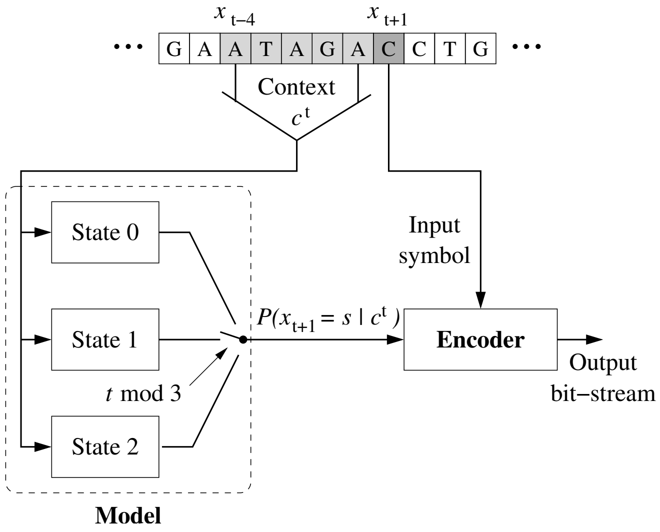

Fig. 2. Three-state model, exploiting the three-base periodicity of protein-coding regions.

With this model, the probabilities depend not only on the M last outcomes, but also on the value of $t \bmod 3$. Note that although the model relies on the codon structure of protein-coding regions, it does not require the knowledge of the correct reading frame. But if a particular codon position is chosen to start the model, the corresponding reading frame should be maintained, or the statistics will become mixed.

4. EXPERIMENTAL RESULTS

The reported results were obtained using data from the `ffn` files found in `ftp.ncbi.nlm.nih.gov/genomes`. We included data from *Haemophilus influenzae*, *Escherichia coli K12*, *Schizosaccharomyces pombe*, *Saccharomyces cerevisiae* and *Arabidopsis thaliana*. To avoid undesirable interferences, the files were checked for genes containing spurious symbols or having a length other than a multiple of three.

Table 1 presents the compression results obtained using four coding approaches: The three-state finite-context model, the single-state finite-context model, *DNACompress* [9] (denoted by *DnaC*) and Manzini's method *Dna3* [11]. For each sequence, the optimal context depth M was used.

The results show that the three-state model is always better than the single-state finite-context model, confirming the

Table 1. Compression results, in bits per base (bpb), obtained for *Haemophilus influenzae*, *Escherichia coli K12*, *Schizosaccharomyces pombe*, *Saccharomyces cerevisiae* and *Arabidopsis thaliana*. For the three-state model, columns "bpb0", "bpb1" and "bpb2" indicate the compression rates attained by each of the three states (corresponding to their respective sub-sequence, i.e., to one third of the whole sequence), whereas the "bpb" column indicates overall compression rate. Columns "M" indicate the order of the finite-context model, which was always the best possible. Columns "*DnaC*" and "*Dna3*" show compression results using, respectively, *DNACompress* and Manzini's *Dna3* method.

Haemophilus influenzae

Reference	Sequence	Bases	Three-state					Single-state		*DnaC*	*Dna3*
			M	bpb0	bpb1	bpb2	bpb	M	bpb	bpb	bpb
GI:16271976	—	1 505 271	4	1.918	1.834	1.684	**1.812**	5	1.889	1.902	1.895

Escherichia coli K12

Reference	Sequence	Bases	Three-state					Single-state		*DnaC*	*Dna3*
			M	bpb0	bpb1	bpb2	bpb	M	bpb	bpb	bpb
GI:49175990	—	4 083 231	5	1.897	1.898	1.750	**1.848**	6	1.917	1.920	1.913

Schizosaccharomyces pombe

Reference	Sequence	Bases	Three-state					Single-state		*DnaC*	*Dna3*
			M	bpb0	bpb1	bpb2	bpb	M	bpb	bpb	bpb
GI:19113674	Chr-I	2 996 109	4	1.961	1.884	1.820	**1.889**	4	1.939	1.918	1.921
GI:19111836	Chr-II	2 399 394	4	1.962	1.887	1.818	**1.889**	4	1.940	1.915	1.916
GI:19075172	Chr-III	1 169 991	3	1.961	1.889	1.833	**1.895**	4	1.943	1.925	1.930

Saccharomyces cerevisiae

Reference	Sequence	Bases	Three-state					Single-state		*DnaC*	*Dna3*
			M	bpb0	bpb1	bpb2	bpb	M	bpb	bpb	bpb
GI:50593113	Chr-I	143 157	2	1.937	1.882	1.909	1.911	3	1.954	**1.884**	1.910
GI:50593115	Chr-II	605 184	3	1.936	1.869	1.886	**1.897**	3	1.942	1.912	1.918
GI:42759850	Chr-III	217 332	2	1.946	1.874	1.908	**1.911**	3	1.951	1.918	1.923
GI:50593138	Chr-IV	1 129 605	3	1.931	1.856	1.882	1.890	4	1.936	**1.846**	1.853
GI:7276232	Chr-V	391 086	3	1.935	1.872	1.894	1.901	3	1.947	**1.883**	1.894
GI:42742172	Chr-VI	183 702	2	1.938	1.863	1.904	**1.904**	3	1.949	1.932	1.939
GI:50593213	Chr-VII	784 707	3	1.935	1.861	1.882	**1.893**	3	1.939	1.897	1.902
GI:50882583	Chr-VIII	402 792	3	1.938	1.873	1.896	**1.903**	3	1.946	1.907	1.915
GI:6322016	Chr-IX	310 041	3	1.938	1.869	1.900	**1.903**	3	1.947	1.933	1.942
GI:42742252	Chr-X	557 103	3	1.935	1.866	1.892	**1.899**	3	1.943	1.907	1.914
GI:50593424	Chr-XI	478 620	3	1.935	1.855	1.893	**1.895**	3	1.940	1.938	1.942
GI:42742286	Chr-XII	784 695	3	1.936	1.862	1.893	1.898	3	1.942	**1.863**	1.872
GI:44829554	Chr-XIII	693 291	3	1.934	1.859	1.889	1.894	3	1.940	**1.886**	1.892
GI:50593505	Chr-XIV	576 585	3	1.937	1.869	1.893	**1.900**	3	1.944	1.930	1.934
GI:42742309	Chr-XV	785 568	3	1.937	1.865	1.887	**1.897**	3	1.941	1.901	1.917
GI:50593503	Chr-XVI	687 666	3	1.937	1.862	1.887	1.896	3	1.941	1.889	**1.880**
GI:6226515	MT	24 429	2	1.814	1.767	1.305	1.643	3	1.747	**1.466**	1.511

Arabidopsis thaliana

Reference	Sequence	Bases	Three-state					Single-state		*DnaC*	*Dna3*
			M	bpb0	bpb1	bpb2	bpb	M	bpb	bpb	bpb
GI:42592260	Chr-I	9 595 494	5	1.925	1.904	1.882	1.904	6	1.939	**1.725**	1.743
GI:30698031	Chr-II	5 474 178	4	1.926	1.906	1.886	1.906	6	1.942	**1.710**	1.737
GI:30698537	Chr-III	7 183 863	5	1.925	1.904	1.886	1.905	6	1.941	**1.736**	1.762
GI:30698542	Chr-IV	5 572 038	4	1.926	1.905	1.888	1.906	6	1.942	**1.708**	1.740
GI:30698605	Chr-V	8 462 424	5	1.924	1.902	1.883	1.903	6	1.939	**1.736**	1.759

known three-base periodicity of these DNA regions. The three-state model also attained better compression results than the state-of-the-art DNA compression techniques included in the tests for the *Haemophilus influenzae*, *Escherichia coli K12* and *Schizosaccharomyces pombe*. For the *Saccharomyces cerevisiae*, the results are mixed, although for more than half of the chromosomes the three-state model performed better. Finally, for the *Arabidopsis thaliana*, the three-state finite-context model falls short in comparison to *DNACompress* and *Dna3*.

5. DISCUSSION AND CONCLUSIONS

Table 1 shows the compression gains obtained by exploring the three-base periodicity of the protein-coding regions. For some of the organisms, our approach outperformed the state-of-the-art DNA compression techniques. Since *DNACompress* and *Dna3* rely strongly on sub-sequence repetition, as in fact most of the DNA compression techniques do, we believe that in those sequences repetitions are relatively rare.

The opposite also holds, i.e., the relative performance of the three-state model decreases for sequences containing many repetitions, better exploited by other algorithms. This is precisely what happens with the *Arabidopsis thaliana* genome. Many plant genomes show considerable amounts of repetition, and *Arabidopsis thaliana* is known to have extensive gene repetition [17]. Thus, when compression performance is the only goal, the three-state model has to be complemented with a method able to explore these repetitive patterns.

Another interesting issue that follows from Table 1 is related to the bit-rates for the three states, and how do they compare. The values denoted by "bpb0", "bpb1" and "bpb2" indicate the average number of bits required by the encoder to represent, respectively, the first, the second and the third bases of the codon. For the *Haemophilus influenzae*, *Schizosaccharomyces pombe* and *Arabidopsis thaliana*, the first base is the hardest to compress, then the second and finally the third. Thus, the first base conveys the largest fraction of the codon information.

This is not surprising, since in the genetic code most amino-acids are represented by more than one triplet and, for some, the third base is irrelevant. The really surprising fact is that for all chromosomes of *Saccharomyces cerevisiae* the second base seems to carry less information than the third one. Moreover, although marginally, the second base of *Escherichia coli K12* seems to carry at least as much information as the first base. We are unable to provide an explanation for these facts, which justify further analysis.

In conclusion, our findings indicate that the three-base periodicity found in the protein-coding regions can be explored from the viewpoint of data compression. The compression performance of a three-state finite-context model always behaves better in those regions that the single-state counterpart. For some organisms, the performance exceeds that of state-of-the-art DNA compression techniques. Our model opens up the possibility of analyzing how information is distributed among the three bases of the codon, and we found that for the *Saccharomyces cerevisiae* the third base of the codon carries, on average, more information than the second base, an intriguing fact whose biological significance, if any, remains unclear and seems worth of further study.

6. REFERENCES

[1] N. V. Dokholyan, S. V. Buldyrev, S. Havlin, and H. E. Stanley, "Distribution of base pair repeats in coding and noncoding DNA sequences," *Physical Review Letters*, vol. 79, no. 25, pp. 5182–5185, Dec. 1997.

[2] E. N. Trifonov and J. L. Sussman, "The pitch of chromatin DNA is reflected in its nucleotide sequence," *Proc. Natl. Acad. Sci. USA*, vol. 77, no. 7, pp. 3816–3820, July 1980.

[3] S. Grumbach and F. Tahi, "Compression of DNA sequences," in *Proc. of the Data Compression Conf., DCC-93*, Snowbird, Utah, 1993, pp. 340–350.

[4] ——, "A new challenge for compression algorithms: genetic sequences," *Information Processing & Management*, vol. 30, no. 6, pp. 875–886, 1994.

[5] E. Rivals, J.-P. Delahaye, M. Dauchet, and O. Delgrange, "A guaranteed compression scheme for repetitive DNA sequences," LIFL, Université des Sciences et Technologies de Lille, Tech. Rep. IT–95–285, Nov. 1995.

[6] ——, "A guaranteed compression scheme for repetitive DNA sequences," in *Proc. of the Data Compression Conf., DCC-96*, Snowbird, Utah, 1996, p. 453.

[7] X. Chen, S. Kwong, and M. Li, "A compression algorithm for DNA sequences and its applications in genome comparison," in *Genome Informatics 1999: Proc. of the 10th Workshop*, K. Asai, S. Miyano, and T. Takagi, Eds., Tokyo, Japan, 1999, pp. 51–61.

[8] ——, "A compression algorithm for DNA sequences," *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, pp. 61–66, 2001.

[9] 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.

[10] T. Matsumoto, K. Sadakane, and H. Imai, "Biological sequence compression algorithms," in *Genome Informatics 2000: Proc. of the 11th Workshop*, A. K. Dunker, A. Konagaya, S. Miyano, and T. Takagi, Eds., Tokyo, Japan, 2000, pp. 43–52.

[11] G. Manzini and M. Rastero, "A simple and fast DNA compressor," *Software - Practice and Experience*, vol. 34, pp. 1397–1411, 2004.

[12] G. Korodi and I. Tabus, "An efficient normalized maximum likelihood algorithm for DNA sequence compression," *ACM Trans. on Information Systems*, vol. 23, no. 1, pp. 3–34, Jan. 2005.

[13] J. W. Fickett, "Recognition of protein coding regions in DNA sequences," *Nucleic Acids Research*, vol. 10, no. 17, pp. 5303–5318, 1982.

[14] B. Issac, H. Singh, H. Kaur, and G. P. S. Raghava, "Locating probable genes using Fourier transform approach," *Bioinformatics*, vol. 18, no. 1, pp. 196–197, 2002.

[15] V. Afreixo, P. J. S. G. Ferreira, and D. Santos, "Spectrum and symbol distribution of nucleotide sequences," *Physical Review E*, vol. 70, p. 031910, 2004.

[16] T. C. Bell, J. G. Cleary, and I. H. Witten, *Text compression*. Prentice Hall, 1990.

[17] V. Walbot, "A green chapter in the book of life," *Nature*, vol. 408, pp. 794–795, Dec. 2000.