Available online at www.sciencedirect.com

ScienceDirect

BioSystems 90 (2007) 20–27

www.elsevier.com/locate/biosystems

How to describe genes: Enlightenment from the quaternary number system

Bin-Guang Ma[a,b,*]

[a] College of Chemistry and Chemical Engineering, Suzhou University, Suzhou 215006, PR China
[b] Shandong Provincial Research Center for Bioinformatic Engineering and Technique, Center for Advanced Study, Shandong University of Technology, Zibo 255049, PR China

Received 13 May 2005; received in revised form 15 June 2006; accepted 19 June 2006

Abstract

As an open problem, computational gene identification has been widely studied, and many gene finders (software) become available today. However, little attention has been given to the problem of describing the common features of known genes in databanks to transform raw data into human understandable knowledge. In this paper, we draw attention to the task of describing genes and propose a trial implementation by treating DNA sequences as quaternary numbers. Under such a treatment, the common features of genes can be represented by a "position weight function", the core concept for a number system. In principle, the "position weight function" can be any real-valued function. In this paper, by approximating the function using trigonometric functions, some characteristic parameters indicating single nucleotide periodicities were obtained for the bacteria *Escherichia coli* K12's genome and the eukaryote yeast's genome. As a byproduct of this approach, a single-nucleotide-level measure is derived that complements codon-based indexes in describing the coding quality and expression level of an open reading frame (ORF). The ideas presented here have the potential to become a general methodology for biological sequence analysis.

Keywords: Biological sequence analysis; Codon usage; Gene identification

1. Background

In the past two decades, computational gene identification has been widely studied in response to the rapid growth of DNA sequence databases. Many methodologies have been employed to address this problem and they can be roughly categorized as follows: (i) Treating DNA sequence information as the result of a stochastic process, such as a Markov chain, Hidden Markov Model (HMM), or the corresponding expectation–maximization algorithm (Borodovsky and McIninch, 1993; Salzberg et al., 1998; Lukashin and Borodovsky, 1998; Cardon and Stormo, 1992). (ii) Treating DNA sequences as having spatial or temporal signal distribution, such as using a Fourier transformation (Tiwari et al., 1997), wavelet analysis (Tsonis et al., 1996), or power spectrum analysis (Atsushi et al., 2002). (iii) Transforming DNA sequences into some kind of geometrical representation, such as the early H curves (Hamori and Ruskin, 1983) and the later *Z* curves (Zhang and Zhang, 1994). (iv) Linguistic methods that treat DNA sequences as some kind of language (Dong and Searls, 1994; Pesole et al., 1996). For the assessment and comparison of all kinds of methodologies and gene-finder programs, see Fickett (1996), Guigo (1997), Bajic (2000), Rogic et al. (2001) and Mathe et al. (2002).

* Tel.: +86 533 2780271; fax: +86 533 2780271.
E-mail address: bgMa@sdut.edu.cn.

0303-2647/$ – see front matter © 2006 Elsevier Ireland Ltd. All rights reserved.
doi:10.1016/j.biosystems.2006.06.004

In different methodologies there are different approaches for gene representation. For example, HMM is one of the predominant math tools used for gene finding in recent years, in which genes are usually described as thousands of parameters known as state-transition probabilities and output probabilities (Lukashin and Borodovsky, 1998). These quantitative parameters become an important complement to the qualitative knowledge about protein coding genes that they start from a start codon and stop at a stop codon (a definition of an open reading frame). On the other hand, the use of so many parameters is not very efficient in representing knowledge for human understanding. For another example, we look at geometrical representation. In such methodologies, DNA sequences are transformed into some kind of visible graph (Hamori and Ruskin, 1983; Zhang and Zhang, 1994), which may provide an intuitive impression about what genes may look like but is not very useful for automated gene identification. Additionally, even in such methods, still we cannot easily describe the common features of genes.

Therefore, although more than 90% accuracy has been achieved in some gene finders (Bajic, 2000; Rogic et al., 2001; Mathe et al., 2002), the common features of genes still remain elusive due to the lack of a proper representation method. Is there some more balanced way to describe the common features of genes so as to be not only useful for gene recognition but also meaningful for human understanding? This is the task of describing genes.

2. The task of describing genes

Consider the *Escherichia coli* K12 genome as an example to explain the task. There exist a total of 4289 annotated protein coding genes in the GenBank release 131.0, among which the shortest is 45 base pairs and the longest is 7152 base pairs. In the database, DNA sequences are represented by a coarse-grained form: character sequences, *viz.*, permutations of four nucleotides A, C, G, T from the viewpoint of combinatorics. However, if all the permutation numbers of the four nucleotides are summed from 45 to 7152, the gross number is 1.1429E4306, an astronomical figure. Out of such a large space, only 4289 of them are selected by nature to code proteins. Natural and interesting questions arise: what are the common features of gene permutations that differentiate them from other permutations? Can we find and describe these features?

The key to answering these questions is how to extract information from character sequences. In fact, if we treat a character sequence as an information source, we can partition its information into two components: composition and permutation, where "composition" means the occurrence number of characters in a character sequence and "permutation" means their order of appearance in the sequence. Therefore, in order to describe genes, we need to create a way to describe the two components of information.

Composition information can be described easily by ordinary frequencies that correspond directly to occurrence numbers of nucleotides in DNA sequences. The difficulty is how to describe permutation information. For example, examine the following two sequences s_1 and s_2:

s_1	*A*	*G*	*C*	*A*	*T*	*A*	*G*	*G*	**T**	**C**	*C*	*A*	*C*	*A*	*G*	*T*	*T*	*G*
k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
s_2	*A*	*G*	*C*	*A*	*T*	*A*	*G*	*G*	**C**	**T**	*C*	*A*	*C*	*A*	*G*	*T*	*T*	*G*
k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18

where *k* denotes position. Note the difference between s_1 and s_2. Both s_1 and s_2 have the same numbers of A, C, G, T and if the ninth character T and the 10th character C of s_1 exchange their positions, s_1 turns into s_2, which means that the composition of s_1 is completely the same as that of s_2.

If we use frequencies to describe the composition information of s_1 and s_2, we will get

$$f_{\mathrm{a}1} = 5/18, \quad f_{\mathrm{c}1} = 4/18, \quad f_{\mathrm{g}1} = 5/18, \quad f_{\mathrm{t}1} = 4/18 \tag{1}$$

and

$$f_{\mathrm{a}2} = 5/18, \quad f_{\mathrm{c}2} = 4/18, \quad f_{\mathrm{g}2} = 5/18, \quad f_{\mathrm{t}2} = 4/18. \tag{2}$$

As shown, $f_{i1} = f_{i2}$ for i = a, c, g, t, namely, all the frequencies of the four kinds of nucleotides for s_2 are identical with those for s_1, respectively. So the difference between s_1 and s_2 lies in permutation (appearing order of A, C, G, T) not in composition (occurrence number of them) and ordinary frequencies cannot reflect this difference.

Suppose that s_1 is a protein coding gene and s_2 is not. The difference is reflected in permutation information, which may be represented using a quaternary number system.

3. Quaternary number-based implementation

3.1. Quaternary number system

The base of quaternary number system is four and digits used in it are 0–3. For a number N with L digits, its decimal representation is

$$N = \sum_{k=1}^{L} 4^{L-k} x_k, \tag{3}$$

where x_k ($x_k \in \{0, 1, 2, 3\}$) is the digit at position k and $I(k) = 4^{L-k}$ is the position weight function. Note that "position weight function" is a core concept for a number system, which means that different positions in a numerical sequence have different weights and the weights can be described by a real-valued function.

For an instance of a nine-digit number $N_1 = 233202001$, its decimal representation is

$$N_1 = \sum_{k=1}^{9} 4^{9-k} x_k = 2 \times 4^8 + 3 \times 4^7 + \cdots$$
$$+0 \times 4^1 + 1 \times 4^0 = 194689.$$

For another nine-digit number $N_2 = 203202031$, its decimal representation is

$$N_2 = \sum_{k=1}^{9} 4^{9-k} x_k = 2 \times 4^8 + 0 \times 4^7 + \cdots$$
$$+3 \times 4^1 + 1 \times 4^0 = 145549.$$

As numerical sequences, N_1 (=233202001) and N_2 (=203202031) have the same occurrence numbers of digits 0–3, while the order of appearance of the digits in the two numerical sequences 233202001 and 203202031 is different. This is similar to the situation of the above-mentioned two character sequences s_1 and s_2. The quaternary number system can differentiate the two numerical sequences by mapping them onto different decimal numbers 194689 and 145549.

3.2. Position weight function

A quaternary number system can differentiate the above two numerical sequences thanks to its position weight function, which reflects permutation information. For a numerical sequence, the resultant value depends both on its digits and their order because the result is a weighted sum of the digits and the weights are described by a position weight function.

Therefore, if we simply construct the map A → 0, C → 1, G → 2, T → 3 (for convenience, we call it a ACGT map), we can translate DNA sequences into quaternary numbers and then permutations of A, C, G, T can be differentiated. Note that even in the quaternary number system, 0000 and 00 are the same number, i.e., AAAA and AA cannot be differentiated. To avoid this problem, we modify the above map into A → 1, C → 2, G → 3, T → 4. In this case, AAAA = 1111 = 85 and AA = 11 = 5 so that they can be differentiated.

The position weight function of quaternary number system is an exponential function ($I(k) = 4^{L-k}$) and it increases quickly with the value of k. Therefore, to get the common features of protein coding genes, we generalize the position weight function $I(k)$ into any real-valued function. Then any character sequence s composed of A, C, G, T can be mapped onto a real number by the following formula:

$$S = \sum_{k=1}^{L} I(k) x_k - L, \tag{4}$$

where L is the length of the character sequence, $I(k)$ the generalized position weight function, and x_k is the map:

$$x_k = \begin{cases} 1 & \text{if } s[k] = \text{A}, \\ 2 & \text{if } s[k] = \text{C}, \\ 3 & \text{if } s[k] = \text{G}, \\ 4 & \text{if } s[k] = \text{T}. \end{cases} \tag{5}$$

The symbol $s[k]$ in Eq. (5) denotes the character at position k in the sequence s and S (the left side of Eq. (4)) is a real number whose value is determined by the character sequence s. S can serve as an index for its corresponding character sequence because it contains the full information (both composition and permutation) of that sequence.

With an ACGT map and position weight function, each character sequence composed of A, C, G, T can be transformed into a real number. For any two sequences i and j, we define the distance between them as

$$d_{ij} = \sqrt{(S_i - S_j)^2}, \tag{6}$$

where S_i and S_j are real numbers that correspond to character sequences i and j, respectively. Our aim is to find common features of protein coding genes. Protein coding genes are only a small subset of possible permutations of A, C, G, T and their common features make them share a common position weight function through which these genes can assemble into some cluster. The problem is how to obtain this position weight function.

Since genes can assemble into one cluster under their common position weight function, we can define an objective function as

$$D = \sum_{i<j}^{n} {d_{ij}}^2 = \sum_{i<j}^{n} (S_i - S_j)^2, \tag{7}$$

where S_i and S_j are the indexes (real numbers) of genes i and j, respectively and d_{ij} denotes the distance between them. Then, D represents the total sum of the distances between every two genes. The smaller the value of D, the more compact the gene cluster.

By minimizing the function D, the position weight function $I(k)$ can be determined:

$$\min(D) \Rightarrow I(k) \tag{8}$$

3.3. Approximation using trigonometric functions

Since $I(k)$ can be any real-valued function of position k and its form is unknown, trigonometric functions can be used to approximate $I(k)$. Let

$$I(k) = \sum_{m} a_m \cos\frac{2\pi k}{m} \quad (m = 1, 2, \ldots). \tag{9}$$

Then min(D) means

$$\frac{\partial D}{\partial a_m} = 0 \quad (m = 1, 2, \ldots, 12). \tag{10}$$

Here we take the first 12 items of series (9) as the approximation of $I(k)$. By solving Eq. (10), the coefficients a_m ($m = 1, 2, \ldots, 12$) can be determined.

3.4. Test the efficiency of a position weight function

Since we can only get an approximation of a set of genes' common position weight function, we need an indicator to assess the quality of this approximation in terms of describing the common features of genes.

The Fisher discrimination algorithm is employed. To complete this algorithm, we need some characteristic variables at first. Recalling formula (4), it is a weighted sum of a DNA sequence under an ACGT map and can be rewritten as

$$S = \sum_{k=1}^{L} I(k)x_k - L = S_\mathrm{a} + S_\mathrm{c} + S_\mathrm{g} + S_\mathrm{t}, \tag{11}$$

where

$$S_r = \sum_{k_r} I(k_r)x_{k_r} - L_r \quad (r = \mathrm{a, c, g, t}), \tag{12}$$

in which k_r is the position k where character r appears, and L_r is the occurrence number of character r in the sequence s (namely, $L_\mathrm{a} + L_\mathrm{c} + L_\mathrm{g} + L_\mathrm{t} = L$). In Eq. (11), the weighted sum of a DNA sequence is partitioned into four parts: S_a, S_c, S_g, S_t, and each part represents the contribution of each kind of nucleotide to the weighted sum, respectively. With a known $I(k)$, S_a, S_c, S_g, S_t can serve as characteristic variables for a DNA sequence.

Secondly, we construct two groups of samples. The first group is composed of positive samples that are annotated protein coding genes, and the second group is composed of negative samples (other permutations except protein coding genes) generated by shuffling the corresponding positive samples. Because of the generating method, the negative samples have the same composition to their corresponding positive samples, respectively.

For the *E. coli* K12 genome, we take half of its annotated genes as the training set, and the rest as the test set. For the yeast genome, we take half of its first category ORFs as the training set, and the rest as the test set (see Section 4).

In Fisher discrimination algorithm, S_a, S_c, S_g, S_t are written as a vector **S**. The Fisher linear discrimination equation in this case represents a hyperplane in a four-dimensional space, described by a vector **c**, which has four components c_1, c_2, c_3, c_4. See Zhang and Wang (2000) for the detailed procedure to determine **c**. According to the data in the training set, an appropriate threshold c_0 is determined to make the coding/non-coding decision. The threshold c_0 can be determined uniquely by making the false negative rate and the false positive rate equal to each other. Once the vector **c** and the threshold c_0 are obtained, the decision of coding/non-coding for each ORF in the test set is simply made by the criterion of $\mathbf{c} \cdot \mathbf{S} > c_0 / \mathbf{c} \cdot \mathbf{S} < c_0$, where $\mathbf{c} = (c_1, c_2, c_3, c_4)$ and $\mathbf{S} = (\mathrm{S_a, S_c, S_g, S_t})^\mathrm{T}$, where the superscript 'T' indicates the transposition of a matrix.

Here we define the s score of an ORF as $s = \mathbf{c} \cdot \mathbf{S} - c_0$, and then the coding/non-coding decision of an ORF is exactly to see $s > 0$ or $s < 0$. The s score may act as a single-nucleotide-level index for coding and the expression level of an ORF (see Section 5.3).

4. Materials

The complete DNA sequence of the bacteria *E. coli* K12 genome and related annotation information were downloaded from GenBank Release 131.0 (accession no. U00096). The eukaryote *Saccharomyces cerevisiae*'*s* genome and the latest classification of its ORFs were downloaded from http://speedy.mips.biochem.mpg.de. Here we used the first class (including 3275 ORFs with known functions) to test the efficiency of a position weight function.

5. Results and discussion

5.1. Common features of genes are revealed as single nucleotide periodicities

Since we use trigonometric functions to approximate the position weight function, the common features of genes are revealed as single nucleotide periodicities in the DNA sequences. The coefficient a_m (see series (9)) corresponds directly to the m-periodicity. The absolute value of the coefficient indicates the magnitude of this periodicity in the coding sequences. All the coefficients a_m ($m = 1, 2, \ldots, 12$) of the position weight function for *E. coli* K12 and yeast are listed in Table 1.

From its first row, we can see that in *E. coli* K12 genome, three-periodicity is the strongest signal in the protein coding genes that have been revealed by previous studies (Shepherd, 1981; Gutierrez et al., 1994) consisting with codon usage (Atsushi et al., 2002). As to the 10–11 periodicities which are interpreted as the reflection of helical repeat structure (Trifonov and Sussman, 1980; Tomita et al., 1999), it is not manifested here. In the yeast genome, three-periodicity is not well revealed under the map A → 1, C → 2, G → 3, T → 4 (see the third row of Table 1). But if the map A → 1, C → 3, G → 2, T → 4 is used, three-periodicity can be better revealed (see the fourth row of Table 1). The dependency of these coefficients on the ACGT map will be discussed later.

Specially, when $m = 1$, $\cos(2\pi k/m) = \cos(2\pi k) = 1$, a constant for any integer k. Thus, a_1 is the non-periodic item and actually a threshold. So it is more common than other coefficients in the two different genomes, i.e., the values of a_1 are very close to one another in all the three rows of Table 1.

Excluding a_1 (since it is non-periodic item), the absolute values of other coefficients are all much smaller than that of a_3, which means that the other periodicities are relatively weaker in protein coding genes. Among these weak signals, periodicities 5–7, 11, 12 are slightly stronger than periodicities 2, 8–10 in *E. coli* K12 genome (the second row of Table 1), while in yeast genome, the periodicities 4, 5, 7 are slightly stronger than periodicities 2, 6, 8–12. The intersection of the stronger periodicities comprises periodicities 3, 5 and 7, suggesting that periodicities 3, 5 and 7 of the 11 periodic items (excluding a_1) are more common than other periodicities in the two genomes. Then, we can say that the common features of genes in the two genomes are their strong 3, 5 and 7 periodicities of single nucleotides, which is much clearer knowledge for human understanding than thousands of parameters such as found in a Hidden Markov Model.

Although periodicity is not the only thing that differentiates an ordered sequence from a random or chaotic one, it does play an important role. Here we reemphasize that the position weight function can be any real-valued function. In this study, we use trigonometric functions to approximate it. One can nevertheless choose other functions (such as a wavelet function) to approximate it, and may find something more interesting.

5.2. Common features of genes differentiate them from other permutations

We intend to fulfill the task of describing the common features of genes. In our implementation, the common features of genes are revealed and represented by a shared position weight function which is obtained by minimizing the total sum of squared distances between every two genes (see min(D) in Eq. (8)). This shared position weight function is specific for genes. With this function, the protein coding genes can be distinguished from other permutations by the aid of the Fisher discrimination algorithm.

Table 2 lists the results using the s score to identify protein coding genes from other permutations. The meaning of each row is explained in the annotation of Table 2. Here the commonly adopted criteria for the assessment of gene finding algorithms, sensitivity, specificity and accuracy (Burset and Guigo, 1996) are employed to evaluate the performance. For the *E. coli* K12 genome, accuracy of more than 90% is obtained

Table 1
The coefficients for the approximation of position weight function using trigonometric functions for different species

Species	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}
E. coli K12	0.407699	0.008676	0.419232	−0.000774	0.024878	0.054243	−0.040630	−0.003351	−0.001115	−0.004613	0.013215	−0.034536
Yeast	0.405212	0.025472	0.057558	−0.017774	−0.089865	0.047355	−0.136560	−0.082939	−0.021185	0.017887	−0.002807	0.009648
Yeast[a]	0.413101	0.045268	0.219506	−0.087231	−0.087095	0.009430	−0.125123	−0.05817	−0.024227	−0.013825	−0.033328	−0.01458

[a] The coefficients corresponding to the map A → 1, C → 3, G → 2, T → 4.

Table 2
The average sensitivity, specificity and accuracy for coding/non-coding sequence recognition, averaged over 10-fold cross-validation for *E. coli* K12 and yeast in various ways

Species	ORF no.	Sensitivity (%)	Specificity (%)	Accuracy (%)
E. coli K12	4289	89.10 ± 0.50	94.18 ± 0.23	91.64
Yeast	3275	65.55 ± 0.69	72.95 ± 0.43	69.25
Yeast[a]	3275	86.44 ± 0.51	97.51 ± 0.23	91.98
E. coli K12[b]	4289	74.11 ± 0.49	71.88 ± 0.34	73.00
Yeast[c]	3275	86.80 ± 0.29	85.21 ± 0.48	86.01
E. coli K12[d]	4289	67.63 ± 0.68	62.47 ± 0.97	65.05
E. coli K12[e]	4289	76.33 ± 0.36	82.86 ± 0.59	79.60
E. coli K12[f]	4289	89.04 ± 0.48	94.64 ± 0.21	91.84
E. coli K12[g]	4289	67.65 ± 0.73	62.59 ± 0.79	65.12

[a] Results using *E. coli* K12's position weight function to recognize the protein coding genes in yeast genome.

[b] Results using yeast's position weight function to recognize the protein coding genes in *E. coli* K12 genome.

[c] Results using another position weight function under the map A → 1, C → 3, G → 2, T → 4.

[d] Results using only the a_1 item (threshold).

[e] Results using only the a_3 item (periodicity 3).

[f] Results using the a_1 and a_3 items.

[g] Results using other items excluding a_1 and a_3.

(the second row of Table 2). Fig. 1 shows the distribution of the two groups of samples (for *E. coli* K12) in the sample space (after principal component analysis). Apparently, the two groups of samples are largely separated, which means some common features of genes have been captured and these features are specific for genes that differentiate them from other permutations. For the yeast genome, the accuracy is less than 70% (the third row of Table 2), which means that either the common features of genes have not been nicely captured or the features captured are not exactly specific for genes. But if we try another ACGT map for the yeast genome, the accuracy can be improved to 86% (the sixth row of Table 2). This shows again the dependency of the efficiency of a position weight function on the ACGT map.

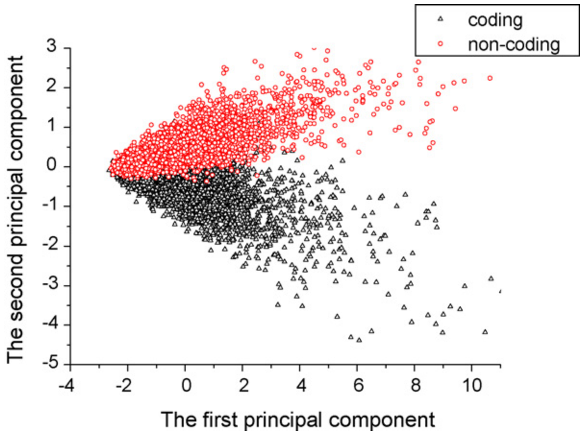

Fig. 1. The distribution of points on the plane spanned by the two most important principal components. With recognition variables S_a, S_c, S_g, S_t, the principal component analysis has been made for the 4289 protein coding sequences (positive samples) and the corresponding negative samples in the *E. coli* K12 genome. The first and second principal components account for 95% of the total inertia of the four components. Apparently the two groups of samples are largely separated, which means some common features of genes have been captured and these features are specific for genes that differentiate them from other permutations.

It is interesting to see whether or not a position weight function obtained from one species can be used to describe genes of another. We use the position weight function obtained from the *E. coli* K12 genome to recognize protein coding genes of the yeast genome, and the accuracy is actually as high as 91% (the fourth row of Table 2). However, while using the position weight function obtained from the yeast genome to recognize protein coding genes of the *E. coli* K12 genome, the results are still not good (the fifth row). This means that genes from different species also share common features. Once these features are characterized based on the genes from one species, they will show their universality in the task of describing the genes from other species.

Another point worth noting is that if only the a_1 and a_3 items of the position weight function for *E. coli* K12 are used, the results (the ninth row of Table 2) are equally good, even slightly better than the results obtained by using all the 12 items (the second row of Table 2). This may result from that the trigonometric functions used here are only cosine functions which cannot construct a perfect set without sine functions. Under such a condition, it is not necessarily true that the more items of the approximation of position weight function are used, the better the results.

From the results shown in Table 2, we can see that if the shared position weight function is found, very high accuracy can be achieved in differentiating genes from other permutations, which means that the present approach coupled with the Fisher discrimination algorithm can be applied to gene identification.

5.3. S score complements to codon-based indices

The well-known common feature of genes, so-called "codingness", was described traditionally using the codon bias index (CBI) (Bennetzen and Hall, 1982) or codon adaptation index (CAI) (Sharp and Li, 1987). In 1998, "codon usage" was introduced as a quantitative indicator to reflect gene expression level (Karlin et al., 1998). More recently, these codon-based indices have been revisited from a whole-genome perspective by Jansen et al. (2003).

As mentioned, to make the coding/non-coding decision of an ORF, the *s* score is defined as a criterion. We are interested in the relationship between the *s* score (single-nucleotide-level measure) and those codon-based indices. We have used the 306 predicted highly expressed (PHX) genes for *E. coli* K12 (Karlin and Mrazek, 2000) and scored these genes with a variety of models including CBI, CAI, $E(g)$ (the general expression measure by "codon usage") (Karlin and Mrazek, 2000) and the *s* score. The $E(g)$ values for the 306 PHX genes are obtained from the paper (Karlin and Mrazek, 2000). The CBI and CAI values of the 306 genes are calculated using the program CodonW (written by John F. Peden) downloaded from ftp://molbiol.ox.ac.uk/cu. For additional information about this program, see Peden (1999). Correlation analysis is made for these scores and the results are listed in Table 3.

From Table 3, we can see that the *s* score is positively correlated with all these codon-based indices. Especially, the correlation coefficient of the *s* score and the "codon usage" general expression level measure $E(g)$ is 0.43, which means that those predicted highly expressed genes by "codon usage" model usually also have high *s* scores. Calculation shows that the mean value of the *s* score averaged over the set of 306 PHX genes is 42.65, much higher than the mean value 18.56 averaged over the rest 3983 genes. On the other hand, the correlation coefficient between the *s* score and any of these codon-based indices is not more than 0.5, which means that the *s* score, as a single-nucleotide-level measure, is highly complementary to the codon-based indices on the description of "codingness" and expression level of ORFs.

Table 3
Correlation coefficients of a variety of scoring models between each other for the 306 PHX genes of *E. coli* K12

	CBI	CAI	$E(g)$	s-Score
CBI	1.0000			
CAI	0.9198	1.0000		
$E(g)$	0.7978	0.8012	1.0000	
s-Score	0.3026	0.3063	0.4361	1.0000

5.4. Position weight function depends on ACGT map

The accuracy (69.25%) listed in the third row of Table 2 is obtained under the map A → 1, C → 2, G → 3, T → 4; the accuracy (86.01%) listed in the sixth row of Table 2 is obtained under the map A → 1, C → 3, G → 2, T → 4. Comparing the two rows, it is easy to find that the accuracy under each ACGT map is remarkably different from that under another, which shows the dependency of a position weight function on the ACGT map in the efficiency of reflecting permutation information. In fact, if we preset a position weight function, we can also obtain a special ACGT map that is shared by genes rather than other permutations, which means that ACGT map also depends on a preset position weight function. Here arise the problems: what are the best map and the best position weight function and how can we get them? We will propose a refinement method in the future for the ACGT map and the position weight function based on their interdependence so as to gain an optimum description of protein coding gene.

6. Conclusion

One purpose of this paper is to draw attention to the task of describing genes. By transforming DNA sequences into quaternary numbers, we presented a trial implementation for this task based on a generalized position weight function which is shared by protein coding genes. By trigonometric approximation of this function, the common features of genes are revealed as the single nucleotide periodicities. The results show that different species may have different strengths of single nucleotide periodicities and for the bacteria *E. coli* K12 and the eukaryote yeast, periodicities 3, 5 and 7 are the strongest signals. As a byproduct of this approach, a single-nucleotide-level measure is derived that complements codon-based indexes in describing "codingness" and expression level of ORF. The implementation presented here has the potential to be developed into a general methodology for analyzing biological sequences.

Acknowledgements

Heng Wei, Hong-Yu Ou and Yun-Tao Dou gave helpful discussions. Many thanks to Prof. Hong-Yu Zhang for his help in polishing this paper. We are grateful to the anonymous reviewer for his suggestions that make this paper clearer for understanding. This work was supported by the National Basic Research Program of China (2003CB114400) and the National Natural Science Foundation of China (30100035 and 30570383).

References

Atsushi, F., Toshimichi, I., Makoto, K., Taku, O., Yoshihiro, K., Hirotada, M., Shigehiko, K., 2002. Periodicity in prokaryotic and eukaryotic genomes identified by power spectrum analysis. Gene 300, 203–211.

Bajic, V.B., 2000. Comparing the success of different prediction software in sequence analysis: a review. Brief Bioinform. 1, 214–228.

Bennetzen, J.L., Hall, B.D., 1982. Codon selection in yeast. J. Biol. Chem. 257, 3026–3031.

Borodovsky, M., McIninch, J., 1993. GENMARK: parallel gene recognition for both DNA strands. Comput. Chem. 17, 123–133.

Burset, M., Guigo, R., 1996. Evaluation of gene structure prediction programs. Genomics 34, 353–367.

Cardon, L.R., Stormo, G.D., 1992. Expectation maximization algorithm for identifying protein-binding sites with variable lengths from unaligned DNA fragments. J. Mol. Biol. 223, 159–170.

Dong, S., Searls, D.B., 1994. Gene structure prediction by linguistic methods. Genomics 23, 540–551.

Fickett, J.W., 1996. The gene identification problem: an overview for developers. Comput. Chem. 20, 103–118.

Guigo, R., 1997. Computational gene identification: an open problem. Comput. Chem. 21, 215–222.

Gutierrez, G., Oliver, J.L., Marin, A., 1994. On the origin of the periodicity of three in protein coding DNA sequences. J. Theor. Biol. 167, 413–414.

Hamori, E., Ruskin, J., 1983. H curves, a novel method of representation of nucleotide series especially suited for long DNA sequences. J. Biol. Chem. 258, 1318–1327.

Jansen, R., Bussemaker, H.J., Gerstein, M., 2003. Revisiting the codon adaptation index from a whole-genome perspective: analyzing the relationship between gene expression and codon occurrence in yeast using a variety of models. Nucl. Acids Res. 31, 2242–2251.

Karlin, S., Mrazek, J., Campbell, A.M., 1998. Codon usages in different gene classes of the *Escherichia coli* genome. Mol. Microbiol. 29, 1341–1355.

Karlin, S., Mrazek, J., 2000. Predicted highly expressed genes of diverse prokaryotic genomes. J. Bacteriol. 182, 5238–5250.

Lukashin, A.V., Borodovsky, M., 1998. GeneMark.hmm: new solutions for gene finding. Nucl. Acids Res. 26, 1107–1115.

Mathe, C., Sagot, M.F., Schiex, T., Rouze, P., 2002. Current methods of gene prediction, their strengths and weaknesses. Nucl. Acids Res. 30, 4103–4117.

Peden, 1999. PhD Thesis. At http://www.molbiol.ox.ac.uk/cu or http://codonw.sourceforge.net/JohnPedenThesisPressOpt_water.pdf.

Pesole, G., Attimonelli, M., Saccone, C., 1996. Linguistic analysis of nucleotide sequences: algorithms for pattern recognition and analysis of codon strategy. Meth. Enzymol. 266, 281–294.

Rogic, S., Mackworth, A.K., Ouellette, F.B., 2001. Evaluation of gene-finding programs on mammalian sequences. Genome Res. 11, 817–832.

Salzberg, S.L., Delcher, A.L., Kasif, S., White, O., 1998. Microbial gene identification using interpolated Markov models. Nucl. Acids Res. 26, 544–548.

Sharp, P.M., Li, W.H., 1987. The codon adaptation index—a measure of directional synonymous codon usage bias, and its potential applications. Nucl. Acids Res. 15, 1281–1295.

Shepherd, J.C., 1981. Periodic correlations in DNA sequences and evidence suggesting their evolutionary origin in a comma-less genetic code. J. Mol. Evol. 17, 94–102.

Tiwari, S., Ramachandran, S., Bhattacharya, A., Bhattacharya, S., Ramaswamy, R., 1997. Prediction of probable genes by Fourier analysis of genomic sequences. Comput. Appl. Biosci. 13, 263–270.

Tomita, M., Wada, M., Kawashima, Y., 1999. ApA dinucleotide periodicity in prokaryote, eukaryote, and organelle genomes. J. Mol. Evol. 49, 182–192.

Trifonov, E.N., Sussman, J.L., 1980. The pitch of chromatin DNA is reflected in its nucleotide sequence. Proc. Natl. Acad. Sci. U.S.A. 77, 3816–3820.

Tsonis, A.A., Kumar, P., Elsner, J.B., Tsonis, P.A., 1996. Wavelet analysis of DNA sequences. Phys. Rev. E 53, 1828–1834.

Zhang, R., Zhang, C.T., 1994. Z curves, an intutive tool for visualizing and analyzing the DNA sequences. J. Biomol. Struct. Dyn. 11, 767–782.

Zhang, C.T., Wang, J., 2000. Recognition of protein coding genes in the yeast genome at better than 95% accuracy based on the *Z* curve. Nucl. Acids Res. 28, 2804–2814.