MATCH
Communications in Mathematical and in Computer Chemistry

MATCH Commun. Math. Comput. Chem. **67** (2012) 845-866

ISSN 0340 - 6253

Discriminate the Falsely Predicted Protein–Coding Genes in *Aeropyrum Pernix* K1 Genome Based on Graphical Representation

Jia-Feng Yu [a, b], Dong-Ke Jiang [a], Ke Xiao [a], Yun Jin [a], Ji-Hua Wang [b], Xiao Sun [a, *]

[a] *State Key Laboratory of Bioelectronics, School of Biological Science and Medical Engineering, Southeast University, Nanjing 210096, P. R. China*

[b] *Shandong Province Key Laboratory of Biophysics for Functional Macromolecules, Department of Physics, Dezhou University, Dezhou 253023, P. R. China*

(Received May 10, 2011)

Abstract

The problem that how many protein-coding genes exist in *Aeropyrum pernix* K1 genome has confused many scientists since 1999. In this paper, we attempt to re-identify the protein-coding genes in this genome by proposing a modified method based on I-TN curve. Consequently, all of the 727 experimentally validated protein-coding genes and 726 of the corresponding negative samples are correctly predicted respectively, then an accuracy of 99.93% of self-test is obtained. In the Jackknife test, two positive samples and two negative samples are falsely predicted, respectively, and then the accuracy of cross-validation is 99.72%. In the testing set, all of the 132 putative genes are correctly predicted as protein-coding and 14 out of the 841 hypothetical genes are predicted as non-coding, the number of protein-coding genes is reduced to 1686 instead of 1700. Further analysis shows the performance of the reannotating algorithm is comparable to other prevalent programs, and the present method is much simple and efficient. We implement the reannotating algorithm trained by *Aeropyrum pernix* K1 to *Chlorobium tepidum* TLS genome, and 217 hypothetical genes are predicted as non-coding. Sufficient sequences analysis indicates most of them are random sequences that are falsely predicted as protein-coding genes. In addition, we also perform some significative analysis aiming to the influence of artificial parameters on the graphical representation approaches, which may provide helpful information for related researches.

* Corresponding author. E-mail: jfyu1979@126.com (J.F. Yu), xsun@seu.edu.cn (X. Sun).

1. INTRODUCTION

The number of sequenced microbial genomes stored in public databases increases explosively with the development of sequencing techniques. In most cases, many people take it for granted that gene finding in prokaryotic genomes is relatively easy due to the fact lacking of introns, whereas more and more researches indicate the issue of gene finding in microbial genomes is far from thoroughly resolved, the annotation quality of microbial genomes has been questioned continuously [1, 2]. In most microbial genomes, it is found some annotated genes do not encode proteins actually, but rather open reading frames that occur by chance [2]. In recent work by Luo et al. [3], 608 protein-coding sequences are excluded from current RefSeq annotation by performing reannotation on *Escherichia coli* CFT073 genome. Hence, over-annotation of protein-coding genes in microbial genomes has been recognized as a serious problem currently. On the other hand, many users deem that all the annotated genes as true, which can lead to wrong conclusions. Then how to improve the annotation quality of proteins encoded in each genome is an important task. Fortunately, some groups [4-15] have carried out different methods to reannotate protein-coding genes in microbial genomes in the past several years.

Aeropyrum pernix K1 (*A. pernix* K1) is a kind of Archaea that grows optimally at 90 to 95 ºC [16]. In the pioneer annotation, 2694 ORFs were predicted as potential genes [17], and its remarkable gene density attracts more and more researchers [18]. Since then, the serious problem of over-prediction of the protein-coding genes in *A. pernix* K1 genome was commonly recognized, many algorithms have been proposed to estimate the total number of its protein-coding genes, and the accepted number of protein-coding genes is estimated from 1400 to 1871 [4, 6, 9-11]. Afterwards, the NITE researchers modified the annotation of *A. pernix* K1 by proteome analysis in 2006 [14]. As a consequent, 1700 ORFs were annotated as potential genes in the current NITE annotation. In 2009, Guo and Lin proposed an Aper_ORFs method to re-identify the protein-coding genes in *A. pernix* K1 genome and they predicted 28 ORFs as non-coding from the 1700 annotated ORFs [8]. Among these works, similarity search, statistics discrimination and parameter estimation were respectively used. Nevertheless, due to the different features of these methods, their results are much different.

Therefore, there is still debate on how many protein-coding genes existing in *A. pernix* K1. In this paper, we attempt to put forward an alternative approach for discriminating the falsely predicted protein-coding genes in *A. pernix* K1 genome based on I-TN curve [15]. Considering the complexity of archaeal genomes, we derive a 36-D numerical vector to demonstrate the specific features of protein-coding genes in *A. pernix* K1 genome in the present work. Consequently, an accuracy of 99.72% for Jackknife test is achieved and 14 annotated potential protein-coding genes are predicted as non-coding. The present algorithm can also provide efficient tools for other microbial genomes.

2. MATERIALS AND METHODS

2.1. The I-TN curve

Graphical representation is a kind of simple and efficient method that has been extensively applied in researches of DNA [19-25], RNA [26] and protein [27-28] sequences. The present reannotation algorithm is based on I-TN curve, which is a specifically designed graphical representation for protein-coding genes. According to I-TN curve, each kind of trinucleotide is uniquely represented by a point (x, y, z) in a 3-D space, where, $z = x \times y$, and (x, y) are defined as follows. For a trinucleotide, the signs of x and y are determined by the category of the base at the third position, i.e., {+, +}→A, {–,+}→G, {–,–}→C and {+,–}→T, the absolute values of x and y are decided by the bases at the first and second positions, respectively, i.e., 1→A, 2→G, 3→C, 4→T. Thus, each kind of trinucleotide (which is also called codon in protein-coding gene) is numerically denoted by a 2-D coordinate (x, y), which helps to discriminate each trinucleotide intuitively. Taking (-2, 3) for example, the negative sign of x and the positive sign of y denote the base at the third position is G; the absolute values of x and y are 2 and 3, which denote the bases at the first and second sites are G and C, respectively. Therefore, (-2, 3) represents the trinucleotide of GCG.

For an arbitrary DNA sequence $S = s_1s_2s_3s_4\ldots s_N$ with the length of N, we have a map ϕ, which can map S into a plot set $\phi(S) = \phi(s_1s_2s_3)\ \phi(s_2s_3s_4)\ldots\phi(s_ns_{n+1}s_{n+2})$, where $\phi(s_ns_{n+1}s_{n+2})=(x_n, y_n, z_n)$, n = 1,2,3,…, N-2. The curve connected all plots of the characteristic

plot set in turn is called I-TN curve. It is noted that when $z>0$, x and y are positive or negative simultaneously. As has been introduced in our previous work [20], the 64 trinucleotides can be classified into two groups by x, y and z according to the physiochemical properties of the third base in three ways, respectively. Defining

$$x'_n = \sum_{i=1}^{n} x_i \,, \quad y'_n = \sum_{i=1}^{n} y_i \quad \text{and} \quad z'_n = \sum_{i=1}^{n} z_i \,.$$

Then, the biological significances of x'_n, y'_n and z'_n can be interpreted by the following equations.

$$x'_n \rightarrow N_{B_1B_2A} + N_{B_1B_2T} - N_{B_1B_2G} - N_{B_1B_2C}$$
$$y'_n \rightarrow N_{B_1B_2A} + N_{B_1B_2G} - N_{B_1B_2C} - N_{B_1B_2T}$$
$$z'_n \rightarrow N_{B_1B_2A} + N_{B_1B_2C} - N_{B_1B_2G} - N_{B_1B_2T}$$

Where B_1, $B_2 \in$ {A, G, C, T} are arbitrary bases at the first and second codon positions, respectively, N_{B1B2B3} is the cumulative occurring numbers of trinucleotide $B_1B_2B_3$ walking along corresponding sequence. Then x'_n, y'_n and z'_n display the cumulative effects of x, y and z, respectively. Note that "→" is used instead of "=" here, for the polynomials on both sides of "→" are not equivalent quantitatively because of the initial numerical assignments of I-TN curve.

Previous researches suggested the first and second bases determine the category of translated amino acid, while the third base is associated with synonymous codon [29, 30]. Because of the unevenly distribution of synonymous codons, protein-coding genes are different from non-coding sequences in gene structure. This non-random usage of codons can be used to find protein-coding sequences [31]. Since many properties of protein-coding genes are related to the base at the third position of codon, I-TN curve may play some specific roles in related researches of protein-coding gene analysis.

2.2. Numeric descriptors for protein-coding gene

How to find efficient quantitative descriptors for protein-coding genes is the core of gene prediction programs. The difference between protein-coding genes and non-coding sequences lies in the former has regularly specific features such as asymmetric nucleotide distributions

at the three codon positions and codon usage bias, while the latter does not. Comparing with viral genomes and phage genomes, there are much more functional genes in archaeal genomes with diverse gene structures caused by many influence factors such as G+C content, gene expressivity, horizontal gene transfer …. Then the numeric descriptors adopted for gene prediction must have the ability to demonstrate the universal commonness of protein-coding genes. On the other hand, some species-specific genes are likely to be missed by using similarity search methods. Therefore, discrimination protein-coding genes from non-coding cannot be merely attributed to detect sequence similarity among these ORFs. In our previous works, we proposed an approach based on 18-D vector to re-annotate the protein-coding genes in viral genomes. Here, considering the complexities of archaeal genome, we attempt to deduce a 36-D instead of the 18-D vector to display the specific features of protein-coding genes in *A. pernix* K1 genome.

As is well known, there are three forward reading frames and three reverse reading frames in a protein-coding gene sequence. The six reading frames lead to six possible protein-coding sequences, of which usually only one is likely to encode protein sequence. Supposing sequence $S = s_1s_2s_3s_4s_5s_6s_7s_8\ldots s_{N-5}s_{N-4}s_{N-3}s_{N-2}s_{N-1}s_N$ is a protein-coding gene, the three forward frames are $\{s_1s_2s_3,\ s_4s_5s_6,\ s_7s_8\ldots\}$, $\{s_2s_3s_4,\ s_5s_6s_7,\ s_8\ldots\}$, $\{s_3s_4s_5,\ s_6s_7s_8,\ \ldots\}$ and the three reverse frames are $\{s_Ns_{N-1}s_{N-2},\ s_{N-3}s_{N-4}s_{N-5},\ \ldots\}$, $\{s_{N-1}s_{N-2}s_{N-3},\ s_{N-4}s_{N-5}s_{N-6},\ \ldots\}$, $\{s_{N-2}s_{N-3}s_{N-4},\ s_{N-5}s_{N-6}s_{N-7},\ \ldots\}$, respectively. For the first forward frame $\{s_1s_2s_3,\ s_4s_5s_6,\ s_7s_8\ldots\}$, each trinucleotide can be numerically denoted by the following equations in turn,

$$\begin{cases} x^{\{+0\}} = \{x_i\} \\ y^{\{+0\}} = \{y_i\} \\ z^{\{+0\}} = \{z_i\} \end{cases}.$$

Where i = 1, 2, 3… denote the trnucleotide numbers in the first forward reading frame +0. Similarly, defining

$$x_j^{\{+0\}\prime} = \sum_{k=1}^{j} x_k^{\{+0\}},\quad y_j^{\{+0\}\prime} = \sum_{k=1}^{j} y_k^{\{+0\}} \quad \text{and} \quad z_j^{\{+0\}\prime} = \sum_{k=1}^{j} z_k^{\{+0\}}$$

to represent the cumulative effects of $x^{\{+0\}}$, $y^{\{+0\}}$ and $z^{\{+0\}}$, respectively, j=1, 2, 3…, i. In this way, a 6-D vector V_1 is obtained to quantitatively describe frame +0, i.e. the mean values of

$x^{\{+0\}}$, $y^{\{+0\}}$, $z^{\{+0\}}$ as well as their cumulative effects $x^{\{+0\}}$', $y^{\{+0\}}$', $z^{\{+0\}}$', respectively. Thereafter, in the same way, we have 6 × 6 = 36 numeric descriptors for a gene sequence, which are denoted as follows.

$$V_1 = \begin{cases} u_1 = (\sum_{i=1}^{N^{\{+0\}}} x_i^{\{+0\}}) / N^{\{+0\}}, u_4 = (\sum_{i=1}^{N^{\{+0\}}} x^{\{+0\}}') / N^{\{+0\}} \\ u_2 = (\sum_{i=1}^{N^{\{+0\}}} y_i^{\{+0\}}) / N^{\{+0\}}, u_5 = (\sum_{i=1}^{N^{\{+0\}}} y^{\{+0\}}') / N^{\{+0\}} \\ u_3 = (\sum_{i=1}^{N^{\{+0\}}} z_i^{\{+0\}}) / N^{\{+0\}}, u_6 = (\sum_{i=1}^{N^{\{+0\}}} z^{\{+0\}}') / N^{\{+0\}} \end{cases},$$

$$V_2 = \begin{cases} u_7 = (\sum_{i=1}^{N^{\{+1\}}} x_i^{\{+1\}}) / N^{\{+1\}}, u_{10} = (\sum_{i=1}^{N^{\{+1\}}} x^{\{+1\}}') / N^{\{+1\}} \\ u_8 = (\sum_{i=1}^{N^{\{+1\}}} y_i^{\{+1\}}) / N^{\{+1\}}, u_{11} = (\sum_{i=1}^{N^{\{+1\}}} y^{\{+1\}}') / N^{\{+1\}} \\ u_9 = (\sum_{i=1}^{N^{\{+1\}}} z_i^{\{+1\}}) / N^{\{+1\}}, u_{12} = (\sum_{i=1}^{N^{\{+1\}}} z^{\{+1\}}') / N^{\{+1\}} \end{cases},$$

$$V_3 = \begin{cases} u_{13} = (\sum_{i=1}^{N^{\{+2\}}} x_i^{\{+2\}}) / N^{\{+2\}}, u_{16} = (\sum_{i=1}^{N^{\{+2\}}} x^{\{+2\}}') / N^{\{+2\}} \\ u_{14} = (\sum_{i=1}^{N^{\{+2\}}} y_i^{\{+2\}}) / N^{\{+2\}}, u_{17} = (\sum_{i=1}^{N^{\{+2\}}} y^{\{+2\}}') / N^{\{+2\}} \\ u_{15} = (\sum_{i=1}^{N^{\{+2\}}} z_i^{\{+2\}}) / N^{\{+2\}}, u_{18} = (\sum_{i=1}^{N^{\{+2\}}} z^{\{+2\}}') / N^{\{+2\}} \end{cases},$$

$$V_4 = \begin{cases} u_{19} = (\sum_{i=1}^{N^{\{-0\}}} x_i^{\{-0\}}) / N^{\{-0\}}, u_{22} = (\sum_{i=1}^{N^{\{-0\}}} x^{\{-0\}}') / N^{\{-0\}} \\ u_{20} = (\sum_{i=1}^{N^{\{-0\}}} y_i^{\{-0\}}) / N^{\{-0\}}, u_{23} = (\sum_{i=1}^{N^{\{-0\}}} y^{\{-0\}}') / N^{\{-0\}} \\ u_{21} = (\sum_{i=1}^{N^{\{-0\}}} z_i^{\{-0\}}) / N^{\{-0\}}, u_{24} = (\sum_{i=1}^{N^{\{-0\}}} z^{\{-0\}}') / N^{\{-0\}} \end{cases},$$

$$V_5 = \begin{cases} u_{25} = (\sum_{i=1}^{N^{\{-1\}}} x_i^{\{-1\}}) / N^{\{-1\}}, u_{28} = (\sum_{i=1}^{N^{\{-1\}}} x^{\{-1\}\prime}) / N^{\{-1\}} \\ u_{26} = (\sum_{i=1}^{N^{\{-1\}}} y_i^{\{-1\}}) / N^{\{-1\}}, u_{29} = (\sum_{i=1}^{N^{\{-1\}}} y^{\{-1\}\prime}) / N^{\{-1\}} \\ u_{27} = (\sum_{i=1}^{N^{\{-1\}}} z_i^{\{-1\}}) / N^{\{-1\}}, u_{30} = (\sum_{i=1}^{N^{\{-1\}}} z^{\{-1\}\prime}) / N^{\{-1\}} \end{cases},$$

$$V_6 = \begin{cases} u_{31} = (\sum_{i=1}^{N^{\{-2\}}} x_i^{\{-2\}}) / N^{\{-2\}}, u_{34} = (\sum_{i=1}^{N^{\{-2\}}} x^{\{-2\}\prime}) / N^{\{-2\}} \\ u_{32} = (\sum_{i=1}^{N^{\{-2\}}} y_i^{\{-2\}}) / N^{\{-2\}}, u_{35} = (\sum_{i=1}^{N^{\{-2\}}} y^{\{-2\}\prime}) / N^{\{-2\}} \\ u_{33} = (\sum_{i=1}^{N^{\{-2\}}} z_i^{\{-2\}}) / N^{\{-2\}}, u_{36} = (\sum_{i=1}^{N^{\{-2\}}} z^{\{-2\}\prime}) / N^{\{-2\}} \end{cases}.$$

Here, $N^{\{+0\}}$, $N^{\{+1\}}$, …, $N^{\{-2\}}$ denotes the total number of trinucleotides in each reading frame. Consequently, the 36-D vector $V = V_1 \oplus V_2 \oplus V_3 \oplus V_4 \oplus V_5 \oplus V_6$ can be used to quantitatively describe a complete protein-coding gene. Comparing with the statistic methods, the 36 parameters u_1, u_2, u_3 …u_{36} are easily calculated by calculating the mean value of each variable, which corresponds to the geometric center of each 2-D curve [20], therefore the running time should be shortened greatly.

2.3. The Fisher discriminant algorithm

The Fisher discriminant algorithm is a simple and efficient method that has been extensively used in gene prediction. For detail introductions, refer to [32]. In the present work, two groups of samples are required, i.e. positive samples corresponding to true protein-coding genes and negative samples corresponding to non-coding ORFs, which are used to train the discriminant coefficients. In microbial genomes, the amount of non-coding DNA sequences is too few to be used. The negative samples generated by shuffling the primary sequences and the complementary sequences of the shuffled sequences are used as non-coding sequences [33]. Thus, the coding and the non-coding sequences have the same length, but with different base composition. The Fisher linear equation for discriminating the positive and negative samples in the 36-D space V represents a super-plane, described by a

vector C that has 36 components. To avoid loss of generality, the vector C was determined according to the criterion $|C|^2=1$. Based on the training set, an appropriate threshold C_0 can be obtained by letting the false negative rate and the false positive rate be identical. However, it is rather a difficult problem to determine the appropriate threshold C_0 because there are so many values meeting this demands. In the present work, we calculate C_0 by the following steps: (1) Removing the minimum point of $C \cdot V$ in the positive samples. (2) Removing the maximum point of $C \cdot V$ in the negative samples. In the new point sets for positive samples and negative samples, we can obtain the unique threshold C_0 by $C_0 = (Max^N + Min^P) / 2$, where Max^N and Min^P are the maximum and minimum of the new point sets of negative samples and positive samples, respectively. Once the vector C and the threshold C_0 are determined, each sequence is assigned a $T\_score = C*V-C_0$. Then the decision of coding/non-coding for each genes in the test set is simply performed by the criterion of $T\_score > 0$ or $T\_score < 0$, where $C = (C_1, C_2, \ldots, C_{36})$ and $V=(u_1, u_2 \ldots, u_{36})$.

2.4. Evaluation index

The accuracy, sensitivity and specificity used in the present study to evaluate the performance are the same by Burset and Guigo [34]. Using TP and FN to denote the number of coding ORFs that have been predicted as coding and non-coding, respectively, the sensitivity s_n is defined as $s_n=TP/(TP+FN)$. That is, s_n is the proportion of the coding ORFs that have been predicted correctly as coding sequences. Similarly, TN and FP denote the number of non-coding sequences that have been predicted as coding and non-coding sequences, respectively. The specificity s_p is defined as $s_p=TN/(TN+FP)$. That is, s_p is the proportion of the non-coding sequences that have been correctly predicted as non-coding. The accuracy is defined as the average of s_n and s_p.

3. RESULTS AND DISCUSSIONS

3.1. Assessment of the predicting algorithm and reannotating results of A. pernix K1 genome

The detailed annotating information of *A. pernix* K1 genome was downloaded from RefSeq [35]. The G+C content among the 1700 annotated potential protein-coding genes ranges from 32.6% to 72.4%. Among the 1700 annotated genes, 727 have validated functions, 132 are marked as putative genes, and the rest 841 are marked as hypothetical genes. For convenience, we divide all the annotated genes into three classes according to their functions, the 727 function-known genes belong to the first class, the 132 putative genes compose the second class and the third class comprises 841 hypothetical genes. The former two classes can be regarded as true protein-coding genes, while some in the third class need to be further validated. The 727 genes in the first class and their corresponding complementary shuffled sequences are used to train the Fisher coefficients, which are also used to evaluate the performance of the gene-finding algorithm, this should be regarded as self-consistency test. Consequently, the 727 function-known genes and 726 shuffled sequences are correctly predicted as coding and non-coding, respectively. Then the sensitivity and specificity of self-consistency test are 727/727 = 100%, 726/727=99.86%, respectively, and the accuracy is 99.93%. Table 1 presents the values of the trained coefficients and the threshold C_0. Using the Fisher coefficients trained by the first class and the criterion $T\_score > 0$ or $T\_score < 0$ for making the coding/non-coding decision, the genes from the second class are re-identified, which should be regarded as cross-validation test. Consequently, all of the 132 putative genes are correctly predicted as protein-coding. Afterwards, the present algorithm is used to identify the 841 genes in the third class. Consequently, 14 annotated potential protein-coding ORFs in the third class are recognized as non-coding, which are presented in column 2 of Table 2. Then the number of protein-coding genes in *A. pernix* K1 is reduced to 1700-14=1686.

The evaluation mentioned above can be used to check the self-consistency of a predictor, especially for its algorithm part. A predictor cannot be deemed a good one if its self-consistency rate is poor. However, the self-test is useful but not sufficient for evaluating a

predictor effectively in most cases. In statistical prediction, there are three methods usually used for cross-validation, namely, the sub-sampling test, independent dataset test and Jackknife test. Among these tests, the Jackknife test is deemed the most objective that can always yield a unique result for a given benchmark dataset [36], and hence has been increasingly and widely used by investigators to examine the accuracy of various predictors. In the Jackknife test, each of the positive and negative samples in the training set mentioned above is singled out in turn as a tested item, and the remaining genes train the predictor. Therefore, both the training dataset and the testing dataset are actually open, and a gene will in turn move from one dataset to the other. Thus, the Jackknife test can exclude the memory effects that exist in the self-test, and the results obtained in this way always are unique for a given dataset. By performing the Jackknife test, two items in the positive and negative samples are falsely predicted respectively, the two falsely predicted positive genes are APE_1588b and APE_1929.1, then the sensitivity = 725/727 = 99.72% and specificity = 725/727 = 99.72%, respectively, and the accuracy of Jackknife test is 99.72%.

Table 1: The trained Fisher coefficients C and the threshold C_0.

Fisher coefficients	**values**	**Fisher coefficients**	**values**	**Fisher coefficients**	**values**	**Fisher coefficients**	**values**
C_1	-0.135	C_{11}	0.0003	C_{21}	-0.043	C_{31}	0.039
C_2	0.4629	C_{12}	-0.0002	C_{22}	-0.0003	C_{32}	-0.1098
C_3	0.0909	C_{13}	0.0986	C_{23}	-0.0004	C_{33}	-0.0409
C_4	-0.0002	C_{14}	-0.0373	C_{24}	0.0001	C_{34}	0.0001
C_5	-0.0001	C_{15}	0.0086	C_{25}	-0.015	C_{35}	0.0001
C_6	0	C_{16}	0.0004	C_{26}	-0.4528	C_{36}	0
C_7	-0.0454	C_{17}	-0.0002	C_{27}	-0.0823	C_0	0.2204
C_8	0.6794	C_{18}	0	C_{28}	0.0005		
C_9	0.0636	C_{19}	0.0397	C_{29}	0.0001		
C_{10}	-0.0001	C_{20}	-0.2228	C_{30}	0.0001		

Table 2: The recognized non-coding sequences based on the present method and Aper_ORFs method. The items marked with F and P denotes function-known genes and putative genes, respectively. The bold items denote the common ORFs obtained by both methods.

Method	36-D vector		Aper_ORFs			
ID	APE_0242.1	APE_1209d	APE_0031.1^P	APE_0471c	APE_0996a	APE_1909.1
	APE_0416a	**APE_1275c**	APE_0047.1^F	APE_0722c	APE_1029^F	APE_2037.1^F
	APE_0470a	APE_1473a	APE_0054.1	APE_0862.1	APE_1177.1	**APE_2065.1**

APE_0762.1	APE_1882a	APE_0156.1	**APE_0885b**	**APE_1275c**	APE_2242a
APE_0816a.1	**APE_2065.1**	APE_0270.1^F	APE_0941^F	APE_1277	**APE_2284a**
APE_0885b	**APE_2284a**	APE_0334	**APE_0954a**	APE_1633^F	APE_2426.1^F
APE_0954a	APE_2480a	**APE_0416a**	APE_0965.1	APE_1840.1	APE_2567

3.2. Comparing the present approach with other algorithms

The original annotation of *A. pernix* K1 genome did not use statistic methods, but employed a similarity search in 1999 that was not nearly as big as it is now. Then it is interesting to compare our gene-reannotating algorithm with other statistic programs based on *A. pernix* K1. Among those reported statistic programs, Glimmer [37] and GeneMark [38] are the two most popular gene-finding programs that are based on HMM methods, which have been broadly used in the annotation systems. After running the two programs (Glimmer 3.02 and GeneMark.hmm 2.04) on *A. pernix* K1 genome, 1789 and 1736 ORFs are predicted as protein coding genes, respectively. Among the 727 function known genes, Glimmer correctly predicts 721 items and GeneMark correctly predicts 726 items, which means that the accuracies for Glimmer and GeneMark are 99.17% and 99.86%, respectively. In addition, Glimmer products 117 additional genes while GeneMark gives 65 additional genes. Besides, many research groups have proposed different methods for reannotation of protein-coding genes in *A. pernix* K1 genome recently. Among these approaches, the Aper_ORFs method proposed by Guo and Lin [8] is specially devised for *A. pernix* K1, which is based on the versatile Z curve method, and then its results are more accurate comparing with others. Based on the Aper_ORFs method, 28 ORFs are predicted as non-coding. For comparison, in the column 3 of Table 2, we present the 28 predicted non-coding ORFs by Aper_ORFs method, which is retrieved from http://tubic.tju.edu.cn/Aper/. In our recent work, we have proposed an 18-D vector to reannotate the protein-coding genes in viral genome [15]. Here, we also compare the present modified method with the 18-D vector. In Table 3, we present the predicting performances obtained by the present approach and Glimmer, GeneMark, Aper_ORFs method and the 18-D vector. Obviously, the present approach can achieve a comparable performance level with these eminent algorithms, such as Glimmer, GeneMark, and Z curve based method, which indicates the 36-D vector can display the specific gene

features excellently. On the other hand, though high accuracies (>99%) can be achieved by different approaches, it seems that there are some differences among these prediction results. As mentioned above, the accuracy of Glimmer is only 0.7 percent lower that that of GeneMark, but its additional genes doubles (117 vs. 65) over the latter. According to the results listed in Table 2, there are only six common items obtained by the present approach and the Aper_ORFs method. For convenience of comparison, we mark corresponding genes with *F* and *P* according to their annotation status, which denote the function-known gene and the putative gene, respectively. As can be seen, all the function known and putative genes are correctly predicted as protein-coding by the present method, while in the results based on Aper_ORFs method, seven function known genes and one putative gene are falsely predicted as non-coding. Then how to propose much more reliable algorithms especially for reannotation of protein coding genes seems to be much necessary in the future, besides the high sensitivity.

Table 3: Comparing the present approach with other programs.

Methods	36-D vector	18-D vector	Aper_ORFs	GeneMark	Glimmer
Accuracy (%)	99.93	99.79	99.04	99.86	99.17

Notations: During the course of the submission of this paper, the careful NITE staffs updated the annotation file of *A.pernix* K1 genome in RefSeq, where 27 putative and hypothetical genes are assigned validated functions, then the number of protein coding genes is expanded to 754. It is noted that the 27 added function known genes are all correctly predicted as protein coding by the present method.

3.3. Extending the present approach to the C. tepidum TLS Genome

As have been discussed, the parameters derived from I-TN curve can display the information of base distributions at different codon positions and the correlativity of adjacent nucleotides in DNA sequences. Therefore, it is conceivable that the present method may be extended to other archaeal genomes, especially with similar G+C content to that of the *A. pernix* K1 genome without additional calculations. Here, *Chlorobium tepidum* TLS (*C. tepidum* TLS), the genomic G+C content of which is also about 56%, is chosen to be reannotated by using the coefficients generated from *A. pernix* K1.

Table 4: The 14 ORFs that are falsely recognized as non-coding in the *C. tepidum* TLS genome. The two genes marked with an asterisk have putative functions and the retaining ones have known function.

CT0103	CT0996	CT1289	CT1802	CT2166
CT0301	CT1040*	CT1342*	CT2019	CT2224
CT0409	CT1174	CT1461	CT2024	

Among the 2245 annotated potential protein-coding ORFs in the *C. tepidum TLS* genome, 1123 are assigned with known-functions, and the functions of 176 are putative, the others are hypothetical genes. Based on the parameters presented in Table 1, the 1123+176=1299 function-validated genes and 946 hypothetical genes are reannotated. Consequently, 1285 function-known and putative genes are correctly predicted as coding. That means the sensitivity of the cross-validation tests are 98.92%. Names of 14 falsely predicted known genes are listed in Table 4. On the other hand, 217 out of the 946 hypothetical ORFs are identified as non-coding. Names of the 217 ORFs are presented in Table 5. Guo and Lin [8] also performed reannotation in *C. tepidum* TLS genome based on Aper_ORFs method, whereas 30 out of 1296 protein-coding genes are falsely predicted as non-coding, then the sensitivity of their work is 1266/1296=97.69%.

Table 5: The 217 hypothetical ORFs identified as non-coding in *C. tepidum* TLS genome.

CT0005	CT0382	CT0573	CT0783	CT0962	CT1210	CT1452	CT1604	CT1872	CT2068
CT0025	CT0396	CT0579	CT0787	CT0967	CT1216	CT1458	CT1617	CT1875	CT2093
CT0044	CT0405	CT0582	CT0788	CT0997	CT1217	CT1460	CT1623	CT1884	CT2094
CT0074	CT0407	CT0583	CT0793	CT1014	CT1223	CT1467	CT1643	CT1902	CT2097
CT0096	CT0449	CT0584	CT0794	CT1022	CT1230	CT1476	CT1651	CT1904	CT2108
CT0101	CT0454	CT0585	CT0820	CT1024	CT1233	CT1481	CT1653	CT1905	CT2110
CT0115	CT0475	CT0596	CT0837	CT1047	CT1251	CT1490	CT1684	CT1911	CT2148
CT0143	CT0482	CT0608	CT0848	CT1056	CT1254	CT1496	CT1686	CT1912	CT2149
CT0165	CT0489	CT0627	CT0849	CT1062	CT1262	CT1515	CT1687	CT1916	CT2157
CT0174	CT0497	CT0628	CT0858	CT1081	CT1264	CT1517	CT1693	CT1917	CT2195
CT0210	CT0500	CT0639	CT0870	CT1083	CT1282	CT1518	CT1694	CT1926	CT2199
CT0218	CT0506	CT0645	CT0871	CT1086	CT1319	CT1520	CT1720	CT1927	CT2201
CT0231	CT0508	CT0671	CT0877	CT1093	CT1354	CT1523	CT1764	CT1933	CT2203
CT0234	CT0510	CT0673	CT0886	CT1104	CT1355	CT1531	CT1789	CT1941	CT2218
CT0271	CT0513	CT0688	CT0888	CT1107	CT1363	CT1532	CT1791	CT1944	CT2226
CT0276	CT0515	CT0692	CT0894	CT1108	CT1370	CT1570	CT1793	CT1952	CT2227
CT0300	CT0516	CT0715	CT0901	CT1116	CT1372	CT1579	CT1796	CT1980	CT2231
CT0341	CT0519	CT0723	CT0914	CT1118	CT1389	CT1581	CT1803	CT1984	CT2277
CT0349	CT0520	CT0728	CT0921	CT1137	CT1390	CT1587	CT1820	CT1996	CT2282

CT0365	CT0526	CT0733	CT0925	CT1138	CT1404	CT1593	CT1836	CT2031
CT0370	CT0533	CT0739	CT0932	CT1140	CT1433	CT1594	CT1858	CT2043
CT0379	CT0539	CT0752	CT0955	CT1172	CT1449	CT1600	CT1865	CT2063

3.4. Influence of the artificial assignment on the predicting results

Artificial assignment natural numbers to different bases or amino acids is common phenomenon in graphical representations for biological sequence [20-22, 27]. In this work, the four kinds of nucleotides are listed in order of purine (A, G) → pyrimidine (C, T), 1, 2, 3 and 4 are respectively assigned to A, G, C and T for the purpose of visualization, with which one can intuitively discriminate different kind of trinucleotide in 2-D space [20]. Of course, we can also assign A, G, C and T with 4, 3, 2 and 1, etc. According to statistical theory, there are 4×3×2×1=24 kinds of encoding strategies in total. Then whether these artificial assignments can influence the calculating results has been debated at all times. However, in despite of which kind of encoding strategy, once the assignment is determined, the relation between the established spatial curve and the primary sequence is one to one, and then the numerical descriptors derived from these curves are unique to the primary DNA sequence, too. To filter the noise caused by the present assigning strategy, six variables are defined, i.e. x, y, z, x'_n, y'_n , z'_n. To clarify this issue further, we propose an modified graphical representation by substituting the four bases A, G, C and T with their physiochemical properties, the electron-ion interaction potential (EIIP), which is unique to the four nucleotides [39]. In some recently proposed graphical representations, especially for protein sequence [28], the unique physicochemical properties are widely used to numerically represent the amino acids in original sequences, which eliminate the argues of the artificial assignments mentioned above. In this way, one can also find some intrinsic properties by transforming the primary protein sequence into 2-D or 3-D curves. Nevertheless, there are much fewer physiochemical properties for nucleotides. EIIP describes the average energy states in valence electrons and has been used for encoding DNA sequence in some exon identification algorithms [40], where A → 0.1260, G → 0.0806, C → 0.1340 and T → 0.1335. Based on this modified graphical representation, we performed reannotation on *A. pernix* K1 genome. Consequently, all the 727 function known protein-coding genes as well as their

negative samples are all correctly predicted, all the putative genes in the second class are also correctly predicted, and 15 hypothetical genes in the third class are predicted as non-coding, which are presented in Table 6. Moreover, among the 15 ORFs in Table 8, there are 12 common items with column 2 of Table 2. Therefore, the accuracy and the number of ORFs predicted as non-coding are almost identical with that of I-TN curve. The high consistency also validate the numerical assignment cannot influence the predicting results. On the other hand, the comparison in this section can also provide useful theoretical support for later researches in graphical representations correlated problems.

Table 6: The recognized non-coding sequences based on the improved method encoding by EIIP. The bold items denote the common ORFs with column 2 of table 2.

APE_0416a	APE_0472c	**APE_0885b**	**APE_1275c**	**APE_2065.1**
APE_0470a	**APE_0762.1**	**APE_0954a**	**APE_1473a**	**APE_2284a**
APE_0471c	APE_0867b	**APE_1209d**	**APE_1882a**	**APE_2480a**

3.5. Why most of the ORFs listed in Table 5 do not appear to encode proteins

Codons usage and nucleotides distribution in protein-coding genes have been studied for many years. It was found that the severe restrictions on the base frequencies at the first two codon positions are universal in protein-coding genes and are independent of species. It is also suggested that purine bases at the first codon position are predominant and the frequency of G+C at the synonymous third position of sense codons (GC3s) is related to the expressivity of protein-coding genes. Highly expressed genes exhibit higher codon usage bias and prefer higher GC3s especially in GC rich genomes [41]. In previous works, the researchers mainly paid attention to the base distributions at different codon positions to discriminate protein-coding genes from non-coding. Here, we also attempt to associate the issue with codons usage bias. The GC content of *C. tepidum* TLS genome is 56%, while the average GC3s value of all the predicted protein-genes is 69.24%. For comparison, we calculate the purine/pyrimidine disparity (the difference between purine content and pyrimidine content at the first codon position) and GC3s/GC content disparity (the difference between GC3s and GC content) of each annotated potential protein-coding gene as shown in Fig. 1.

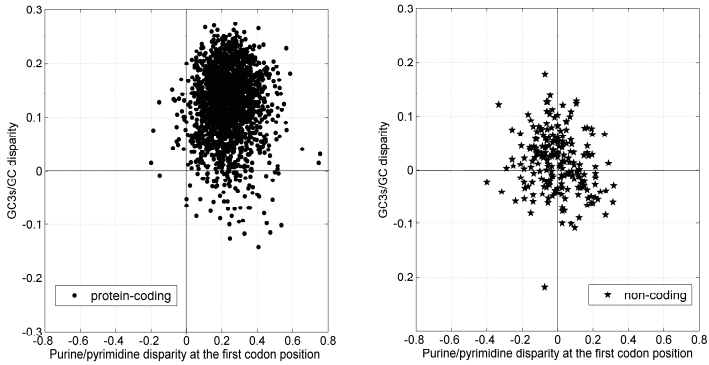

Figure 1: The scatter diagram of purine/pyrimidine disparity against GC3s/GC content disparity of *C. tepidum* TLS genome

In Fig. 1, each point corresponds to one ORF. As can be seen, most of the function-known genes are restricted to the first quadrant, which shows that these points have prominent content of purine bases at the first codon position and the higher GC3s values, whereas the distribution of these identified non-coding ORFs are randomly distributed around origin. Then, the base distributions at the first and the synonymous codon position of the function-known genes meet the previous observations, whereas that of the ORFs presented in Table 5 is not the case. On the other hand, observing the GC3s/GC content disparity, almost all the protein-coding gens prefer high GC3s, while most of the identified non-coding ORFs locate a region around the origin. The lower GC3s indicates the lower codons usage bias, which implies that they are much likely not true protein-coding genes but random sequences.

Codon adaptation index (CAI) is a useful parameter proposed to measure the gene expression level [42]. High CAI genes are presumed to be highly expressed while low CAI genes are presumed to be lowly expressed. In this study, CAI value of each gene is calculated to display the differences between the protein-coding genes and the predicted as non-coding ORFs. The value of relative synonymous codon usage (RSCU) is an index used to examine synonymous codons usage without the confounding influence of amino acid composition of different gene samples [43]. Correspondence analysis (COA) can be used to investigate the major trend in codon usage variation among genes. To display the differences between the

protein-coding genes and the predicted non-coding ORFs presented in Table 5, we perform COA on the RSCU values of all the potential protein-coding ORFs in *C. tepidum* TLS genome. COA plots RSCU values of all the ORFs in a multidimensional space of 59 axes (excluding Met, Trp and termination codons) and identifies a series of new orthogonal axes accounting for the greatest variation among genes. In this study, the axis 1 and axis 2 of COA account for 14.6% and 4.3% of the total variation among genes. The prominent weight of the first principle suggests a strong codon bias trend. The analyses were conducted using CodonW version 1.4.2. In Fig. 2, we present the scatter diagram of axis 1 generated by COA on RSCU of the ORFs against their corresponding CAI values.

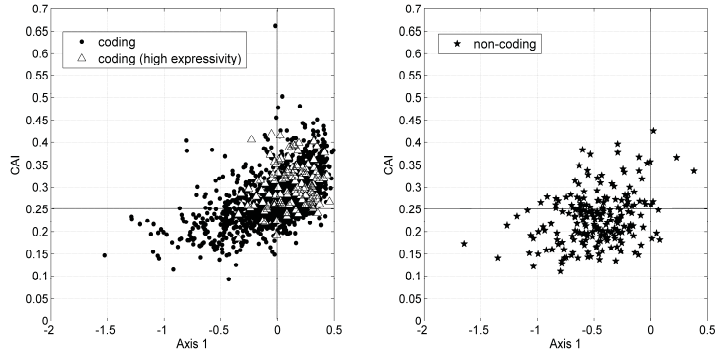

Figure 2: The scatter diagram of axis1 generated by correspondence analysis against their CAI values of *C. tepidum* TLS genome

Seen from Fig. 2, there are obvious differences between the regions in which the protein-coding genes and the predicted as non-coding ORFs distribute. The average value of axis 1 and CAI for protein-coding genes is 0.0427 and 0.2824, respectively, while -0.4750 and 0.2298 for non-coding. For convenience of observation, an axis is drawn in both plots at CAI=0.25, which is approximately near the midpoint between centers of protein-coding and non-coding (0.2824+0.2298)/2=0.2561. We observed that some highly expressed genes, such as ribosomal proteins, ATP synthase, etc., were clustered at the positive along the axis 1. On the other hand, axis 1 coordinates of protein-coding genes are significantly positively correlated with GC3s ($r = 0.9449$, $P<0.01$) and the CAI values ($r = 0.5369$, $P< 0.01$), respectively. Then, we infer that nucleotide compositions and gene expression level play

important roles in shaping codon usage in *C. tepidum* TLS, genes exhibit a greater degree of codon usage bias, and they always prefer for the codons with G/C at the synonymous position, which is consistent with the statistical results in Fig. 1. On the contrary, most of those ORFs predicted as non-coding locate the region where axis 1<0 and CAI<0.25 in Fig. 2, which should indicate their low gene expression level and low condon usage bias, which may be caused by random sequences. Therefore, the analysis in Fig. 2 is highly consistent with Fig. 1, based on which we infer that most of the 217 hypothetical ORFs presented in Table 5 are not true protein-coding genes.

4. CONCLUSIONS

In summary, Over-annotation of protein-coding genes has been a common phenomenon in microbial genomes. The increasing utility of public databanks makes it urgent to confirm the coding reliability of hypothetical ORFs. However, it is not practical to validate these sequences datum by 'wet' experiments because of the expensive cost and time consuming. Then computer-aided methods provide a key role in such issues. In this paper, we discriminate the falsely annotated protein-coding genes in *A.pernix* K1 by a 36-vector on the basis of graphical representation method. From the present work, three contributions can be concluded.

(1) The problem that how many protein-coding genes exist in *A. pernix* K1 genome has confused many scientists in the past ten years. Based on the present method, we reannotate the protein-coding genes in *A. pernix* K1 genome and a high accuracy is obtained. Consequently, 14 annotated hypothetical ORFs are predicted as non-coding. Then the number of protein-coding genes is reduced to 1686 instead of 1700 in the current annotation. Further analysis show our results are reliable. The identifying results by extension to *C. tepidum* TLS genome show the present approach can be applied to other microbial genomes.

(2) Although many approaches have been applied to the problem of annotation of protein-coding genes in microbial genomes and high accuracy can be achieved, the identified results seem to differ greatly in some cases. The causes perhaps lie in the different mechanisms adopted in those methods. In addition, the predicting results by several prevalent gene-finding programs validate the fact that false positive prediction caused by the high ratio of additional genes has been the bottleneck for gene annotations. The 36 numerical

descriptors deduced in our work correspond to the six possible reading frames of given DNA sequence, which are easily obtained just by calculating the geometrical centers of each component of x, y, z and x', y', z'. In previous works, extensive statistic works have to be done to describe information of protein-coding genes, therefore the workload and the running time can be shortened remarkably. On the other hand, abundant analysis show that the 36 numerical descriptors could display the universal features of protein-coding genes efficiently.

(3) Graphical representations have been proved to be convenient and efficient in biological sequences analysis. However, the open problem that how to propose graphical approaches that can provide more information is still difficult. As have been discussed previously [21], most graphical representation are based on individual nucleotides, it is ineluctable to assign some natural numbers to transform the biological sequences into intuitive curves, especially in those graphical representations attempting to describe polymers. In this work, we perform some meaningful analysis on the debates on the influences of the artificial parameters. Although these analysis are not sufficient and rigorous enough, the results obtained can provide usefully theoretical support for further construction and application of graphical representation based methods.

To facilitate the potential users, a convenient software named TN_curve NumG 1.0 is exploited to generate the 36-D vector of any DNA sequence, which can be retrieved by emailing us. TN_curve NumG 1.0 supports multi-sequence input in fasta format (confirming there is no blank space existing in file name), and there are two options for users:

(1). I-TN curve. This option generates the 36 numerical descriptors based on the encoding strategy of 1→A, 2→G, 3→C, 4→T.

(2). EIIP. This option generates the 36 numerical descriptors based on the encoding strategy of 0.1260→A, 0.0806→G, 0.1340→C, 0.1335→T.

Acknowledgements

The authors would like to thank Dr. F.B. Guo for providing excellent data sources and the anonymous referee that help us to improve the section of condons usage analysis, we are also grateful to any people for their valuable suggestions that have improved this manuscript. We thanks the support of National Natural Science Foundation of China (Projects No. 61073141 and No. 30970561), Shandong Natural Science Foundation (Project No. ZR2010CQ041) and the Scientific Research Foundation of Graduate School of Southeast University (Project No. YBJJ1010).

References

[1] A. Pallejà, E. D. Harrington, P. Bork, Large gene overlaps in prokaryotic genomes: result of functional constraints or mispredictions? *BMC Genomics* **9** (2008) 335.

[2] M. S. Poptsova, J. P. Gogarten, Using comparative genome analysis to identify problems in annotated microbial genomes, *Microbiology-SGM* **156** (2010) 1909–1917.

[3] C. W. Luo, G. Q. Hu, H. Q. Zhu, Genome reannotation of Escherichia coli CFT073 with new insights into virulence, *BMC Genomics* **10** (2009) 552.

[4] S. Bocs, A. Danchin, C. Medigue, Re-annotation of genome microbial CoDing-Sequences: finding new genes and inaccurately annotated genes, *BMC Bioinformatics* **3** (2002) 5.

[5] L. L. Chen, B. G. Ma, N. Gao, Reannotation of hypothetical ORFs in plant pathogen Erwinia carotovora subsp. atroseptica SCRI1043, *FEBS J.* **275** (2008) 198–206.

[6] F. B. Guo, J. Wang, C. T. Zhang, Gene recognition based on nucleotide distribution of ORFs in a hyper–thermophilic crenarchaeon, *Aeropyrum Pernix* K1, *DNA Res.* **11** (2004) 361–370.

[7] F. B. Guo, X. J. Yu, Re-prediction of protein coding genes in the genome of Amsacta moorei entomopoxvirus, *J. Virol. Methods* **146** (2007) 389–392.

[8] F. B. Guo, Y. Lin, Identify protein coding genes in the genomes of *Aeropyrum pernix* K1 and *Chlorobium tepidum* TLS, *J. Biomol. Struct. Dyn.* **26** (2009) 413–420.

[9] D. A. Natale, U. T. Shankavaram, M. Y. Galperin, Y. I. Wolf, L. Aravind, E. V. Koonin, Towards understanding the first genome sequence of a crenarchaeon by genome annotation using clusters of orthologous groups of proteins (COGs), *Genome Biol.* **1** (2000) 0009.1–0009.19.

[10] K. D. Pruitt, T. Tatusova, D. R. Maglott, NCBI reference sequence project: Update and current status, *Nucleic Acids Res.* **31** (2003) 34–37.

[11] M. Skovgaard, L. J. Jensen, S. Brunak, D. Ussery, A. Krogh, On the total number of genes and their length distribution in complete microbial genomes, *Trends Genet.* **17** (2001) 425–428.

[12] M. D. Silva, C. Upton, Using purine skews to predict genes in AT-rich poxviruses, *BMC Genomics* **6** (2005) 22.

[13] J. Wang, C. T. Zhang, Identification of protein coding genes in the genome of Vibrio cholerae with more than 98% accuracy using occurrence frequencies of single nucleotides, *Eur. J. Biochem.* **268** (2001) 4261–4268.

[14] S. Yamazaki, J. Yamazaki, K. Nishijima, R. Otsuka, M. Mise, H. Ishikawa, K. Sasaki, S. Tago, K. Isono, Proteome analysis of an aerobic hyperthermophilic crenarchaeon, *Aeropyrum pernix* K1, *Mol. Cell Proteomics* **5** (2006) 811–823.

[15] J. F. Yu, X. Sun, Reannotation of protein coding genes based on an improved graphical representation of DNA sequence, *J. Compu. Chem.* **31** (2010) 2126–2135.

[16] Y. Sako, N. Nomura, A. Uchida, Y. Ishida, H. Morii, Y. Koga, T. Hoaki, T. Maruyama, Aeropyrum pernix gen. nov., sp. nov., a novel aerobic hyperthermophilic archaeon growing at temperatures up to 100 degrees C, *Int. J. Syst. Bacteriol.* **46** (1996) 1070–1077.

[17] Y. Kawarabayasi, Y. Hino, H. Horikawa, S. Yamazaki, Y. Haikawa, K. Jin-no, M. Takahashi, M. Sekine, S. Baba, A. Ankai, H. Kosugi, A. Hosoyama, S. Fukui, Y. Nagai, K. Nishijima, H. Nakazawa, M. Takamiya, S. Masuda, T. Funahashi, T. Tanaka, Y. Kudoh, J. Yamazaki, N. Kushida, A. Oguchi, K. Aoki, K. Kubota, Y. Nakamura, N. Nomura, Y. Sako, H. Kikuchi, Complete genome sequence of an aerobic hyper-thermophilic crenarchaeon, Aeropyrum pernix K1, *DNA Res.* **6** (1999) 83–101.

[18] C. Cambillau, J. M. Claverie, Structural and genomic correlates of hyperthermostability, *J. Biol. Chem.* **275** (2000) 32383–32386.

[19] W. Chen, Y. Zhang, Comparisons of DNA sequences based on dinucleotide, *MATCH Commun. Math. Comput. Chem.* **61** (2009) 533–540.

[20] J. F. Yu, X. Sun, J.H. Wang, TN curve: A novel 3D graphical representation of DNA sequence based on trinucleotides and its applications, *J. Theor. Biol.* **261** (2009) 459–468.

[21] J. F. Yu, J. H. Wang, X. Sun, Analysis of similarities/dissimilarities of DNA sequences based on a novel graphical representation, *MATCH Commun. Math. Comput. Chem.* **63** (2010) 493–512.

[22] C. L. Yu, Q. Liang, C. C. Yin, R. L. He, S. S. Yau, A novel construction of genome space with biological geometry, *DNA Res.* **17** (2010) 155–168.

[23] W. Chen, Y. Zhang, Three distances for rapid similarity analysis of DNA sequences, *MATCH Commun. Math. Comput. Chem.* **61** (2009) 781–788.

[24] Y. Li, G. Huang, B. Liao, Z. Liu, H-L curve: A novel 2D graphical representation of protein sequences, *MATCH Commun. Math. Comput. Chem.* **61** (2009) 519–532.

[25] R. Wu, R. Li, B. Liao, G. Yue, A novel method for visualizing and analyzing DNA sequences, *MATCH Commun. Math. Comput. Chem.* **63** (2010) 679–690.

[26] B. Liao, W.Y. Chen, X. M. Sun, W. Zhu, A binary coding method of RNA secondary structure and its application, *J. Comput. Chem.* **30** (2009) 2205–2212.

[27] B. Liao, B. Y. Liao, X.M. Sun, Q. G. Zeng, A novel method for similarity analysis and protein sub-cellular localization prediction, *Bioinformatics* **26** (2010) 2678–2683.

[28] Y. H. Yao, Q. Dai, L. Li, X.Y. Nan, P. A. He, Y. Z. Zhang, Similarity/dissimilarity studies of protein sequences based on a new 2D graphical representation, *J. Comput. Chem.* **31** (2010) 1045–1052.

[29] M. L. Chiusano, F. Alvarez-Valin, M. G. Di, G. D'Onofrio, G. Ammirato, G. Colonna, G. Bernardi, Second codon positions of genes and the secondary structures of proteins. Relationships and implications for the origin of the genetic code, *Gene* **261** (2000) 63–69.

[30] S. K. Gupta, S. Majumdar, T.K. Bhattacharya, T. C. Ghosh, Studies on the relationships between the synonymous codon usage and protein secondary structural units, *Biochem. Bioph. Res. Co.* **269** (2000) 692–696.

[31] J. W. Fickett, C. S. Tung, Assessment of protein coding measures, *Nucleic Acids Res.* **20** (1992) 6441–6450.

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

[33] F. B. Guo, H. Y. Ou, C. T. Zhang, ZCURVE: a new system for recognizing protein coding genes in bacterial and archaeal genomes, *Nucleic Acids Res.* **31** (2003) 1780–1789.

[34] M. Burset, R. Guigo, Evaluation of gene structure prediction programs, *Genomics* **34** (1996) 353–357.

[35] K. D. Pruitt, T. Tatusova, D. R. Maglott, NCBI reference sequences (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins, *Nucleic Acids Res*. **35** (2007) 61–65.

[36] K. C. Chou, C. T. Zhang, Prediction of protein structural classes, *Crit. Rev. Biochem. Mol. Biol.* **30** (1995) 275–349.

[37] L. D. Arthur, A. B. Kirsten, C. P. Edwin, L. S. Steven, Identifying bacterial genes and endosymbiont DNA with Glimmer, *Bioinformatics* **23** (2007) 673–679.

[38] A. Lukashin, M. Borodovsky, GeneMark.hmm: new solutions for gene finding, *Nucleic Acids Res.* **26** (1998) 1107–1115.

[39] I. Cosic, Macromolecular bioactivity: Is it resonant interaction between macromolecules? – Theory and applications, *IEEE Trans. Biomed. Eng.* **41** (1994) 1101–1114.

[40] A. S. Nair, S. P. Sreenadhan, A coding measure scheme employing electron-ion interaction pseudopotential (EIIP), *Bioinformation* **1** (2006) 197–202.

[41] R. Banerjee, D. Roy, Codon usage and gene expression pattern of Stenotrophomonas maltophilia R551-3 for pathogenic mode of living, *Biochem. Bioph. Res. Co.* **390** (2009) 177–181.

[42] P. M. Sharp, W. H. Li, The codon adaptation index - a measure of directional synonymous codon usage bias, and its potential applications, *Nucleic Acids Res.* **15** (1987) 1281–1295.

[43] P. M. Sharp, W. H. Li, Codon usage in regulatory genes in Escherichia coli does not reflect selection for ‘rare’ codons, *Nucleic Acids Res.* **14** (1986) 7737–7749.