Chemical Physics Letters 530 (2012) 107–112

Contents lists available at SciVerse ScienceDirect

Chemical Physics Letters

journal homepage: www.elsevier.com/locate/cplett

A relative Lempel–Ziv complexity: Application to comparing biological sequences

Liwei Liu [a,*], Dongbo Li [b], Fenglan Bai [a]

[a] College of Science, Dalian Jiaotong University, Dalian 116028, PR China
[b] Department of Otolaryngology, Affiliated Xinhua Hospital of Dalian University, Dalian 116021, PR China

ARTICLE INFO

Article history:
Received 19 November 2011
In final form 24 January 2012
Available online 1 February 2012

ABSTRACT

One of the main tasks in biological sequence analysis is biological sequence comparison. Numerous efficient methods have been developed for sequence comparison. Traditional sequence comparison is based on sequence alignment. In this report, we propose a novel alignment-free method based on the relative Lempel–Ziv complexity to compare biological sequences. The vertebrate transferring genomes and the spike protein sequences are prepared and tested to evaluate the validity of the method. We use this method to build phylogenetic tree of two groups of the sequences. The result demonstrates that our method is powerful and efficient.

1. Introduction

Multiple sequence alignments are an essential tool for protein structure and function prediction, phylogeny inference and other common tasks in sequence analysis. Traditional sequence alignment method is much empirical to select a sequence alignment scoring matrix and gap penalty parameters, the difference of which may affect alignment results tremendously. In order to avoid this problem, during the last twenty years, several alignment-free methods for sequence comparison have arisen much interest in the field of computational biology. For example, graphical representations of biological sequences have been one kind of alignment-free methods to sequence analysis. Graphical method for visualizing DNA sequence is early proposed by Hamori and Ruskin [1]. Using graphic approaches to study biological systems can provide intuitive and useful insights, as indicated by many previous studies on a series of important biological topics [2,3]. Liao et al. reported on a sort of binary coding method of RNA secondary structures and coding rules based on the exclusive-OR operation [4]. In another paper, Liao et al. propose a 4-D representation of RNA secondary structures and outline an approach to make mathematical analysis and to compute the similarities between RNA secondary structures [5]. Liao et al. proposed a 6D representation of protein sequences consisting of 20 amino acids. Based on this 6D representation, they provided a proteome distance measure for constructing phylogenic tree [6]. In the report [7], Jia et al. proposed a novel 2D representation for protein secondary structure sequences. Word-based measure is one of widely used alignment-free approaches. In this method, each sequence is mapped into an *n*-dimensional vector according to its *k*-word frequencies, and similarity between the sequences is then defined by *n*-dimensional vector [8,9]. The LZ algorithm is another widely used alignment-free algorithm. Based on the relative information between the sequences using Lempel–Ziv complexity, a new sequence distance measure is proposed [10]. Zhang and Wang used conditional The LZ algorithm to compare the linear characteristic sequences of RNA secondary structures [11].

Our approach is motivated by the LZ algorithm. In order to depict the complexity relationship between two sequences, in the present report, we use relative the LZ complexity to analyze biological sequences. The proposed method is tested by phylogenetic analysis on two different data sets: 24 transferring sequences from vertebrates and 26 spike protein sequences from coronavirus. The results demonstrate that relative LZ complexity provides more information about phylogenetic and improves the efficiency of sequence comparison.

2. Methods

The LZ algorithm was developed to analyze the complexity of linear sequences by Lempel and Ziv [12]. Lempel–Ziv (LZ) complexity of a sequence is measured by the minimal number of steps required for its synthesis in a certain process. For each step only two operations are allowed in the process: either generating an additional symbol which ensures the uniqueness of each component or copying the longest fragment from the part of a synthesized sequence [13]. In recent years, some authors applied the algorithm to compare sequences and construct phylogenic trees. For instance, Otu and Sayood applied the LZ algorithm to phylogenic analysis and had successfully constructed phylogenic trees for real and simulated DNA data sets [10]. Liu and Wang take the physicochemical properties of amino acids into account, and used the LZ algorithm to construct phylogenic trees [14].

[*] Corresponding author. Fax: +86 411 84106380.
E-mail address: daliguowei@163.com (L. Liu).

0009-2614/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.cplett.2012.01.061

In this study, the measure of relative LZ complexity between two sequences is proposed according to the principle of the LZ complexity. The LZ complexity distance metric between two non-null sequences is defined by utilizing relative LZ complexity.

Next, we will give some basic definitions and properties about relative LZ complexity. Let the sequence $S = S_1S_2\cdots S_n$, $l(S) = n$ represents the length of S, the subsequence $S_iS_{i+1}\cdots S_j$ of S be denoted as $S(i, j)$. The set that contains all subsequence $S(i, j)$ is called the vocabulary $\upsilon(S)$ of S. Note that $S(i, j) = \varphi$, for $i > j$.

Let $S = S_1S_2\cdots S_n$ be a non-null sequence, then produce S from null sequence according the following algorithm [15].

(1) At the beginning we have a null-sequence φ, then add prefix $S = S_{1.}$ If $n > 1$, need add a dot after S_1.
(2) Let a prefix $Q = S_1S_2\cdots S_r$, $0 < r < n$ be available, check if $R = S_{r+1}$ can be reproduced from $S(1, r)$, or if R cannot reproduced from a subsequence of $S(1, r)$, then join Q and R to get a new prefix QR, and add a dot following QR. If $R = S_{r+1}$ can be reproduced from a subsequence of $S(1, r)$, then check again if $R = S_{r+1}S_{r+2}$ can reproduced from $S(1, r + 1)$. If so, check again if $R = S_{r+1}S_{r+2}S_{r+3}$ can reproduced from $S(1, r + 2)\cdots$ and so on. There two possible cases: In the case $R = S_{r+1}\cdots S_n$, then we end the procedure, and get new prefix $QR = S$, in another case $R = S_{r+1}\cdots S_k$ cannot be reproduced from any subsequence of $S(1, k - 1)$, then get a new prefix QR and add a dot behind it.
(3) Repeat the step (2) until produce S.

This algorithm is the LZ algorithm. Then, the sequence S can partition into some subsequences that arrange one after another. Denote this partition as follows:

$$H_{LZ}(S) = S(h_0+1,h_1)\cdot S(h_1+1,h_2)\cdot\cdots\cdot S(h_k+1,hk_1\cdot\cdots\cdot S(h_{m-1}+1,h_m).$$

Lempel and Ziv proved the exclusive partition about H_{LZ} and defined the LZ complexity $C_{LZ}(S)$ of S as the number of subsequence in $H_{LZ}(S)$, namely $C_{LZ}(S) = m$. For instance, $H_{LZ}(S)$ of the sequence $S = AAACACCACAC$ is $H_{LZ}(S) = A\cdot AAC\cdot ACC\cdot ACAC$, so $C_{LZ}(S) = 4$.

Given two sequences Q, R, according to the theory of Lempel and Ziv about sequence partition, we can also partition sequence Q into the subsequences one after another, called it relative

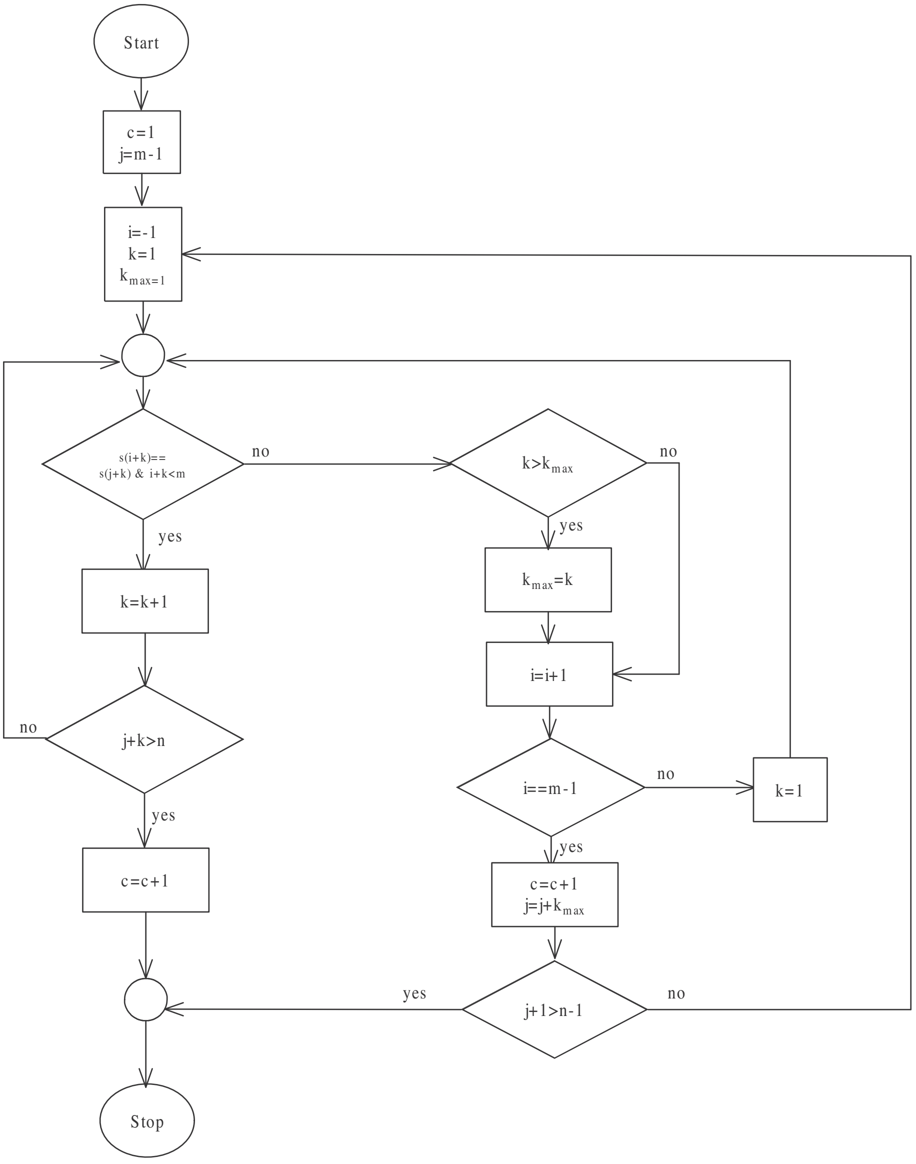

Figure 1. Flow diagram for the algorithm to calculate the r_{LZ}.

partition of sequence Q corresponding to sequence R. Denote it as follows:

$$H_{LZ}(Q|R) = Q(h_0+1,h_1)Q(h_1+1,h_2)\cdots Q(h_k+1,h_{k+1})\cdots Q(h_{m'-1}+1,h_{m'}).$$

$H_{LZ}(Q|R)$ satisfies the following three properties:

(1) $h_0 = 0$;
(2) $\forall 1 \leqslant k \leqslant (m'-2), Q(h_k+1,h_{k+1}-1) \in v(R), Q(h_k+1,h_{k+1}) \notin v(R)$;
(3) $h'_m = l(Q)$.

The complexity $r_{LZ}(Q|R)$ of Q corresponding to sequence R is the number of subsequence in $H_{LZ}(Q|R)$, namely $r_{LZ}(Q|R) = m'$. This relative partition is also exclusive. In order to compute the relative LZ complexity of Q, we only need to add the sequence R as the prefix of sequence Q. The flow diagram for the algorithm to calculate r_{LZ} is shown by Figure 1. The time complexity of this algorithm is $O(l(Q) \times l(R))$. The time complexity of Zhang's algorithm is $O(l(Q) \times (l(Q) + l(R))$ [11].

Clearly, $1 \leqslant r_{LZ}(Q|R) \leqslant l(Q)$. If Q which deletes its last character is subsequence of R, the result is $r_{LZ}(Q|R) = 1$. If all characters in Q do not belong to R, the result is $r_{LZ}(Q|R) = l(Q)$.Genome rearrangement is an important area of computational biology. The goal is to find the shortest sequence of genome arrangements operations that transform one genome into another. There are several basic operations: reversal, translocation and transposition. Traditional sequence alignment methods can only operate locally (e.g. insert, delete, replace) and thus ignore aspects of global sequence information (e.g. reversal, translocation, transposition).

For example, the sequence Q = *AAAAAAAAGGGGGGGG* is converted to R = *GGGGGGGGAAAAAAAA* by reversal. We use CLUSTAL 2.1 to compare these two sequences. The result is

```
– – – – – – – – A A A A A A A A G G G G G G G G
G G G G G G G G A A A A A A A A – – – – – – – –
```

This result suggests that the time to perform the operations (insert, delete) is 16. On the other hand, we calculate the $r_{LZ}(Q|R) = 2$. Clearly, the $r_{LZ}(Q|R)$ can to a great extent reflect the sequence similarity.

In order to eliminate the effect of the length of sequence Q on the distance measure, we normalize the distance measure as $r_{LZ}(Q|R)/l(Q)$. Therefore we define the distance as

$$d(Q,R) = \begin{cases} r_{LZ}(Q|R)/l(Q) + r_{LZ}(R|Q)/l(R), & Q \neq R \\ 0, & Q = R \end{cases} \tag{1}$$

3. Results

In this section, we illustrate the performance of our method on both DNA sequences and protein sequences. The validity of a phylogenetic tree can be tested by comparing it with authoritative ones. All the phylogenetic trees are drawn by using MEGA program.

3.1. Experiment 1: Phylogenetic trees of DNA sequences

In order to test the validity of our method, we select transferrin sequences from 24 vertebrates as a dataset [16]. Vertebrate transferrins (including lactoferrin and ovotransferrin) are iron-binding proteins found in blood serum, interstitial spaces, milk, tears, and egg whites [17]. It can be involved in iron storage and resistance to bacterial disease [16]. The 24 vertebrate transferrins genomes used in this report are downloaded from GenBank (data arepresented in Table 1).

Table 1
Transferrin sequences, sources, and accession numbers.

Sequence Name	Species	Accession No.
Human TF	*Homo sapien*	S95936
Rabbit TF	*Oryctolagus coniculus*	X58533
Rat TF	*Rattus norvegicus*	D38380
Cow TF	*Bos Taurus*	U02564
Buffalo LF	*Bubalus arnee*	AJ005203
Cow LF	*Bos Taurus*	X57084
Camel LF	*Camelus dromedaries*	AJ131674
Pig LF	*Sus scrofa*	M92089
Human LF	*H. sapiens*	NM_002343
Mouse LF	*Mus musculus*	NM_008522
Possum TF	*Trichosurus vulpecula*	AF092510
Frog TF	*Xenopus laevis*	X54530
Japanese flounder TF	*Paralichthys olivaceus*	D88801
Atlantic salmon TF	*Salmo salar*	L20313
Brown trout TF	*Salmo trutta*	D89091
Lake trout TF	*Salvelinus namaycush*	D89090
Brook trout TF	*Salvelinus fontinalis*	D89089
Japanese char TF	*Salvelinus pluvius*	D89088
Chinook salmon TF	*Oncorhynchus tshawytscha*	AH008271
Coho salmon TF	*Oncorhynchus hisutch*	D89084
Sockeye salmon TF	*Oncorhynchus nerka*	D89085
Rainbow trout TF	*Oncorhynchus mykiss*	D89083
Amago salmon TF	*Oncorhynchus masou*	D89086

We will consider the 24 transferrin sequences and calculate their distances Eq. (1). By arranging all these distances into a matrix, a pair-wise distance matrix is derived. This distance matrix contains the distance information on the 24 transferrin sequences. Lastly, this pair-wise distance matrix may be input to the Neighbor-joining program in PHYLIP package for a phylogenetic tree. The result is shown in Figure 2.

Figure 2 presents the phylogenetic trees reconstructed by our method. From Figure 2 we can observe that all the proteins that belong to transferring (TF) proteins and lactoferrin (LF) proteins have been separated well and grouped into respective taxonomic classes accurately. Human TF, Rabbit TF, Cow TF and Rat TF are clustered into the same branch. The Rat TF, Cow TF are separated from Human TF and Rabbit TF. The tree in Figure 2 is the most consistent with the trees constructed by Ford [16], which is the most classical result in the publicized existing trees.

Summing up, our method has significant advantage, and our results are almost agreement with that of previous studies.

3.2. Experiment 2: Phylogenetic trees of protein sequences

Phylogenetic analysis on genome sequences and protein sequences of coronaviruses has been studied by different methods, such as multiple sequence lignments, graphical representation, and word frequency. Here the phylogenetic tree for 26 spike protein sequences in Table 2 from coronavirus is instructed by our method, which is presented in Figure 3.

On March 12, 2003, WHO issued a global alert on severe acute respiratory syndrome (SARS). Since the outbreak of atypical pneumonia referred to as SARS, some researchers have considered the mutation analysis and phylogenetic analysis [18–20]. Moreover, mutation analysis and phylogenetic analysis will help to develop effective vaccines.

Based on the relative LZ complexity, we next consider to infer the phylogenetic relationships of coronaviruses with the spike protein sequences. The 26 spike protein sequences used in this report were downloaded from GenBank, of which 12 are from SARS-CoVs and 14 are from other groups of coronaviruses. The name, accession number and abbreviation for the 26 spike protein sequences

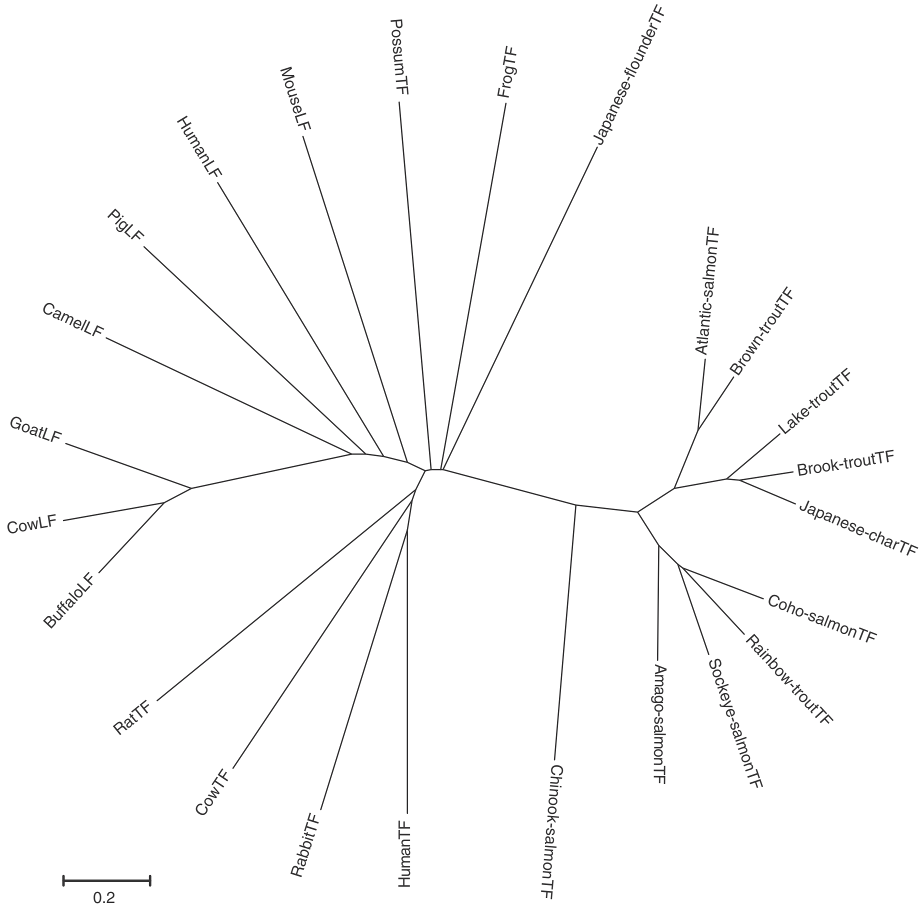

Figure 2. Phylogenetic tree obtained by our method using transferrin sequences.

Table 2
Coronavirus spike proteins sequences, sources, and accession numbers.

Sequence Name	Species	Accession No.
TGEV	Transmissible gastroenteritis virs	NP_058424.1
PEDV	Porcine epidemic diarrhea virus	NP_ 598310.1
HCoV-OC43	Human coronavirus OC43	NP_ 937950.1
BCoVM	Bovine coronavirus strain Mebus	AAA66399.1
BCoVL	Bovine coronavirus isolate BCoV-LUN	AAL57308.1
BCoVQ	Bovine coronavirus strain Quebec	AAL40400.1
BCOV	Bovine coronavirus	NP_150077.1
MHVM	Mouse hepatitis virus strain ML-10	AAF69344.1
MHVP	Mouse hepatitis virus strain Penn 97–1	AAF69334.1
MHVJHM	Murine hepatitis virus strain JHM	YP_209233.1
MHVA	Mouse hepatitis virus strain MHV-A59C12 mutant	AAB86819.1
IBVBJ	Avain infectious bronchitis virus isolate BJ	AAP92675.1
IBV Avain	Infectious bronchitis virus	NP_040831.1
GD03T0013	SARS coronavirus GD03T0013	AAS10463.1
PC4-127	SARS coronavirus PC4-127	AAU93318.1
PC4-137	SARS coronavirus PC4-137	AAV49720.1
Civet007	SARS coronavirus civet007	AAU04646.1
A022	SARS coronavirus A022	AAV91631.1
GD01	SARS coronavirus GD01	AAP51227.1
GZ02	SARS coronavirus GZ02	AAS00003.1
CUHK-W1	SARS coronavirus CUHK-W1	AAP13567.1
TOR2	SARS coronavirus TOR2	AAP41037.1
Urbani	SARS coronavirus Urbani	AAP13441.1
Frankfurt1	SARS coronavirus Frankfurt 1	AAP33697.1
Sino1-11	SARS coronavirus Sin01-11	AAR23250.1

are listed in Table 2. Given a set of protein sequences, their phylogenetic tree can be obtained through the following main operations: first, we calculate the relative LZ complexity for protein sequences; second, by arranging all these distances into a matrix, we obtain a pair-wise distance matrix. Finally, we put the pair-wise distance matrix into the neighbor-joining program in the

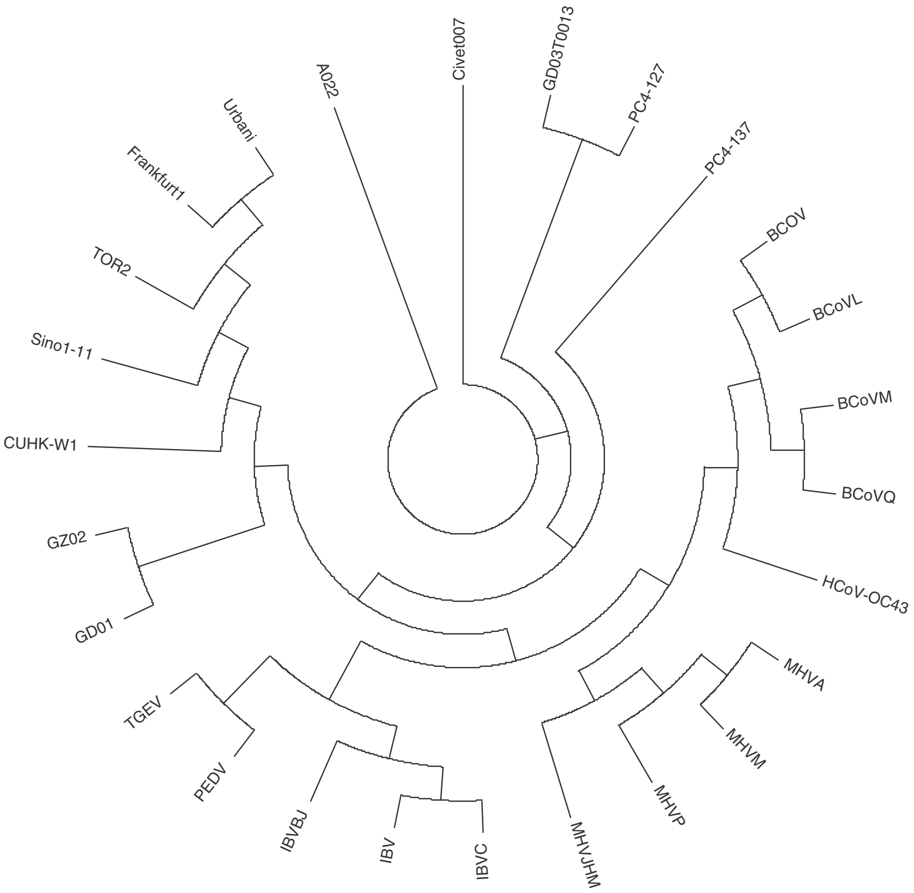

Figure 3. Phylogenetic tree obtained by our method using spike protein sequences.

PHYLIP package. We obtain the phylogenetic relationships drawn by MEGA program [21]. In Figure 3, we present the unrooted phylogenetic tree belonging to 26 species.

As can be seen from Figure 3, SARS-CoVs appear to cluster together and form a separate branch, which can be easily distinguished from other coronaviruses. The topology of the tree obtained by our method is quite consistent with the results obtained by other authors [22,23].

4. Discussion

Sequence comparison is rapidly becoming an essential tool for bioinformatics applications. It has been used to support other types of analysis, from searching a database with a query DNA sequence to the phylogenetic tree construction. Despite the prevalence of alignment-based methods, it is noteworthy that alignment-based method is computationally intensive and consequently unpractical for querying large data sets, which forces the use of some heuristics to reduce the running times, as exemplified by BLAST. Alignment-free comparison method is therefore of great value as it reduces the technical constraints of alignments.

A novel alignment-free method for sequence comparison is proposed in this work. The relative LZ complexity has been introduced into biological sequence comparison. The time complexity of this algorithm is $O(l(Q) \times l(R))$. The main advantage is that this method can extract repeated patterns from biological sequence. Therefore, when two sequences are compared, the subsequences that they share can be detected. In this report, we use the relative LZ complexity to describe the similarity of the biological sequences. The two experiments have shown that the approach proposed in this report is a powerful and useful tool for the comparison of biological sequence. Studies of the application of this method to whole coding DNA sequences, RNA sequence and protein sequence will appear in the future. Furthermore, this method can be used to prediction of protein secondary structure. The shortage of this method is that some information may be lost when protein primary structures are converted to protein feature sequences. However, the tests have proven that our method can extract phylogenetic information from proteins and hence it can complement phylogenetic analysis.

Acknowledgments

This work was partially supported by the Science Research Project of Educational Department of Liaoning Province of China (L2010074) and the National Natural Science Foundation of China (10871219).

References

[1] E. Hamori, J. Ruskin, J. Biol. Chem. 258 (1983) 1318.
[2] M. Randic, Chem. Phys. Lett. 386 (2004) 468.
[3] G. Huang, B. Liao, Y. Li, Z. Liu, Chem. Phys. Lett. 462 (2008) 129.
[4] B. Liao, W. Chen, X. Sun, W. Zhu, J. Comput. Chem. 30 (2009) 2205.
[5] B. Liao, W. Zhu, P. Li, J. Math. Chem. 42 (2007) 1015.
[6] B. Liao, J. Luo, R. Li, W. Zhu, Int. J. Quantum Chem. 107 (2007) 1295.
[7] C. Jia, T. Liu, X. Zhang, S. Yan, Int. J. Quantum Chem. 109 (2009) 819.
[8] Q. Dai, X. Liu, Y. Yao, F. Zhu, J. Theor. Biol. 276 (2011) 174.
[9] S. Karlin, M. Ladunga, Proc. Natl. Acad. Sci. U S A 91 (1994) 12832.
[10] H.H. Otu, K. Sayood, Bioinformatics 19 (2003) 2122.
[11] S. Zhang, T. Wang, J. Biomol. Struct. Dyn. 28 (2) (2010) 247.
[12] A. Lempel, J. Ziv, IEEE Trans. Inform. Theory 22 (1976) 75.
[13] V.D. Gusev, L.A. Nemytikova, N.A. Chuzhanova, Bioinformatics 15 (1999) 994.
[14] N. Liu, T. Wang, FEBS Lett. 580 (2006) 5321.
[15] L. Liu, T. Wang, J. Theor. Biol. 251 (2008) 159.

[16] M. Ford, Mol. Biol. Evol. 18 (2001) 639.
[17] T.M. Loehr, Iron Carriers and Iron Proteins, VCH, NewYork, 1989.
[18] A. Grigoriev, Trends Genet. 20 (3) (2004) 131.
[19] W. Gu, T. Zhou, J. Ma, X. Sun, Z. Lu, Virus Res. 101 (2004) 155.
[20] Q. Dai, X. Liu, L. Li, Y. Yao, B. Han, L. Zhu, J. Comput. Chem. 31 (2010) 351.
[21] S. Kumar, K. Tamura, M. Nei, Briefings Bioinf 5 (2004) 150.
[22] W. Zheng, L. Chen, H. Ou, F. Gao, C. Zhang, Mol. Phylogen. Evol. 36 (2005) 224.
[23] H. Song, C. Tu, G. Zhang, et al., Proc. Natl. Acad. Sci. U S A 102 (2005) 2430.