Information Theoretic Approaches to Whole Genome Phylogenies *†

Igor Ulitsky David Burstein Tamir Tuller ‡ Benny Chor§

School of Computer Science
Tel Aviv University

May 2nd, 2005

Abstract

We describe a novel method for efficient reconstruction of phylogenetic trees, based on sequences of whole genomes or proteomes, whose lengths may greatly vary. The core of our method is a new measure of pairwise distances between sequences. This measure is based on computing the average lengths of maximum common substrings. It is intrinsically related to information theoretic tools (Kullback-Leibler relative entropy). We present an algorithm for efficiently computing these distances. In principle, the distance of two ℓ long sequences can be calculated in $O(\ell)$ time. We implemented the algorithm, using suffix arrays. The implementation Is fast enough to enable the construction of the proteome phylogenomic tree for hundreds of species, and the genome phylogenomic forest for almost two thousand viruses. An initial analysis of the results exhibits a remarkable agreement with "acceptable phylogenetic and taxonomic truth". To assess our approach, it was compared to the traditional (single gene or protein based) maximum likelihood method. It was compared to implementations of a number of alternative approaches, including two that were previously published in the literature, and to the published results of a third approach. Comparing their outcome and running time to ours, using a "traditional" trees and a standard tree comparison method, our algorithm improved upon the "competition" by a substantial margin. The simplicity and speed of our method allows for a whole genome analysis with the greatest scope attempted so far. We describe here five different applications of the method, which not only show the validity of the method, but also suggest a number of novel phylogenetic insights.

Key words: Phylogenomics, whole genome and proteome phylogenetic, tree reconstruction, compressibility, distance matrix.

1 Introduction

The elucidation of the evolutionary history of extinct and extant species is a major scientific quest, dating back to Darwin [14] and before. Early approaches were based on morphological and palaeontological data, but with the advent of molecular biology, the emphasis has shifted to molecular (amino acid and nucleotide) sequence data. Rapid sequencing technologies have produced the genome sequences of over 200 cellular organisms, and many more projects are underway [21]. Full genomes contain huge amounts of sequence data that should undoubtedly be

*An extended abstract of this paper will appear in *Proc. of the tenth Annual International Conference on Computational Molecular Biology (RECOMB)*, May 2005, Cambridge, MA.

†Research supported by ISF grant 418/00

‡corresponding author

§emails: ulitskyi,davidbur,tamirtul,bchor@post.tau.ac.il

useful in constructing phylogenetic trees. However, this insight is not yet reflected in the practice of phylogenetic trees reconstruction. The vast majority of published works are based on a single gene or protein. Most others are based on combining a few gene trees to a species tree [30], on quartet methods [38, 2], or on super tree methods [3].

There are many compelling reasons to consider whole genomes or proteomes as a basis for phylogenetic reconstruction, and in some contexts, such methods are essential. One example are the viruses, where different families often have very few genes in common, and it is undesirable to base the whole phylogenetic reconstruction on one or very few genes. Traditional methods (such as maximum parsimony or maximum likelihood) are thus inapplicable for viruses, and whole genome methods are naturally called for.

An alternative method that encorporates global, genome wide information, is building trees based on gene order. Here, the goal is to find the shortest sequence of rearrangements between two genomes, and to use it for tree reconstruction. This approach has been used for quite some time, initially employing heuristics [17]. In 1995, Hannanelly and Pevzner [25] made a breakthrough, finding a polynomial time algorithm for the problem of computing the inversion distance. Following numerous improvements and refinements, it is now possible to compute the inversion distance "metric" for a large number of species. Usually, the major emphasis is on finding the shortest inversion sequence rather than on deducing a tree. This may explain why practically this method was used for constructing trees with less than two dozen taxa. A number of trees based on inversion distances have been constructed, *e.g.* [33, 5, 17], but their biological significance is still under investigation. Furthermore, before the inversion distance methods can be applied, an identification of the "genetic units" (usually genes) under study is required in each genome. A mapping of each unit to its counterpart in the other genome should follow. So far, these stages are done manually, and not automatically. By way of contrast, no such manual preprocessing is needed in our approach, which is based on the sequence itself.

In this work we apply string algorithms, rooted in information theoretic tools, to construct phylogenies that are based on complete genomes or proteomes. These methods are essentially distance methods: The first step is to compute all the pairwise "distances" between species. Our "distance" is intuitively appealing, and we also show it is closely related to information theoretic tools. We exhibit info-theoretical tools and results that justify this measure in case the strings were generated by *unknown* Markov processes. Specifically, our distance is related to the relative compressibility [13] of two Markov induced distributions. Given two strings, our method computes a quantity, which is close to the relative compressibility, without knowing the parameters of the Markov processes. Since DNA and proteins sequences can reasonably be modelled as a Markovian random process [19], the use of this measure is natural and may explain the success of our approach. Furthermore, it should be emphasized that the algorithm employs string operations that can be applied to any set of sequences, regardless of its origin. This situation is similar to the Lempel–Ziv compression algorithm [27, 28], whose properties were proved under the assumption of an underlying finite state Markovian source, but is then applicable to sequences of any source.

Our algorithm takes $O(\ell)$ steps to compute the distance between two ℓ long genomes. This runtime is fast enough to compute the $\binom{n}{2}$ pairwise distances between n species for a moderate to large n. We then apply a distance-based phylogenetic reconstruction method such as neighbor joining, NJ [42], to build a tree from the $n \times n$ distance matrix. The efficiency of our algorithm enables us to generate trees for all $n = 191$ cellular organisms whose complete proteomes were published in the NCBI database [21]. These include Archea, Bacteria, and Eukaryotes. A forest of phylogentic trees for $n = 1,865$ viruses has also been constructed.

Prior to our work, only about six major works for constructing trees from complete genomes or proteomes sequences were published. Stuart *et al.* [45] used singular value decomposition (SVD) of large sparse data matrices. Each proteome is represented as a vector of tetrapeptide

frequencies. The distance between two species is determined by the cosines of the angle between the corresponding vectors. A similar idea was used by Qi *et al.* [37]. In their method the frequencies of amino acid K-mers in the complete proteomes of two species determines the distance between them. In our view, the main drawback of these two methods is their rigidity: The analysis is based on *fixed length* K-mers (usually $3 \leq K \leq 8$). Otu *et al.* [36] used Lempel-Ziv complexity as a basis of a strings' distance. This method is closer in nature to the one we use.

In a series of two papers, Chen *et al.* [9] and Li *et al.* [29] develop tools that are inspired by Kolmogorov complexity to compress biosequences, and then to compute pairwise distance based on the compression outcome. Since Kolmogorov complexity is incomputable, what their GenCompress algorithm actually uses is a generalizion of the Lempel-Ziv algorithm [27, 28]. This compression algorithm reportedly outperforms other DNA compression methods. It has been applied to construct a whole mitochondrial genome phylogeny. It was also applied to sequences of non biological source (chain letters, and music).

We compared our approach to those of Otu *et al.* and those of Qi *et al.* [36, 37], by implementing and applying them to the a dataset of proteomes and genomes of 75 organisms whose whole genomes and proteomes were published [21]. The performance of the various methods has been compared using a standard measure of phylogenetic trees comparison (the Robinson-Foulds tree distance) with respect to a "reference" maximum likelihood tree, based on the small ribosomal subunit rRNA [11]. Our algorithm outperformed all other ones. Compared to the best alternative method, its improvement was 2% on genome sequences, and as much as 17% on proteome sequences. In the next step, we checked our method on the small set of mitochondrial DNA of 34 mammalian species that were used by Li *et al.* [29]. We compared our results to the maximum likelihood (ML) trees for 13 genes for which a multiple sequence alignment of all the taxa in the set is available, and also to the published results Li *et al.* [29] for the set.

We then ran our algorithm to produce a tree of all 191 available proteome sequences, and then to a forest of 1,865 viral genomes. The results in general were very good, exhibiting high agreement with the accepted taxonomies. We examined some portions of this large forest (*e.g.* the retroviral and the ssRNA negative-strand trees), and observed that in a vast majority of the cases, the species classification imposed by the reconstructed phylogeny is in good agreement with the current taxonomic knowledge. In the few cases where the exact placement disagrees with the accepted taxonomy, there is support in the literature to this alternative placement.

The remaining of this work is organized as follows: In section 2 describe the algorithm and its properties, and then give a brief mathematical intuition for our method. In section 3 we describe the results of running our algorithm on real data sets: in subsection 3.1 we compare it to other methods on 75 species, in section 3.2 we check our method on set of mitochondrial DNA of 34 species, and compare it to ML trees and the tree of Li *at el.* [29]. In subsection 3.3 we present the results of our algorithm on all 191 known proteomes, and in subsection 3.4 the results on a large set of $1,865$ viruses. Finally, Section 4 contains concluding remarks and directions for further research.

2 Materials and Methods

In this section we describe our main method, the average common substring (ACS) algorithm, and its information theoretical basis. Let A and B be two strings (genomes or proteomes) of lengths n and m respectively. For any position i in A, we identify the length $\ell(i)$ of longest substring $A(i), A(i+1), .., A(i-1+\ell(i))$ that *exactly matches* a substring $B(j+1), B(j+1), .., B(j+\ell(i))$ in B starting at some position, j. We average all these lengths $\ell(i)$ to get a measure $L(A,B) = \sum_{i=1}^{n} \ell(i)/n$. Intuitively, the larger this $L(A,B)$ is, the more similar the two genomes are. For a given A, to account for B's length (longer B will tend to have larger

ACS), we "normalize" $L(A,B) = \sum_{i=1}^{n} \ell(i)/n$. Intuitively, the larger this it by $\log(m)$ to get $L(A,B)/\log(m)$. Now this is a *similarity* measure, while we are after *distance*: So we take the inverse, then subtract a "correction term" that guarantees $d(A,A)$ will always be zero, yielding $d(A,B) = \log(m)/L(A,B) - \log(n)/L(A,A)$. Notice that $L(A,A) = n/2$, so the correction term equals $2\log(n)/n$ and converges rapidly to 0 with $n \to \infty$. This last measure $d(A,B)$ is not symmetric, so we compute $d_s(A,B) = d_s(B,A) = (d(A,B) + d(B,A))/2$, which is our final ACS measure between the two strings. In the rest of this section we will give a theoretical background to our method, and more details about our algorithm and its implementation.

In information theory, it is well known that if a string has been generated by a finite state Markov process, then asymptotically, the minimum number of bits needed to describe the string, equal the source entropy times the string's length [13]. If a string has been generated by a Markovian process, there are compression algorithms, like Lempel-Ziv [43], which asymptotically achieve the optimal compression ratio. A natural way to measure the distance between two strings is the amount of bits needed in order to describe one sequence, given the other. Using a dictionary that was generated for optimally compressing one string, to optimally compress the other string, asymptotically achieves this ratio. For two independent identically distributions (i.i.d) probability distributions, $p(x)$ and $q(x)$, this measure is defined as:

$$\tilde{D}(p||q) = -\sum_{x \in X} p(x) \log q(x) = -E_p(\log q(X)) \tag{1}$$

For a pair of Markovian probability distributions a natural extension of the definition is as following:

$$\tilde{D}(p||q) = \lim_{n \to \infty} -\frac{1}{n} \sum_{x^n \in X^n} p(x^n) \log q(x^n) = -E_p(\log q(X)) \tag{2}$$

In general $\tilde{D}$ is *not* a metric. For example $\tilde{D}(p||q) \neq \tilde{D}(q||p)$, and the triangle inequality may not hold.

By [50, 22, 49], if A, B are a pair of strings generated by a pair of Markovian distributions p and q, then the quantity $d(A,B)$ computed by our ACS algorithm converges to $\tilde{D}(q||p)$ (as the length of the strings goes to infinity). Since $\tilde{D}(p||q) = -E_p(\log q(X))$ is a natural distance measure between Markovian distributions, this gives a theoretical basis for using ACS.

We remark that in addition to the ACS estimate of $\tilde{D}(p||q) = E_p(\log q(X))$, we have also used estimators for the divergence, or KL relative entropy [13], $D(p||q) = -E_p(\log q(X)) - H(p)$, and used it as a distance measure. Empirically, this proved inferior to using $\tilde{D}(p||q)$.

Given a set of DNA or amino acid sequences, our algorithm computes the pairwise distances for this set according to our ACS based metric, $d_s(A,B)$. We can efficiently perform the subsequence search by using suffix trees [48]. Creating the generalized suffix tree for two sequences of lengths ℓ_1, ℓ_2 requires $O(\ell_1 + \ell_2)$ time. It is created once for each pair of sequences and allows to compute the ACS distance of these two sequences A suffix tree in $O(\ell_1 + \ell_2)$ time. After this is done, this tree is discarded. All in all, comparing m sequences of length up to ℓ to each other, takes $O(m^2 \cdot \ell)$ time. In practice, we have chosen suffix arrays as the data structure in our implementation. The suffix array is a lexicographically sorted array of the suffixes of a string. We used the "lightweight suffix array" implementation [44]. In this case too, creating the suffix array for each sequence requires $O(\ell)$ time for a sequence of length ℓ. However, as the suffix array provides a sorted array of the suffixes of the sequence, it allows to search for each subsequence in $O(\log(\ell))$ time. Thus pairwise comparing all m sequences of length up to ℓ, takes $O(m^2 \cdot \ell \cdot \log(\ell))$ time. The main advantage of using suffix arrays is the smaller constant in the space requirements.

In terms of space complexity, each suffix array requires $O(\ell)$ space, and an additional $O(\ell/\sqrt{\log \ell})$ is required in the construction stage and then reclaimed. The suffix arrays are stored in the secondary memory (disk) and loaded to primary memory only when needed for a pairwise comparison session. Since entries of the distance matrix can be calculated independently, the process can easily be parallelized, yielding a substantial acceleration of the running time.

3 Results and Discussion

We also developed and implemented a different, new method. In this method we use the relative compressibility between two probability mass functions of K-mers (fixed K) in the two genomes. It is faster and simpler than our ACS method, but proved inferior in practice. We compared our ACS method to methods that were suggested in other works. The first method is a LZ-based [13, 34, 27, 28], where the distance between two strings is inferred by the compressibility of one string given the other string's dictionary, using the LZ algorithm. This method is due to Otu *et al.* [36]. The second method is by Qi *et al.* [37]. In this method, first the vector of K-mer frequencies in each genome is calculated, and then the scalar products of the vectors are used to generate a distance measures. We implemented these two methods, and used these implementations in our comparisons.

The methods have been applied to sets of genomic and proteomic sequences. For species with multiple chromosomes the genomic sequence is a concatenation of all the chromosomes with delimiters, recognized as end points, by the algorithms. The proteomic sequences are a concatenation of all the known amino-acid sequences for an organism, also with delimiters. All the sequences have been obtained from the NCBI Genome database [21] in FASTA format (.fna and .faa files).

In this section we describe the results of running our algorithm on three sets of sequences. The first dataset has 75 species, and we used both genome and the proteome sequences. We used this dataset to checked all the other suggested reconstruction methods including two of the leading methods from the literature. Our algorithm outperformed all the alternative ones. The second dataset include the complete mtDNA sequences of 34 mammals, for this dataset we compared our performances to the performances of maximum likelihood, and to the published results of another leading method. We also ran it on two larger datasets: The first contains all known proteomes (191 species), the second includes the genomes of $1,865$ viruses.

3.1 Comparison to Other Methods

We compared the trees that was constructed by our and other methods to a "traditional tree", generated from the sequences of the small ribosomal database project (RDP) [11]. The "traditional tree", obtained from the data in RDP release 8.1 , is the maximum likelihood tree for the aligned set of small ribosomal subunit rRNA.

We obtained a dataset of 75 full genome and proteome sequences. This dataset contains archea, bacteria, and eukarya for which both genomic and proteomic sequences are available, and that also appear in the Ribosomal Database Project. The different "competing methods" were applied to these sequences, generating different distance matrices. Phylogenetic trees have been constructed from the distance matrices using the Neighbor Joining algorithm (NJ) [42] (as implemented by the *NEIGHBOR* program in the PHYLIP package [24]) . We used the Robinson-Foulds measure to compare the topology (we assume all branch lenths equal 1) of each tree to the reference, ribosomal tree. Each edge in a tree partition the leaves, or species, to two disjoint sets. The Robinson-Fouldes (RF) method counts the number of partitions that are *not* common to both trees. For two trees on n leaves, the RF "distance" is in the range $[0, 2n-6]$ and is always

even [40].

We implemented the Qi-Wang-Hao method [37] and the method of Otu *et al.* [36] (an LZ - based method). We also implemented a K-mer based divergence method (FLS method), an improved LZ - based method (a bit different than the method of Otu *et al.*) where we try to compress one genome by the dictionary of the other, and a method which produces a matrix with random distances (entries are independently, uniformly and identically distributed), for comparison purposes. Furthermore, a method of comparison based solely on the relative length of sequences has been implemented, in order to test possible correlations between certain methods and the sequence length. The results of all the methods are summarized in Table 1. We chose K-mers of size 5 for the FLS method when the input were proteomes, and 11 when the inputs were genomes, as these parameters gave the best results for this method.

Table 1: Comparing the ACS to other methods.

	RF distance from the reference tree	
Method	*Genomes*	*Proteomes*
Random	140	144
Length	142	142
LZ (ours)	126	114
LZ (Otu-Sayood)	118	126
FLS	120	96
Qi-Wang-Hao	110	92
Our method (ACS)	108	76

It should be observed that our ACS method improve upon all other methods for both genomes and proteomes. But while the genome improvement compared to [37] is only about 2%, the improvement for the proteomes is about 17% (all in the RF measure with respect to the reference tree).

3.2 A Tree Based on Mitochondrial DNA

In this subsection we demonstrate the performance of our method on a set of mitochondrial genome proteomes. Our input contains the mitochondrial genomes and proteomes of 34 mammals. Since the multiple alignments for 13 mitochondrial proteins of these species are available in NCBI genomes, it allows us to compare our tree to trees that are constructed by the maximum likelihood method (PHYLIP's PROTML [1] was used for the ML reconstructions). We also compared our results to the published results of Li *at el.* [29], which used the same dataset as an input. The resulting tree for the genomic input sequences is described in figure 1.

The quality of the obtained tree can be seen by viewing the splits: The correct clustering of the primates (Pygmy Chimpanzee, Human, Gorilla, Orangutan, Ggibbon, Baboon), marsupials and monotremes (Platypus, Wallaroo, and Opossum), rodents (House Mouse, Rat, Guinea Pig) and Laurasiatheria (Dog, cat, Greay Seal, Harbor Seal, Hippo, White Rhino, Great Rhino, Donkey, Horse, Pig, Sheep, Blue Whale, Finback Whale). The overall structure of the tree agrees with the trees in Reyes *at al.* [39] and in Li *at el.* [29]. Here, however the pig is clustered closer to the cetartiodactyls than the perisodactyls (as in [39], and opposed to [29]), the guinea pig is close to the muridae and the laporidae (as in [39], and opposed to [29]).

In order to quantitavely compare the trees produced using whole-genome methods to the trees obtained using maximum likelihood with single gene sequence inputs, we have computed the RF distances between the trees generated using the different methods, which is presented in table 2. We used PHYLIB for the RF calculations.

A summary of the distances between the evolutionary trees built from the mammalian mtDNA data by different methods is described in table 2. The distances in the table are normalized RF distances. The first 13 rows (and columns) correspond to the maximum likelihood trees for each of the 13 proteins (ATP6, ATP8, COX1, COX2, COX3, CYTB, ND1, ND2, ND3, ND4, ND4L, ND5, and ND6). The row and column $ACSg$ correspond to the tree reconstructed by the ACS method when the inputs were whole mt genomes. The row (and column) LI stand for the tree of Li *at el.* [29]. The row (and column) $Cons$ stand for the majority consensus tree of all the ML trees. The last raw, *Average*, is the average of the first 13 rows (single gene ML trees). Here (as opposed to the previous subsection) the results for the whole genomes are better than those for the proteomes. A possible explanation is that in mtDNA there are less non-coding regions then in the non-mitochondrial genomes. The average RF distance of the ASCg tree from all the ML trees is lower than the average distance between any ML tree and the other ML trees. Our tree has a slightly larger average distance from all the ML trees than the tree of Li *et al*, and a lower distance from the consensus tree. The results here suggest that our method usually describes the phylogenetic truth of the complete mtDNA better than ML for any single gene. Furthermore our method is much faster than ML method, which is known to be NP-hard [10]. The accuracy of our method on mtDNA (as judged by RF distance) is comparable to the method of Li *et al.*, while it is simpler and much faster.

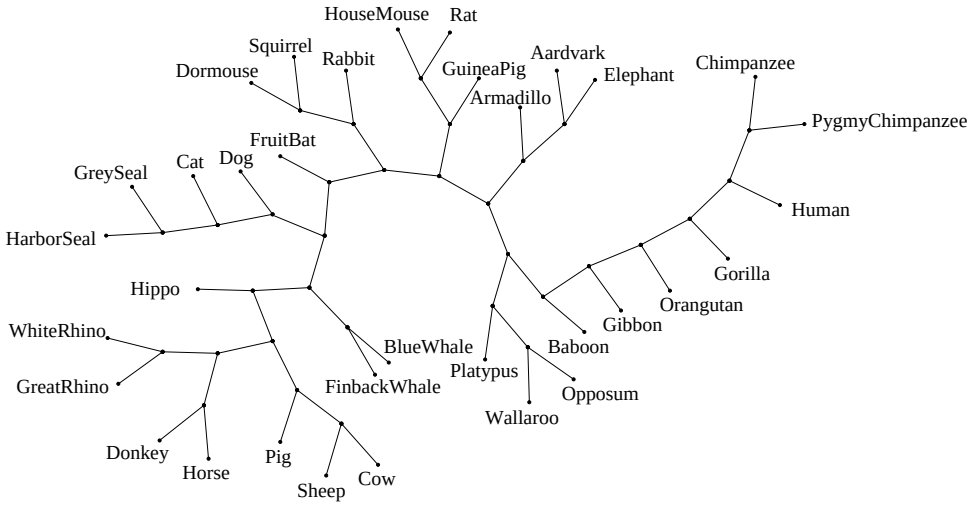

Figure 1: The evolutional tree built from complete mammalian mtDNA of 34 taxa by the ACS method.

3.3 A Tree Based on All Existing Proteomes

We collected all 191 available proteome sequences from the NCBI databank [21] as of October 2004. This dataset includes 19 proteomes of archea, 161 proteomes of bacteria, and 11 proteomes of eukaroytes. Our tree is presented in Fig. 2, for the sake of clarity all the branches have the same length.

The ACS method has correctly partitioned the species into the 3 main domains: *Eukaryota*, *Archaea* and *Prokayota*, with the exception of 2 archaeal species which will be discussed further. Within the 11 eukaroyte species in the dataset, the *Fungi*, *Eumetazoa*, plants and rodents are all correctly separated by the algorithm. This isn't surprising, given the major differences between the representative eukaryote genomes that have been completely sequenced to date. Thus, more challenging and interesting is the correspondence between the results of the ACS algorithm and

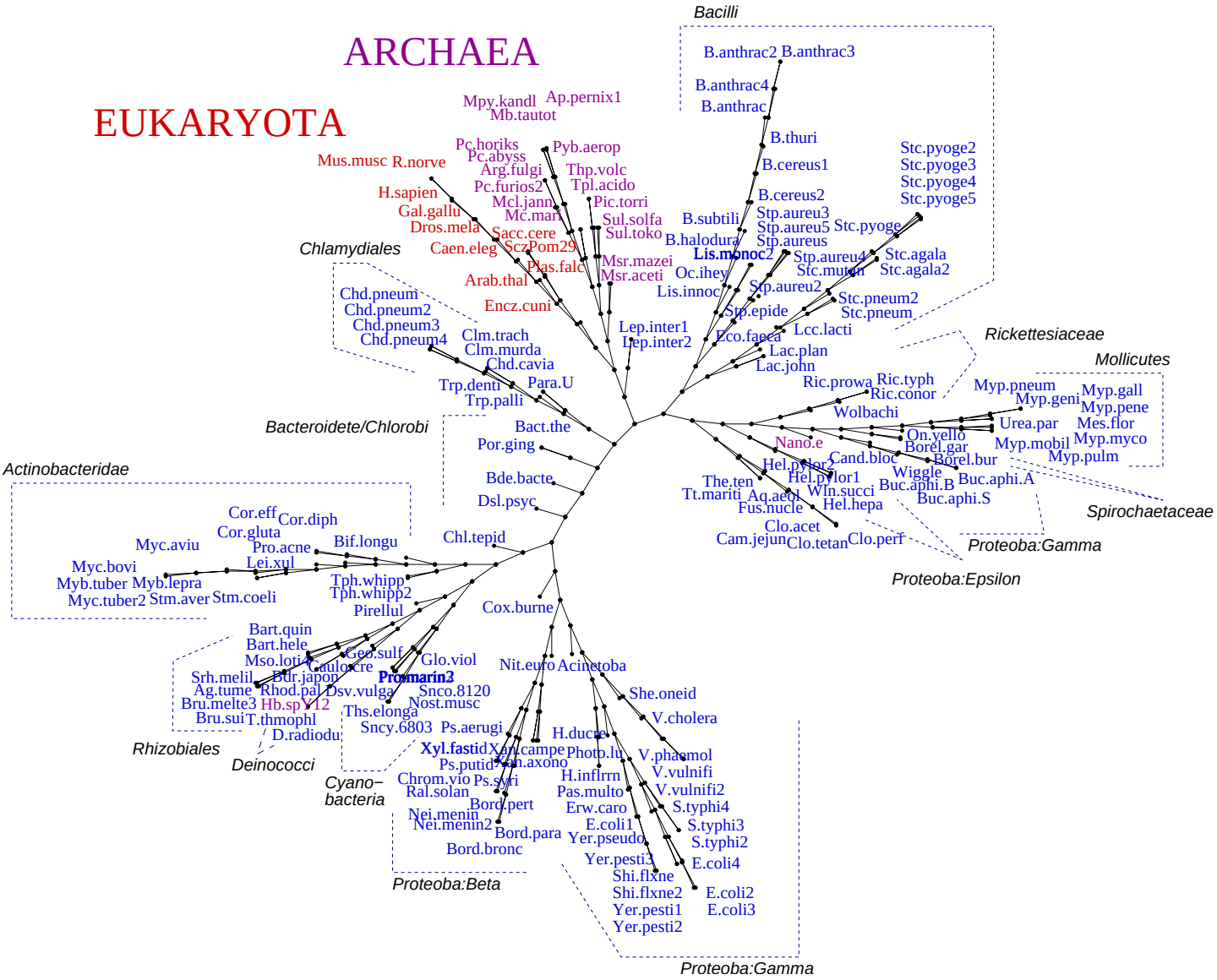

Figure 2: Tree of 191 proteomes generated by the ACS method. The tree has been drawn using DRAWTREE program of the PHYLIP package [24].

	A6	A8	C1	C2	C3	CB	N1	N2	N3	N4	NL	N5	N6	ACS	Li	Con
A6	0	42	34	46	40	44	32	40	44	40	48	40	40	40	40	36
A8	42	0	42	48	38	46	42	40	46	42	50	44	40	42	42	38
C1	34	42	0	40	40	32	30	26	34	26	44	28	36	28	26	22
C2	46	48	40	0	48	42	36	40	40	40	50	38	44	42	40	38
C3	40	38	40	48	0	46	42	40	46	44	48	42	36	42	42	38
CB	44	46	32	42	46	0	38	34	32	34	44	30	38	30	24	26
N1	32	42	30	36	42	38	0	28	38	32	44	28	30	30	30	22
N2	40	40	26	40	40	34	28	0	38	30	40	16	32	24	24	18
N3	44	46	34	40	46	32	38	38	0	34	48	32	44	28	30	32
N4	40	42	26	40	44	34	32	30	34	0	44	28	36	28	24	28
NL	48	50	44	50	48	44	44	40	48	44	0	40	38	36	40	34
N5	40	44	28	38	42	30	28	16	32	28	40	0	32	18	18	18
N6	40	40	36	44	36	38	30	32	44	36	38	32	0	28	32	28
ACS	40	42	28	42	42	30	30	24	28	28	36	18	28	0	14	16
Li	40	42	26	40	42	24	30	24	30	24	40	18	32	14	0	18
Con	36	38	22	38	38	26	22	18	32	28	34	18	28	16	18	0
Ave	40.83	43.33	34.33	42.66	42.5	38.33	35	33.66	39.66	35.83	44.83	33.16	37.16	32	31.69	29.07

Table 2: Summary of the distances (RF calculations using TREEDIST from the PHYLIP package) between evolutionary trees built from complete mtDNA of 34 mammalian taxa by different methods. The first 13 rows (and columns) correspond to the maximum likelihood trees for 13 mt-proteins. The rows and column ACS correspond to the tree reconstructed by the ACS method when the inputs were the 34 mitochondria genomes. The row and column Li stand for the tree of Li *at el.* [29] when the inputs were the 34 mitochondria genomes. The row and column Con stand for the majority consensus tree of the 13 ML trees constructed using CONSENSE from the PHYLIP package. The row Ave stands for the average distance from every tree to all the ML trees (excluding itself). For space (width) considerations, we abbreviated the protein names follows: Ai stands for $ATPi$, Ci for $COXi$, CB for $CYTB$, NL for $ND4L$, and Ni for NDi.

the known taxonomic division within the bacterial and archeal domains. This correspondence has been examined using the taxonomic information found in *NCBI Taxonomy Database* [15]. At large, the tree correctly distinguishes between most of the taxonomic groups in the dataset, making the disagreements between the trees a fertile ground for further comparison between the known taxonomy and the phylogeny revealed using genome comparison.

In the *Archaea* domain, we found a clear separation of genera represented by several species such as the *Pyrococcus* and *Methanosarcina*. The organization of the genera into classes, orders and families is less evident, possibly due to the relatively small number of specimen examined, as discussed below. Two archean species seem to be "misplaced" in the tree - *Nanoarchaeum equitans* (Nano.eq) and *Halobacterium NRC-1* (Hb.spY12) are both mapped on the tree within a mixture of prokaryote species. For *Nanoarchaeum* the reason may be the fact that it is one of a few known archeal parasites, lacking genes for lipid, cofactor, amino acid, or nucleotide biosynthesis [47], making it more problematic in light of our complete genome comparisons. The *Halobacterium sp.NRC-1* is the only archeal species present from the entire *Halobacteria* class, which might explain the difficulty of its classification. Its is found in the tree close to other stress-resistant species such as *D.radiodurans* and *T.Thermophilus*, belonging to the *Deinococci* class.

In the *Prokaryota* domain, the *Actinobacteria* class (high G+C Gram-positive bacteria) is clustered together with a correct separation of the majorly represented *Mycobacterium* (5 species) and *Corynebacterium* (3 species) genera. The same holds for the *Chlamydiae* class and *Cyanobacteria*

phylum (including correct separation of *Prochlorales* and *Chroococcales*). The largely represented *Firmicutes* phylum (Gram-positive bacteria) is clustered almost entirely on a single branch of the tree, with all the species within the *Clostridia* and the *Bacilli* classes clustered together, including divisions into orders, families and genera, which is largely in agreement with the taxonomic knowledge. Within the *Proteobacteria* (purple non-sulfur bacteria), the *Beta* and *Epsilon* classes are accurately separated. In the *Alpha* class the represented Rhizobiales class and Rickettsiaceae are both monophyletic groups in the tree (but not clustered together), the small *Delta* class is split, as is the large *Gamma* class, which is partitioned into two major branches on the tree.

Overall, the algorithm's ability to provide a phylogeny in good agreement with the taxonomic knowledge (which is largely based on the 16S rRNA sequences) is good at the lower levels of genera, families and classes. The method accuracy is decreasing for higher taxonomic groups, a common problem to the whole-genomic approach to phylogenetic inference, as has been reported in [37]. It is expected that the performance on these taxonomy groups will improve as more genomic sequences will become available, as we have experienced with the gradual increase in the number of species that were used in this study. It is natural that the tree construction Neighbor Joining algorithm will perform better when supplied with more specimen from each group.

3.4 Viruses forests

Viruses are known to be partitioned to a small number of superfamilies, according to their nuclear acid type: DNA or RNA, double strand or single strand, positive or negative. Each of these superfamilies is believed to have a different evolutionary origin [35]. We used our method for generating a forest for a large viruses' dataset. We collected $1,865$ viral genomes, where for $1,837$ of the viruses we had prior knowledge about their superfamily. We partitioned the viruses with known superfamily to one of following six superfamilies: dsDNA (double stranded DNA), dsRNA (double stranded RNA), retroid (reverse transcriptase viruses), ssDNA (single stranded DNA), ssRNAplus (single stranded RNA plus), ssRNAminus (single stranded RNA minus), and satellite nucleic acids. We attributed each virus with unknown family to the family that is closest to it, according the average ACS distance between the members of a family and the unknown virus. Then, we applied the ACS method and generated a tree for each of these superfamilies. We hereby describe in detail two trees in the forest (Fig. 3 and 4). For the sake of clarity all the branches have the same length.

3.4.1 Retroids

In this subsection we evaluate the consistency of the phylogenetic tree, constructed using the 83 viral genomes classified as *Retroid viruses*. The tree has been compared to the taxonomy appearing in the NCBI Taxonomy and ICTVdb [16]. The partition of the viruses to the 3 main families of reverse transcriptases : Hepadnaviridae, Caulimoviridae (Circular dsDNA reverse transcriptases) and Retroviridae (ssRNA reverse transcriptase), has been fully supported by our tree.

Within the *Hepadnaviridae* family (Hepatitis B viruses) the algorithm distinguished between the *Orthohepadnavirus* (mammalian) and the *Avihepadnavirus* (avian) genera. This included the *Ross Goose Hepatitis B virus* which is currently not classified, with evidence of its belonging to the Avihepadnavirus genus [46] and the *Arctic Squirrel Hepatitis virus* (classified with the *Orthohepadnaviruses*).

Among the *Caulimoviridae* family, the two main genera - the *Badnaviruses* (bacilliform DNA viruses) and *Caulimoviruses*, are separated in full accordance with the taxonomic data. The *Cestrum Yellow Leaf Curling virus*, considered a tentative member of the genus, is clustered together with the rest of the *Caulimoviruses*. Another genus of the *Caulimoviridae* - the *Petunia Vein Clearing virus*, is clustered close to the Caulimoviruses, as is the *Soybean Chlorotic Mottle*

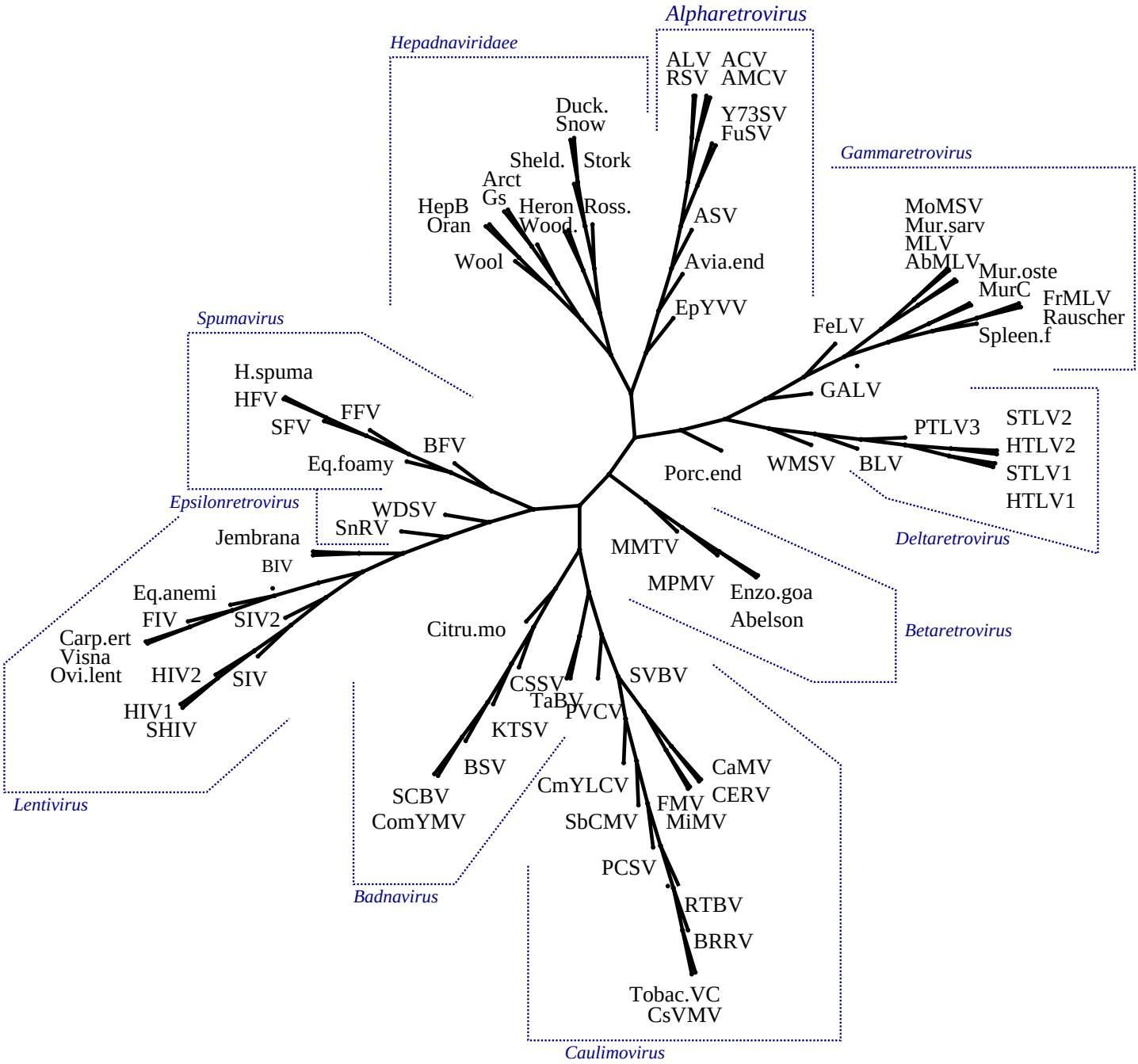

Figure 3: A tree of the retroid family generated using the ACS method. The common shortcut name is used where available.

virus, *Peanut Chlorotic Streak virus*, *Cassava Vein Mosaic virus*, and the *Rice Tungro Bacilliform-like viruses*. The location of these families suggests they are a possible members of the *Caulimovirus* genus.

The *Retroviridae* Family is correctly separated according to the *Orthoretrovirinae* and *Spumaretrovirinae* subfamilies. The sub-division of the Orthorerovirinae to the Alpha, Beta, Gamma, Delta, and Epsilon genera also fits the taxonomic data, except for some spreading of the *Gammaretroviruses*. The widely studied *Procine Endogenous retrovirus*, which is currently classified as a *Mammalian Type-C virus*, is classified among the Gammaretroviruses, fitting existing evidence of its protease resembling the protease of *MLV* within the gamma genus [4]. The yet unclassified *Avian Endogenous Retrovirus EAV-HP* clusters close to the Alpharetrovirus family, following a sequence identity previously reported in [41]. The *Lentivirus* genus members are clustered together, with a clear separation of the Primate (containing the *HIV*), Avian and Bovine species of the viruses. In the *Spumaretrovirinae* family the *Spumavirus* genus (foamy viruses) is clustered together.

The observations above that suggest our method is a valid predictive tool in the context of viral taxonomy, overcoming the shortcoming of various traditional methods.

3.4.2 ssRNAminus

The phylogenetic tree reconstructed using ACS for the family of ssRNA negative-strand viruses is described in figure 4. The ssRNA negative-strand viruses have been used for several small scale phylogeny analysis in the past [23, 7]. Few crucial points should be taken into account in the analysis: First, several of the ssRNA negative genomes are partitioned into two, three or more segments, which were treated separately in our analysis. They are denote by S,M and L suffixes, indicating the Small, Medium and Large segment, respectively. As could have been expected, it turn out that for every viral genus, the segments of the same relative size are clustered together in our tree, indicating their common origin. Second, the genomes, or genome segments used to perform this analysis were very small, sometimes less the 1Kbp. Such small sequences contain relatively little information for an algorithm as the one we used. Still, the tree which was created by our method nicely agrees with the accepted viral taxonomy, as described below:

- *Arenavirus* Genus (*Arenaviridae* Family): Arenaviruses are rodent-borne bisegmented viruses. The division of the Arenaviruses into Old World and New World is recognized. Bowen et al. [6] suggested a division of the latter into three lineages which were denoted as A, B and C. This division has been further reinforced by phylogeny analysis based on three genes common to the Arenaviruses [8]. Our analysis includes the *Pirital* virus of the A lineage, the *Junin* virus, the *Machupo* virus, the *Guanarito* virus and the *Tacaribe* virus from the B lineage and *Lassa virus* and *Lymphocytic choriomeningitis* virus from the Old World Arenaviruses. The large segments of the Arenaviruses are clustered closely and maintain the division between lineage A and B as well as the clustering of New World viruses together separately from the Old World viruses. The small segments are also clustered closely, but do not maintain as accurately the relationship between the lineages (although all the species in lineage B are adjacent in the tree). The Arenaviruses' large segment and small segment are clustered together, suggesting both segments possibly share a common ancestor.

- *Bunyavirus*, also known as *Orthobunyavirus* (*Bunyaviridae* family): Like all members of *Bunyaviridae* family, bunyaviruses have a trisegmented genome [12]. The segments of three bunyaviruses are included in the tree: *Bunyamwera* virus, *La Crosse* virus and *Oropouche* virus. The large segments of the viruses are included in a monophyletic group and so are the medium segments and the small segments.

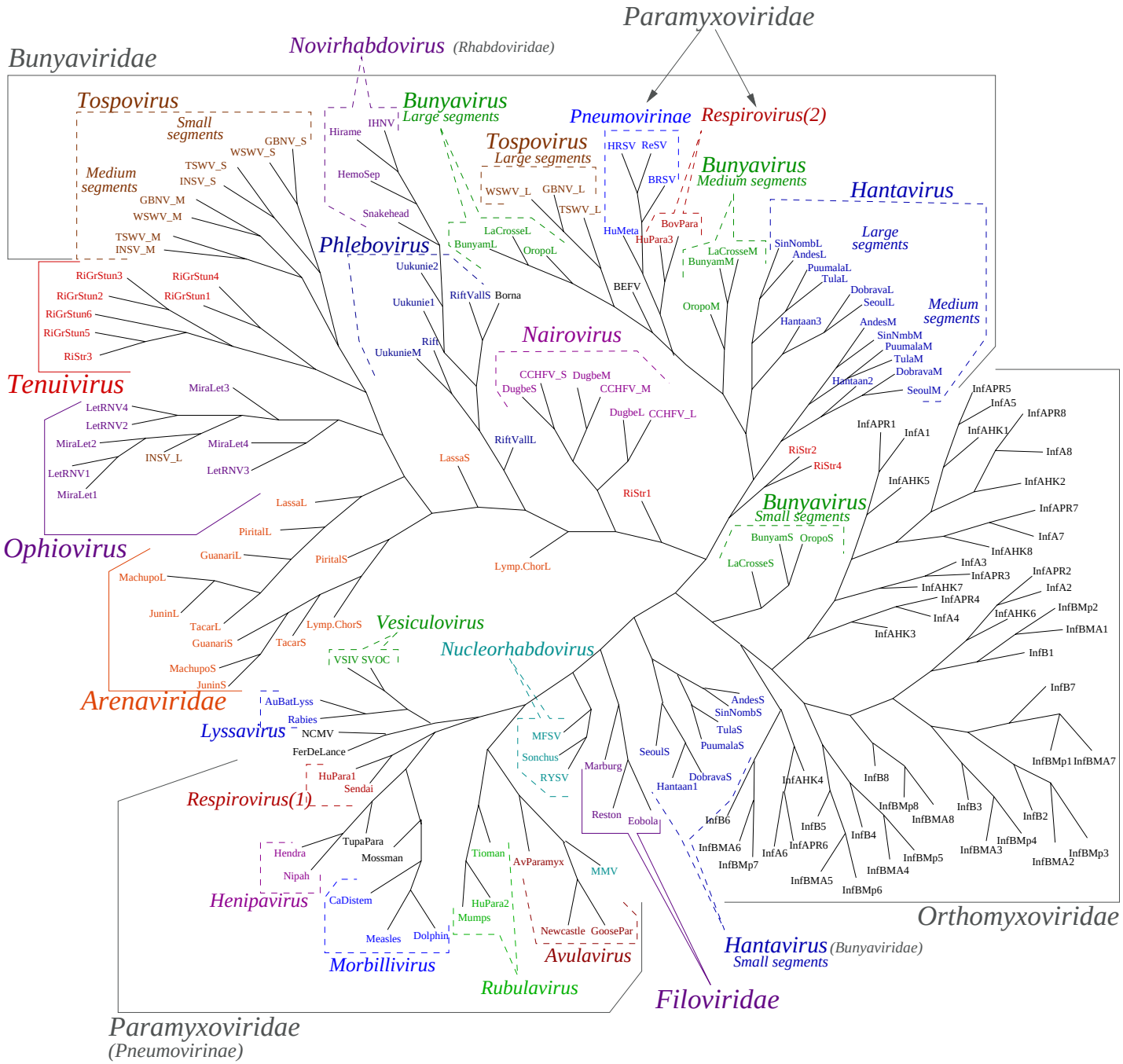

Figure 4: A tree of the ssRNAnegative family generated using the ACS method. The common shortcut name is used where available.

- *Hantavirus* (*Bunyaviridae*): Each hantavirus is specific to a different rodent or insectivore host. Consequently, virus phylogeny very closely reflects rodent phylogeny, implying that hantaviruses are ancient infectious agents which have coevulated with their rodents hosts. The hantaviruses analyzed in this work infect the rodents from the *Muridea* Family. The *Sin Nombre* and the *Andes* viruses infect species from the *Sigmodontinea* subfamily. The *Puumala* and *Tula* viruses infect members of the*Arvicolinae* subfamily, and the *Dobrova*, *Hantaan* and *Seoul* infect rodents from the *Murinae* subfamily [31]. In our tree all the large segments of hantaviruses are clustered together, and so are the medium and small segments. Moreover, the tree maintains the phylogeny relationship of the hosts subfamilies, clustering together the *Sin Nombre* and *Andes* viruses; the *Puumala* and *Tula* viruses, and the *Dobrova*, *Hantaan* and *Seoul*. This applies to the large, medium, and small segments.

- *Nairovirus* (*Bunyaviridae*): Two naioviruises are included in this analysis: the *Dugbe* virus and the *CCHF* (Crimean Congo Hemorrhagic Fever). Both are clustered together among other *Bunyaviridae*, furthermore the small, medium, and large segments together form a monophyletic group. This could mean that possibly all the segments of nairoviruses have a common origin.

- *Phlebovirus* (*Bunyaviridae*): The segments of the two phleboviruses (*Rift Valley fever* virus and *Uukuniemi* virus) are clustered in the same region of the tree.

- *Tospovirus* (*Bunyaviridae*): The small and medium segments of the tospoviruses are all clustered together, with an expected partition between the small and medium segments. The small and medium subtrees maintain the same evolutionary relationship: *GBNC* (Groundnut bud necrosis virus) closer to *WSWV* (Watermelon spotted wilt virus), and *INSV* (Impatiens necrotic spot virus) closer to *TSWV* (Tomato spotted wilt virus). In the cluster containing the large segment of the viruses the *INSV* virus is missing (the other three maintain the above mentioned relationship), and is clustered together with *Ophioviruses*.

- The *Paramyxoviridae* family: *Paramyxoviridae* viruses are located in two subtrees closely reflecting the family partition to the *Paramyxovirinae* subfamily and the *Pneumovirinae* subfamily. The three *Avulaviruses* are clustered together in proximity to the *Rubulaviruses*, which are in one cluster as well. The three *Morbilliviruses* form a monophyletic group close to the two viruses of the newly established genus *Henipavirus*. Between these family are located two unclassified *Paramyxoviridae* viruses: *Tupaia Paramyxovirus* and *Mossman* virus. Their position between the *Morbilliviruses* and *Henipavirus* in our tree agree with their accepted phylogeny location [32]. The *Respiroviruses* are partitioned to two clusters: One contains the *Sendai* virus and the *Human Parainfluenza virus 1* in the *Paramyxovirinae* subtree. The second *Respirovirus* cluster includes the *Human Parainfluenza virus 3* and *Bovine Parainfluenza virus 3*, and is clustered with all the examined species from the *Paramyxovirinae* family in a different subtree. Those species include viruses form the *Pneumovirus* and *Metapneumovirus* genera.

- *Fer-De-Lance* virus is an unclassified member of the *Paramyxovirinae* subfamily. Previous phylogenetic studies of the virus' proteins indicate that it is not consistently more closely related to any known paramyxovirus genus or species than to others. Specific protein phylogenies suggested that *Fe-De-Lance* virus was slightly closer to respiroviruses than to other genera [26]. The Fer-De-Lance virus is indeed located in our tree in a certain proximity to respiroviruses but is clustered together with viruses from the *Rhabdoviridae* family. It had been found that Fer-De-Lance virus contains an ORF similar to ORFs reported to encode small basic proteins in few rhabdoviruses [26]. This might mean the Fer-De-Lance virus is evolutionary close to the *Rhabdoviridae*, which explains their proximity in our tree.

- The *Filoviridae* family: The three members of the Filoviridae family, including the widely studied Ebola virus, are clustered together in close proximity to a large family of Paramyxoviridae viruses.

- The *Rhabdoviridae* family: The species of the *Vesiculovirus*, *Lyssavirus* and *Cytorhabdovirus* genera of the Rhabdoviridae are clustered together in one subtree. The *Novirhabdovirus* genus members creates a monophyletic group which is close to the *Phlebovirus* of the *Bunyaviridae* family. Three out of four members of the *Nucleorhabdovirus* genus are situated in a subtree with the fourth (*MMV*, *Maize Mozaic* virus). The only member of the *Ephemerovirus* genus, *BEFV*: *Bovine Ephemernal Fever* Virus is clustered distantly from any other member of the Rhabdoviridae genus.

- The *Orthomyxoviridae* family includes various influenza viruses which are all clustered together in the tree.

- The *Ophiovirus* genus: the ophioviruses are ssRNA negative viruses which are not part of any of the existing families in the order. All the members of the ophiovirus genus are situated in the tree together, close to the *Tenuivirus* genus. Indeed, according to ICTV [12], the virus morphology of the ophioviruses resembles the Tenuviruses. It further mentions that their internal nucleocapsid component is similar to that of members in the *Bunyaviridae* family. This similarity can explain the proximity of the Ophioviruses to the Topsoviruses of the *Bunyaviridae* family in the tree.

- The *Tenuivirus* genus also doesn't belong to any of the ssRNA negative families. Two species were analyzed in the tree: the first is *Rice Grassy Stunt* Virus (RiGrStun), which consists of six segments, all of which where clustered together adjacent to the medium and small segments of the *Tospoviruses* (*Bunyaviridae* family). The second is the *Rice Stripe Virus* (RiStr), which has four segments, one of which clustered with the RiGrStun segments and the other three in different locations among viruses from the *Bunyaviridae* family. This proximity to the viruses from the *Bunyaviridae* family fits the ICTV record on the genus which mentionS that Tenuiviruses share some similarities with viruses classified in the family Bunyaviridae.

4 Conclusion

In this work we presented a novel method, the ACS algorithm, for phylogenetic reconstruction, based on complete genome or proteome sequences. As with any new reconstruction method, its adequacy will be determined with time, when sufficient experience is gained. Yet the several large cases analysed in this work indicate its high potential. The comprehensive comparisons with other whole genome methods show that ACS is at least as good, and usually better, than all of them, both in terms of reconstruction accuracy and of computational efficiency (speed). We used our method for large scale phylogenetic analysis of two hundred species and two thousand viruses. This is the first time such a large scale phylogeny is performed. It provides many new phylogenetic insights, which can be further investigated in light of the available taxonomic knowledge.

The viral phylogenies provided by ASC allow a comprehensive view of the ancestral relationships between multiple genomic elements in organisms that are currently vaguely described in phylogenetic terms. Analysis of the obtained trees provides novel evidence for the taxonomic placements of multiple species, in many cases augmented by the limited available morphological details. We believe that in the future, the use of whole genome methods should assist in rapid classification of novel viral organism, following the relatively easy sequencing of the genetic content.

The ASC method also allows the comparison of the phylogenies constructed traditionally, using the information derived from individual genes, and those based on whole genomes or proteomes. This can be used for a systematic comparison of the "history" as told by the genome, to the histories told by separate genes. Such task was carried out here with the ML trees obtained from single mtDNA genes and the ASC whole-mtDNA tree.

We believe that our ACS approach is promising, and that its outcomes are interesting, so that overall this is an important step in the of direction constructing whole genome or proteome phylogenies. The experimental results support further exploration of the proposed method. However, this work is certainly not the last algorithmic word in this direction, and many improvements remain to be discovered and developed.

For example, our distance matrices were generated using either proteomic or genomic data. We believe that combining those two sources of information can improve the quality of the reconstruction. However, theoretical based approaches for combining such two different sources of information are still missing. Two similar genomes may share many reversed subsequences (subsequences that have direction that is reversed in one genome compared to the other) and not only subsequences with the same orientation. Another interesting direction is to generalize our algorithm to deal with this observation.

Supplementary Material

Some of the phylogenies, distance matrices used to generate the phylogenies, and the species listing in the phylogenies are available at
http://www.cs.tau.ac.il/∼bchor/whole/GREPS.html.

Acknowledgements

We would like to thanks Eran Bacharach, Tal Pupko, and Jacob Ziv for helpful discussions.

References

[1] J. Adachi and M. Hasegawa. Protml: Maximum likelihood inference of protein phylogeny. http://cmgm.stanford.edu/phylip/protml.html.

[2] A. Ben-Dor, B. Chor, D. Graur, R. Ophir, and D. Pelleg. Constructing phylogenies from quartets: elucidation of eutherian superordinal relationships. *Journal of computational Biology*, 5:377–390, 1998.

[3] O. Bininda-Emonds. *Phylogenetic Supertrees: Combining Information to Reveal the Tree of Life*. Kluwer series in Computational Biology, 2004.

[4] J. H. Blusch, S. Seelmeir, and K. V. Helm. Molecular and enzymatic characterization of the porcine endogenous retrovirus protease. *Virol*, 76(15):7913–17, August 2002.

[5] G. Bourque and P. A. Pevzner. Genome-scale evolution: Reconstructing gene order in ancestral species. *Genome Research*, 12:26–36, 2002.

[6] M. D. Bowen, C. J. Peters, and S.T. Nichol. The phylogeny of new world (tacaribe complex) arenaviruses. *Virology*, 219:285–290, 1996.

[7] J. M. Bujnicki and L. Rychlewski. In silico identification, structure prediction and phylogenetic analysis of the 2'-o-ribose (cap 1) methyltransferase domain in the large structural protein of ssrna negative-strand viruses. *Protein Engineering*, 15(2):101–108, 2002.

[8] R. N. Charrel, J. J. Lemasson, M. Garbutt, R. Khelifa, P. De Micco P, H. Feldmann, and de X. Lamballerie. New insights into the evolutionary relationships between arenaviruses provided by comparative analysis of small and large segment sequences. *Virology*, 317:191–196, 2003.

[9] X. Chen, S. Kwong, and M. Li. A compression algorithm for dna sequences and its applications in genome comparison. *RECOMB*, pages 107–117, 2000.

[10] B. Chor and T. Tuller. Maximum likelihood of evolutionary trees is hard. *RECOMB2005*, 2005.

[11] JR Cole, Chai B, Marsh TL, Farris RJ, Wang Q, Kulam SA, Chandra S, McGarrell DM, Schmidt TM, Garrity GM, and Tiedje JM. The ribosomal database project (rdp-ii): previewing a new autoaligner that allows regular updates and the new prokaryotic taxonomy. *Nucleic Acids Res*, 31:442–443, 2003.

[12] B. O. Cornelia. Ictvdb (international committee on taxonomy of viruses database). http://phene.cpmc.columbia.edu/Ictv/index.htm.

[13] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. J. Wiley and sons, New York, 1991.

[14] C. Darwin. *On the origin of species*. First edition edition, Nov 24 1859.

[15] NCBI Taxonomy Database. http://www.ncbi.nlm.nih.gov/entrez/linkout/tutorial/taxtour.html.

[16] The Universal Virus Database. http://www.ncbi.nlm.nih.gov/ictvdb/ictvdb/.

[17] S. Downie and J. Palmer. Use of chloroplast dna rearangements in reconstructing plant phylogeny. *In: P. Soltis and D. Soltis and J. Doyle (Eds.), Plant Molecular Systematics, Chapman and Hall*, pages 14–35, 1992.

[18] A. Drummond and K. Strimmer. Pal: An object-oriented programming library for molecular evolution and phylogenetics. *Bioinformatics*, 17:662–663, 2001.

[19] R. Durbin, S. R. Eddy, A. Krogh, and G. Mitchison. *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge University Press, 1998.

[20] D. M. Endres and J. E. Schindelin. A new metric for probability distribution. *IEEE Tran. Inf. Theory.*, 49(7), 2003.

[21] NCBI Genome Entrez. http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=genome.

[22] M. Farach, M. Noordewier, S. Savari, L. Shepp, A. Wyner, and J. Ziv. On the entropy of dna: Algorithms and measurements based on memory and rapid. *Symposium on Discrete Algorithms*, 1994.

[23] H. Feldmann, H. D. Klenk, and A. Sanchez. Molecular biology and evolution of filoviruses. *Arch Virol Suppl*, 7:81–100, 1993.

[24] J. Felsenstein. Phylip (phylogeny inference package) version 3.5c. *Distributed by the author. Department of Genetics, University of Washington, Seattle*, 1993.

[25] S. Hannenhalli and P. Pevzner. Transforming cabbage into turnip (polynomial algorithm for sorting signed permutations by reversals). *In Proc. 27th Annual ACM Symposium on the Theory of Computing*, pages 178–189, 1995.

[26] G. Kurath, W. N. Batts, W. Ahne, and J. R. Winton. Complete genome sequence of fer-delance virus reveals a novel gene in reptilian paramyxoviruses. *Journal of Virology*, 78(4):2045–2056, 2004.

[27] A. Lempel and J. Ziv. On the complexity of finite sequences. *IEEE Trans. Inf. Theory*, 22:75–88, 1976.

[28] A. Lempel and J. Ziv. A universal algorithm for sequential data compression. *IEEE Trans. Inf. Theory.*, 1977.

[29] M. Li, J. Badger, X. Chen, S. Kwong, P. Kearney, and H. Zhang. An information-based sequence distance and its application to whole mitochondrial genome phylogeny. *Bioinformatics*, 17(2):149–154, 2001.

[30] B. Ma, M. Li, and L. Zhang. From gene trees to species trees. *SIAM*, 1998.

[31] C. McCaughey and C. A. Hart. Hantaviruses. *J. Med. Microbiol.*, 49:587–599, 2000.

[32] P. J. Miller, D. B. Boyle, B. T. Eaton, and L. Wang. Full-length genome sequence of mossman virus, a novel paramyxovirus isolated from rodents in australia. *Virology*, 317:330–334, 2003.

[33] B. M. E. Moret, L. S. Wang, T. Warnow, and S. K. Wyman. New approaches for reconstructing phylogenies from gene order data. *bioinformatics*, 17:165–173, 2001.

[34] M. Nelson. *LZW Data Compression.* 1989.

[35] Origins of viruses. http://www.mcb.uct.ac.za/tutorial/virorig.html.

[36] H. H. Otu and K. Sayood. A new sequence distance measure for phylogenetic tree construction. *Bioinformatics*, 19(16), 2003.

[37] J. Qi, B. Wang, and B. Hao. Whole proteome prokaryote phylogeny without sequence alignment: a k-string composition approach. *J. Mol. Evol.*, 58(1):1–11, 2004.

[38] P. T. Raul, B. Gordon, and E. Oliver. *in Bininda-Emonds, Olaf R.P. (ed), Phylogenetic Supertrees: Combining Information to Reveal the Tree of Life*, chapter Quartet Supertrees, pages 173–191. Kluwer Academic (In Press), Dordrecht, the Netherlands, 2004.

[39] A. Reyes, C. Gissi, G. Pesole, F.M. Catzeflis, and C. Saccone. Where do rodents fit? evidence from the complete mitochondrial genome of sciurus vulgaris. *Mol. Biol. Evol.*, 17:979–983, 2000.

[40] D. R. Robinson and L. R. Foulds. Comparison of phylogenetic trees. *Mathematical Biosciences*, 53:131–147, 1981.

[41] M. A. Sacco, D. M. J. Flannery, K. Howes, and K. Venugopal. Avian endogenous retrovirus eav-hp shares regions of identity with avian leukosis virus subgroup j and the avian retrotransposon art-ch. *J Virol*, 74(3):1296–1306, February 2000.

[42] N. Saitou and M. Nei. The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol. Biol. Evol.*, 4:406–425, 1987.

[43] K. Sayood. *Introduction to data compression.* second edition.

[44] B. Stefan and J. Krkkinen. Fast lightweight suffix array construction and checking. pages 55–69, 2003.

[45] G. W. Stuart and M. W. Berry. A comprehensive whole genome bacterial phylogeny using correlated peptide motive defined in a high dimensional vector space. *Journal of Bioinformatics and Computational Biology*, 1(3):475–493, 2003.

[46] M. Triyatnib, P. L. Ey, T. Tran, M. L. Mire, M. Qiao, C. J. Burrell, and A. R. Jilbert. Sequence comparison of an australian duck hepatitis b virus strain with other avian hepadnaviruses. *Journal of General Virology*, 82:373–378, 2001.

[47] E. Waters, M. J. Hohn, I. Ahel, D. E. Graham, M. D. Adams, M. Barnstead, K. Y. Beeson, L. Bibbs, R. Bolanos, M. Keller, K. Kretz, X. Lin, E. Mathur, J. Ni, M. Podar, T. Richardson, G. G. Sutton, M. Simon, D. Sll, K. O. Stetter, J. M. Short, and M. Noordewier. The genome of nanoarchaeum equitans: Insights into early archaeal evolution and derived parasitism. *Proc Natl Acad Sci U S A*, 100(22):12984–12988, October 2003.

[48] P. Weiner. Linear pattern matching algorithms. *Proc. 14th IEEE Annual Symp. on Switching and Automata Theory*, pages 1–11, 1973.

[49] A. D. Wyner and A. J. Wyner. An improved version of lempel-ziv algorithm. *IEEE Tran. Inf. Theory.*, 1995.

[50] A. J. Wyner. *String matching theorems and applications to data compression and statistics.* Ph.d. thesis, Stanford, 1993.