Molecular Therapy
Nucleic Acids
Original Article

ENJ algorithm can construct triple phylogenetic trees

Yan Hong,[1] Maozu Guo,[2,3] and Juan Wang[1,4]

[1]School of Computer Science, Inner Mongolia University, Hohhot 010021, P.R. China; [2]School of Electrical and Information Engineering, Beijing University of Civil Engineering and Architecture, Beijing 100044, P.R. China; [3]Beijing Key Laboratory of Intelligent Processing for Building Big Data, Beijing 100044, P.R. China; [4]Stage Key Laboratories of Reproductive Regulation & Breeding of Grassland Livestock, Hohhot 010021, Inner Mongolia, P.R. China

Phylogenetic analysis is used to analyze the evolution of species according to the characteristics of biological sequences. The analytical results are generally represented by phylogenetic trees. NJ (neighbor joining) is a frequently used algorithm for constructing phylogenetic trees because of its few assumptions, fast operation, and high accuracy, and is based on the distance between taxa. It is known that NJ usually constructs different phylogenetic trees for the same dataset with differences in input order, which are known as "tied trees." This article proposes an improved method of NJ, called ENJ (extended neighbor joining). The ENJ can join several (currently limited to three) nodes with the same minimum distance into a new node, rather than joining two nodes in one iteration, so it can construct triple phylogenetic trees. We have inferred the formulas for updating the distance values and calculating the branch lengths for the ENJ algorithm. We have tested the ENJ with simulated and real data. The experimental results show that, compared with other methods, the trees constructed by the ENJ have greater similarity to the initial trees, and the ENJ is much faster than the NJ algorithm. Moreover, we have constructed a phylogenetic tree for the novel coronavirus (COVID-19) and related coronaviruses by ENJ, which shows that COVID-19 and SARS-CoV are closer than other coronaviruses. Because it differs from the existing phylogenetic trees for those coronaviruses, we constructed a phylogenetic network for them. The network shows those species have had a reticulate evolution.

INTRODUCTION

The aim of molecular phylogenetic analysis is primarily to construct phylogenetic trees for revealing evolutionary relationships of the species being researched.[1] The phylogenetic tree not only reflects the evolutionary history of species with a common ancestor but also the evolutionary time between them.[2,3] Reconstructing phylogenetic trees through the existing biological sequences is a vital topic in bioinformatics. Accurately and reliably inferring the evolutionary relationships between species can help people understand the evolutionary history and mechanism of organisms.[4–6] The novel coronavirus (COVID-19) has caused an epidemic of human acute respiratory syndrome throughout the world. By constructing a phylogenetic tree for COVID-19, researchers can reveal the evolutionary relationships among coronaviruses and identify the source of infection. The construction of a phylogenetic tree provides the basis for the development of drugs to treat the novel coronavirus.

The NJ (neighbor joining) algorithm is a widely used method for constructing phylogenetic trees, based on the distance between species. NJ is a greedy algorithm, which endeavors to minimize the sum of all the branch lengths of the resulting tree.[7,8] Researchers have, for a long time, made improvements in the NJ algorithm,[9–14] especially in the speed with which it constructs phylogenetic trees. For example, the RNJ (relaxed neighbor-joining) algorithm[15,16] joins any pair of plausible neighbor nodes and is faster and more suitable for inferring large trees. The Rapid NJ algorithm[17] proposes a new search strategy for selecting the next pair of neighbor nodes, and experiments have shown that it accelerates the construction of a phylogenetic tree compared with the NJ algorithm. The ERapid NJ algorithm[18] improves on the performance of Rapid NJ with external memory and reduces the memory requirement for reconstructing the phylogenetic tree for large datasets. The INJ (improved neighbor joining) algorithm[19] constructs phylogenetic trees by iteratively joining two pairs of neighbor nodes and joining them as two new nodes, which accelerates the construction process. FastJoin combines the INJ algorithm with the upper-bound computation optimizations of the Rapid NJ and the external storage of the ERapid NJ to reconstruct phylogenetic trees. The Neighbor Joining Method Plus[20] reconstructs phylogenetic trees by taking the sequence with children as the internal node and the sequence without children as the leaf node. There is no limit to the number of neighbor nodes in the tree obtained, which means that the resulting tree is not solely a binary tree. The FastNJ[21] is a fast implementation of the RNJ and the FastJoin and has shown a significant increase in the speed with a minimal loss in accuracy. The LNJ (live neighbor joining) algorithm[22] extends the numeric rationale of the NJ algorithm and introduces live ancestors to flexibly join taxa with minimum branch lengths.

Received 19 June 2020; accepted 5 November 2020;
https://doi.org/10.1016/j.omtn.2020.11.004.

Correspondence: Juan Wang, School of Computer Science, Inner Mongolia University, Hohhot 010021, P.R. China.
E-mail: wangjuan@imu.edu.cn

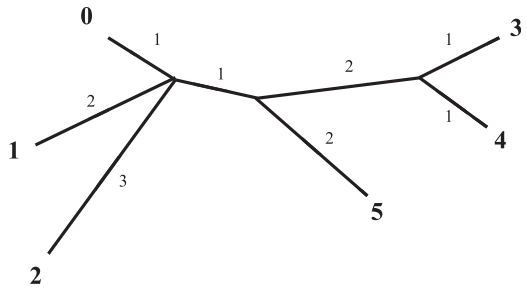

Figure 1. A simulated initial tree with six taxa, 0–5; the number on the branches is the branch length

In this article, we introduce another improved NJ algorithm, called the ENJ (extended neighbor joining). It constructs phylogenetic trees with a distance matrix between species as the input, just like other NJ algorithms. Experiments show that the ENJ can effectively and efficiently construct triple phylogenetic trees, which can better represent real evolutionary information.

RESULTS

We tested the performance of the ENJ on the simulated data and with real data and compared it with NJ and other improved NJ methods. The experiments were performed on a personal computer with an Intel Core i5-4200 U, 1.6 GHz CPU, and 4 GB RAM. All programs are written in Java.

Data

The experimental data are divided into two parts: simulated data and real data. The simulated phylogenetic trees were generated with Java program, known as the initial trees. Each initial tree contains only one node of degree 3; of which, the branch lengths were randomly assigned integers from 1 to 3. For example, an initial tree with six taxa randomly generated by the program is shown in Figure 1. The branch lengths of the initial tree were added to obtain an additive distance matrix, which was used as the input data. The real data comes from the article by Backeljau et al.,[23] namely, the distance matrix between the mtDNA cytochrome *b* sequences of the bear, which can be used directly as the input data.

Comparison of the dissimilarity between phylogenetic trees

To verify the dissimilarity between the resulting tree and the initial tree, we ran the ENJ, NJ, INJ, and Rapid NJ algorithms on the simulated data, measuring the dissimilarity between the trees by partition distance.[24] If the distance between two phylogenetic trees is smaller, it indicates the similarity between the two trees is greater. In contrast, if the distance between two phylogenetic trees is greater, it indicates the similarity between the two trees is less.

With increases in taxa, it takes a long time to calculate the partition distance between phylogenetic trees. Therefore, we chose small simulated data with $5 \leq$ taxa ≤ 250 for the experiment. Table S1 shows the partition distances and the degree of dissimilarity among the trees constructed with the ENJ, NJ, INJ, Rapid NJ, and the initial tree on the simulated data with $5 \leq$ taxa ≤ 250. In Table 1, the row labeled "$(d_p)_{\text{mean}}$" is the mean of the partition distances between the tree constructed by that method and the initial tree on the simulated data with $5 \leq$ taxa ≤ 250; the row labeled "$(d_p/d_{\max})_{\text{mean}}$" is the mean of the degree of dissimilarity on the simulated data with $5 \leq$ taxa ≤ 250, where $d_{\max}$ is the maximum value of the partition distance $d_{\max} = n - 1$. The value range for $(d_p/d_{\max})_{\text{mean}}$ is [0, 1], where closer it is to 0 the less dissimilarity exists; conversely, the greater the distance, the greater is the dissimilarity. Experimental results show that the mean of the partition distances with the ENJ is 18.37398, which is less than the other three methods, especially the NJ and the Rapid NJ. In terms of the degree of dissimilarity, the means of the ENJ, NJ, INJ, and Rapid NJ are 0.117919, 0.205531, 0.131936, and 0.173476; among which, the ENJ is closer to 0. It was, therefore, concluded that, compared with the NJ, INJ, and Rapid NJ, the partition distances between trees constructed with the ENJ and the initial tree are the smallest, which indicates that the similarity between the trees constructed by the ENJ and the initial trees is greater, and the accuracy is better. Similarly, the degree of dissimilarity between the tree constructed by ENJ and the initial tree is the smallest. Therefore, the ENJ algorithm better represents the information of the initial trees.

Table 1. The mean of partition distances and the degree of dissimilarity among the trees constructed by the ENJ, NJ, INJ, and Rapid NJ and the initial tree on the simulated data with $5 \leq$ taxa ≤ 250

Partition distance	ENJ trees and initial trees	NJ trees and initial trees	INJ trees and initial trees	Rapid NJ trees and initial trees
$(d_p)_{\text{mean}}$	18.37398	31.28862	19.54065	25.69919
$(d_p/d_{\max})_{\text{mean}}$	0.117919	0.205531	0.131936	0.173476

Comparison of the running time

We know that the INJ and Rapid NJ are effective improvements on the NJ algorithm; both of which accelerate the tree construction process. Here, we compare the running time of the ENJ, NJ, INJ, and Rapid NJ on the small simulated trees with $5 \leq$ taxa < 500 and the

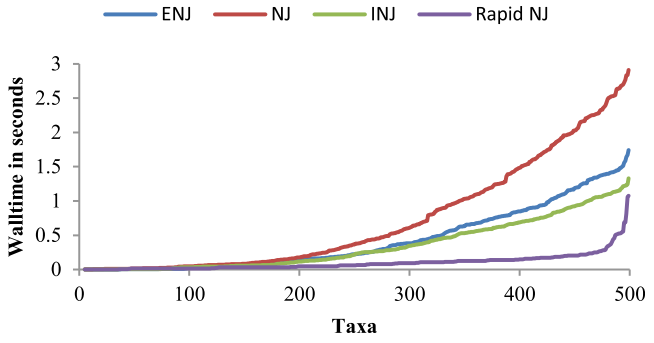

Figure 2. Running time of ENJ, NJ, INJ, and Rapid NJ on the small simulated data (5 ≤ taxa < 500)

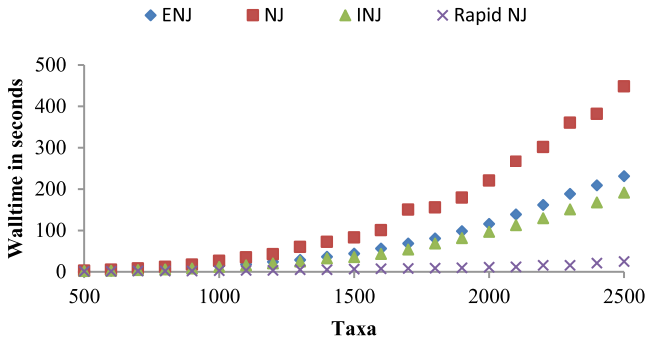

Figure 3. Running time of ENJ, NJ, INJ, and Rapid NJ on the large simulated data (500 ≤ taxa ≤ 2,500)

larger simulated trees with 500 ≤ taxa ≤ 2,500 (with a step size of 100), where the running time is in seconds. Figure 2 shows the running times of the ENJ, NJ, INJ, and Rapid NJ on the small simulated data (5 ≤ taxa < 500), and the means of the running times are, respectively, 0.43 s, 0.72 s, 0.35 s, and 0.10 s. Figure 3 shows the running times of the ENJ, NJ, INJ, and Rapid NJ on large simulated data (500 ≤ taxa ≤ 2,500), and the means of the running times are, respectively, 73.34 s, 139.41 s, 60.05 s, and 8.01 s. Tables S2 and S3 record the running time for each simulated data set. It can clearly be seen in Figures 2 and 3 that the running time of the ENJ is longer than that of the Rapid NJ and is close to that of the INJ but, importantly, is almost half as long as that of the NJ.

Comparison of the consistency of phylogenetic trees

The order of input data is known to have a great influence on the results of a NJ algorithm. That is, given a set of homologous sequences, the NJ may construct several different phylogenetic trees when the order of the input data is different. Therefore, we tested the consistency between the resulting trees and the real data. Table 2 shows the distance matrix of the mtDNA cytochrome *b* sequence of the bear as the input data. When the order of the input data is changed, the NJ algorithm constructs two different phylogenetic trees, as shown in Figure 4. However, the ENJ algorithm constructs only one phylogenetic tree, as shown in Figure 5.

DISCUSSION

Phylogenetic analysis of the novel coronavirus

Since December 2019, an outbreak of pneumonia caused by a novel coronavirus in Wuhan, P.R. China, has touched people around the world and has been named by the World Health Organization as "2019-nCoV." With joint effort from many scientists, the related research on the novel coronavirus has made great progress.[25–28] Throughout history, there were previously six kinds of coronaviruses known to infect humans; among which, HCoV-OC43, HCoV-NL63, HCoV-HKU1, and HCoV-229E can cause cold symptoms, and SARS-CoV and MERS-CoV can cause a severe respiratory syndrome. Understanding the evolutionary relationships among coronaviruses is the basis for further study of the novel coronavirus. In the latest study, Lu et al.[29] compared the complete genome of the COVID-19 with the other known coronaviruses and found that the similarity between COVID-19 and SARS-CoV was 79%, whereas that of COVID-19 and MERS-CoV was 50%. In the following results, the ENJ algorithm was used to construct a phylogenetic tree for all seven coronaviruses.

We obtained genomes for the seven coronaviruses from the NCBI database, which included HCoV-OC43 (NCBI: NC_006213.1), HCoV-NL63 (NCBI: NC_005831.2), HCoV-HKU1 (NCBI: NC_006577.2), HCoV-229E (NCBI: NC_002645.1), SARS-CoV (NCBI: NC_004718.3), MERS-CoV (NCBI: NC_019843.3), and COVID-19 (NCBI: NC_045512.2). Next, we calculated the distance between the sequences with the JCV algorithm.[30] The JCV algorithm uses the FASTA format to input genome data and produces the distance matrix between the sequences as the output, which, therefore, does not require complex alignment and avoids the uncertainty of phylogenetic trees constructed with a single gene. Considering the evolutionary direction of natural selection, the JCV algorithm finds all possible DNA sequences for a given species and codes them in order into a feature vector, which is then used to calculate the evolutionary distance between different species. Finally, the distance matrix that is obtained by adding the distances between the sequences is used as the input data.

The ENJ was used to construct the phylogenetic tree for the seven coronaviruses known to infect humans, as shown in Figure 6. The phylogenetic tree shows that COVID-19 and SARS-CoV are obviously

Table 2. Distance matrix of the mtDNA cytochrome *b* sequence of the bear

Taxa	abruz	pyren	kodia	capt3	capt4	capt5	grizz	pola2
Pyren	0.013							
Kodia	0.043	0.043						
Capt3	0.043	0.043	0.007					
Capt4	0.027	0.023	0.050	0.050				
Capt5	0.030	0.030	0.013	0.013	0.037			
Grizz	0.017	0.017	0.027	0.027	0.023	0.020		
Pola2	0.020	0.020	0.03	0.030	0.027	0.023	0.003	
Black	0.087	0.080	0.100	0.100	0.100	0.087	0.090	0.094

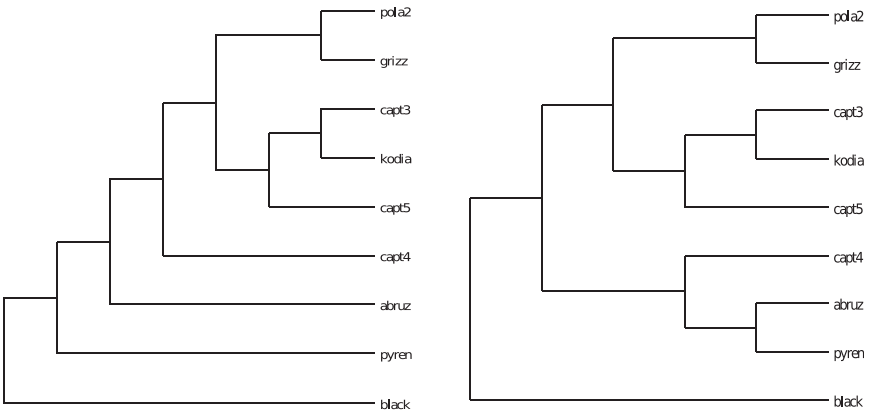

Figure 4. Two different phylogenetic trees constructed by the NJ algorithm for the mtDNA cytochrome *b* sequence of the bear

clustered, while they are far away from MERS-CoV. Therefore, the COVID-19 coronavirus is different from SARS-CoV and MERS-CoV and, of the two, is more closely related to SARS-CoV. Accurately inferring the evolutionary history of COVID-19 can provide an important basis for subsequent vaccine development.

Zhou et al.[31] constructed a phylogenetic tree based on the nucleotide sequences from complete coronavirus genomes using the maximum-likelihood method, and the evolutionary relationships of the seven coronaviruses are shown in Figure 7. The results show that the COVID-19 shares a common ancestor with the SARS-CoV and is slightly distant from the MERS-CoV. Similarly, the phylogenetic tree constructed by the ENJ also demonstrates that relationship. The difference is that the maximum-likelihood method joins SARS-CoV and COVID-19 into a class, HCoV-OC43 and HCoV-HKU1 into a class and HCoV-NL63 and HCoV-229E into another class. The ENJ algorithm joins SARS-CoV and COVID-19 into a class and joins with HCoV-229E; HCoV-NL63 and HCoV-HKU1 into a class and joins with HCoV-OC43.

Because of the conflicting information contained in those two phylogenetic trees, we performed further phylogenetic analyses. We constructed a phylogenetic network for the coronaviruses with the Frin algorithm.[32] The Frin algorithm constructs phylogenetic networks based on taxa frequency and the degree of incompatibility. The two phylogenetic trees in Figures 6 and 7 were used as input data, and the Frin algorithm was used to construct a level-2 network with r = 2 and c = 4, as shown in Figure 8. The resulting network shows that COVID-19 and SARS-CoV share a direct common ancestor, which is far from MERS-CoV, and two reticular events occurred during the evolution of the seven coronaviruses. As shown, HCoV-NL63 was derived from reticular evolutionary events, as was the parent of COVID-19 and SARS-CoV. Therefore, it is concluded that the evolution of the seven coronaviruses may involve complex reticular evolutionary relationships and are not limited to simple, linear

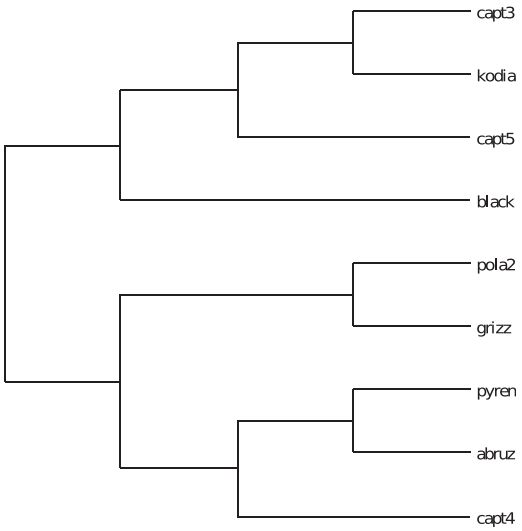

Figure 5. The phylogenetic tree constructed by the ENJ algorithm for the mtDNA cytochrome *b* sequence of the bear

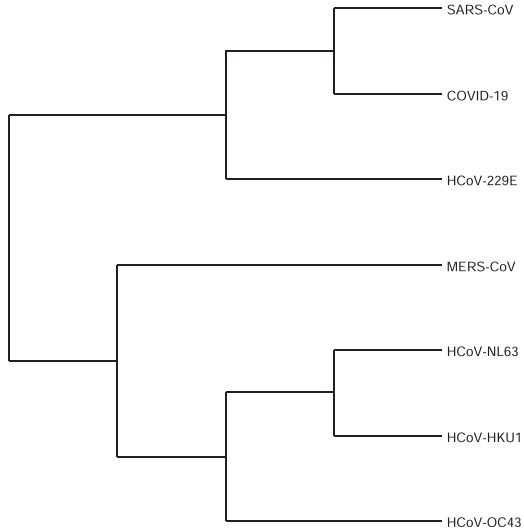

Figure 6. ENJ constructs the phylogenetic tree for COVID-19, SARS-CoV, MERS-CoV, HCoV-229E, HCoV-OC43, HCoV-NL63, and HCoV-HKU1

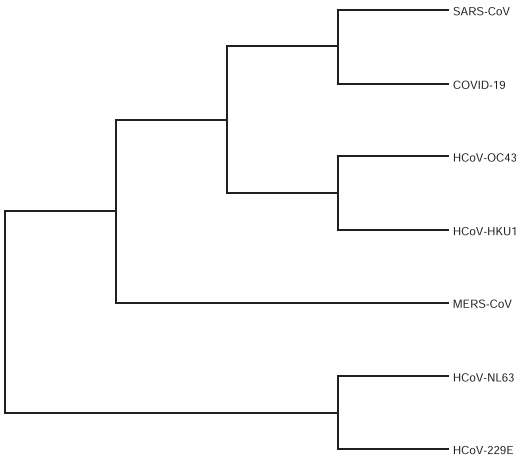

Figure 7. Maximum-likelihood method constructs the phylogenetic tree for the coronaviruses that are known to infect humans

evolutionary relationships. To prevent the recurrence of human infections, further studies on the evolution and variation of coronaviruses are needed.

Conclusions

In this article, we propose an improved method for constructing phylogenetic tress, called the ENJ. In the construction process, the ENJ can simultaneously deal with the two smallest values in the same row or column of the sum matrix and can join three nodes to construct a triple phylogenetic tree. Experiments with simulated data show that the phylogenetic trees constructed by the ENJ have greater similarity with the initial trees than the other methods, and the ENJ greatly accelerates the speed of the NJ algorithm in constructing the phylogenetic trees. In addition, the experiment with real data shows that the ENJ can effectively avoid the problem of "tied trees," and the resulting trees are more consistent. Application of the ENJ algorithm on the novel coronavirus shows that it can effectively reflect the evolutionary relationships among coronaviruses. Furthermore, we constructed the phylogenetic network for COVID-19 and the related coronaviruses, which indicates the reticular relationships among the coronaviruses. Experimental results show that the ENJ algorithm is an efficient and effective method. First, the trees reconstructed with the ENJ better represent the information of the initial trees. Second, the ENJ constructs phylogenetic trees swiftly. Third, the ENJ effectively constructs phylogenetic trees to solve the problem in which the resulting tree is not unique. These facts indicate that the ENJ algorithm can better describe biological evolution.

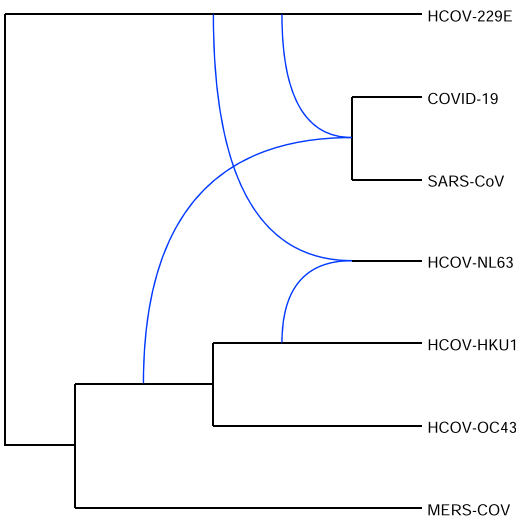

Figure 8. Frin algorithm constructs a level-2 phylogenetic network for COVID-19, SARS-CoV, MERS-CoV, HCoV-229E, HCoV-OC43, HCoV-NL63, and HCoV-HKU1

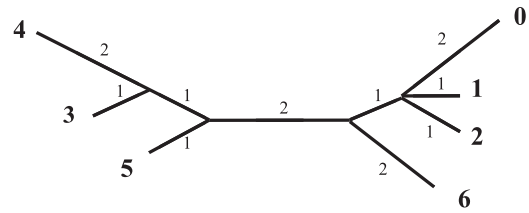

Figure 9. An unrooted tree *T* with seven taxa

MATERIALS AND METHODS

Preliminaries

The NJ algorithm begins with a star topology, iteratively selects two nodes neighboring the root, and then joins them by inserting a new internal node between the root and the two selected nodes. Based on the principle of minimum evolution, the NJ algorithm minimizes the sum of the branch lengths for the resulting tree. The article by Saitou and Nei[7] has proven that when the distance matrix $D = (D_{ij})_{n\times n}$ is purely additive, taxa k and taxa l are true neighbor nodes when S_{kl} is the smallest in the sum matrix S. However, the minimum running time for the algorithm as formulated was unclear. Studier and Keppler[8] presented alternative formulas that run in $O(n^3)$, and the sum of the branch lengths is calculated as follows:

$$S_{ij} = (n-2)D_{ij} - R_i - R_j, \quad (1 \le i \ne j \le n), \qquad \text{(Equation 1)}$$

where $R_i = \sum_{k=0}^{n-1} D_{ik}$.

Table 3. Lower triangular distance matrix *D* for the tree *T*

Taxa	0	1	2	3	4	5
1	3					
2	3	2				
3	7	6	6			
4	8	7	7	3		
5	6	5	5	3	4	
6	5	4	4	6	7	5

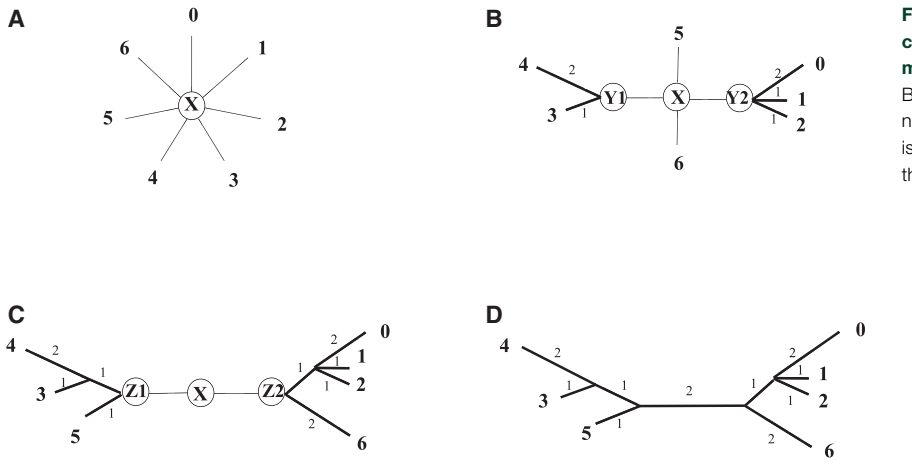

Figure 10. The process by which the ENJ algorithm constructs a phylogenetic tree for the distance matrix *D*

Blackbody numbers are taxa, which need to cluster. The numbers on the branches indicate the branch lengths. (X) is the center node, (Y) is the inserted new nodes, and (Z) is the remaining internal nodes.

The distance between a new node $a = k \cup l$ and an old node o is calculated as follows:

$$D_{ao} = \frac{1}{2}(D_{ko} + D_{lo} - D_{kl}), \quad (o \neq k, l) \qquad \text{(Equation 2)}$$

The branch lengths of a new node $a = k \cup l$ are calculated as follows:

$$D_{ak} = \frac{1}{2(n-2)}[(n-2)D_{kl} + R_k - R_l] \qquad \text{(Equation 3)}$$

$$D_{al} = \frac{1}{2(n-2)}[(n-2)D_{kl} + R_l - R_k] \qquad \text{(Equation 4)}$$

The INJ algorithm is an improvement on the NJ algorithm by accelerating the NJ when constructing phylogenetic trees. The article by Wang et al.[19] proved that, when the distance matrix $D = (D_{ij})_{n\times n}$ is purely additive, removing members in row k, column k, row l, and column l, when S_{kl} is the smallest in matrix S, taxa p and q are also true neighbor nodes when S_{pq} is the smallest of the remaining members. The INJ algorithm differs from the NJ algorithm in that the INJ iteratively joins two pairs of true neighbor nodes to create two new nodes. The distance between the two new nodes $a = k \cup l$ and $b = p \cup q$ is calculated as follows:

$$D_{ab} = \left(D_{kp} + D_{lp} + D_{kq} + D_{lq} - D_{kl} - D_{pq}\right)/4 \qquad \text{(Equation 5)}$$

Table 4. Lower triangular sum matrix *S* for the distance matrix *D*

First iteration: neighbor nodes = (4, 3), (1, 0, 2)						
Taxa	0	1	2	3	4	5
1	−44.0					
2	−44.0	−44.0				
3	−28.0	−28.0	−28.0			
4	−28.0	−28.0	−28.0	−52.0		
5	−30.0	−30.0	−30.0	−44.0	−44.0	
6	−38.0	−38.0	−38.0	−32.0	−32.0	−34.0

The shortcomings of the NJ algorithm are as follows: first, it may produce several resulting trees, and it is difficult to determine which one reflects the real evolutionary history. Second, it can only construct binary trees, which sometimes cannot reflect complex evolutionary relationships. Third, the NJ algorithm usually takes a long time to construct phylogenetic trees.

The ENJ algorithm

When searching for true neighbor nodes, the NJ algorithm ignores the case in which the same value exists in the same column and row as the smallest value in sum matrix S, which is the direct reason behind the problem of tied trees.

The ENJ is an effective improvement on the NJ algorithm. The ENJ uses the searching strategy for the true neighbor nodes of the INJ to effectively combine the possible three nodes and constructs a triple phylogenetic tree. Therefore, the ENJ iteratively joins one or two pairs of neighbor nodes, until a single node remains. When searching for true neighbor nodes, the ENJ algorithm accounts for the possibility of several minimum values in the row and column of the smallest value in sum matrix S and avoids the problem of tied trees by joining three true neighbor nodes. When the problem of tied trees does not exist, the ENJ is the same as the INJ algorithm. When the problem of tied trees exists, the ENJ algorithm joins three true neighbor nodes and constructs a triple phylogenetic tree.

The ENJ algorithm also uses the additive matrix formed by the distance between the given taxa as the input and begins with a star tree. Equation 1 is used to calculate the elements of the sum matrix, which is the criterion for selecting true neighbor nodes. When the ENJ algorithm joins two true neighbor nodes to create a new node, Equation 2 is used to calculate the distance between the new node and the other nodes, and Equations 3 and 4 are used to calculate the branch lengths.

When the ENJ joins three true neighbor nodes k, l, and z to create a new node, the following equations are inferred:

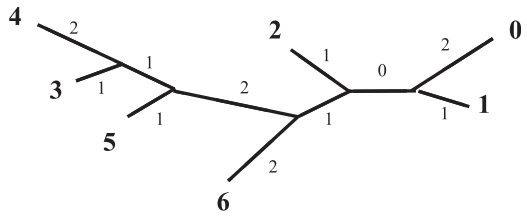

Figure 11. The phylogenetic tree constructed by NJ for the distance matrix *D*

The distance between a new node $a = k \cup l \cup z$ and an old node o is calculated as follows:

$$D_{ao} = \frac{2(D_{ko} + D_{lo} + D_{zo}) - D_{kz} - D_{lz} - D_{kl}}{6}, \ (o \neq k, l, z) \quad \text{(Equation 6)}$$

The branch lengths of a new node $a = k \cup l \cup z$ are calculated as follows:

$$D_{ak} = \frac{1}{4(n-2)}[(n-2)(D_{kl} + D_{kz}) + 2R_k - R_l - R_z] \quad \text{(Equation 7)}$$

$$D_{al} = \frac{1}{4(n-2)}[(n-2)(D_{lk} + D_{lz}) + 2R_l - R_z - R_k] \quad \text{(Equation 8)}$$

$$D_{az} = \frac{1}{4(n-2)}[(n-2)(D_{zk} + D_{zl}) + 2R_z - R_k - R_l] \quad \text{(Equation 9)}$$

The detailed procedure for the ENJ algorithm is listed in Box 1.

The search time for the NJ is $n^2 + (n-1)^2 + (n-2)^2 + \ldots + 4^2 = n^3/3$; the worst-case search time for the ENJ is $n^2 + (n-2)^2 + (n-4)^2 + \ldots + 4^2 = n^3/6$, and the time for updating the distance matrix of the ENJ is half of that of the NJ. The ENJ improves the running speed of the NJ algorithm, and the most important thing is that the ENJ extends the NJ theoretically.

Example 1 describes the process of the ENJ algorithm constructing phylogenetic trees. Figure 9 shows an unrooted tree T with seven taxa, 0–6, which is the true topology and is called the initial tree. Table 3 provides the lower triangular distance matrix D between any two taxa inferred by the initial tree T, which is the input data.

The process of construction of the phylogenetic tree for the distance matrix D by the ENJ algorithm is shown in Figure 10. The ENJ begins with the star tree shown in Figure 10A. In the first iteration, we calculated the lower triangular sum matrix S with Equation 1, as shown in Table 4. It can be seen that S_{43} is the smallest in the sum matrix S, and its row and column do not have the same value. S_{10} and S_{20} are the smallest for the same column in the remaining elements after removing row 4, column 4, row 3, and column 3. Hence, (4, 3) and (1, 0, 2) are two pairs of true neighbor nodes that have joined them to obtain the tree topology shown in Figure 10B. Equations 3, 4, 7, 8, and 9 are used to calculate branch lengths. Afterward, Equations 2, 5, and 6 are used to update the distance matrix D. At that time, the number of remaining taxa is 5, and the next iteration continues. In the second iteration, we find two pairs of true neighbor nodes (4–3, 5) and (1–0–2, 6) and join them to obtain the tree topology shown in Figure 10C, calculating the branch lengths and updating the distance matrix. At that time, the number of remaining taxa is 3, and the ENJ algorithm stops and the resulting tree is output. The ENJ algorithm finally constructs the phylogenetic tree, as shown in Figure 10D.

Given distance matrix D in Table 3, the ENJ constructs the phylogenetic tree shown in Figure 10D and the NJ constructs the phylogenetic tree shown in Figure 11. The partition distance between the ENJ tree and the initial tree is 0, indicating that the tree constructed by the ENJ

Box 1 ENJ algorithm

Input: a distance matrix $D = (D_{ij})_{n \times n}$

Output: a phylogenetic tree T

$m \neq k \neq l$, $t \neq p \neq q$;

1. According to Equation 1, compute the sum matrix $S = (S_{ij})_{n \times n}$;
2. Find the smallest S_{kl} from S, remove the member in the row k, column k, row l, and column l, and find the smallest S_{pq};
3. If $S_{kl} = S_{km}/S_{ml}$, then join k, l, and m as a new node, denoted as $a = k \cup l \cup m$, $n = n - 2$;
4. If $S_{kl} \neq S_{km} \neq S_{ml}$, $S_{pq} = S_{pt}/S_{tq}$, then join k and l as a new node, denoted as $b1 = k \cup l$, and join p, q, and t as a new node, denoted as $b2 = p \cup q \cup t$, $n = n$-3;
5. If$S_{kl} \neq S_{km} \neq S_{ml}$,$S_{pq} \neq S_{pt} \neq S_{tq}$, then join k and l as a new node, denoted as $c1 = k \cup l$, and join p, q as a new node, denoted as $c2 = p \cup q$, $n = n$-2;
6. According to Equations 2, 5, and 6, update the distance matrix;
7. If $n > 3$, go to 1, otherwise output a phylogenetic tree.

algorithm is exactly the same as the initial tree. The distance between the NJ tree and the initial tree is 0.5, which indicates that the tree constructed by NJ algorithm is different from the initial tree to some extent. Example 1 shows that the ENJ algorithm can simultaneously join three taxa to construct a triple phylogenetic tree, and the resulting tree is closer to the initial tree. Therefore, the following conclusion is drawn: when necessary, the ENJ algorithm can effectively join three true neighbor nodes to construct a triple phylogenetic tree, and the resulting tree has greater similarity with the initial tree.

SUPPLEMENTAL INFORMATION

Supplemental Information can be found online at https://doi.org/10.1016/j.omtn.2020.11.004.

ACKNOWLEDGMENTS

This work has been supported by the National Natural Science Foundations of China (62061035, 62002181, and 61661040) and the Inner Mongolia Science & Technology Plan (2020GG0186).

AUTHOR CONTRIBUTIONS

Y.H. proposed the method; Y.H. and J.W. derived the formulas and designed the experiments; and all authors wrote the paper.

DECLARATION OF INTERESTS

The authors declare no competing interests.

REFERENCES

1. Nei, M., and Kumar, S. (2000). Molecular evolution and phylogenetics (Oxford University Press).
2. Elias, I., and Lagergren, J. (2008). Fast neighbor joining. Theor. Comput. Sci. *410*, 1993–2000.
3. St. John, K. (2017). Review paper: the shape of phylogenetic treespace. Syst. Biol. *66*, e83–e94.
4. Thomas, R.H. (2001). Molecular evolution and phylogenetics. Heredity *86*, 385.
5. Price, M.N., Dehal, P.S., and Arkin, A.P. (2009). FastTree: computing large minimum evolution trees with profiles instead of a distance matrix. Mol. Biol. Evol. *26*, 1641–1650.
6. McTavish, E.J., Drew, B.T., Redelings, B., and Cranston, K.A. (2017). How and why to build a unified tree of life. BioEssays *39*, 1700114.
7. Saitou, N., and Nei, M. (1987). The neighbor-joining method: a new method for reconstructing phylogenetic trees. Mol. Biol. Evol. *4*, 406–425.
8. Studier, J.A., and Keppler, K.J. (1988). A note on the neighbor-joining algorithm of Saitou and Nei. Mol. Biol. Evol. *5*, 729–731.
9. Gascuel, O. (1997). BIONJ: an improved version of the NJ algorithm based on a simple model of sequence data. Mol. Biol. Evol. *14*, 685–695.
10. Pearson, W.R., Robins, G., and Zhang, T. (1999). Generalized neighbor-joining: more reliable phylogenetic tree reconstruction. Mol. Biol. Evol. *16*, 806–816.
11. Bruno, W.J., Socci, N.D., and Halpern, A.L. (2000). Weighted neighbor joining: a likelihood-based approach to distance-based phylogeny reconstruction. Mol. Biol. Evol. *17*, 189–197.
12. Nakhleh, L., Moret, B.M.E., Roshan, U., St John, K., Sun, J., and Warnow, T. (2002). The accuracy of fast phylogenetic methods for large datasets. Pac. Symp. Biocomput. *7*, 211–222.
13. Mailund, T., and Pedersen, C.N. (2004). QuickJoin–fast neighbour-joining tree reconstruction. Bioinformatics *20*, 3261–3262.
14. Mailund, T., Brodal, G.S., Fagerberg, R., Pedersen, C.N., and Phillips, D. (2006). Recrafting the neighbor-joining method. BMC Bioinformatics *7*, 29.
15. Evans, J., Sheneman, L., and Foster, J. (2006). Relaxed neighbor joining: a fast distance-based phylogenetic tree construction method. J. Mol. Evol. *62*, 785–792.
16. Sheneman, L., Evans, J., and Foster, J.A. (2006). Clearcut: a fast implementation of relaxed neighbor joining. Bioinformatics *22*, 2823–2824.
17. Simonsen, M., Mailund, T., and Pedersen, C.N.S. (2008). Rapid neighbour-joining. In Proceedings of Algorithms in Bioinformatics—the 8th International Workshop, Lecture Notes in Computer Science, *Volume 5251*, K.A. Crandall and J. Lagergren, eds. Proceedings of Algorithms in Bioinformatics—the 8th International Workshop, Lecture Notes in Computer Science (Springer), pp. 113–122.
18. Simonsen, M., Mailund, T., and Pedersen, C.N.S. (2011). Inference of large phylogenies using neighbour-joining. Biomed. Eng. Syst. Technol. Int. *127*, 334–344.
19. Wang, J., Guo, M.Z., and Xing, L.L. (2012). FastJoin, an improved neighbor-joining algorithm. Genet. Mol. Res. *11*, 1909–1922.
20. Płoński, P., and Radomski, J.P. (2013). Neighbor joining plus—algorithm for phylogenetic tree reconstruction with proper nodes assignment. arXiv Pop. Evol., arXiv:1310.2114v1 [q-bio PE].
21. Li, J.F. (2015). A fast neighbor joining method. Genet. Mol. Res. *14*, 8733–8743.
22. Telles, G.P., Araújo, G.S., Walter, M.E.M.T., Brigido, M.M., and Almeida, N.F. (2018). Live neighbor-joining. BMC Bioinformatics *19*, 172.
23. Backeljau, T., De Bruyn, L., De Wolf, H., Jordaens, K., Van Dongen, S., and Winnepennincks, B. (1996). Multiple UPGMA and neighbor-joining trees, and the performance of some computer packages. Mol. Biol. Evol. *13*, 309–313.
24. Gusfield, D. (2002). Partition-distance: a problem and class of perfect graphs arising in clustering. Inf. Process. Lett. *82*, 159–164.
25. Benvenuto, D., Giovanetti, M., Ciccozzi, A., Spoto, S., Angeletti, S., and Ciccozzi, M. (2020). The 2019-new coronavirus epidemic: evidence for virus evolution. J. Med. Virol. *92*, 455–459.
26. Chen, N., Zhou, M., Dong, X., Qu, J., Gong, F., Han, Y., Qiu, Y., Wang, J., Liu, Y., Wei, Y., et al. (2020). Epidemiological and clinical characteristics of 99 cases of 2019 novel coronavirus pneumonia in Wuhan, China: a descriptive study. Lancet *395*, 507–513.
27. Chan, J.F.-W., Yuan, S., Kok, K.-H., To, K.K.-W., Chu, H., Yang, J., Xing, F., Liu, J., Yip, C.C., Poon, R.W., et al. (2020). A familial cluster of pneumonia associated with the 2019 novel coronavirus indicating person-to-person transmission: a study of a family cluster. Lancet *395*, 514–523.
28. Paraskevis, D., Kostaki, E.G., Magiorkinis, G., Panayiotakopoulos, G., Sourvinos, G., and Tsiodras, S. (2020). Full-genome evolutionary analysis of the novel corona virus (2019-nCoV) rejects the hypothesis of emergence as a result of a recent recombination event. Infect. Genet. Evol. *79*, 104212.
29. Lu, R., Zhao, X., Li, J., Niu, P., Yang, B., Wu, H., Wang, W., Song, H., Huang, B., Zhu, N., et al. (2020). Genomic characterisation and epidemiology of 2019 novel coronavirus: implications for virus origins and receptor binding. Lancet *395*, 565–574.
30. Wang, J., Guo, M., Che, K., Wang, C., Liu, X., and Liu, Y. (2013). A new distance computing method for DNA sequences in phylogenetic analysis. In Proceedings of the 10th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), J. Chen, X. Wang, L. Wang, J. Sun, and X. Meng, eds. (Institute of Electrical and Eelectronics Engineers), pp. 713–717.
31. Zhou, P., Yang, X.-L., Wang, X.-G., Hu, B., Zhang, L., Zhang, W., Si, H.R., Zhu, Y., Li, B., Huang, C.L., et al. (2020). A pneumonia outbreak associated with a new coronavirus of probable bat origin. Nature *579*, 270–273.
32. Hong, Y., and Wang, J. (2019). Frin: an efficient method for representing genome evolutionary history. Front. Genet. *10*, 1261.