**Development of Complex Curricula for Molecular Bionics and Infobionics Programs within a consortial* framework****

**Consortium leader**

# PETER PAZMANY CATHOLIC UNIVERSITY

**Consortium members**

# SEMMELWEIS UNIVERSITY, DIALOG CAMPUS PUBLISHER

The Project has been realised with the support of the European Union and has been co-financed by the European Social Fund ***

**Molekuláris bionika és Infobionika Szakok tananyagának komplex fejlesztése konzorciumi keretben

***A projekt az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

# INTRODUCTION TO BIOINFORMATICS

**(BEVEZETÉS A BIOINFORMATIKÁBA )**

## CHAPTER 12

## Phylogenetic Tree Algorithms

**(Filogenetikai fa algoritmusok)**

**András Budinszky**


TÁMOP – 4.1.2-08/2/A/KMR-2009-0006

# Definitions

A phylogenetic tree or evolutionary tree is a type of two-dimensional graph illustrating the inferred evolutionary relationships among various organisms or genes (including paralogs) based upon similarities and differences in their physical and/or genetic characteristics.

The outer nodes (or leaves) represent existing (observed) entities and the inner nodes represent (usually hypothetical) ancestors that are not part of the data.

Phylogenic prediction is trying to construct such trees based on available evidence (similarities and differences of characteristics, these days typically DNA, RNA and protein sequences).

# Morphological vs. Molecular Analysis

The morphological phylogenetics collects representative measurements for each of the phenotypic characteristics of the species into a matrix. The types of phenotypic data assembled depend on the individual species being compared; they may involve measurements of average body size, lengths or sizes of particular bones or other physical features, or even behavioral manifestations.

The molecular phylogenetics uses data that are immediate and discretely defined: distinct nucleotide sequences of DNA or RNA, and distinct amino acid sequences of proteins. However, defining homology is not easy due to the difficulties of multi-sequence alignment.

# History of Phylogenetic Trees

1840 Early representations of *branching* phylogenetic trees include a "Paleontological chart" showing the geological relationships among plants and animals (Hitchcock)

1859 Publishing of the book “The Origin of Species”. This book produced one of the first illustrations and crucially popularized the notion of an evolutionary tree based on anatomical features. However, using relatively subjective observations were often inconclusive, and sometimes the conclusion are incorrect (Darwin).

# History of Phylogenetic Trees (cont)

1963 Publishing the notion of a so-called "molecular clock and constructing the molecular phylogenies based on evolutionary information held by DNA (Zuckerkandl & Pauling)

1967 Publishing the first phylogenetic algorithm in the Science magazine starting the molecular phylogenetics (Fitch and Margoliash)

# History of Phylogenetic Trees (cont)

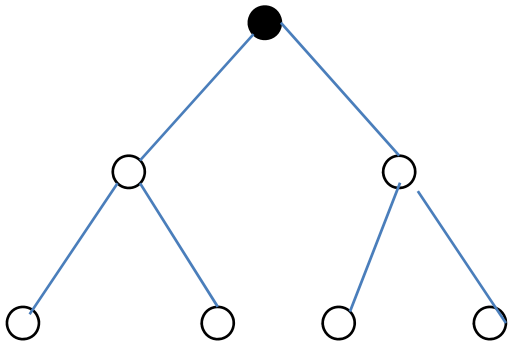
1985 Construction of evolutionary tree of humans (based on mtDNA, mitochondrial DNA with about 16,500 base point). This showed the so-called Out-of-Africa hypothesis (Wilson)

1986 Creation 100 distinct trees that were also consistent with the mtDNA data suggestion the incorrectness of the Out-of-Africa hypothesis (Templeton).

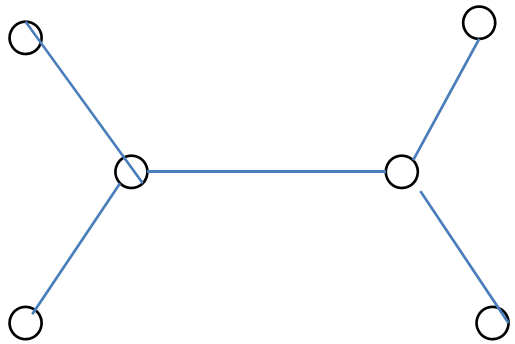
# Characteristics of Trees

As phylogeny trees binary trees are used; that makes sense based on evolutionary steps.

There are two different types of trees: rooted (when one of the nodes is special) and unrooted one. Unrooted tree is used when either the position of the root is of no interest or its position is difficult to determine.



rooted tree



unrooted tree

# Main Methods for Phylogenetic Predictions

There are usually one of the three following methods is being used to construct a phylogenetic tree that best fits the observations:

- Parsimony method – this approach assumes an implicit model of evolution (parsimony) and thus attempts to minimize the overall number of mutations for the entire tree.
- Distance-based method – this is simplest to implement, but does not assume an evolutionary model.
- Likelihood-based method – this is the most complex one and needs an explicit evolutionary model.

# Parsimony Method

This method looks for the tree with the lowest possible so-called parsimony score **(**sum of cost of all mutations in the tree).

The method is also referred to as the minimum evolution method.

The base data for the comparison of the different species could be morphological but – nowadays – it is molecular.

Its logic is based on Occam's razor principle.

# Occam (or Ockham) Razor Principle

This principle comes from a 14th-century English logician, theologian and Franciscan friar Father William of Ockham.

He wrote that "entities must not be multiplied beyond necessity".

This is also phrased as "plurality should not be assumed without necessity" (or as usually paraphrased: "keep it simple, stupid").

Nevertheless, in a scientific method, Occam's razor should not be considered an irrefutable principle of logic.

# Parsimony Method, Algorithm

Steps:

A. Selects those positions (columns) that are “informative” for parsimony analysis. A position is informative if at least 2 different characters occur on that position and each one appears at least in two sequences (there).

B. For every informative position

   a) Generates all possible unrooted trees with the given species (e.g., sequences of the same gene) at the leaves.

   b) Determines which of these trees require the smallest number of changes (see details of this step later, at Fitch algorithm) .

C. The tree with the smallest total changes will be returned.

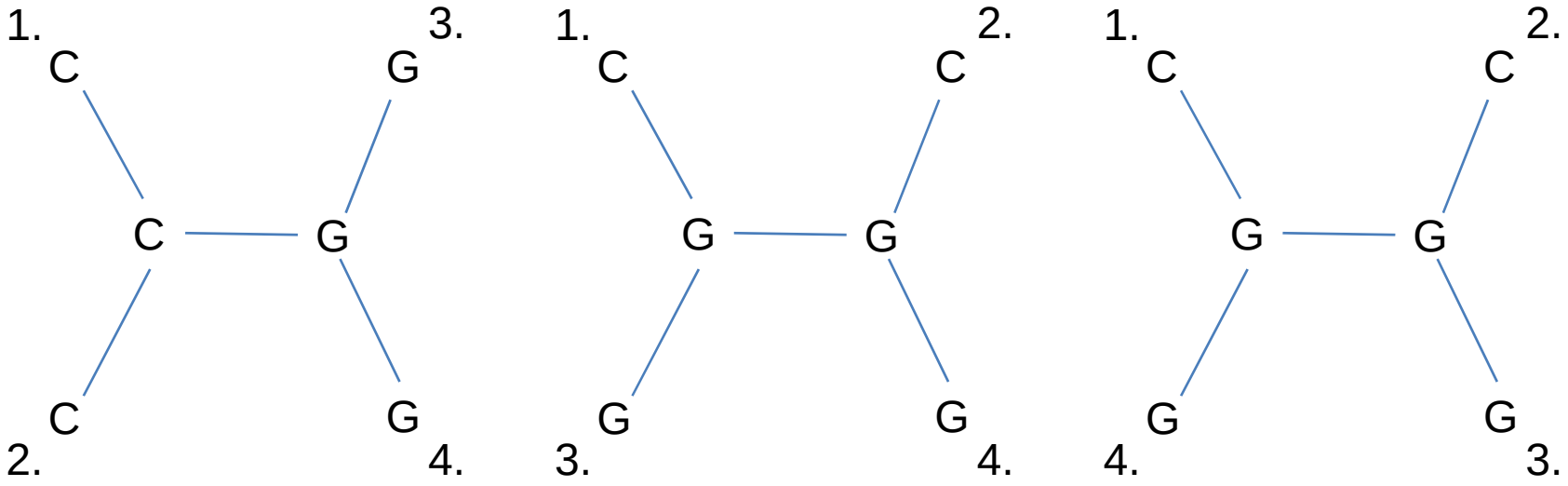
# Example for Parsimony Method

| Species | character positions | | | | | | |
|---|---|---|---|---|---|---|---|
| | 1. | 2. | 3. | 4. | 5. | 6. | 7. |
| 1. | C | A | G | A | G | A | A |
| 2. | C | A | A | T | A | A | G |
| 3. | G | A | A | C | A | G | C |
| 4. | G | A | G | G | A | G | A |

- There are 3 possible unrooted tree configurations.
- Positions 2.  is not informative because it has only character ‘A’.
- Positions 4., 5. and 7. are not informative because they have characters that appears only once in the position.
- Positions 1., 3, and 6 are informative.

# Parsimony Score for Position 1 in the Example

Here are three possible trees to be considered:




1 2 2

scores for the trees of position 1

# Fitch Algorithm

It is used to determine the associated parsimony score for a given tree configuration (on a given character position).

Steps of the algorithm (for convenience, the algorithm considers the tree rooted and works its way up from leaves to root):

A. For every leaf node $i$, assigns zero as initial parsimony score ($P_i$), and the character on that position as the only element in the possible base assignments set ($S_i$).

# Fitch Algorithm

Steps (continued):

A. For every parent node $k$ with children $i$ and $j$, computes $P_k$ and $S_k$ as
   - If $S_i \cap S_j$ is empty, then $P_k = P_i + P_j + 1$, $S_k = S_i \cup S_j$
   - If $S_i \cap S_j$ is not empty, then $P_k = P_i + P_j$, $S_k = S_i \cap S_j$

B. Returns $P_{root}$ as the parsimony score of this tree.

Note: This algorithm is executed for every tree configuration at every informative position.

# Weighted Parsimony Method

On the previous slide in the formula of computing the parsimony score we have used value 1 to score each change (mutation).

As we have discussed this at the alignment problems, studies of mutations show that the different mutations do not have the same frequency.

Therefore it makes sense to use a scoring matrix when computing the parsimony score.

# Difficulties with Parsimony Method

In the example we had only four sequences and thus three possible unrooted tree configurations.

In general, the number of possible unrooted trees for $n$ leaves is

$$\frac{(2n-3)!}{(n-2)!2^{n-2}}$$

Clearly this makes this algorithm NP-hard and for larger $n$ values heuristics must be applied.

# Distance-Based Method

This method explicitly based on a measure of "genetic distance" between the sequences provided by an MSA (multiple sequence alignment).

The first step of the method is finding the distances between all pairs of sequences and place these distances into a matrix.

Then the goal of the method is to identify a tree that fits the matrix; that is, places the sequences on the leaves and creates internal vertices with such lengths that adding up the lengths between each sequence pair, the distance measures in the matrix are all matched.

# Requirements/Definitions

The distances should meet the following three criteria:

- Symmetry: $d_{ij} = d_{ji}$ for each $i, j$
- Distinguishability: $d_{ij} \neq 0$ if, and only if $i \neq j$
- Triangle inequality: $d_{ij} \leq d_{ik} + d_{ki}$ for each $i, j, k$.

A tree is additive if the distance between any pair of leaves is the sum of distances between those leaves and the first node they share on the tree.

# Testing for additivity

In order to decide whether an additive tree fitting a given distance matrix $D$ can be constructed, the following so-called *four-point condition* for additive trees could be tested:

> $D$ corresponds to an additive tree if and only if for any of the four sequences (labeled 1, 2, 3, 4) two of the sums $d_{12} + d_{34}$, $d_{13} + d_{24}$, $d_{14} + d_{14}$, are equal and greater than or equal to the third.

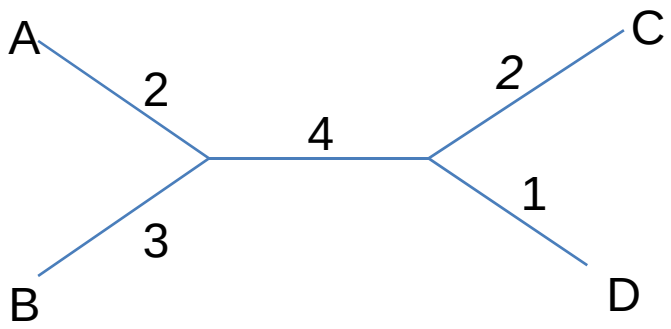
# Additive/Non-Additive Examples

additive matrix/tree

| | A | B | C | D |
|---|---|---|---|---|
| A | 0 | 5 | 8 | 7 |
| B | | 0 | 9 | 8 |
| C | | | 0 | 3 |
| D | | | | 0 |

non-additive matrix

| | A | B | C | D |
|---|---|---|---|---|
| A | 0 | 3 | 3 | 3 |
| B | | 0 | 5 | 3 |
| C | | | 0 | 3 |
| D | | | | 0 |




?

# Fitch-Margoliash Algorithm

Steps:

A. Finds the most closely related sequence pair in the distance matrix (say A and B), and makes a temporary distance matrix with A and B, and the rest of the sequences combined (as X).
B. Calculates the average distance from A and B to the node X.
C. From this special, 3-node case, computes the direct branches leading to A and B (see a follow-up slide).

# Fitch-Margoliash Algorithm

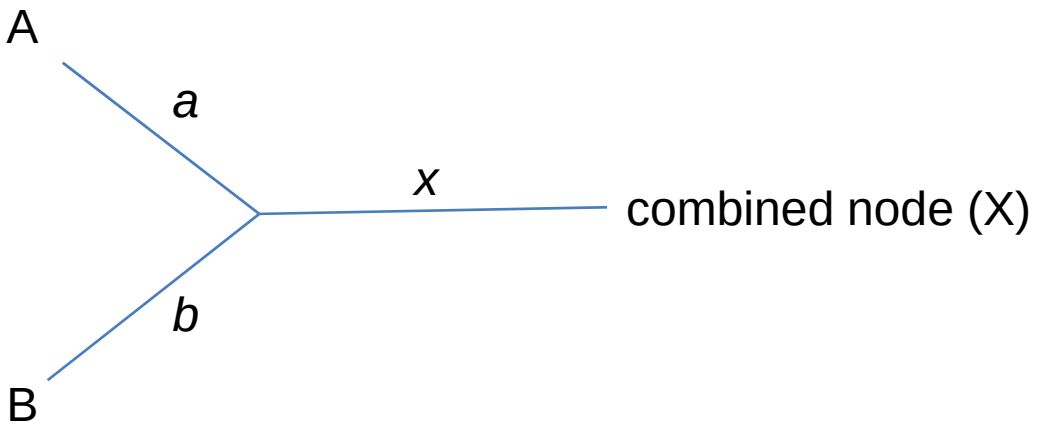
Steps (continued):

D. Reduces the original distance matrix by deleting rows and columns for A and B and inserting a new row and column for a composite (AB) entity with averaged distances (e.g. the distance from C to (AB) is the average distance of C to A and C to B).

E. Repeats steps A. through D. until a matrix is reduced to a 2 by 2 size.

Note: Step A. assumes that the closest sequence pairs are not necessarily neighbors, so the algorithm should be repeated starting with another pair and check if get the same (correct) tree.

# Solving the Special 3-Node Case




$$\text{distance(AB)} = a + b$$

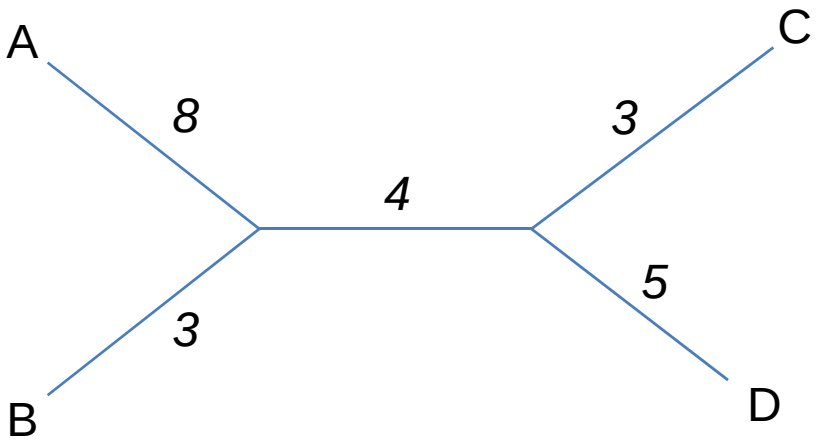
$$\text{distance(AX)} = a + x$$

$$\text{distance(BX)} = b + x$$

This is solved for $a$ and $b$ to get the direct branches to A and B

Note: This computation is used in step C. of the F-M algorithm

# Finding Neighboring Leaves




Closest leaves aren't necessarily neighbors.

Counter example: A and B are neighbors, but $(d_{AB} = 13) > (d_{BD} = 12)$

Finding neighbors could be difficult task.

# None-Additive Matrix

If there is no additive tree fitting a given distance matrix $D$ (that is, one of the quadruplets do not satisfy the 4-point condition), then we look for a tree $T$ that approximates $D$ the best:

$$\textbf{\textit{Squared Error}} : \sum_{i,j} (d_{ij}(T) - D_{ij})^2$$

Squared error is a measure of the quality of the fit between distance matrix and the tree: we want to minimize it.

Finding the best approximation tree $T$ for a non-additive distance matrix $D$ is NP-hard.

# Neighbor-Joining Algorithm

Very much like the Fitch-Margoliash algorithm except the choice of sequence pair (step A.) is controlled.

Developed by Naruya Saitou and Masatoshi Nei in 1987.

Finds a pair of leaves that are close to each other but far from other leaves: implicitly finds a pair of neighboring leaves.

Works well not only for additive distance matrices but for others as well.

The general neighbor-joining is available from

ftp.virginia.edu/pub/fasta/GNJ

# Maximum Likelihood Approach

This method uses standard statistical techniques for inferring probability distributions to assign probabilities to particular possible phylogenetic trees.

The method requires an a priory substitution model to assess the probability of particular mutations.

Roughly speaking, a tree that requires more mutations at interior nodes to explain the observed phylogeny will be assessed as having a lower probability.

It is similar to the parsimony method in the sense that the analysis is performed on each column of the multiple sequence alignment.

# Maximum Likelihood, Algorithm

Steps:

A. Generates all possible trees for the given sequences.
B. Places the characters at the first (next) position of the sequences on the leaves of all these trees.
C. Select first (next) tree ($T_i$).
D. Generates all variations of $T_i$ (a set of $T_i'$) by placing all different patterns of changes at the inner nodes.
E. Looks up the probability of change at each of the branching in each $T_i'$ tree in a probability table of an applied evolutionary model.

# Maximum Likelihood, Algorithm

Steps (continued):

F. Calculates the probability of pattern change defined in each $T_i'$ tree by multiplying the branch probabilities.

G. Sums the probabilities of all $T_i'$ trees to get the probability of $T_i$.

H. Repeats steps C. through G for each tree $T$.

I. Picks the highest probability value to the determine the most probable tree for this position.

# Maximum Likelihood, Algorithm

Steps (continued):

J. Repeats steps B. through I. for each position of the aligned sequences.

K. The most likely tree will be the one that provides the highest overall probability at all of the positions found by summing the position probabilities for each tree.

Note: Due to the fact that the maximum likelihood method considers all possible trees, it is only feasible for a small number of sequences.

# Problems with Building Phylogenetic Trees

Some of the assumptions used by the various algorithms are violated but not quite addresses yet:

- Evolution on the different sequence positions happens according to the same stochastic model
  - Not true; for example third positions evolve faster than first
- Sequence positions evolve independently of one another
  - Not true for example in sequences with secondary structures
- Sequences are aligned
  - Usually gaps are removed before building trees, however, the effects of this is hard to assess.

# Phylogenetic Tree Software

## PHYLIP

It is a free package of programs for inferring phylogenies. It is distributed as source code, documentation files, and a number of different types of executables. The program includes parsimony, distance matrix, and likelihood methods as well. (Felsenstein, 1989)

http://evolution.genetics.washington.edu/phylip.html

## PAUP

It has been the most widely used software package for the inference of evolutionary trees. Originally it used parsimony, but current version (4.0) extended with maximum likelihood and distance methods. (Swofford)

http://paup.csit.fsu.edu/

# Phylogenetic Tree Software (cont)

MacClade

It is a program for phylogenetic analysis. Its analytical strength is in studies of character evolution. It runs only on Mac platform. (David & Wayne Maddison)

http://macclade.org/macclade.html

MESQUITE

It is designed to analyze comparative data about organisms. Its emphasis is on phylogenetic analysis, but some of its modules concern population genetics, while others do non-phylogenetic multivariate analysis. (also David & Wayne Maddison)

http://mesquiteproject.org/

# Phylogenetic Tree Software (cont)

## MEGA (now MEGA5)

It is an integrated tool for conducting automatic and manual sequence alignment, inferring phylogenetic trees, mining web-based databases, estimating rates of molecular evolution, inferring ancestral sequences, and testing evolutionary hypotheses. It is a multi-threaded Windows application and runs on all releases of Microsoft Windows operating system. It is the most quoted phylogenetic program package.

http://www.megasoftware.net/

# References

Tree-of-life web site

http://tolweb.org/tree/phylogeny.html

Felsenstein's Phylogenetic Program Directory

http://evolution.genetics.washington.edu/phylip.html

UT Austin Phylogenetics Lab

http://kristin.csres.utexas.edu/

Woese Lab

http://www.life.uiuc.edu/micro/woese.html