Molecular Phylogenetics (Hannes Luz)

Contents:

- Phylogenetic Trees, basic notions
- A character based method: Maximum Parsimony
- Trees from distances
- Markov Models of Sequence Evolution, Maximum Likelihood Trees

References for lectures

- Joseph Felsenstein, Inferring Phylogenies, Sinauer Associates (2004)

- Dan Graur, Weng-Hsiun Li, Fundamentals of Molecular Evolution, Sinauer Associates

- D.W. Mount. Bioinformatics: Sequences and Genome analysis, 2001.

- D.L. Swofford, G.J. Olsen, P.J.Waddell & D.M. Hillis, Phylogenetic Inference, in: D.M. Hillis (ed.), Molecular Systematics, 2 ed., Sunderland Mass., 1996.

- R. Durbin, S. Eddy, A. Krogh & G. Mitchison, Biological sequence analysis, Cambridge, 1998

References for lectures, cont'd

- S. Rahmann, Spezielle Methoden und Anwendungen der Statistik in der Bioinformatik (http://www.molgen.mpg.de/~rahmann/afw-rahmann.pdf)

- K. Schmid, A Phylogenetic Parsimony Method Considering Neighbored Gaps (Bachelor thesis, FU Berlin, 2007)

- Martin Vingron, Hannes Luz, Jens Stoye, Lecture notes on 'Algorithms for Phylogenetic Reconstructions', http://lectures.molgen.mpg.de/Algorithmische_Bioinformatik_WS0405/phylogeny_script.pdf

Recommended reading/watching

- Video streams of Arndt von Haeseler's lectures held at the Otto Warburg Summer School on Evolutionary Genomics 2006 (http://ows.molgen.mpg.de/2006/von_haeseler.shtml)

- Dirk Metzler, Algorithmen und Modelle der Bioinformatik, http://www.cs.uni-frankfurt.de/~metzler/WS0708/skriptWS0708.pdf

Software links

- Felsenstein's list of software packages:
  `http://evolution.genetics.washington.edu/phylip/software.html`

- PHYLIP is Felsenstein's free software package for inferring phylogenies,
  `http://evolution.genetics.washington.edu/phylip.html`

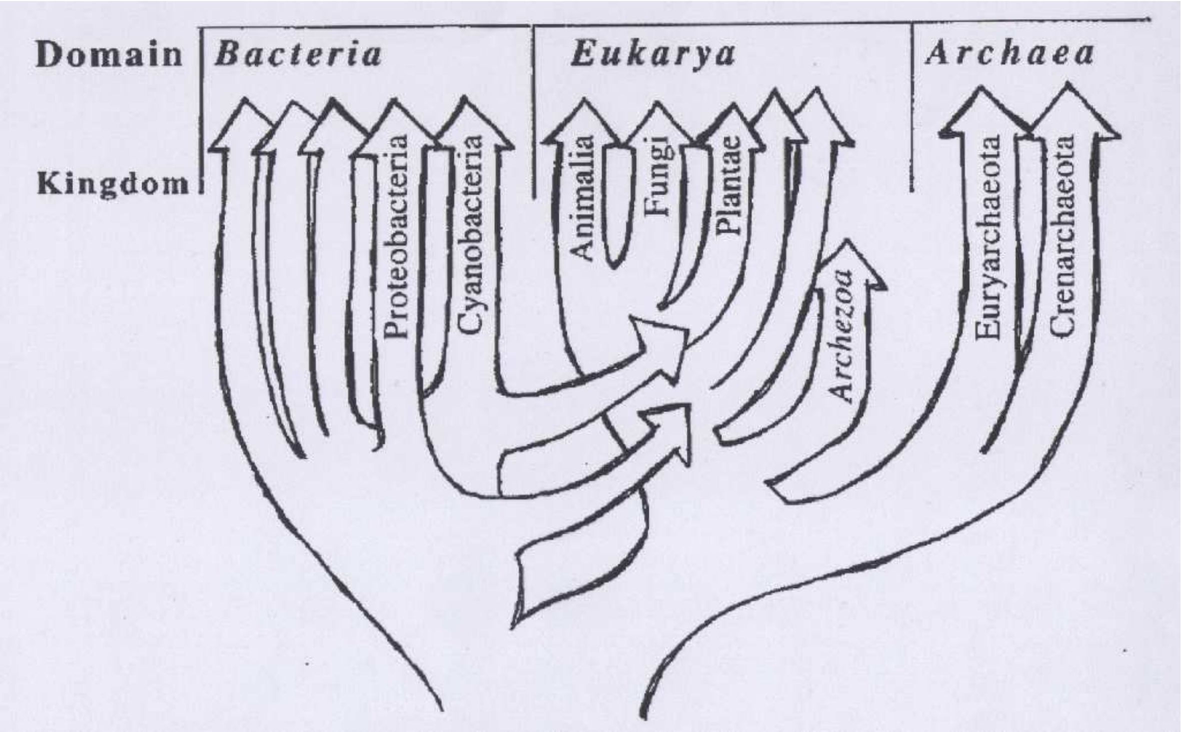
- Webinterface for PHYLIP maintained at Institute Pasteur,
  `http://bioweb.pasteur.fr/seqanal/phylogeny/phylip-uk.html`

- Puzzle (Strimmer, v. Haeseler 1996)
  `http://www.tree-puzzle.de/`

- PAML, Phylogenetic Analysis by Maximum Likelihood,
  `http://abacus.gene.ucl.ac.uk/software/paml.html`

Phylogeny, the tree of life

Essential molecular mechanisms like replication and gene expression are similar among all organisms. A *phylogenetic tree model* captures the assumption that present day organisms have evolved from common ancestors. The evolutionary relationships are called *phylogeny*.



(figure taken from Doolittle, Science 284, 1999)

Molecular Phylogenetics

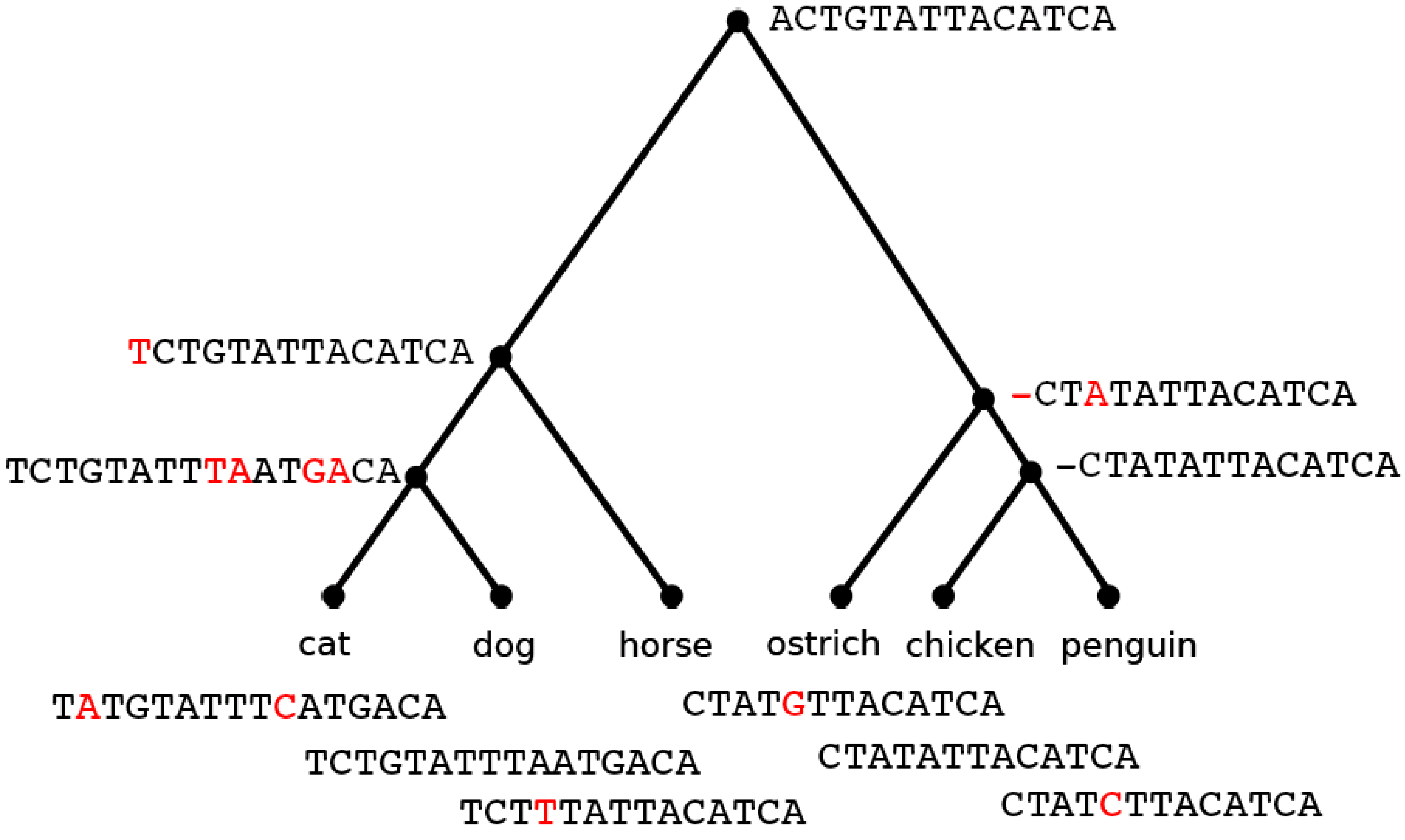
Pioneers in the field of molecular phylogenetics were Zuckerkandl and Pauling. They observed that the number of amino acid exchanges between hemoglobins of two species is approximately proportional to the divergence time of the species.

E. Zuckerkandl and L. Pauling (1962), Molecular disease, evolution and genetic heterogeneity, In *Horizons in Biochemistry*, ed. M. Marsha and B. Pullman, Academic Press, pp. 189–225.

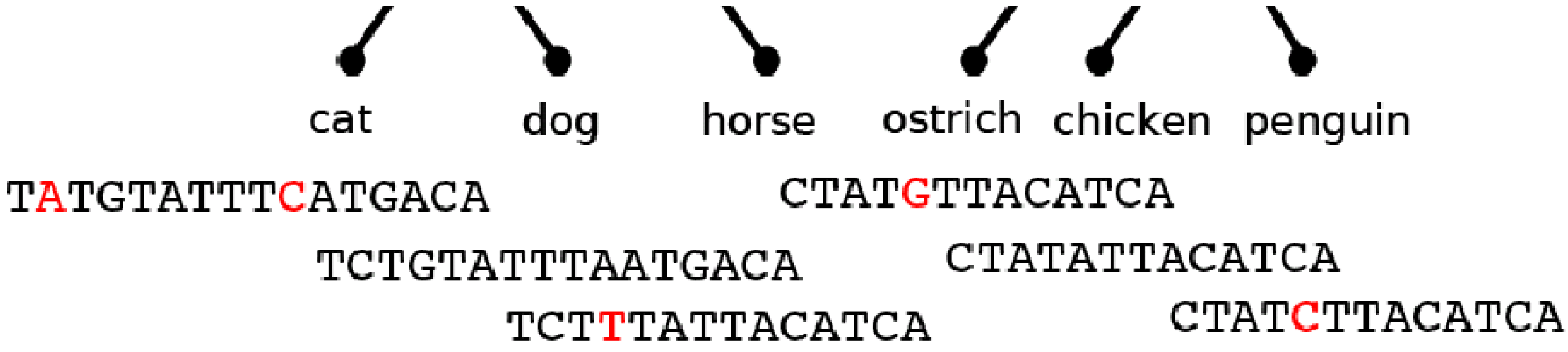
Human	STPDAVMGNPKVKAHGKKVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFRL
Gorilla	STPDAVMGNPKVKAHGKKVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFKL
Horse	SNPGAVMGNPKVKAHGKKVLHSFGEGVHHLDNLKGTFAALSELHCDKLHVDPENFRL
Pig	SNADAVMGNPKVKAHGKKVLQSFSDGLKHLDNLKGTFAKLSELHCDQLHVDPENFRL
Cow	STADAVMNNPKVKAHGKKVLDSFSNGMKHLDDLKGTFAALSELHCDKLHVDPENFKL
Deer	SSAGAVMNNPKVKAHGKRVLDAFTQGLKHLDDLKGAFAQLSGLHCNKLHVNPQNFRL
Gull	SSPTAINGNPMVRAHGKKVLTSFGEAVKNLDNIKNTFAQLSELHCDKLHVDPENFRL

(A window of an alignment of beta hemoglobin genes)

Molecular Phylogenetics, cont'd



Molecular Phylogenetics, cont'd



cat	T	A	T	G	T	A	T	T	T	C	A	T	G	A	C	A
dog	T	C	T	G	T	A	T	T	T	A	A	T	G	A	C	A
horse	T	C	T	T	T	A	T	T	A	C	A	T	–	–	C	A
ostrich	–	C	T	A	T	G	T	T	A	C	A	T	–	–	C	A
chicken	–	C	T	A	T	A	T	T	A	C	A	T	–	–	C	A
penguin	–	C	T	A	T	C	T	T	A	C	A	T	–	–	C	A

Nucleotides of one alignment column are *homologous*: They have evolved from nucleotides in ancestral species.

Molecular Phylogenetics, cont'd

Advantages of molecular sequences over morphological characters for molecular phylogenetics:

- DNA and amino acid sequences are strictly heritable units
- Unambigious description of molecular characters and character states
- Amenability to mathematical modeling and quantitative analysis
- Homology assessment is easy (?)
- Distant evolutionary relationships may be revealed
- huge amounts of data available

Molecular Phylogenetics, cont'd

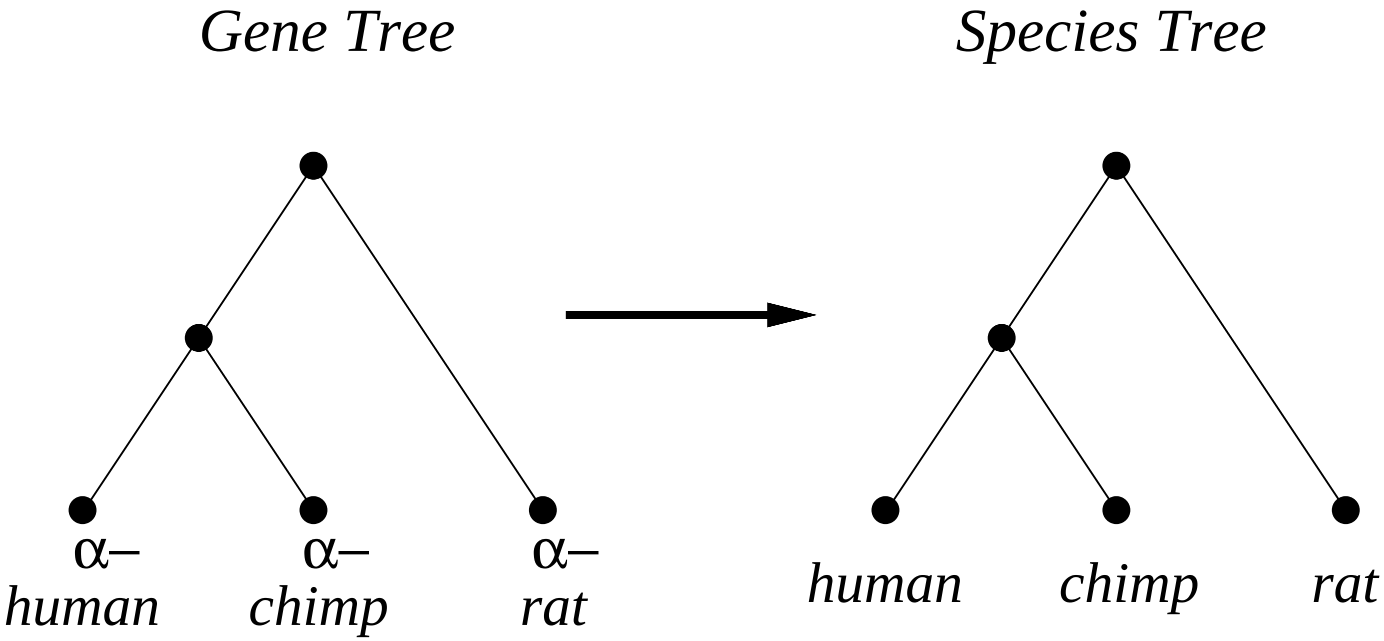
Reconstructed molecular phylogenies are used to

- gain insights into (molecular) evolution
- predict gene functions
- predict that gene functions diversify
- detect various regimes of selective pressures (pharmacology)
- epidemiology
- ...
- ...

Gene Tree and Species Tree

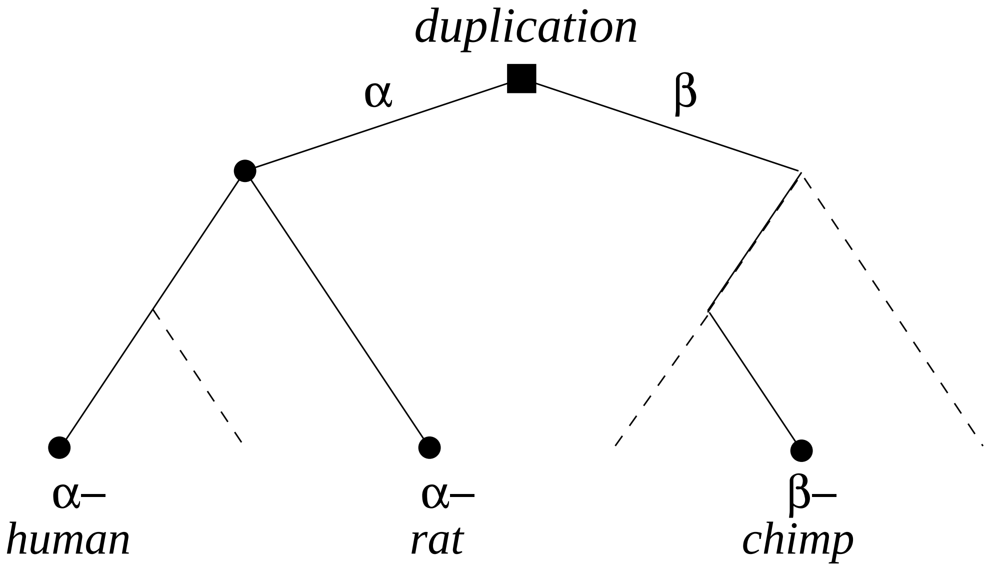
The traditional objective of a phylogenetic tree is to represent the evolutionary relationship between species. In molecular phylogenetics, usually an alignment of homologous genes is put into the tree reconstruction. The phylogeny of the species can be transferred from the gene tree, if the genes are *orthologous*.

Consider the evolution of alpha–hemoglobins in human, chimp and rat:



Gene Tree and Species Tree, cont'd

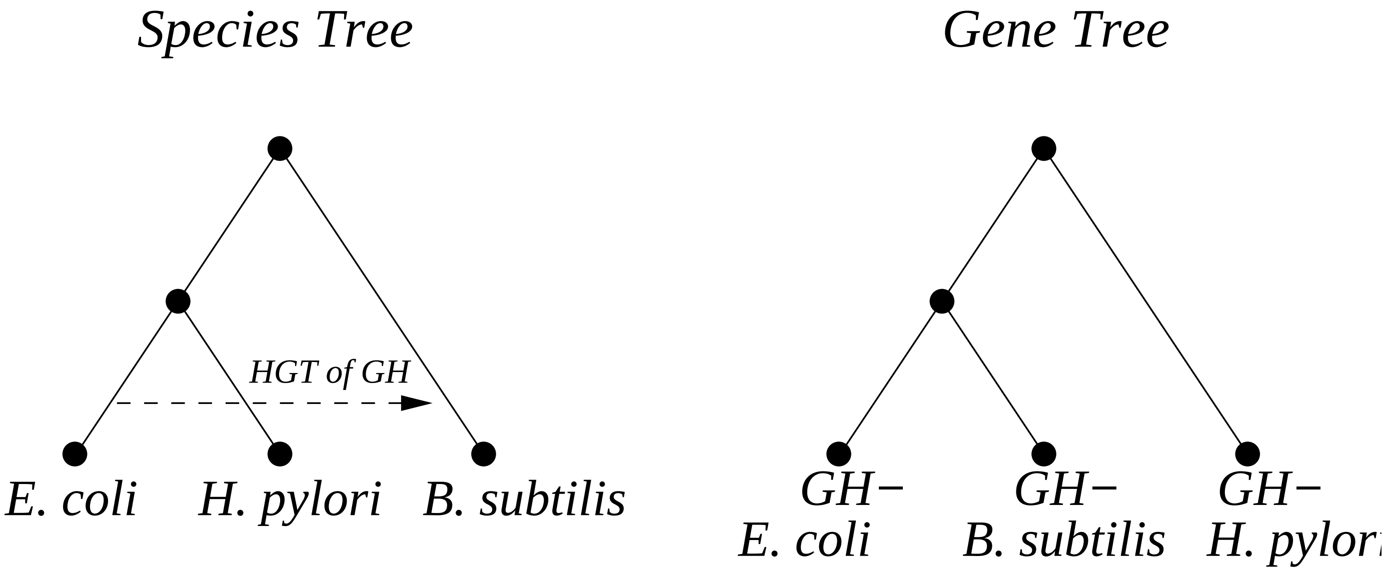
- *Homologous genes* have evolved from a common ancestor
- *Orthologous genes* have evolved from a common ancestor by a speciation event (the last common ancestor (LCA) of orthologous genes represents a speciation event).
- *Paralogous genes* have evolved from a common ancestor by a duplication event (the LCA represents a duplication event).



β–chimp is paralogous to α–rat (and to α–human) since the least common ancestor of the two genes corresponds to a duplication event.

Gene Tree and Species Tree, cont'd

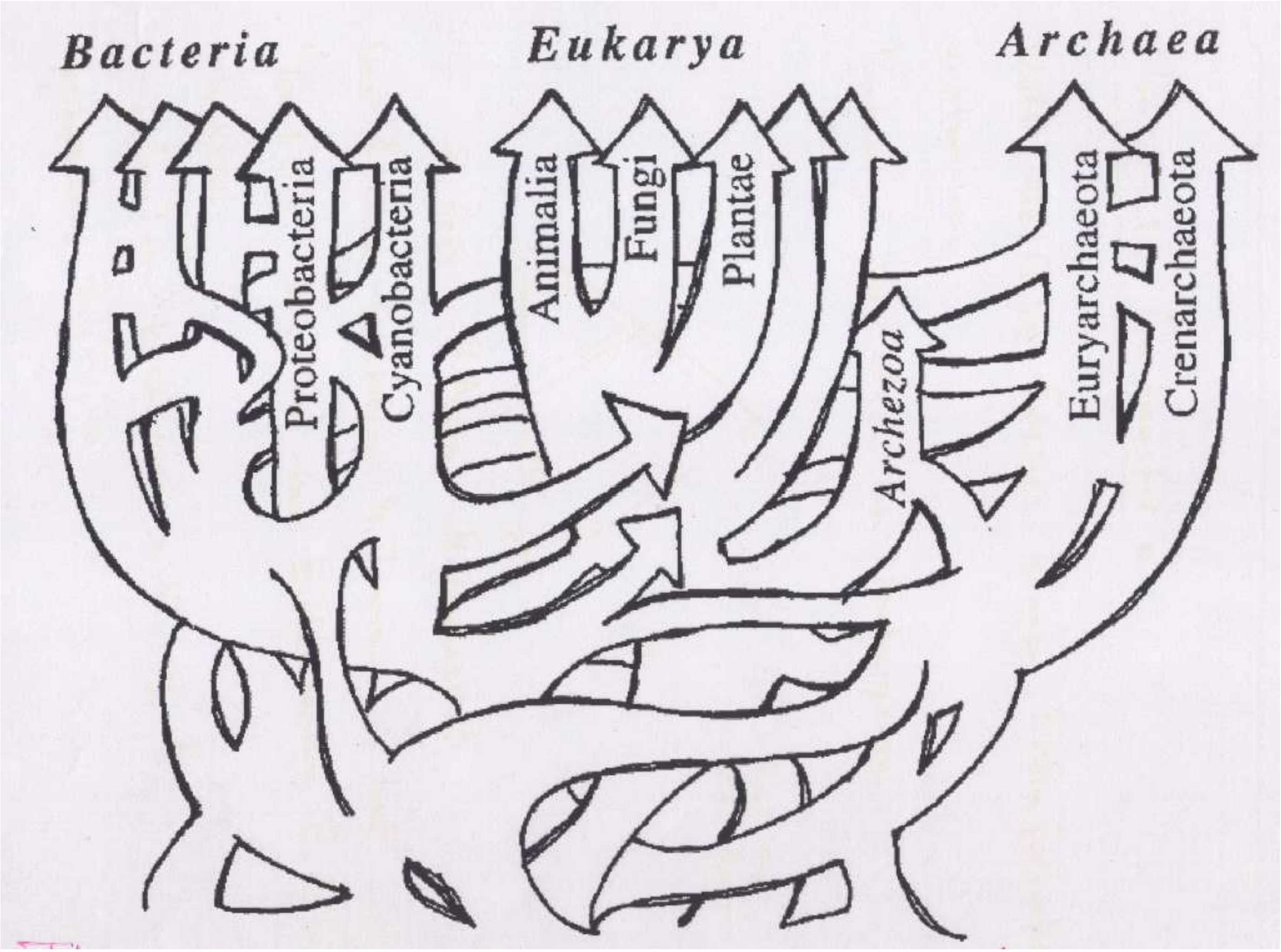
Species may exchange hereditary information. This mainly occurs in Prokaryotes and is called *Horizontal Gene Transfer (HGT)* . Consider that a B. subtilis strain recently obtained the gene encoding Glyclosyl Hydrolase (GH) from an E. coli strain.



The gene tree and the species tree are incongruent and it is not possible to infer the species phylogeny based on the gene tree for Glycosyl Hydrolase.

Gene Tree and Species Tree, cont'd

There is no unique universal phylogenetic tree.

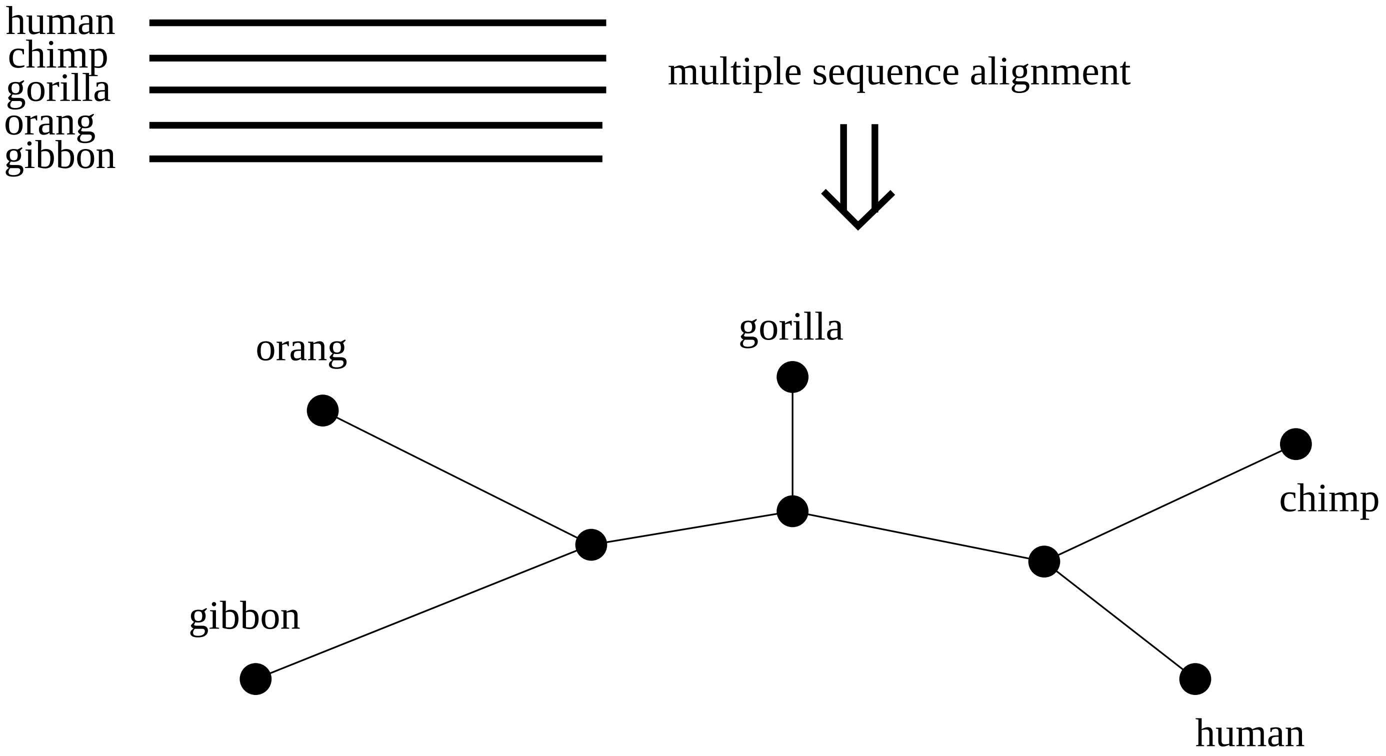


(figure taken from Doolittle, Science 284, 1999)

Phylogenetic tree reconstruction:

We are given a multiple alignment of homologous molecular sequences.

Find a leaf labeled binary tree that explains the data (best).



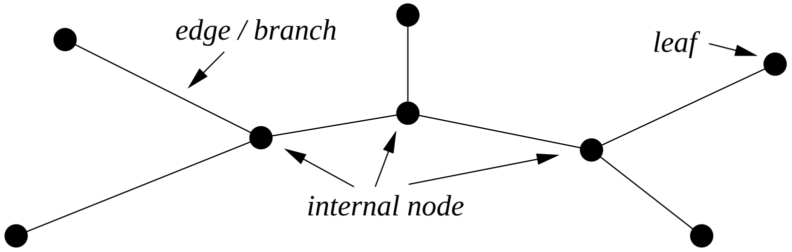
Binary trees

- A *graph* is a pair $G = (V, E)$ consisting of a set of *nodes* (or *vertices*) and a set $E \subseteq V \times V$ of *edges* (or *branches*) that connect nodes. The *degree* of a node $v \in V$ is the number of edges incident to v.

- A *path* is a sequence of nodes $v_1, v_2, \ldots, v_n$ where v_i and v_{i+1} are connected by an edge for all $i = 1, \ldots, n-1$.

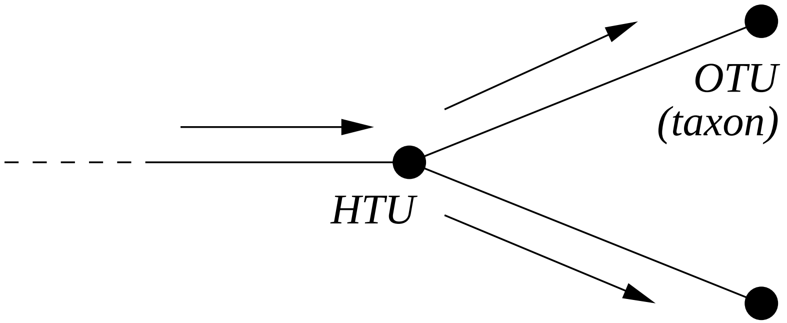
- A *cycle* is a simple path in which the first and last vertex are the same. A graph without cycles is called *acyclic*.

- A *tree* is a connected acyclic graph. Any two nodes of the tree are connected by a unique simple path. A *binary tree* is a tree where the nodes have degree 3 (*iternal nodes*) or degree 1 (*leaves*).



Bifurcations

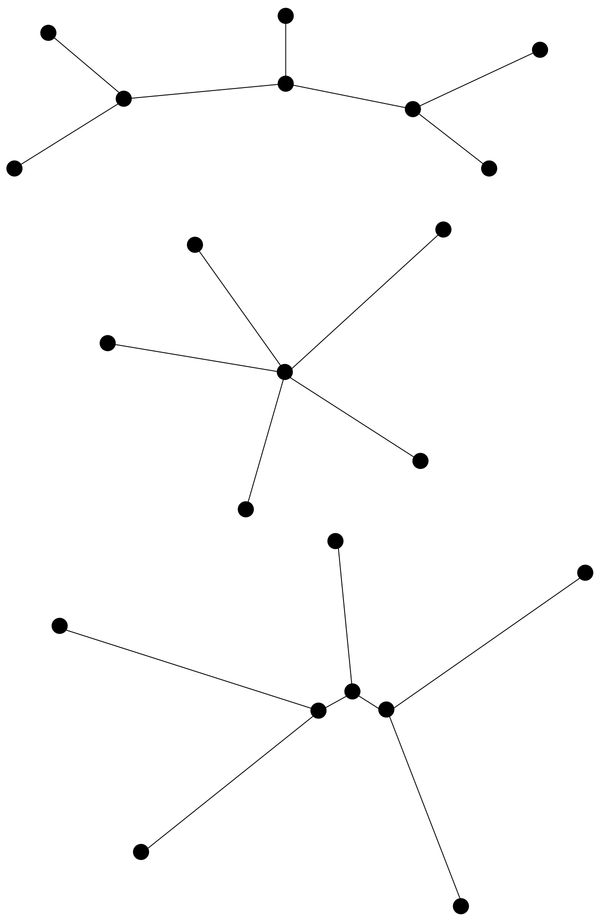
- Species evolve in time. In the simplified tree model, we assume that a species evolves along an edge. Internal nodes reflect ancestral species and split into two new species. This is reflected by *bifurcations* in the binary tree.



- Internal nodes correspond to hypothetical ancestors. In phylogeny, they are referred to as *HTUs* (hypothetical taxonomic units). Leaves are called **taxa** or *OTUs* (operational taxonomic units). Phylogeny is reconstructed for a set of taxa, which e.g. are given as genes or proteins.

Binary tree vs. star tree

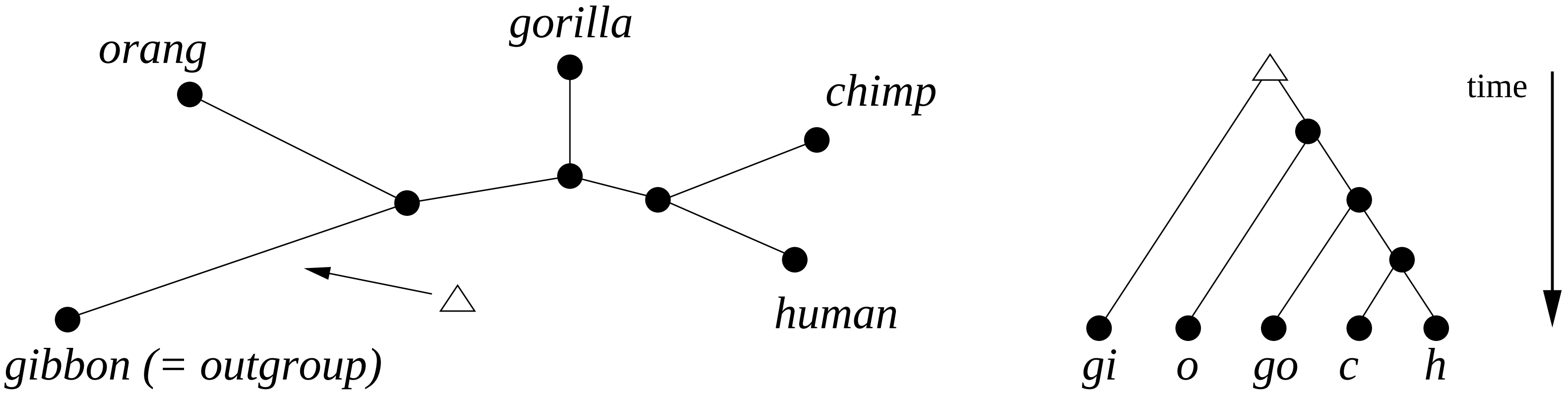
- Binary trees are said to be *fully resolved*. They do not exhibit multifurcations.
- A star tree only has one internal node with a multifurcation (unresolved node, *polytomy*). It is not resolved at all and does not provide information about phylogenetic relationships.
- Reconstruction of phylogenies on data with a weak phylogenetic signal sometimes yields fully resolved trees which look starlike.



Where is the root?

Almost all phylogenetic tree reconstruction methods reconstruct an unrooted binary tree which cannot be interpreted with respect to a time scale. In an unrooted tree, one does not know whether an internal node is the ancestor or the descendant of its neighboring internal node.

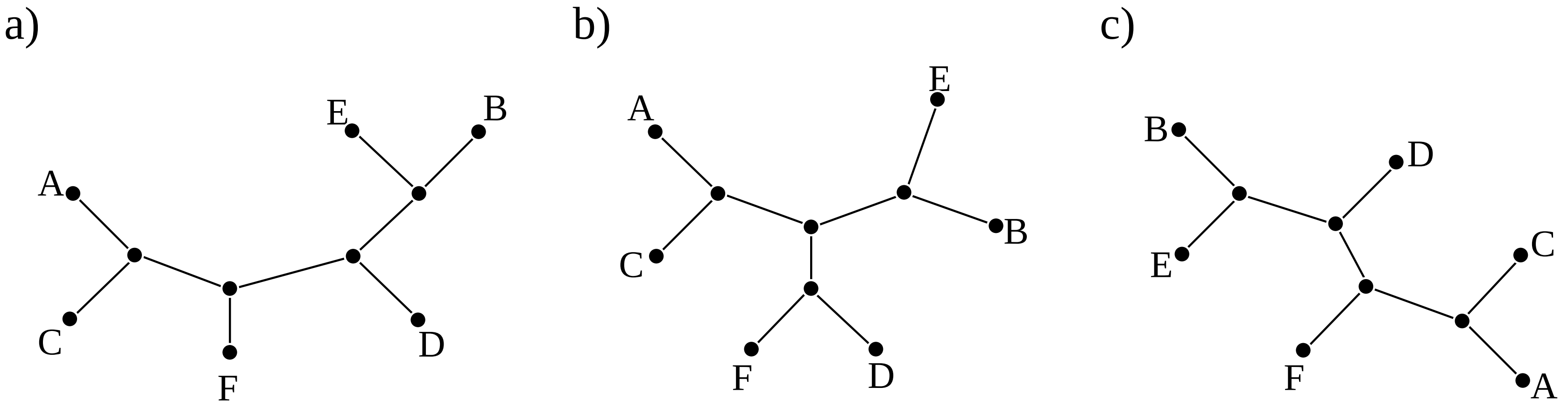
Sometimes it is possible to obtain external information that a certain taxon is more distantly related to the other taxa than the other ones among themselves. Such a taxon is called *outgroup*. Adding a root node to the edge to the outgroup then allows interpreting bifurcations with respect to time.



The inclusion of an outgroup that is too distantly related may lead to incorrect tree reconstructions.

Topology and splits

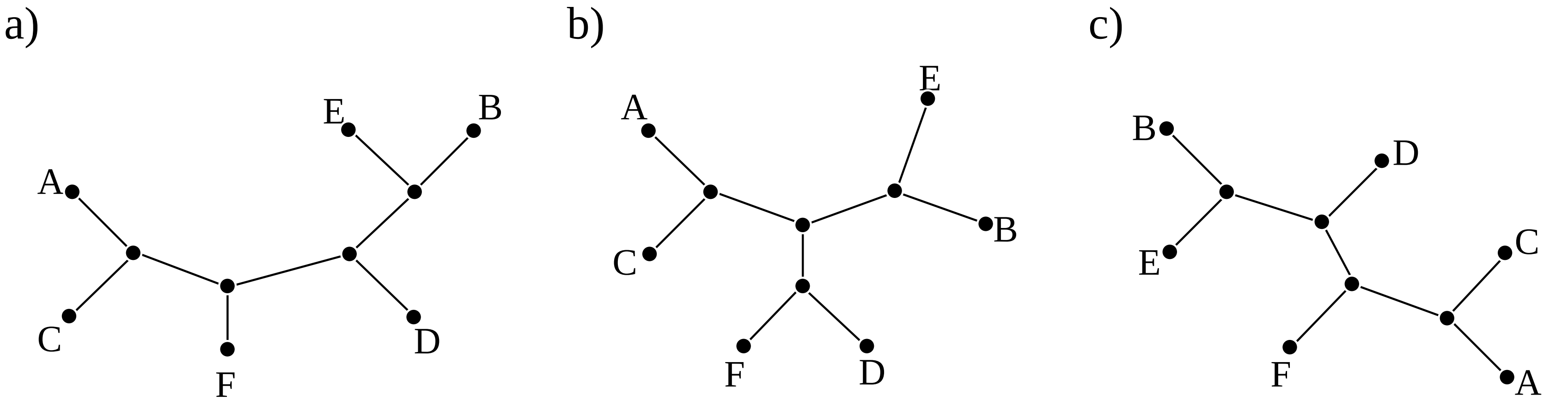
- Two trees showing the same branching pattern are said to have the same *tree topology*.



- The trees shown in a) and c) have the same topology whereas the topology of the tree in b) is different.

Basic notions, topology and splits, cont'd

- A *split* (*bipartition*) at an edge partitions the set of taxa into two disjoint sets.

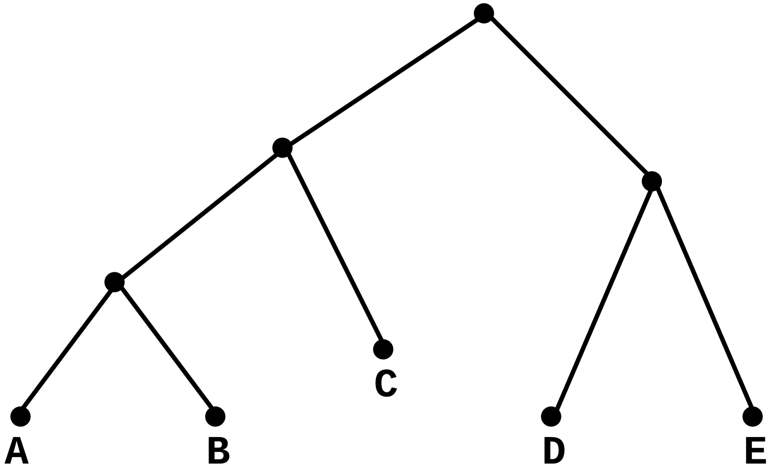


- A split at an edge is phylogenetically informative, if the edge is not connected to a leaf. For the tree in b) the splits $(AC \| FDEB), (FD \| ACEB), (EB \| ACFD)$ are phylogenetically informative.

- The topology of a binary tree is given by its set of phylogenetically informative splits.

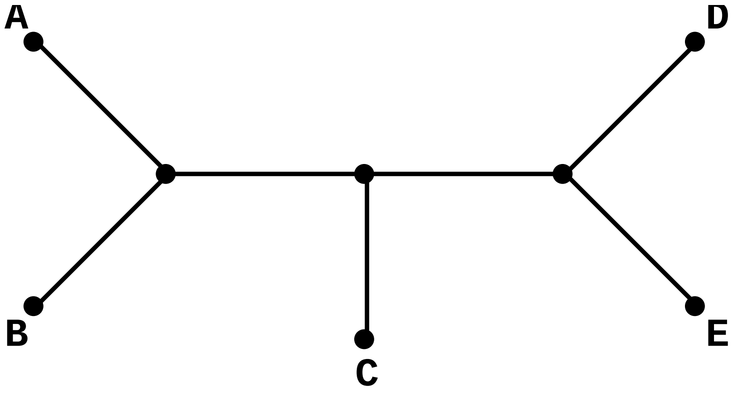
Newick format

Electronically, trees are usually held in a readable text file in the *Newick format*.



```
(((A,B),C),(D,E))
```

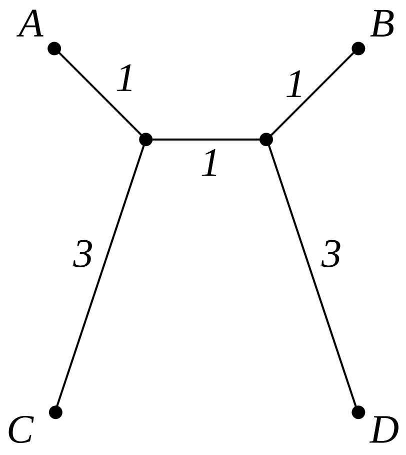
The root is represented by the outmost parenthesis. There are many ways to represent unrooted trees.



```
(((A,B),C),(D,E))
((A,B),(C,(D,E)))
(A,(B,(C,(D,E))))
((A,B),C,(D,E))
...
```

Weighted trees

Reconstructed phylogenetic trees normally are *weighted trees*. That is, each edge is assigned an edge length. Edge lengths represent mutation events which are supposed to have occured on the evolutionary path.



Differences in edge lenghts in the above tree reflect the fact, that the rates at which mutations accumulate in the sequences vary among the lineages to the taxa.

Methods for phylogeny reconstruction ...

... are classified according to their input data.

1. **Character based methods** take as input a *character state matrix*. Examples for characters are 'number of extremities', 'existence of a backbone, 'nucleotide at a site in a molecular sequence', ...

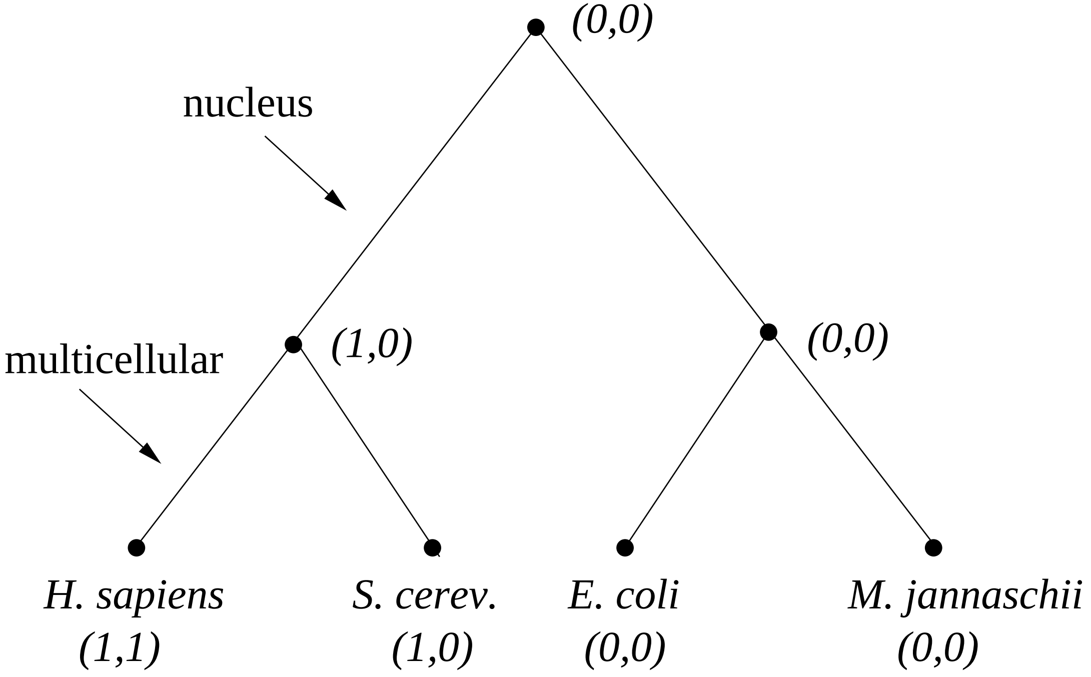
   - Maximum Parsimony
   - Maximum Likelihood

2. **Distance based methods** take as input a *distance matrix*, which is obtained by measuring the dissimilarity or the evolutionary distance between the taxa.

   - UPGMA, clustering
   - Neighbor Joining
   - Least Squares (Fitch–Margoliash)
   - Minimum Evolution

Character state matrix

	nucleus	multicellular
E. coli	0	0
M. jannaschii	0	0
S. cerevisiae	1	0
H. sapiens	1	1



This tree in accordance with *Ockham's razor*: The best hypothesis is the one requiring the smallest number of assumptions.

Character state matrix, cont'd

An alignment is a *character state matrix*. The characters are the sites of the alignment, the character states are the nucleotides a taxa holds at a site.

```
Human    STPDAVMGNPKVKAHGKKVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFRL
Gorilla  STPDAVMGNPKVKAHGKKVLGAFSDGLAHLDNLKGTFATLSELHCDKLHVDPENFKL
Horse    SNPGAVMGNPKVKAHGKKVLHSFGEGVHHLDNLKGTFAALSELHCDKLHVDPENFRL
Pig      SNADAVMGNPKVKAHGKKVLQSFSDGLKHLDNLKGTFAKLSELHCDQLHVDPENFRL
Cow      STADAVMNNPKVKAHGKKVLDSFSNGMKHLDDLKGTFAALSELHCDKLHVDPENFKL
Deer     SSAGAVMNNPKVKAHGKRVLDAFTQGLKHLDDLKGAFAQLSGLHCNKLHVNPQNFRL
Gull     SSPTAINGNPMVRAHGKKVLTSFGEAVKNLDNIKNTFAQLSELHCDKLHVDPENFRL
```

The characters, i.e. the alignment columns, are treated (or modeled) independently of each other.

Maximum Parsimony

According to Ockham's razor ("entia non sunt multiplicanda praeter necessitatem") *Maximum Parsimony* identifies a tree which can be explained by a minimum number of substitution events.

	1	2	3	4
a	G	A	A	T
b	G	G	C	C
c	G	A	C	C
d	T	G	A	T

Consider the above alignment. There are three tree topologies for the four taxa. For each tree topology, we place the sequences of the taxa at its leaves. We are ignorant about sequences at internal nodes (HTUs). But we assign sequences to internal nodes, such that the number of substitutions along the edges which are required to describe the transition from one sequence to another in the tree gets minimal. Among the three topologies, the one(s) which can be explained by the smallest number of substitution events is (are) the *Maximum Parsimony Tree(s)*.

(Obtain Maximum Parsimony Tree for above alignment at blackboard. Which sites are phylogenetically informative ?).

Maximum Parsimony, tree length

The *tree length* l_i of a tree T_i is the minimal number of substitutions which is necessary to explain the tree when assigning sequences to internal nodes. In order to identify the Maximum Parsimony Tree, we applied the following procedure:

```
- For each tree topology $T_i$
    - $l_i \leftarrow 0$
    - For each alignment column $j$
        - Assign nucleotides to internal nodes in $T_i$ such that the number
          of substitutions $s_{ij}$ along the edges is minimal
        - $l_i \leftarrow l_i + s_{ij}$
```

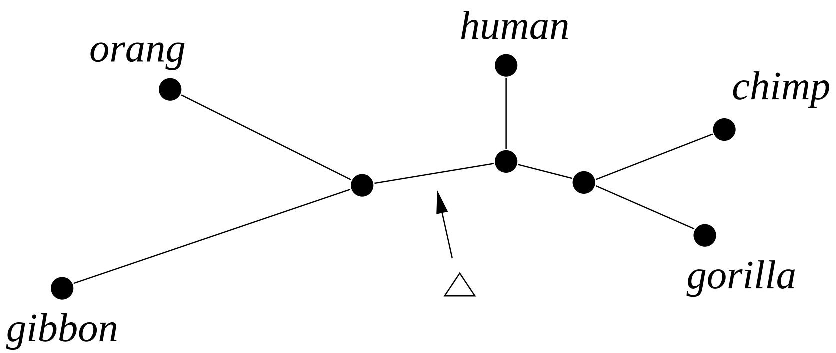
A tree T_i with the smallest tree length l_i is a Maximum Parsimony Tree.

Maximum Parsimony, Fitch algorithm

The first algorithm to compute s_{ij} efficiently was proposed by Fitch (1971).

Given a set of taxa, a tree topology i and an alignment column j

human	A
chimp	A
gorilla	C
orang	C
gibbon	G

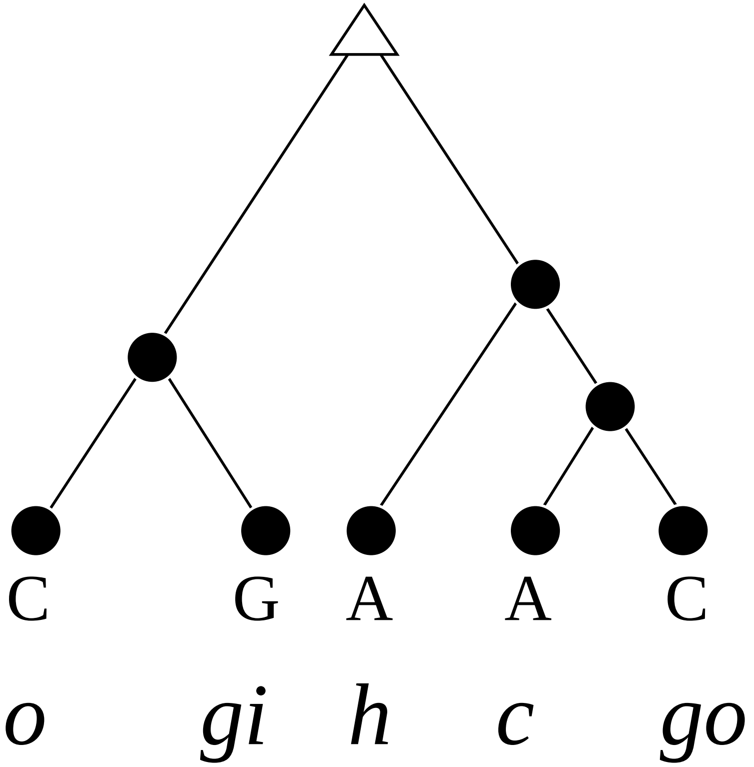


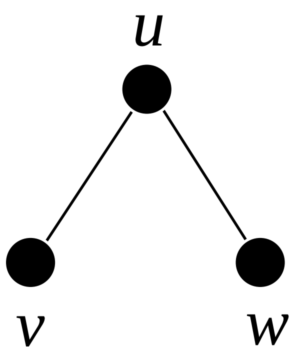
1.) Add a root node to any edge

Maximum Parsimony, Fitch algorithm, cont'd

2.) **bottom-up-pass**
The rooted tree is traversed from the leaves to the root. According to the following rule, sets of nucleotides (character states) are assigned to internal nodes. Say, u is the ancestor of v and w and $\mathcal{U}, \mathcal{V}, \mathcal{W}$ are the respective sets of nucleotides, then set

$$\mathcal{U} = \begin{cases} \mathcal{V} \cup \mathcal{W}, & \text{if } \mathcal{V} \cap \mathcal{W} = \emptyset \\ \mathcal{V} \cap \mathcal{W}, & \text{else} \end{cases}$$





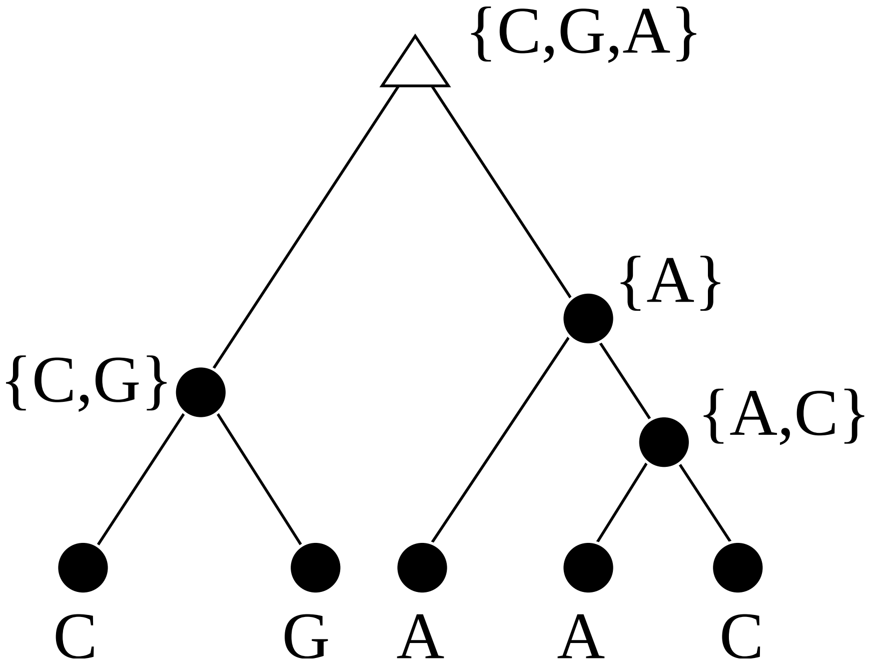
Maximum Parsimony, Fitch algorithm, cont'd

3.) **top-down-pass**

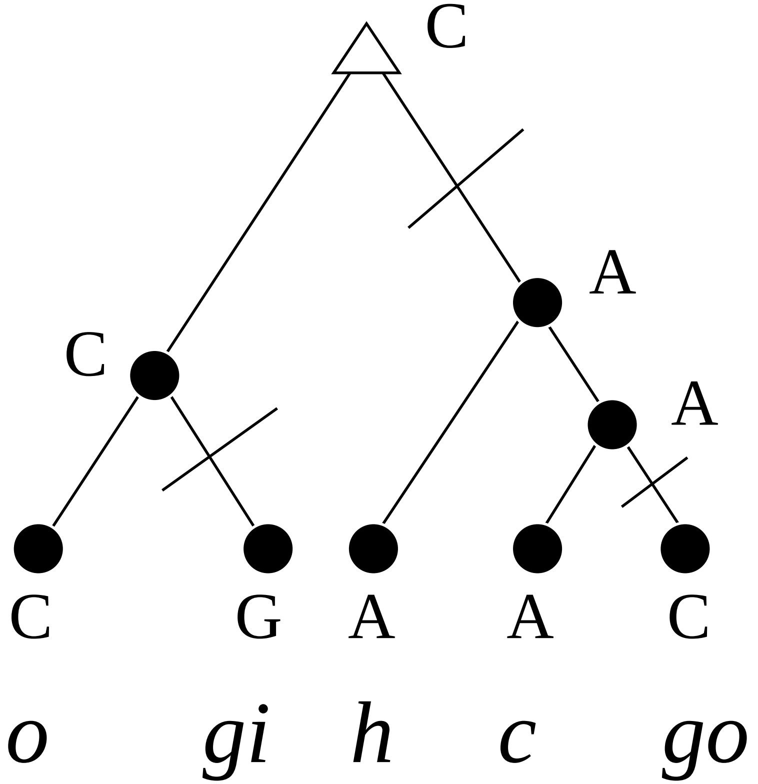
The tree is traversed from the root node to the leaves and the internal nodes are assigned nucleotides according to the following rules

- Assign the root node any nucleotide x out of its set of states $\mathcal{U}_{root}$.
- Assign the child v of parent u the nucleotide

$$\begin{cases} x, & \text{if } x \in \mathcal{U} \\ \text{any nucleotide}, & \text{else} \end{cases}$$



Maximum Parsimony, Fitch algorithm, cont'd



For the given topology i and the alignment column j, the number of substitutions in the tree is $s_{ij} = 3$. The time complexity of the Fitch algorithm is $O(n)$ where n is the number of taxa.

(Is there a tree topology with fewer substitutions for this column? Consider the taxas' set of character states.)

Maximum Parsimony, Sankoff algorithm

Sankoff (1975) suggests a Dynamic Programming algorithm to compute s_{ij}. The Sankoff algorithm is more general than the Fitch algorithm. For example, it allows to score different changes differently. Further, it is possible to apply Sankoff's algorithm to trees with multifurcations (polytomies) at internal nodes.

The algorithm traverses the tree bottom-up from the leaves to the root in a way such that when a node is processed, all its children have already been processed. Each node is assigned to a map with all possible character states λ_i as keys and the tree length l_i of the subtree rooted at this node when assigning it to λ_i, as entries. A leaf's map contains the leafs character state as the only key with entry 0.

Maximum Parsimony, Sankoff algorithm, cont’d

Initialisation: Root the tree T at any internal edge. Label each leaf of T with the respective character state and set $l\,(\text{leaf}) = 0$.

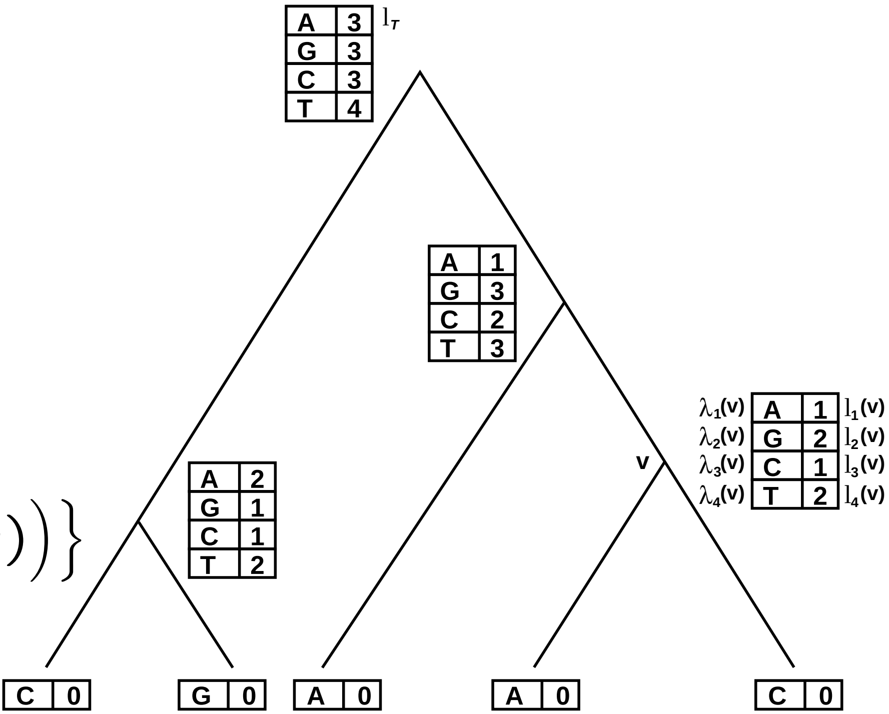
Recursion:

Let v be an internal node in T. Let $\lambda_i\,(v)$ be the $i-th$ state of node v and $l_i\,(v)$ be the length of the subtree rooted at v when assigning λ_i to v.

foreach $\lambda_i\,(v)$, **do**

$$l_i\,(v) = \sum\nolimits_{(w \text{ child of } v)} \min_j \left\{ l_j\,(w) + c\left(\lambda_j\,(w)\,, \lambda_i\,(v)\right) \right\}$$

where $c\left(\lambda_j\,(w)\,, \lambda_i\,(v)\right)$ is the cost function for transitions.

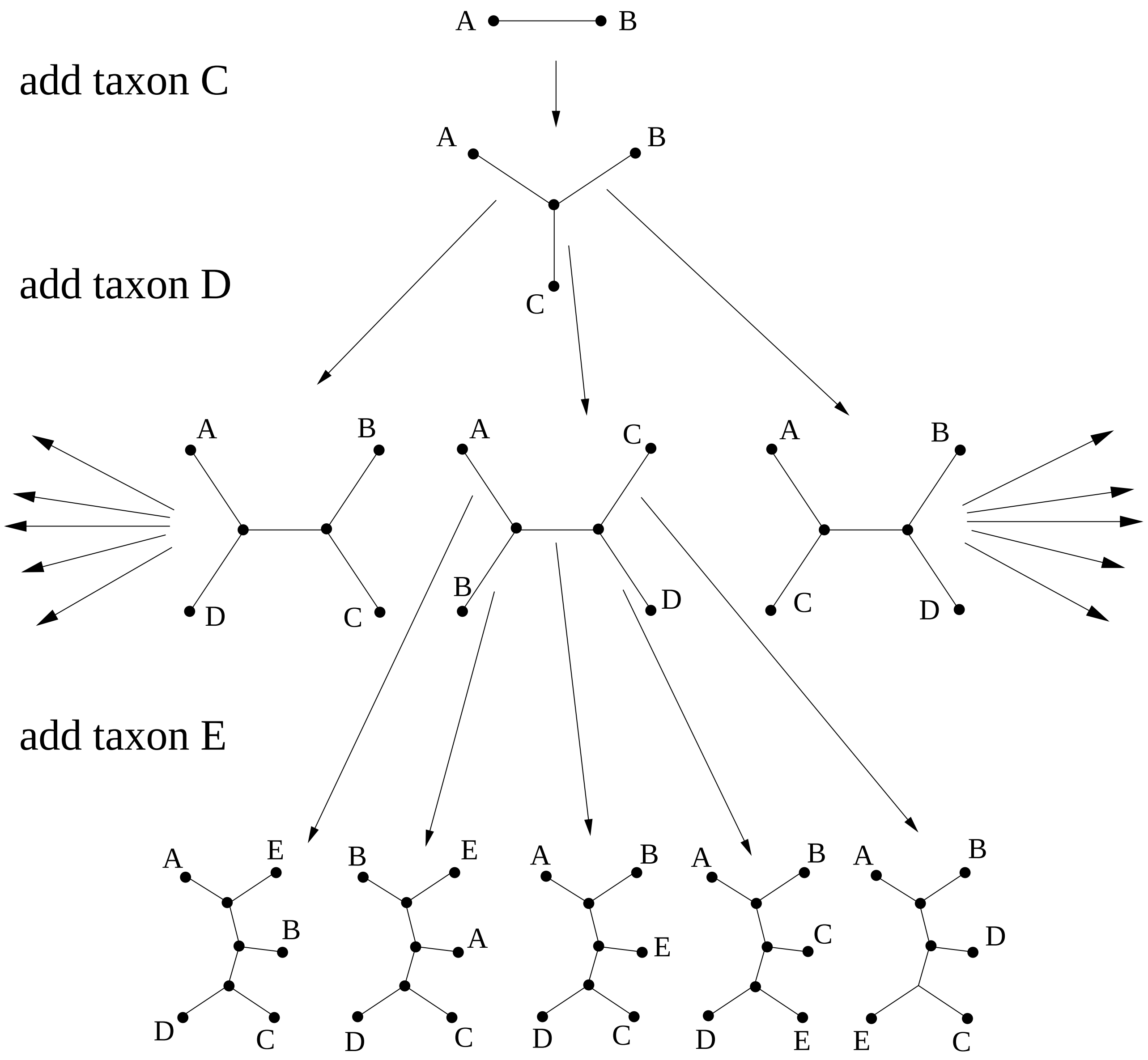

A 3
G 3
C 3
T 4
l_T
A 1
G 3
C 2
T 3
A 2
G 1
C 1
T 2
v
λ_1(v) A 1 l_1(v)
λ_2(v) G 2 l_2(v)
λ_3(v) C 1 l_3(v)
λ_4(v) T 2 l_4(v)
C 0
G 0
A 0
A 0
C 0

The minimal entry in the roots map then is the parsimony tree length l_T.

The number of leaf labeled binary trees for n taxa

All possible leaf labeled binary tree topologies for n taxa can be enumerated by the following procedure: We start with a tree containing any two taxa and subsequently add the other taxa to the tree by inserting internal nodes and edges to the taxa.

(Derive formula as exercise. Hint: the number of edges of a binary tree with n leaves equals $2n-3$)



The number of binary trees given n taxa, cont'd

The numbers of different unrooted and rooted binary tree topologies U_n and R_n are

$$U_n = \prod_{i=3}^{n} (2i - 5), \qquad R_n = \prod_{i=3}^{n+1} (2i - 5)$$

where n is the number of taxa.

n	U_n	R_n
2	1	1
3	1	3
4	3	15
5	15	105
6	105	945
7	945	10395
8	10395	135135
9	135135	2027025
10	2027025	34459425

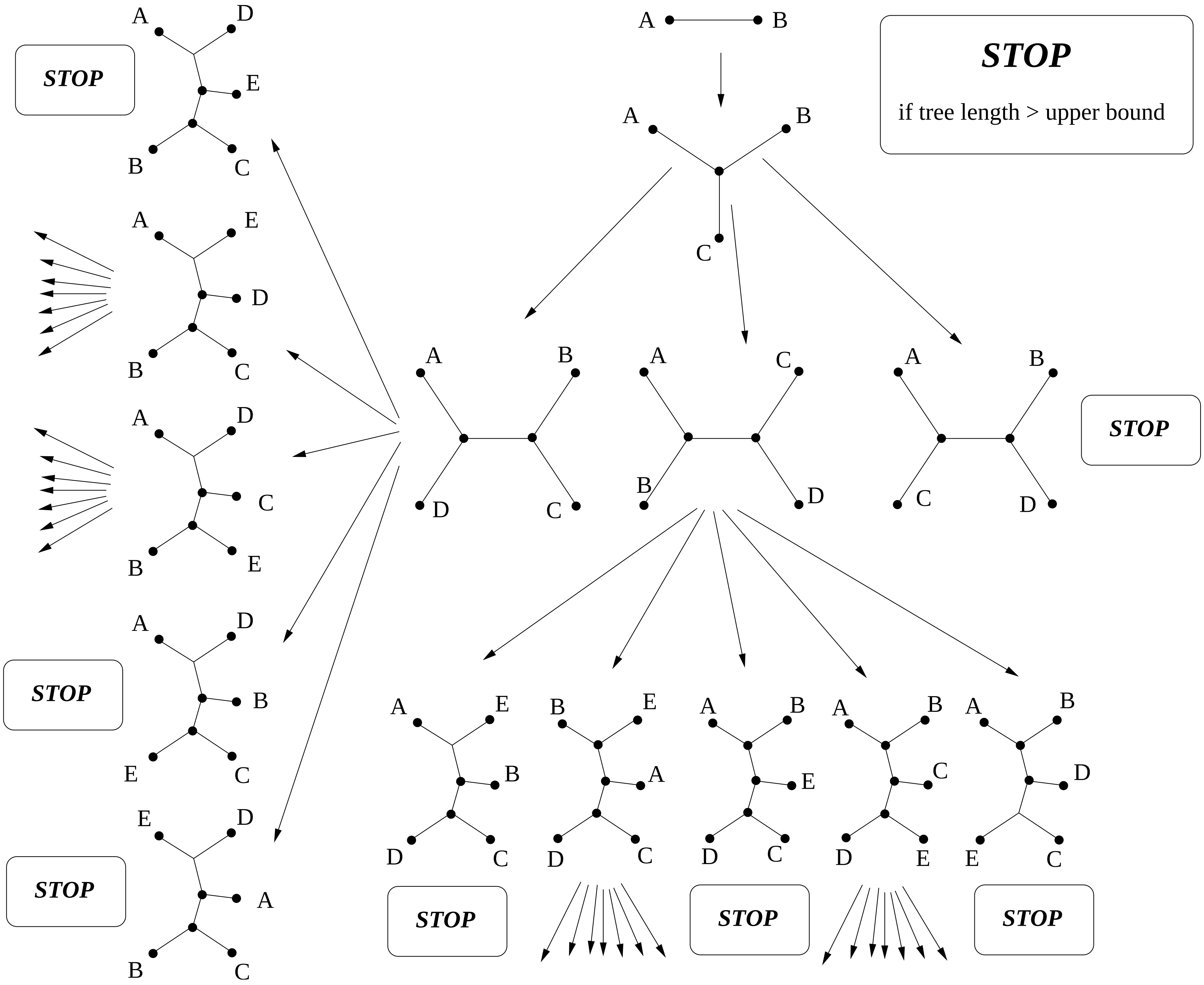
$R_{20} = 8\ 200\ 794\ 532\ 637\ 891\ 559\ 375$

Maximum Parsimony, Branch and Bound

The identification of a Maximum Parsimony Tree requires checking the tree lengths for exponentially many tree topologies. For treelike data, the application of a *branch and bound*–strategy (Hendy and Penny 1982) drastically reduces the tree search space and exact solutions for 20 or more taxa are obtained in manageable time.

Concept:

- Obtain un upper bound for the tree length (e.g. by Neighbor Joining)
- Construct all tree topologies by consecutively adding edges and taxa (see above)
- If the tree length of an intermediate tree is larger than the upper bound, searching the corresponding subtrees is halted.


A
B
A
B
C
STOP
if tree length > upper bound
A
B
D
C
A
C
B
D
A
B
C
D
STOP
A
D
E
B
C
STOP
A
E
D
B
C
A
D
C
B
E
A
D
B
E
C
STOP
E
D
A
B
C
STOP
A
E
B
D
C
STOP
B
E
A
D
C
A
B
E
D
C
STOP
A
B
C
D
E
A
B
D
E
C
STOP

Maximum Parsimony, miscellaneous

- If branch and bound methods are too slow, heuristic searches are used. Usually an initial tree is obtained, e.g. by Neighbor Joining, and this tree is rearranged, for example by *subtree pruning and regrafting*.

- Transitions (exchanges between either two pyrimidines or two purines) occur more often than transversions (a pyrimidine is exchanged by a purin or vice versa). *Weighted parsimony* therefore assigns a larger 'substitution weight' to transversions. Then, the Sankoff algorithm (and not the Fitch algorithm) has to be used to compute the cost of an alignment column under a tree topology.

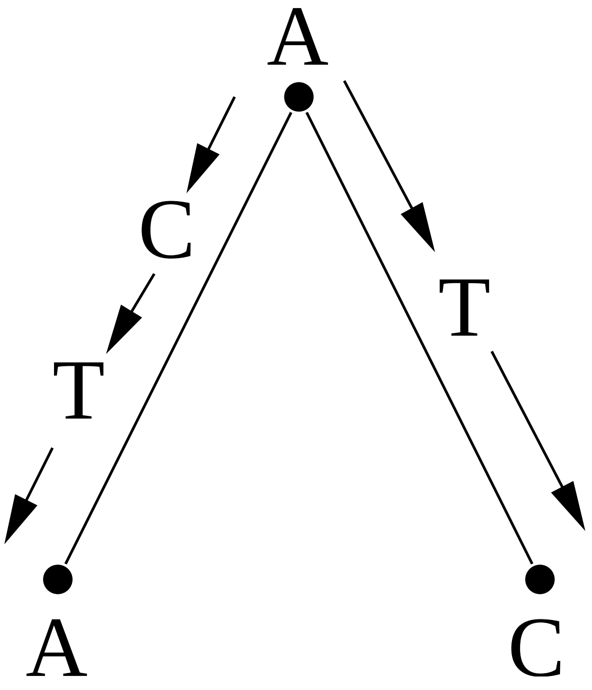
Maximum Parsimony, miscellaneous, cont'd

- A character is called *phylogenetically uninformative* if it does not contribute to resolving relationships among sequences. For example, any conserved alignment column that has the same character state for each taxon is phylogenetically uninformative.

- Sometimes, Maximum Parsimony trees are represented as weighted trees. The weight of an edge then is the cost (the parsimony score) for the transition between the two sequences at the nodes. Note however, that the assignment of sequences to internal nodes is not unique. As a consequence, the most parsimonious tree topology might be represented by different weighted trees.

Maximum Parsimony, miscellaneous, cont'd

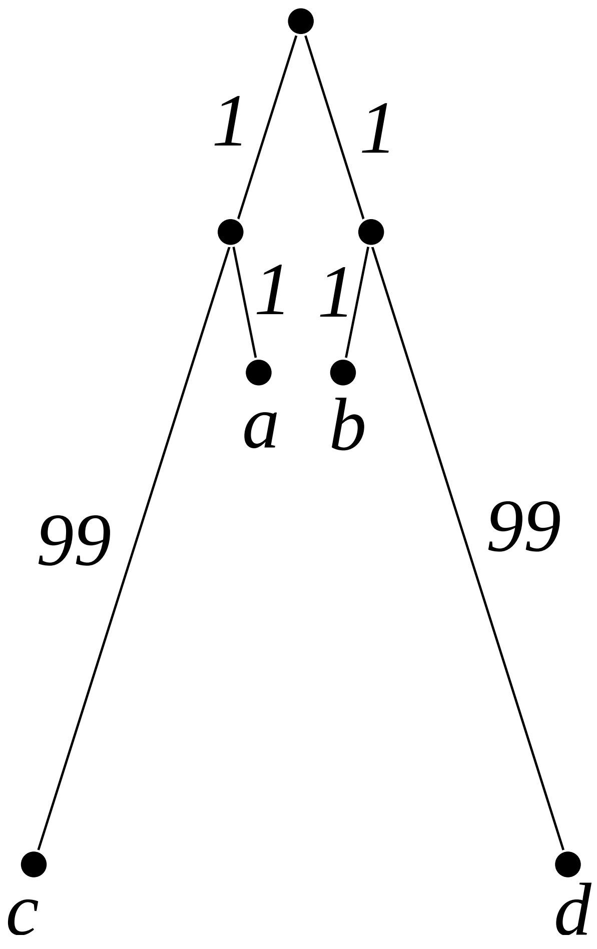
- Maximum Parsimony is widely used. However, Maximum Parsimony does not take into account that the observed character states of taxa being neighbors in the tree may have been multiply substituted. Maximum Parsimony therefore should only be applied to closely related sequences where the probability that multiple substitutions occured is small.



Inconsistency of Maximum Parsimony

An estimation method is consistent, if it approaches the true value of the quantity estimated as more and more amounts of data are available. When reconstructing phylogenies, the estimated quantities are the edge lengths of the tree and the tree topology. Assume, we know that the four sequences representing taxa a, b, c, d have evolved according to the tree shown on the right. The edges to a and b are short whereas the edge lengths to c and d are much larger. In other words, rates of evolution for taxa c and d are relatively high compared to the rates at which taxa a and b evolved.

The 'true' tree



Inconsistency of Maximum Parsimony, cont'd

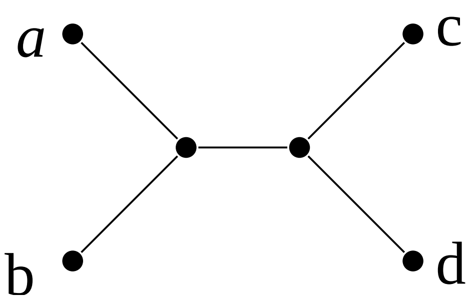
The following sequence family was generated by REFORM (Random Evolutionary FORests MOdel, see `http://www.molgen.mpg.de/~rahmann/`). The root sequence was drawn from the uniform distribution of nucleotides, and the sequences were simulated according to the Jukes–Cantor model (see below) and the tree shown above, where the edge lengths correspond to the expected number of substitutions per 100 sites.

```
         s    s                        s            s            | 4
a  ATAAAGAGAAATGAGGACTACCCCAGACAAAATACTTAGTCATTAGAGGATGCACGAGAG  |60
b  ATAAAGCGAAAGGAGGAGTACCCCAGACAAAATACTCAGTCATTAGAGGCTGCACGAGAG  |60
c  AGCAAGAACTCGTCACCCTGCCACACACACAAAGCTGTATCGACCAACAAATGTCAAGAA  |60
d  ATAATGTGATTGGGGCTGCGGGGCACTGGACATTCTTCGCCCGCAACTCCAGCACGAGCA  |60
   *  * *   * *   *   *   ***     *  **    *  * *    *   * ** *  |21
            i     i   i     i                 i i   ii        i  | 9
s - sites where nucleotides in sequences a and b differ
* - sites with identical nucleotides in c and d
i - phylogenetically informative sites
```

Inconsistency of Maximum Parsimony, cont'd

Sequences a and b are well conserved. Only four susbstitutions have accumulated in the sequences on their evolutionary paths. On the other hand, sequences c and d are very divergent. But even for two random sequences we expect that $\frac{1}{4}$ of the sites have the same nucleotide. In the alignment there are 21 sites with the same nucleotide in c and d. With respect to a Maximum Parsimony reconstruction, these sites become the phylogenetically informative ones in case that the nucleotides between a and b are still conserved and different from the nucleotides in c and d. In the alignment there are 9 informative sites, but only one of them favors the correct topology. The Maximum Parsimony tree therefore has the wrong topology.

Maximum Parsimony tree



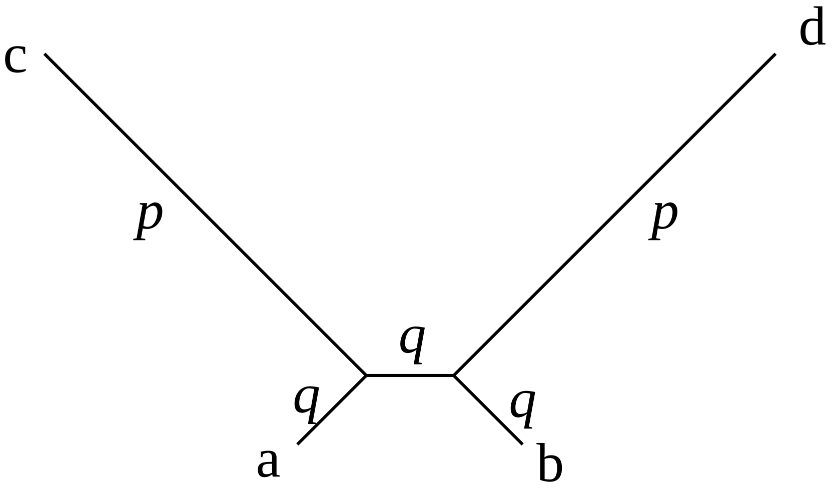
This effect is called *long branch attraction*. A ML estimation finds the correct topology.

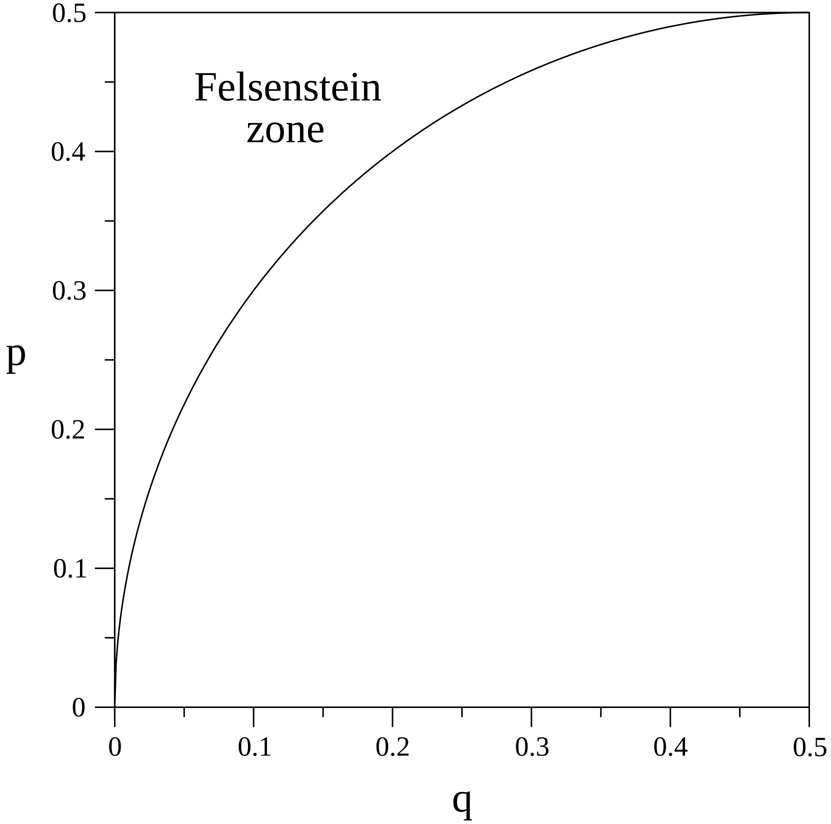
Inconsistency of Maximum Parsimony, cont'd

For a binary alphabet and probabilities of change p and q, Felsenstein (1978) showed that for enough long sequences, Maximum Parsimony will find the wrong tree topology, if

$$q\,(1-q) \leq p^2$$

This area is called *Felsenstein zone*



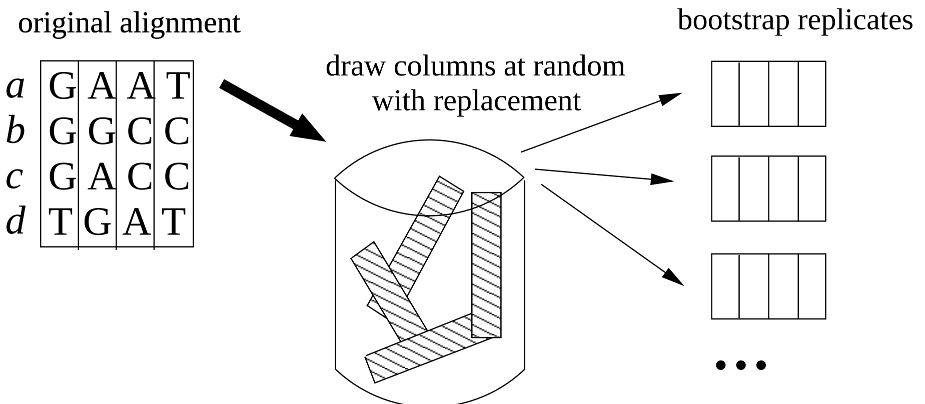


Non–parametric Bootstrapping

Non–parametric bootstrapping is the most commonly used method to obtain a quantity telling us something about the uncertainty of tree reconstructions.” (Felsenstein 1983)

The idea of non–parametric bootstrapping is to disturb the observed data, that is the composition of alignment columns, and to check if the reconstructed trees are similar to the original one (or among each other). The order of the columns is irrelevant for the outcome of the tree estimate.

Procedure: In one bootstrap simulation step, a new alignment or *bootstrap replicate* is generated by randomly drawing columns from the original alignment with replacement. This is repeated until the bootstrap replicate contains as many columns as the original alignment.

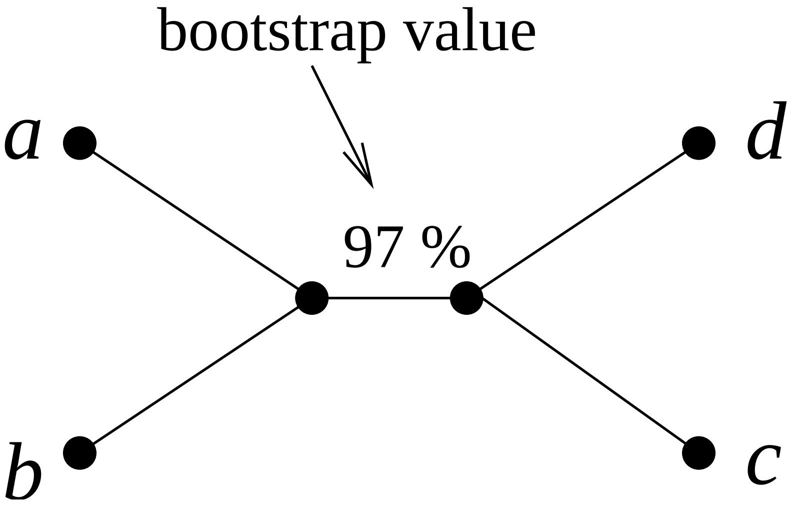


a	G	A	A	T
b	G	G	C	C
c	G	A	C	C
d	T	G	A	T

Non–parametric Bootstrapping, cont'd

In this way, n bootstrap replicates are obtained where typical values for n range from $100 - 1000$. The tree estimation is applied to all bootstrap replicates in turn. We end up with n "bootstrap trees" each coming with a set of splits. *Bootstrap values* (or the *bootstrap support*) correspond to the relative frequency at which a split of the tree (estimated on the original alignment) occured in the bootstrap replicates.

If we apply the non-parametric bootstrap to the above alignment (section Incosistency of MP), the wrong Maximum Parsimony tree is highly supported



Non–parametric Bootstrapping, cont'd

Non-parametric bootstrapping ...

"... is not a test of how accurate your tree is; it only gives information about the stability of the tree topology (the branching order), and it helps assess wether the sequence data is adequate to validate the topology." (Berry and Gascuel, 1996)

And, please note:

Bootstrapping does not provide information about the adequacy of the method !

Summarizing Maximum Parsimony (MP): keywords to remember

- MP is NP-hard.
  Branch and Bound strategies (exact solution) or heuristics to search the tree space have to be applied.

- Fitch algorithm (1971):
  time complexity $O(n)$, n - number of taxa

- Sankoff algorithm (1975), the DP version

- Felsenstein (1978):
  MP is inconsistent (*Long Branch Attraction*)

Trees from distances

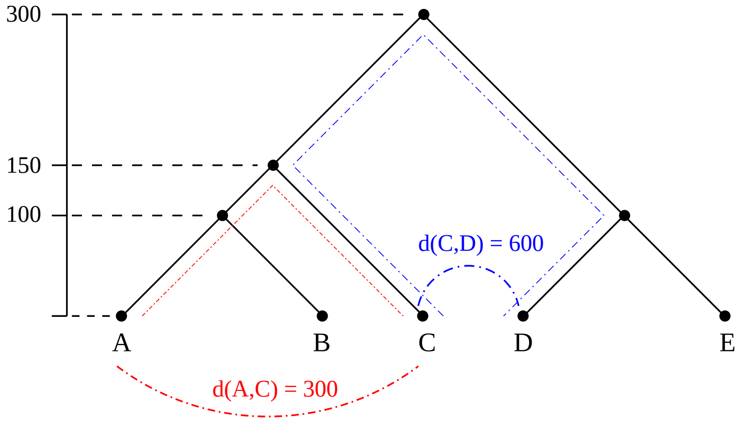
The input for distance based tree reconstruction methods are pairwise distances between taxa. The pairwise distances normally are computed from the multiple alignment.

Consider the set of taxa $\{A, B, C, D, E\}$ and that the measured distances between the taxa are given in the distance matrix d^M.

d^M

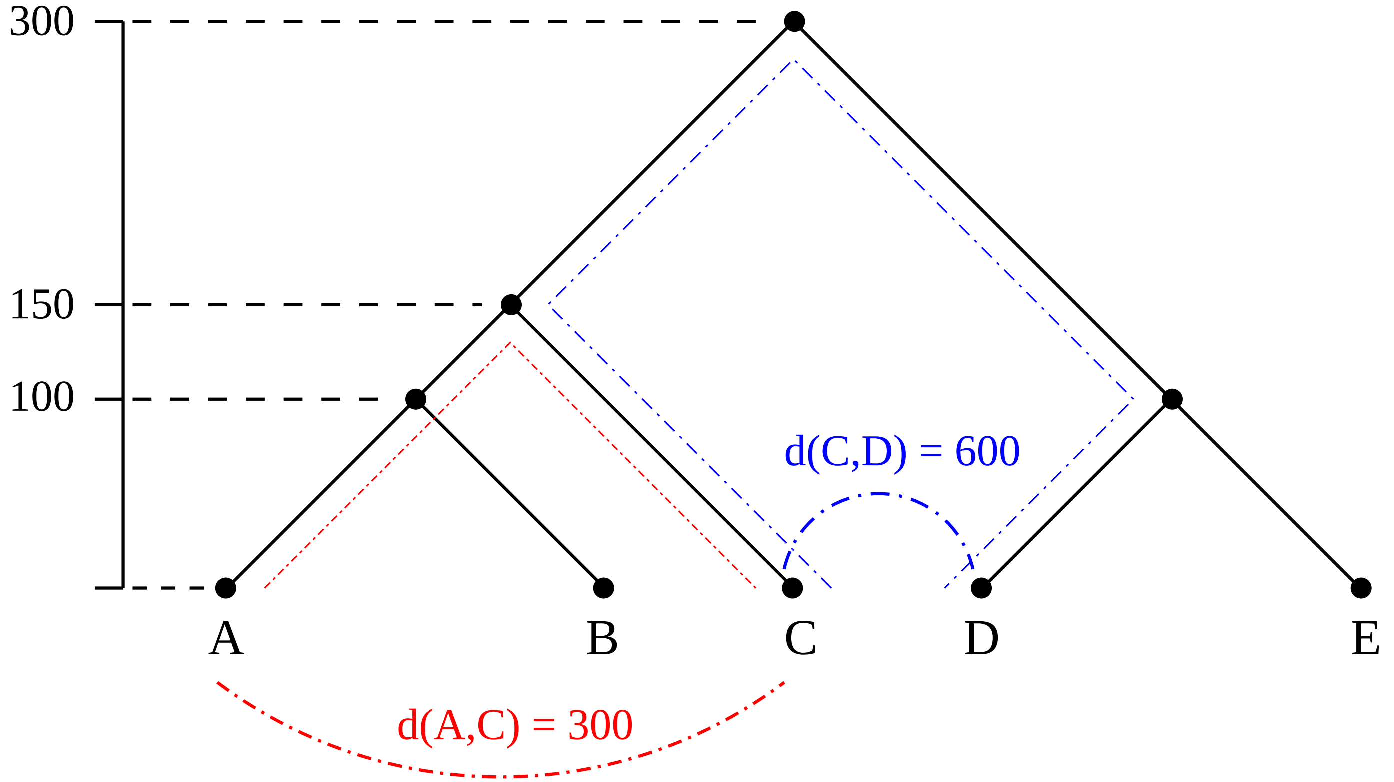
	A	B	C	D	E
A	0	200	300	600	600
B		0	300	600	600
C			0	600	600
D				0	200
E					0

d^T



We want to find a tree T with its *path metric* d^T. T can be reconstructed algorithmically or by fitting d^T to d^M. For the above tree we see that $d^T = d^M$.

Trees from distances, cont'd



The above representation of a phylogenetic tree is called a *dendrogram*. In a dendrogram, we can read off the edge lengths from the vertical axis.

For example, we can check that the path length between leaves A and B is $d^T(A, B) = 100 + 100 = 200$. All path lengths between two leaves form the so called *path metric* d^T on the set of leaves in the tree.

Metric

Definition: A *metric* on a set of objects O is given by an assignment of a real number d_{ij} (a distance) to each pair $i, j \in O$, where d_{ij} fulfills the following requirements:

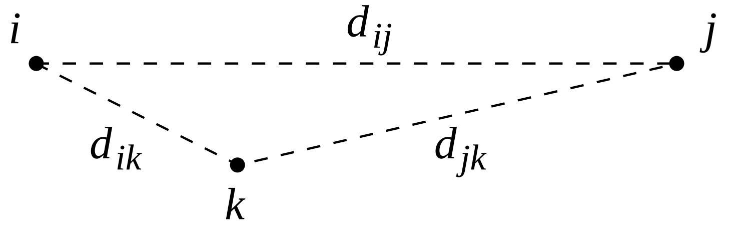
(i) $d_{ij} > 0 \quad for\ i \neq j$

(ii) $d_{ij} = 0 \quad for\ i = j$

(iii) $d_{ij} = d_{ji} \quad \forall\ i, j \in O$

(iv) $d_{ij} \leq d_{ik} + d_{kj} \quad \forall\ i, j, k \in O$

The latter requirement is called the *triangle inequality*

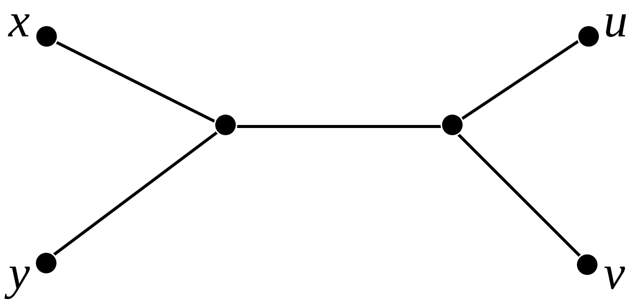


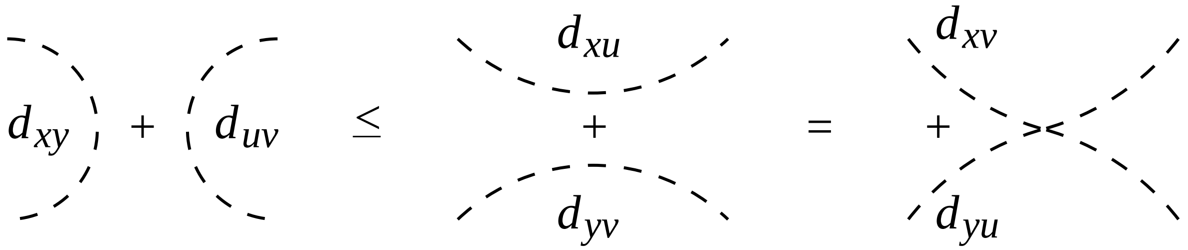
Additive metric

Let d be a metric on O. d is an *additive metric* if it satisfies the *four point condition* (Bunemann 1971).

Four point condition: d is an *additive metric* on O, if the elements of every four-element-subset of O can be labeled by x, y, u and v such that

$$d_{xy} + d_{uv} \;\leq\; d_{xu} + d_{yv} \;=\; d_{xv} + d_{yu}.$$





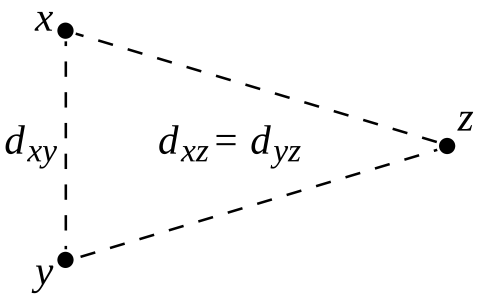
The four point condition is a strengthened version of the triangle inequality. It implies that the path metric of a tree is an additive metric.

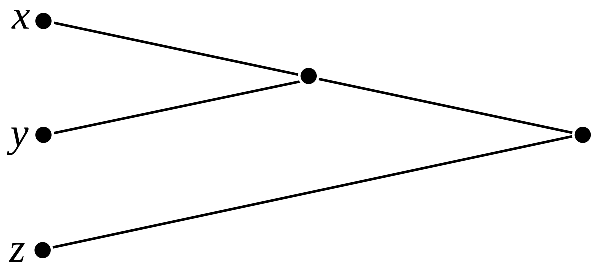
Ultrametric

d is an *ultrametric* if it satisfies the *three point condition*.

Three point condition: d is an *ultrametric* on O, if the elements of every three-element-subset of O can be labeled by x, y, z such that

$$d_{xy} \leq d_{xz} = d_{yz}.$$





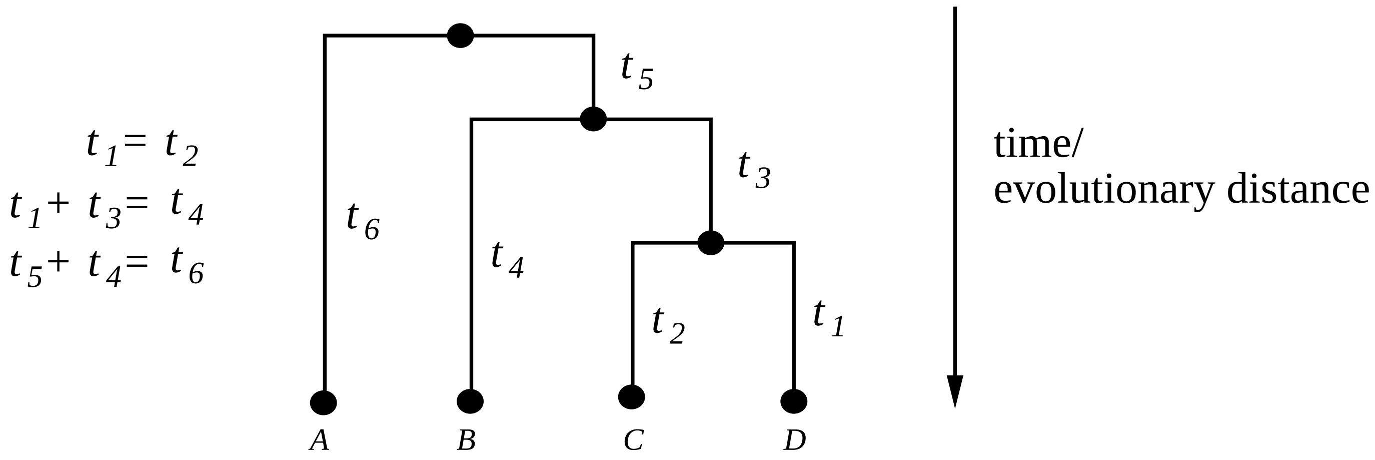
This is an even stronger version of the triangle inequality.

If d is an ultrametric, it is an additive metric.

If d is an additive metric, it is a metric.

Ultrametric trees

A weighted tree is called an *ultrametric tree* if it can be rooted in such a way that the distances from the root to each leaf are equal.



$t_1 = t_2$

$t_1 + t_3 = t_4$

$t_5 + t_4 = t_6$

There is a clear interpretation inherent to ultrametric trees: Sequences have evolved from a common ancestor at constant rate (molecular clock hypothesis).

The path metric of an ultrametric tree is an ultrametric. Conversely, if distances d^M between a set of objects form an ultrametric, there is one ultrametric tree T corresponding to the distance measure, that is $d^T = d^M$. Given an ultrametric, this ultrametric tree can easily be reconstructed by one of the agglomerative clustering procedures described below.

UPGMA (Unweighted pair group method using arithmetic averages

Given a set of objects O with n elements and distances $d_{i,j},\ i,j \in O$, initially each object is assigned a singleton cluster. Then the algorithm proceeds as follows:

While there is more than one cluster left, do:

1. Find the pair (i,j) with the smallest distance d_{ij} and create a new cluster u that joins clusters i and j.

2. Define the *height* (i.e. distance from leaves) of u to be $l_{ij} := d_{ij}/2$

3. Compute the distance d_{ku} of u to any other cluster: $d_{ku} := \frac{n_i d_{ki} + n_j d_{kj}}{n_i + n_j}$ where n_i is the number of elements in cluster i. d_{ku} is the arithmetic average of the original distances of all elements in k and all elements in u.

4. Remove i,j from the list of objects

Clustering

Different clustering methods differ in how they define the distance d_{ku} between two clusters.

single linkage clustering:

$$d_{ku} := \text{min}(d_{ki}, d_{kj})$$

complete linkage clustering:

$$d_{ku} := \text{max}(d_{ki}, d_{kj})$$

average linkage clustering or WPGMA (weighted pair group method using arithmetic averages):

$$d_{ku} := \frac{d_{ki} + d_{kj}}{2}$$

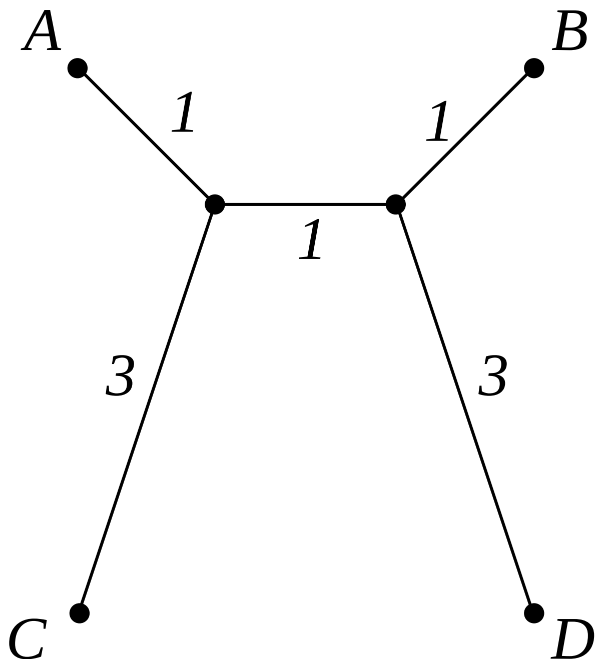
UPGMA

$$d_{ku} := \frac{n_i d_{ki} + n_j d_{kj}}{n_i + n_j}$$

UPGMA, cont'd

UPGMA was originally developed for phenetics, i.e. for constructing phenograms reflecting phenotypic similarities rather than evolutionary distances.

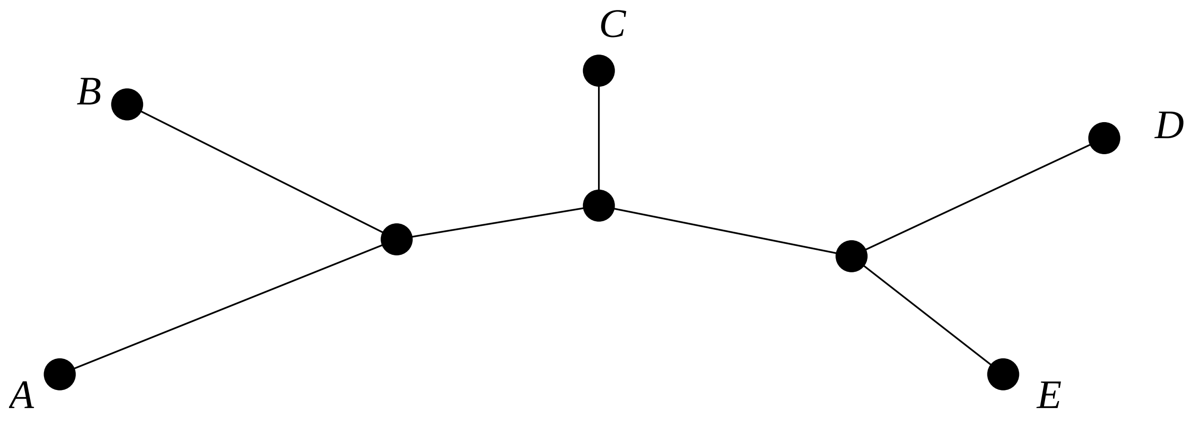
If the assumption of an approximately constant rate of evolution among the lineages does not hold, UPGMA fails to find the correct topology. Consider that taxa evolved according to the below tree:



Additive trees

We call a weighted binary tree an *additive tree*. Rates of evolution vary among species, among gene families, among genes, among sites in molecular sequences, and in time. *Additive trees* do not presume a constant evolutionary rate.

Given an additive metric there is exactly one tree topology that allows for realization of an additive tree.



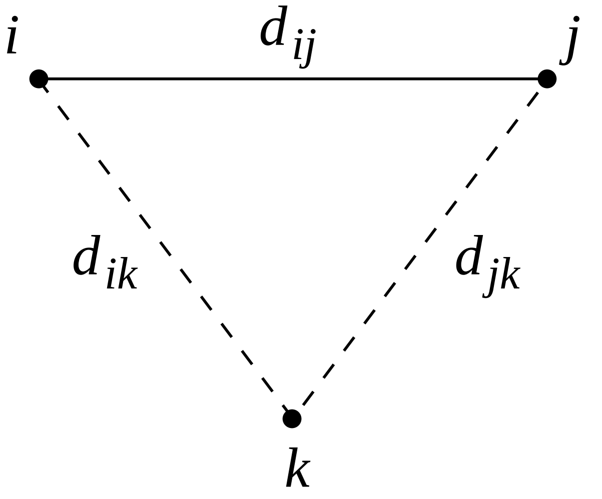
Exact reconstruction of additive trees

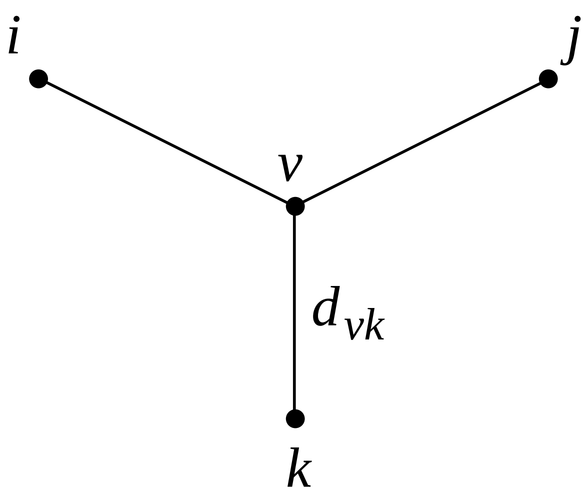
An additive metric can be represented as a unique additive tree which can be reconstructed in time complexity $O(n^2)$ (Waterman, Smith, Singh, Beyer, 1977).

The algorithm successively inserts objects into intermediate trees until no objects are left to insert. It makes use of the following rationale:

Given an intermediate tree T' containing leaf i and leaf j, one tests if one can insert an edge connecting leaf k to the intermediate tree along the path connecting i and j. Denote the node connecting i, j and k as v and the weight of the edge being inserted as d_{vk}.

Exact reconstruction of additive trees, cont'd





$$d_{ik} + d_{jk} = d_{iv} + d_{vk} + d_{jv} + d_{vk} = 2 \cdot d_{vk} + d_{ij}$$

$$d_{vk} = \frac{1}{2}(d_{ik} + d_{jk} - d_{ij})$$

$$\begin{aligned} d_{iv} &= d_{ik} - d_{vk} \\ d_{jv} &= d_{jk} - d_{vk} \end{aligned}$$

Neighbor Joining

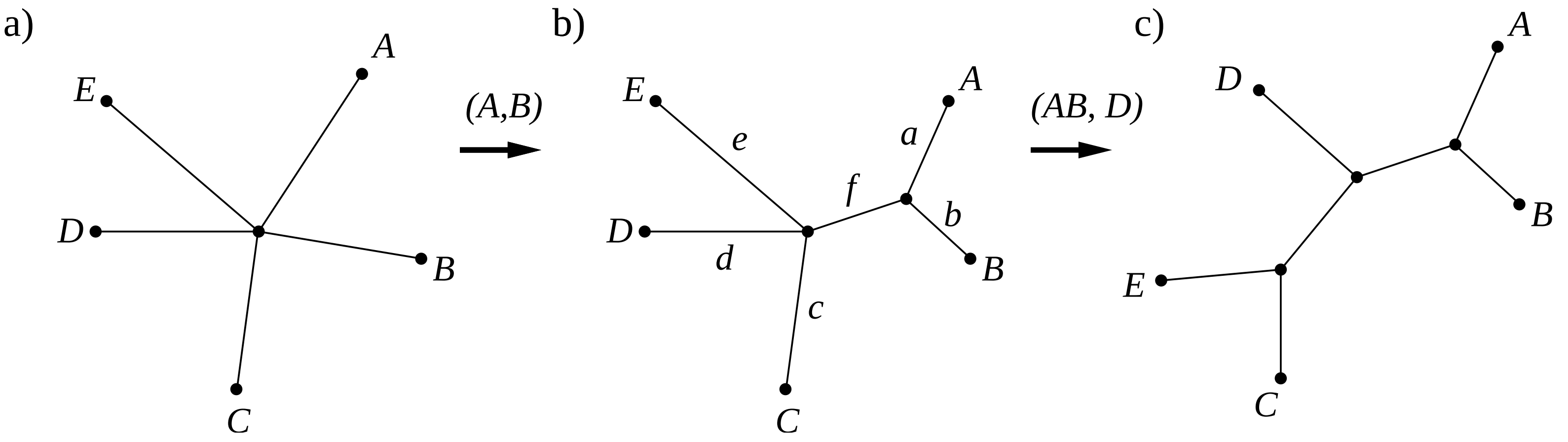
Since distance measures on multiple alignments practically do not provide an additive metric, the above algorithm to reconstruct the additive tree from an additive metric is not applicable to real data. The *neighbor joining* method (Saitou, Nei, 1987) is similar to cluster analysis in some ways. The individual taxa are iteratively grouped together, forming larger and larger clusters of taxa. In contrast to UPGMA, neighbor joining does not assume a molecular clock, but it assumes that observed distances are close to an additive metric. Given an additive metric, the neighbor joining method identifies the correct tree and it also correctly reconstructs trees if additivity only holds approximately.

Definition: Two taxa are *neighbors* in a tree if the path between them contains only one node.

As neighbor relationships of nodes in a binary tree uniquely define the tree topology, successively identifying neighbors is a way to reconstrut the tree. The time complexity of the Neighbor Joining algorithm is $O(n^3)$ given n taxa.

Neighbor Joining, cont'd

The concept to identify neighbors is the following: A star tree is decomposed



such that the tree length is minimized in each step. Consider the above star tree with N leaves shown in a). The star tree corresponds to the assumption that there is no clustering of taxa. In general there is a clustering of taxa and if so, the overall tree length (the sum of all branch lengths) S_F of the true tree or the final NJ tree (see c)) is smaller than the overall tree length of the star tree S_0.

Neighbor Joining, cont'd

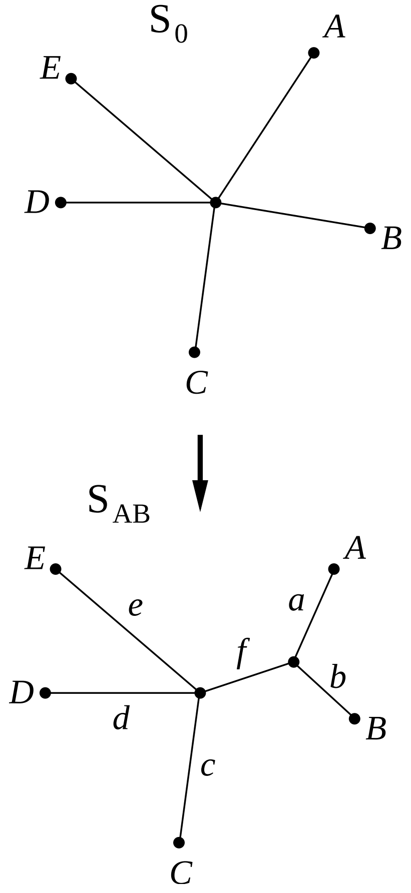
The tree length of the tree where neighbors i and j are resolved is

$$S_{ij} = \sum_{\substack{k=1 \\ k \neq i,j}}^{N} \frac{d_{ki} \; + \; d_{kj}}{2(N-2)} + \frac{d_{ij}}{2} + \sum_{\substack{k<l \\ k,l \neq i,j}}^{N} \frac{d_{kl}}{N-2}$$

where N is the number of taxa.

For example,

$$\begin{aligned} S_{AB} \quad &= \quad (3a + 3b + 6f + 2c + 2d + 2e) \cdot \frac{1}{6} \\ &+ \; \frac{a+b}{2} \; + \; (2c + 2d + 2e) \cdot \frac{1}{3} \\ &= \quad a + b + f + c + d + e \end{aligned}$$



Neighbor Joining, cont'd

Theorem: Given an additive tree T. O is the set of leaves of T. Values of S_{ij} are computed by means of the path metric d^T. Then $m, n \in O$ are neighbors in T, if $S_{mn} \leq S_{ij} \quad \forall\ i, j \in O$.

In other words, if our distances form an additive metric, we can identify neighbors in the additive tree by computing S_{ij} for all pairs of taxa if the distances form an additive metric.

The neighbors are combined into one composite taxon and the procedure is repeated.

We rewrite S_{ij}:

$$\begin{aligned} S_{ij} &= \frac{1}{2(N-2)} \left(2 \cdot \sum_{\substack{k<l \\ k,l \neq i,j}}^{N} d_{kl} + \sum_{\substack{k=1 \\ k \neq i,j}}^{N} (d_{ki} + d_{kj}) \right) + \frac{d_{ij}}{2} \\ &= \frac{1}{2(N-2)} \left(2 \cdot \sum_{k<l}^{N} d_{kl} - r_i - r_j \right) + \frac{d_{ij}}{2} \end{aligned}$$

with $r_i := \sum_{k=1}^{N} d_{ik}$.

Neighbor Joining, cont'd

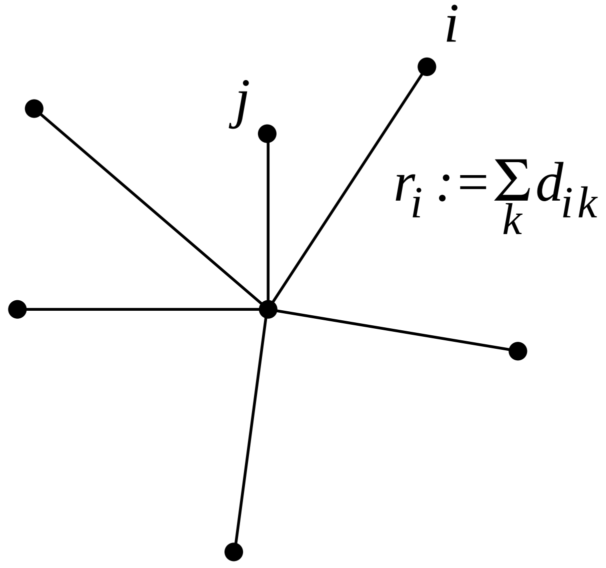
Since the sum $\sum_{k<l}^{N} d_{kl}$ is the same for all pairs of taxa k and l, we can replace S_{ij} by

$$M_{ij} := d_{ij} - \frac{r_i + r_j}{N-2}$$

for the purpose of easier computation of relative values of S_{ij}.

r_i is also called *net divergence*.

$\frac{r_i+r_j}{N-2}$ holds averaged distances of i and j to all other leaves. Thus, if i and j were neighbors in evolution and i or j evolved fast such that d_{ij} is large, $\frac{r_i+r_j}{N-2}$ is also large and M_{ij} gets small.



Neighbor Joining, cont'd

Algorithm: Given distances d_{ij} between members of a set O of N objects. Represent the objects as terminal nodes in a starlike tree:

1. For each terminal node i compute

$$r_i := \sum_{k=1}^{N} d_{ik}.$$

2. For all pairs of terminal nodes (i, j) compute

$$M_{ij} := d_{ij} - \frac{r_i + r_j}{N - 2}.$$

   Let (i, j) be a pair with minimal value M_{ij} for $i \neq j$.

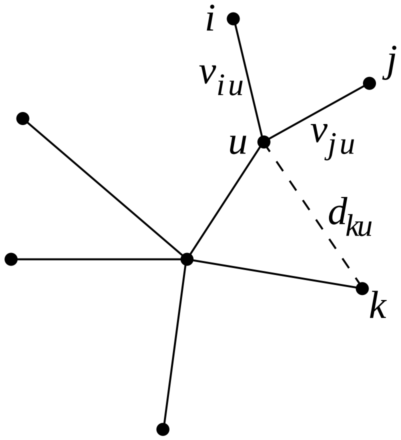
3. Join nodes i and j into a new terminal node u. The branch lengths from u to i and j are:

$$v_{iu} = \frac{d_{ij}}{2} + \frac{r_i - r_j}{2N - 4} \quad \text{and} \quad v_{ju} = d_{ij} - v_{iu}.$$

Neighbor Joining, cont'd

4. Obtain the distances from u to another terminal node k by

$$d_{ku} = \frac{d_{ik} + d_{jk} - d_{ij}}{2}.$$



5. Delete i and j from the set of objects. If there are more than two clusters left, continue with Step 1

Neighbor Joining, cont'd

- NJ is fast ($O(n^3)$) and therefore it is suited to be applied to large data sets
- takes rate differences into account
- makes use of distance measure and its model
- result is one tree ($\rightarrow$ Bootstrapping)
- reduction of sequence information
- no objective function

Least Squares on Distances

The problem addressed in reconstructing trees on distances is to find a tree T with path metric d^T on measured distances d^M. This problem can be divided into identifying the topology and reconstructing the edge lengths. Neighbor Joining solves the problem algorithmically and all at once.

Given a tree topology, Fitch and Margoliash (1967) apply an objective function to fit d^T to d^M. They define the disagreement between a tree and the distance measure by the sum of squared weighted differences in distances:

$$E := \sum_{i<j} |d_{ij}^T - d_{ij}^M|^2 \cdot \frac{1}{(d_{ij}^M)^2}$$

The weights take into account relative uncertanties in the distance measures and may be adapted. d^T is obtained by minimizing E.

Methods for phylogeny reconstruction ...

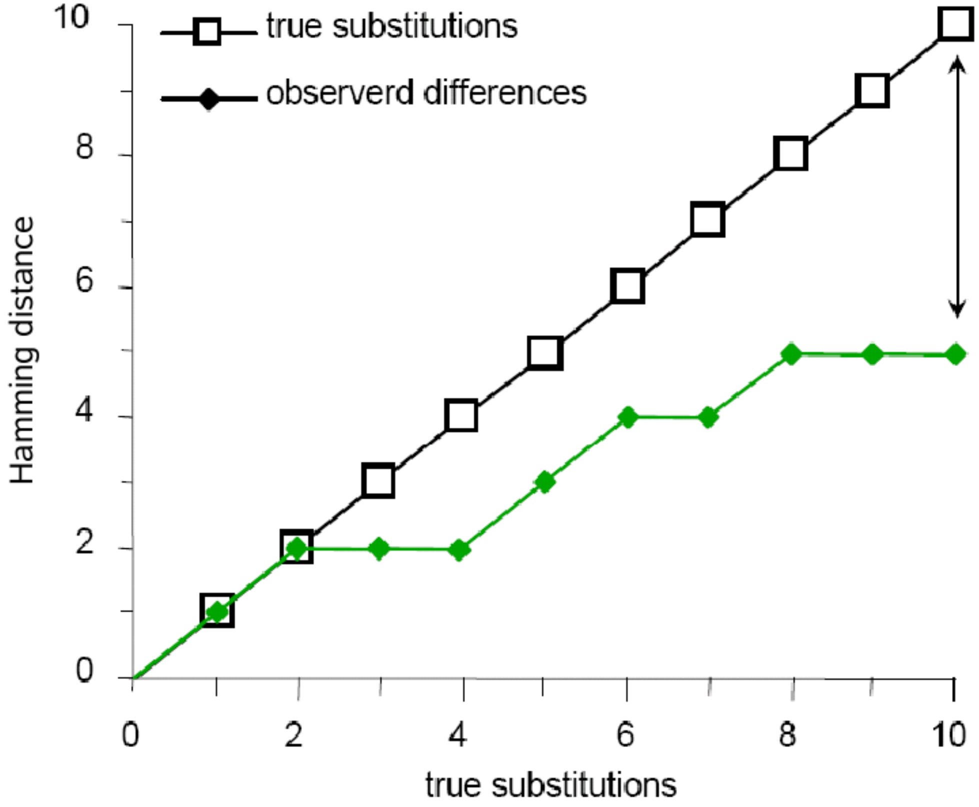
... can also be classified according to whether they find the tree algorithmically or whether they optimize an objective function

- **with objective function**
  - Maximum Parsimony
  - Least Squares (Fitch–Margoliash)
  - Maximum Likelihood
- **algorithmic**
  - UPGMA, clustering
  - Neighbor Joining

Summarizing distance based methods, keywords to remember:

- Additive metrics and ultrametrics
- UPGMA and hierarchical clustering, time complexity $O(n^2)$
- concept of Neighbor Joining, time complexity $O(n^3)$

Evolutionary distances



$t = 0$	C	C	A	T	G	C	G
				↓			
$t = 1$	C	C	A	C	G	C	G
			↓				
$t = 2$	C	C	G	C	G	C	G
			↓				
$t = 3$	C	C	C	C	G	C	G
				↓			
$t = 4$	C	C	C	G	G	C	G
		↓					
$t = 5$	C	A	C	G	G	C	G
	↓						
$t = 6$	G	A	C	G	G	C	G
				↓			
$t = 7$	G	A	C	T	G	C	G
							↓
$t = 8$	G	A	C	T	G	C	A
	↓						
$t = 9$	A	A	C	T	G	C	A
			↓				
$t = 10$	A	A	T	C	G	C	A

Evolutionary distances, cont'd

- We are given a multiple alignment and want to obtain pairwise evolutionary distances

- With u as the number of mismatches in an alignment of length n, the Hamming distance per 100 sites is

$$D(u, n) = 100 \; \frac{u}{n}$$

- The distance D does not take multiple substitutions into account. As a consequence, pairwise distances are not additivie.

- For any number of mismatches u and alignment lengths n, we have

$$0 <= D <= 100$$

. For example

$$D(u = 0, n = 100) = 0 \quad \text{and} \quad D(u = 75, n = 100) = 75$$

Evolutionary distances, cont'd

- Pairwise evolutionary distances $d(u, n)$ are meant to scale in units of substitutions (per 100 sites) that *most likely* have occured on the evolutionary paths.

- If we assume (as in the Jukes-Cantor model, see below)

  i) that sequence positions are i.i.d. (*independently identically distributed*)

  ii) that nucleotides are uniformly distributed and independently substituted such that the probabilities for nucleotide substitutions are all the same and do not depend on the particular nucleotides

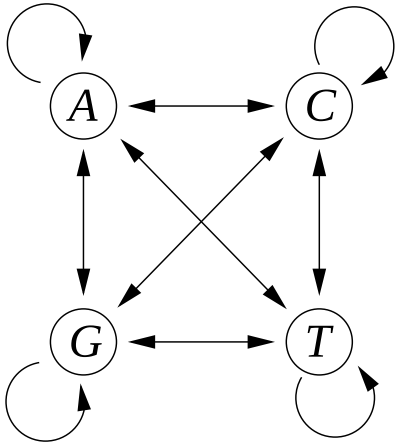
  we require that

  $$d(u = 0, n = 100) = 0 \quad \text{and} \quad d(u = 75, n = 100) = \infty$$

  (the latter follows from the requirement that the evolutionary distance for two random sequences is $d = \infty$)

Markov chains ("time-discrete Markov processes")

1. The states are A, C, G, T
2. A starting distribution of states $\rho^0 = (\rho_A, \rho_C, \rho_G, \rho_T)$
3. Transition probabilites in one "time step" between states $P_{ij} = \mathrm{Pr}(j|i)$



State probabilities depend only on the previous state and not on the past of the chain (Markov property). If transition probabilities don't change in time (homogeneity) the probability of a sequence $x = (x_1, ..., x_L)$ is

$$\mathrm{Pr}(x) = \rho_{x_1} \prod_{i=2}^{L} \mathrm{Pr}(x_i|x_{i-1})$$

Transition probabilities for n steps are obtained from the n-th power of the stochastic one-step transition matrix P, from P^n.

The Markov model of sequence evolution

Sequence evolution is modeled by a (time-continuous) *Markov process* that acts **independently** on the sites of the sequence.

$$\begin{array}{rcccccc}
X_{t_1} = & \mathbf{A} & \mathrm{T} & \mathrm{C} & \mathrm{G} & \mathrm{C} & \cdots \\
 & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \\
X_{t_2} = & \mathbf{G} & \mathrm{T} & \mathrm{C} & \mathrm{A} & \mathrm{G} & \cdots \\
 & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \\
X_{t_3} = & \mathbf{G} & \mathrm{T} & \mathrm{C} & \mathrm{A} & \mathrm{C} & \cdots \\
 & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \\
X_{t_4} = & \mathbf{A} & \mathrm{G} & \mathrm{C} & \mathrm{A} & \mathrm{G} & \cdots
\end{array}$$

A *Markov process* is a sequence of random variables $(X_t)_{t \geq 0}$ given by a triple $\left(\mathcal{A}, \rho^0, Q\right)$, where $\mathcal{A} = \{1, ..., n\}$ is the set of states (nucleotides or amino acid residues) (X_t) takes, ρ^0 is the initial probability distribution of states ($\rho_i^0 = Pr[X_0 = i]$) and the rate matrix Q as a $n \times n$ matrix with substitution rates (something like transition probabilities for infinitesimal small time steps) between states.

The Markov model of sequence evolution, cont'd

- **Markov property** (the process is memoryless):
  $\Pr[X(t_n) = s|X(t_1) = i_1, X(t_2) = i_2, ..., X(t_{n-1}) = i_{n-1}]$
  $= \Pr[X(t_n) = s|X(t_{n-1}) = i_{n-1}]$

- **Homogeneity**:
  Transition probabilities only depend on the time interval:
  $P_{ij}(t) = \Pr[X_{t+s} = j|X_s = i] \; = \; \Pr[X_t = j|X_0 = i]$

- The time t of the Markov process is measured in units of substitutions

- The transition probablity $P_{ij}(t)$ is the probability that state i changes into state j in time t

- We think of the distribution $\rho(t)$ as a row vector. The evolution of the distribution of states at time s in time t is given by

$$\rho(s)P(t) = \rho(s+t)$$

The Markov model of sequence evolution, cont'd

- **Stationary distribution:**

  π is the *stationary distribution* of the process, if π doesn't change in time:

  $$\pi_j = \sum_{i \in \mathcal{A}} \pi_i P_{ij}(t) \quad \text{for all } j$$

  $$\pi P(t) = \pi$$

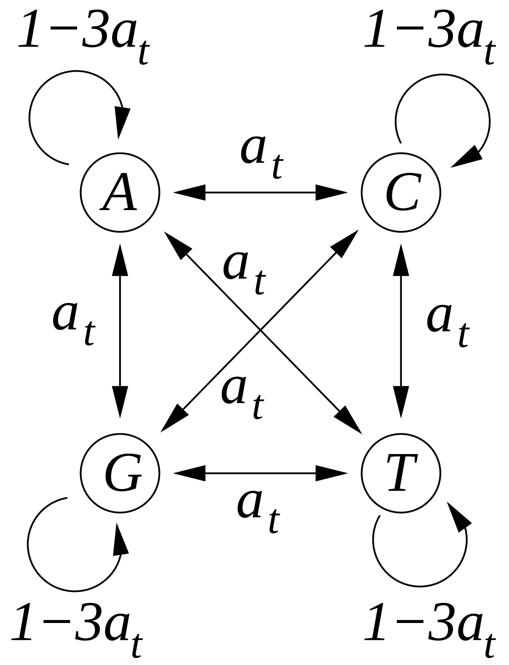
  We say that the process is in equilibrium if the distribution of the process is the stationary distribution π.

  π exists if any state can be reached by any other state.

The transition probability matrix

For nucleotides, the simplest model is the *Jukes–Cantor–model* (1969). The set of states comprises the nucleotides ($\mathcal{A} = \{1, 2, 3, 4\}$). The stationary distribution π of nucleotides is the uniform distribution ($\pi = (\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$) and the probabilities that any nucleotide is substituted by another any other nucleotide are equal.

Thus, the *transition probability matrix* of the Jukes–Cantor model has the form



$$P(t) = \begin{pmatrix} 1-3a_t & a_t & a_t & a_t \\ a_t & 1-3a_t & a_t & a_t \\ a_t & a_t & 1-3a_t & a_t \\ a_t & a_t & a_t & 1-3a_t \end{pmatrix}$$

The transition probability matrix, cont'd

The transition probability matrix $P(t)$ is a stochastic matrix and has the following properties:

- $P(0) = I, \qquad I$ - identity matrix,
- $P_{ij}(t) \geq 0$ and $\sum_j P_{ij}(t) = 1$,
- $P(s+t) = P(s)P(t)$

The latter equation is called *Chapman–Kolmogorov equation*. E.g. think of $\mathcal{A} = \{1, 2, 3, 4\}$ and the process being in state 1 reaching state t in time $s + t$. The transition probability $P_{14}(s+t)$ is

$$\begin{aligned}
Pr[X_{s+t} = 4 | X_0 = 1] &= Pr[X_s = 1 | X_0 = 1] \cdot Pr[X_{s+t} = 4 | X_s = 1] \\
&\quad + Pr[X_s = 2 | X_0 = 1] \cdot Pr[X_{s+t} = 4 | X_s = 2] \\
&\quad + Pr[X_s = 3 | X_0 = 1] \cdot Pr[X_{s+t} = 4 | X_s = 3] \\
&\quad + Pr[X_s = 4 | X_0 = 1] \cdot Pr[X_{s+t} = 4 | X_s = 4] \\
&= \textstyle\sum_{k \in \mathcal{A}} P_{1k}(s) P_{k4}(t)
\end{aligned}$$

Maximum Likelihood and coin tossing

Assume, we have flipped a coin 10 times and got 7 times its head and 3 times its tail. We want to estimate the probability Prob(head), that the head shows up when the coin is flipped?

The likelihood $\mathcal{L}(p)$ is the probability to observe one outcome (of many possible outcomes) of a random experiment (one data set) under the probabibilistic model with its model parameter p.

$\mathcal{L}(p) = \mathrm{Pr}(\mathrm{data}|p) = p^7(1-p)^3$

We think of the likelihood as a function depending on the model parameters. Note that the sum or the integral over the parameter space is not 1!

$\widehat{p} =$ Prob(head) is determined as the p where $\mathcal{L}$ assumes its maximum.

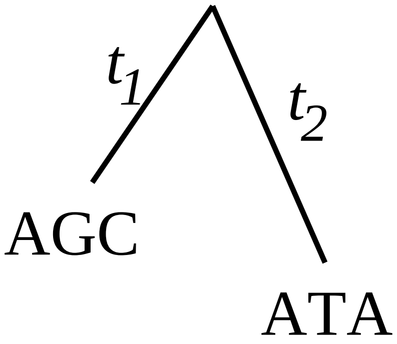
The variance of the estimate depends on the sample size and can be estimated from the likelihood curvature. If the data was generated under the model, the ML estimate of the parameters yields exact or true values for infinite sample sizes.

Evolutionary distances with Maximum Likelihood

We think of the observed alignment $\mathcal{D}$ as the outcome of the Markovian evolution.

Consider the following alignment $\mathcal{D}$:

A G C
A T A



We assume that the process is in equilibrium.

The *likelihood* to observe the alignment $\mathcal{D}$ (the data) with distance $t = (t_1 + t_2)$ given the Markov model $\mathcal{M}$ then is

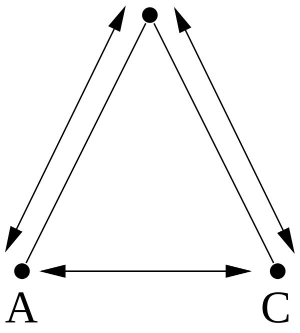
$$\Pr(\mathcal{D}|t, \mathcal{M}) = \sum_{i \in \mathcal{A}} \pi_i P_{iA}(t_1) P_{iA}(t_2) \cdot \sum_{i \in \mathcal{A}} \pi_i P_{iG}(t_1) P_{iT}(t_2) \cdot \sum_{i \in \mathcal{A}} \pi_i P_{iC}(t_1) P_{iA}(t_2)$$

Evolutionary distances with Maximum Likelihood, cont'd

The Markov process is called *reversible*, if the evolution of state i into state j in time t is modelled by the same process as the evolution of state i into state j in time t:

$$\pi_i P_{ij}(t) = \pi_j P_{ji}(t) \quad \text{for all} \;\; i, j, t$$

(*detailed balance equations*)



We assume the time-reversible Jukes-Cantor model and apply the Chapman-Kolmogorov equations:

$$\begin{aligned} \text{Pr}(\mathcal{D}|t, \mathcal{M}) &= \pi_A P_{AA}(t) \cdot \pi_G P_{GT}(t) \cdot \pi_C P_{CA}(t) \\ &= \pi_A P_{AA}(t) \cdot \pi_T P_{TG}(t) \cdot \pi_A P_{AC}(t) \end{aligned}$$

If the process is reversible and if we are given a pairwise alignment, we are ignorant about the location of the root node.

Evolutionary distances with Maximum Likelihood, cont'd

Consider $\mathsf{Pr}(\mathcal{D}|t,\mathcal{M})$ as likelihood function depending on the distance t as model parameter:

$$\log \mathcal{L}(t) = \log \mathsf{Pr}(\mathcal{D}|t,\mathcal{M})$$

The evolutionary distance is estimated as distance $\widehat{t}$ where the likelihood function assumes its maximum.

If sequences have evolved according to the evolutionary model $(\mathcal{M},t)$, and if we have infinitely many samples (alignment columns) of the outcome of this evolution, the evolutionary distance can be exactly reestimated by Maximum Likelihood (ML), i.e. the ML distance estimator is consistent. For finite sample sizes, ML estimates $\widehat{t}$ are normally distributed around the 'true' value for t.

We have to evaluate the likelihood function and thus the transition probabilities for different times or distances t. This is achieved by means of the rate matrix...

The rate matrix

The *rate matrix* Q of a time-continuous Markov process provides an infinitesimal description of the process.

We assume that the probability transition matrix $P(t)$ of a time continuous Markov process is continuous and differentiable at any $t > 0$. I.e. the limit

$$Q := \lim_{t \searrow 0} \frac{P(t) - I}{t}$$

exists. Q is known as the *rate matrix* or the *generator* of the Markov chain. For very small time periods $h > 0$, transition probabilities are approximated by

$$\begin{aligned} P(h) &\approx I + hQ \\ P_{ij}(h) &\approx Q_{ij} \cdot h, \qquad i \neq j. \end{aligned}$$

From the last equation we see, that the entries of Q may be interpreted as substitution rate.

The rate matrix, cont'd

From the Chapman-Kolmogorov equation we get

$$\begin{aligned}
\frac{d}{dt}P(t) &= \lim_{h \searrow 0} \frac{P(t+h) - P(t)}{h} \\
&= \lim_{h \searrow 0} \frac{P(t)P(h) - P(t)I}{h} \\
&= P(t) \lim_{h \searrow 0} \frac{P(h) - P(0)}{h} \\
\frac{d}{dt}P(t) &= P(t)Q
\end{aligned}$$

Under the initial condition $P(0) = I$ the differential equation can be solved and yields (as in the one–dimensional case)

$$P(t) = \exp(tQ) = \sum_{k=0}^{\infty} \frac{Q^k t^k}{k!}.$$

Transition probabilities for any $t > 0$ are computed from the matrix Q.

The rate matrix, cont'd

Recall, that for very small h we have $P(h) \approx I + hQ$.

Q has the following properties:

- $Q_{ij} \geq 0 \quad \text{for} \quad i \neq j$
- $Q_{ij} \geq 0,\ i \neq j \ \Rightarrow \ Q_{ii} \leq 0$
- $\sum_j Q_{ij} = 0,\ Q_{ii} = -\sum_{j \neq i} Q_{ij}$

Further,

- π is stationary distribution if $\quad \pi Q = 0$
- the process is reversible, if $\quad \pi_i Q_{ij} = \pi_j Q_{ji} \quad$ for all $\ i, j$

The rate matrix, cont'd

The rate matrix of the Jukes-Cantor model is

$$Q = \begin{pmatrix} -3\alpha & \alpha & \alpha & \alpha \\ \alpha & -3\alpha & \alpha & \alpha \\ \alpha & \alpha & -3\alpha & \alpha \\ \alpha & \alpha & \alpha & -3\alpha \end{pmatrix}.$$

where $\alpha \geq 0$.

Due to the simple structure of Q, exp(tQ) can be calculated analytically. The transition probability matrix is

$$P(t) = \begin{pmatrix} 1-3a_t & a_t & a_t & a_t \\ a_t & 1-3a_t & a_t & a_t \\ a_t & a_t & 1-3a_t & a_t \\ a_t & a_t & a_t & 1-3a_t \end{pmatrix},$$

where

$$a_t = \frac{1 - \exp(-4\alpha t)}{4}$$

The rate matrix, cont'd

If we assume the stationary distribution, Q summarizes all model parameters of the Markov process, since $\pi Q = 0$. Clearly, Q can be multiplied with a factor and the distribution π doesn't change. In other words: The model parameters hold substitution rates. And rates hold the information how many substitutions per time unit one expects.

The rate matrix can be calibrated to *PAM (percent accepted mutations)*–units. 1 PAM is the time (or evolutionary distance) where one substitution event per 100 sites is expected to have occured.

Given Q, one expects $E = \sum_i \pi_i \sum_{j \neq i} Q_{ij} = -\sum_i \pi_i Q_{ii}$ substitution events per time unit.

The Jukes–Cantor rate matrix Q is calibrated to PAM-units by setting $E = \frac{1}{100} \Leftrightarrow -4 \cdot \frac{1}{4} \cdot -3\alpha = \frac{1}{100} \Leftrightarrow \alpha = \frac{1}{300}$.

Evolutionary distances with Maximum Likelihood

Again, consider the log likelihood of the alignment $\mathcal{D}$: $\begin{matrix} \mathsf{A} & \mathsf{G} & \mathsf{C} \\ \mathsf{A} & \mathsf{T} & \mathsf{A} \end{matrix}$

We had

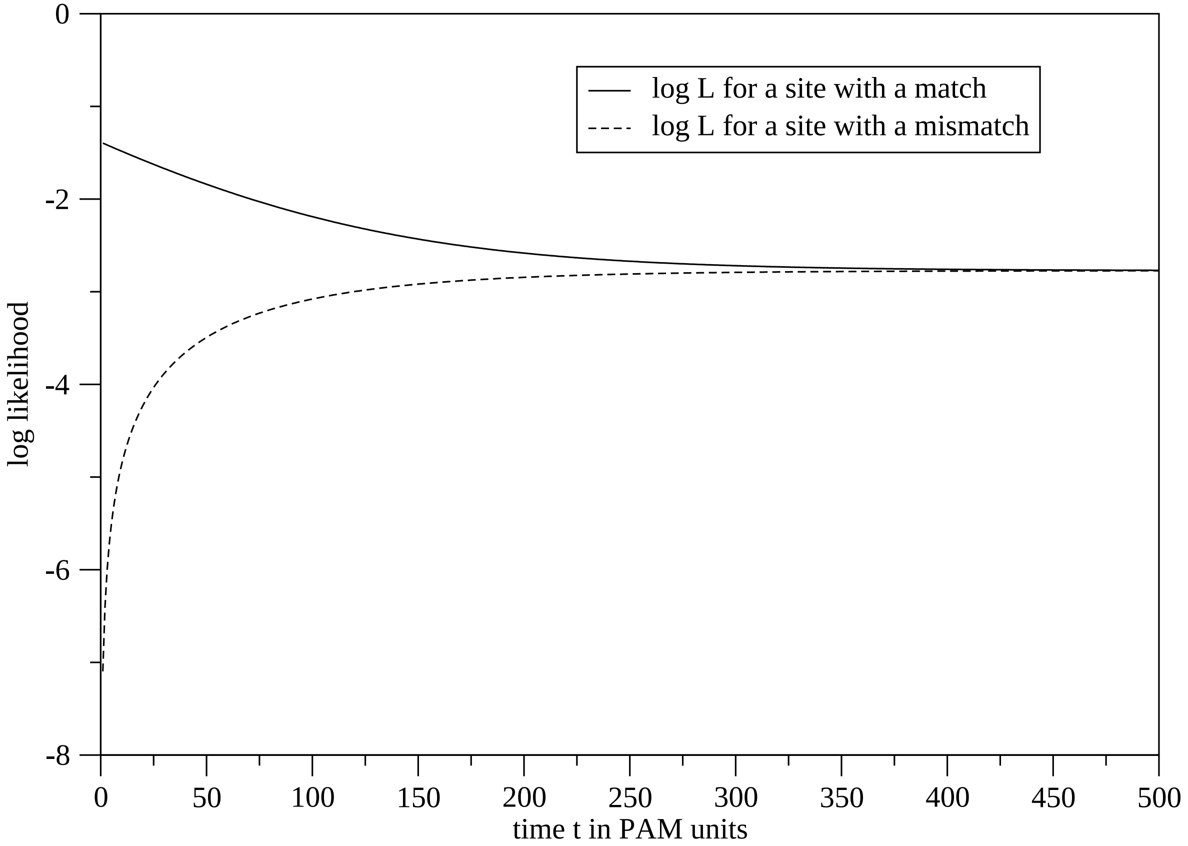
$$\log \mathcal{L}(t) = \log(\pi_A P_{AA}(t)) + \log(\pi_G P_{GL}(t)) + \log(\pi_C P_{CA}(t))$$

with the Jukes-Cantor model:

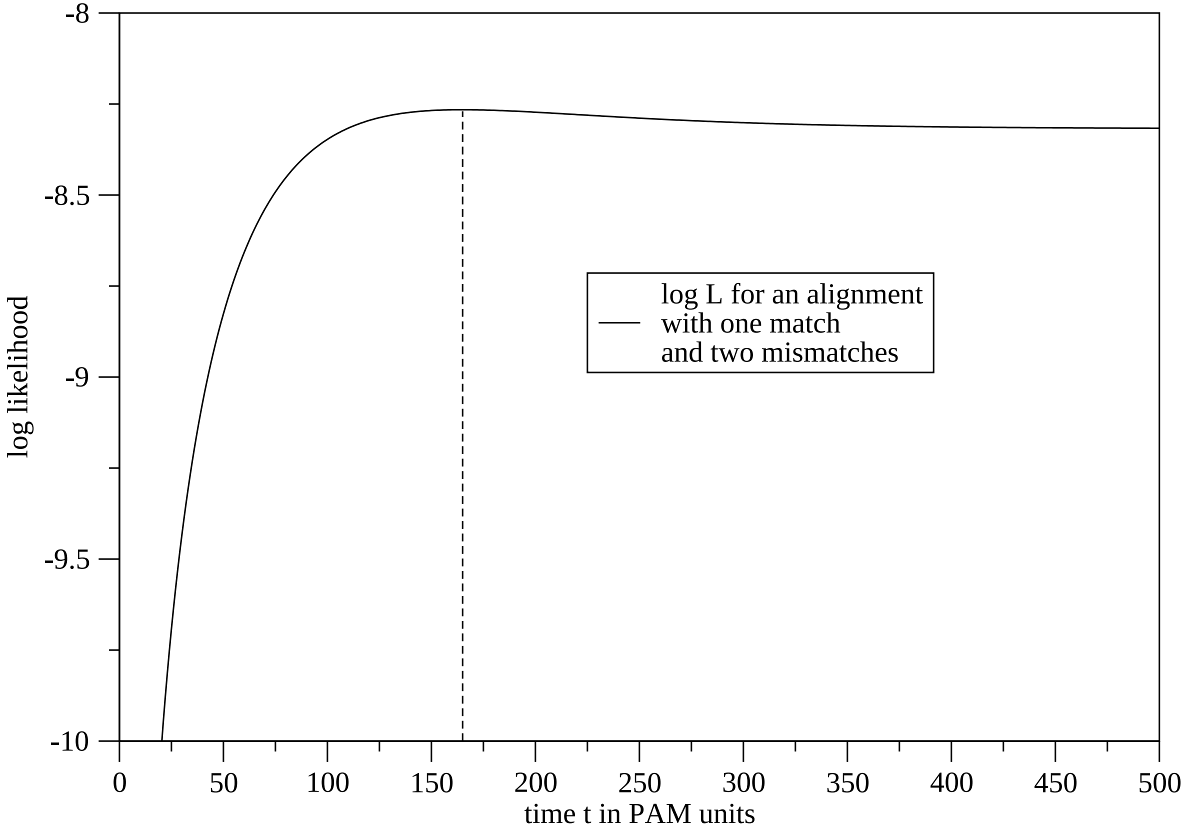
$$\log \mathcal{L}(t) = \log \Big(\frac{1}{4} \cdot \big(1 - \frac{3}{4} \cdot (1 - \exp(\frac{-4}{300}\ t)) \big) \Big) + 2 \cdot \log \Big(\frac{1}{4} \cdot \frac{1 - \exp(\frac{-4}{300}\ t)}{4} \Big)$$

Evolutionary distances with Maximum Likelihood, cont'd

The log likelihood functions for the single alignment columns and JC69:



Evolutionary distances with Maximum Likelihood, cont'd



The Maximum Likelihood estimate $\widehat{t} = 165$ PAM is the value for t where where $\log \mathcal{L}(t)$ is maximal. The variance of the estimate is huge because i) the small sample size, ii) the large distance. Variances can be computed from the second derivative of $\log \mathcal{L}(\widehat{t})$.

The Jukes-Cantor correction

The Hamming distance $D = \frac{100 \cdot u}{n}$ (u-mismatches, n- sequence length) for the distance between two DNA sequences ignores the putative occurence of multiple substitutions. The Jukes-Cantor correction d provides a formula for the evolutionary distance d of two DNA sequences, i.e. $d(u,n)$ holds the number of substitutions which are expected to have occured per 100 sites.

The probability p to observe that a nucleotide is not substituted after time t is

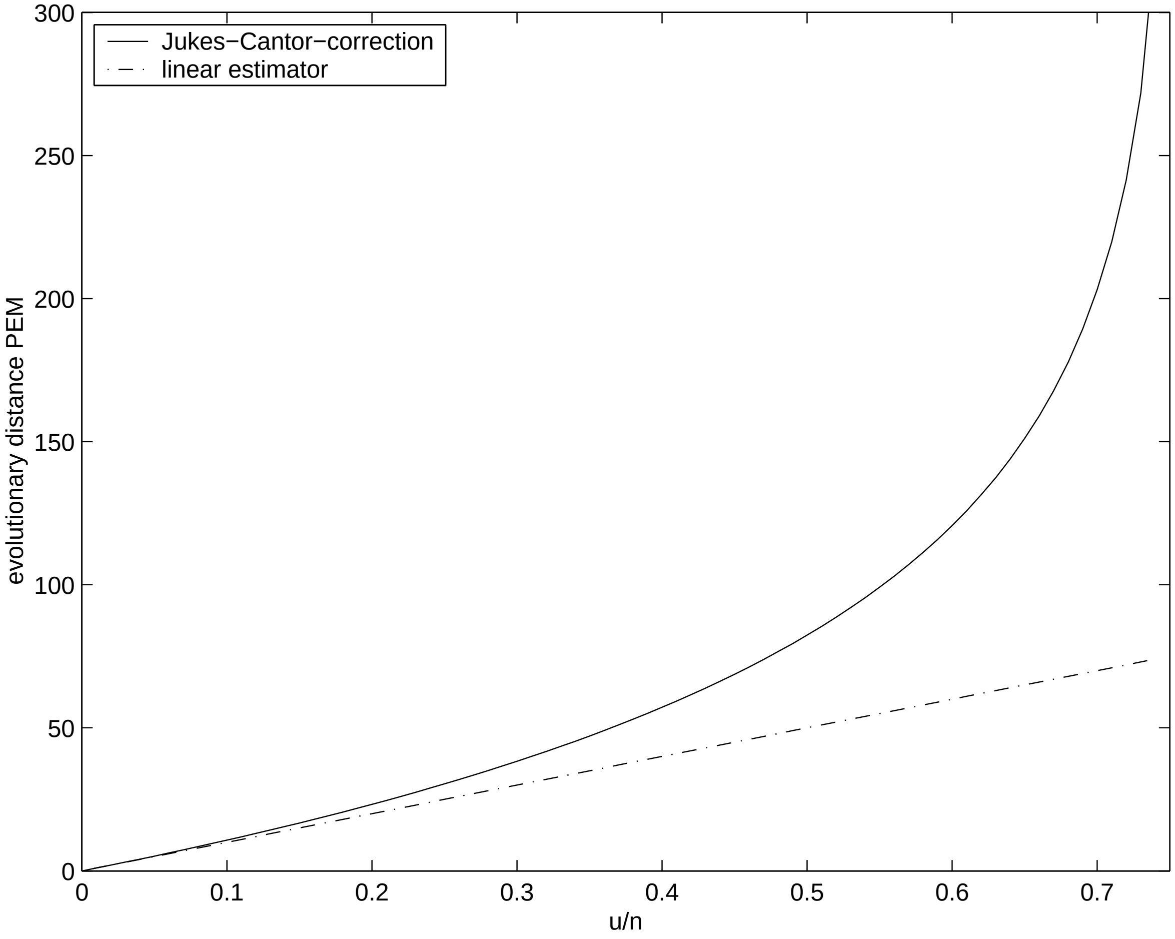
$p = \sum_i \pi_i P_{ii}(t) = 4 \cdot \frac{1}{4}(1 - 3a_t) = 1 - \frac{3}{4}(1 - \exp(-4\alpha t)) = \frac{1+3\exp(-4\alpha t)}{4}$

There are u mismatches among n sites. That is, we observe $p = 1 - \frac{u}{n}$. Calibration to PAM–units and setting $t = d$ yields

$$1 - \frac{u}{n} = \frac{1 + 3\exp(-4d/300)}{4}$$

Jukes–Cantor correction, cont'd

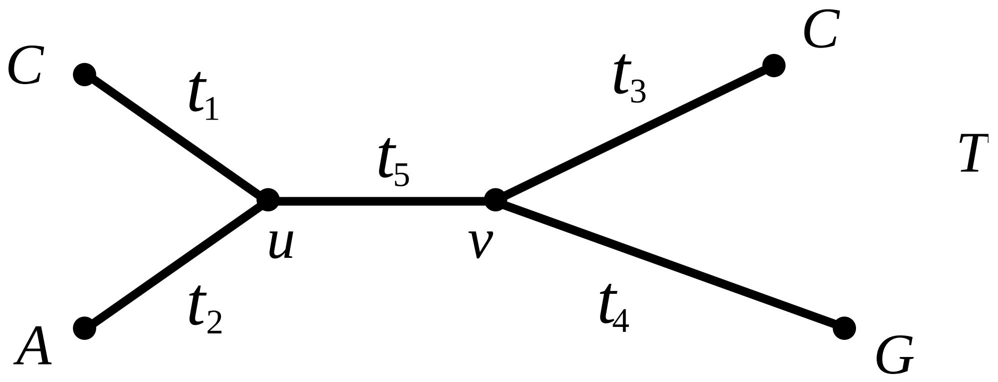
$$d = -\frac{300}{4} \ln\left(1 - \frac{4u}{3n}\right) \text{ PAM}$$



If $\frac{u}{n} \geq 0.75$, d is not defined.

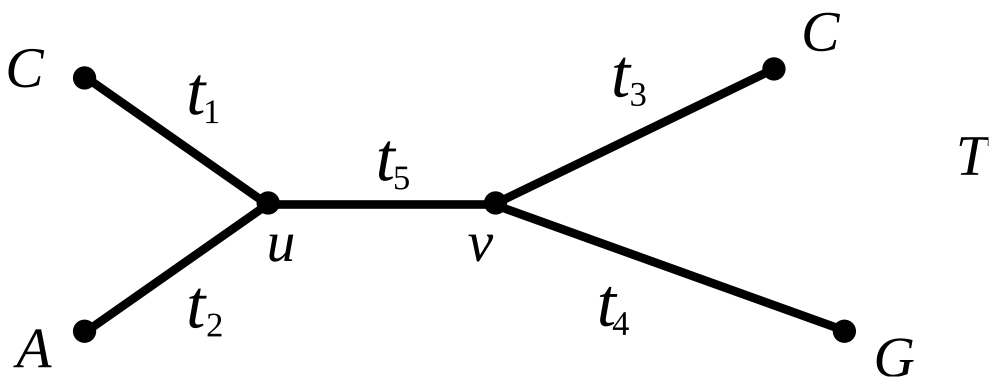
Maximum Likelihood Trees

Consider one site $\mathcal{D}_s$ of an alignment with the states A,C,C,T $\in \mathcal{A}$. We consider a particular tree topology $\mathcal{T}$ with edge lengths $\vec{t} = (t_1, ..., t_5)$ and label the leaves with the states of the alignment.



We want to compute the likelihood to observe the states under this tree $(T, \vec{t})$ and the Markov model Q. Reversibility implies that the likelihood does not depend on the position of a root node.

Maximum Likelihood Trees, cont'd



We choose node u as root node. First assume that we know states at internal nodes u and v and that both of them are C. Then

$$\mathcal{L}(T, \vec{t}, Q \mid \mathcal{D}_s, [C, C]) = \pi_C P_{CC}(t_1) P_{CA}(t_2) P_{CC}(t_5) P_{CC}(t_3) P_{CG}(t_4)$$

Because we do not know states at internal nodes

$$\mathcal{L}(T, \vec{t}, Q \mid \mathcal{D}_s) = \sum_{i \in \mathcal{A}} \pi_i P_{iC}(t_1) P_{iA}(t_2) \sum_{j \in \mathcal{A}} P_{ij}(t_5) P_{jC}(t_3) P_{jG}(t_4)$$

Note that we have 4^n summands for n internal nodes.

Maximum Likelihood Trees, cont'd

Recursive definition of the likelihood

We want to apply a dynamic programming strategy to compute the likelihood. The algorithm requires a rooted tree which is traversed from the leaves to the root (as the Sankoff algorithm does).

Felsenstein (1981) defines the conditional likelihood

$$\mathcal{L}_k(w)$$

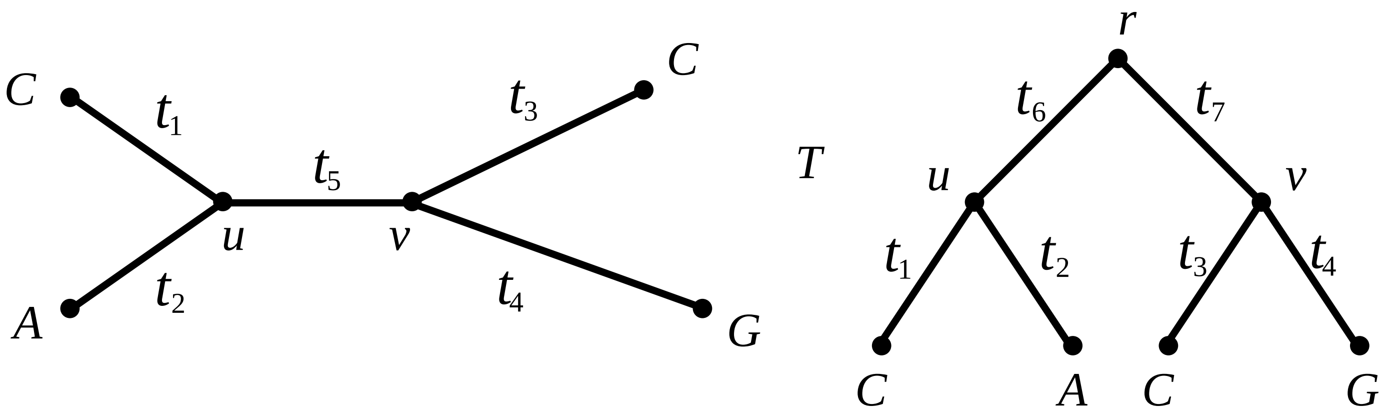
as the likelihood of the subtree rooted at node w, given that node w has state $k \in \mathcal{A}$.

At a leaf node l we have

$$\mathcal{L}_k(l) = \begin{cases} 1 & \text{if the leaf has state } k \\ 0 & \text{else} \end{cases}$$

Maximum Likelihood Trees, cont'd

For ease of illustration, we now insert a root node r at the internal edge such that $t_5 = t_6 + t_7$.



The conditional likelihood at the node r is

$$\mathcal{L}_k(r) = \left(\textstyle\sum_{i \in \mathcal{A}} P_{ki}(t_6) \mathcal{L}_i(u) \right) \cdot \ \left(\textstyle\sum_{i \in \mathcal{A}} P_{ki}(t_7) \mathcal{L}_i(v) \right)$$

r is already the root of the tree. Thus

$$\mathcal{L}(T, \vec{t}, Q \mid \mathcal{D}_s) = \sum_{i \in \mathcal{A}} \pi_i \mathcal{L}_i(r)$$

Maximum Likelihood Trees, cont'd

Note that the number of summands in the likelihood function now is linear in the number of internal nodes.

Sites are modeled independently of each other. The likelihood to observe an alignment $\mathcal{D}$ with n sites is the product over the site likelihoods

$$\mathcal{L}(T, \vec{t}, Q \mid \mathcal{D}) = \prod_{s=1}^{n} \mathcal{L}(T, \vec{t}, Q \mid \mathcal{D}_s)$$

Accordingly, the log likelihood is a sum over the site log likelihoods.

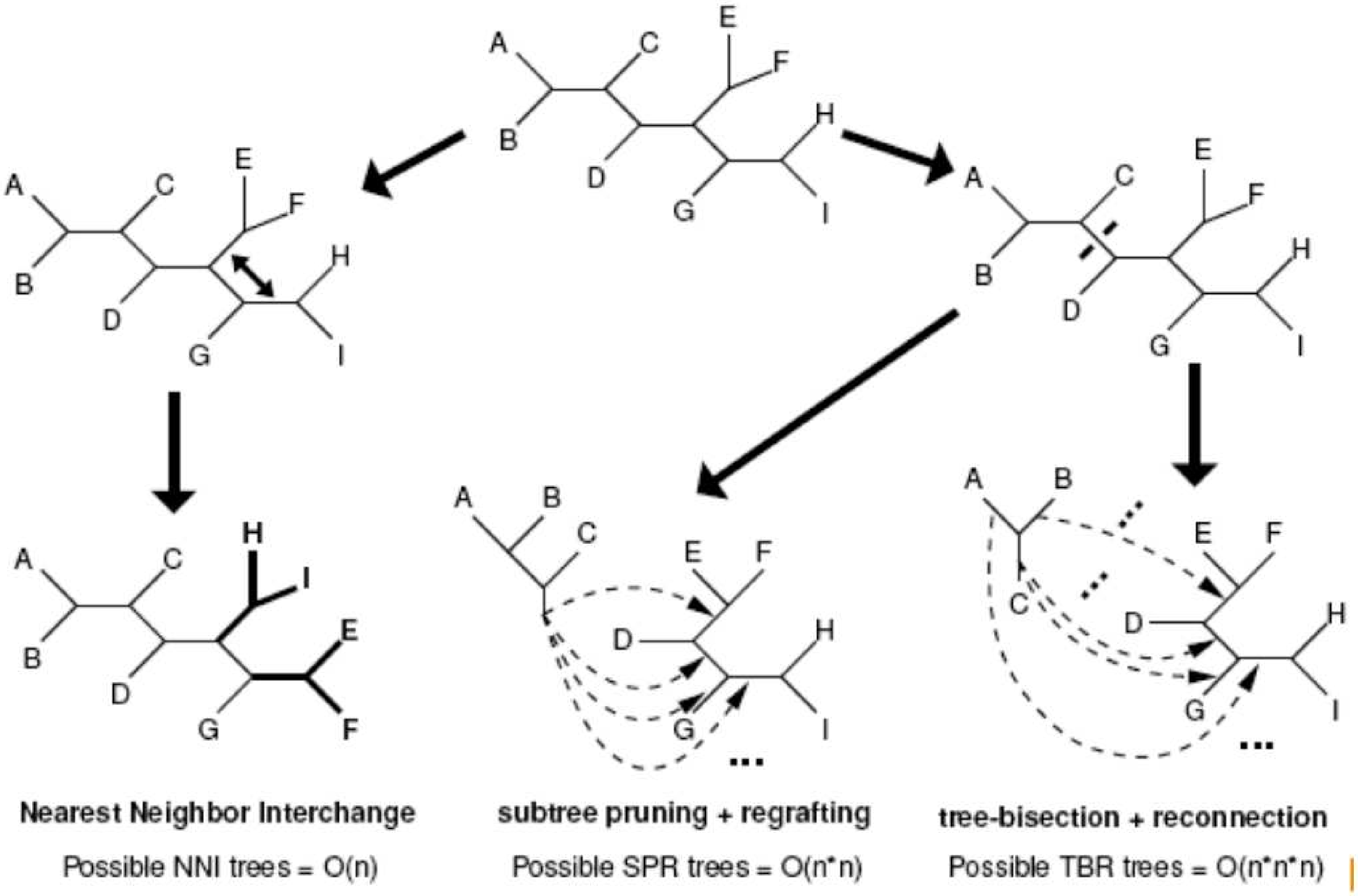
The likelihood $\mathcal{L}(T|\mathcal{D})$ to observe the alignment D under the tree T depends on the model parameters, the edge lengths $\vec{t}$ and the rate matrix elements in Q. In order to compute the likelihood one has to numerically optimize over $\vec{t}$ and the rate matrix Q (for a rate matrix Q with more parameters than the JC69-Q).

A *Maximum Likelihood Tree* $\widehat{\mathcal{T}}$ is the one with the largest likelihood $\mathcal{L}(T|\mathcal{D})$ among all possible tree topologies.

Heuristics to search the tree space

As discussed in the Maximum Parsimony section, the tree space is enormous. If it's not possible to examine all possible tree topologies, heuristic methods to search the tree space are applied.

Start with some 'good' tree (for example a Neighbor Joining tree) ...


A
B
C
D
E
F
G
H
I
...
Nearest Neighbor Interchange
Possible NNI trees = O(n)
subtree pruning + regrafting
Possible SPR trees = O(n*n)
tree-bisection + reconnection
Possible TBR trees = O(n*n*n)

Heuristics to search the tree space, cont'd

A fast and widely used heuristic to reduce the tree search space is *Quartet Puzzling* (Strimmer, v. Haeseler 1996, see also `http://www.tree-puzzle.de/`). The optimal tree for all subsets of sequences consisting only of four sequences (=quartet) is computed. Subsequently, the quartet trees are combined into a larger tree for all sequences.

Note that heuristics may get stuck in local optima of the likelihood landscape. The heuristic tree search procedure possibly should be repeated several times (with different initializations or starting points).

Evolutionary Markov processes

Müller and Vingron (2000) have summarized the properties of a Markov process being that describes the substitution process at a site of a molecular sequence. A π–EMP has the following properties:

- (X_t) is time homogeneous.

  $P_{ij}(t) = \text{Prob}[X_{s+t} = j | X_s = i] = \text{Prob}[X_t = j | X_0 = i]$.

- (X_t) is stationary w.r.t. π.
  $\pi_j = \sum_i \pi_i P_{ij}(t), \ \pi = \pi P(t) \quad \forall \ t$.

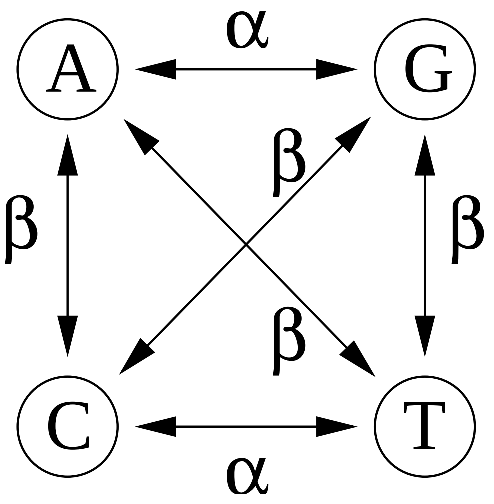
- (X_t) is reversible. $\pi_i P_{ij}(t) = \pi_j P_{ji}(t)$.

Evolutionary Markov processes, cont'd

The assumptions of the Jukes-Cantor model for the evolution of a DNA sequence are simplistic regarding substitution rates and the stationary distribution.

The *Kimura 2-parameter model* takes into account that transitions ($A \leftrightarrow G$ and $C \leftrightarrow T$) are more frequently observed than transversions.

$$Q_{\mathsf{K2P}} = \begin{pmatrix} . & \alpha & \beta & \beta \\ \alpha & . & \beta & \beta \\ \beta & \beta & . & \alpha \\ \beta & \beta & \alpha & . \end{pmatrix}$$



Normally, the ML estimate $\widehat{\alpha}$ is larger than $\widehat{\beta}$.

The stationary distribution π is still the uniform distribution..

Evolutionary Markov processes, cont'd

The *Felsenstein 81 model* has one parameter for a substitution rate, but three parameters for a non-uniform nucleotide distribution:

$$Q_{F81} = \begin{pmatrix} . & \pi_C & \pi_A & \pi_G \\ \pi_T & . & \pi_A & \pi_G \\ \pi_T & \pi_C & . & \pi_G \\ \pi_T & \pi_C & \pi_A & . \end{pmatrix}$$

The *GTR* model is the most general time reversible model for nucleotide sequence evolution with 9 parameters (if one does not care about calibration 8 parameters)

$$Q_{\mathsf{GTR}} = \begin{pmatrix} . & \alpha\pi_C & \beta\pi_A & \gamma\pi_G \\ \alpha\pi_T & . & \delta\pi_A & \epsilon\pi_G \\ \beta\pi_T & \delta\pi_C & . & \pi_G \\ \gamma\pi_T & \epsilon\pi_C & \pi_A & . \end{pmatrix}$$

Empirical models of amino acid evolution

The number of model parameters specifying transitions between amino acids amounts to 209. This large number of parameters cannot be estimated from a single alignment of homologous amino acid sequences. Therefore the empirical approach has become generally accepted. The rate matrix is estimated by considering a large set of aligned sequences from a database and the obtained fixed parameter set is supposed to apply to other datasets.

Dayhoff proposed her pioneering and prominent model of amino acid replacement in the 1970ies from which she derived the PAM family of amino acid similarity matrices. The model is based on global alignments of closely related sequences and the reconstruction of phylogenetic trees followed by the estimation of ancestral sequences. Within the trees she counts the frequency of residues and residue pairs which are used to set up the 1-step transition matrix $P(1)$ of a time-discrete Markov chain. Transition matrices for larger evolutionary distances are obtained from multiples of $P(1)$, for example $P(250) = P(1)^{250}$, that is by extrapolating the observed replacement frequencies between close sequences.

Empirical models of amino acid evolution, cont'd

Similarity scores in the PAM similarity matrices for pairs of amino acids (i, j) are defined as a log likelihood ratio. For example, in the PAM250 similarity matrix,

$$S_{ij}(250) := \log \frac{\pi_i P_{ij}(250)}{\pi_i \pi_j}$$

The nominator is the probability that the residues have diverged from an ancestral residue according to Dayhoff's evolutionary model. The denominator is the probability to observe two residues by chance. The score is positive if the pair (i, j) frequently occurs in the alignments that were used to estimate transition probabilities of the Markov model.

Other empirical models of amino acid evolution are the VT models of Müller and Vingron (2000) and the WAG model of Wheelan and Goldman (2001).

Maximum Likelihood vs. Maximum Parsimony

- Compared to parsimony, Markov models take all possible evolutions into account (there is a small probability for each possible evolution)

- MP trees and ML trees are the same for well conserved alignments, that is, if the probability of change is very small

- We can estimate the variance of real valued parameters with ML

- One can test evolutionary hypothesis with Likelihood Ratio Tests and ask questions like:
  - Did the sequences evolve like a molecular clock and can thus be used to infer divergence times (in physical time units) ?
  - Were the substitution rates different for different nucleotide pairs?
  - Was some gene subject to positive selection in some lineage?

Summarizing probabilistic methods, keywords to remember:

- Time-continuous Markov Models:
  - stationary distribution
  - reversibility (detailed balance eq.)
  - rate matrix exponential
- Likelihood concept and Likelihood as objective function
- Jukes-Cantor correction
- Maximum Likelihood trees
- PAM matrices: the one-step transition probability matrix and the PAM series of similarity matrices