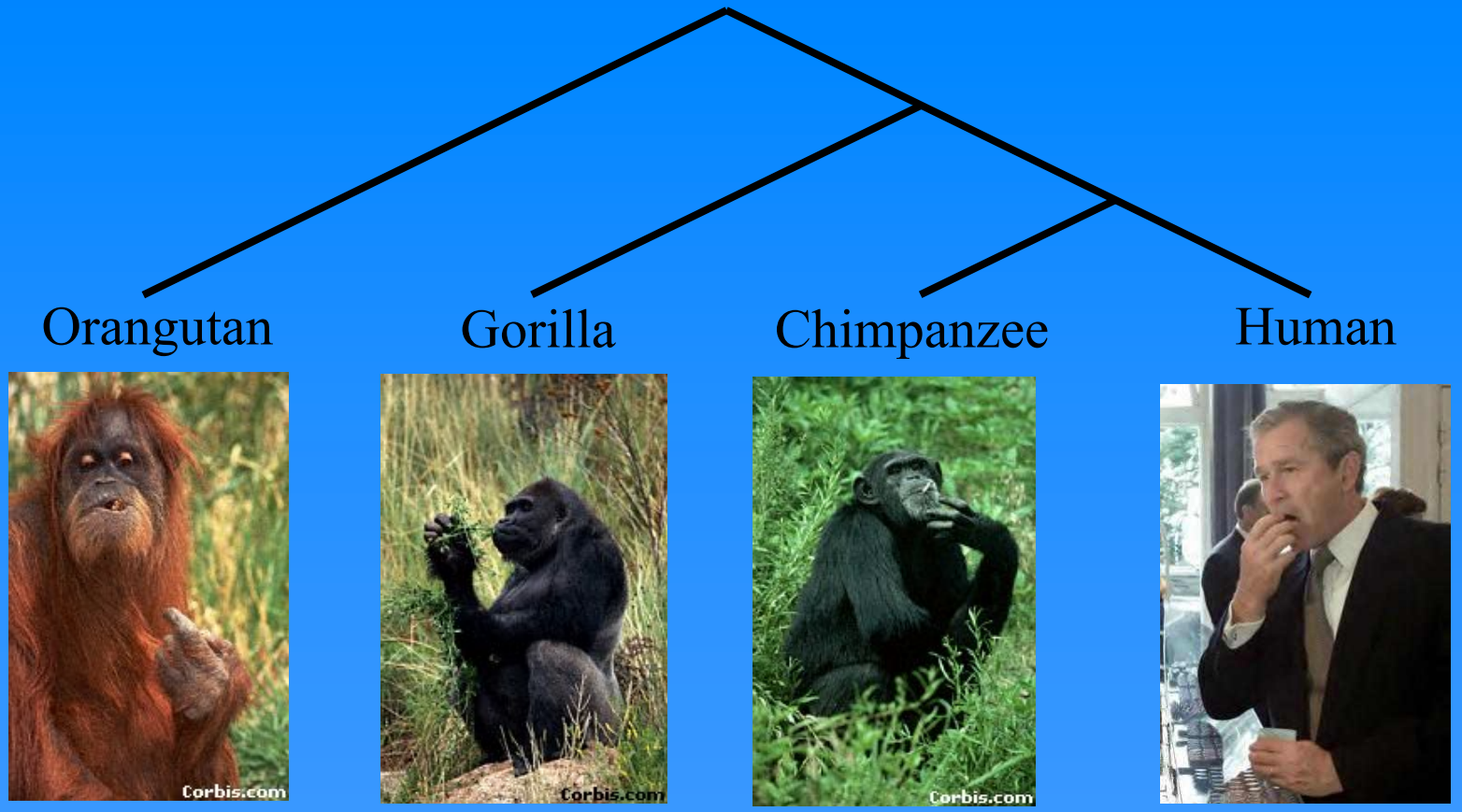
Introduction to Phylogenetics II



From the Tree of the Life Website, University of Arizona

Sagi Snir

Dept. of Evol. Env. Biol. and The Inst. of Evolution,

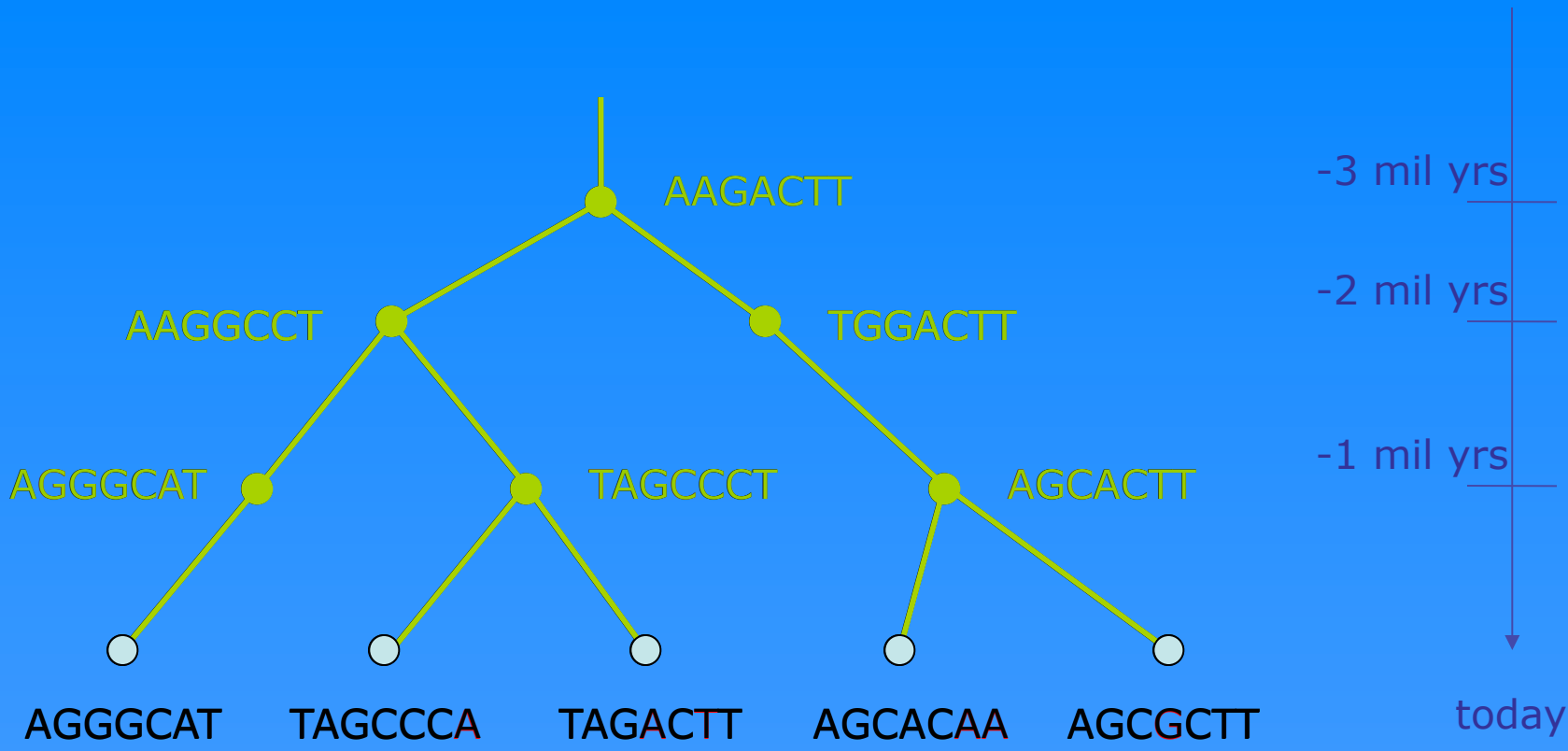
University of Haifa

Introduction to Phylogenetics II

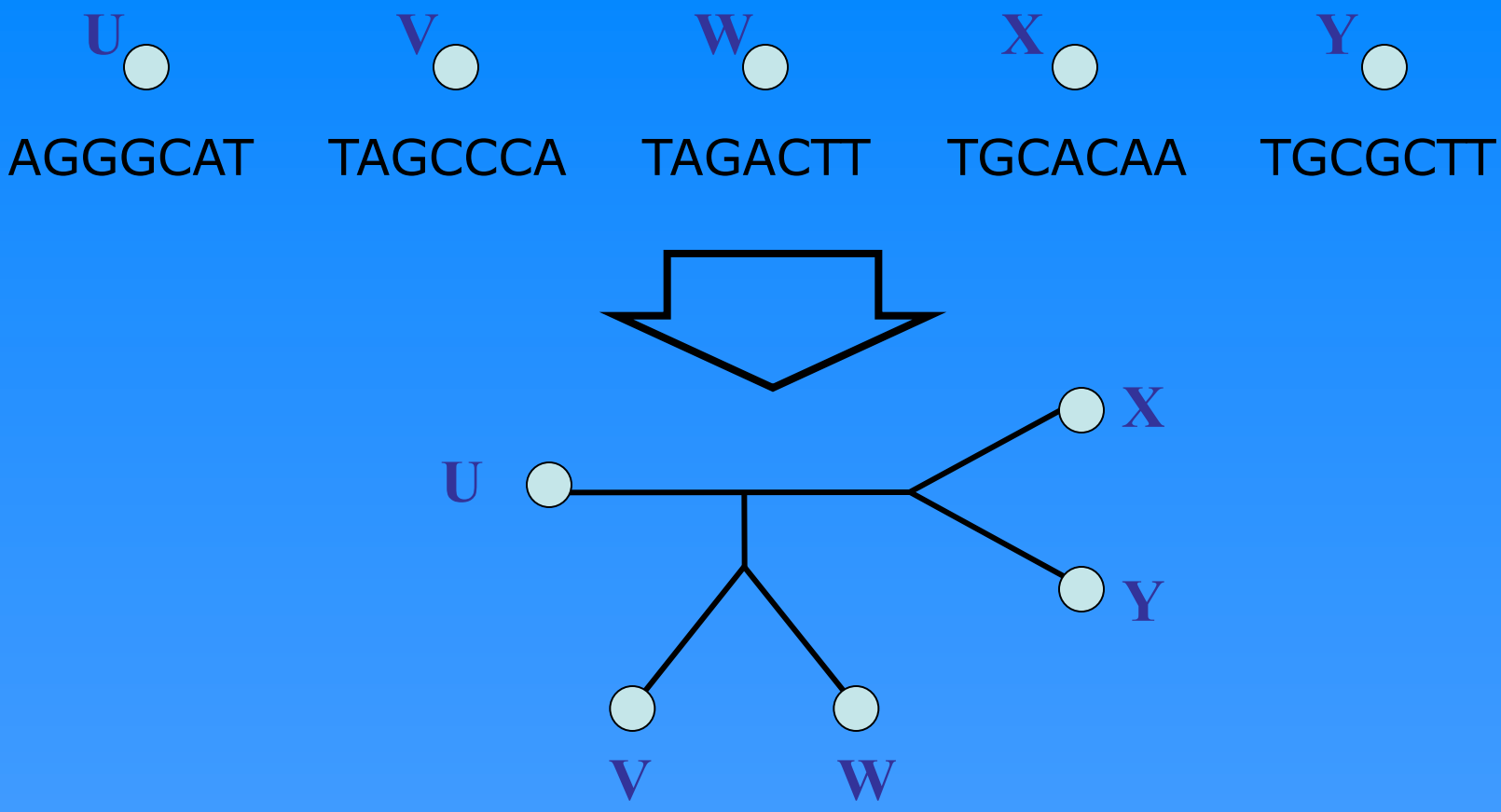
Character based methods – Maximum Parsimony

- Sequence based methods:
  - –Two main categories:
    - *Character based methods*: Trees are constructed by comparing the characters of the corresponding sequences. Characters are mainly molecular (nucleotides in homologous DNA).
    - *Distance based methods*: Input is a square symmetric distance matrix. Seeks trees (edge-weighted) best-describing these distances.
- Supertree methods:
  - –Construct small (reliable) trees form any data and combine it to a complete tree by combinatorial algorithms.
  - –Quartet based methods.

Sequence Evolution
(substantially simplified)



Reconstructing the Tree



Unrooted trees!

Character Based Methods

species	C_1	C_2	C_3	C_4	…												C_m
dog	A	A	C	A	G	G	T	C	T	T	C	G	A	G	G	C	C	C
horse	A	A	C	A	G	G	C	C	T	A	T	G	A	G	A	C	C	C
frog	A	A	C	A	G	G	T	C	T	T	T	G	A	G	T	C	C	C
human	A	A	C	A	G	G	T	C	T	T	T	G	A	T	G	A	C	C
pig	A	A	C	A	G	T	T	C	T	T	C	G	A	T	G	G	C	C
	*	*	*	*	*			*	*			*	*				*	*

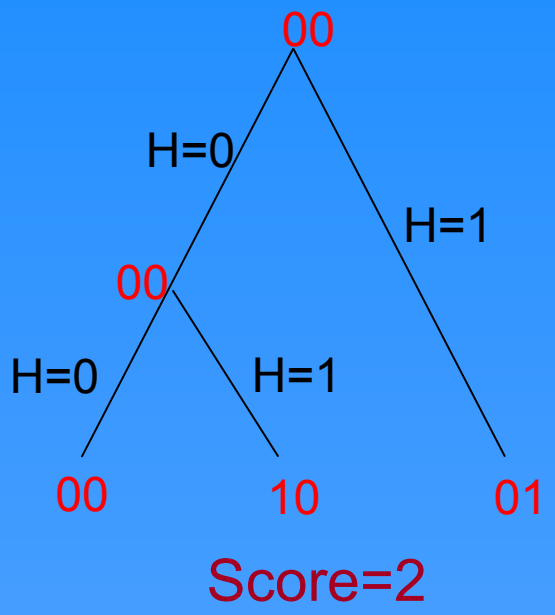
1. Input: A n*m matrix.
2. Each character (column) is processed independently.
3. Task: Find a tree that best explains simultaneously all characters.

Maximum Parsimony

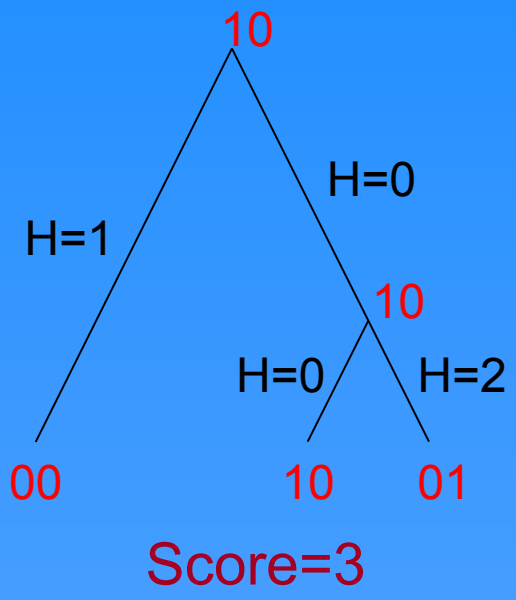
- Introduced at 1971 by Walter Fitch: *Fitch, Toward defining the course of evolution: minimum change for a specific tree topology,* 1971.
- based on the "*Occam's Razor*" principle that evolution is parsimonious.
- A combinatorial, non-parametric method.
- Seeks for the tree that minimizes the number of changes along the tree branches.
- A very widespread technique in biology.
- “If you know only one method for phylogenetics, MP should be the one”

The Parsimony Criterion on Trees

- Given a tree (topology) with *equal length* sequences labeling its nodes.
- The *parsimony score*: The number of changes along the edges of the tree.
- Can be thought of as the sum of *Hamming distances* along the edges.



Score=2



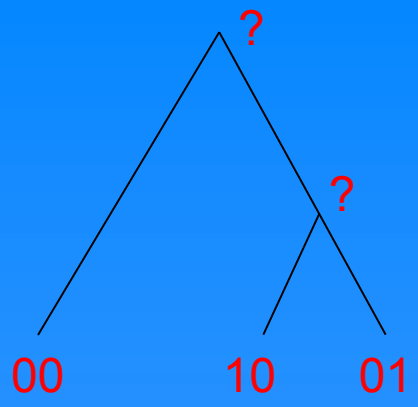
Score=3

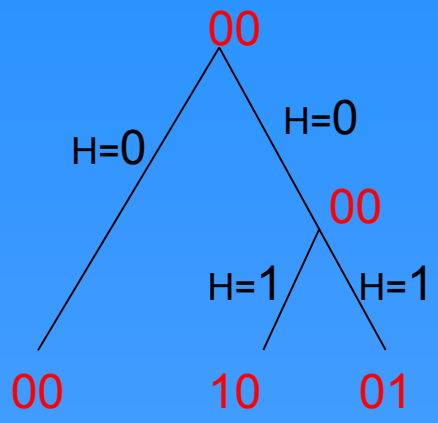
The *Maximum Parsimony* Problem

- Input: A set of sequences (representing some *gene* at a group of species).
- Task: Find a topology leaf labeled by the input sequences, and labeling to internal nodes *minimizing the parsimony score*.
- Decomposes into two problems:
  - A *Small Problem*: Given a topology leaf labeled by a set of sequences, find *internal nodes labeling* minimizing the parsimony score.
  - A *Big Problem*: Find a topology under which the small problem is minimized.

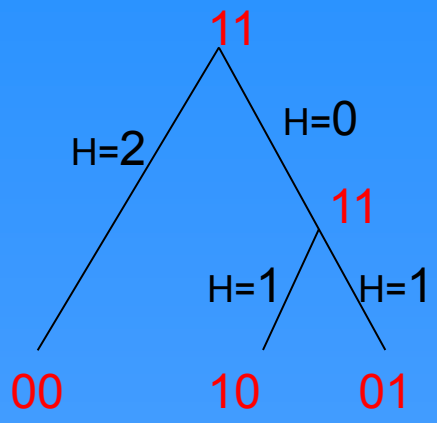
The *Small* MP problem on Trees

Task: finding internal labeling.

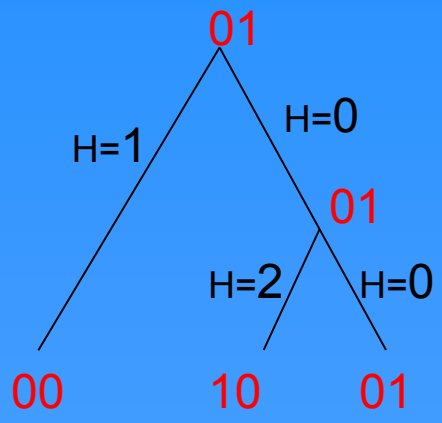




Score=2



Score=4



Score=3

Fitch Algorithm for Small MP on Trees

- A classical DP style algorithm.
- Works separately on each column.

First Phase: bottom up (v_1 and v_2 are children of v):

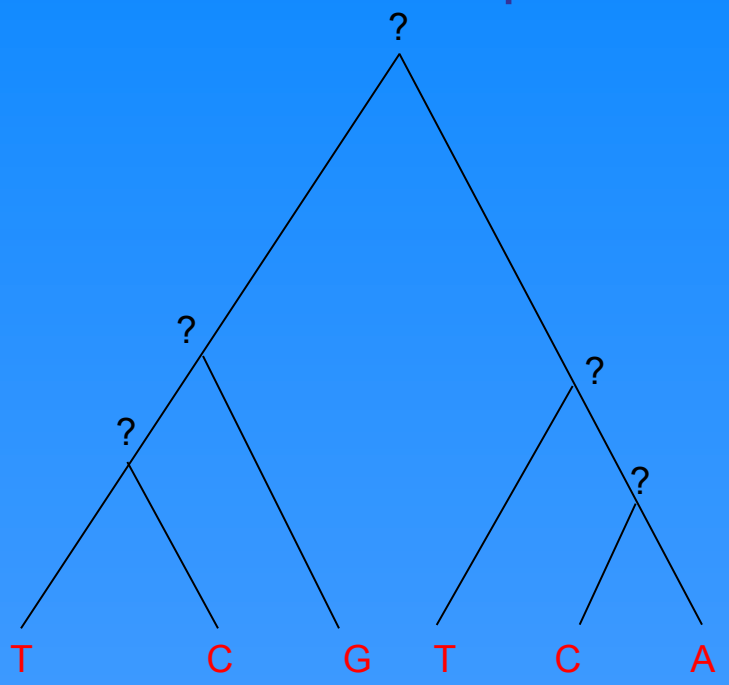
$$A(v) = \begin{cases} A(v_1) \cap A(v_2) & \text{if } A(v_1) \cap A(v_2) \neq \phi \\ A(v_1) \cup A(v_2) & \text{otherwise} \end{cases}$$

Second Phase: top down ($f(v)$ is a parent of v):

$$B(v) = \begin{cases} \sigma \in A(v) \cap A(f(v)) & \text{if } A(v) \cap A(f(v)) \neq \phi \\ \sigma \in A(v) & \text{otherwise} \end{cases}$$

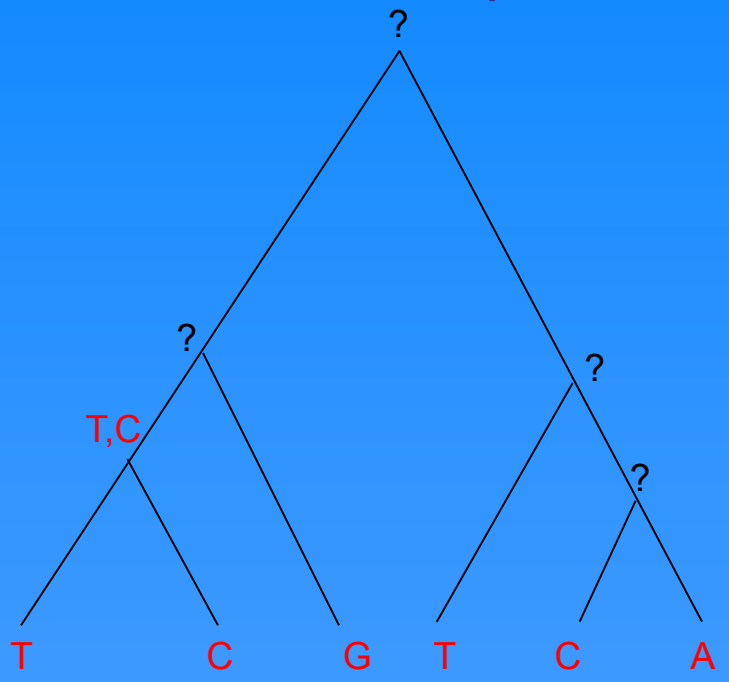
Fitch Algorithm (example)

Input:

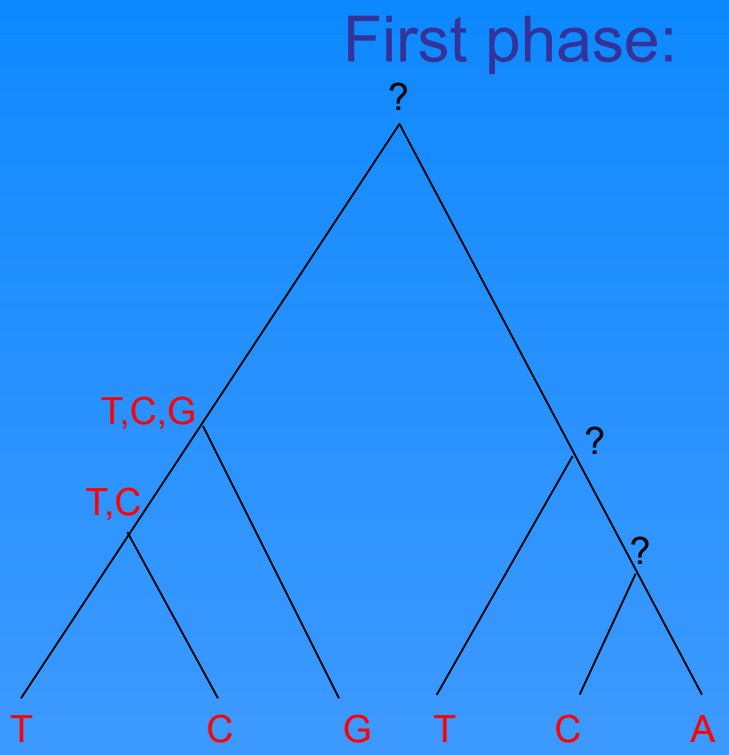


Fitch Algorithm (example)

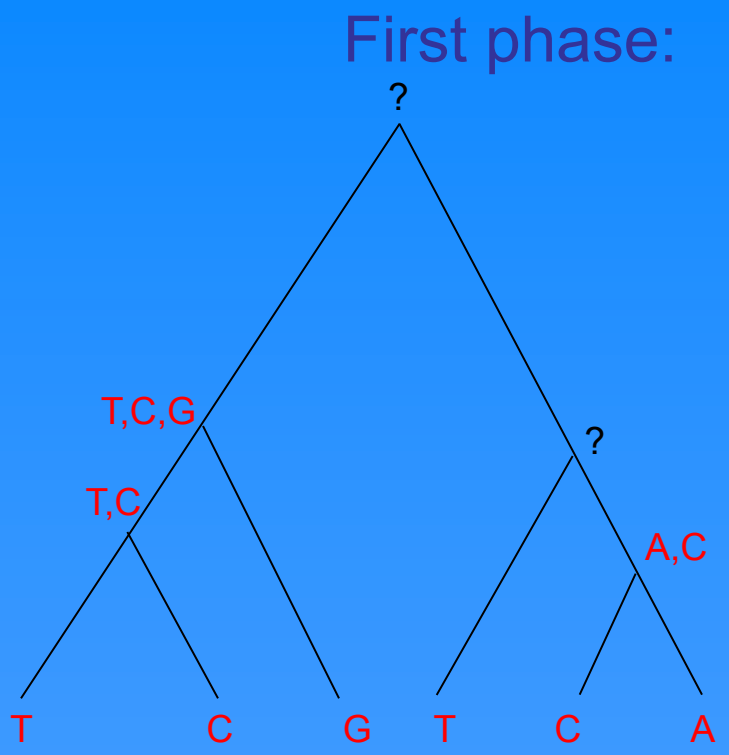
First phase:



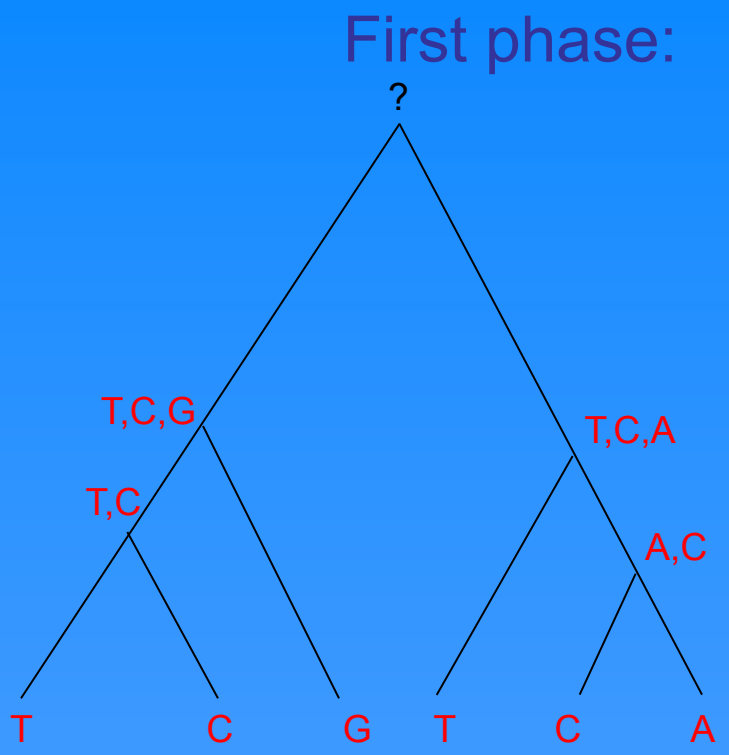
Fitch Algorithm (example)



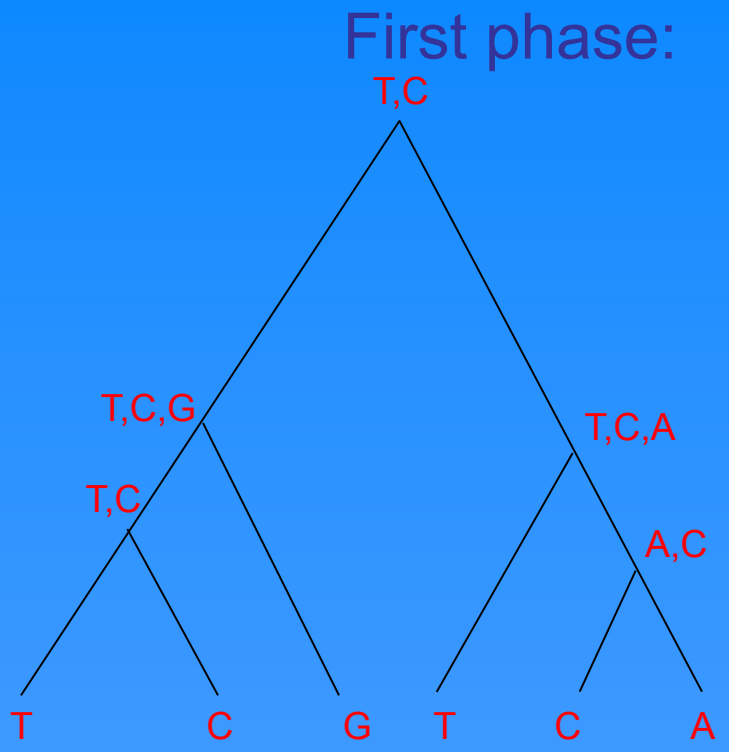
Fitch Algorithm (example)



Fitch Algorithm (example)

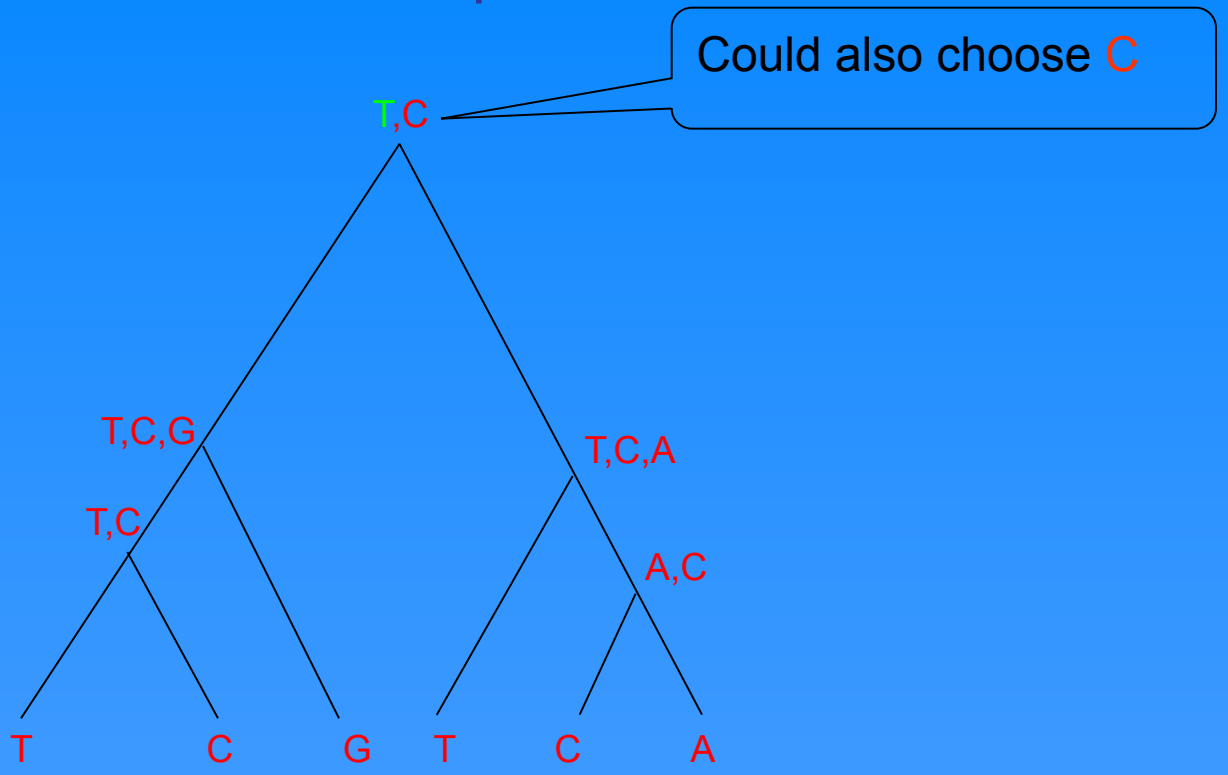


Fitch Algorithm (example)



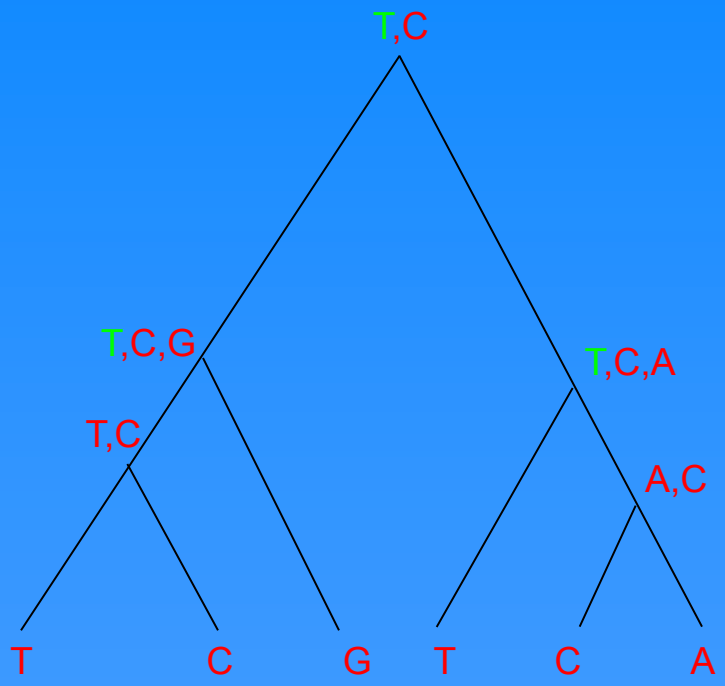
Fitch Algorithm (example)

Second phase:



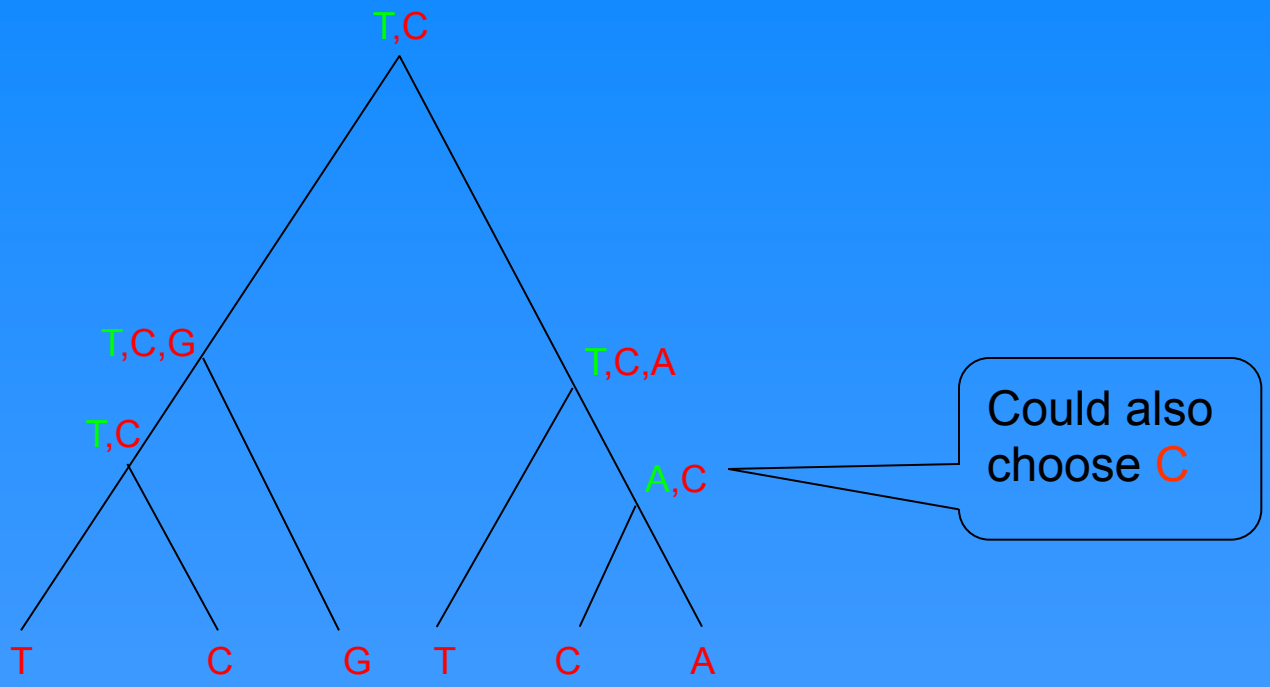
Fitch Algorithm (example)

Second phase:



Fitch Algorithm (example)

Second phase:



Fitch Algorithm

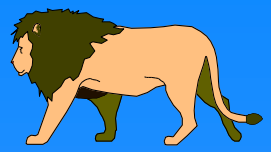
Claim: Fitch algorithm solves small MP

- Optimality for single character (simple induction).
- Global optimum (change summation order).

The *Big* MP problem on Trees

- Input: A set of sequences.
- Task: Find a topology over the sequences under which the *small problem minimizes*.

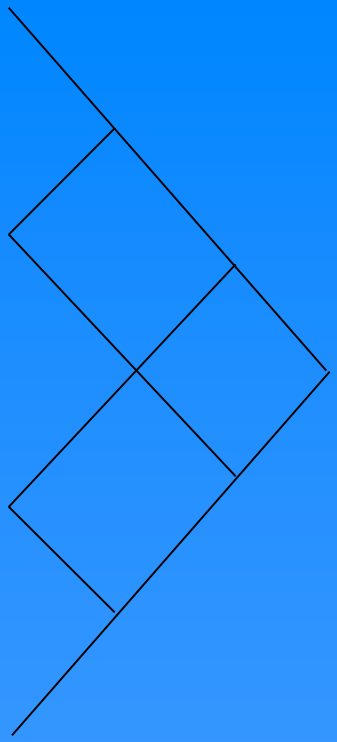
0001111



0000000

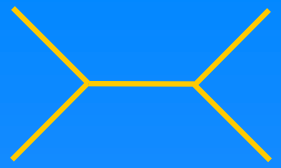
1110000

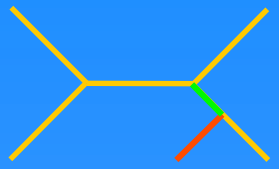
1111111

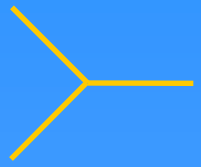


Number of trees

- Binary unrooted trees
- First cuont number of edges
- Divide into
  - External edges (always *n*).
  - Internal edges.
- New taxons are always added in a middle of an existing edge.
- Observation: adding a taxon splits an existing edge, creating a new internal edge.
- Summarizing: adding a new taxon adds 2 edges – the new internal edge and the external edge leading to that taxon.
- As for n=3 we have no internal edges, we obtain $|E_n| = 2n-3$

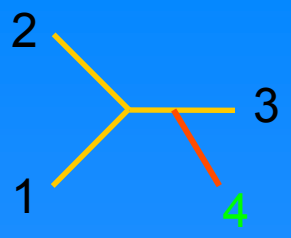






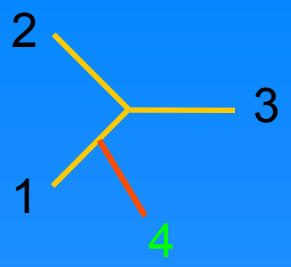
Number of trees

- Back to number of trees, N_u.
- For n=3 we have $N_u(3) = 1$
- We can insert the new taxon 4 on any existing edge



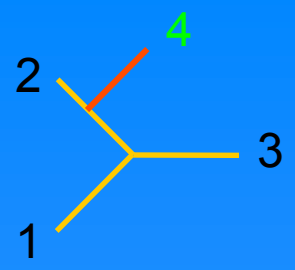
Number of trees

- Back to nomber of trees, N_u.
- For n=3 we have $N_u(3) = 1$
- We can insert the new taxon 4 on any existing edge



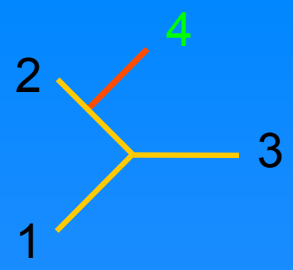
Number of trees

- Back to nomber of trees, N_u.
- For n=3 we have $N_u(3) = 1$
- We can insert the new taxon 4 on any existing edge



Number of trees

- Back to nomber of trees, N_u.
- For n=3 we have $N_u(3) = 1$
- We can insert the new taxon 4 on any existing edge
- Therefore we get
  - $N_u(n+1) = N_u(n)\ |E_n| = N_u(n)\ (2n-3)$
  - Or $N_u(n+1)/N_u(n) = 2n-3$
  - Or $N_u(n)/N_u(n-1) = 2n-5$



$$\frac{(2n-5)!}{2*4*6*8*10} = \frac{(2n-5)!}{\prod_{n-3} 2i} = \frac{(2n-5)!}{2^{n-3}(n-3)!}$$

3	1
4	1*3
5	1*3*5
6	1*3*5*7
7	1*3*5*7*9
8	1*3*5*7*9*11

2n - 5

2n - 5

2n - 5

2n - 5

2n - 5

Solving NP-hard problems exactly is … unlikely

- Number of (unrooted) binary trees on *n* leaves is (2n-5)!! = (2n-5)!/[2^{n-3}(n-3)!]
- If each tree on **1000** taxa could be analyzed in **0.001** seconds, we would find the best tree in **2890 millennia**
- NP-hard in general.
- Heuristics use **branch and bound** techniques.

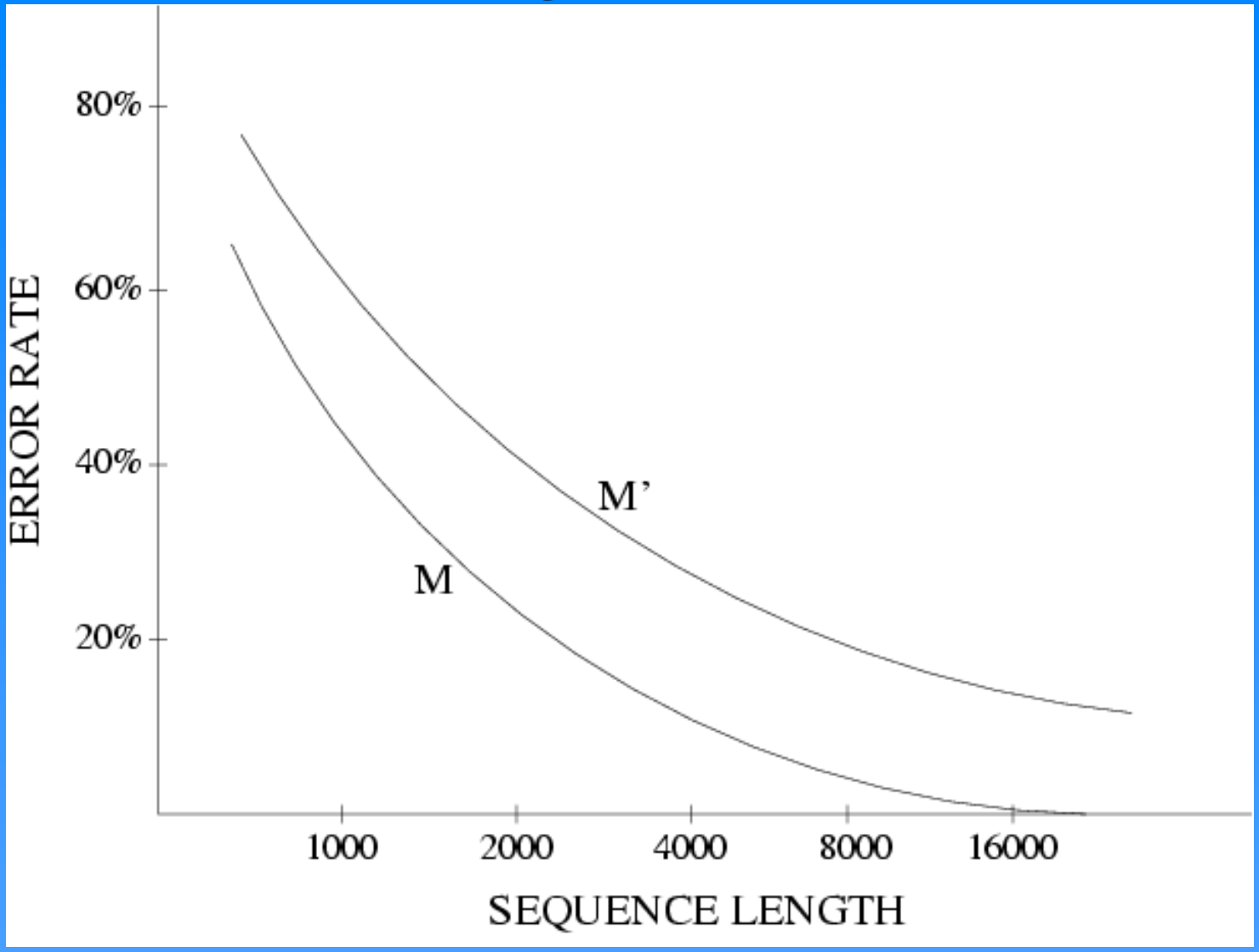
#leaves	#trees
4	3
5	15
6	105
7	945
8	10395
9	135135
10	2027025
20	2.2×10^{20}
100	4.5×10^{190}
1000	2.7×10^{2900}

Statistical performance issues

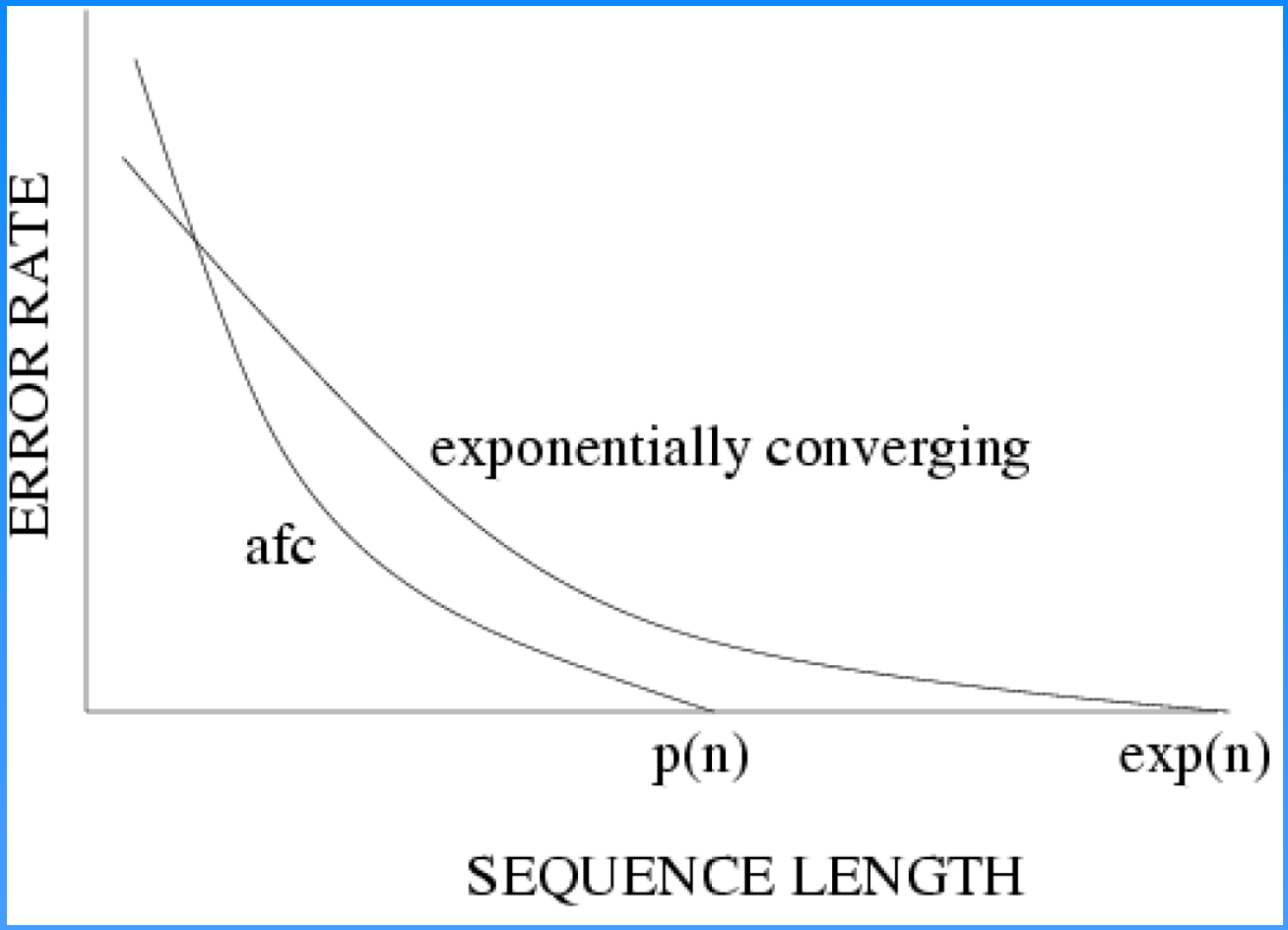
- An estimation method is *statistically consistent* under a model if the probability that the method returns the true tree goes to 1 as the sequence length goes to infinity.

- Convergence rate: the amount of data that a method needs to return the true tree *with high probability*, as a function of the model tree.

Statistical consistency and convergence rates

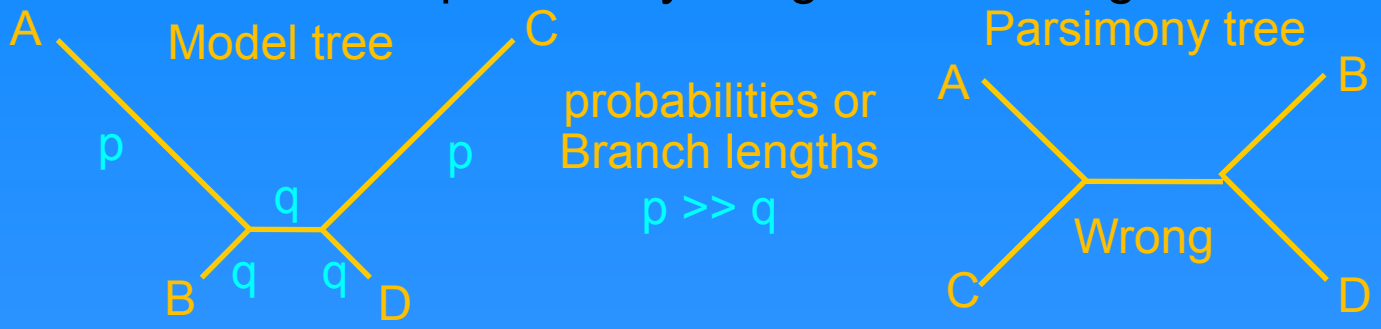


Absolute fast convergence vs. exponential convergence



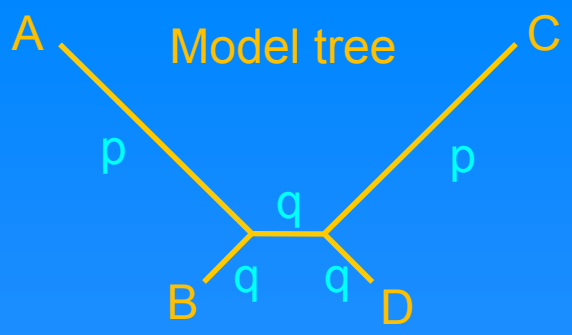
Parsimony can be inconsistent

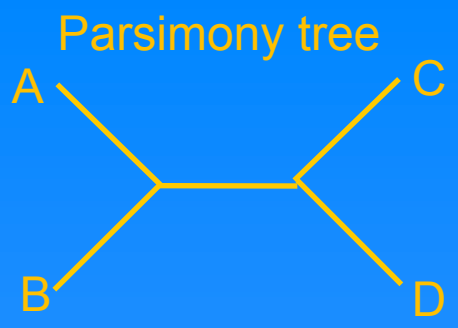
- Felsenstein (1978) developed a simple phylogeny model including four taxa and a mixture of short and long branches, *p* and *q*, indicating low and high substitution probabilities resp.
- Under this model parsimony will give the wrong tree.



- **The assumption is that *p,q* < ½ as these actually derived from *rate of substitutions* and by definition cannot exceed ½.**
- **With more data the certainty that parsimony will give the wrong tree increases - so that parsimony is statistically inconsistent.**

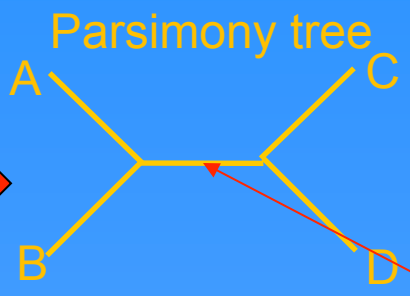
Parsimony can be inconsistent



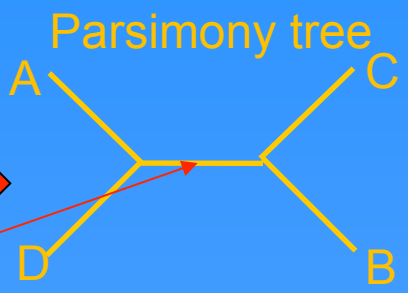


- Parsimony does not care of branch lengths (non parametric).
- Aims to minimize mutations (changes) over branches.
- That means putting together (nearby) taxa with same state.

A	1
B	1
C	0
D	0



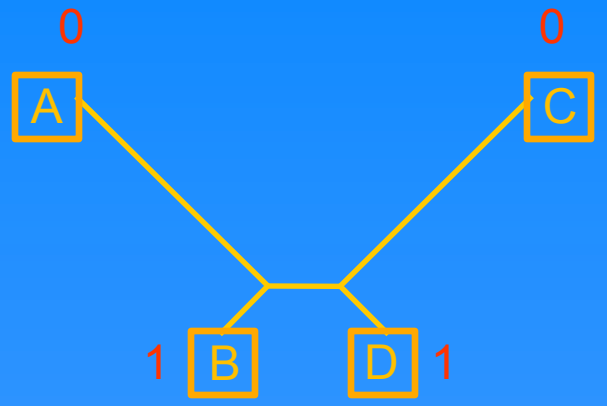
A	1
B	0
C	0
D	1

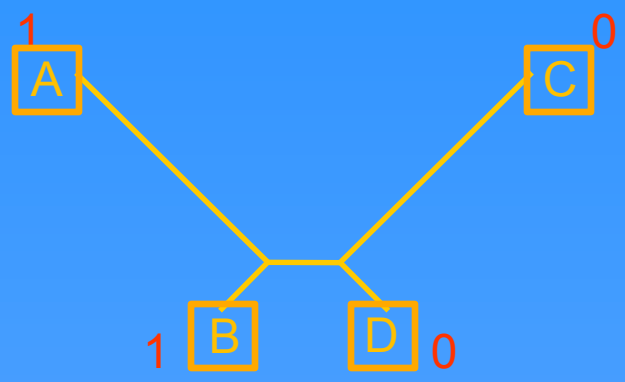


A change from 1 to 0

Parsimony inconsistency

- Our alphaet is {0,1} (can indicate purines/pyrimidines)
- Let C_{xyxy} be the case when leaves A,C get different values than B,D.
- Note that in this case, parsimony will return AC|BD.
- That is, parsimony *errs!*
- Equivalently, define C_{xxyy} to be the case when leaves A,B get different values than C,D.

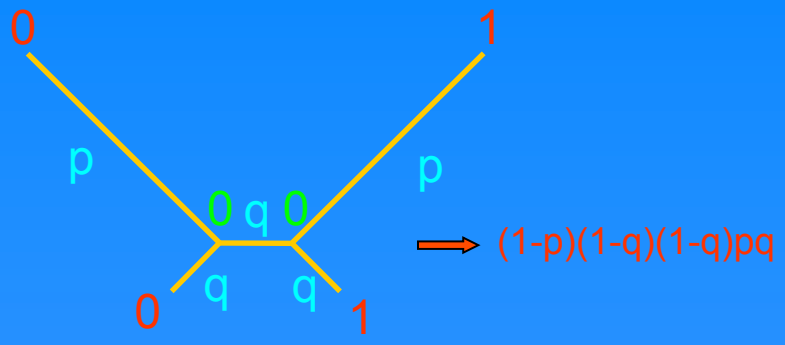




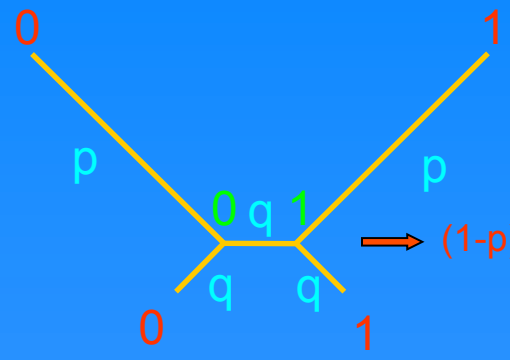
Parsimony inconsistency

- Let us calculate $P(C_{xxyy})$: the probability seeing C_{xxyy} seeing

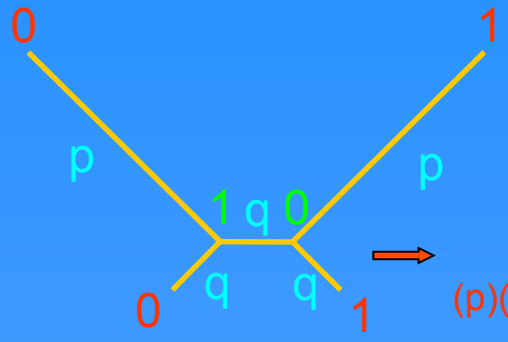
(note, species names were removed, but topology is A,B|C,D)



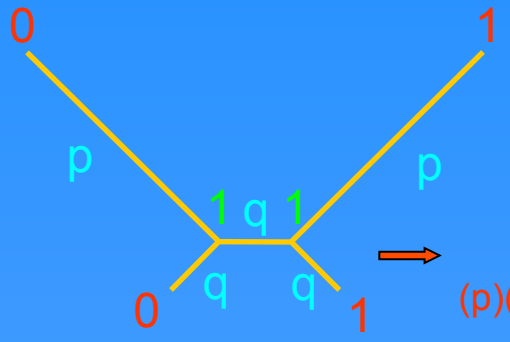
$\rightarrow (1-p)(1-q)(1-q)pq$



$\rightarrow (1-p)(1-q)(q)(1-p)(1-q)$



$\rightarrow (p)(q)(q)(p)(q)$

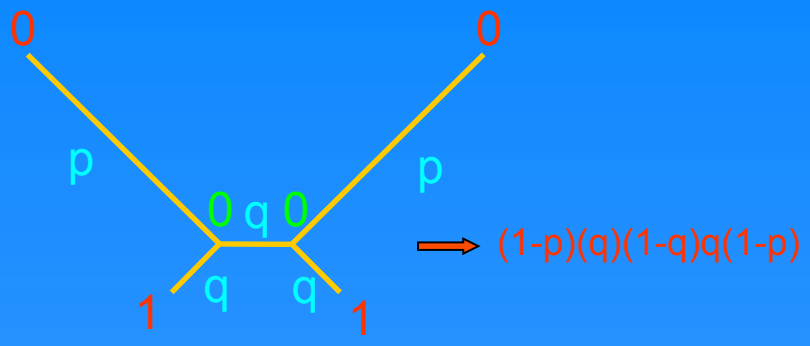


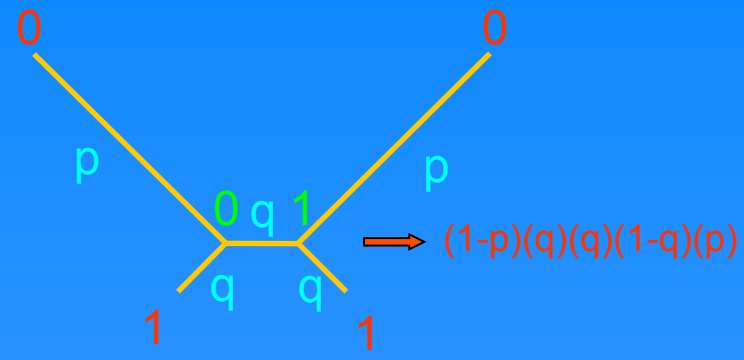
$\rightarrow (p)(q)(1-q)(1-p)(1-q)$

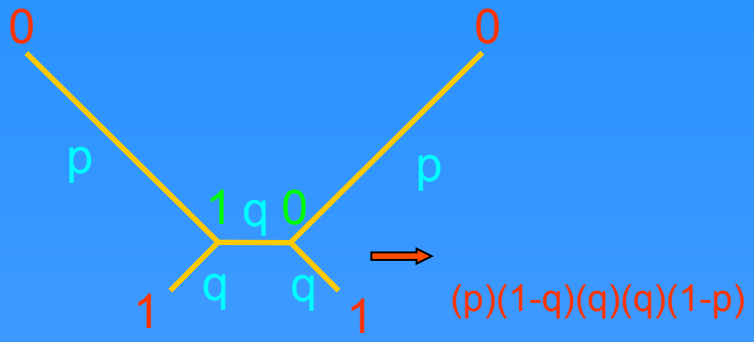
- We get $P(C_{xxyy}) = (1-p)(1-q)^2pq+(1-p)^2(1-q)2q+p^2q^3+(1-p)(1-q)^2pq$

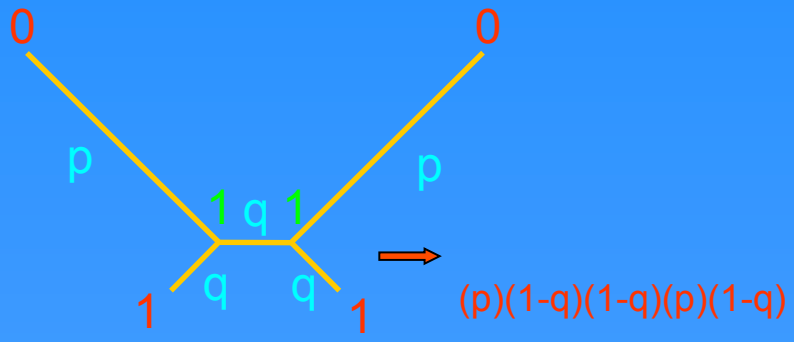
Parsimony inconsistency

- Let us calculate $P(C_{xyxy})$: the probability seeing C_{xyxy}









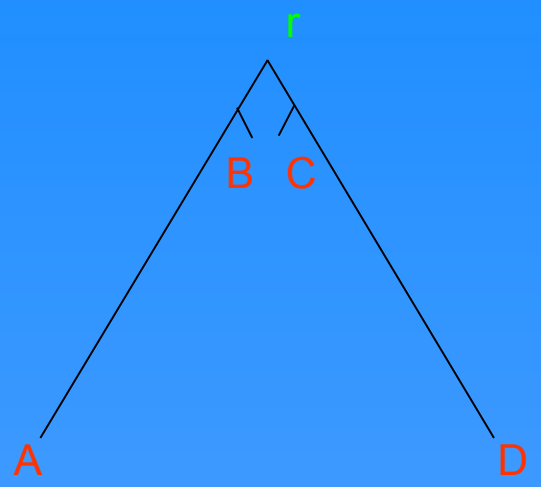
- We get $P(C_{xyxy}) = (1-q)(1-p)^2pq+(1-p)(1-q)pq^2+ (1-q)(1-p)pq^2 +(1-q)^3p^2$

Parsimony inconsistency

1. $P(Cxyxy) - P(Cxxyy) = (1-2q)[q^2(1-p)^2+(1-q)^2p^2]$

2. This is always positive as $q<1/2$.

- More intuitively:
  - B and C will mostly have *r*'s state (no mutation).
  - Whatever states A and D take is either uninformative for mp or misleading.



Long-branch Attraction

- Is all this realistic? *Very much!!!*
- Advocates of parsimony initially responded by claiming that Felsenstein's result showed only that his model was unrealistic.
- It is now recognised that the ***long-branch attraction*** (in the **Felsenstein Zone**) is one of the most serious problems in phylogenetic inference.