Wellcome trust advanced course
"Molecular Evolution"

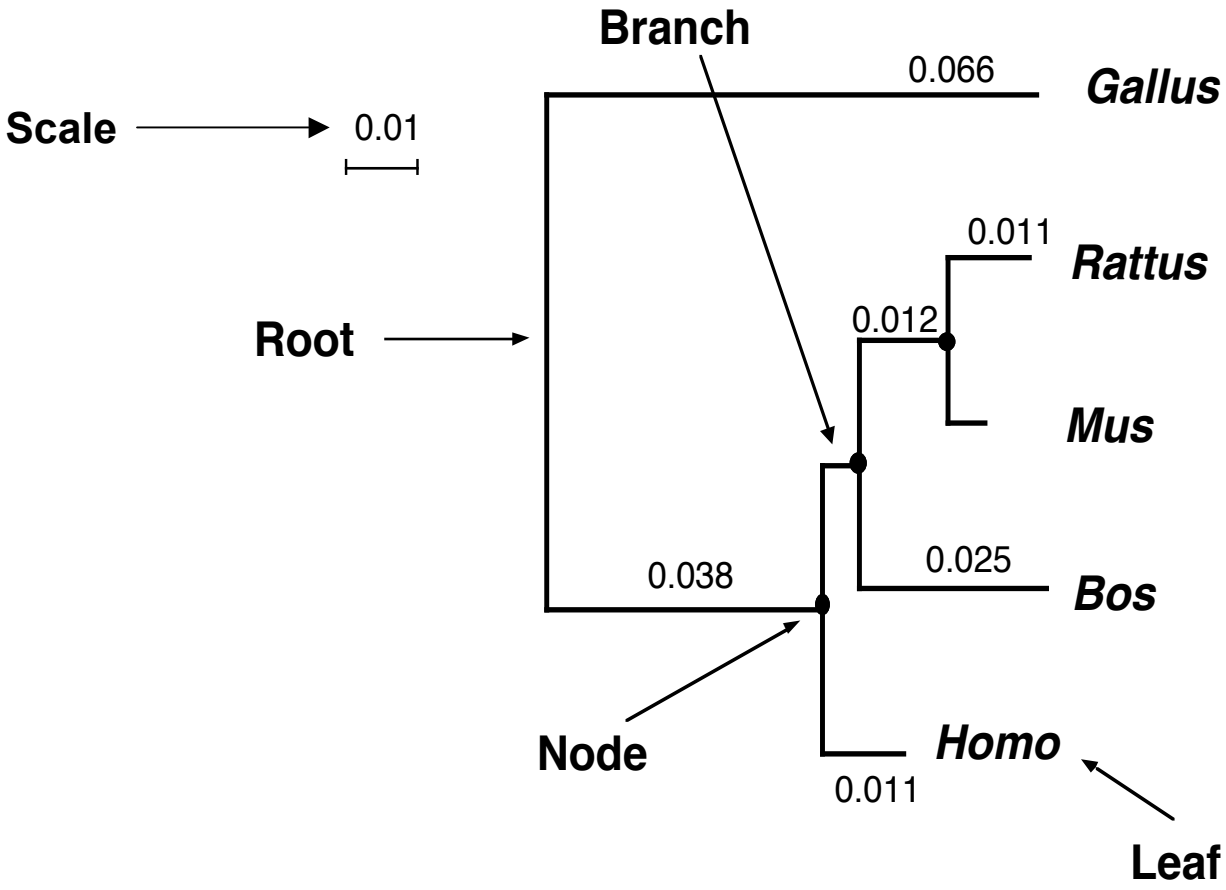
Molecular Phylogenetics with Parsimony

Manolo Gouy
CNRS - Université de Lyon

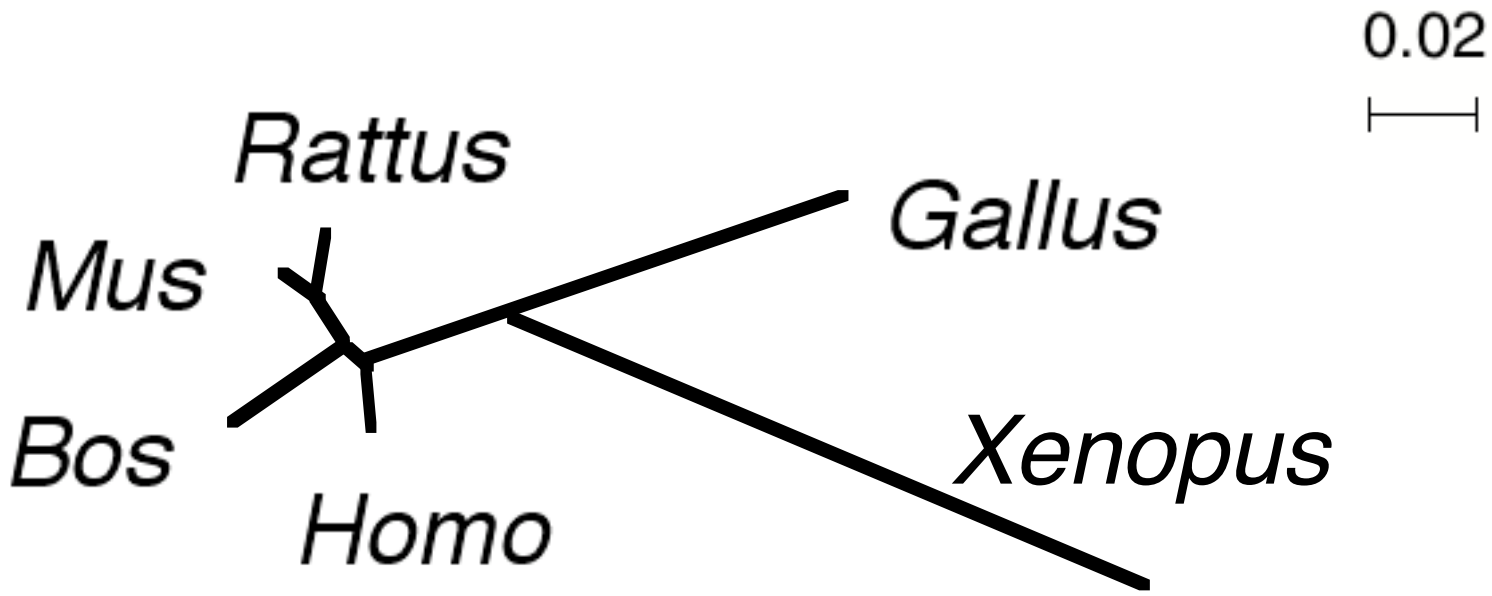
Hinxton 29 March - 9 April 2009

Phylogenetic trees

- Internal Branch: connects 2 nodes. External Branch: connects a node to a leaf
- Lengths of horizontal branches are proportional to evolutionary distances between ancestral or extant sequences (unit = substitution / site).
- Tree Topology = tree shape = node branching order

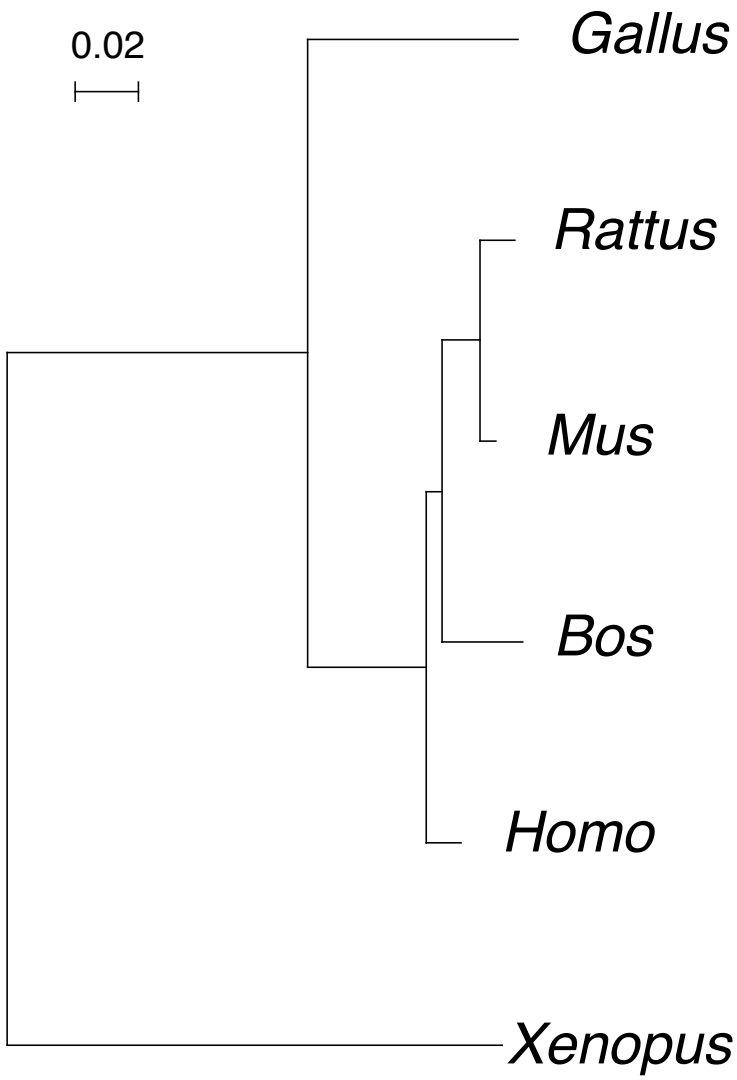


Unrooted tree



Each branch represents molecular evolution between its tips, but the direction in which this evolution occurred is not specified.

Rooted tree



Evolutionary time flows from left to right on each branch.

Number of distinct unrooted tree shapes for n taxa

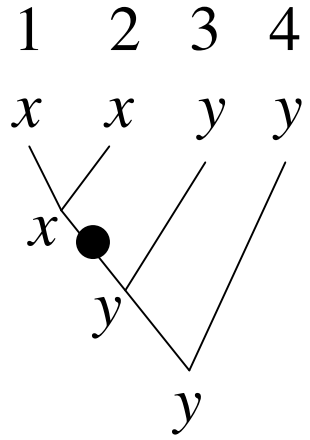
$$N_{trees} = 3.5.7...(2n-5) = \frac{(2n-5)!}{2^{n-3}(n-3)!}$$

n	N_{trees}
3	1
4	3
5	15
6	105
7	945
...	...
10	2,027,025
...	...
20	$\sim 2 \times 10^{20}$

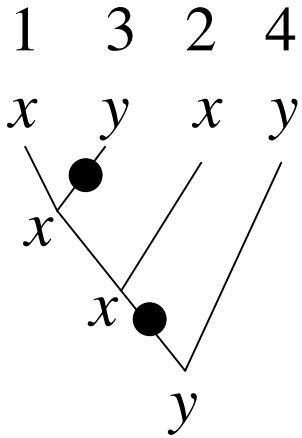
Why parsimony ?

Let us consider a character in 4 species {1, 2, 3, 4} with states {*x*,*x*,*y*,*y*}. What evolutionary history can have led to this final state ?

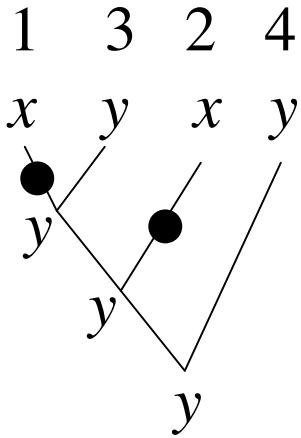
Identity by common descent: two species share the same character state because they inherited it from their last common ancestor without change.



Presence of homoplasy: identical states are observed although they were not inherited, unchanged, from the last ancestor.



reversion



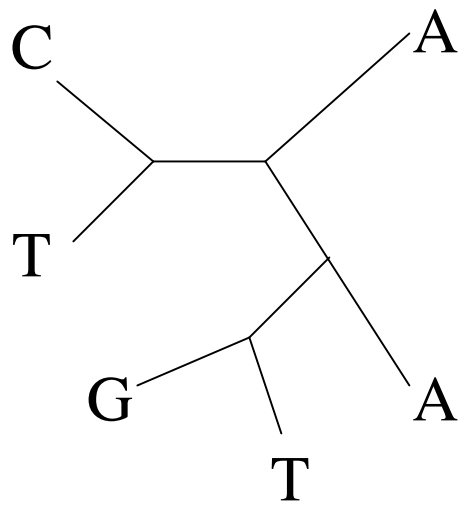
convergence

Scenarios with homoplasy require more evolutionary changes. Parsimony assumes that convergence and reversions are rare and search for the history that minimize these events. Parsimony applies very well if changes are rare.

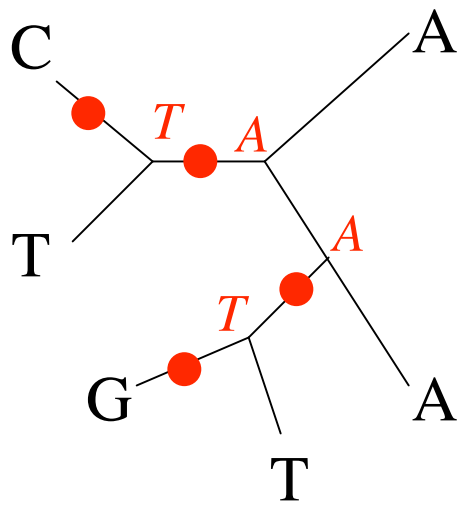
Fitch Algorithm

The data
- a tree shape
- residues at the leaves of this tree

The problem : compute the minimal number of changes in the tree that can explain these data.



4 changes required



How to compute this number (4) exactly and efficiently for any tree size ?

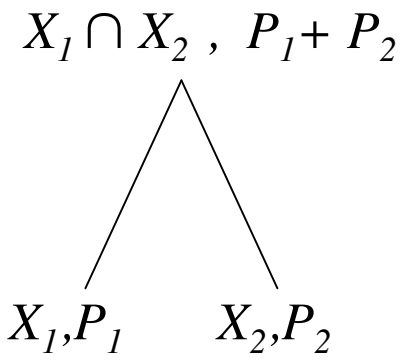
Fitch Algorithm *(recipe)*

Arbitrarily root the tree and recursively compute, at each node, two things:

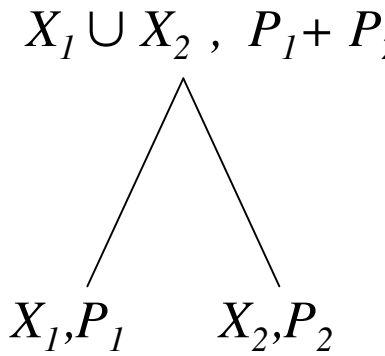
- P: minimal number of changes in the sub-tree rooted by this node
- X: set of residues each equally possible for this node

To go one step up the tree, consider whether the sets X_1, X_2 share common residues.

common residues



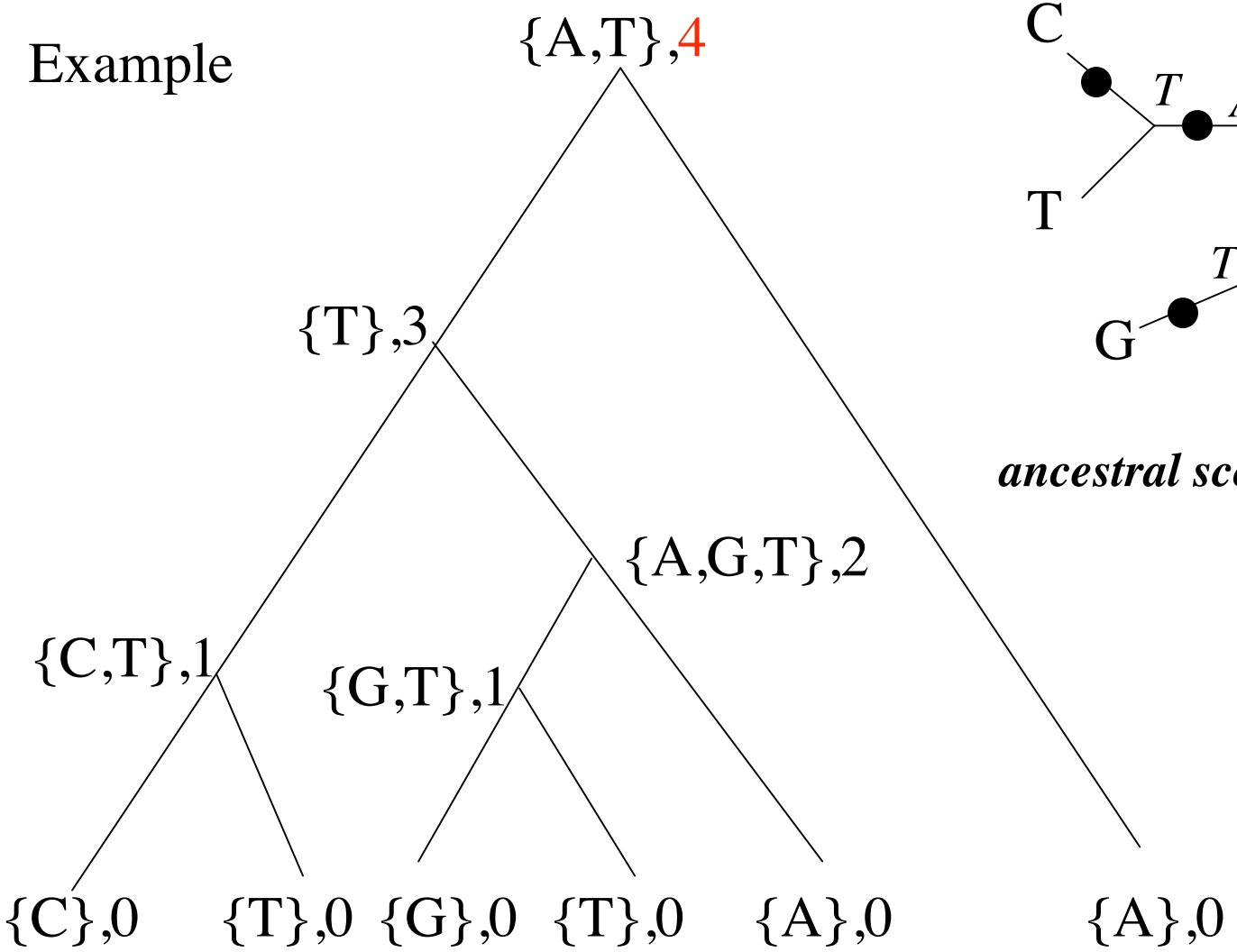
no common residue

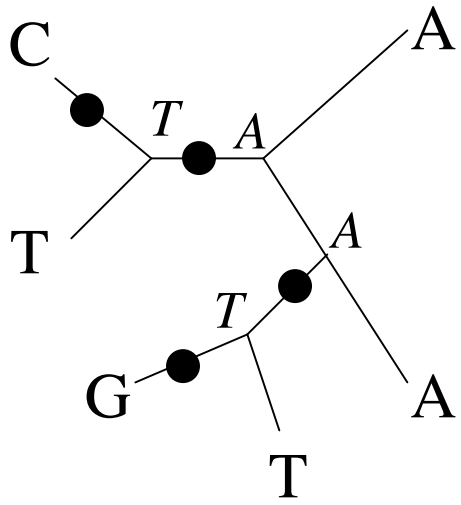


Fitch Algorithm *(example)*

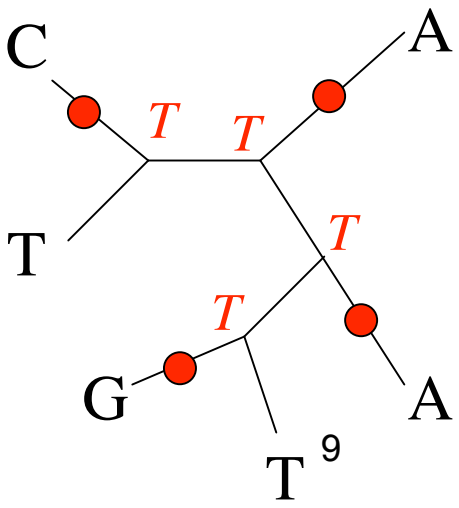
The calculation is initialized at tree leaves with:
X={residue present at this leave}, P = 0

Example





ancestral scenario is not unique !

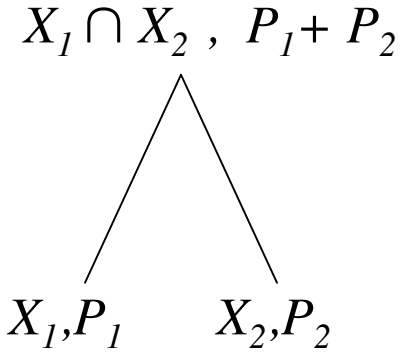


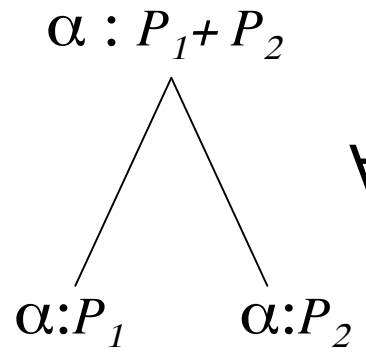
Fitch Algorithm *(proof)*

Arbitrarily root the tree and recursively compute, at each node, two things:

- P: minimal number of changes in the sub-tree rooted by this node
- X: set of residues each equally possible for this node

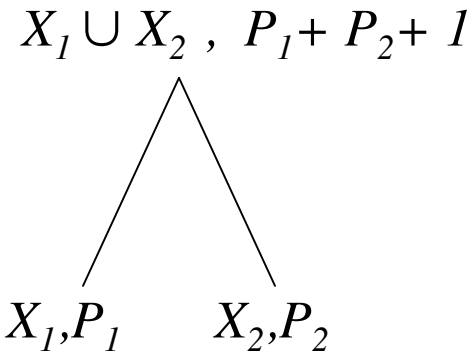
1rt case: $X_1 \cap X_2$ is not empty

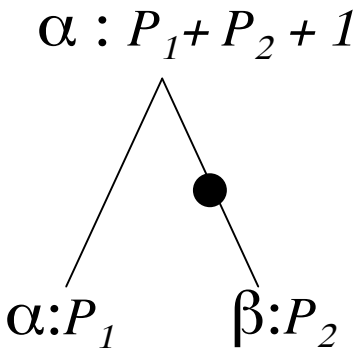


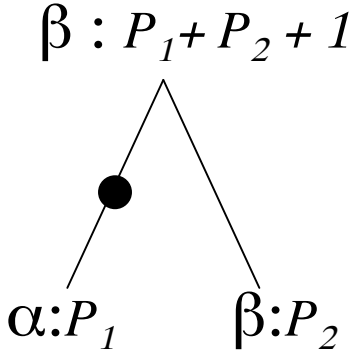


$\forall\ \alpha \in X_1 \cap X_2$

2nd case: $X_1 \cap X_2$ is empty





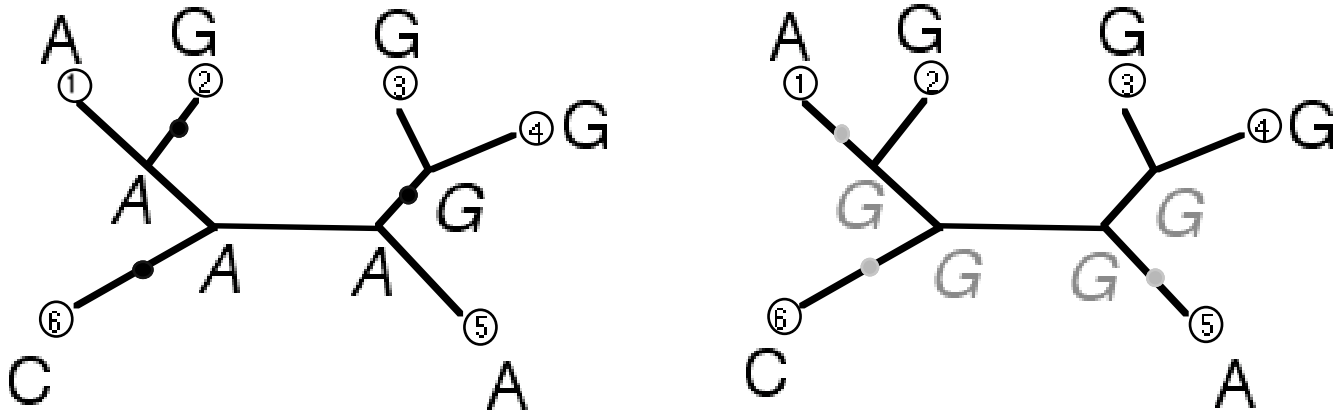


$\forall\ \alpha \in X_1$,
$\forall\ \beta \in X_2$

Parsimony (1)

- Step 1:

For a given tree topology and a given alignment site, put site residues at the leaves of this tree. Then, use Fitch algorithm to compute *d*, the smallest total number of changes in the tree.



X: ancestral nucleotide

• : substitution event

Example: For this site and this tree shape, at least 3 changes are necessary to explain the pattern of nucleotides present at tree leaves. Several distinct scenarios are possible.

Parsimony (2)

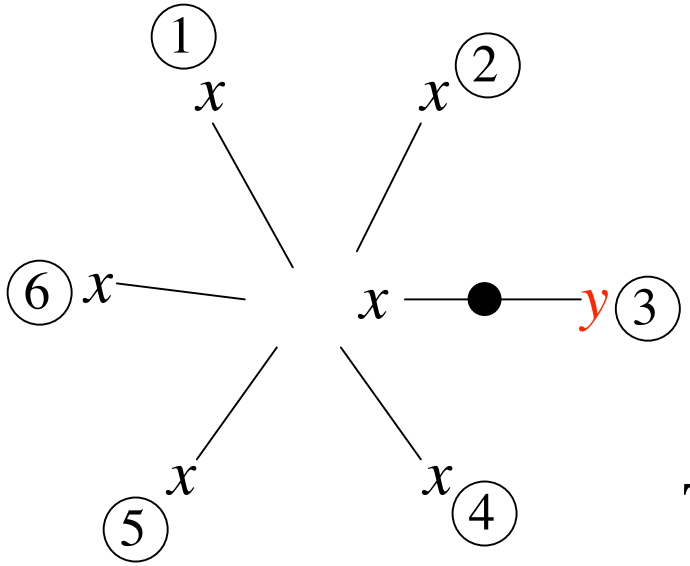
- Step 2:
  - Compute *d* (step 1) for each alignment site
  - Sum up *d* values for all sites
  - This gives the length *L* of the tree

- Step 3:
  - Compute value *L* (step 2) for all possible tree shapes.
  - Retain the shortest tree (*i.e.*, with smallest *L* value)

    = the tree(s) requiring the smallest possible number of evolutionary changes
    = the most parsimonious tree(s).

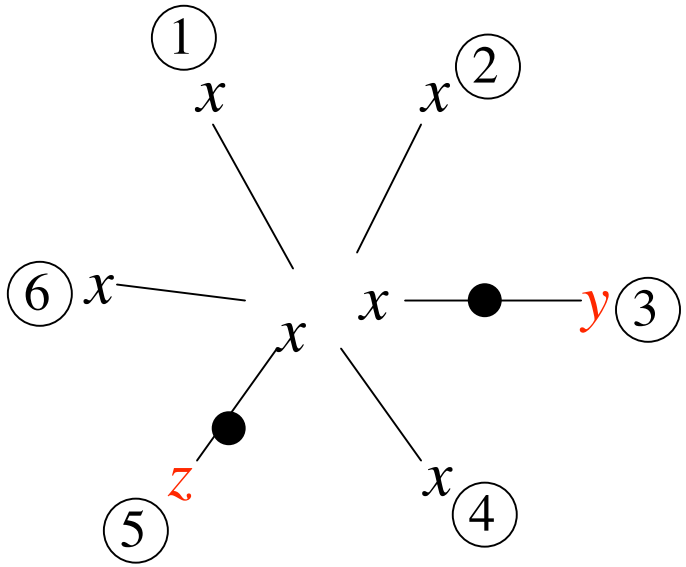
Two major programs implement parsimony for molecular sequences
- PAUP* by David Swofford
- Protpars and dnapars programs of Joe Felsentein's PHYLIP package

Parsimony : informative sites



Whatever the chosen tree shape, d = 1 for this site. Thus, this site does not contribute to choosing which tree shape has the smallest L value.

These sites do not contain information supporting any tree shape: they are uninformative. A site is **informative** iff it contains at least 2 states each present at least twice.



Whatever the chosen tree shape, d = 2 for this site.

Some properties of Parsimony

- Produces unrooted trees.
- Algorithm and the principle apply very generally (*e.g.*, DNA, proteins, morphological data).
- Changes cannot be uniquely located on a specific branch.

  ==> Parsimony does not allow to define unambiguously the length of each tree branch. Only the sum total of branch lengths is unambiguously defined.
- Very often, several tree shapes are equally parsimonious (have same *L* value, the smallest one).
- The number of tree shapes grows very fast with the number of analyzed sequences.

  ==> The search for the shortest tree must be restricted to a fraction of all possible tree shapes.

  A heuristic procedure determines the fraction of the space of tree shapes that is explored.

  There is no mathematical certainty to find the shortest tree.

PHYLIP's tree space exploration heuristic

1. Define an arbitrary order of sequences. Start with the first 3 sequences and the unique possible tree. This gives the current candidate tree.

2. Evaluate addition of the next sequence in all possible positions in the candidate tree; retain the best one. This gives a candidate tree with one more sequence.

3. Do local rearrangements : each internal branch defines 4 sub-trees: a, b, c, d and a topology between them: $\begin{matrix} a & c \\ & \\ b & d \end{matrix}$ evaluate the 2 alternative topologies: $\begin{matrix} a & b \\ & \\ c & d \end{matrix}$ $\begin{matrix} a & c \\ & \\ d & b \end{matrix}$

and retain any better alternative as new candidate tree.

4. Repeat 2. and 3. as long as there remains sequences to process.

5. Do global rearrangements: evaluate all alternative positions of all sub-trees of the candidate tree; retain any better alternative as new candidate tree. Stop when no alternative position reduces the total tree length L.

PHYLIP's tree space exploration heuristic *(continued)*

This heuristics transforms an impossible computation (evaluate all possible tree shapes) into one feasible in a few minutes for up to 30 - 40 sequences.

It is wise to repeat all of steps 1 - 5 changing the initial sequence order. Very often, a shorter tree pops out in one of the repeats.

Local rearrangements are better called **NNI**'s (Nearest Neighbor Interchanges). Global rearrangements are better called **SPR**'s (Subtree Pruning Regrafting).

PAUP's tree space exploration heuristic

Options to control the number of repeats of initial sequence orders:

```
hsearch addseq=random nreps=20
```

Options to control tree space exploration:

```
hsearch swap=NNI|SPR|TBR
```

where NNI and SPR are as above and TBR is Tree Bisection Reconnection: a subtree is pruned, then rerooted, and regrafted somewhere else. Thus, `swap=SPR` is equivalent to PHYLIP's heuristics. Use of TBR produces a much more extensive tree space search.

PAUP can also perform an exhaustive tree space exploration:

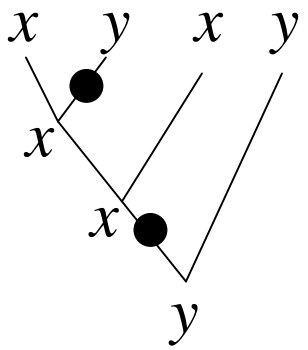
```
BandB
```

(stands for branch-and-bound) but this will last forever unless the number of sequences is very small.

Tree branch lengths in parsimony

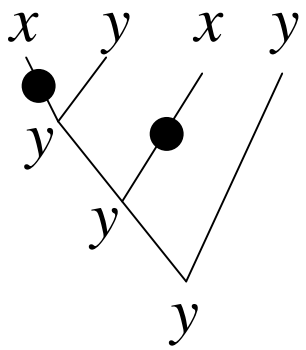
There are very often different equally parsimonious ways to put changes on tree branches. Consequently, there are no unique parsimony-defined branch lengths. It is nevertheless possible to compute minimum and maximum branch lengths.

However, on rooted trees, PAUP offers the possibility to always choose the same strategy to place changes on branches.



accelerated transformation

`OPT=ACCTRAN`



delayed transformation

`OPT=DELTRAN`

Under this condition, it is possible to compute unambiguous branch lengths. But this is entirely arbitrary.

Dealing with sequence gaps in parsimony

Gaps can be processed in two ways

- As missing data. This amounts to replacing the gap site by the residue requiring the least number of changes at this site.

- As a 5th base or a 21st amino acid. Gap-to-residue changes count as a residue-to-other-residue change. This is not satisfactory because a gap of length *n* counts as *n* independent events.

In PAUP:

```
GAPMODE = MISSING|NEWSTATE
```

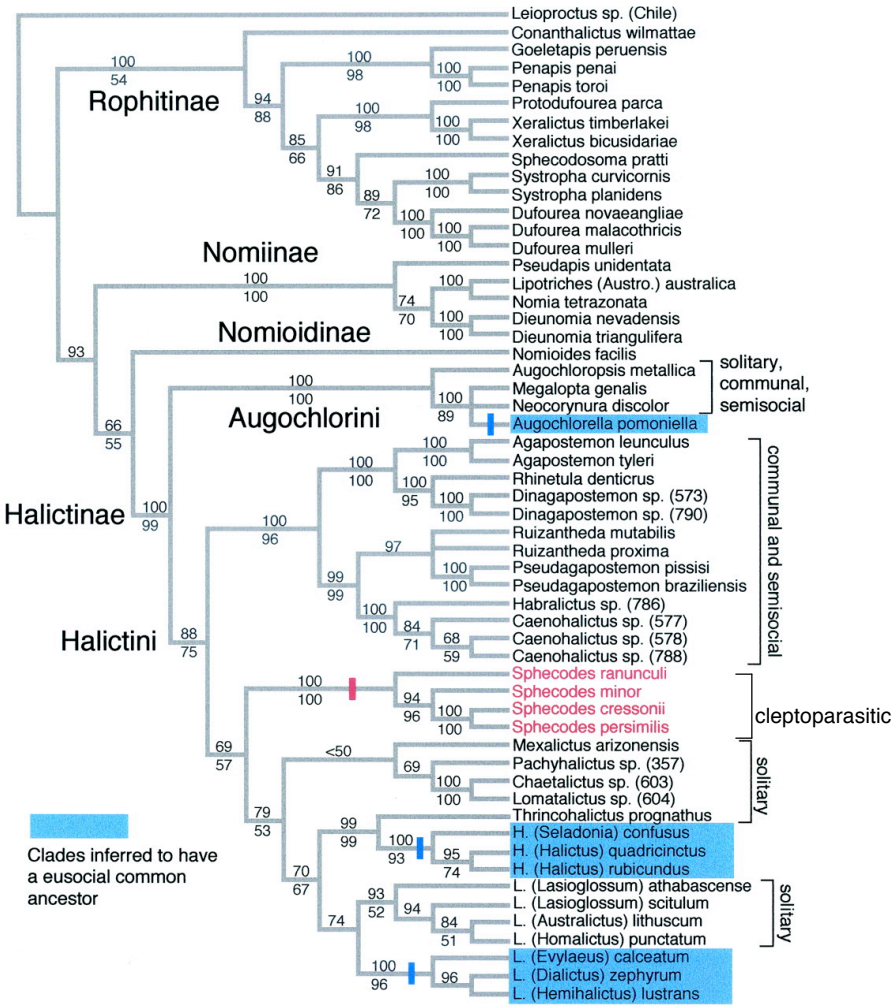
PHYLIP's implementation of protein sequence parsimony

Amino acid replacements are scored as the minimum number of changes between two synonymous codons of these amino acids according to the genetic code.

TTT	Phe	TCT	Ser	TAT	Tyr	TGT	Cys
TTC	Phe	TCC	Ser	TAC	Tyr	TGC	Cys
TTA	Leu	TCA	Ser	TAA	stop	TGA	stop
TTG	Leu	TCG	Ser	TAG	stop	TGG	Trp
CTT	Leu	CCT	Pro	CAT	His	CGT	Arg
CTC	Leu	CCC	Pro	CAC	His	CGC	Arg
CTA	Leu	CCA	Pro	CAA	Gln	CGA	Arg
CTG	Leu	CCG	Pro	CAG	Gln	CGG	Arg
ATT	Ile	ACT	Thr	AAT	Asn	AGT	Ser
ATC	Ile	ACC	Thr	AAC	Asn	AGC	Ser
ATA	Ile	ACA	Thr	AAA	Lys	AGA	Arg
ATG	Met	ACG	Thr	AAG	Lys	AGG	Arg
GTT	Val	GCT	Ala	GAT	Asp	GGT	Gly
GTC	Val	GCC	Ala	GAC	Asp	GGC	Gly
GTA	Val	GCA	Ala	GAA	Glu	GGA	Gly
GTG	Val	GCG	Ala	GAG	Glu	GGG	Gly

Examples: Met/Val scores 1; Val/Thr scores 2; Phe/Gln scores 3

Evolution of sociality in a primitively eusocial lineage of bees



Phylogeny of the halictid subfamilies, tribes, and genera. Strict consensus of six trees based on equal weights parsimony analysis of the entire data set of three exons and two introns. Two regions within the introns were excluded because they could not be aligned unambiguously. Gaps coded as a fifth state or according to the methods described in ref. 23 yielded the same six trees. Bootstrap values above the nodes indicate bootstrap support based on the exons introns data set. Bootstrap values below the nodes indicate support based on an analysis of exons only. For the exons introns analysis the data set included 1,541 total aligned sites (619 parsimony-informative sites), the trees were 3,388 steps in length.

Advanced eusocial insects, such as ants, termites, and corbiculate bees, cannot provide insights into the earliest stages of eusocial evolution because eusociality in these taxa evolved long ago (in the Cretaceous) and close solitary relatives are no longer extant. In contrast, primitively eusocial insects, such as halictid bees, provide insights into the early stages of eusocial evolution because eusociality has arisen recently and repeatedly. I show that eusociality has arisen only three times within halictid bees.

Danforth, Bryan N. (2002) Proc. Natl. Acad. Sci. USA 99, 286-290