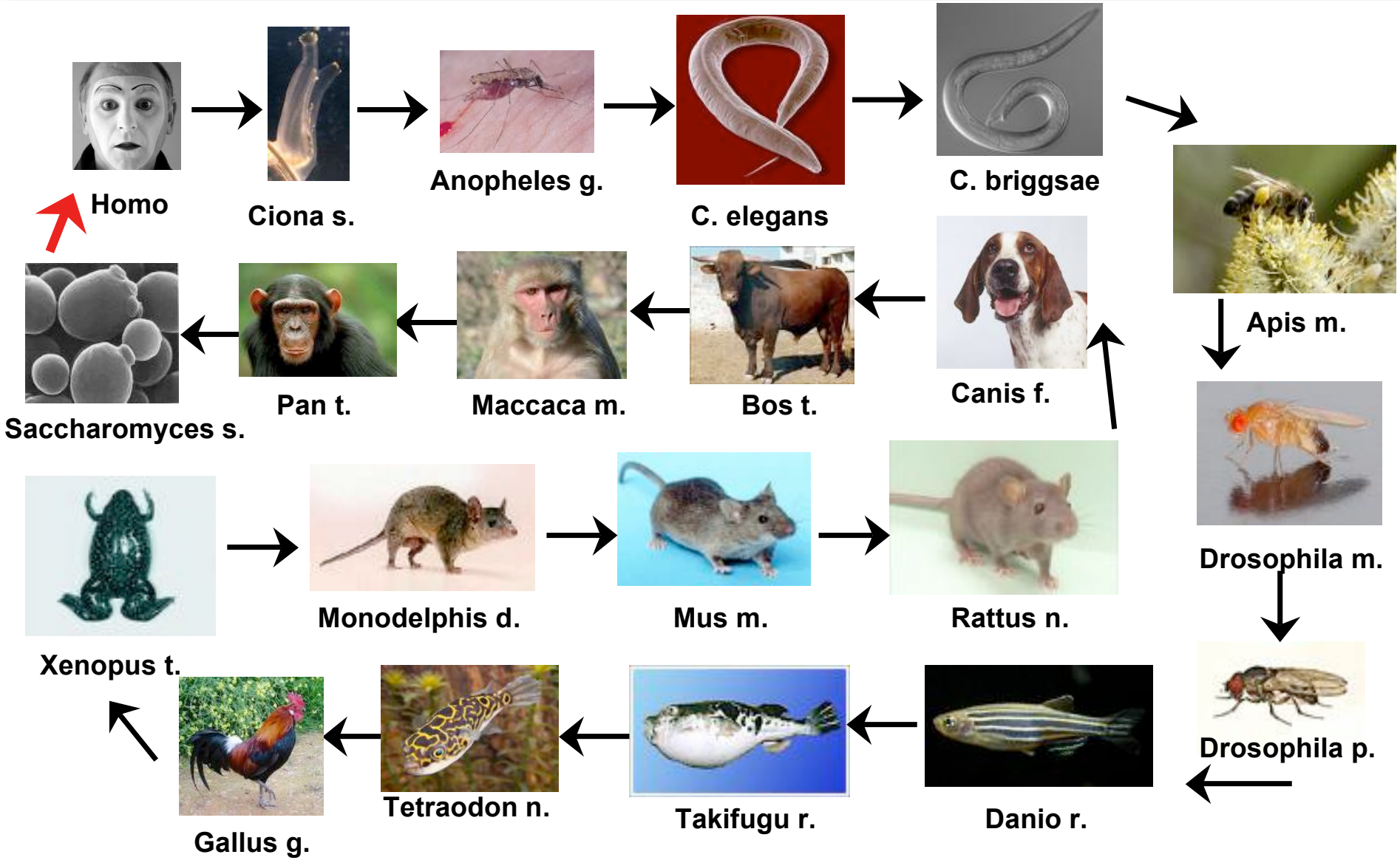
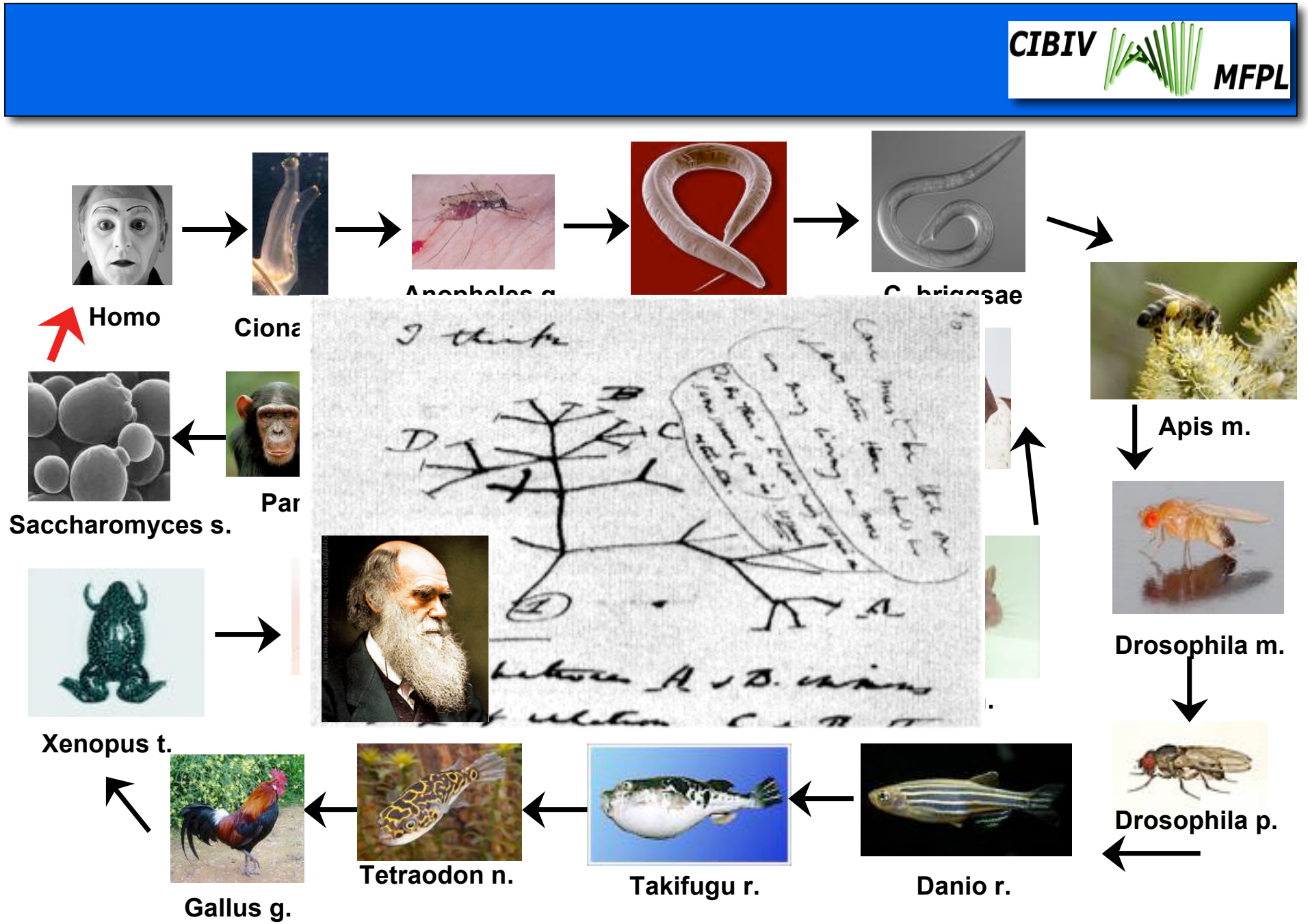
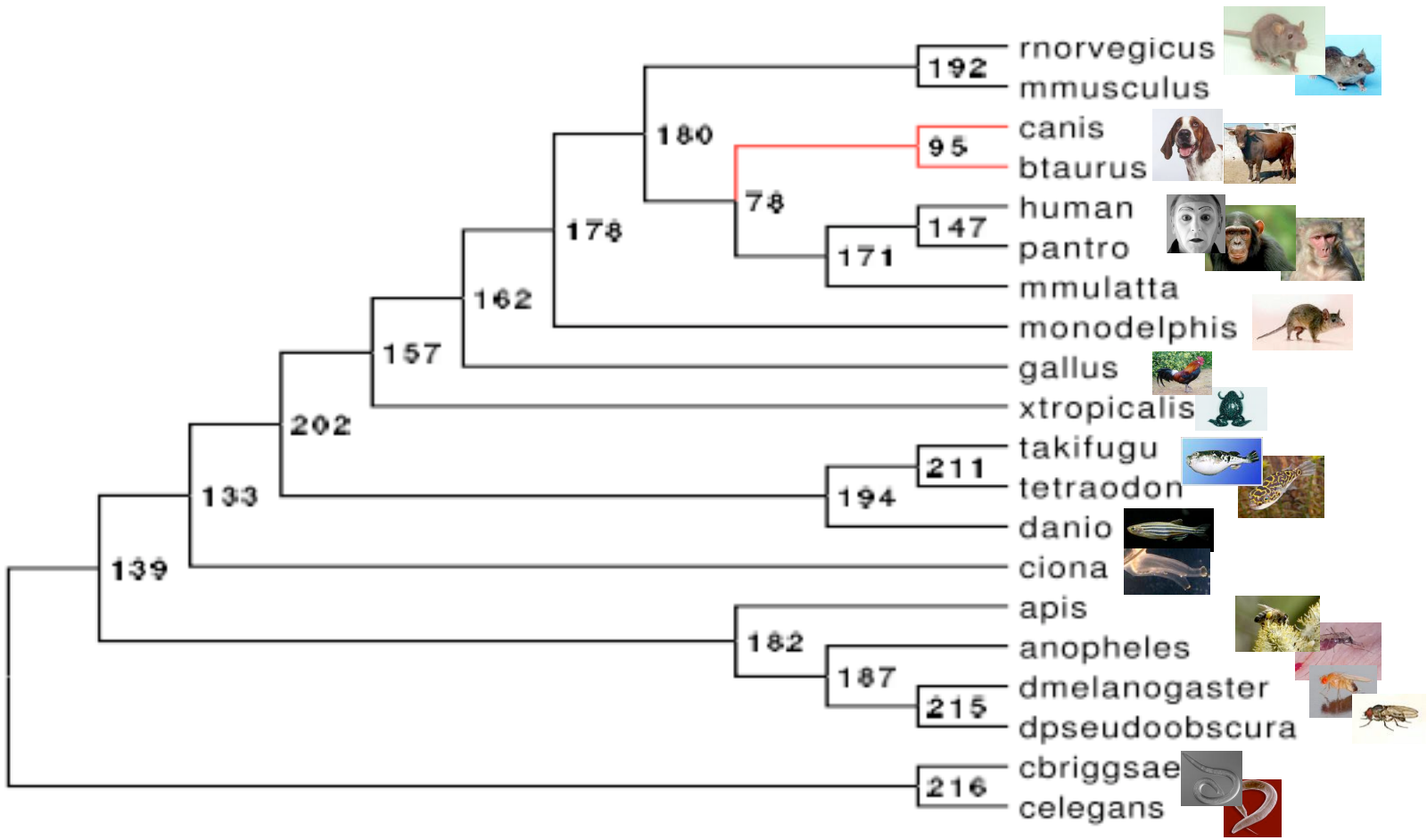
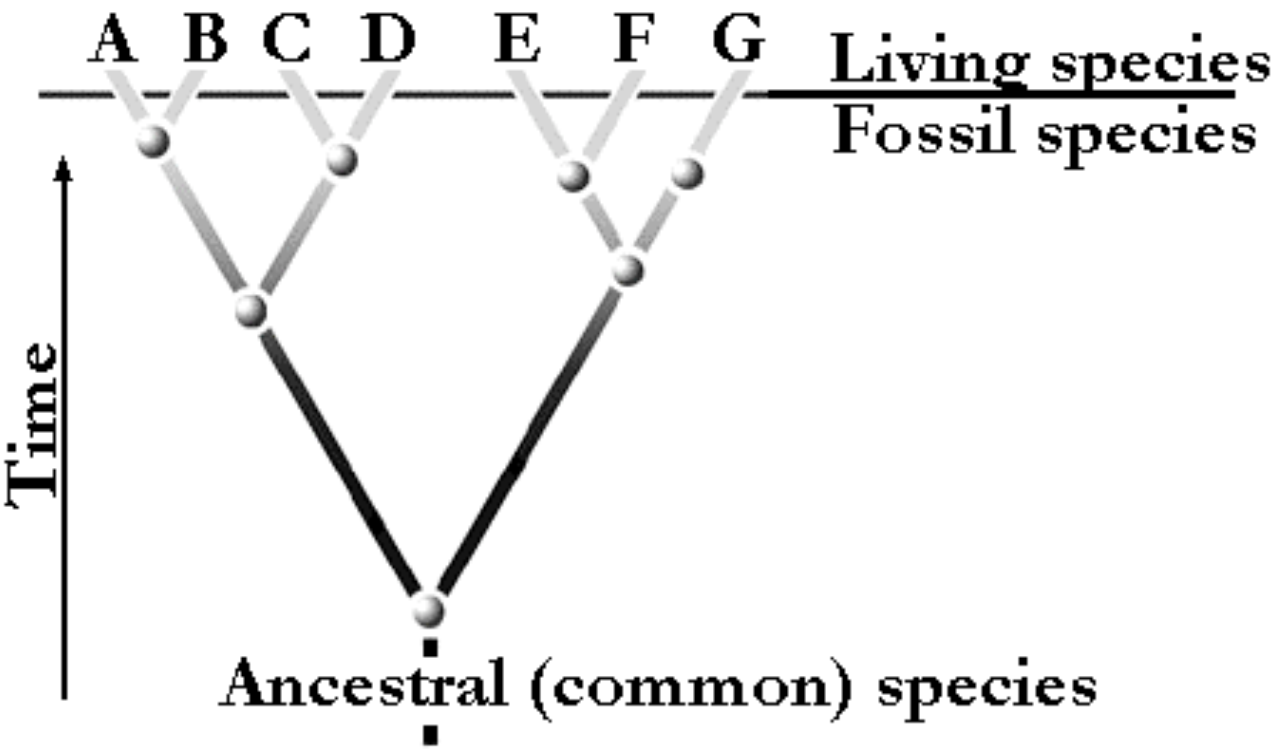
PHYLOGENY RECONSTRUCTION: THE BASICS

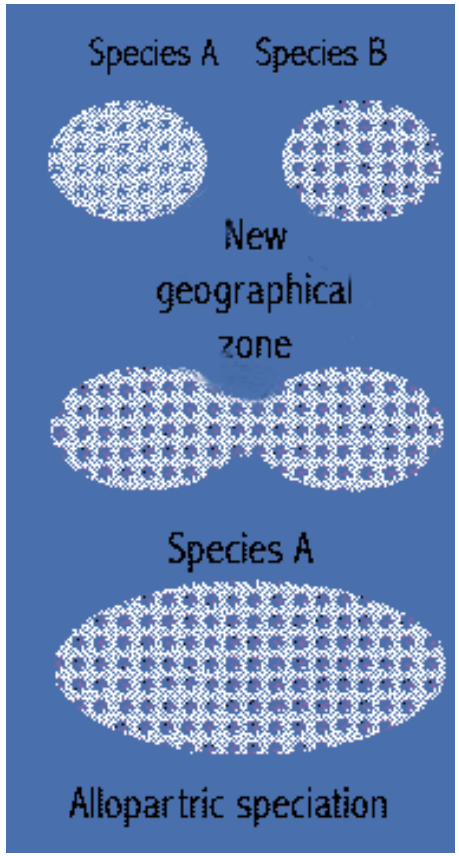

Homo
Ciona s.
Anopheles g.
C. elegans
C. briggsae
Apis m.
Drosophila m.
Drosophila p.
Danio r.
Takifugu r.
Tetraodon n.
Gallus g.
Xenopus t.
Monodelphis d.
Mus m.
Rattus n.
Canis f.
Bos t.
Maccaca m.
Pan t.
Saccharomyces s.


CIBIV
MFPL
Homo
Ciona
Anopheles g.
C. briggsae
Apis m.
Drosophila m.
Drosophila p.
Danio r.
Takifugu r.
Tetraodon n.
Gallus g.
Xenopus t.
Par
Saccharomyces s.

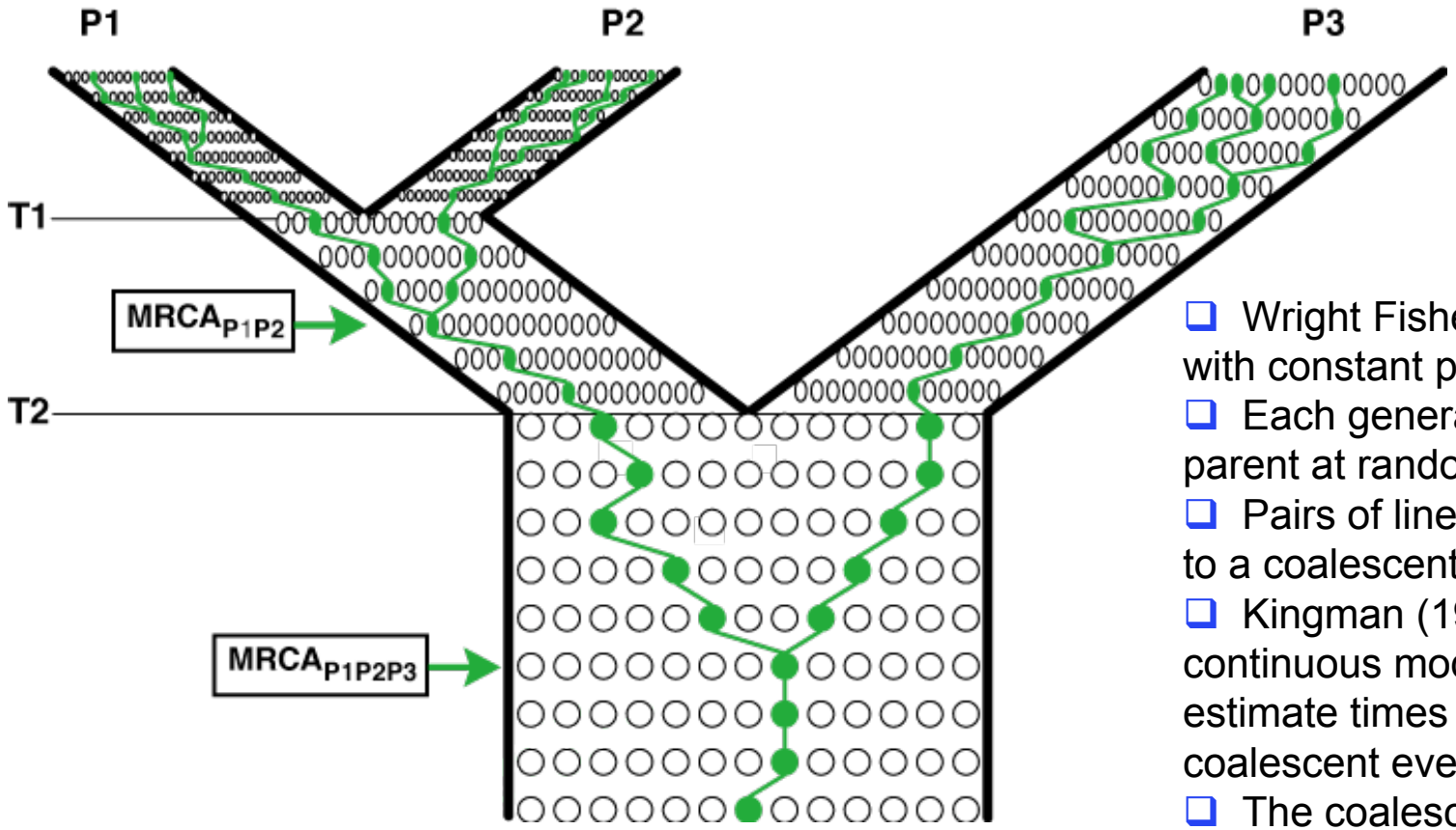

rnorvegicus
mmusculus
canis
btaurus
human
pantro
mmulatta
monodelphis
gallus
xtropicalis
takifugu
tetraodon
danio
ciona
apis
anopheles
dmelanogaster
dpseudoobscura
cbriggsae
celegans
192
95
147
211
215
216
180
78
171
178
162
157
202
194
133
182
187
139

A Simple Concept of Speciation



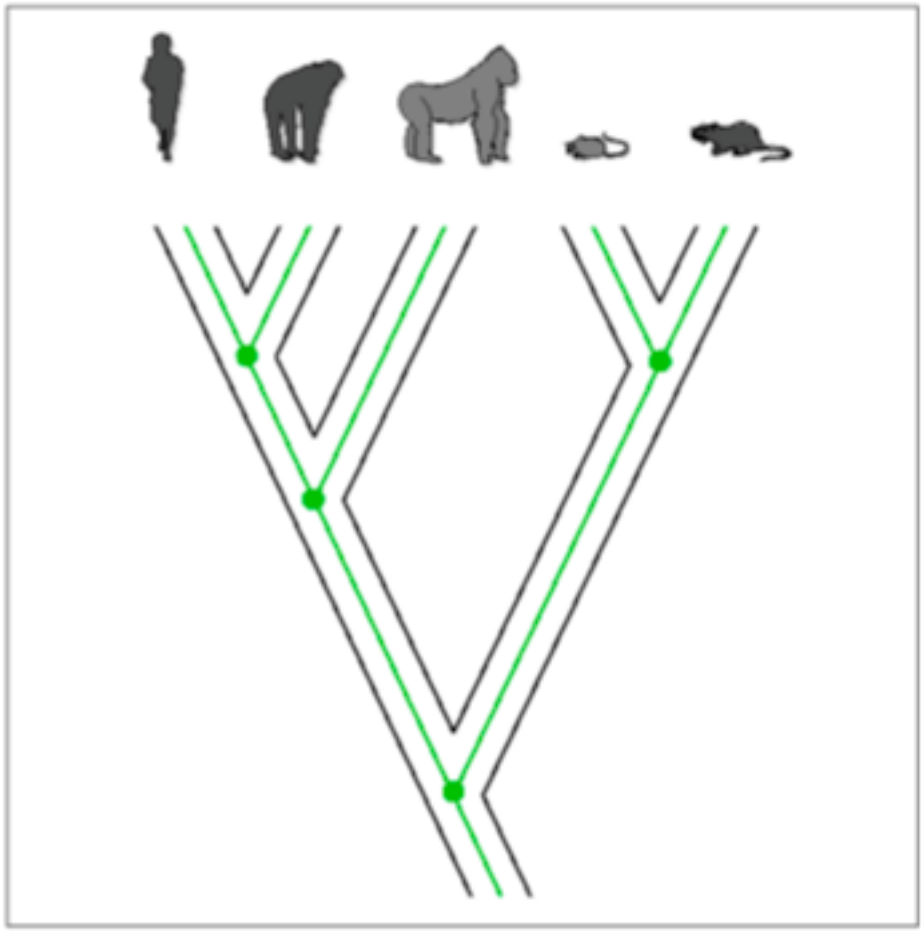


The Coalescent Model

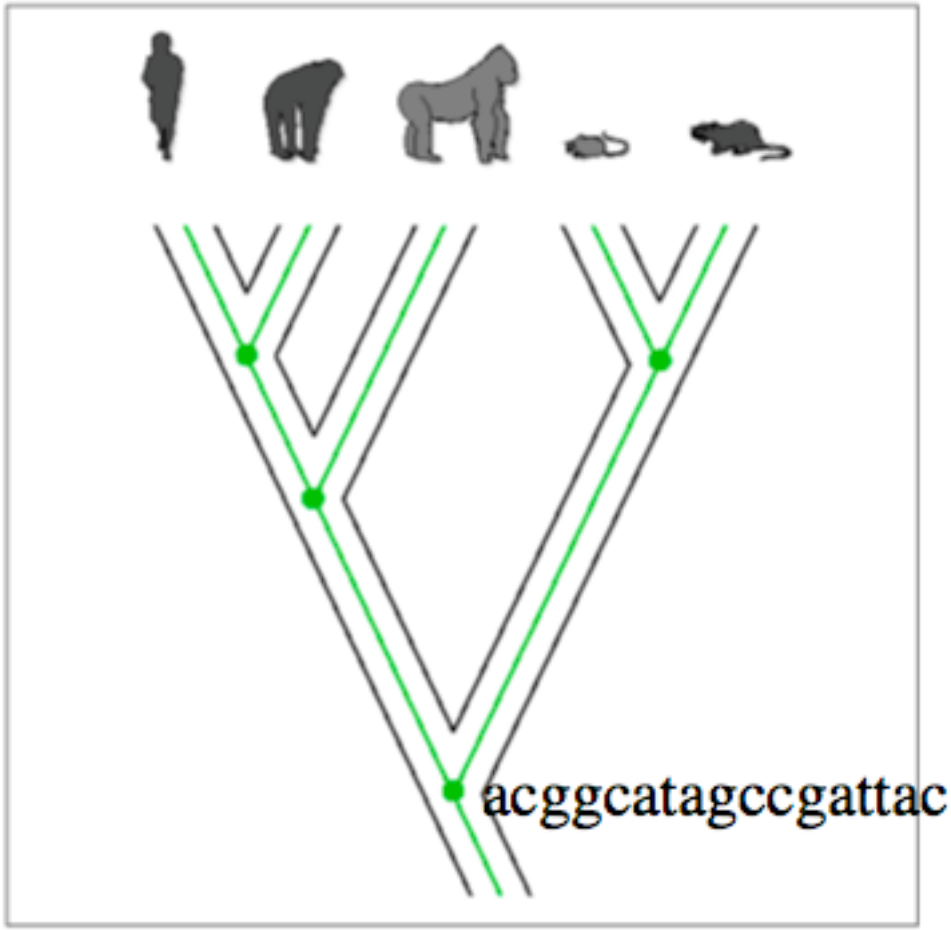


- Wright Fisher population model with constant population size.
- Each generation chooses its parent at random.
- Pairs of lineages are traced back to a coalescent event.
- Kingman (1982) developed a continuous model that allows to estimate times between the coalescent events.
- The coalescent rate for any pair of genetic lineages is proportional to 1/Ne in generations or to 1/θ in substitutions.

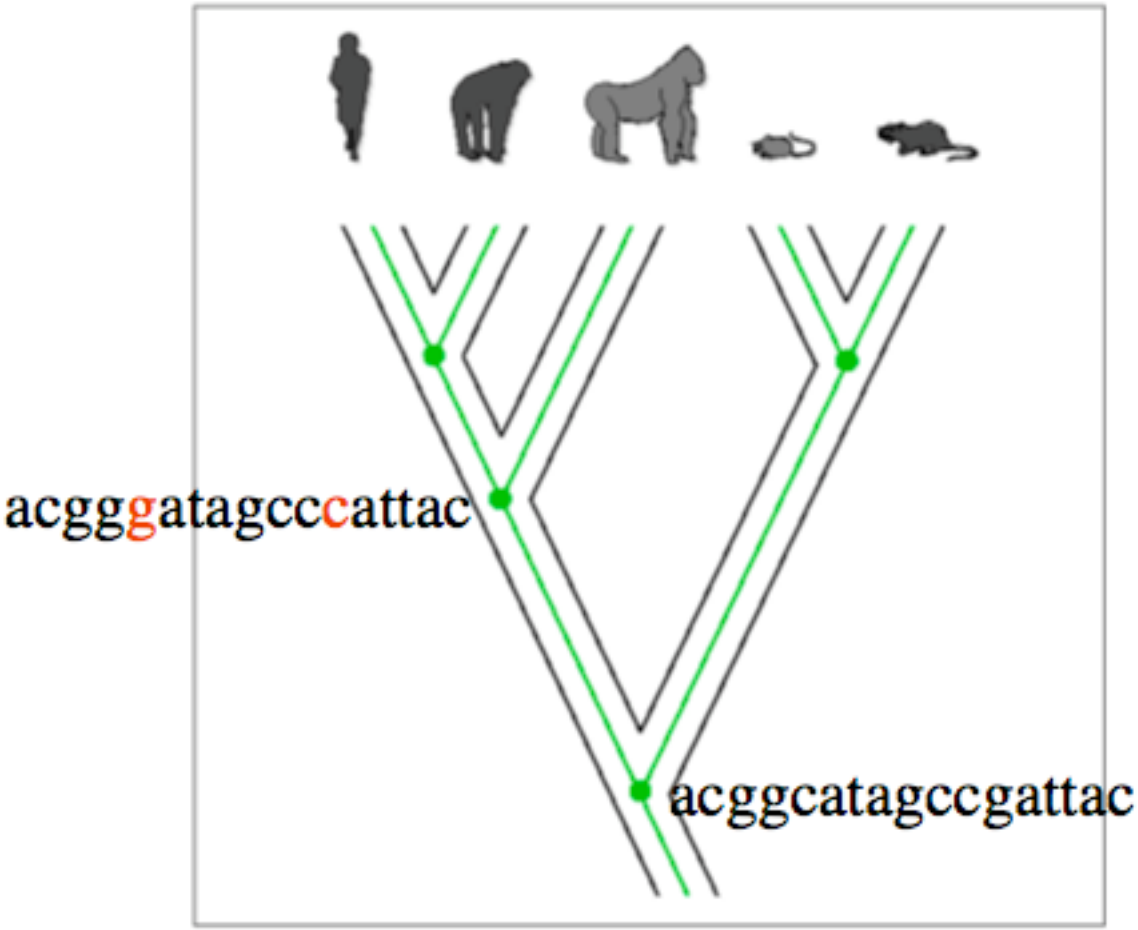
The evolution of biological sequences



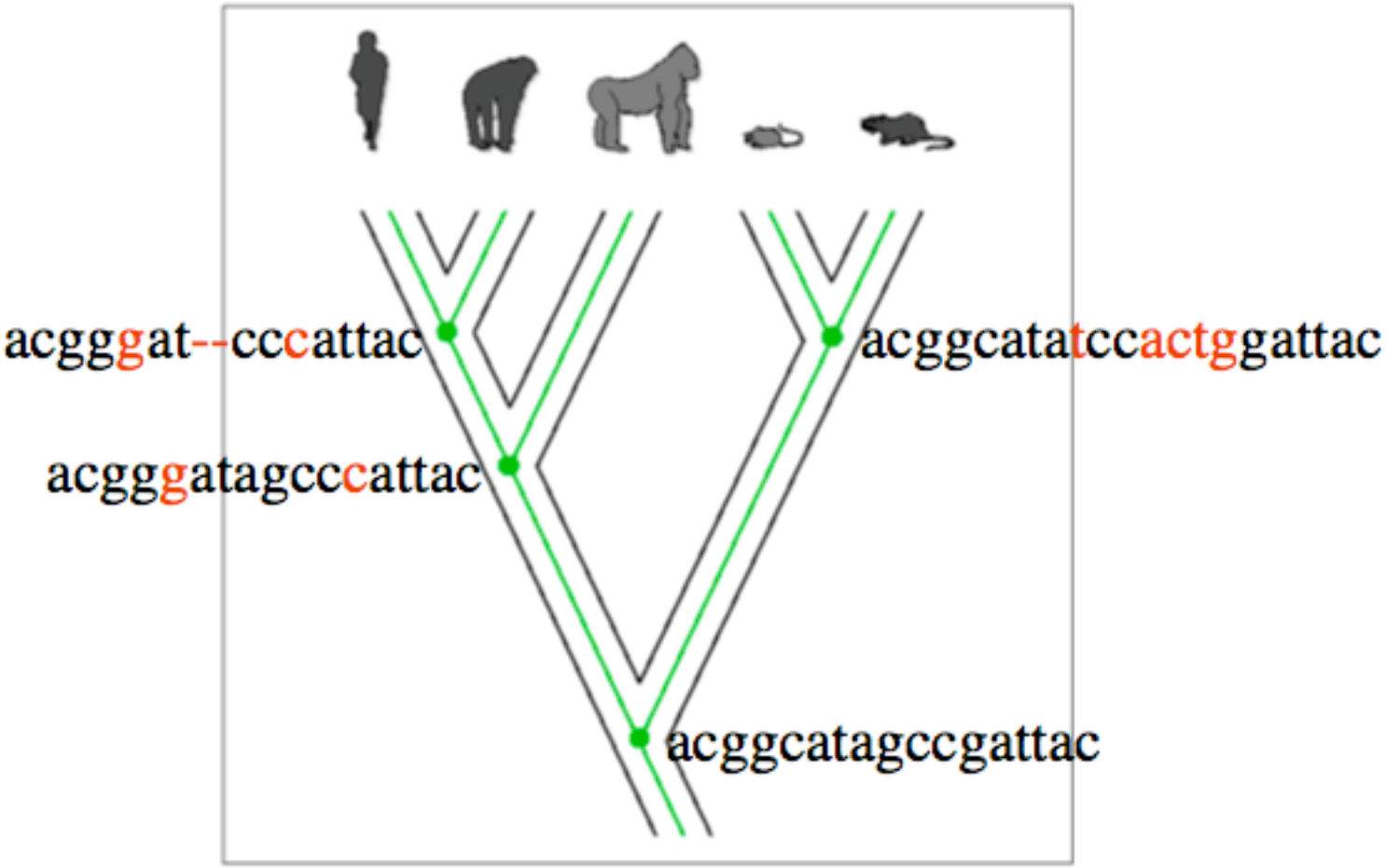
The evolution of biological sequences



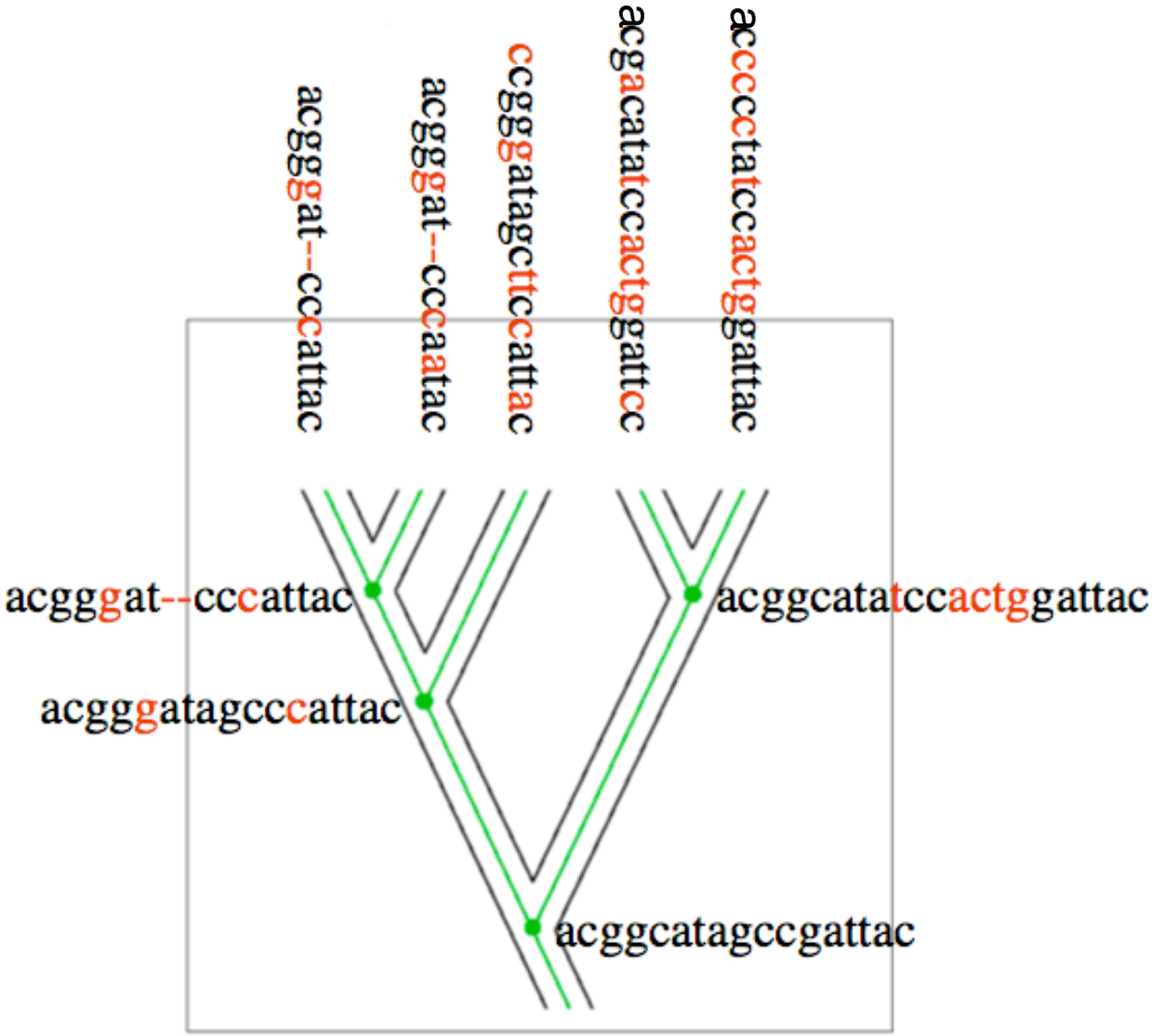
Biological sequences can change by *substitutions*



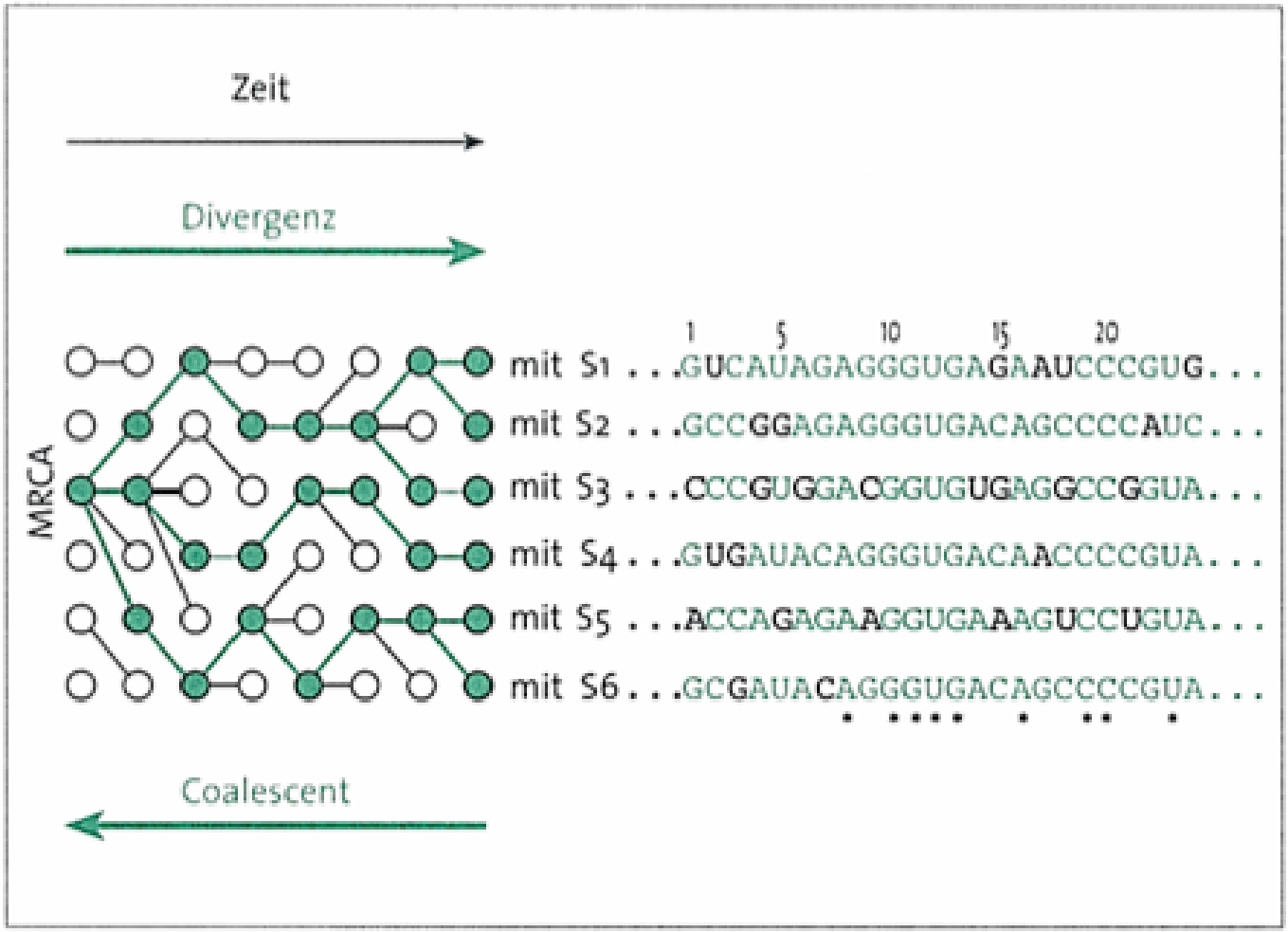
...deletions and insertions



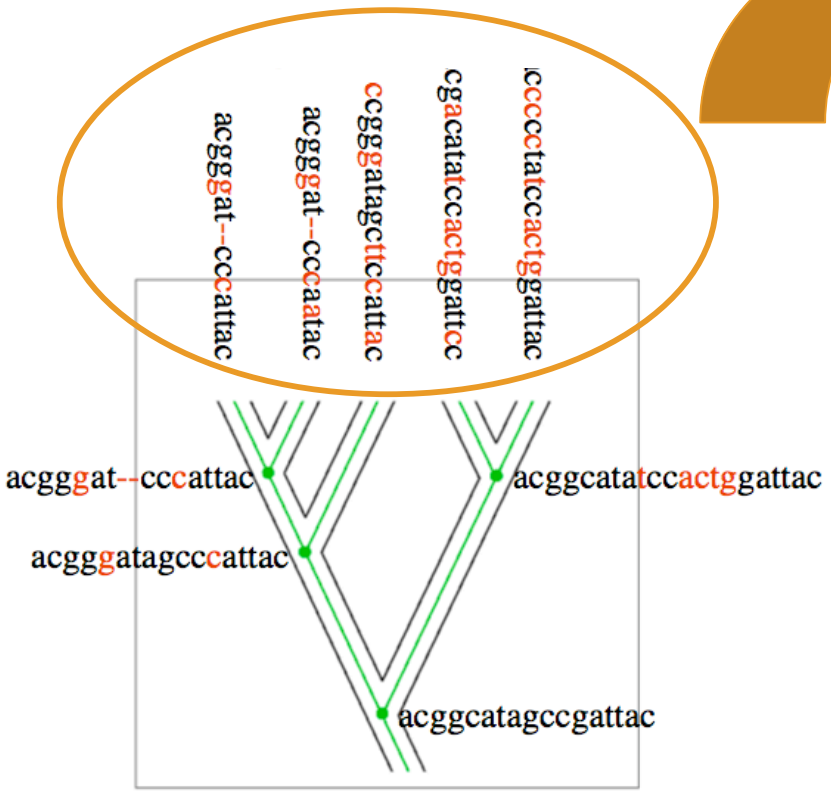
At the leafs of the tree we find the contemporary sequences



Divergence and Coalescent



What we have...



What do we get:

acgggatcccattac

acgggatcccaatac

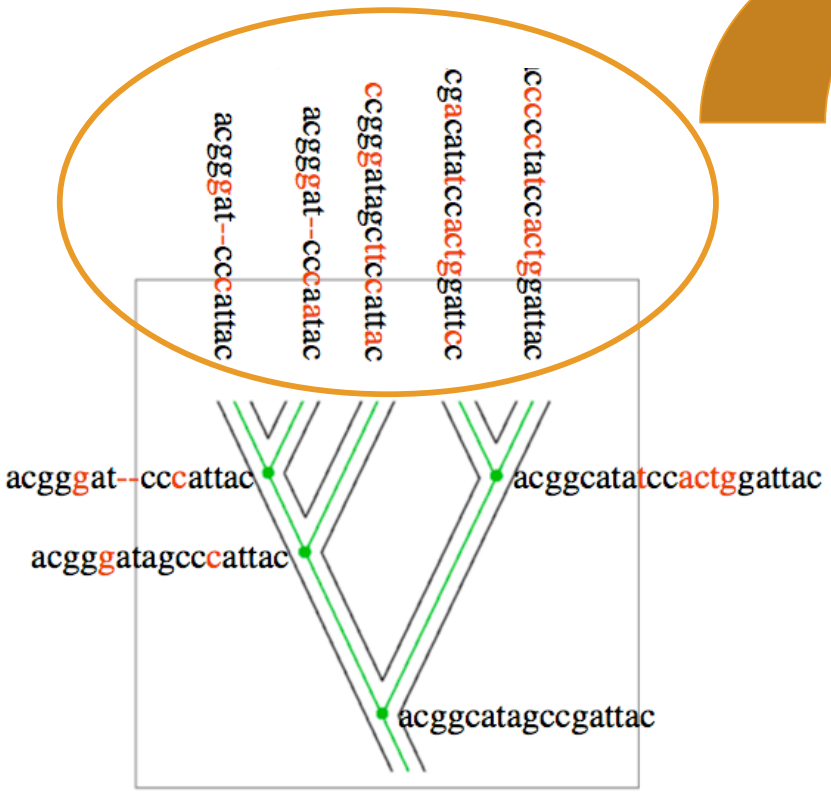
ccgggatagcttccattac

acgacatatccactggattcc

acccccctatccactggattac

A collection of homologous sequence that vary slightly in their nucleotide or amino acid composition and in their length

What we want...



What do we get:

acgggatcccattac

acgggatcccaatac

ccgggatagcttccattac

acgacatatccactggattcc

acccccctatccactggattac

A collection of homologous sequence that vary slightly in their nucleotide or amino acid composition and in their length

?

Public Data Sources

CIBIV MFPL

Human (Homo sapiens) Genome Browser Gateway

http://genome.ucsc.edu/cgi-bin/hgGateway

Home Genomes Blat Tables Gene Sorter PCR Session FAQ Help

Human (*Homo sapiens*) Genome Browser Gateway

clade genome

Vertebrate Human

Human Chimp Rhesus Mouse Rat Cat Dog Horse Cow Opossum

FlyBase Homepage

http://flybase.bio.indiana.edu/

FlyBase

A Database of *Drosophila* Genes & Genomes

Home Tools Files Species Documents Resources News Help Archives

NCBI

National Center for Biotechnology Information

National Library of Medicine

PubMed All Databases BLAST OMIM

Search All Databases for

SITE MAP Alphabetical List Resource Guide

About NCBI An introduction to NCBI

GenBank Sequence submission support and software

Literature databases PubMed, OMIM, Books, and PubMed Central

Molecular databases Sequences, structures, and taxonomy

What does NCBI do?

Inparanoid

Home
All species vs all
Gene search
Text search
Blast search
Downloads
Previous version
Summary
FAQ

version 6.0

InParanoid: Eukaryotic Ortholog Groups

35 organisms: 610047 sequences

Version 6.0, Updated August 2007

- BROWSE the database - Select two species and view all their orthologs
- SEARCH BY SEQUENCE IDs - View orthologs of a specific gene or protein
- TEXT SEARCH - Query InParanoid by keywords
- BLAST SEARCH - Find orthologs in InParanoid similar to your protein sequence
- DOWNLOAD DATA - Obtain tables, html, xml, sequences and core data
- SUMMARY OF INPARANOID - Statistics of the database and genomes used

Stockholm Bioinformatics Centre SBC

Ensembl

Ensembl release 44 - Apr 2007

Your Ensembl

Login or Register

About User Accounts

Help & Documentation

About Ensembl

Genomic Data

Help & Information

Software

Select a species

Mammals

Other chordates

Other eukaryotes

Ensembl Archive

View previous release of page in Archive!

Stable Archive! link for this page

Ground squirrel *Spermophilus*

Important Notice

More genomes

Aedes aegypti AaegL1

Anopheles gambiae AgamP3

Bos taurus Btau_3.1

Caenorhabditis elegans WS170 UPDATED!

Canis familiaris CanFam 2.0

Cavia porcellus cavPor2

Ciona intestinalis JGI 2

Ciona savignyi CSAV 2.0

http://www.wormbase.org/

Home Genome Blast / Blat WormMart Batch

WormBase Release WS175

Find: Any Gene

Exact match Results as

Web Site Directory

Release Notes New/Changed Genes, release notes

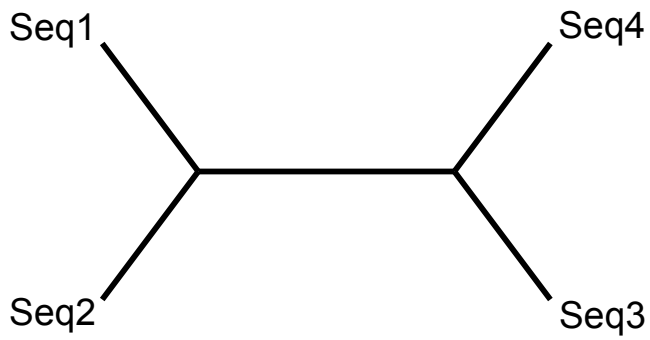
General Searches WormBase Class Browser, Wormbase Query Language Search, AQL Search

Sequences *C. elegans* Genome, *C. briggsae* Genome, Gene, Blast / Blat, e-PCR, Gene Ontology, Synteny Viewer, Cis-Elements (CisOrtho),

Cells and Gene Expression Cell and Pedigree, Neurons,

May 22, 2007: Service to WormBase restored

The Problem: Finding the homologous positions




```
Seq1: - A C G A
Seq2: T A C G T
Seq3: - A T - T
Seq4: - A T G T
```

A C G A T A C G T A T T A T G T

The objective function

An mathematical function able to measure the biological quality of an alignment...

The objective function

An mathematical function able to measure the biological quality of an alignment...

Related questions:

- What should a biologically correct alignment look like?
- To what extent can we define and formalize its properties?

The objective function

An mathematical function able to measure the biological quality of an alignment...

Related questions:

- What should a biologically correct alignment look like?
- To what extent can we define and formalize its properties?

Mathematical Optimal Alignment ⟷ (minimize) Biologically Optimal Alignment

The objective function

A mathematical function meant to measure the biological quality of an alignment...

$$\sigma(\alpha) = \sum_{i=1}^{n} S(a_i, b_i)$$

$\sigma(\alpha)$: the score of the pairwise alignment α

n : length of α

a_i : letter of sequence A at position i in α

b_i : letter of sequence B at position i in α

The objective function

A mathematical function meant to measure the biological quality of an alignment...

$$\sigma(\alpha) = \sum_{i=1}^{n} S(a_i, b_i)$$

Objective: find α that maximizes $\sigma(\alpha)$!

The scoring function *S, an example*

Given two sequences A ={a_1,a_2,....,a_n} and B={b_1,b_2,....,b_m} and a scoring function *S* such that

$$S(a_i,b_j) = \begin{cases} +5, if\ \ a_i = b_j \\ -2, if\ \ a_i \neq b_j \\ -6, \text{for introduction of a gap} \end{cases}$$

then we look for that alignment, that gives us the highest score by summing up the column scores S(a_i,b_j) for all columns of the alignment.

The scoring function *S, an example*

Given two sequences A $=\{a_1,a_2,\ldots,a_n\}$ and B$=\{b_1,b_2,\ldots,b_m\}$ and a scoring function *S* such that

$$S(a_i,b_j)=\begin{cases}+5, \text{ if } a_i=b_j\\ -2, \text{ if } a_i\neq b_j\\ -6, \text{ for introduction of a gap}\end{cases}$$

then we look for that alignment, that gives us the highest score by summing up the column scores $S(a_i,b_j)$ for all columns of the alignment.

For example:

```
 T    G    C    T    C    G    T    A
 T    -    -    T    C    A    T    A
+5   -6   -6   +5   +5   -2   +5   +5  = 11
```

continue.....

A1:	T	G	C	T	C	G	T	A	
	T	-	-	T	C	A	T	A	
	+5	-6	-6	+5	+5	−2	+5	+5	= 11

A2:	T	G	C	T	C	G	T	A	
	T	-	T	-	C	A	T	A	
	+5	-6	-2	-6	+5	−2	+5	+5	= 4

etc...

Why not just scoring all alignments?

- There are far too many
  - number of possible pairwise alignments: $\binom{2n}{n}$
  - for two sequences of length N=300 there are 10^{179} possibilities

Why not just scoring all alignments?

- There are far too many
  - number of possible pairwise alignments: $\binom{2n}{n}$
  - for two sequences of length N=300 there are 10^{179} possibilities

Hence, we need a smart way to cut the computation short, like the **dynamic programming** approach for pairwise alignments by *Needleman and Wunsch* (1970).

Re-use of previous results

A1:	T	G	C	T	C	G	T	A	
	T	-	-	T	C	A	T	A	
	+5	-6	-6	+5	+5	−2	+5	+5	= 11
A2:	T	G	C	T	C	G	T	A	
	T	-	T	-	C	A	T	A	
	+5	-6	-2	-6	+5	−2	+5	+5	= 4

etc...

Dynamic Programming

A **dynamic programming** approach usually includes:

- A mathematical description of the (biological) quality of an solution, i.e. an recursive objective function
- The computation of all intermediate values needed to obtain the globally optimal solution, thereby avoiding double-computations
- The reconstruction of the globally optimal solution from the values obtained in the previous step (backtracking)

The Needleman-Wunsch pair-wise alignment

		0	1	2	3	4	5	6	7	8
			T	G	C	T	C	G	T	A
0										
1	T									
2	T									
3	C									
4	A									
5	T									
6	A									

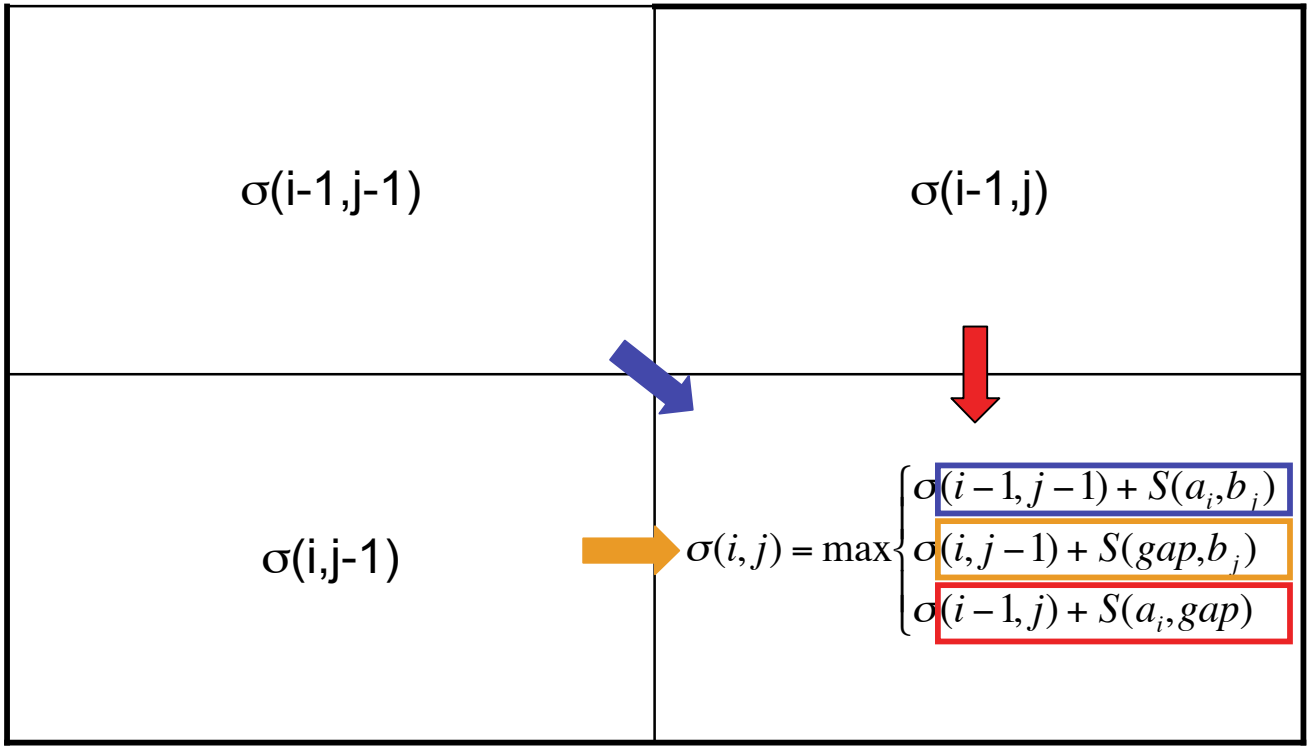
Scoring function

$$S(a_i,b_j) = \begin{cases} +5, \text{ if } a_i = b_j \\ -2, \text{ if } a_i \neq b_j \\ -6, \text{ for introduction of a gap} \end{cases}$$

Objective function

$$\sigma(i,j) = \max \begin{cases} \sigma(i-1,j-1) + S(a_i,b_j) \\ \sigma(i,j-1) + S(gap,b_j) \\ \sigma(i-1,j) + S(a_i,gap) \end{cases}$$

The Needleman-Wunsch algorithm



- σ(i,j) is the optimal alignment score up to and including a_i and b_j

$$S(a_i,b_j) = \begin{cases} +5, \text{ if } a_i = b_j \\ -2, \text{ if } a_i \neq b_j \\ -6, \text{ for introduction of a gap} \end{cases}$$

Needleman-Wunsch algorithm: Initialization

		0	1	2	3	4	5	6	7	8
			T	G	C	T	C	G	T	A
0		0	−6	−12	−18	−24	−30	−36	−42	−48
1	T	−6								
2	T	−12								
3	C	−18								
4	A	−24								
5	T	−30								
6	A	−36								

$$S(a_i, b_j) = \begin{cases} +5, \text{ if } a_i = b_j \\ -2, \text{ if } a_i \neq b_j \\ -6, \text{ for introduction of a gap} \end{cases}$$

The Needleman-Wunsch algorithm: Recursion

		0	1	2	3	4	5	6	7	8
			T	G	C	T	C	G	T	A
0		0	−6	−12	−18	−24	−30	−36	−42	−48
1	T	−6	5							
2	T	−12								
3	C	−18								
4	A	−24								
5	T	−30								
6	A	−36								

$$S(a_i, b_j) = \begin{cases} +5, \text{ if } a_i = b_j \\ -2, \text{ if } a_i \neq b_j \\ -6, \text{ for introduction of a gap} \end{cases}$$

The Needleman-Wunsch algorithm: Recursion

		0	1	2	3	4	5	6	7	8
			T	G	C	T	C	G	T	A
0		0	−6	−12	−18	−24	−30	−36	−42	−48
1	T	−6	5	−1						
2	T	−12								
3	C	−18								
4	A	−24								
5	T	−30								
6	A	−36								

$$S(a_i, b_j) = \begin{cases} +5, \text{ if } a_i = b_j \\ -2, \text{ if } a_i \neq b_j \\ -6, \text{ for introduction of a gap} \end{cases}$$

The Needleman-Wunsch algorithm: Recursion

		0	1	2	3	4	5	6	7	8
			T	G	C	T	C	G	T	A
0		0	−6	−12	−18	−24	−30	−36	−42	−48
1	T	−6	5	−1	−7	−13	−19	−25	−31	−37
2	T	−12	−1	3	−3	−2	−8	−14	−20	−26
3	C	−18	−7	−3	8	2	3	−3	−9	−15
4	A	−24	−13	−9	2	6	0	1	−5	−4
5	T	−30	−19	−15	−4	7	4	−2	6	0
6	A	−36	−25	−21	−10	1	5	2	0	11

$$S(a_i,b_j) = \begin{cases} +5, \text{ if } a_i = b_j \\ -2, \text{ if } a_i \neq b_j \\ -6, \text{ for introduction of a gap} \end{cases}$$

Needleman-Wunsch algorithm: Backtrack

		0	1	2	3	4	5	6	7	8
			T	G	C	T	C	G	T	A
0		0	−6	−12	−18	−24	−30	−36	−42	−48
1	T	−6	5	−1	−7	−13	−19	−25	−31	−37
2	T	−12	−1	3	−3	−2	−8	−14	−20	−26
3	C	−18	−7	−3	8	2	3	−3	−9	−15
4	A	−24	−13	−9	2	6	0	1	−5	−4
5	T	−30	−19	−15	−4	7	4	−2	6	0
6	A	−36	−25	−21	−10	1	5	2	0	11

*

*

Needleman-Wunsch algorithm: Backtrack

		0	1	2	3	4	5	6	7	8
			T	G	C	T	C	G	T	A
0		0	−6	−12	−18	−24	−30	−36	−42	−48
1	T	−6	5	−1	−7	−13	−19	−25	−31	−37
2	T	−12	−1	3	−3	−2	−8	−14	−20	−26
3	C	−18	−7	−3	8	2	3	−3	−9	−15
4	A	−24	−13	−9	2	6	0	1	−5	−4
5	T	−30	−19	−15	−4	7	4	−2	6	0
6	A	−36	−25	−21	−10	1	5	2	0	11

A*

A*

Needleman-Wunsch algorithm: Backtrack

		0	1	2	3	4	5	6	7	8
			T	G	C	T	C	G	T	A
0		0	−6	−12	−18	−24	−30	−36	−42	−48
1	T	−6	5	−1	−7	−13	−19	−25	−31	−37
2	T	−12	−1	3	−3	−2	−8	−14	−20	−26
3	C	−18	−7	−3	8	2	3	−3	−9	−15
4	A	−24	−13	−9	2	6	0	1	−5	−4
5	T	−30	−19	−15	−4	7	4	−2	6	0
6	A	−36	−25	−21	−10	1	5	2	0	11

```
TA*
TA*
```

Needleman-Wunsch algorithm: Backtrack

		0	1	2	3	4	5	6	7	8
			T	G	C	T	C	G	T	A
0		0	−6	−12	−18	−24	−30	−36	−42	−48
1	T	−6	5	−1	−7	−13	−19	−25	−31	−37
2	T	−12	−1	3	−3	−2	−8	−14	−20	−26
3	C	−18	−7	−3	8	2	3	−3	−9	−15
4	A	−24	−13	−9	2	6	0	1	−5	−4
5	T	−30	−19	−15	−4	7	4	−2	6	0
6	A	−36	−25	−21	−10	1	5	2	0	11

```
*TGCTCGTA*
*T--TCATA*
```

Alignment Score: 11

Smith-Waterman pairwise local alignment

		0	1	2	3	4	5	6	7	8
			T	G	C	T	C	G	T	A
0		0	0	0	0	0	0	0	0	0
1	T	0	5	0	0	5	0	0	5	0
2	T	0	5	3	0	5	3	0	5	3
3	C	0	0	3	8	2	10	4	0	3
4	A	0	0	0	2	6	4	8	2	5
5	T	0	5	0	0	7	4	2	13	7
6	A	0	0	3	0	1	5	2	7	18

$$S(a_i,b_j) = \begin{cases} +5, \text{ if } a_i = b_j \\ -2, \text{ if } a_i \neq b_j \\ -6, \text{ for introduction of a gap} \end{cases}$$

$$\sigma(i,j) = \max \begin{cases} \sigma(i-1,j-1) + S(a_i,b_j) \\ \sigma(i,j-1) + S(gap) \\ \sigma(i-1,j) + S(gap) \\ 0 \end{cases}$$

Smith-Waterman pairwise local alignment

		0	1	2	3	4	5	6	7	8
			T	G	C	T	C	G	T	A
0		0	0	0	0	0	0	0	0	0
1	T	0	5	0	0	5	0	0	5	0
2	T	0	5	3	0	5	3	0	5	3
3	C	0	0	3	8	2	10	4	0	3
4	A	0	0	0	2	6	4	8	2	5
5	T	0	5	0	0	7	4	2	13	7
6	A	0	0	3	0	1	5	2	7	18

```
*TCGTA*
*TCATA*
```

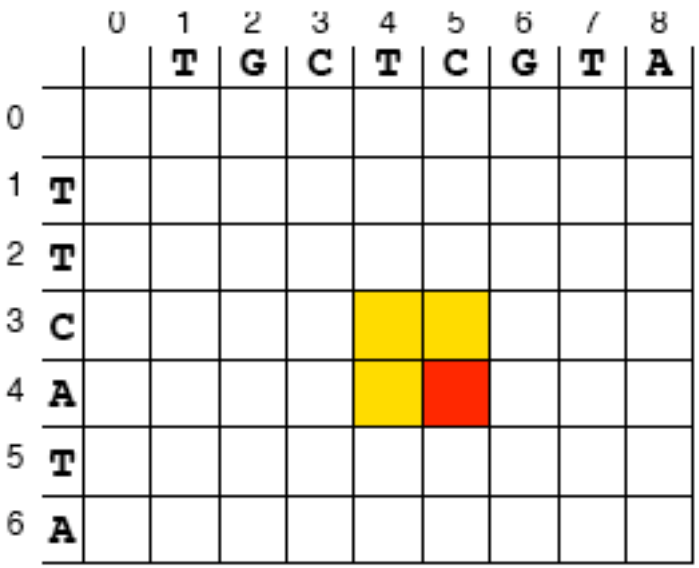
Alignment Score: 18

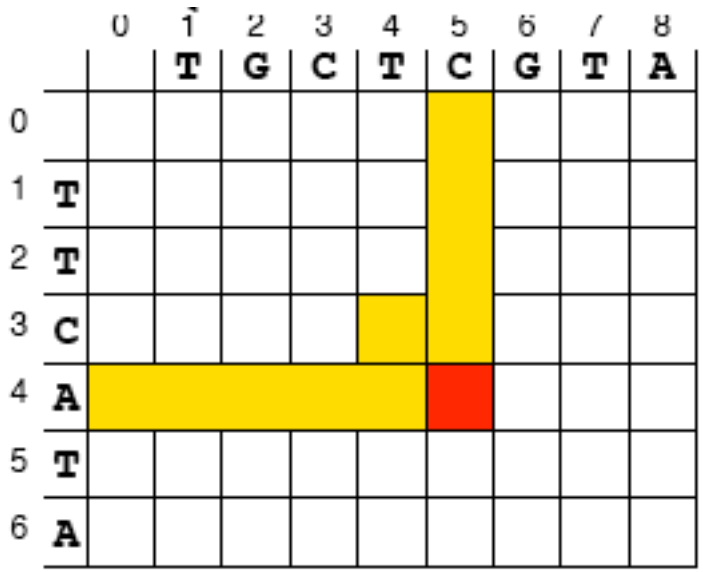
Affine Gap costs

$$g(l) = g_o + l * g_e$$

$$\sigma(i,j) = \max\begin{cases} \sigma(i-1,j-1) + S(a_i,b_j) \\ \sigma(i,j-1) + S(gap,b_j) \\ \sigma(i-1,j) + S(a_i,gap) \end{cases}$$

$$\sigma(i,j) = \max\begin{cases} \sigma(i-1,j-1) + S(a_i,b_j) \\ \max_{k=0}^{i-1}(\sigma(k,j) + g(i-k)),\ gap\ in\ B \\ \max_{k=0}^{j-1}(\sigma(i,k) + g(j-k)),\ gap\ in\ A \end{cases}$$





Alternative Scoring Functions

Blosum62:

	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W	
C	9																				C
S	-1	4																			S
T	-1	1	5																		T
P	-3	-1	-1	7																	P
A	0	1	0	-1	4																A
G	-3	0	-2	-2	0	6															G
N	-3	1	0	-2	-2	0	6														N
D	-3	0	-1	-1	-2	-1	1	6													D
E	-4	0	-1	-1	-1	-2	0	2	5												E
Q	-3	0	-1	-1	-1	-2	0	0	2	5											Q
H	-3	-1	-2	-2	-2	-2	1	-1	0	0	8										H
R	-3	-1	-1	-2	-1	-2	0	-2	0	1	0	5									R
K	-3	0	-1	-1	-1	-2	0	-1	1	1	-1	2	5								K
M	-1	-1	-1	-2	-1	-3	-2	-3	-2	0	-2	-1	-1	5							M
I	-1	-2	-1	-3	-1	-4	-3	-3	-3	-3	-3	-3	-3	1	4						I
L	-1	-2	-1	-3	-1	-4	-3	-4	-3	-2	-3	-2	-2	2	2	4					L
V	-1	-2	0	-2	0	-3	-3	-3	-2	-2	-3	-3	-2	1	3	1	4				V
F	-2	-2	-2	-4	-2	-3	-3	-3	-3	-3	-1	-3	-3	0	0	0	-1	6			F
Y	-2	-2	-2	-3	-2	-3	-2	-3	-2	-1	2	-2	-2	-1	-1	-1	-1	3	7		Y
W	-2	-3	-2	-4	-3	-2	-4	-4	-3	-2	-2	-3	-3	-1	-3	-2	-3	1	2	11	W

PAM250:

C Cys	12																			
S Ser	0	2																		
T Thr	-2	1	3																	
P Pro	-3	1	0	6																
A Ala	-2	1	1	1	2															
G Gly	-3	1	0	-1	1	5														
N Asn	-4	1	0	-1	0	0	2													
D Asp	-5	0	0	-1	0	1	2	4												
E Glu	-5	0	0	-1	0	0	1	3	4											
Q Gln	-5	-1	-1	0	0	-1	1	2	2	4										
H His	-3	-1	-1	0	-1	-2	2	1	1	3	6									
R Arg	-4	0	-1	0	-2	-3	0	-1	-1	1	2	6								
K Lys	-5	0	0	-1	-1	-2	1	0	0	1	0	3	5							
M Met	-5	-2	-1	-2	-1	-3	-2	-3	-2	-1	-2	0	0	6						
I Ile	-2	-1	0	-2	-1	-3	-2	-2	-2	-2	-2	-2	-2	2	5					
L Leu	-6	-3	-2	-3	-2	-4	-3	-4	-3	-2	-2	-3	-3	4	2	6				
V Val	-2	-1	0	-1	0	-1	-2	-2	-2	-2	-2	-2	-2	2	4	2	4			
F Phe	-4	-3	-3	-5	-5	-5	-4	-6	-5	-5	-2	-4	-5	0	1	2	-1	9		
Y Tyr	0	-3	-3	-5	-3	-5	-2	-4	-4	-4	0	-4	-4	-2	-1	-1	-2	7	10	
W Trp	-8	-2	-5	-6	-6	-7	-4	-7	-7	-5	-3	2	-3	-4	-5	-2	-6	0	0	17
	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W

Many others...

Exact vs. Heuristic searches

Both, Needleman-Wunsch and Smith-Waterman alignment methods are **exact** methods since they guarantee a globally optimal solution for the optimization problem!

Drawback: Computational expensive, i.e. O(nm) in time and memory

Exact vs. Heuristic searches

Solutions:

- omit regions from the grid, that cannot contribute to the optimal alignment (reduction of the search space, by remaining exact)

		0	1	2	3	4	5	6	7	8
			T	G	C	T	C	G	T	A
0		0	−6	−12	−18	−24	−30	−36	−42	−48
1	T	−6	5	−1	−7	−13	−19	−25	−31	−37
2	T	−12	−1	3	−3	−2	−8	−14	−20	−26
3	C	−18	−7	−3	8	2	3	−3	−9	−15
4	A	−24	−13	−9	2	6	0	1	−5	−4
5	T	−30	−19	−15	−4	7	4	−2	6	0
6	A	−36	−25	−21	−10	1	5	2	0	11

Exact vs. Heuristic searches

Solutions:

- use of heuristics (more rigorous reduction of the search space, sacrificing the guaranteed optimal solution for search speed)

Hashing

- Lookup method for finding an alignment

```
Pos:    1  2  3  4  5  6  7  8  9 10 11
Seq 1:  k  c  s  p  t  a  .  .  .  .  .
Seq 2:  .  .  .  .  .  a  c  s  p  r  k
```

Amino acid	Pos in Seq 1	Pos in Seq 2	Offset
k	1	11	10
c	2	7	-5
s	3	8	-5
p	4	9	-5
t	5	-	-
a	6	6	0
r	-	10	-

Hashing

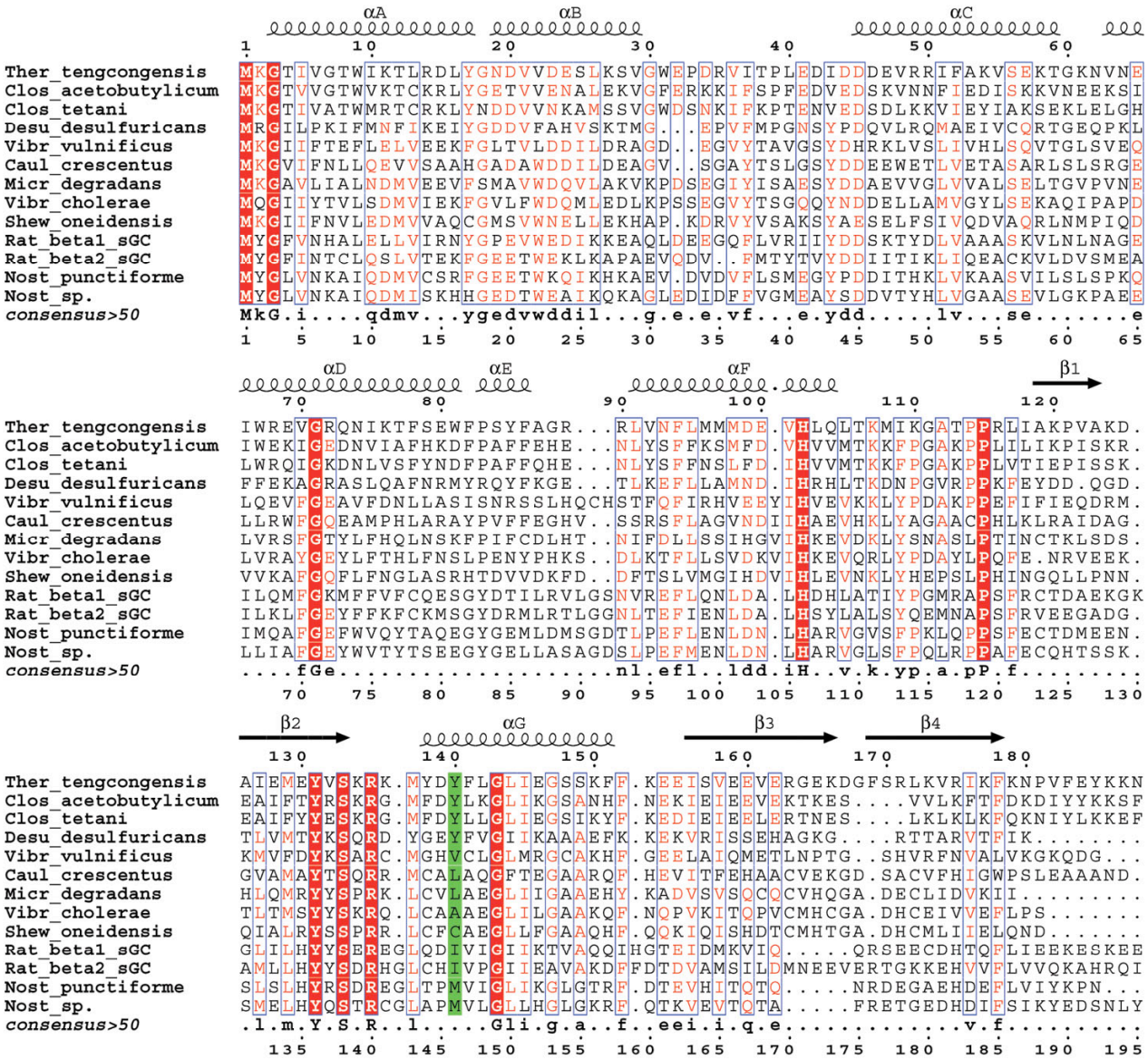
- Lookup method for finding an alignment

```
Pos:    1  2  3  4  5  6  7  8  9 10 11
Seq 1:  k  c  s  p  t  a  .  .  .  .  .
Seq 2:  .  .  .  .  .  a  c  s  p  r  k
```

Amino acid	Pos in Seq 1	Pos in Seq 2	Offset
k	1	11	10
c	2	7	-5
s	3	8	-5
p	4	9	-5
t	5	-	-
a	6	6	0
r	-	10	-

```
Resulting alignment: Seq 1: k  c  s  p  t  a
                     Seq 2: a  c  s  p  r  k
```

What we are really looking for:

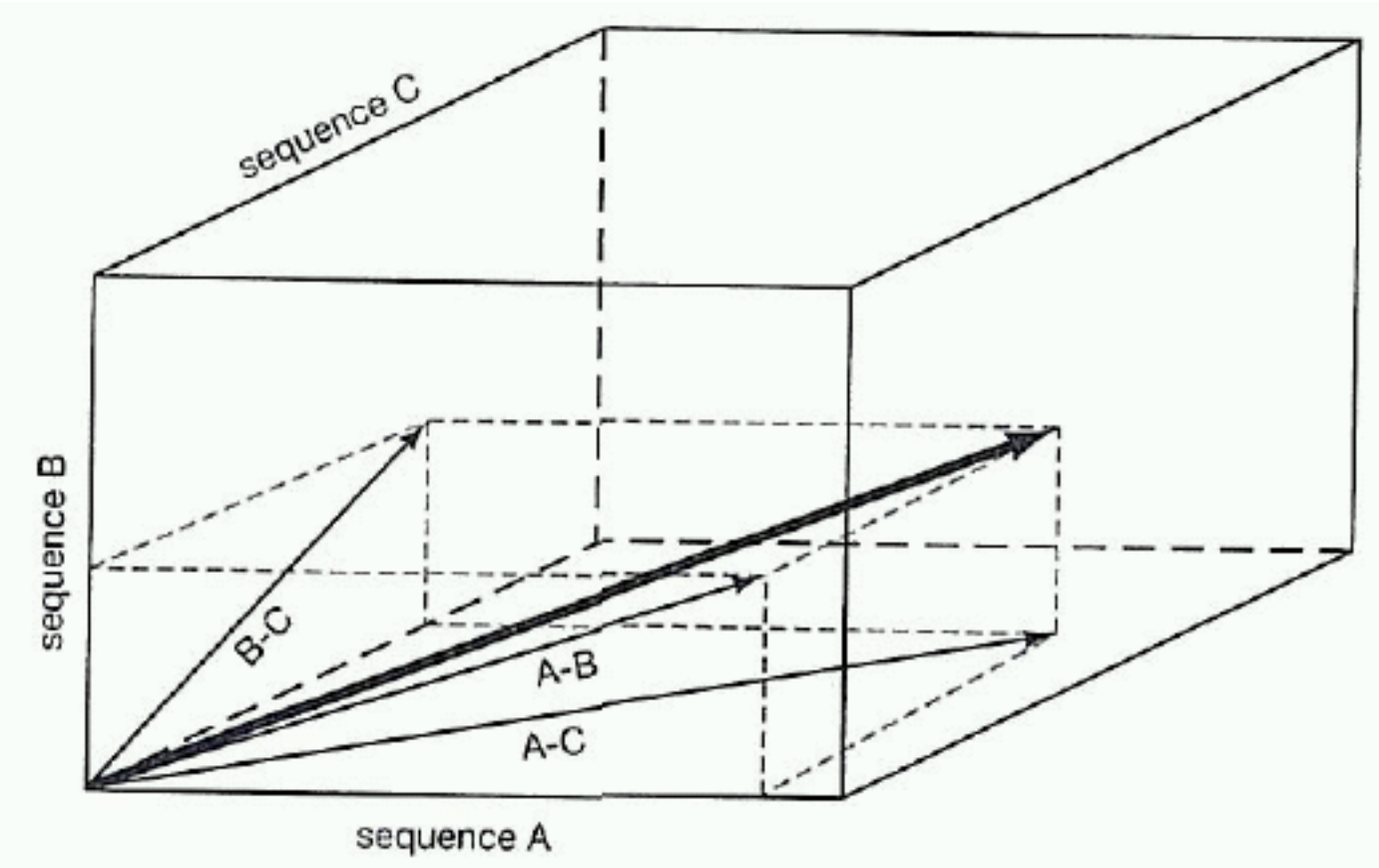


How to construct Multiple Sequence Alignments?

Optimal Solution:
Extend Needleman-Wunsch or Smith-Waterman to multiple sequences

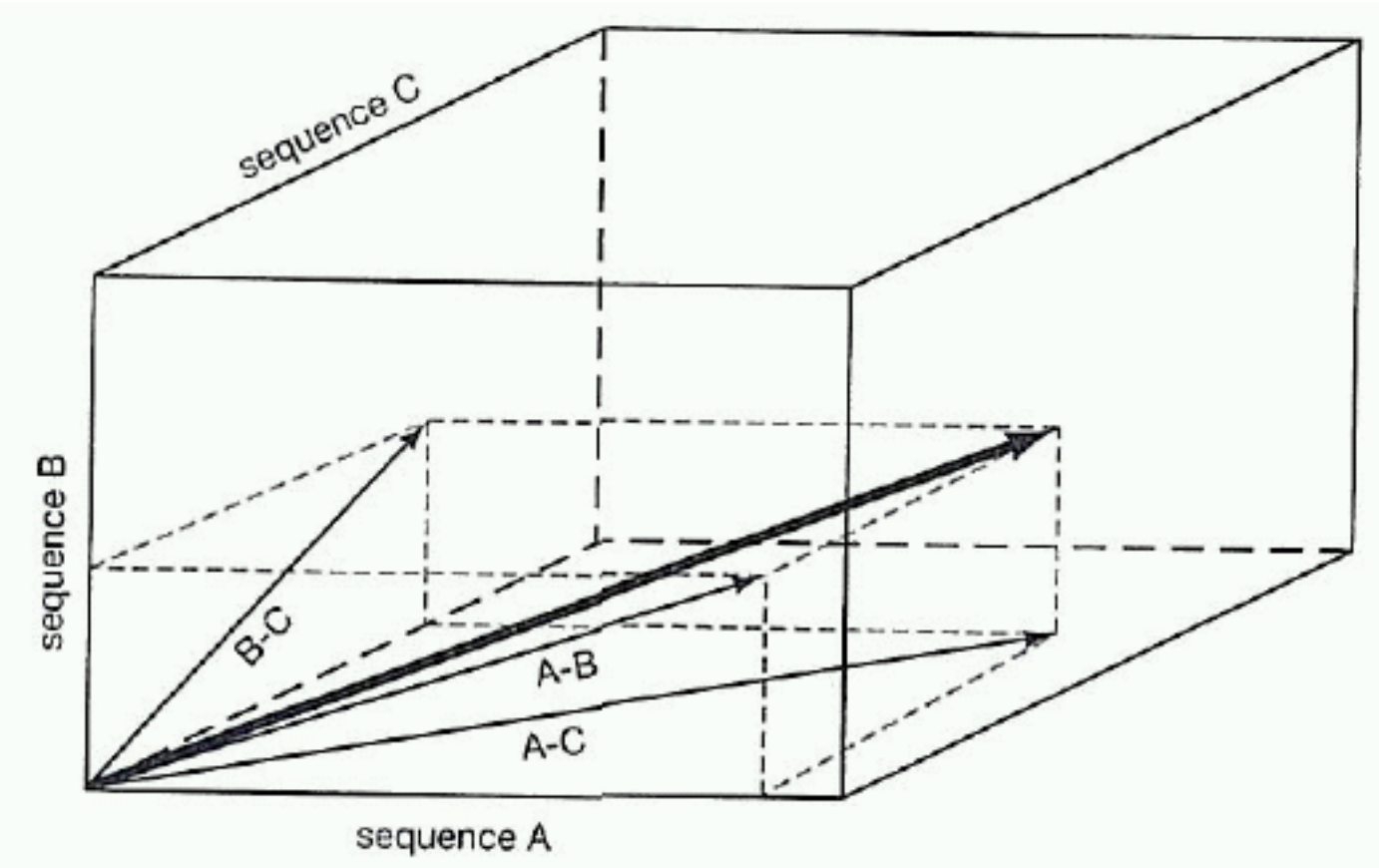
How to construct Multiple Sequence Alignments?

Optimal Solution:
Extend Needleman-Wunsch or Smith-Waterman to multiple sequences



How to construct Multiple Sequence Alignments?

Optimal Solution:
Extend Needleman-Wunsch or Smith-Waterman to multiple sequences



But $O(n^m)$ in time and memory:
Computationally not feasible... 4 sequences of length 1000 -> 1TB RAM

A new objective function: Sum of Pairs

```
Seq1: AGA--CTA
Seq2: G-A--CTT
Seq3: AGAAACTT
```

A new objective function: Sum of Pairs

```
Seq1: AGA--CTA
Seq2: G-A--CTT
Seq3: AGAAACTT
```

```
Seq1: AGA--CTA
Seq2: G-A--CTT
```

```
Seq1: AGA--CTA
Seq3: AGAAACTT
```

```
Seq2: G-A--CTT
Seq3: AGAAACTT
```

$$S(a_i,b_j) = \begin{cases} +5, \text{ if } a_i = b_j \\ -2, \text{ if } a_i \neq b_j \\ -6, \text{ for introduction of a gap} \end{cases}$$

```
Seq1: AGA--CTA
Seq2: G-A--CTT
```

Score: +5

```
Seq1: AGA--CTA
Seq3: AGAAACTT
```

Score: +11

```
Seq2: G-A--CTT
Seq3: AGAAACTT
```

Score: 0

A new objective function: Sum of Pairs

```
Seq1: AGA--CTA
Seq2: G-A--CTT
Seq3: AGAAACTT
```

```
Seq1: AGA--CTA
Seq2: G-A--CTT
```

```
Seq1: AGA--CTA
Seq3: AGAAACTT
```

```
Seq2: G-A--CTT
Seq3: AGAAACTT
```

$$S(a_i, b_j) = \begin{cases} +5, \text{ if } a_i = b_j \\ -2, \text{ if } a_i \neq b_j \\ -6, \text{ for introduction of a gap} \end{cases}$$

```
Seq1: AGA--CTA
Seq2: G-A--CTT
```

Score: +5

```
Seq1: AGA--CTA
Seq3: AGAAACTT
```

Score: +11

```
Seq2: G-A--CTT
Seq3: AGAAACTT
```

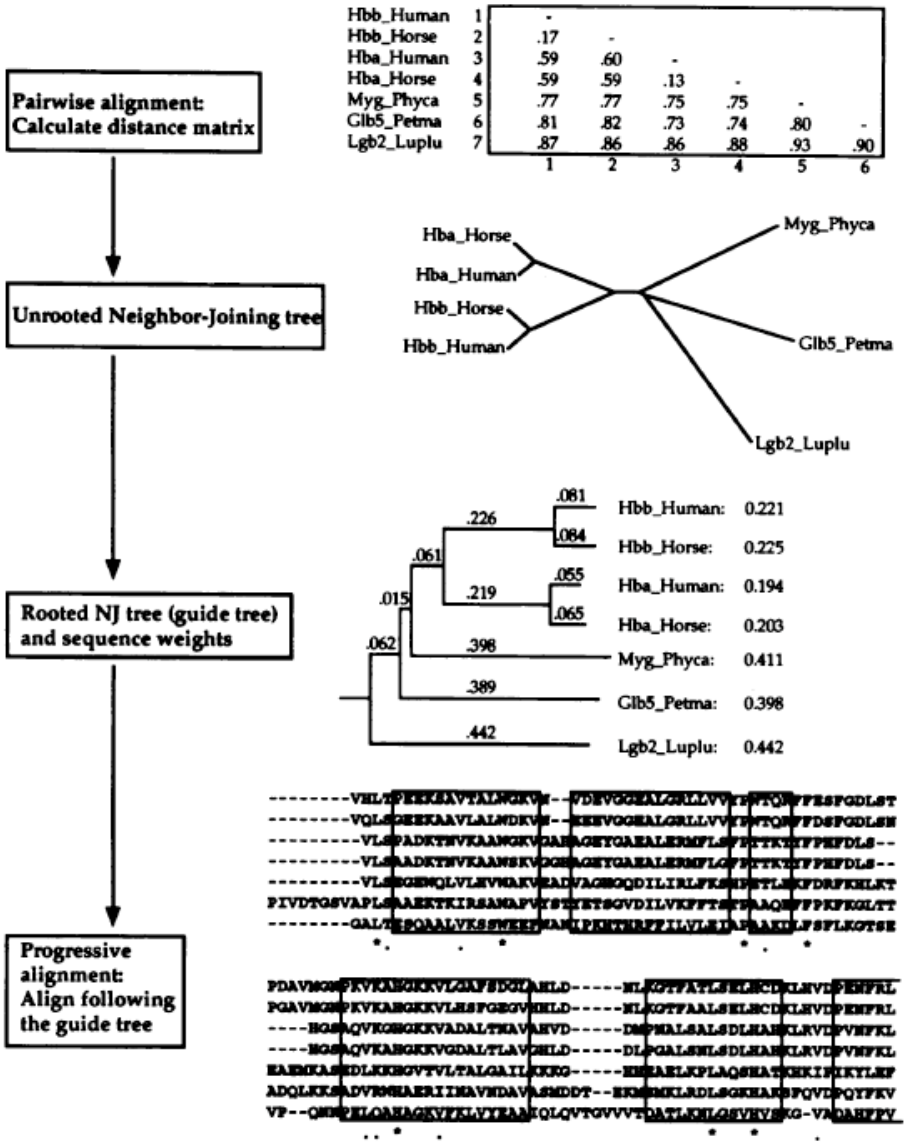
Score: 0

SUM OF PAIRS SCORE: 16

Progressive Alignment Strategies (ClustalW)

- The sequences are added stepwise. Thus, never more than two sequences (or multiple sequence alignments) are simultaneously aligned
- Sequences or MSAs are aligned using **Dynamic Programming**

Progressive Alignment Strategies (ClustalW)



Scoring for the alignment of two alignments

$$\sigma(a^i, b^j) = \frac{1}{n+m} \sum_{x=1}^{n} \sum_{y=1}^{m} S(a_x^i, b_y^j) \times \omega_x \times \omega_y$$

$\sigma(a^i,b^j)$:	score for aligning column i from alignment (or sequence) ***a*** to column j from alignment or sequence ***b***
n,m	number of sequences in alignments ***a*** and ***b***, respectively
$S(a_x^i, b_y^j)$	score for aligning position ***i*** in sequence ***x*** from alignment ***a*** to position ***j*** in sequence ***y*** from alignment ***b***
ω_x, ω_y	respective weights of the sequences ***x*** and ***y***

Scoring for the alignment of two alignments

$$\sigma(a^i, b^j) = \frac{1}{n+m} \sum_{x=1}^{n} \sum_{y=1}^{m} S(a_x^i, b_y^j) \times \omega_x \times \omega_y$$

```
1 peeksavtal
2 geekaavlal
3 padktnvkaa
4 aadktnvkaa

4 egewglvlhv
5 aaektkirsa
```

With sequence weights:

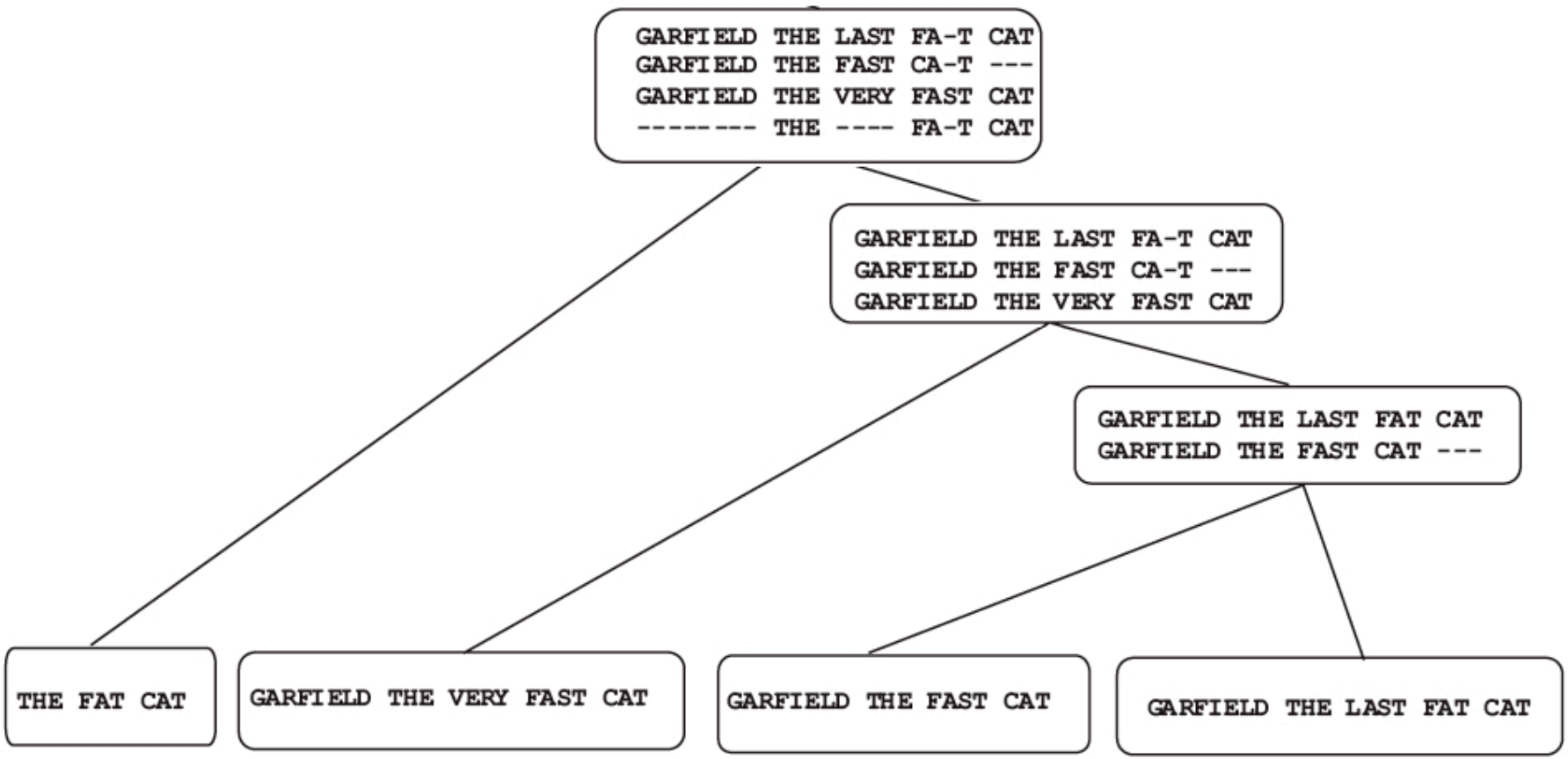
Score = $(S(t,v)*\omega_1\omega_5$
+ $S(t,i)*\omega_1\omega_6$
+ $S(l,v)*\omega_2\omega_5$
+ $S(l,i)*\omega_2\omega_6$
+ $S(k,v)*\omega_3\omega_5$
+ $S(k,i)*\omega_3\omega_6$
+ $S(k,v)*\omega_4\omega_5$
+ $S(k,i)*\omega_4\omega_6)/8$

Features of ClustalW

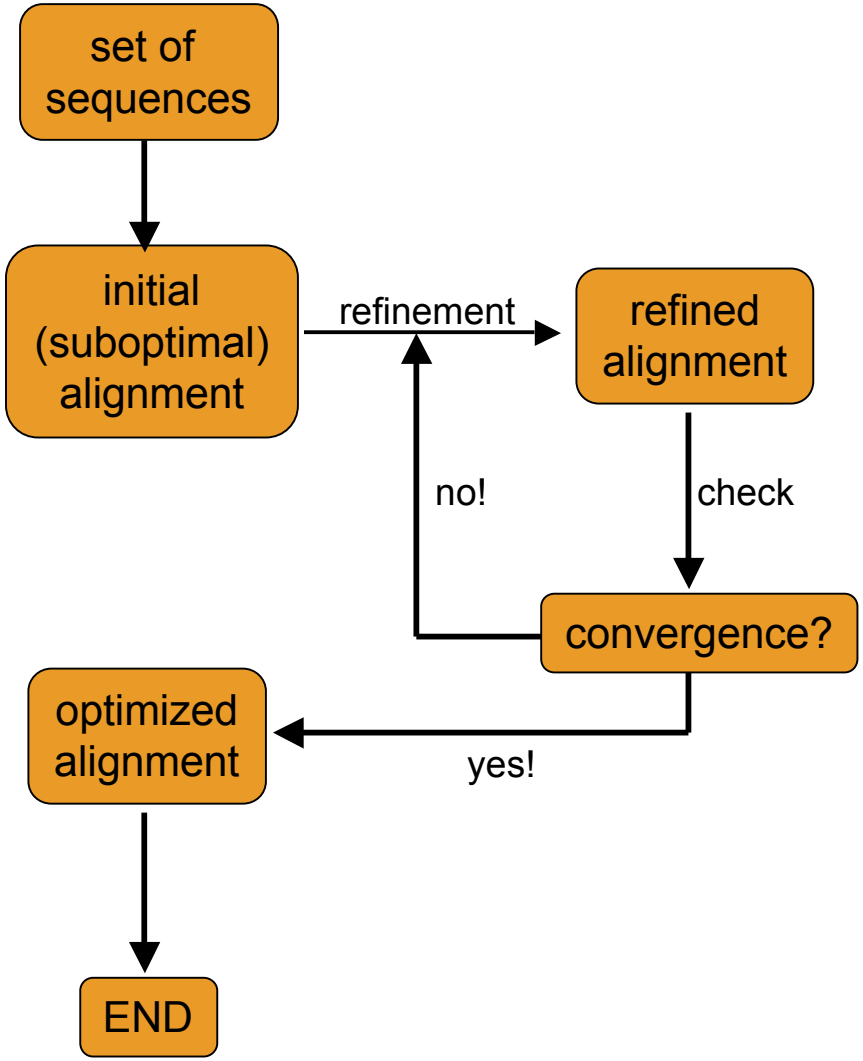
- progressive strategy
- Distance based generation of a guide tree (approximative or exact)
- tree-guided (NJ) alignment
- change of the scoring matrix as the alignment proceeds (adaptation to increasing divergence of the sequences
- dynamic variation of gap penalties in position- and residue-specific manner
  - gap opening penalties are locally reduced in stretches of 5 or more hydrophilic residues (indicative of loop or random coil regions).
  - gap penalties are locally increased within eight residues of existing gaps.
- sequence weighting

(Known) Problem of ClustalW: Local Optima

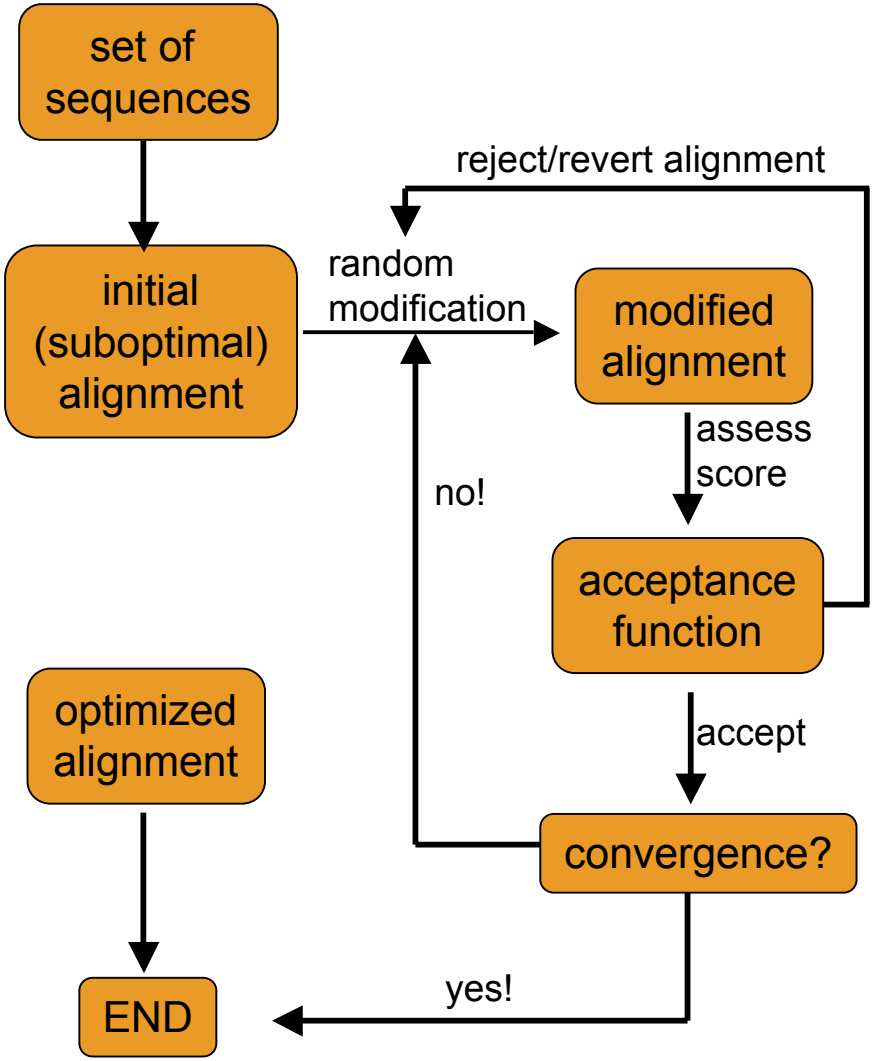
a.k.a: Once a gap always a gap



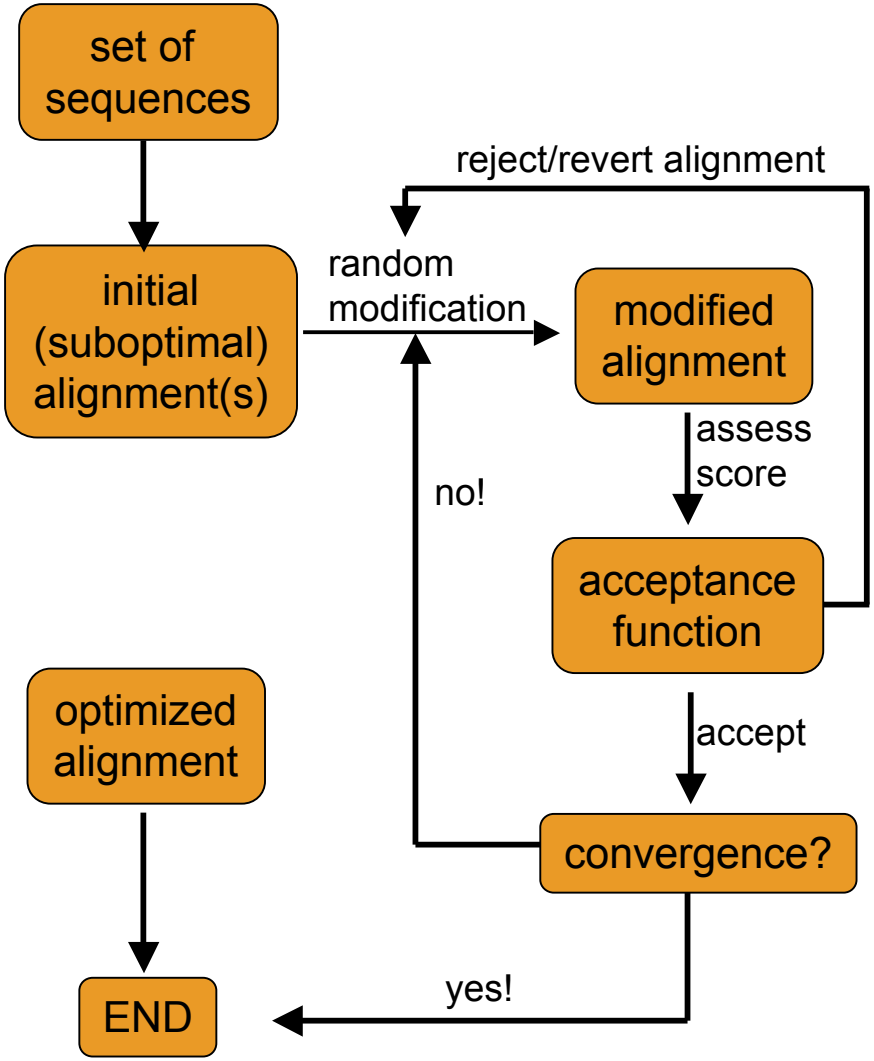
Iterative Alignment Strategy



Stochastic Iterative Alignment

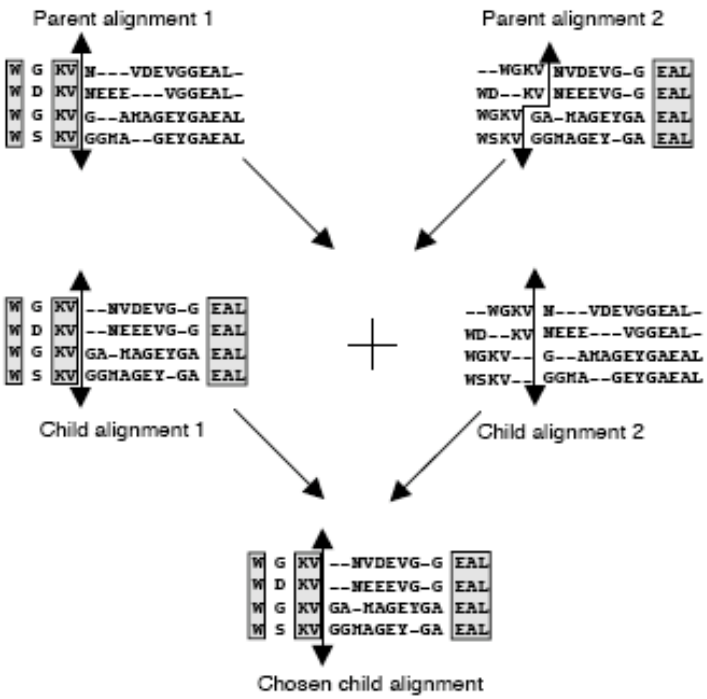


Stochastic Iterative Alignment (SAGA)



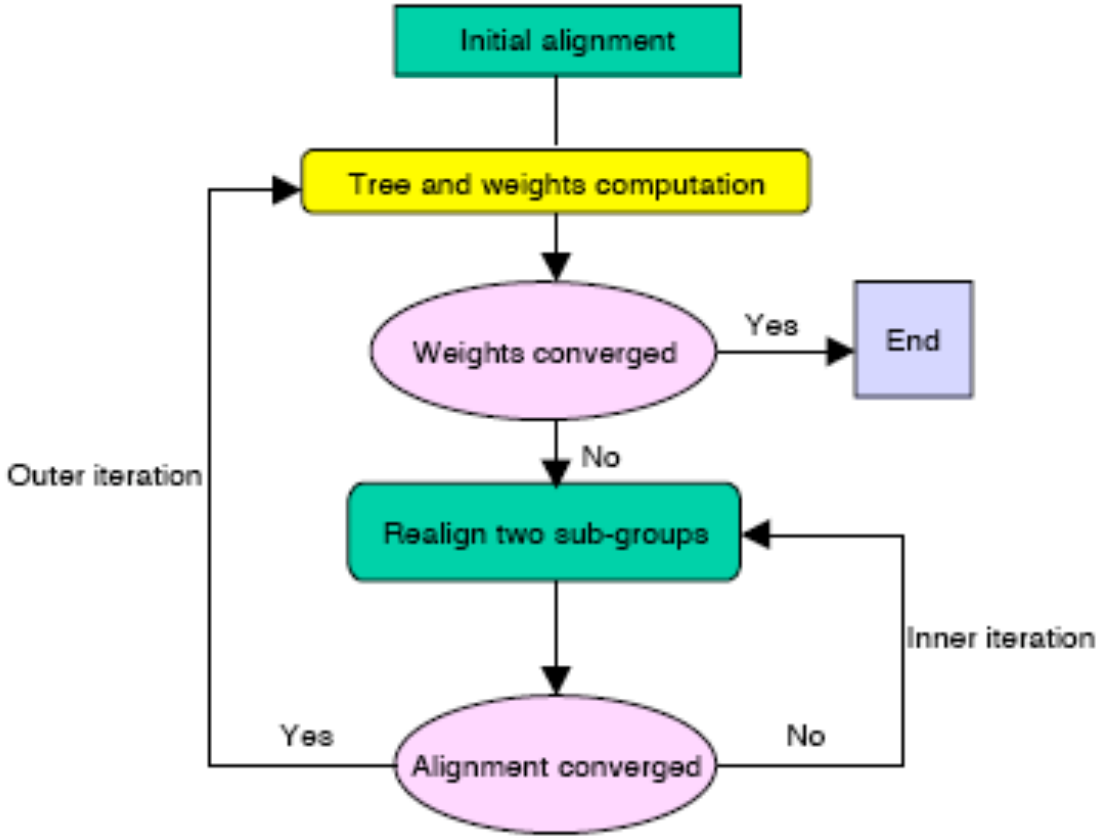
Genetic Algorithm:

- Alignments evolve by 'mutation' and crossing over
- alignments score determines fitness
- over the generations, alignments survive and reproduce or die



Non-Stochastic Iterative Alignment

Point: The initial alignment is modified by splitting it into two groups and re-aligning them with dynamic programming.



Example: Prrp, both, alignment (inner loop) and tree/weight (outer loop) are optimized.

Consistency based algorithm

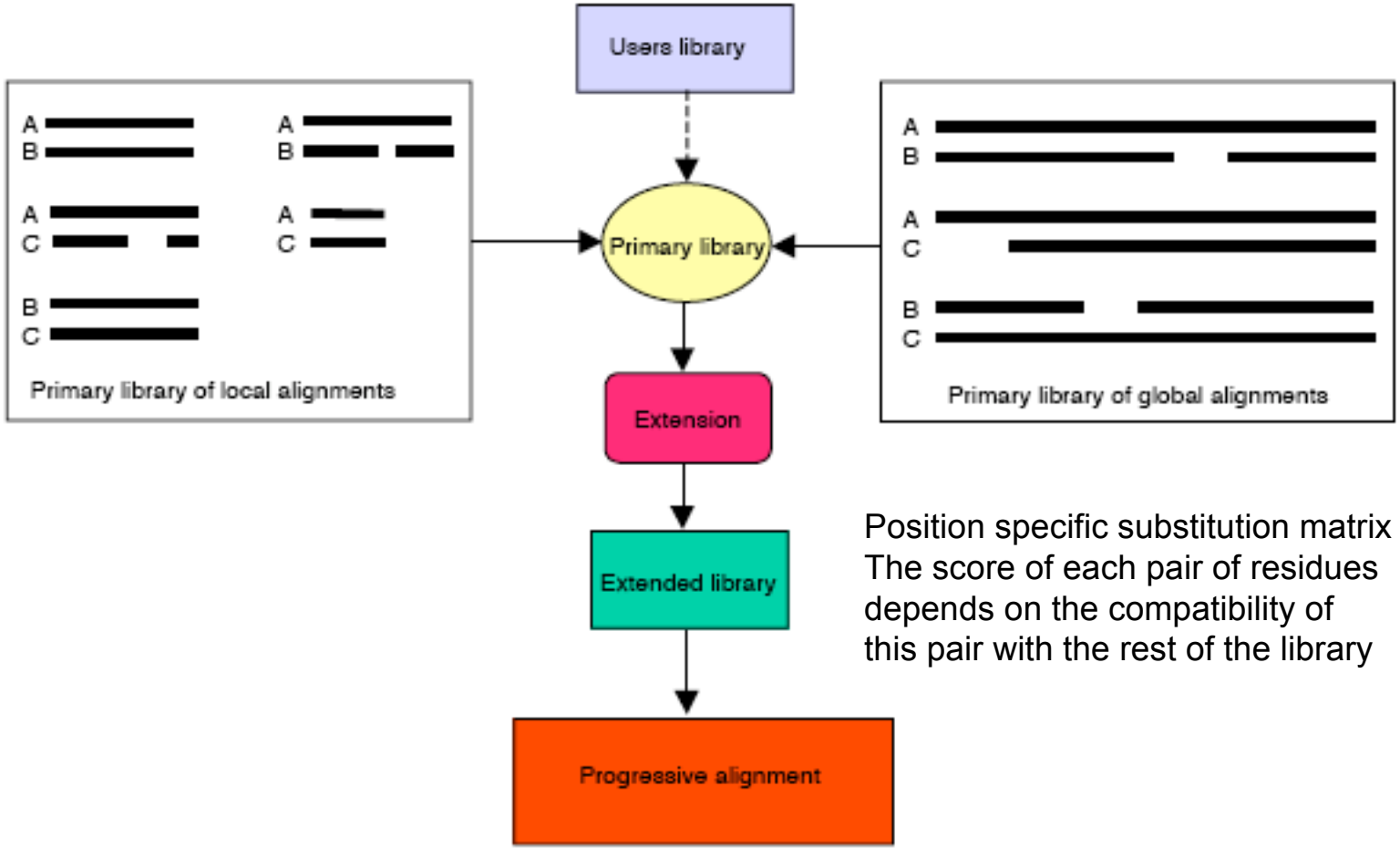
Point: The optimal MSA is defined as the one that agrees the most with all optimal pair-wise alignments

Features:

- does not depend on a specific substitution rate
- can apply any method capable to align two sequences
- position dependent, i.e. the score associated with the alignment of two residues depends on their position within the sequence rather that their individual nature
- rationale: given a set of independent observations, the constellation most often observed is often closer to the truth

Consistency based **O**bjective **F**unction **F**or align**E**ment **E**valuation (COFFEE)

The Principle of T-Coffee



Position specific substitution matrix
The score of each pair of residues depends on the compatibility of this pair with the rest of the library

A comparison

CIBIV MFPL

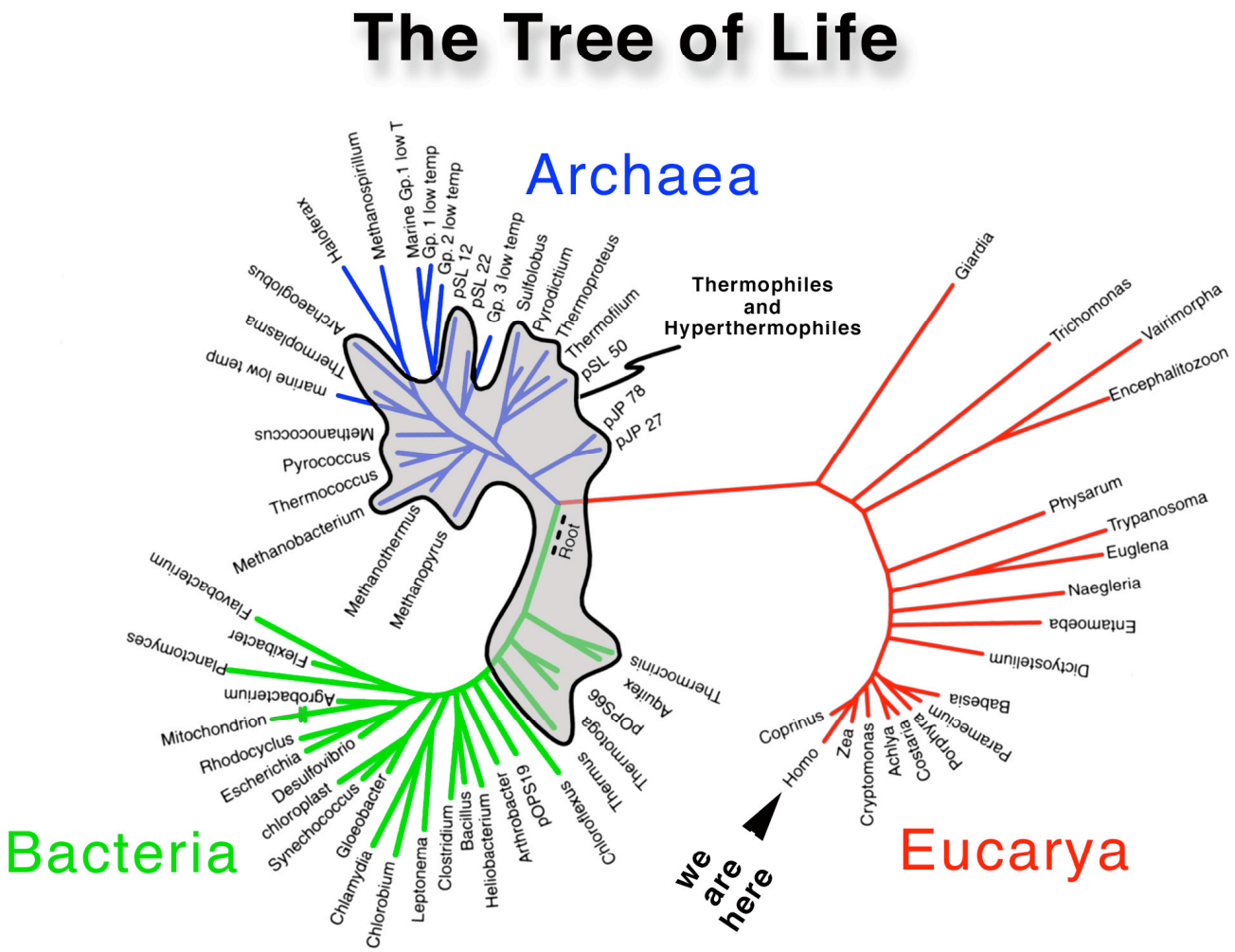
Table 2. Some elements of validation on BAliBASE.

Method	Ref1	Ref2	Ref3	Ref4	Ref5	Total
DiAlign	71.0	25.2	35.1	74.7	80.4	57.3
ClustalW	78.5	32.2	42.5	65.7	74.3	58.7
Prrp	78.6	32.5	50.2	51.1	82.7	59.0
T-Coffee	80.7	37.3	52.9	83.2	88.7	68.7

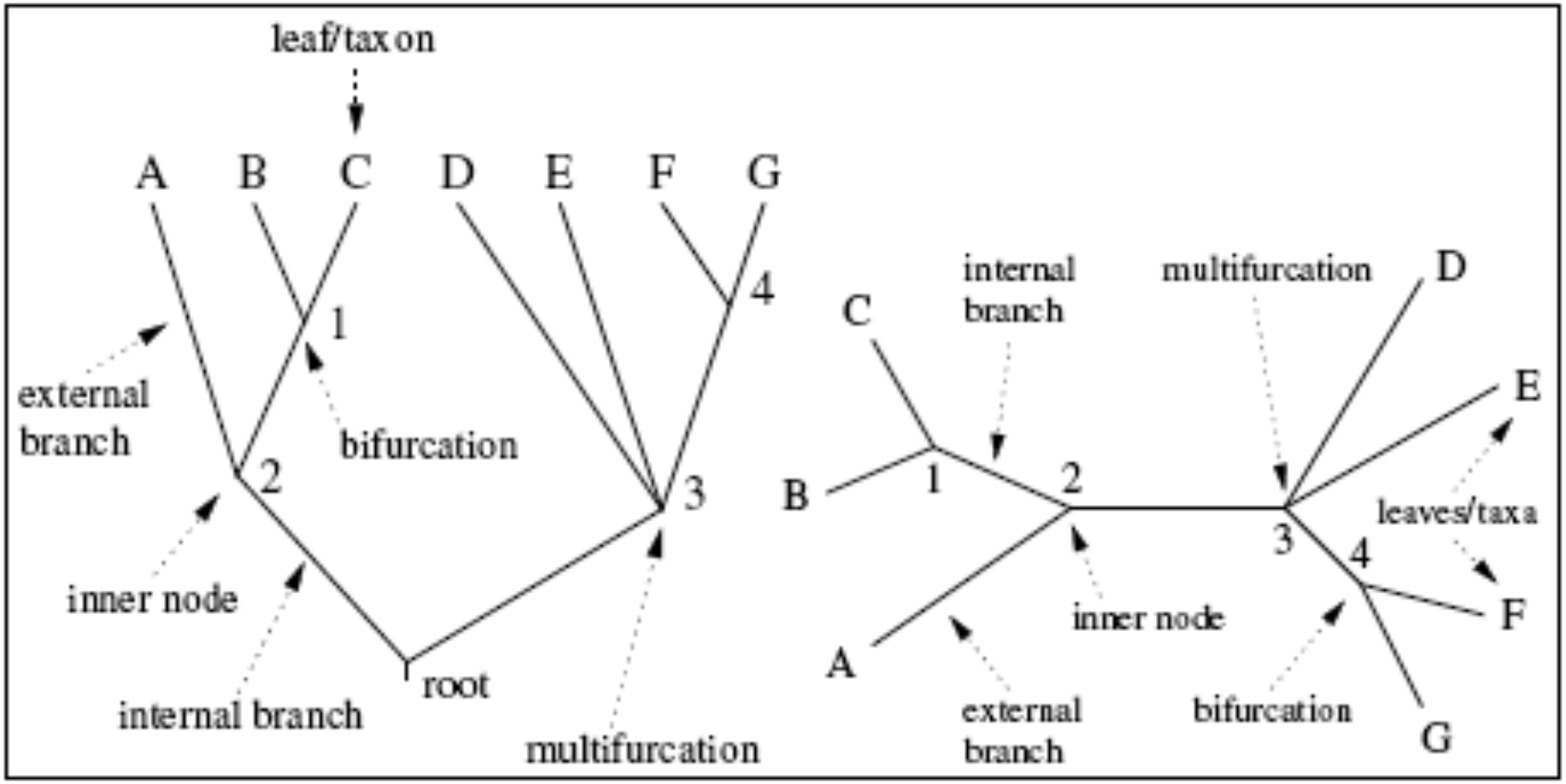
Each method in the Method column was used to align the 141 test-sets contained in BAliBASE. The alignments were then compared with the reference BAliBASE alignment using aln_compare [34]. Ref1–5 indicates the five BAliBASE categories. Results obtained in each category were averaged. All the observed differences are statistically significant, as assessed by the Wilcoxon rank-based test [34,47]. Ref1 contains a homogenous set of sequences, ref2 contains a homogenous group of sequences and an outlayer, ref3 contains two distantly related groups of sequences. Ref4 contains sequences that require long internal gaps to be properly aligned and ref5 contains sequences that require long-terminal gaps to be properly aligned. Total is the average of ref1–5.

Reconstructing Trees from Sequences

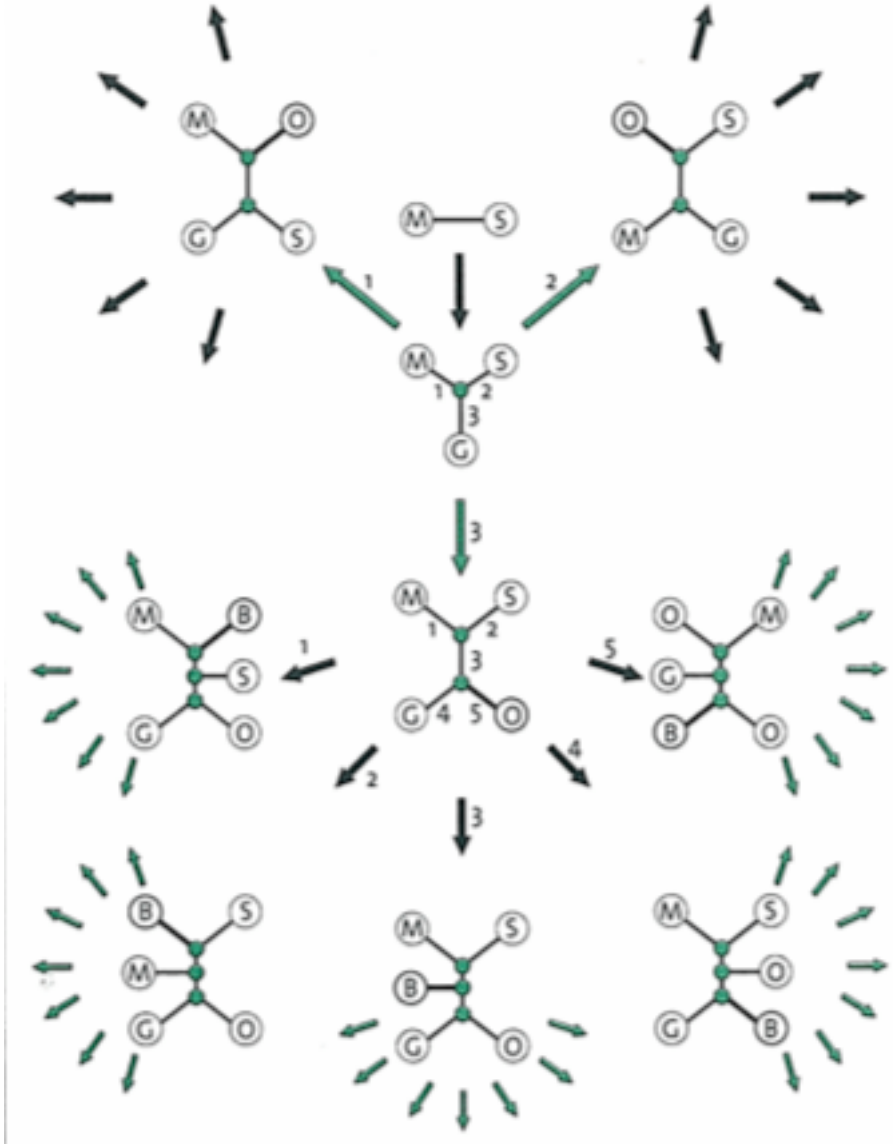
Reconstructing the tree of life



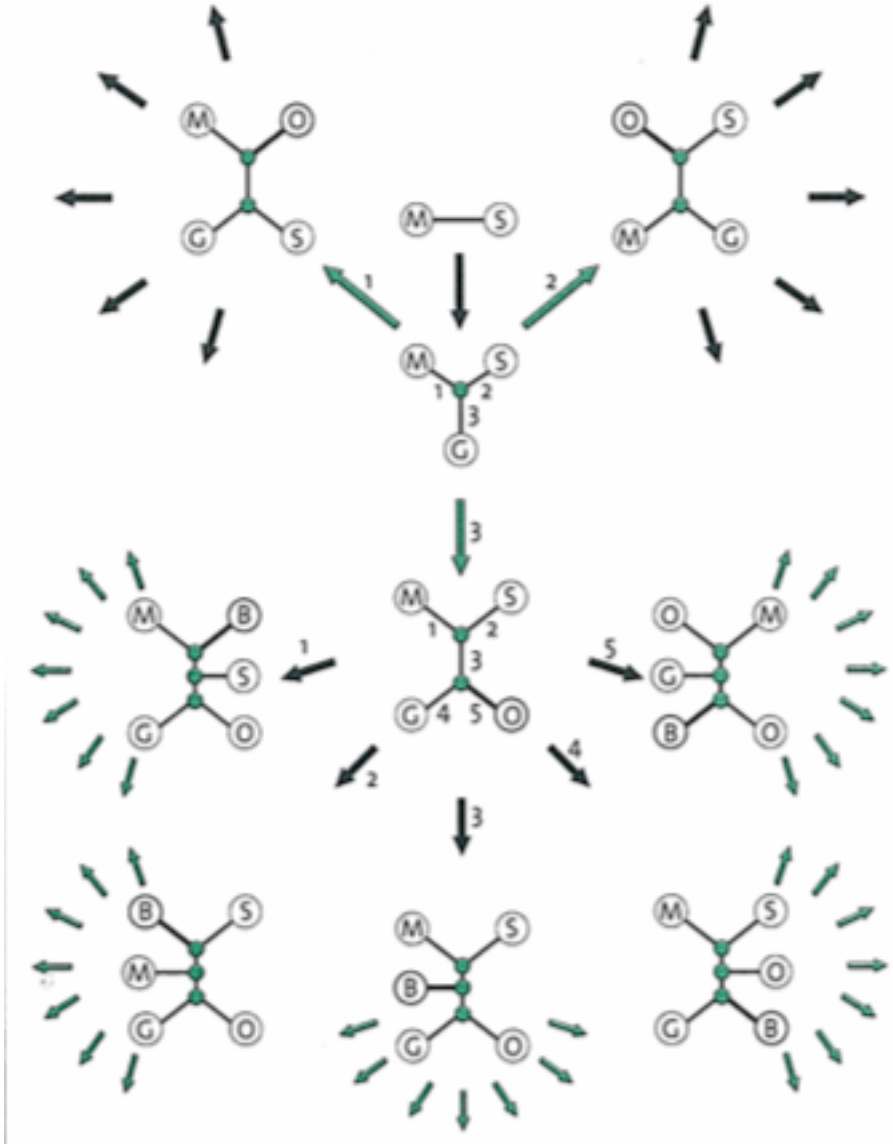
Notations


leaf/taxon
A
B
C
D
E
F
G
1
2
3
4
external branch
bifurcation
inner node
internal branch
root
multifurcation
internal branch
multifurcation
C
B
A
D
E
F
G
1
2
3
4
leaves/taxa
inner node
external branch
bifurcation

How many possible trees are there?



How many possible trees are there?



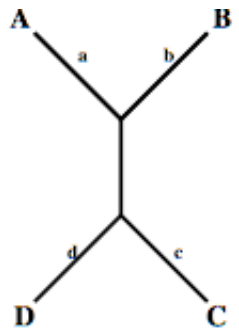
$$b(n) = \frac{(2n-5)!}{2^{n-3}(n-3)!}$$

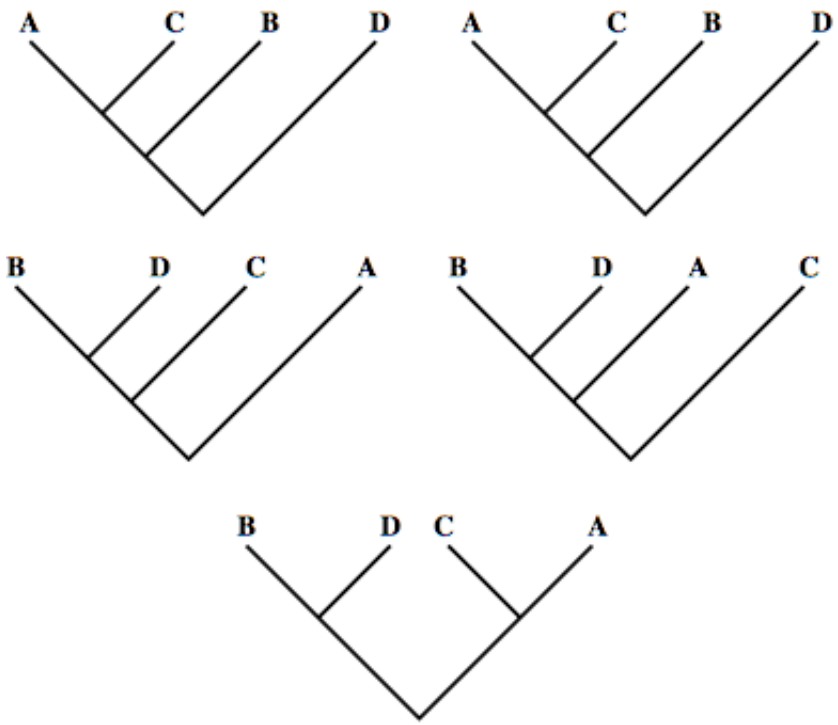
$$b(10) = 2027025$$

$$b(55) = 2.9 \times 10^{84}$$

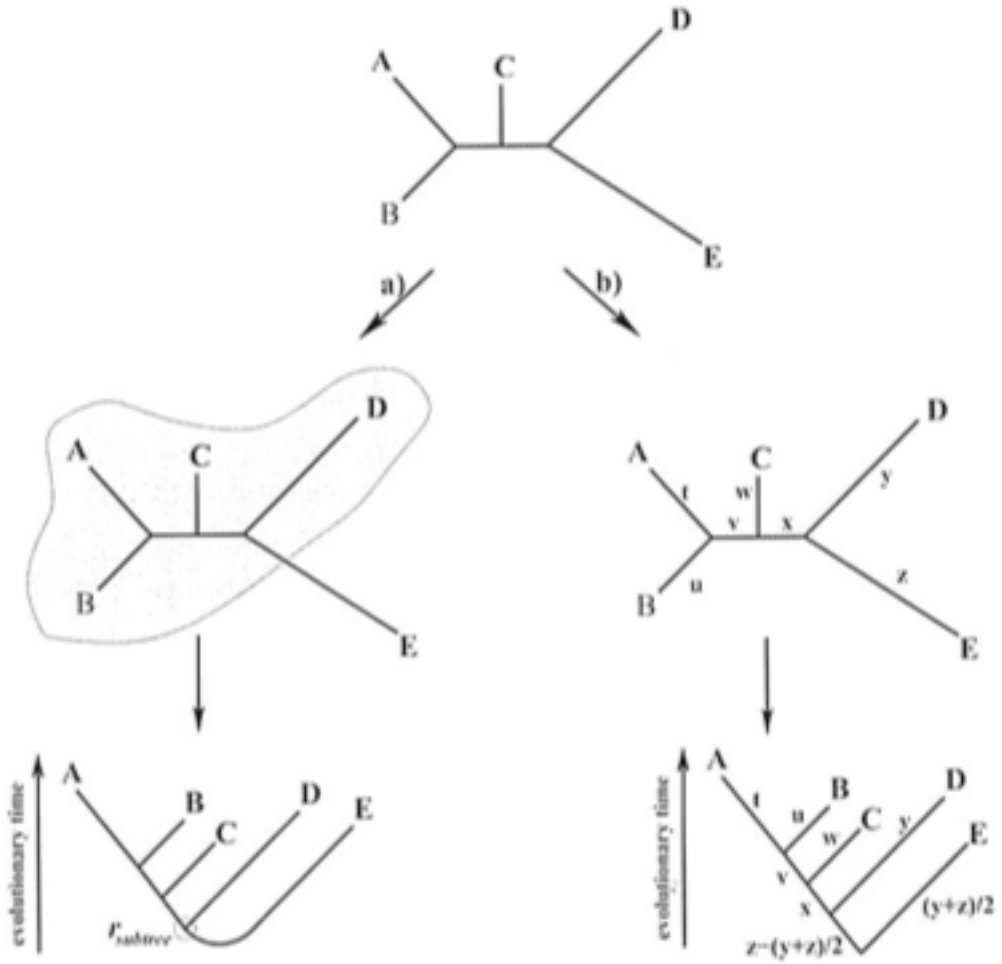
$$b(100) = 1.7 \times 10^{182}$$

Finding the root of the tree





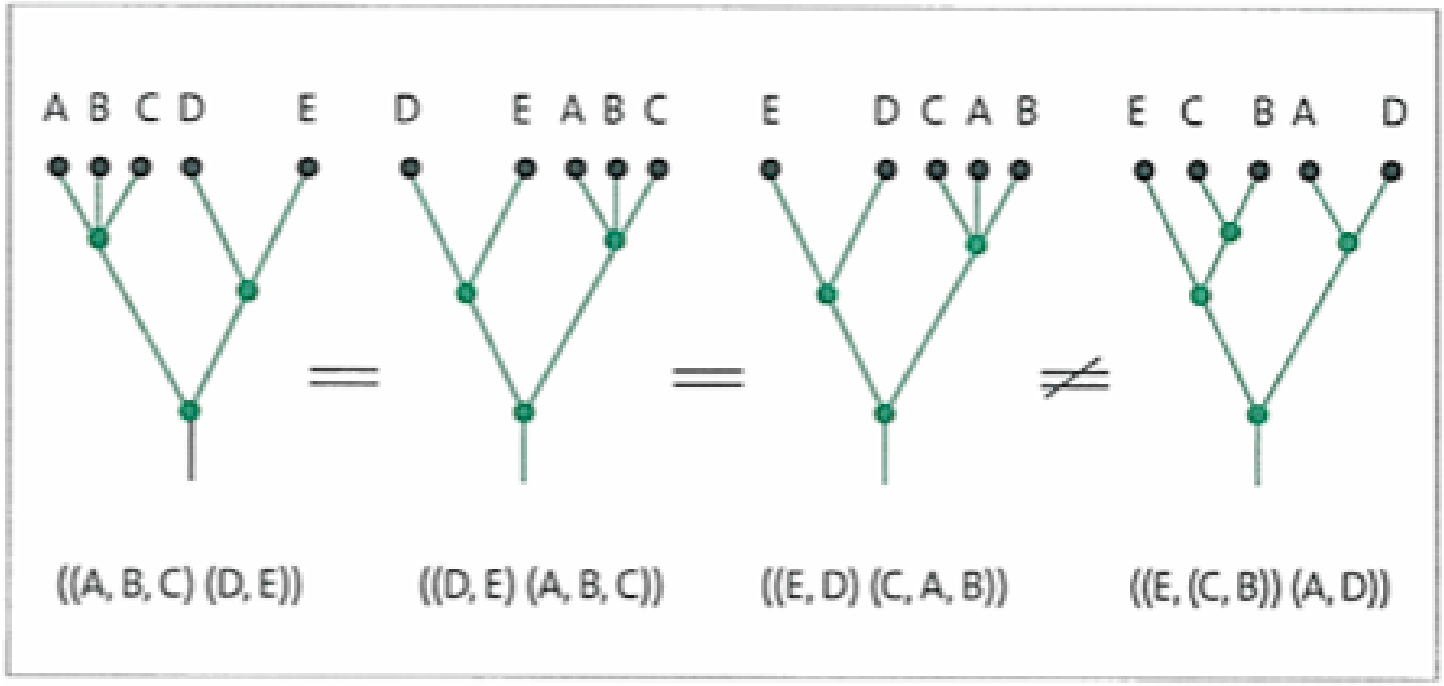
Finding the root of the tree



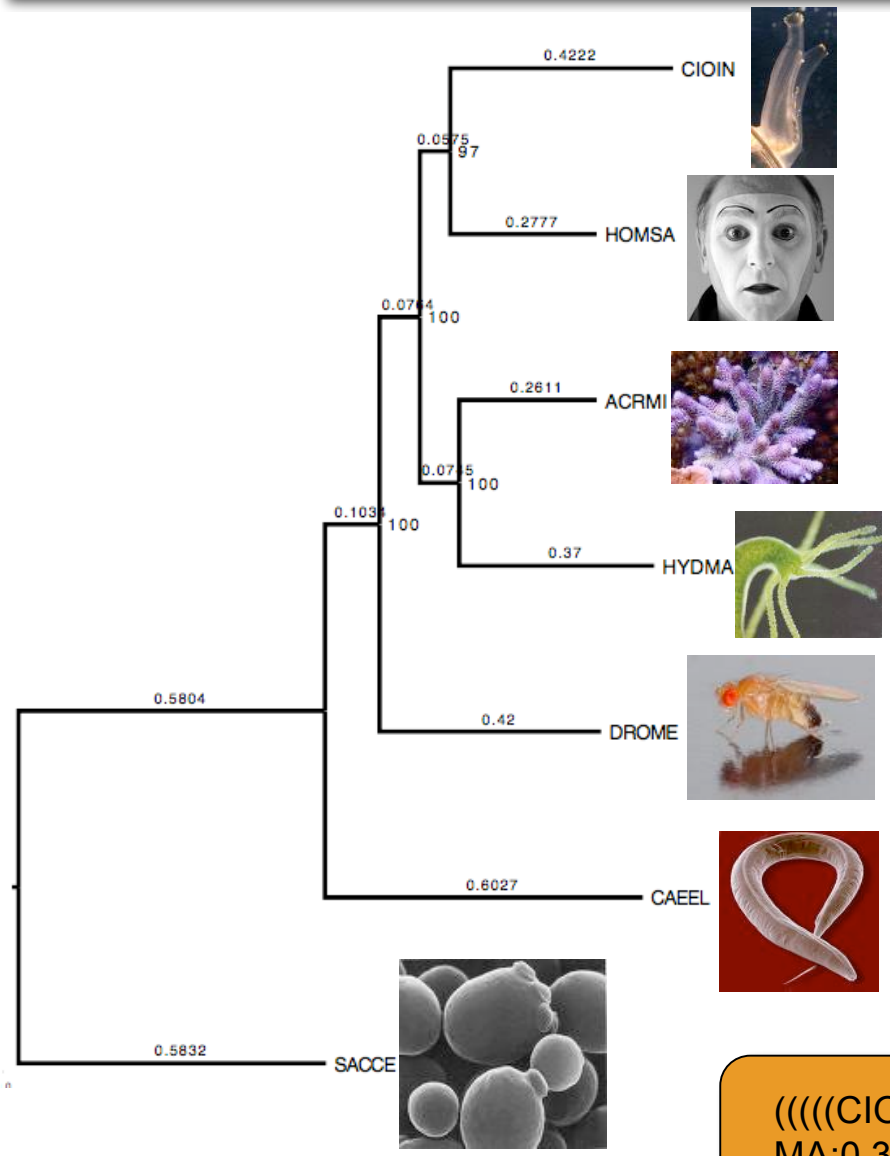
Outgroup-routing

Midpoint-routing

Tree Formats



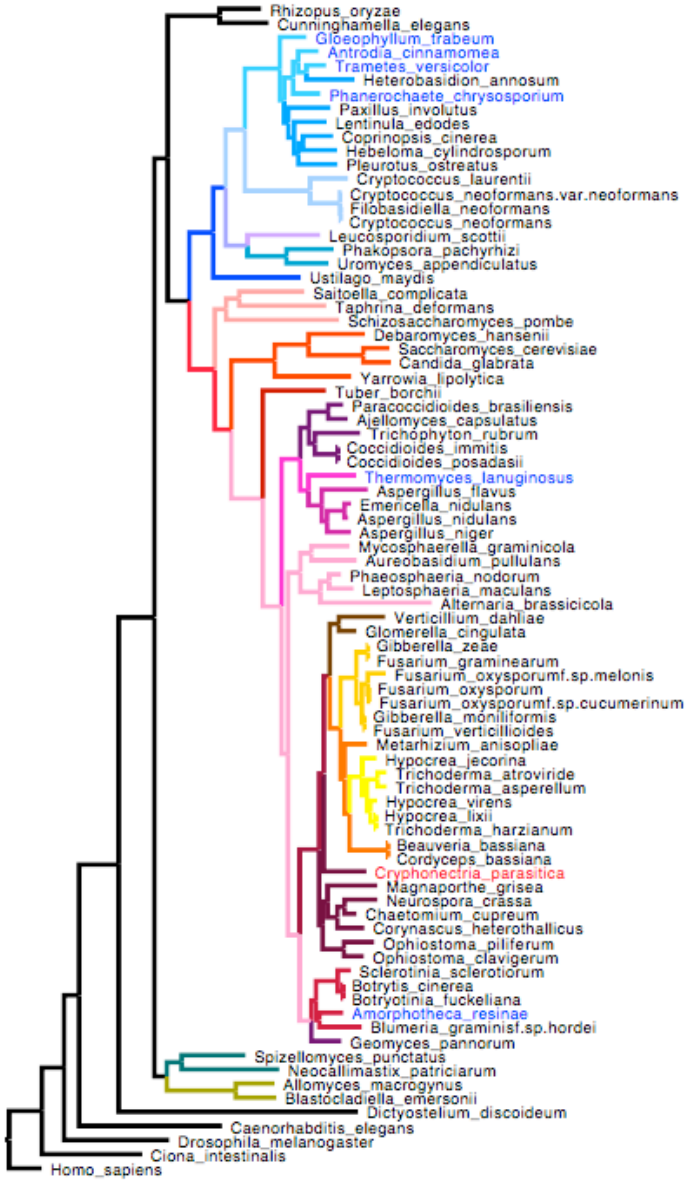
Three typical representations of the same tree




```
#NEXUS
begin taxa;
        dimensions ntax=7;
        taxlabels
        DROME
        CIOIN
        HYDMA
        SACCE
        CAEEL
        ACRMI
        HOMSA
;
end;

begin trees;
        tree [&r] tree_1 = (((((CIOIN:0.4222,HOMSA:0.2777)
[&label=97]:0.0575,
(ACRMI:0.2611,HYDMA:0.37)[&label=100]:0.0745)
[&label=100]:0.0764,
DROME:0.42)[&label=100]:0.1034,CAEEL:0.6027):0.5804,
SACCE:0.5832);
end;

begin figtree;
        set appearance.backgroundColour=#-1;
end figtree;
```

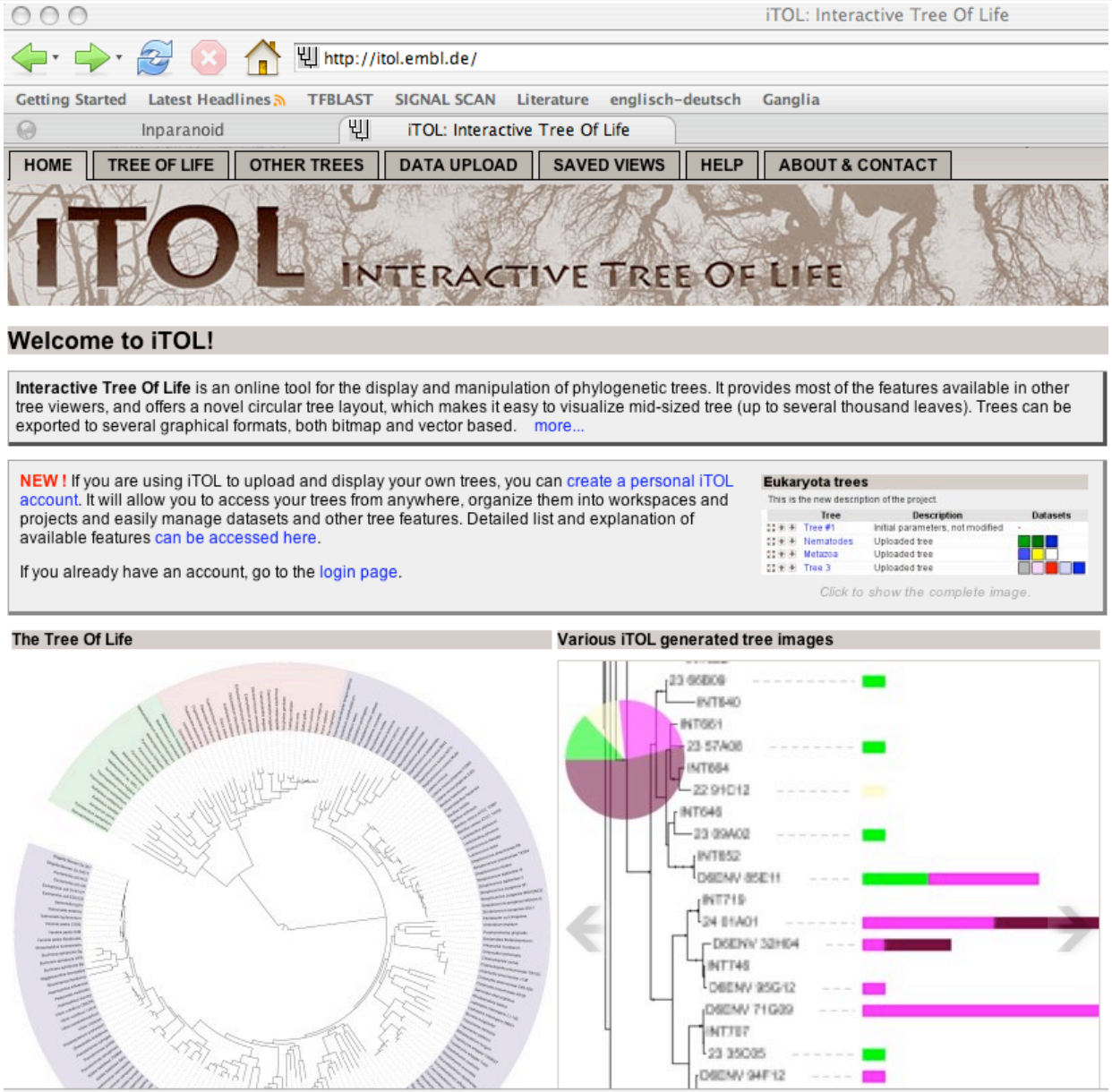
```
(((((CIOIN:0.4222,HOMSA:0.2777)97:0.0575,(ACRMI:0.2611,HYD
MA:0.3700)100:0.0745)100:0.0764,DROME:0.4200)100:0.1034,
CAEEL:0.6027):0.5804,SACCE:0.5832);
```

Complex trees

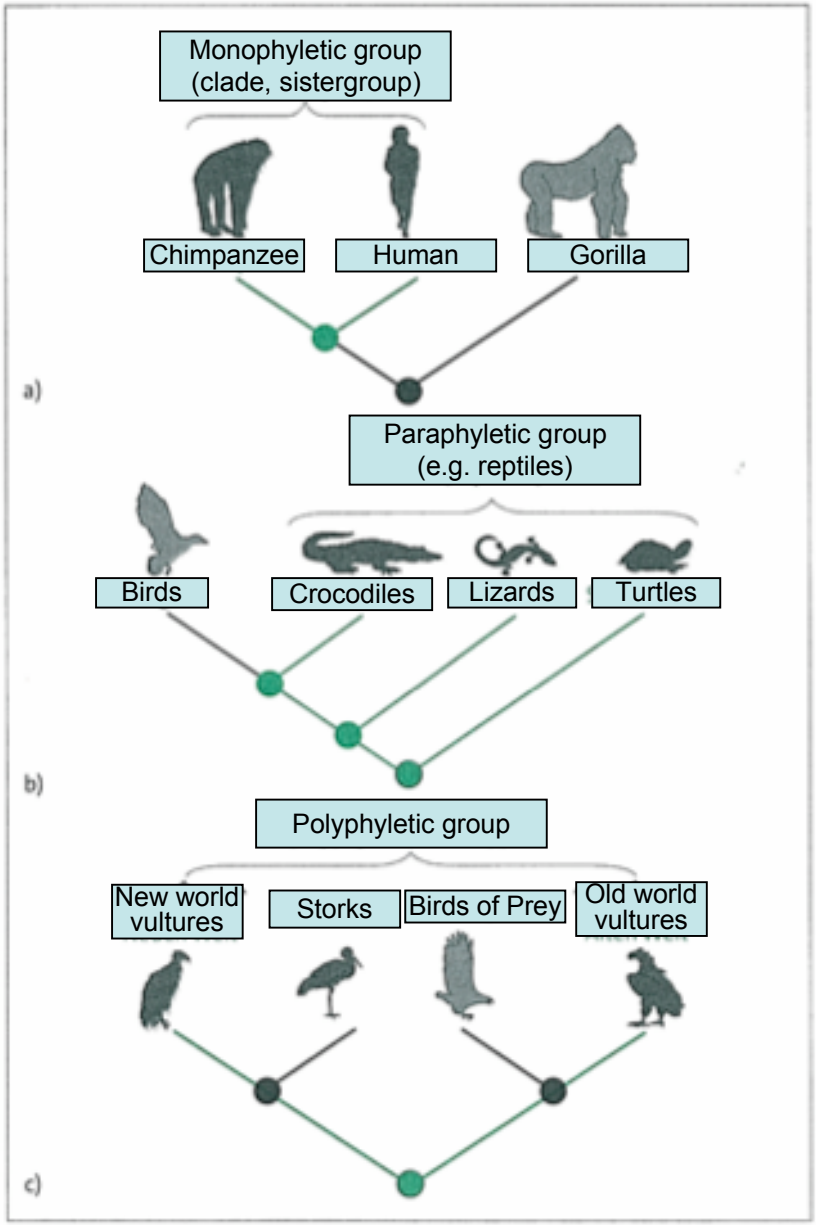



```
(((((((Rhizopus_oryzae:0.12338953118750640991,Cunninghamella_elegans:0.14505462914970748689)100:0.15326253481223306441,((((
(Gloeophyllum_trabeum:0.07924812458330700304,(((Antrodia_cinnamomea:0.06125179490008491462,(Trametes_versicolor:0.05859793008498467848,Het
erobasidion_annosum:0.14265163114578949255)100:0.02449899345507637605)100:0.01491996194868821894,Phanerochaete_chrysosporium:0.08142901435
819373257)100:0.02639850355850581476,(Paxillus_involutus:0.12507930174040107763,(Lentinula_edodes:0.09578001724170893960,(((Coprinopsis_cin
erea:0.08955096569898440828,Hebeloma_cylindrosporum:0.10085378088044757994)100:0.01366340764780078071,Pleurotus_ostreatus:0.11634253457851
556623)100:0.01235187547388479599)100:0.01711017593272916998)100:0.01240279259156270436)100:0.01259472218900263042)100:0.10333051305697128
763,(Cryptococcus_laurentii:0.11261447295515064626,(Cryptococcus_neoformans.var.neoformans:0.00521778134981755806,(Filobasidiella_neoforma
ns:0.00237156362893179741,Cryptococcus_neoformans:0.00009861600456473006)100:0.00212229753415758490)100:0.08816498134973375922)100:0.19263
372956588356577)100:0.05656266668807796022,(Leucosporidium_scottii:0.21499585065698342823,(Phakopsora_pachyrhizi:0.13976400349035095205,Ur
omyces_appendiculatus:0.12460741232125499556)100:0.11477257364870681322)100:0.06460589202493333127)100:0.03684017462332440235,Ustilago_may
dis:0.25934488225560420327)100:0.07030546047039644419,(((Saitoella_complicata:0.19266317415358455434,Taphrina_deformans:0.2570307906593770
0642)100:0.03883872443363502874,Schizosaccharomyces_pombe:0.33367484237741701358)100:0.03345088821164741988,((((Debaromyces_hansenii:0.1660
7000789678139085,(Saccharomyces_cerevisiae:0.07821550711422489699,Candida_glabrata:0.08167802849027459844)100:0.16114727512467122428)100:0
.09989437041082575852,Yarrowia_lipolytica:0.22680534204146188260)100:0.12819522989497894594,(Tuber_borchii:0.18713075564612405288,((((Para
coccidioides_brasiliensis:0.04492412585008645487,Ajellomyces_capsulatus:0.05846155095529741164)100:0.05682479020008983361,(Trichophyton_ru
brum:0.13478802145894250297,(Coccidioides_immitis:0.00009861600456473006,Coccidioides_posadasii:0.00013275597489488007)100:0.0679130311481
0437469)100:0.01752185514248328843)100:0.02570153410152175091,(Thermomyces_lanuginosus:0.14047726034864610467,(Aspergillus_flavus:0.13625
86602121553559,((Emericella_nidulans:0.01993789235659861395,Aspergillus_nidulans:0.01174995531954378149)100:0.05110312543732857538,ASPNI:0
.07010364253857682970)100:0.01568149707407409318)100:0.03916392516601811119)100:0.02443387059988460716)100:0.05504556307901582041,(((Mycos
phaerella_graminicola:0.11382805992322068966,Aureobasidium_pullulans:0.13479608922737240650)100:0.03222541190278756240,((Phaeosphaeria_nod
orum:0.03601370297481657629,Leptosphaeria_maculans:0.07294024018301276113)100:0.02604153023893227351,Alternaria_brassicicola:0.32946258067
893718913)100:0.05892130233221552771)100:0.03664059956326781720,(((((Verticillium_dahliae:0.13743026980278238991,Glomerella_cingulata:0.05
26623717878586736 2)100:0.02659336222615297021,(((Gibberella_zeae:0.00009861600456473006,Fusarium_graminearum:0.00009861600456473006)100:0.
0341471005725198090 1,((Fusarium_oxysporumf.sp.melonis:0.05303860967867460308,(Fusarium_oxysporum:0.00009861600456473006,Fusarium_oxysporum
f.sp.cucumerinum:0.00473059578454169526)100:0.00357223653413873228)100:0.01712573793927751192,(Gibberella_moniliformis:0.00009861600456473
006,Fusarium_verticillioides:0.00009861600456473006)100:0.00755012976207619646)100:0.01660747744541156770)100:0.04668007882194781522,(Meta
rhizium_anisopliae:0.06747750588644643721,((Hypocrea_jecorina:0.04281168204986635084,((Trichoderma_atroviride:0.02758662551141205857,Trich
oderma_asperellum:0.02793167656539112878)100:0.03084253786435780034,(Hypocrea_virens:0.01967361655791701955,(Hypocrea_lixii:0.00009861600
4 56473006,Trichoderma_harzianum:0.00009861600456473006)100:0.01640815227372549590)100:0.01031552333099211745)100:0.01407183499215602582)100
:0.03728293661804039927,(Beauveria_bassiana:0.00009861600456473006,Cordyceps_bassiana:0.00017480976326447167)100:0.11379015715330136060)10
0:0.01457733600595170070)100:0.01214032674197134364)100:0.03171283347589626267)100:0.01752107533795072003,Cryphonectria_parasitica:0.12585
408688986942434)100:0.01404062191232230070,((Magnaporthe_grisea:0.14740839006608871808,((Neurospora_crassa:0.09790422357055546254,Chaetomi
um_cupreum:0.03622272647936440604)80:0.02088625388322065332,Corynascus_heterothallicus:0.08006458123554179018)100:0.02903048934692216074)8
0:0.01464997564221958391,(Ophiostoma_piliferum:0.09029085190095893776,Ophiostoma_clavigerum:0.05979185389895070452)100:0.06271238182555409
491)80:0.01044724186692970631)100:0.05716276100425224382,(((Sclerotinia_sclerotiorum:0.03879729201106129483,(Botrytis_cinerea:0.00009861600
456473006,Botryotinia_fuckelianaC:0.00009861600456473006)100:0.01606152746805612069)100:0.06410562334506864079,(Amorphotheca_resinae:0.07
099850624761384899,Blumeria_graminisf.sp.hordei:0.12573210322159533714)100:0.01007185390375272237)100:0.01372048555964769899,Geomyces_pann
orum:0.08942182088142704155)100:0.03434252931156196037)100:0.03464174207103632580)100:0.01879298933467762195)100:0.05538850743967564660)10
0:0.09299628620991588768)100:0.04867625580472154101)100:0.07498954862375889485)100:0.06386475531945318140)100:0.03914280476745674031,((Spi
zellomyces_punctatus:0.18931369689379472212,Neocallimastix_patriciarum:0.28843028482622112829)100:0.04371345560798906016,(Allomyces_macrog
ynus:0.11469719607743417145,Blastocladiella_emersonii:0.11999224967666824448)100:0.20328647132686597510)100:0.03664269039449321 19)100:0.1
05263740867638913 58,Dictyostelium_discoideum:0.70552689158042469764)100:0.11685579381042721092,Caenorhabditis_elegans:0.42189930531844038 7
41)100:0.04434817203516049772,Drosophila_melanogaster:0.31104455613948034376)100:0.06514413900702456517,Ciona_intestinalis:0.2957740483821
7330463)100:0.09667503731183066384,Homo_sapiens:0.09667503731183066384)100;
```

Tree display is an unsolved problem



Some more notations



Character based phylogeny reconstruction:

- A character has to be expressed in at least two states in the taxa under study. Taxa are grouped on the basis of shared character states.
- An evolutionary derived character (state) is called an ***Apomorphy***
- ***Aut-Apomorphy:*** an evolutionary derived character (state) present only in a single taxon
- ***Syn-Apomorphy:*** an evolutionary derived character (state) shared by a group of taxa.
- ***Plesiomorphy:*** an ancestral character (state) shared by a group of extant taxa.
- ***Homoplasy:*** A derived character (state) that is shared for reasons other than common decent.

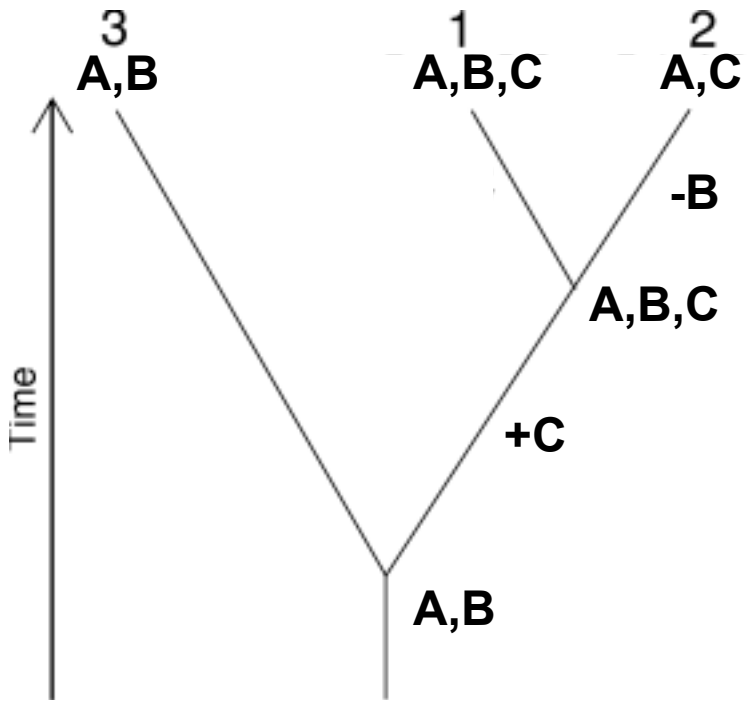
The Parsimony Principle

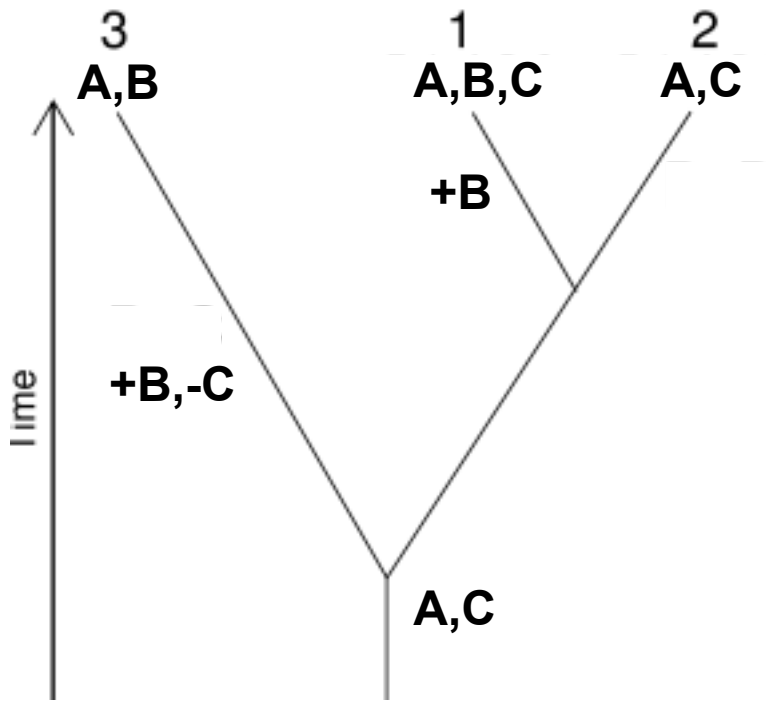
A rule in science and philosophy stating that entities should not be multiplied needlessly.

This rule is interpreted to mean that the simplest of two or more competing theories is preferable and that an explanation for unknown phenomena should first be attempted in terms of what is already known.

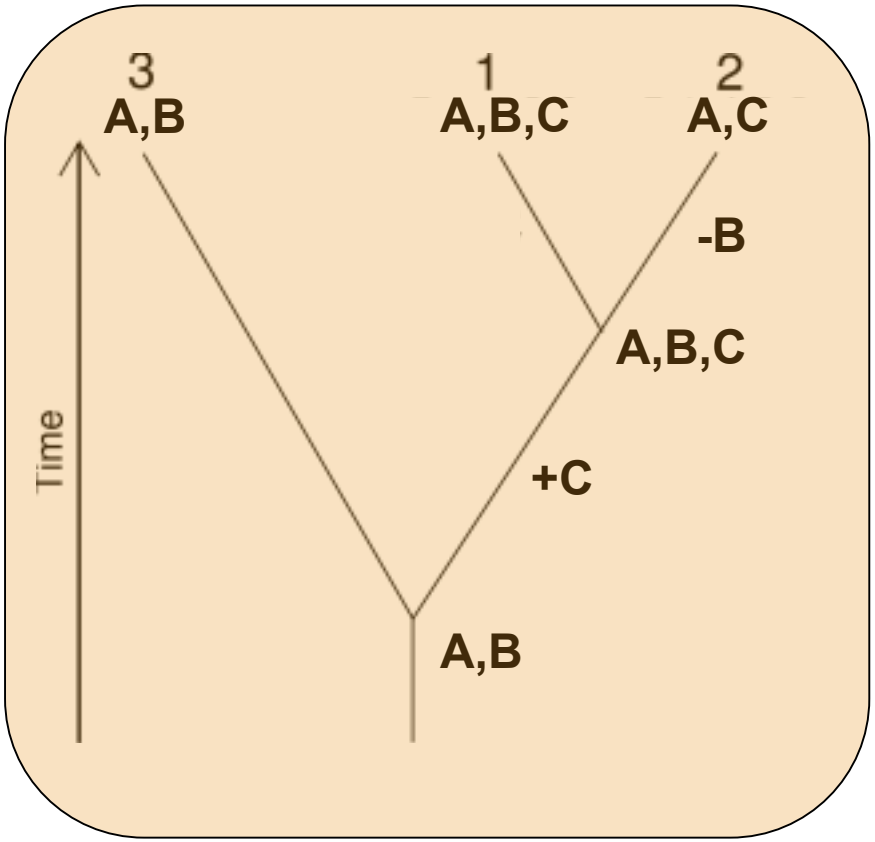
Also called law of parsimony. (Ockham's razor, ca 1285-1350)

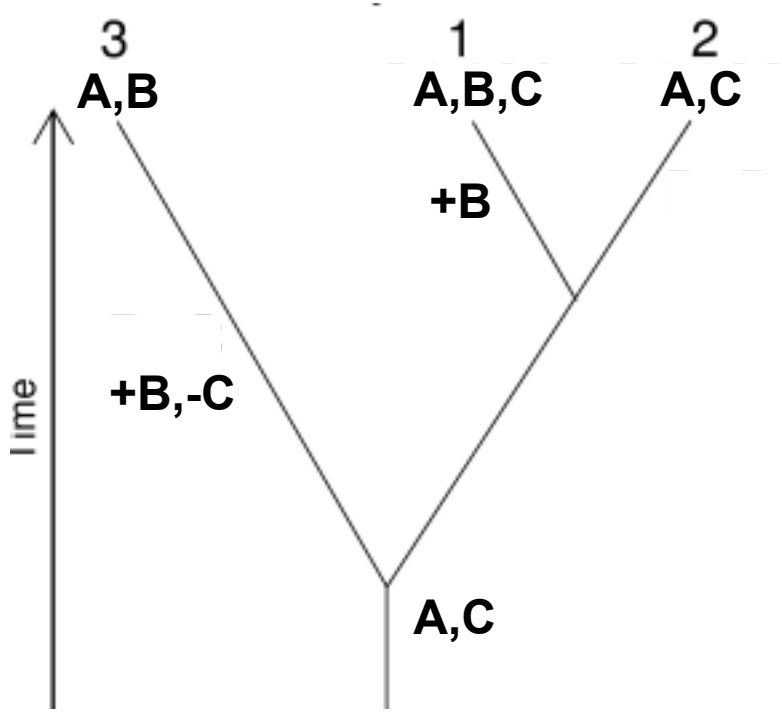
The Parsimony Principle





The Parsimony Principle



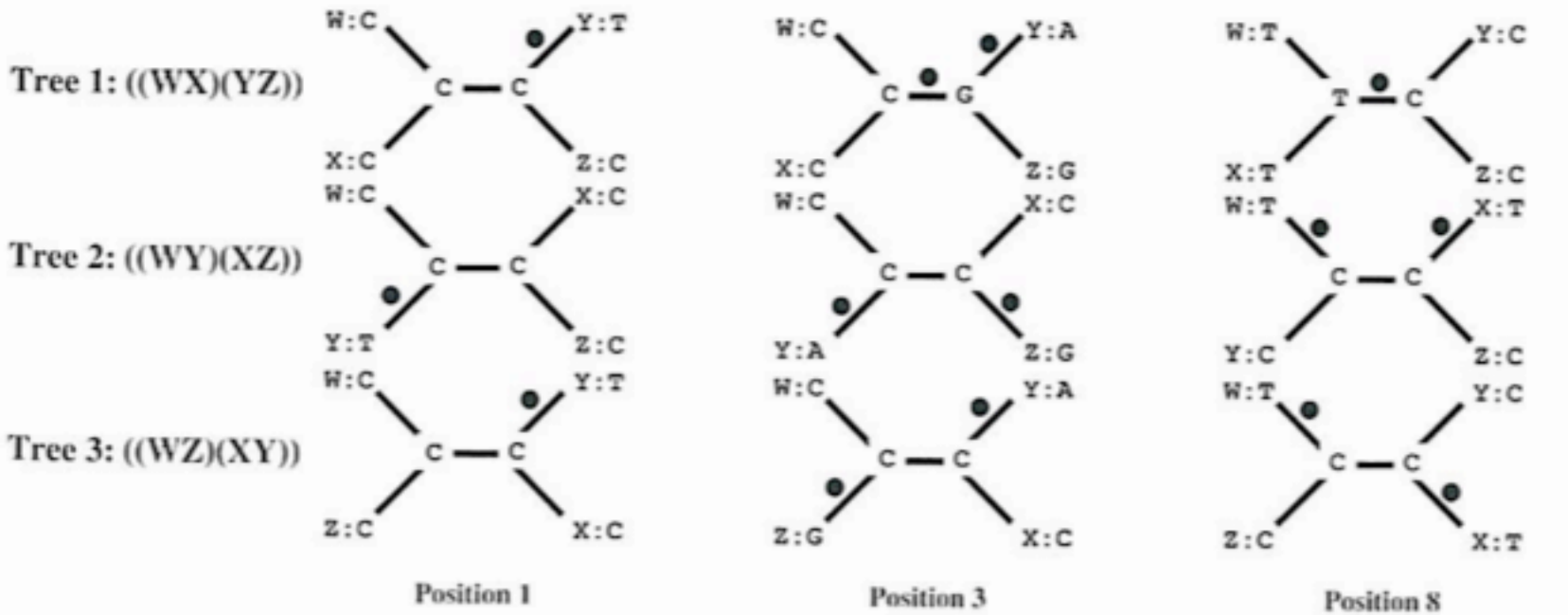


How to infer a tree from the data

Data	Method	Evaluation Criterion
Characters (Alignment)	**Maximum Parsimony**	Parsimony
	Statistical Approaches: Likelihood, Bayesian	Evolutionary Models
Distances	**Distance Methods**	

The Criterion of Maximum Parsimony

	Position								
	1	2	3	4	5	6	7	8	9
Sequence W:	C	G	C	A	C	T	G	T	T
Sequence X:	C	G	C	A	C	T	G	T	T
Sequence Y:	T	G	A	A	C	T	G	C	T
Sequence Z:	C	G	G	A	C	T	G	C	T
	*		*					*	



The Criterion of Maximum Parsimony

Find the tree τ that minimizes the following expression:

$$L(\tau) = \sum_{k=1}^{B} \sum_{j=1}^{L} \omega_j \cdot \mathrm{diff}(x_{k'j}, x_{k''j})$$

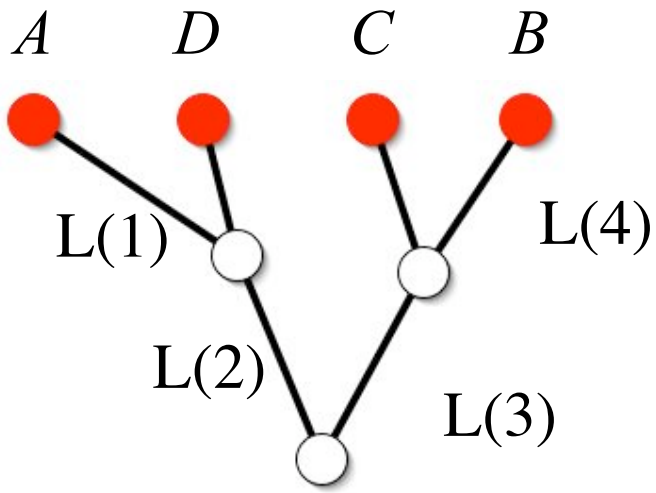
where `diff` measures the distance between two characters
ω_j is an alignment specific weight factor
L alignment length
B number of branches in the tree
k‘ and *k*‘‘ are the two nodes connected by branch *k*

The Criterion of *Distance (Hamming Distance)*

seq 1	a	g	c	t	t	a	c	c	t	g	t	t	a	c	t
seq 2	c	g	t	a	a	a	t	t	t	c	c	c	g	a	t
seq 3	c	g	c	a	a	g	t	t	t	c	c	c	g	a	t
seq 4	c	a	c	t	t	a	t	t	a	g	t	c	a	a	c

	Seq 1	Seq 2	Seq 3	Seq 4
Seq 1	0	11	11	8
Seq 2	11	0	2	10
Seq 3	11	2	0	9
Seq 4	8	10	9	0

Distance-based tree reconstruction



Find branch lengths L(b) such that the sum of the branch lengths connecting any two leaves gets close to the measured distances between all pairs of leaves. That is

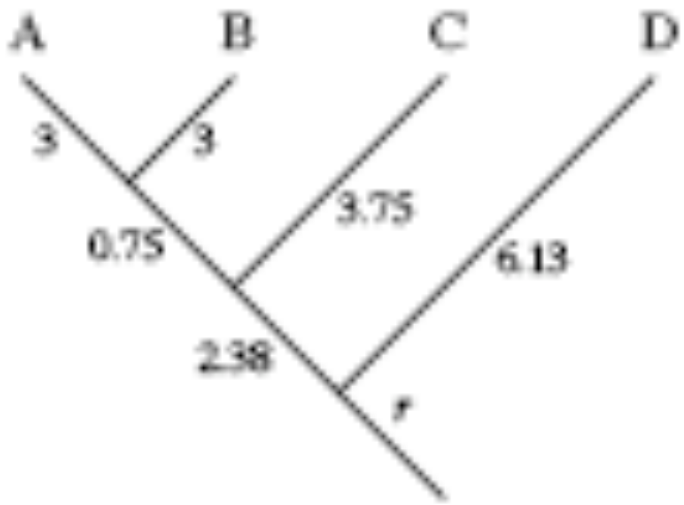
$$D_{measured}(A,B) = L(1)+L(2)+L(3)+L(4)$$

Clustering Methods

UPGMA = Unweighted Pair Group Methods using Arithmetic means.

	A	B	C
B	6		
C	7	8	
D	13	14	11

a)



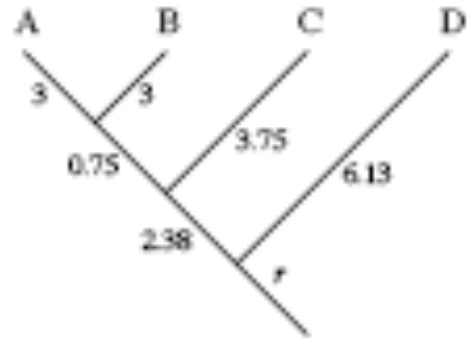
b)

The ultrametric condition implies the molecular clock

Clustering methods work well, if sequences evolve according to a molecular clock

	A	B	C
B	6		
C	7	8	
D	13	14	11

a)



b)

or equivalently: if the ultrametric inequality is holds:

$$d(A,B) \leq \max\{d(A,C), d(B,C)\}$$

for each triple (A,B,C)

Theorem: Four-Point-Condition

A distance matrix $(d_{i,j})_{i,j=1....n}$ is representable as a tree, if and only if

$$d(u,v) + d(x,z) \leq \max\{d(u,x) + d(v,z), d(u,z) + d(v,x)\}$$

for all $u, v, x, z \in \{1,2,...,n\}$

Getting rid of the molecular clock requirement

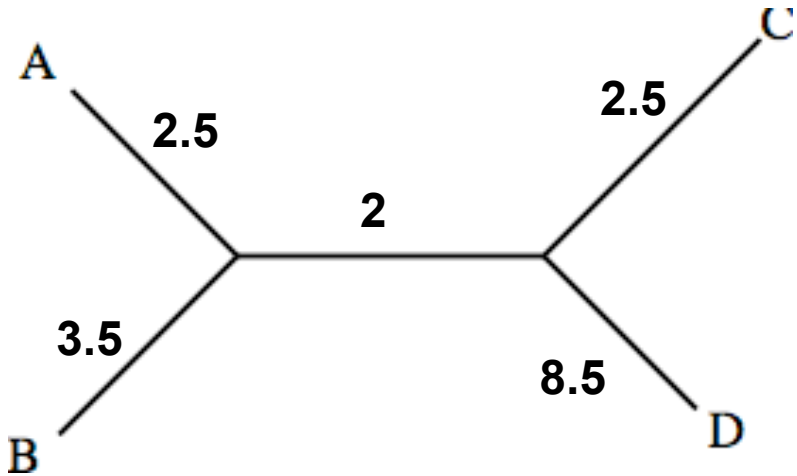
Theorem: Four-Point-Condition

A distance matrix $(d_{i,j})_{i,j=1....n}$ is representable as a tree, if and only if

$$d(u,v) + d(x,z) \leq \max\{d(u,x) + d(v,z), d(u,z) + d(v,x)\}$$

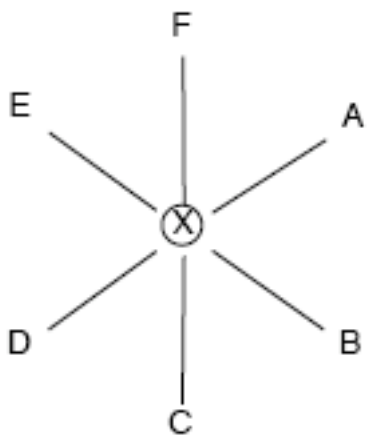
for all $u, v, x, z \in \{1,2,\ldots,n\}$

	A	B	C
B	6		
C	7	8	
D	12	14	11



The Neighbour Joining Algorithm

1. begin with **star tree**:



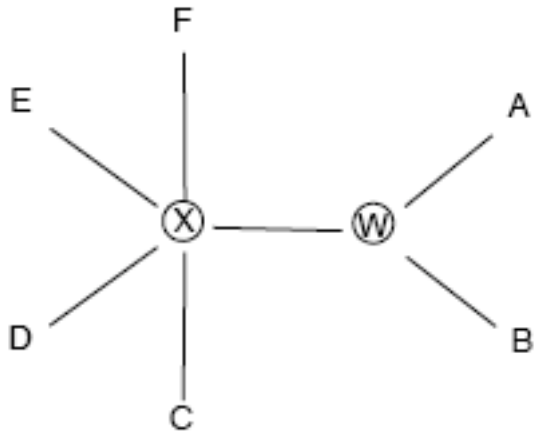
2. compute for each pair (*1,2*) the **net-divergence**

$$\frac{1}{2(N-2)}\sum_{k=3}^{N}(D_{1k}+D_{2k})+\frac{1}{2}D_{12}+\frac{1}{N-2}\sum_{3\le i<j}D_{ij}. \qquad (1)$$

3. take the pair (*A,B*) that minimizes Eq. (1)

The Neighbour Joining Algorithm

4. cluster (*A,B*) and define an interior node *W*



5. compute branch lengths for the external edges:

$$L(A,W) = \frac{1}{2}\left(D(A,B) + \frac{1}{m-2}\sum_{k=1}^{m} D(A,k) - D(B,k) \right)$$

$$L(B,W) = \frac{D(A,B)}{2} - L(A,W)$$

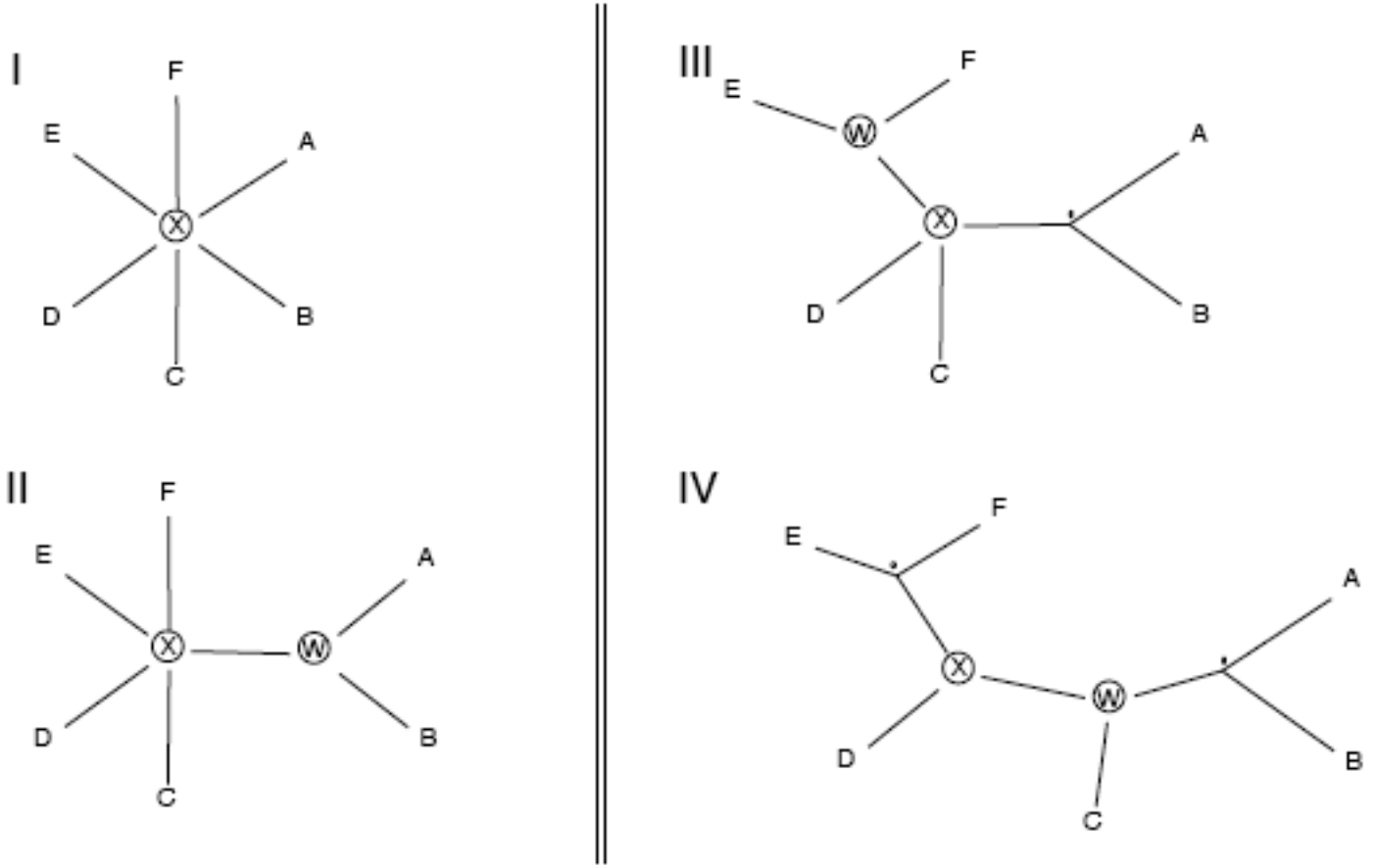
The Neighbour Joining Algorithm

6. compute distance *W* to the remaining *m-2* leaves:

$$D(W,k) = \frac{1}{2}\left(D(A,k) + D(B,k) - D(A,B)\right)$$

7. continue with step 1 with the reduced set of leaves

The Neighbour Joining Algorithm



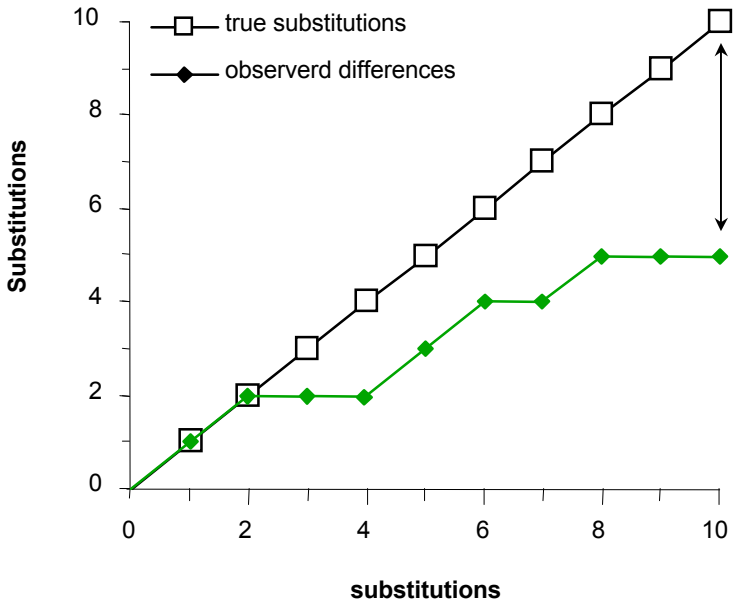
Least-squares Methods

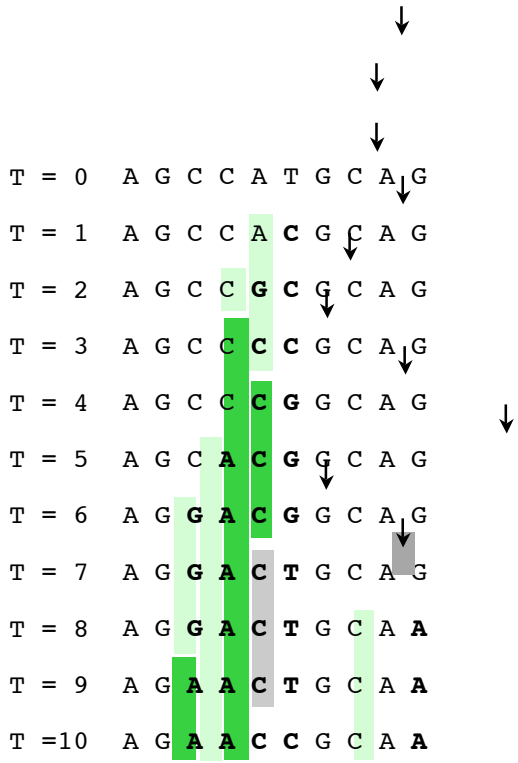
Find a tree τ that minimizes

$$S(\tau) = \sum_{i,k} \left(\rho(i,k) - D(i,k) \right)^2$$

where $\rho(i,k)$ is the length of the unique path connecting leaves *i* and *k* in the tree.

Distance Correction





Jukes Cantor Correction Forumla

$$obs(d) = \frac{3}{4} - \frac{3}{4} Exp[-4d/3]$$

obs(d) can be estimated from the number of observed different pairs of positions n_1 between two aligned sequences of length *l.*

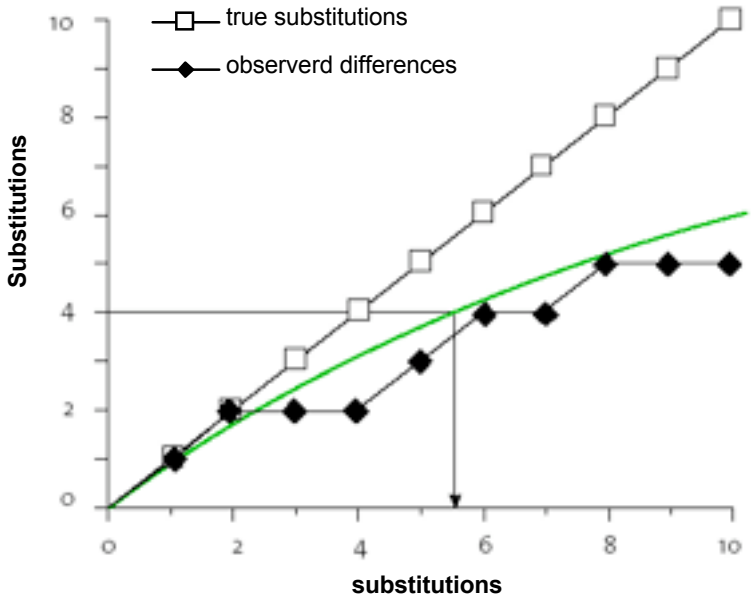
Solving

$$\frac{n_1}{l} = \frac{3}{4} - \frac{3}{4} Exp[-4d/3]$$

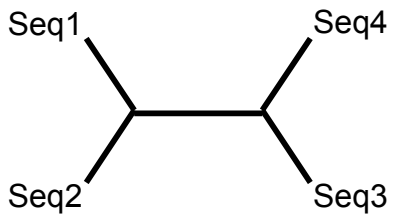
leads to **Jukes Cantor correction**:

$$d = -\frac{3}{4} Log\left[1 - \frac{4}{3}\frac{n_1}{l}\right]$$

Distance Correction



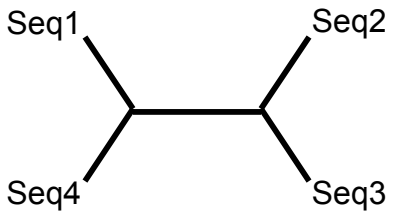
The Problem: Different alignments, different trees



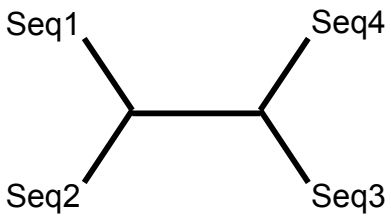

```
Seq1: - N Y L S
Seq2: N K Y L S
Seq3: - N F - S
Seq4: - N F L S
```

N Y L S N K Y L S N F S N F L S

```
Seq1: N - Y L S
Seq2: N K Y L S
Seq3: N - F - S
Seq4: N - F L S
```



The Problem: Different alignments, different trees




```
Seq1: - N Y L S
Seq2: N K Y L S
Seq3: - N F - S
Seq4: - N F L S
```

N Y L S N K Y L S N F S N F L S

The alignment strategy may have more impact on the reconstructed tree than does the type of tree building method.
Morrison and Ellis (1997) Mol. Biol. Evol. 14:428-441

Focussing on stable parts of the alignment

Gblocks (Castresana (2000) Mol. Biol. Evol. 17:540-552
Objective:
Define a set of conserved blocks from an alignment to be used in phylogeny reconstuction

Approach:

1) Classification of Columns
   - non-conserved : <n/2 + 1 identical residues, or a gap
   - conserved : ≥n/2 + 1 and < 85% identical residues
   - highly conserved :>85% identical residues

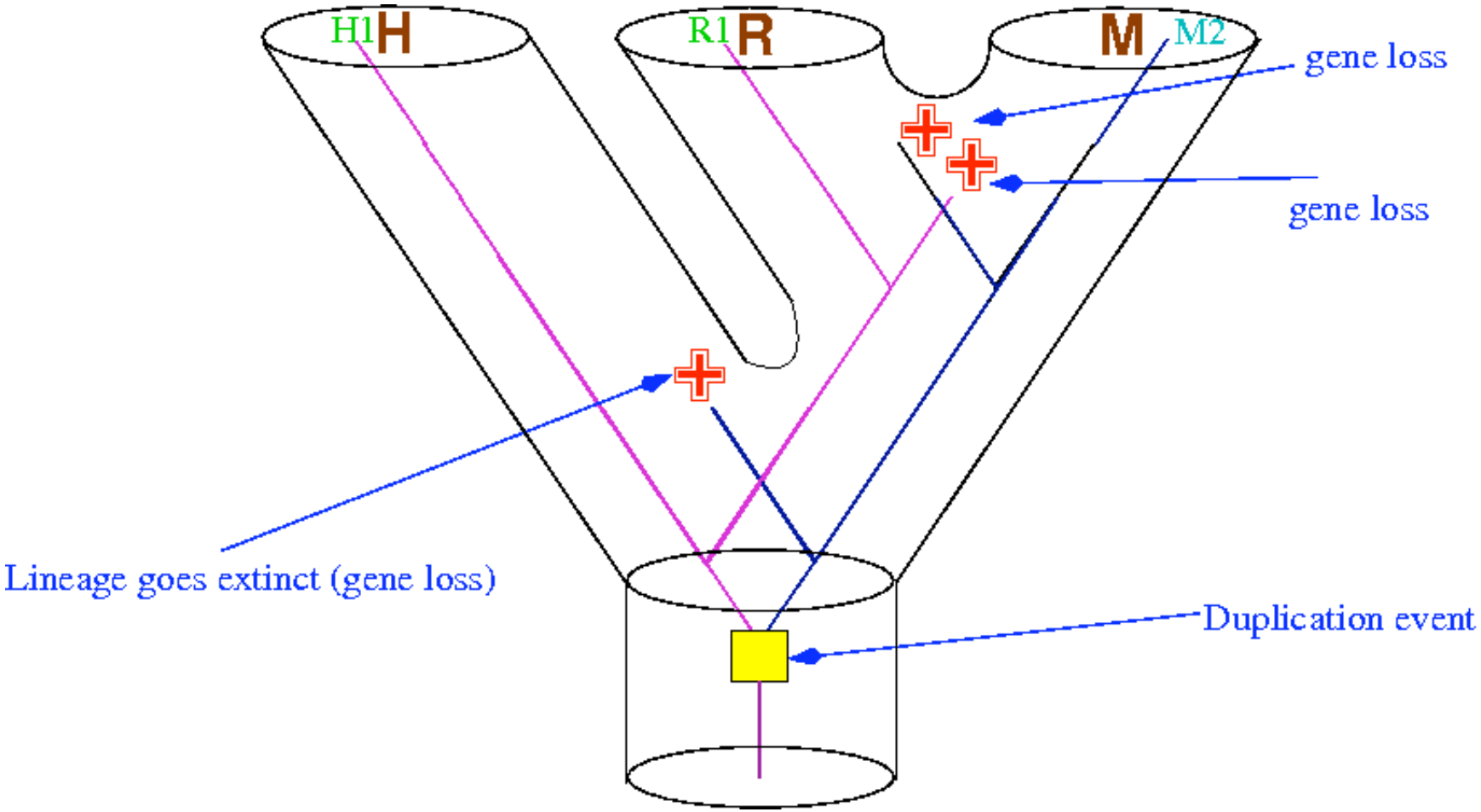
2) discard contiguous stretches of non-conserved positions (default l = 8)
3) from remaining blocks: remove flanking positions until blocks begin and end with highly conserved positions, i.e. selected blocks are anchored by positions that can be aligned with high confidence
4) discard blocks with l < 15
5) remove all positions with gaps together with adjacent positions until a conserved position is reached
6) discard blocks with l < 10

Note: all given values are the program defaults as given in the original publication

Focussing on stable parts of the alignment

```
                         70        80        90       100       110       120
                ---------+---------+---------+---------+---------+---------+
nad3_parde      D-DARMKFDVRFYLVSILFIIFDLEVAFLFPWAVSFASLS-DVAFWGLMVFLAVLTVGFA
nad3_acaca      E-DTRSKFNVRYYLIAILFMIFDLEIMYLFPWSISISTGS-FFGVWAIFLFLIILTVGFI
nad3_allma      G-DARQKFDVSFYLIAILFIIFDLEVVFILPFASVIHNVS-LLGGWITIIFLVILTIGFI
nad3_apec       N-SARVPFSFRFFLVAILFLLFDLEIALLFPLPFSVFFH--P--IHTP----LILTVGLI
nad3_arath      G-DARSRFDIRFYLVSILFLIPDLEVTFFFPWAVPPNKID-LFGFWSMMAFLFILTIGFL
nad3_balca      A-SARLPFSLRFFLVAILFLLFDLEIALLLPFPAALSARDPQLSFTLAFLILLILTIGLI
nad3_chocr      D-DARATFDIRFYLVAILFLIFDLEISFLFPWSLVLGEIS-IIGFWSMIVFLVILTIGFI
nad3_drome      S-SSRLPFSLRFFLITIIFLIFDVEIALILPMIIIMKYSNIMIWTITSIIFILILLIGLY
nad3_human      S-PARVPFSMKFFLVAITFLLFDLEIALLLPLPWALQTTNLPLMVMSSLLLIIILALSLA
nad3_ktun       S-SARLPFSMRFFLITVVFLVFDVEIVLLLPYLFSSGWSIDVFSLVGSMMILVILIIGVL
nad3_lter       S-TARIPFSTRFFLLAIIFIVFDIEIVLLMPLPTILHTSDVFTTVTTSVLFLMILLIGLI
nad3_marpo      D-DARSRFDIRFYLVSILFIIFDLEVTFLFPWAVSLNKIG-LFGFWSMMVFLFILTIGFV
nad3_metse      G-TPGRPFSIRFFLIGILFLIFDLEISFLFPWCVVCNQVF-PFGYWTMIVFLAVLTLGLV
nad3_picca      R-QSRTTYSIKFILIAILFLPFDLELTSILPYTLSMYNTN-IYGLFILLYFLLPLIIGFI
nad3_podan      LGQNRTQFGVKFFIFALVYLLLDLEILLTFPFAVSEYVNN-IYGLIILLGFITIITIGFV
nad3_prowi      D-DARGRFDIQFYLVAILFIIFDLEVAFLFPWALTLNKIG-YFGFWSMMLFLFILTVGFI
nad3_recam      D-DARGAFDVRFYLVAILFIIFDLEVAFLFPWAVALSDVT-IFGFWTMFIFLLILTVGFI

                        130       140       150
                ---------+---------+---------+---
nad3_parde      YEWKKGALEWA----------------------
nad3_acaca      YEWQKGALEWD----------------------
nad3_allma      YEFVSGAITDSF---------------------
nad3_apec       FEWVQGGLDWAE---------------------
nad3_arath      YEWKRGASDRE----------------------
nad3_balca      YEWMEGGLEWAE---------------------
nad3_chocr      YEWYKGALEWE----------------------
nad3_drome      HEWNQGMLNWSN---------------------
nad3_human      YEWLQKGLDWTE---------------------
nad3_ktun       HEWSEGSLEWFSSSN------------------
nad3_lter       HEWKEGSLDWSS---------------------
nad3_marpo      YEWKKGALDWE----------------------
nad3_metse      YEWLKGGLEWE----------------------
nad3_picca      IEINTKAIYMTKMFNRNVKSMTSYVKYNNKI--
nad3_podan      YELGKSALKIDSRQVITMTRFNYSSTIEYLGKI
nad3_prowi      YEWRKGALDWS----------------------
nad3_recam      YEWKKGALDWE----------------------
```

Hidden paralogy mimics orthology



ma :0)

A typical variant: Weighted Sum of Pairs

```
Seq1: AGA--CTA
Seq2: AGA--CTA
Seq3: G-A--CTT
Seq4: AGAAACTT
```

```
Seq1: AGA--CTA
Seq2: AGA--CTA
```

Score: +30

```
Seq1: AGA--CTA
Seq3: G-A--CTT

Seq2: AGA--CTA
Seq3: G-A--CTT
```

Score: 2*(+5)

```
Seq1: AGA--CTA
Seq4: AGAAACTT

Seq2: AGA--CTA
Seq4: AGAAACTT
```

Score: 2*(+11)

```
Seq3: G-A--CTT
Seq4: AGAAACTT
```

Score: 0

SUM OF PAIRS SCORE: 62

A typical variant: Weighted Sum of Pairs

```
Seq1: AGA--CTA
Seq2: AGA--CTA
Seq3: G-A--CTT
Seq4: AGAAACTT
```

```
Seq1: AGA--CTA
Seq2: AGA--CTA
```

Score: +30

```
Seq1: AGA--CTA
Seq3: G-A--CTT

Seq2: AGA--CTA
Seq3: G-A--CTT
```

Score: 2*(+5)

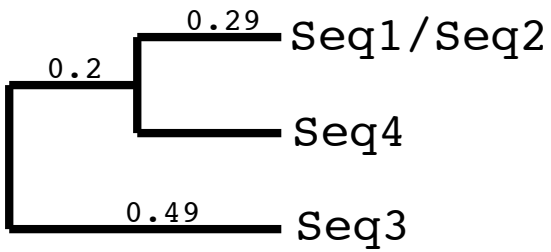
```
Seq1: AGA--CTA
Seq4: AGAAACTT

Seq2: AGA--CTA
Seq4: AGAAACTT
```

Score: 2*(+11)

```
Seq3: G-A--CTT
Seq4: AGAAACTT
```

Score: 0

SUM OF PAIRS SCORE: 62



Weighting of sequences: one variant

Dataset:

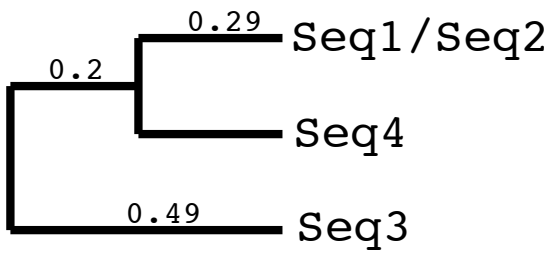
```
Seq1: AGACTA
Seq2: AGACTA
Seq3: GACTT
Seq4: AGAAACTT
```

Compute

Pairwise Distance Matrix

	1	2	3	4
1	-			
2		-		
3			-	
4				-

Reconstruct



Apply weights

```
Seq1: (0.29/2+0.2/3)=0.21
Seq2: (0.29/2+0.2/3)=0.21
Seq3: 0.49
Seq4: (0.29+0.2/3)=0.36
```

Normalize

```
Seq1: 0.43
Seq2: 0.43
Seq3: 1
Seq4: 0.73
```

A typical variant: Weighted Sum of Pairs

$$\sigma_{wsop}(\alpha) = \sum_{i<j} \omega_i \omega_j S(\alpha_i, \alpha_j)$$

```
Seq1: AGA--CTA
Seq2: AGA--CTA
Seq3: G-A--CTT
Seq4: AGAAACTT
```

```
Seq1: AGA--CTA
Seq2: AGA--CTA
```

Score: 0.43^2x30

```
Seq1: AGA--CTA
Seq3: G-A--CTT

Seq2: AGA--CTA
Seq3: G-A--CTT
```

Score: (0.43x5)2

```
Seq1: AGA--CTA
Seq4: AGAAACTT

Seq2: AGA--CTA
Seq4: AGAAACTT
```

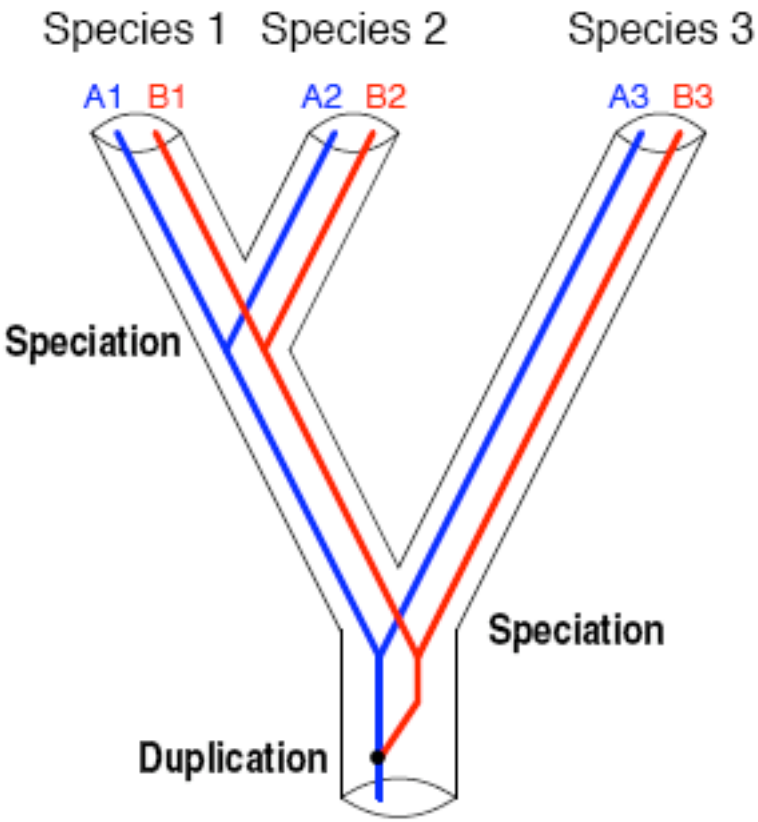
Score: (0.43x0.73x11)2

```
Seq3: G-A--CTT
Seq4: AGAAACTT
```

Score: 0

SUM OF PAIRS SCORE: 16.7

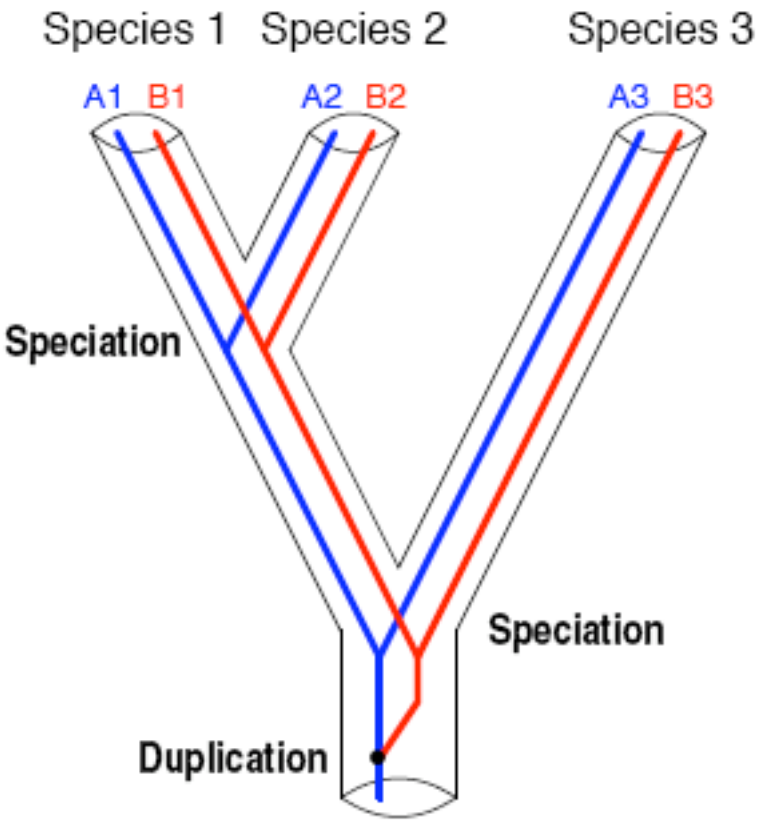
Orthologous Sequences, Please!!



Arguments for orthology assumption:

- a sequence tree that is congruent to the species tree
- conservation of genomic position
- sequence similarity (typically, reciprocal best blast hit)
- similarity of function

Orthologous Sequences, Please!!



Arguments for orthology assumption:

- a sequence tree that is congruent to the species tree
- conservation of genomic position
- sequence similarity (typically, reciprocal best blast hit)
- similarity of function