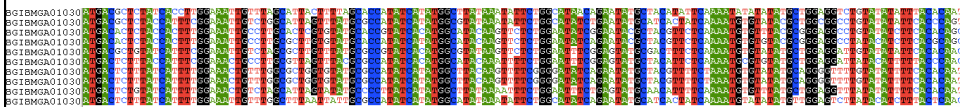
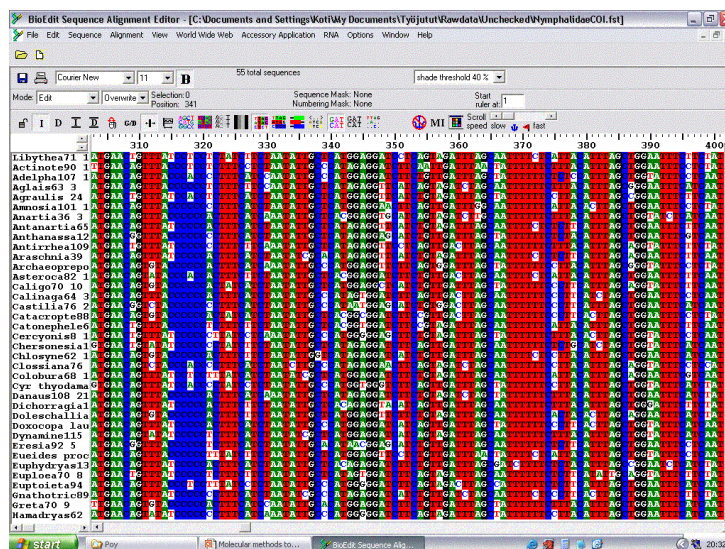


Multiple Sequence Alignment



Alignment can be easy...

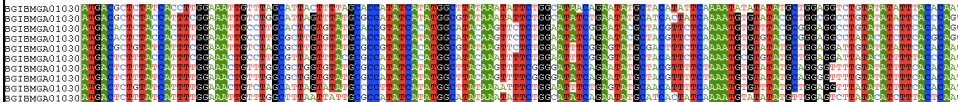


...or difficult



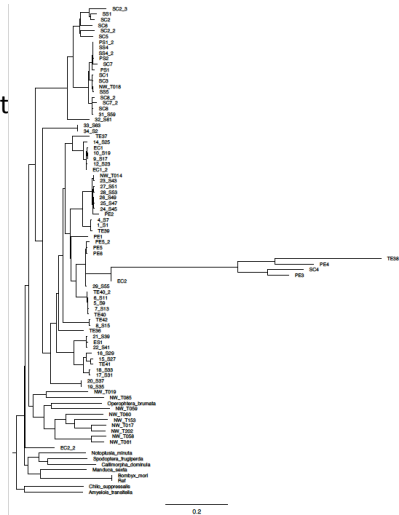
Homology: Definition

- Homology: similarity that is the result of inheritance from a common ancestor - identification and analysis of homologies is central to phylogenetic systematics
- An alignment is a hypothesis of positional homology between bases/amino acids



Multiple sequence alignment- goals

- To generate a concise, information-rich summary of sequence data
- Alignments can be treated as models that can be used to test hypotheses
- Does this model of events accurately reflect known biological evidence?



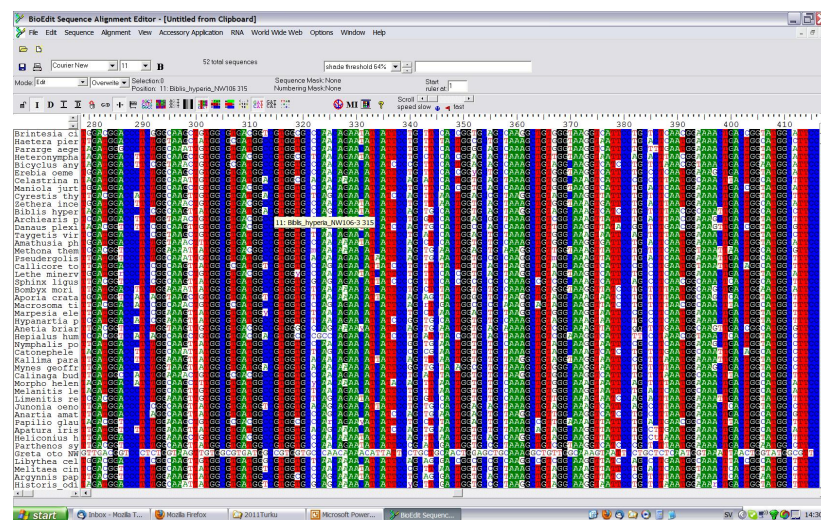
Multiple sequence alignment

- Manual
- Dynamic programming
- Heuristic methods
 - Progressive alignment
 - Consistency-based scoring
 - Iterative refinement methods

Manual alignment - reasons

- Might be carried out because:
- Alignment is easy
- There is some extraneous information (structural)
- Automated alignment methods have encountered a local minimum problem
- An automated alignment method can be “improved”

Protein-coding genes can often be manually aligned



How to align these sequences:

AGGGCTTTAA
AGGCTA
AATGGCTCTAA
GGAGCCCTAA

How to align these sequences:

A-AGGGCTTTAA
A--GGCT--A-
AATGGCTCTAA
GGAG-CCCTAA

How to align these sequences:

-AGGGCTTTAA
-A-GGC--TA-
AATGGCTCTAA
-GGAGCCCTAA

Multiple sequence alignment

- Is not easy! How to be objective?
- Dynamic programming
- Heuristic methods
 - Progressive alignment
 - Consistency-based scoring
 - Iterative refinement methods

Dynamic programming

- For two sequences, the best alignment can be found by scoring all possible pairs of aligned nucleotides and penalizing gaps
- An optimality criterion
- Time and computer memory needed grows exponentially with number of sequences
- Becomes impossible to align more than 4 sequences of modest length
- Fails to fully exploit phylogeny and does not incorporate an evolutionary model

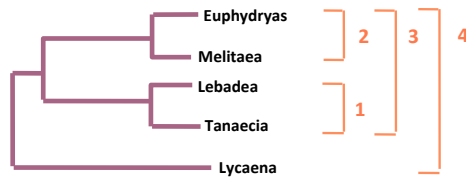
Heuristics: Progressive alignment

- Devised by Feng and Doolittle in 1987
- A heuristic method and as such is not guaranteed to find the 'optimal' alignment
- Requires $n-1+n-2+n-3...n-n+1$ pairwise alignments as a starting point
- Most successful implementation is Clustal
 - ClustalW
 - ClustalX

Overview of Clustal procedure

Euphydryas 1 -
 Melitaea 2 .17 -
 Lebadea 3 .59 .60 -
 Tanaecia 4 .59 .59 .13 -
 Lycaena 5 .77 .77 .75 .75 -

Quick pairwise alignment:
 calculate distance matrix



Neighbour-joining tree
 (guide tree)

Lycaena hell G G G G G A G G G A A G A G G G A G G G G G
 Euphydryas m G G G G G G A G G A A G A G A G A G A G A G A G A G A G
 Melitaea amb G G G G G G A G G A A G A G A G A G A G A G A G A G A G
 Lebadea mart G G G G G G A G G A A G A G A G A G A G A G A G A G A G
 Tanaecia jul G G G G G G A G G A A G A G A G A G A G A G A G A G A G

Progressive alignment
 following guide tree

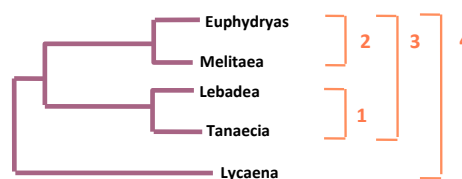
Clustal - pairwise alignments

- First perform all possible pairwise alignments between each pair of sequences
- Calculate the 'distance' between each pair of sequences based on these isolated pairwise alignments
- Generate a distance matrix

Euphydryas 1 -
 Melitaea 2 .17 -
 Lebadea 3 .59 .60 -
 Tanaecia 4 .59 .59 .13 -
 Lycaena 5 .77 .77 .75 .75 -

Clustal - guide tree

- Generate a Neighbour-Joining 'guide tree' from these pairwise distances
- This guide tree gives the order in which the progressive alignment will be carried out



Multiple alignment- first pair

- Align the two most closely-related sequences first
- This alignment is then 'fixed' and will never change
- If a gap is to be introduced subsequently, then it will be introduced in the same place in both sequences, but their relative alignment remains unchanged

```

3GAAAAGTCACCAAACCTGTGGCAGCTTGTGCCATGCCTGT
3TAGAGAAGTCCCCCAAGCCTGCAGCAGCTTGCGCCATGCC
  
```

↓

```

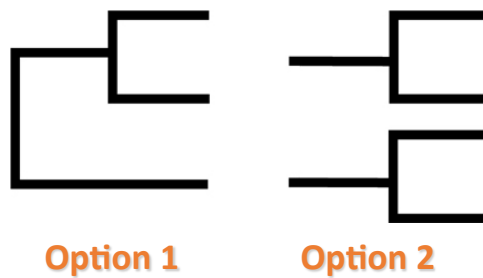
GGAAGAAGTCACCAAACCTGTGGCAGCTTGTGCCATGCCTGT
AGAGAAGTCCCCCAAGCCTGCAGCAGCTTGCGCCATGCC
  
```

```

[
GAAAAGTCACCAAACCTGTGGCAGCTTGTGCCATGCCTGT
AGAGAAGTCCCCCAAGCCTGCAGCAGCTTGCGCCATGCC
TGAAGAAGTCACCAAACCTGTGGCAGCTTGTGCCATGCC
]
  
```

Clustal - decision time

- Consult the guide tree to see what alignment is performed next.
 - Align a third sequence to the first two
 - Or
 - Align two entirely different sequences to each other.



Clustal - progression

- The alignment is progressively built up in this way, with each step being treated as a pairwise alignment, sometimes with each member of a 'pair' having more than one sequence

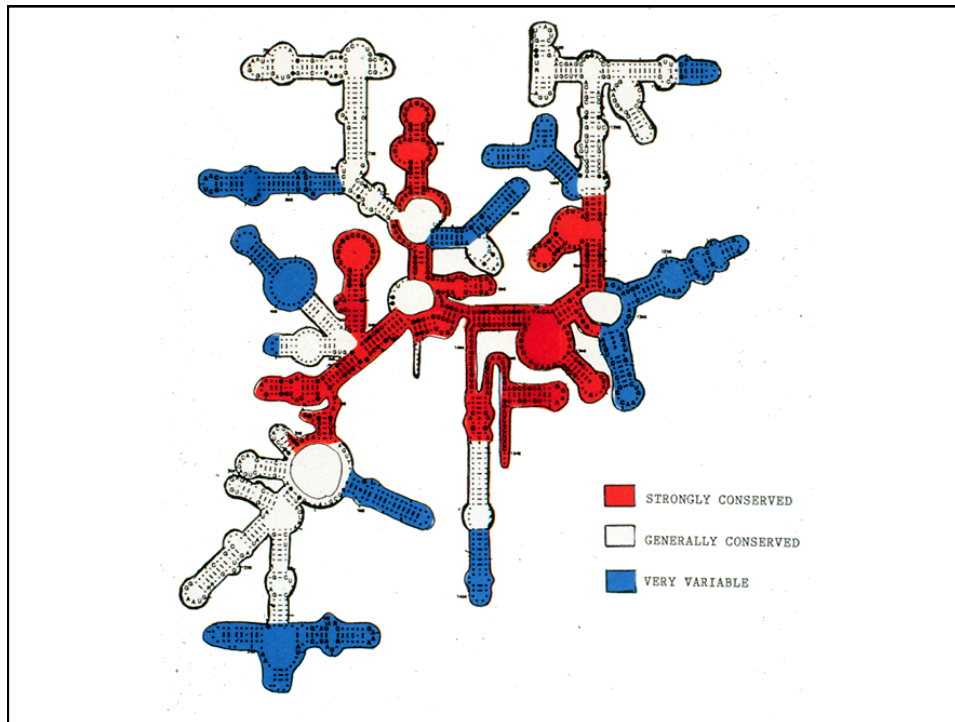
Lycaena hell	G	C	C	G	T	G	-	-	C	A	G	A	G	C	A	A	G	A	T	G	G	G	A	C	A	G	A	G	T	G	G	C				
Euphydryas m	G	C	C	C	T	G	C	G	A	C	A	G	C	A	A	A	A	G	A	T	G	A	G	A	C	C	G	A	C	A	G	A	T	T	C	C
Melitaea amb	G	C	C	G	T	T	A	G	A	C	A	A	C	A	A	A	A	G	A	T	G	A	G	A	C	C	G	A	C	A	G	A	T	T	C	C
Lebadea mart	T	C	C	G	G	T	T	-	-	C	A	A	C	A	A	A	A	T	G	A	T	G	A	C	T	A	G	A	T	T	C	C	C	C		
Tanaecia jul	T	C	C	A	G	T	T	-	-	C	A	G	C	A	A	A	T	G	A	T	G	A	C	T	A	G	A	T	T	C	C	C	C			

Clustal - good points/bad points

- Advantages:
 - Speed
- Disadvantages:
 - Hierarchic structure introduced that is not necessarily phylogenetic
 - No way of quantifying whether or not the alignment is good
 - No way of knowing if the alignment is 'correct'
 - Local minimum problem. If an error is introduced early in the alignment process, it is impossible to correct this later in the procedure
 - Arbitrary alignment

Increasing the sophistication of the alignment process

- Should we treat all the sequences in the same way?
 - some sequences are closely related and some sequences are distant relatives.
- Should we treat all positions in the sequences as though they were the same?
 - they might have different functions and different locations in the 3-dimensional structure.
 - codon structure – how to retain this?



Consistency-based scoring

- One way to avoid the problems of getting stuck in local minima or fixed gaps
- Based on optimizing a multiple alignment using information from all pairwise alignments
- Identifies those nucleotides that are aligned most consistently across the different alignments
- Used in e.g. T-Coffee

Iterative refinement methods

- Initial alignments split into two groups randomly
- Within groups the alignment is kept fixed
- Dynamic programming used to align the two groups to each other
- This is repeated until score converges
- Used in e.g. Muscle and MAFFT

Using models in alignment

- New methods are being developed all the time
- Latest methods include using a Bayesian statistic framework, DNA evolutionary models and alignment concomitantly with estimation of phylogentic relationships
- Still not feasible with a moderately sized dataset

Phylogenomics: Crucial step 1 – orthology!

- We need to know that the genes we are studying are the same (homology)
- Old style PCR primers amplifying orthologous genes
- Genomics relies on bioinformatic methods to determine orthology

Orthology and BUSCO

- For phylogenomics single copy, protein coding, orthologous genes are often preferred
- Benchmarking Universal Single-Copy Orthologs (BUSCO)
 - Taxon specific sets

- Old style manual alignment

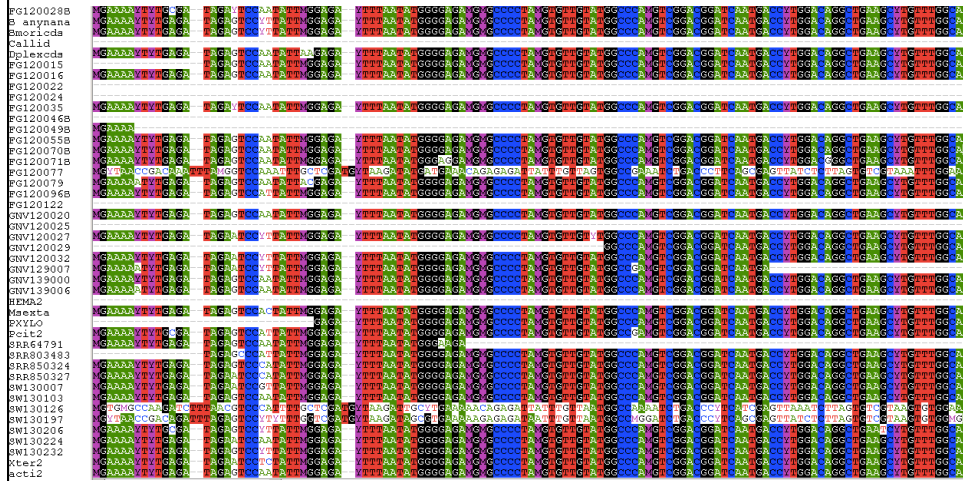
[illegible]

- Old style manual alignment

[illegible]

- Phylogenomic datasets rely on bioinformatic tools to align

The dirty secrets of bioinformatics, part 2



Kawahara & Breinholt 2014

Bottom line

- Alignments are extremely important in phylogenetics
- A bad alignment means many wrong statements of homology, which means pure rubbish as output
- A good alignment can be hard to attain

The Tree

Finding the optimal trees

Numbers of possible trees for N taxa

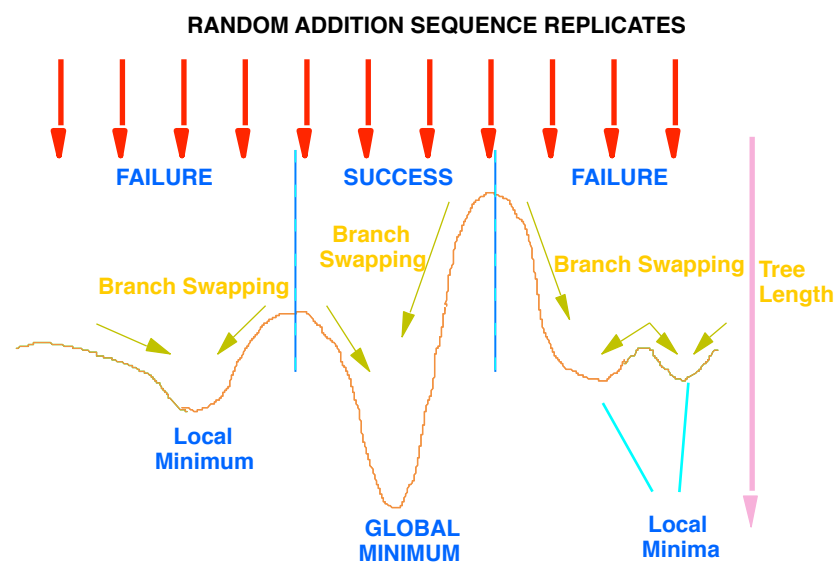
1	1
2	1
3	1
4	3
5	15
6	105
7	945
8	10395
9	135135
10	2027025
11	34459425
12	654729075
13	13749310575
14	316234143225
15	7905853580625
16	213458046676875
17	6190283353629370
18	191898783962510625
19	6332659870762850625
20	221643095476699771875 (2×10^{20})
50	3×10^{74}

How can
we find
the most
optimal
tree?

Criteria for the best tree?

- Optimum tree is the best with today's methods
- Optimality criteria
 - Parsimony: minimizing tree length
 - Modeling methods: maximizing likelihood

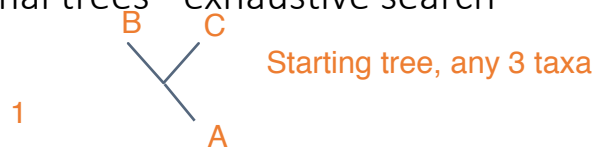
Tree space may be populated by local optima and islands of optimal trees



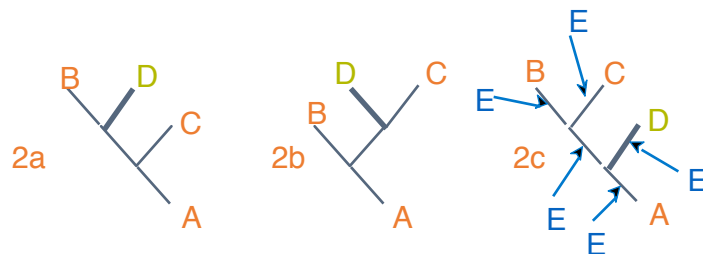
Finding optimal trees - exact solutions

- Exact solutions can only be used for small numbers of taxa
- **Exhaustive search** examines all possible trees
- **Branch and bound** does not examine all trees, but will find optimal tree(s)
- Typically used for problems with 10 -20 taxa

Finding optimal trees - exhaustive search



Add fourth taxon (D) in each of three possible positions -> three trees



Add fifth taxon (E) in each of the five possible positions on each of the three trees -> 15 trees, and so on

Finding optimal trees - exact solutions

- Branch and bound saves time by discarding families of trees during tree construction that cannot be shorter than the shortest tree found so far
- Can be enhanced by specifying an initial upper bound for tree length
- Typically used only for problems with fewer than 20 taxa

Finding optimal trees - heuristics

- The number of possible trees increases faster than exponentially with the number of taxa making exhaustive searches impractical for many data sets (an NP-complete problem)
- Heuristic methods are used to search tree space for optimal trees by building or selecting an initial tree and swapping branches to search for better ones
- The trees found are not guaranteed to be optimal - they are best guesses

Finding optimal trees - heuristics

- Stepwise addition

Asis - the order in the data matrix

Closest - starts with shortest 3-taxon tree, adds taxa in order that produces the least increase in tree length (greedy heuristic)

Simple - the first taxon in the matrix is taken as a reference
- taxa are added to it in the order of their decreasing similarity to the reference

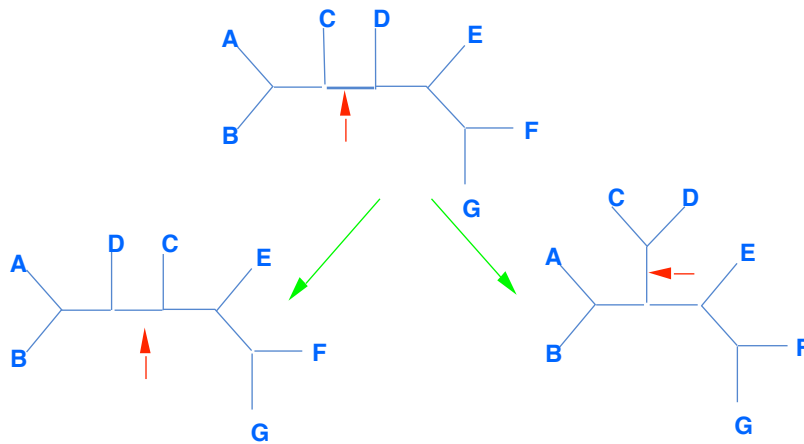
Random - taxa are added in a random sequence, many different sequences can be used

Finding optimal trees – branch swapping

- Nearest neighbor interchange (NNI)
- Subtree pruning and regrafting (SPR)
- Tree bisection and reconnection (TBR)

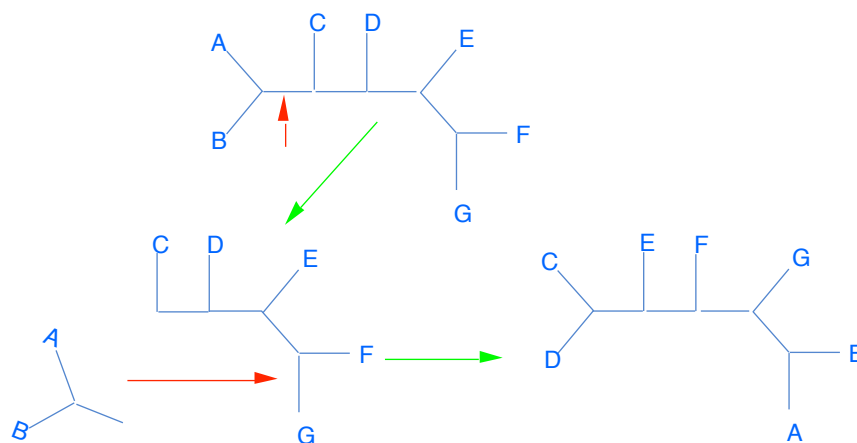
Finding optimal trees - heuristics

Nearest neighbor interchange (NNI)



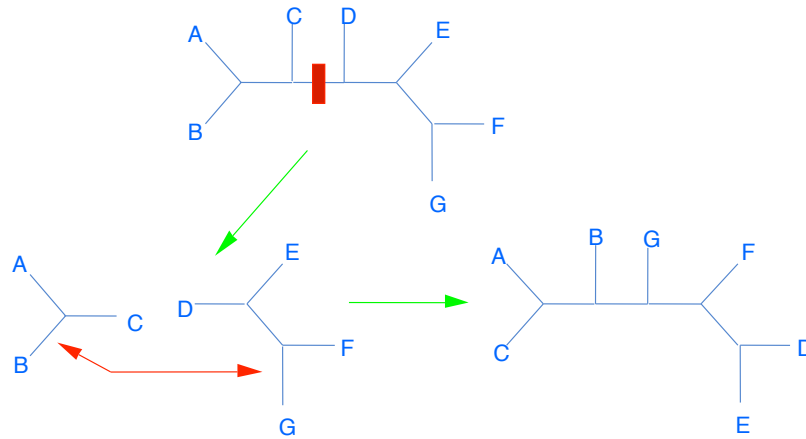
Finding optimal trees - heuristics

Subtree pruning and regrafting (SPR)



Finding optimal trees - heuristics

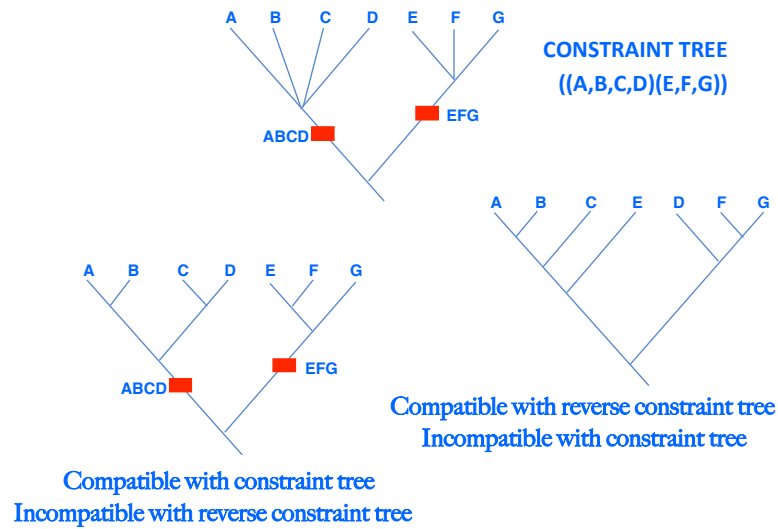
Tree bisection and reconnection (TBR)



Searching with topological constraints

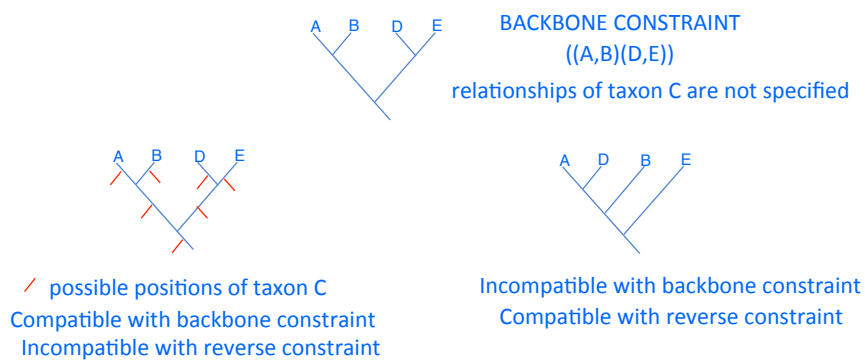
- Topological constraints are user-defined phylogenetic hypotheses
- Can be used to find optimal trees that either:
 1. include a specified clade or set of relationships
 2. exclude a specified clade or set of relationships (reverse constraint)

Searching with topological constraints



Searching with topological constraints backbone constraints

- Backbone constraints specify relationships among a subset of the taxa



Consensus methods

Multiple optimal trees

- Many methods can yield multiple equally optimal trees
- We can further select among these trees with additional criteria, but
- Typically, relationships common to all the optimal trees are summarised with *consensus trees*

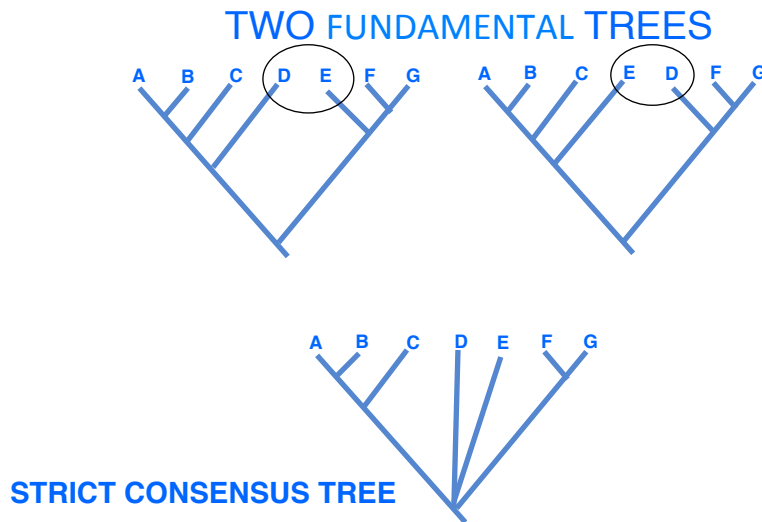
Consensus methods

- A consensus tree is a summary of the agreement among a set of fundamental trees
- There are many consensus methods that differ in:
 1. the kind of agreement
 2. the level of agreement
- Consensus methods can be used with multiple trees from a single analysis or from multiple analyses

Strict consensus methods

- Strict consensus methods require agreement across all the fundamental trees
- They show only those relationships that are unambiguously supported by the parsimonious interpretation of the data
- The commonest method (*strict component consensus*) focuses on clades/components/full splits
- This method produces a consensus tree that includes all and only those full splits found in all the fundamental trees
- Other relationships (those in which the fundamental trees disagree) are shown as unresolved polytomies

Strict consensus methods

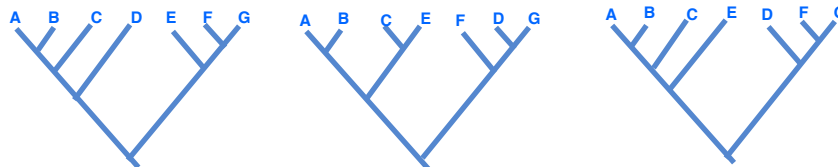


Majority-rule consensus methods

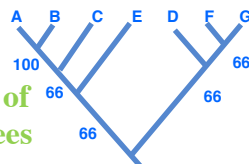
- Majority-rule consensus methods require agreement across a majority of the fundamental trees
- May include relationships that are not supported by the most parsimonious interpretation of the data
- The commonest method focuses on clades/components/full splits
- This method produces a consensus tree that includes all and only those full splits found in a majority (>50%) of the fundamental trees
- Other relationships are shown as unresolved polytomies
- Of particular use in bootstrapping

Majority rule consensus

THREE FUNDAMENTAL TREES



Numbers indicate frequency of clades in the fundamental trees



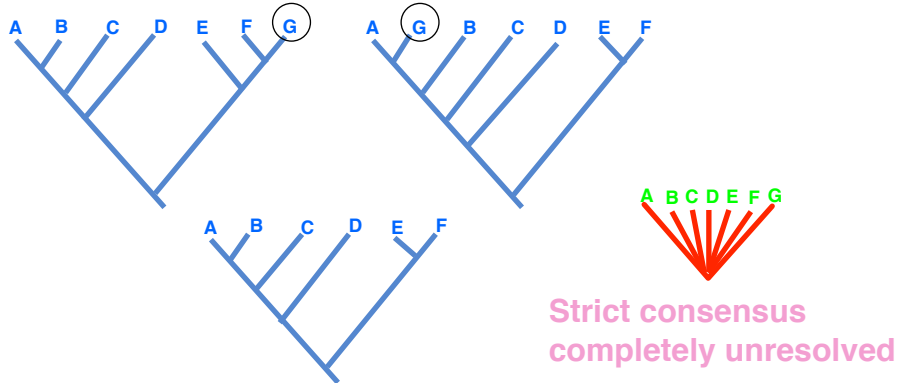
MAJORITY-RULE CONSENSUS TREE

Reduced consensus methods

- Focuses upon any relationships (not just full splits)
- Reduced consensus methods occur in strict and majority-rule varieties
- Other relationships are shown as unresolved polytomies
- May be more sensitive than methods focusing only on clades/components/full splits

Reduced consensus methods

TWO FUNDAMENTAL TREES

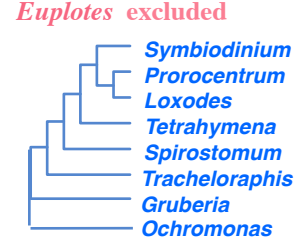
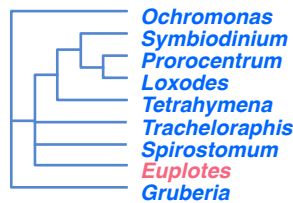
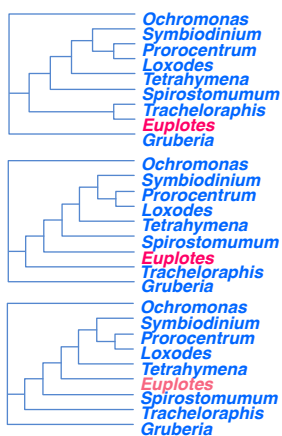


STRICT REDUCED CONSENSUS TREE

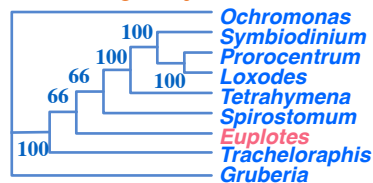
Taxon G is excluded

Consensus methods

Three fundamental trees Strict (component) Strict reduced cladistic



Majority-rule



Consensus methods – use

- Currently majority-rule methods mainly used
 - bootstrapping
 - Bayesian methods
- Reduced methods can be useful to identify problem taxa
 - E.g. RogueNaRok
- Strict methods mainly used in parsimony analyses
 - rarely used with molecular data

Take home messages from today

- Statements of homology are the basis of phylogenetics
- Alignments of molecular sequences are very strong statements of positional homology
- Finding an optimal tree is not a trivial task