

Maximum Likelihood Tree Searching

Emily Jane McTavish

Life and Environmental Sciences
University of California, Merced
`ejmctavish@ucmerced.edu`, `twitter:snacktavish`

(With thanks to Derrick Zwickl, Mark Holder, Dave Swofford and Paul Lewis for slides!)

Finding the tree with the best likelihood score is a hard problem

Finding the tree with the best likelihood score is a hard problem

- Enormous numbers of topologies to consider

Finding the tree with the best likelihood score is a hard problem

- Enormous numbers of topologies to consider
- May be multiple local optima

Finding the tree with the best likelihood score is a hard problem

- Enormous numbers of topologies to consider
- May be multiple local optima
- Need to maximize the likelihood for each topology

Enormous numbers of topologies to consider

Taxa	Unrooted binary trees	Rooted binary trees
3	1	3
4	3	15
5	15	105
6	105	945
7	945	10,395
8	10,395	135,135
9	135,135	2,027,025
10	2,027,025	3×10^7
15	7×10^{12}	2×10^{14}
20	2×10^{20}	8×10^{21}
50	3×10^{74}	
100	2×10^{182}	
1,000	2×10^{2860}	
10,000	8×10^{38658}	
1,000,000	1×10^{5866723}	

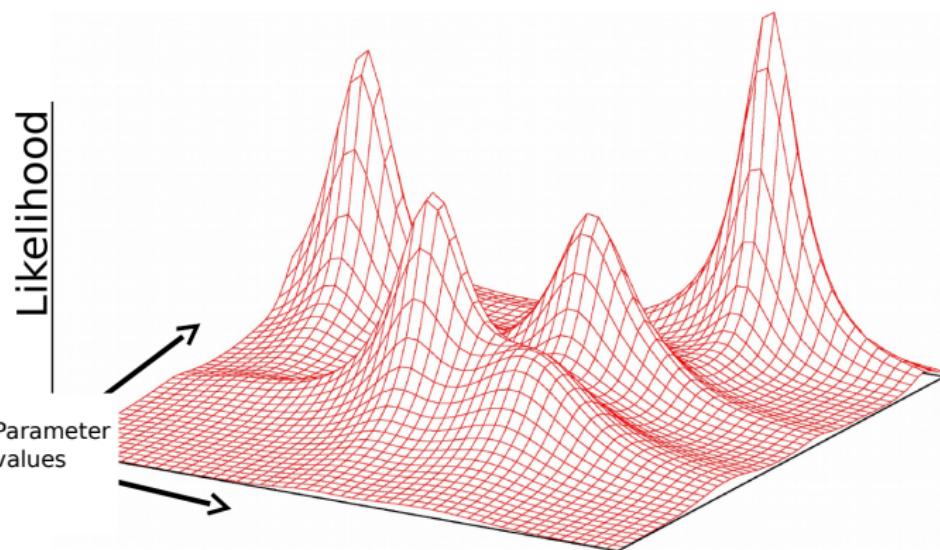
Enormous numbers of topologies to consider

Taxa	Unrooted binary trees	Rooted binary trees
3	1	3
4	3	15
5	15	105
6	105	945

it is estimated that there are between 10^{78} to 10^{82} atoms in the known, observable universe.

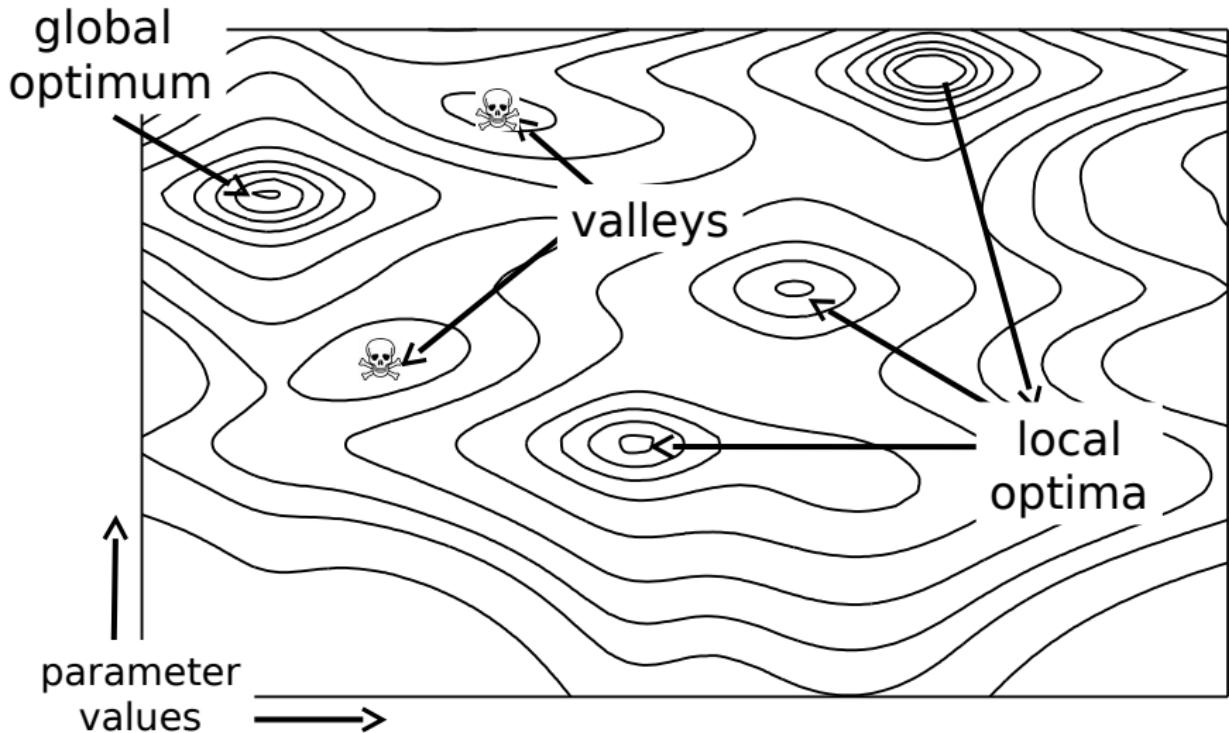
Taxa	Unrooted binary trees	Rooted binary trees
10	2,027,025	3×10^7
15	7×10^{12}	2×10^{14}
20	2×10^{20}	8×10^{21}
50	3×10^{74}	
100	2×10^{182}	
1,000	2×10^{2860}	
10,000	8×10^{38658}	
1,000,000	1×10^{5866723}	

There may be multiple local likelihood optima



(From Zwickl)

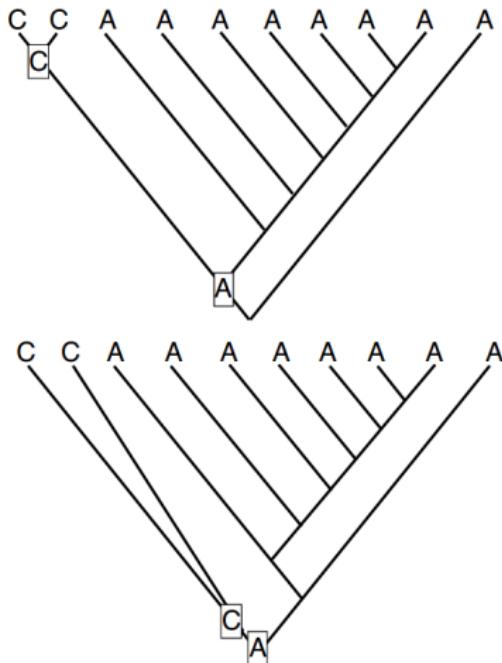
A likelihood surface (from above)



Need to maximize the likelihood for each topology

- ▶ Update numerical parameters of the model of sequence evolution
- ▶ Branch-length parameters

The Relevance of Branch Lengths



(From Swofford)

Neat widgets created by Mark Holder:

<http://phylo.bio.ku.edu/mephytis/brlen-opt.html>

<http://phylo.bio.ku.edu/mephytis/tree-opt.html>

How do algorithms for Maximum Likelihood phylogenetics estimation solve these problems?

How do you know that you have gotten the ML tree?

How do algorithms for Maximum Likelihood phylogenetics estimation **attempt to** solve these problems?

How do you know that you have gotten the ML tree?
you don't!

How do algorithms for Maximum Likelihood phylogenetics estimation **attempt to** solve these problems?

How do you know that you have gotten the ML tree?
you don't!

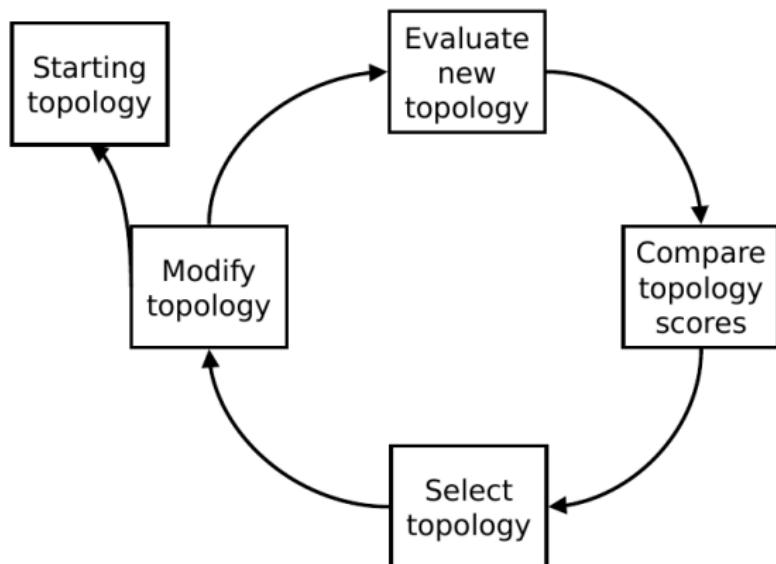
Use a heuristic search to find the best tree you can.

The general concept of heuristic tree search:

1. Start with a tree
2. Calculate the likelihood of that tree given your data (alignment)
3. Look at some trees that are similar
4. Calculate the likelihood for those trees
5. See if you did any better! Return to step 3.

Heuristic runtimes

$$\text{Inference time} = \# \text{ of topologies to evaluate} \times \text{time to evaluate each}$$



Both are strongly a function of the # of sequences when calculating maximized likelihood

Questions for a heuristic search:

- ▶ Where to start the search?

Questions for a heuristic search:

- ▶ Where to start the search?
- ▶ How are new trees proposed?

Questions for a heuristic search:

- ▶ Where to start the search?
- ▶ How are new trees proposed?
- ▶ How do we decide to continue looking at trees at are similar to your new tree, or to your old tree?

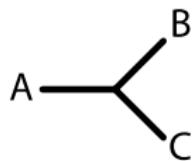
Questions for a heuristic search:

- ▶ Where to start the search?
- ▶ How are new trees proposed?
- ▶ How do we decide to continue looking at trees at are similar to your new tree, or to your old tree?
- ▶ How do you know if you are done?

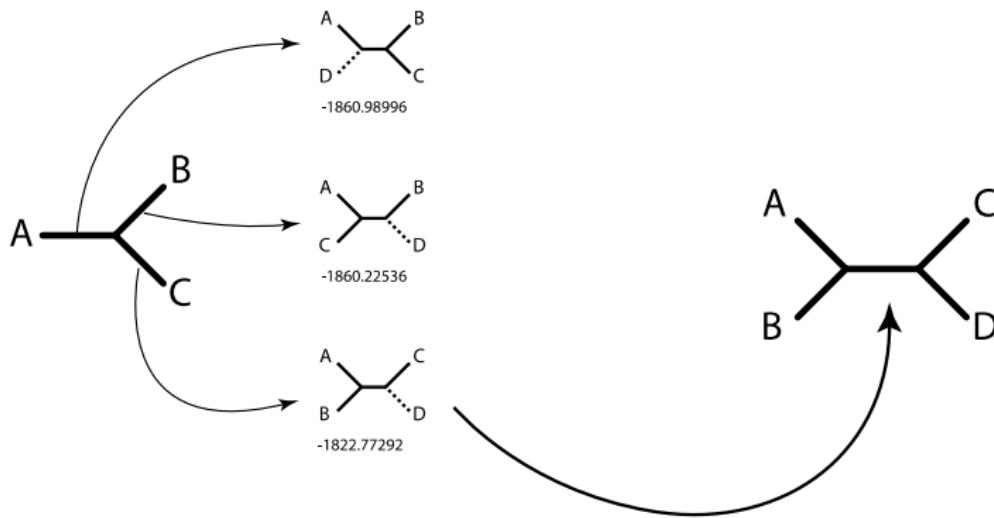
Where to start the search?

- ▶ User supplied starting tree
- ▶ Star decomposition or Stepwise Addition
- ▶ A randomly chosen tree

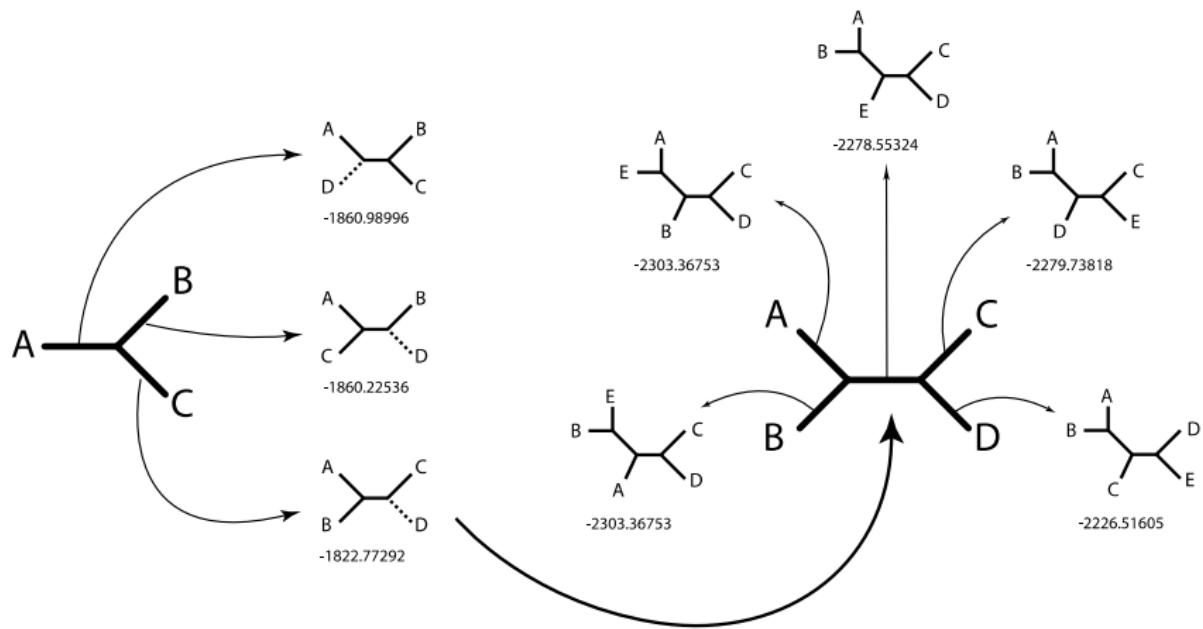
Stepwise addition



Stepwise addition



Stepwise addition



(slide from POL)

Stepwise addition

- Greedy, but can introduce a new taxon on the path between taxa that have already been joined.
- The tree can depend on the input order of the taxa
- Number of trees scored for N taxa :

$$\begin{aligned}\# \text{ trees scored} &= \sum_{i=3}^{N-1} (2i - 3) \\ &= (N-1)(N-3)\end{aligned}$$

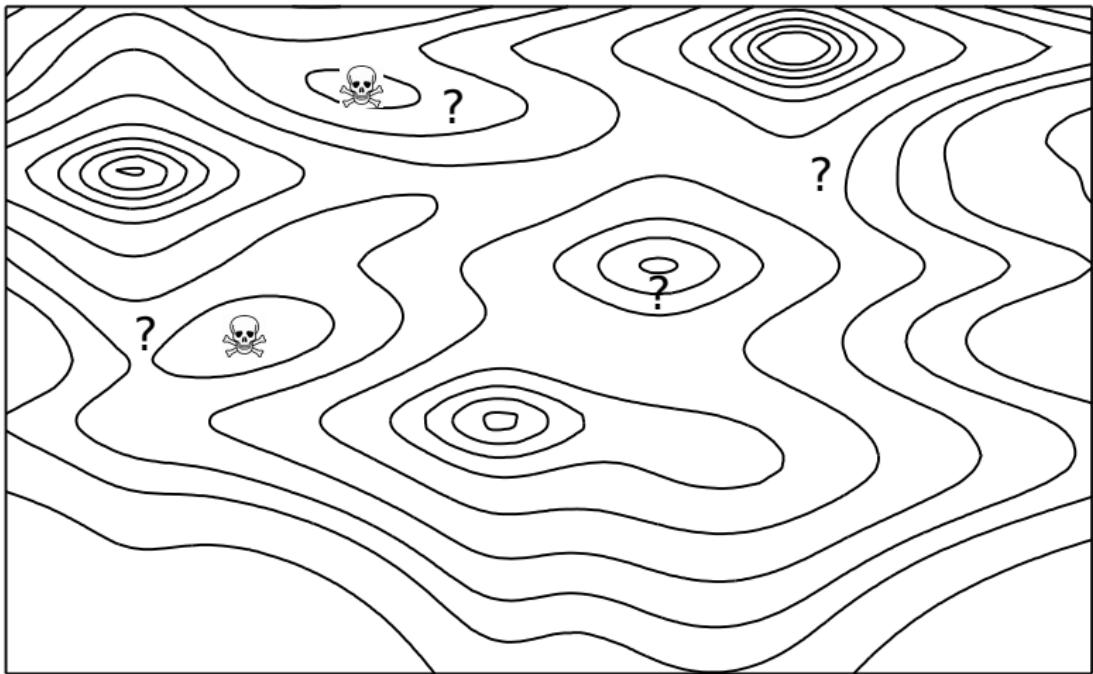
Thus, stepwise addition is $O(N^2)$. For N=10:

$$63 = 3 + 5 + 7 + 9 + 11 + 13 + 15$$

Does your starting tree matter?

- ▶ Can help escape local optima
- ▶ When data is uninformative, bias in starting tree can affect estimate

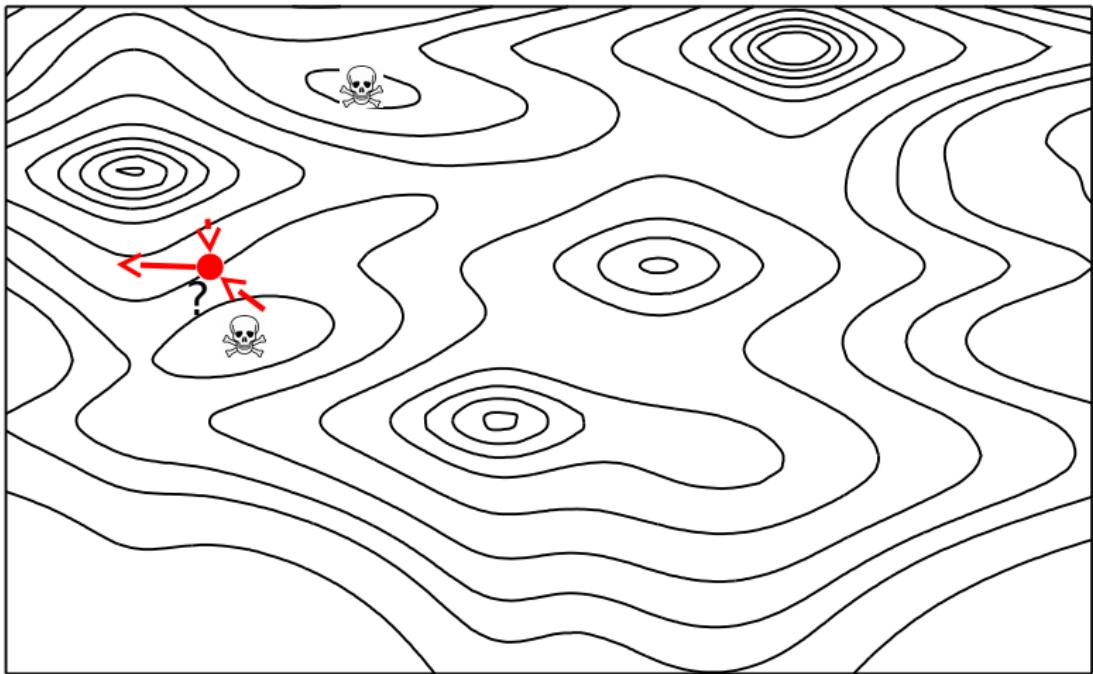
Heuristics: starting point



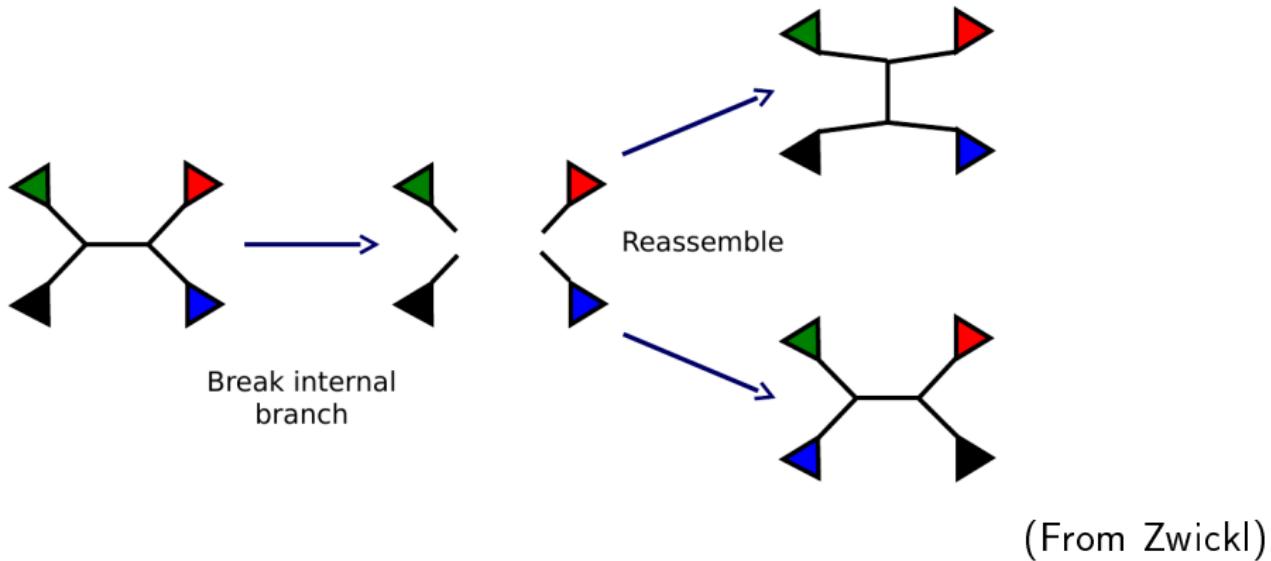
How are new topologies proposed?

- ▶ Branch swapping and tree rearrangement

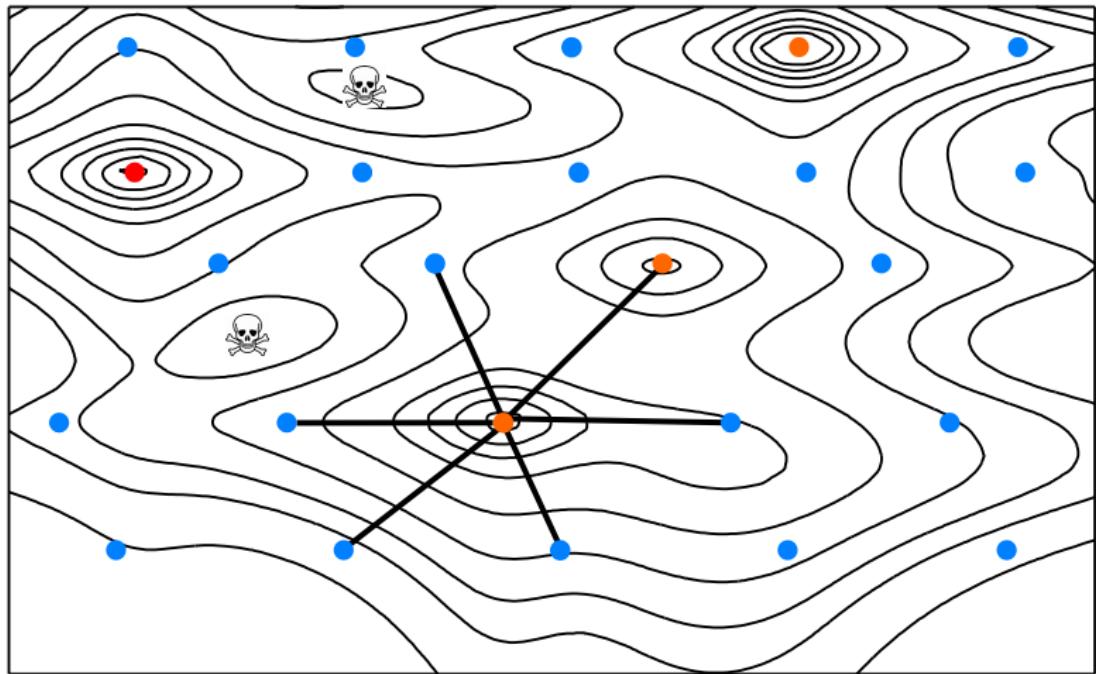
Heuristics: proposing new values



Nearest neighbor interchange (NNI)

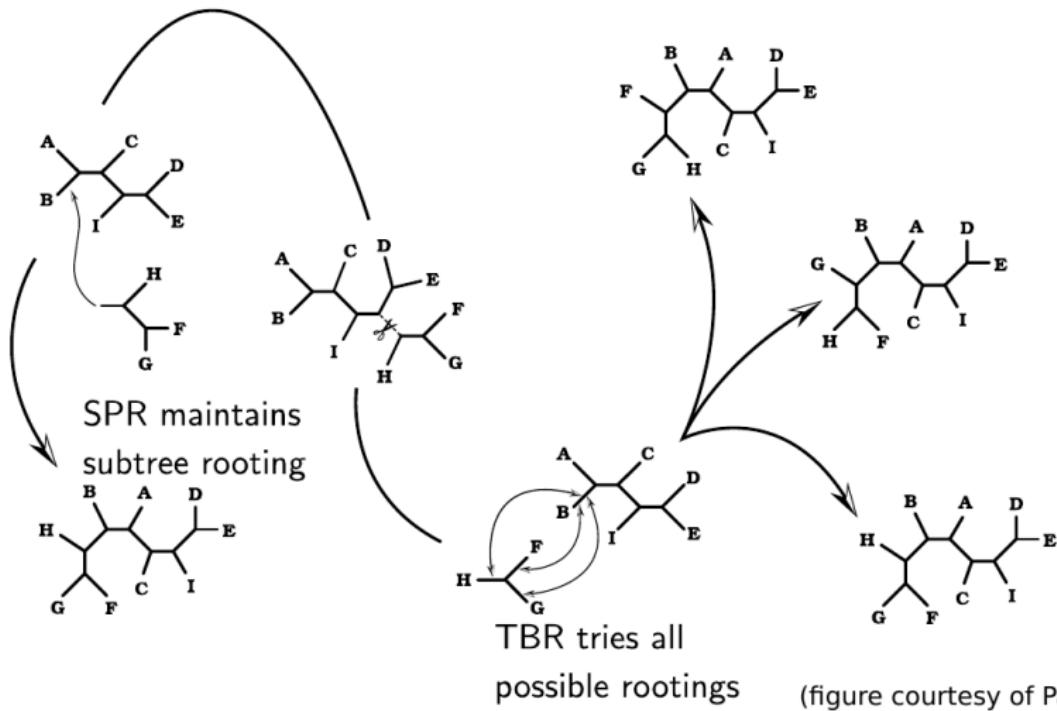


NNI Treespace



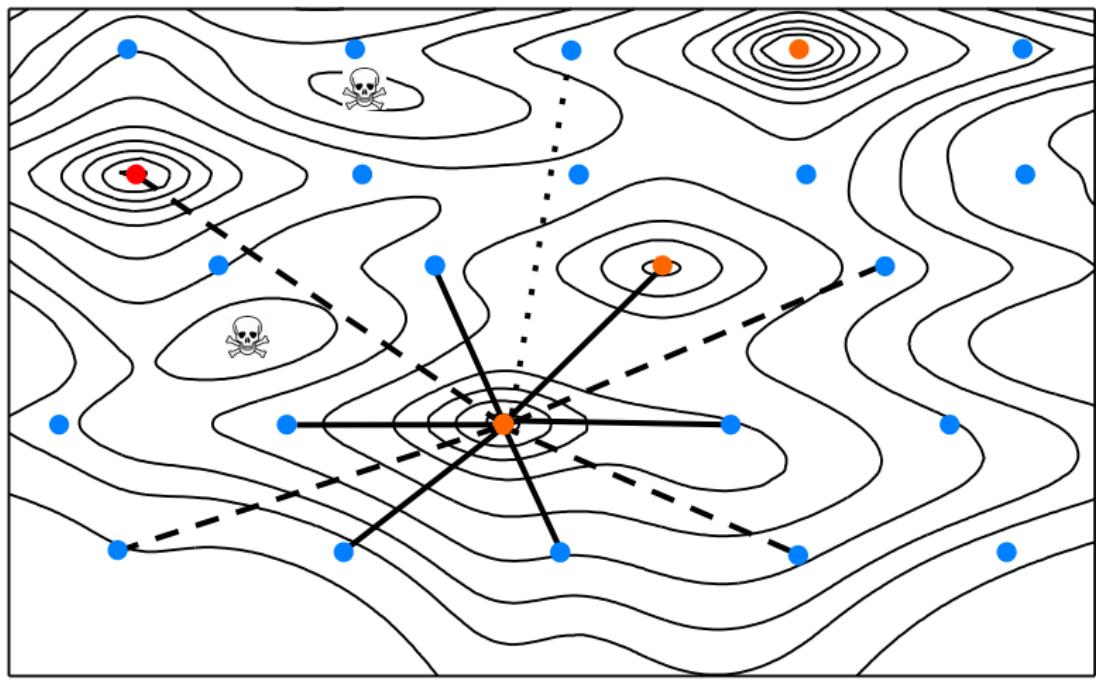
NNI —————

Subtree Pruning Refactoring (SPR) Tree Bisection Reconnection (TBR)



(figure courtesy of Paul Lewis)

SPR/TBR moves in NNI treespace



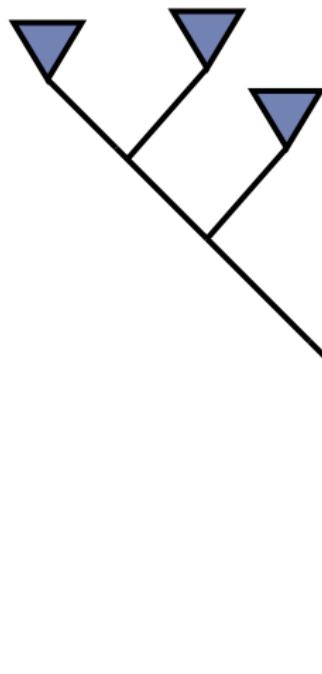
NNI —————

SPR - - - - -

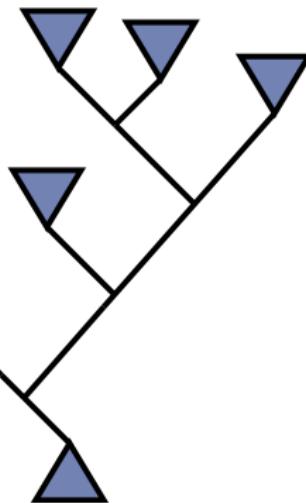
TBR · · · · ·

Re-arranging your tree requires updating branch lengths and evolutionary model parameters.

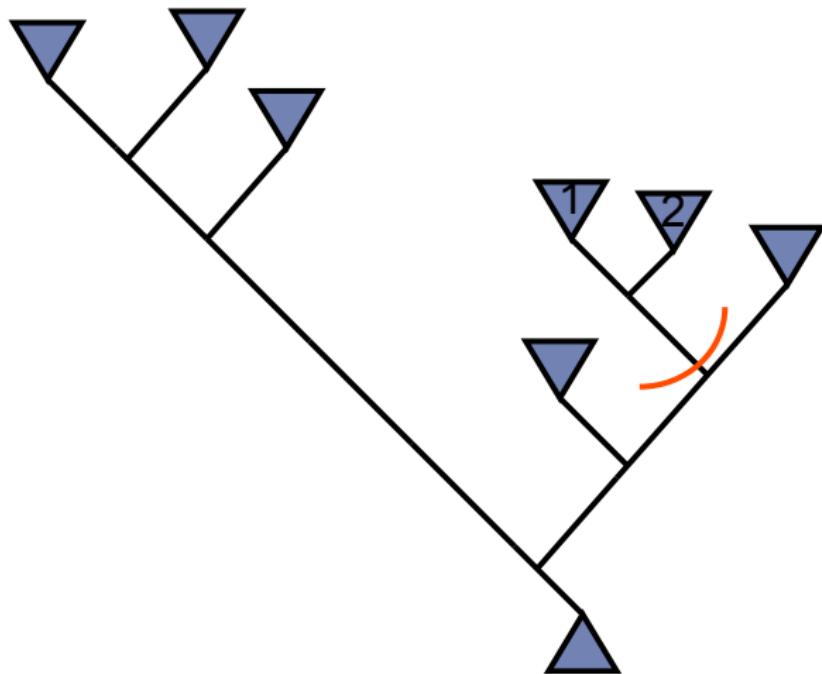
Searching with approximate likelihoods



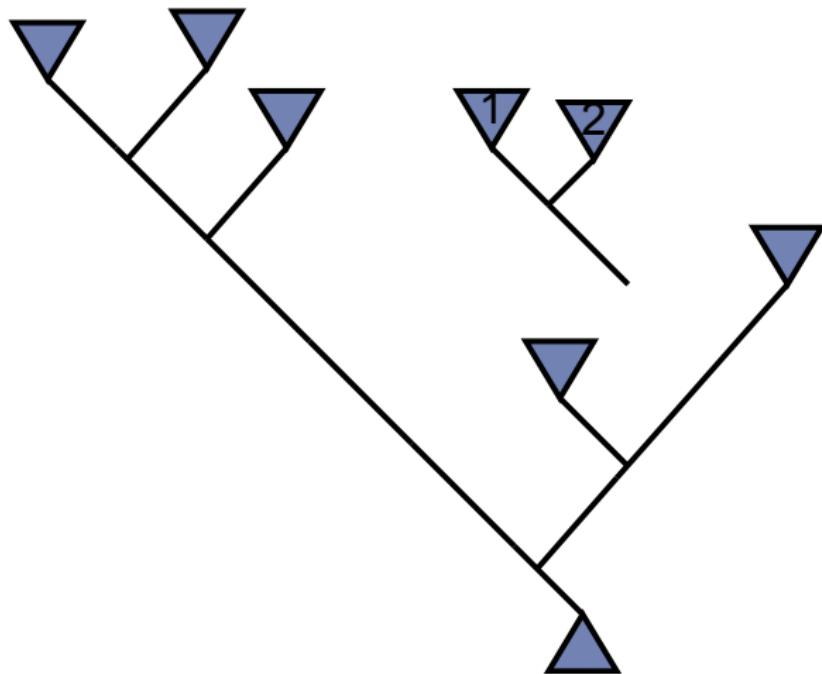
Branch lengths are
optimized on a starting
topology



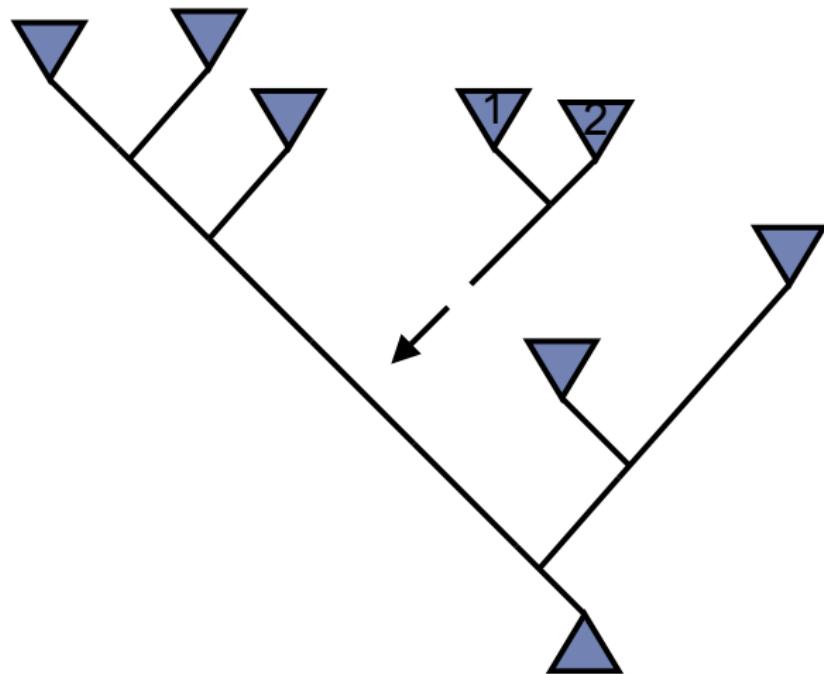
Altering the tree: subtree pruning-regrafting (SPR)



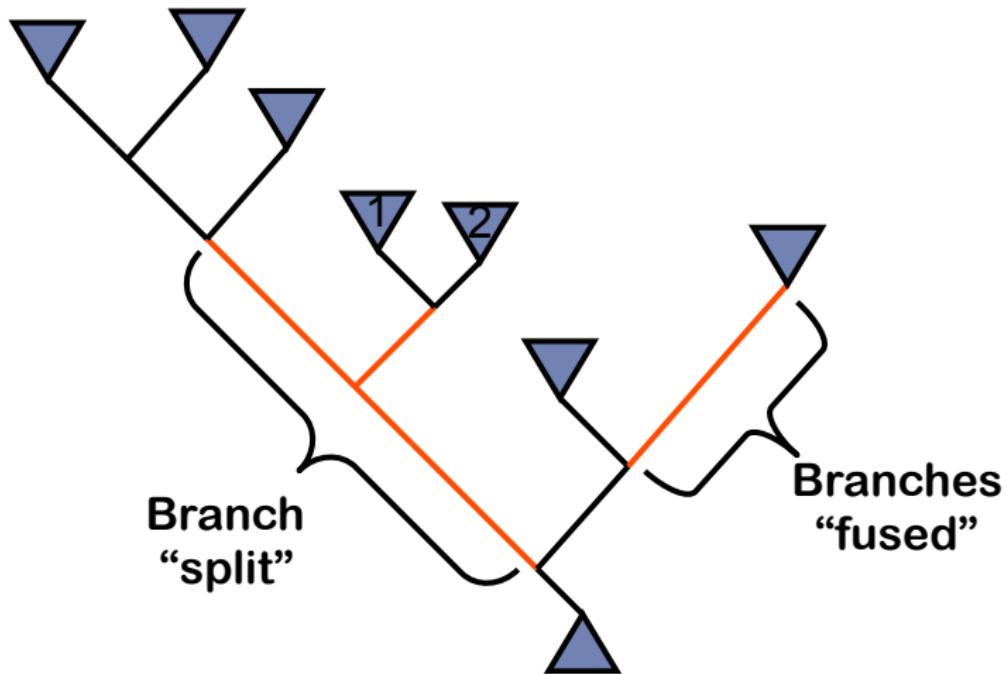
Altering the tree: subtree pruning-regrafting (SPR)



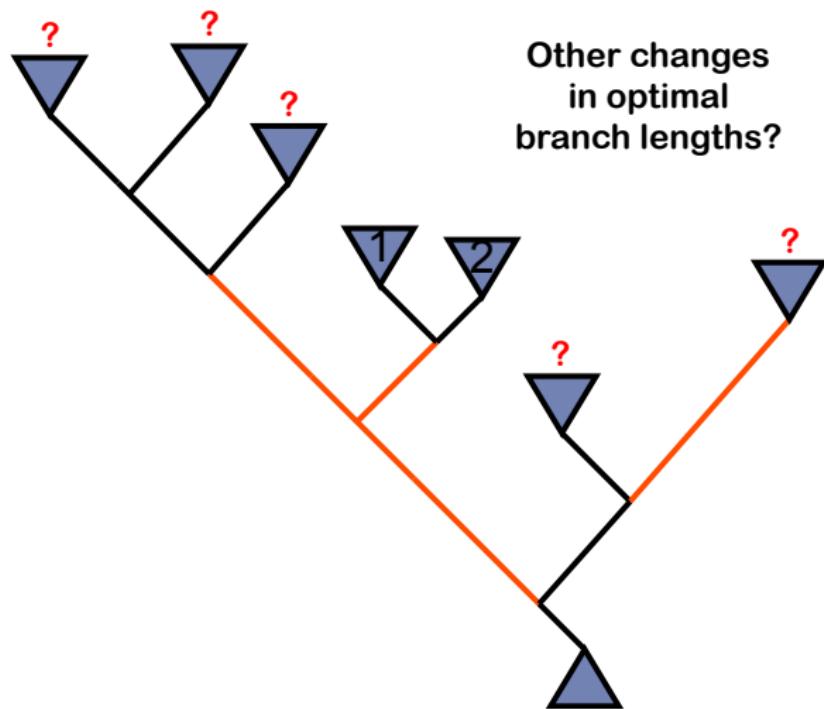
Altering the tree: subtree pruning-regrafting (SPR)



Scoring and optimizing the new topology



Scoring and optimizing the new topology



Localizing branch length optimization important for speed of analysis

How do you decide if you should accept a new tree?

- ▶ Hill climbing: likelihood score is better (RAxML)
- ▶ Computational analog of evolution by natural selection (Garli)

How do you know if you are done?

- ▶ Stop tree search when likelihood stops improving.

How do you know if you are done?

- ▶ Stop tree search when likelihood stops improving.
- ▶ Searches are stochastic, so there is no guarantee that any search finds the true maximum likelihood topology and parameter values!

How do you know if you are done?

- ▶ Stop tree search when likelihood stops improving.
- ▶ Searches are stochastic, so there is no guarantee that any search finds the true maximum likelihood topology and parameter values!
- ▶ Continue searching until you run at least one additional search that finds the same topology as the best overall result.

In lab today we will discuss and apply two software packages that estimate ML trees

- ▶ Garli (Zwickl, 2006)
 - ▶ Stochastic, genetic algorithm-like approach
 - ▶ Computational analog of evolution by natural selection.
- ▶ RAxML (Stamatakis, 2006)
 - ▶ Hill-climbing algorithm
 - ▶ GTR+CAT approximation major speedup over GTR+G
 - For modeling rate heterogeneity across very large trees (e.g., hundreds of taxa), and is not recommended for smaller trees.
 - Different than Lartillot CAT model using empirical amino acid profiles (named independently around same time)

ML tree inference software:

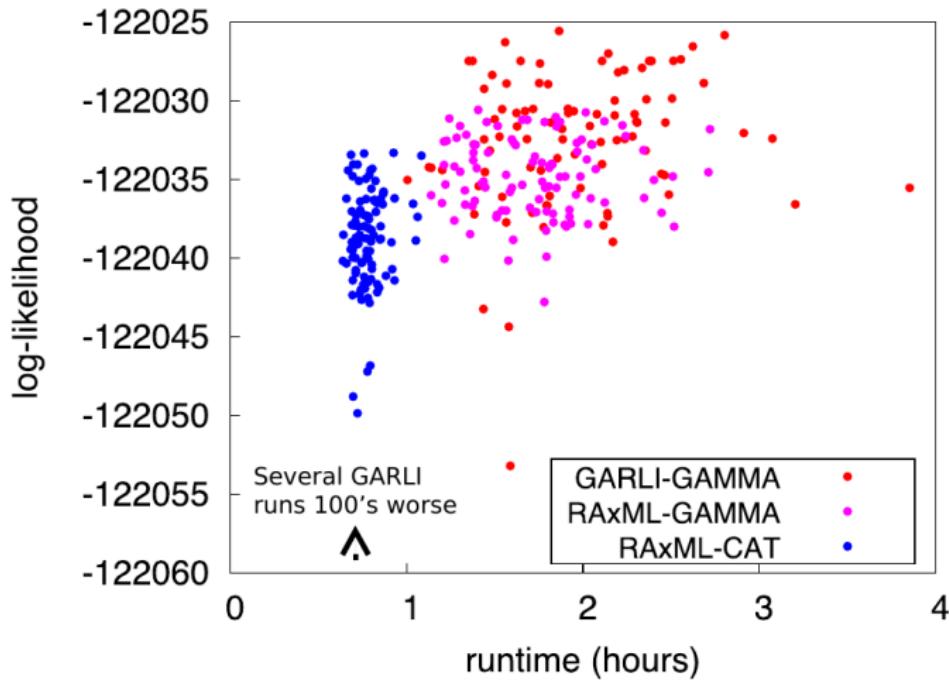
For small datasets (< 50 taxa), all of the ML tree inference programs perform well

For large datasets (hundreds of sequences):

- ▶ PAUP* is very rigorous, but slowest
- ▶ RAxML is generally the fastest
- ▶ GARLI often has a slight edge over RAxML in optimality (although often more variability)

Simulations by Zwickl (Garli)

Performance comparison:
228 taxon x 4811 nucleotide dataset

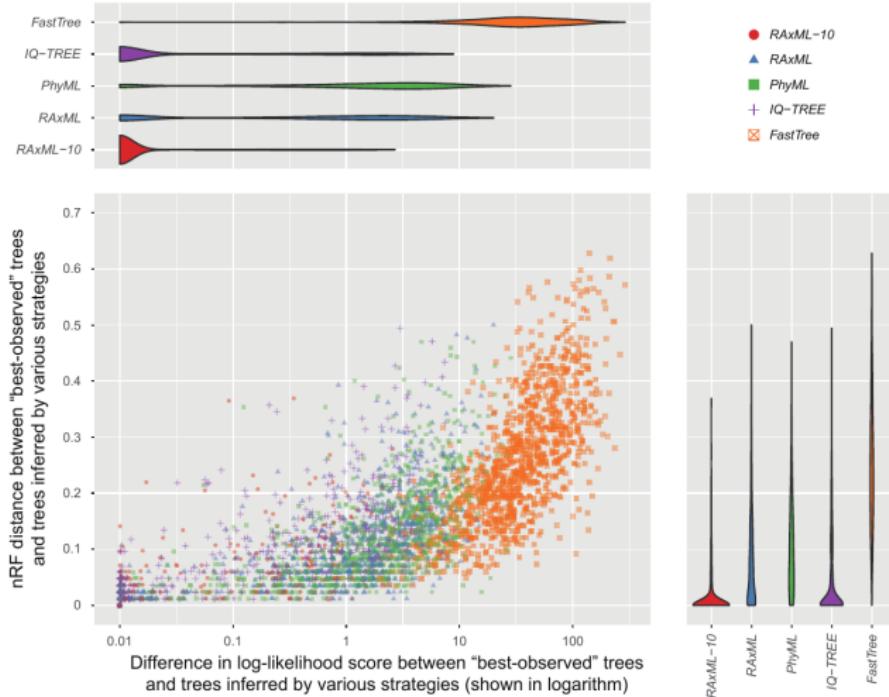


ML tree inference software:

For VERY large datasets (1000+sequences):

- ▶ RAxML/EXaML (Kozlov et al., 2015) is very efficient, especially with multiple runs
- ▶ IQ-TREE (Nguyen et al., 2015) also fast and relatively accurate
- ▶ FASTTREE(Price et al., 2009) is very fast, but (excessive) tradeoffs with accuracy (per Zhou et al. (2017))

Figure 3



Log-likelihood score differences between inferred trees and “best-observed” trees plotted against topological distances.(Zhou et al., 2017)

Summary

- ▶ For >15 sequences, an unfathomably large number of trees are possible.
- ▶ We have to rely on heuristics that are not guaranteed to find the actual (“global”) optimal solution.
- ▶ We have control on how thorough our searches are
- ▶ You should conduct multiple searches to look for evidence that you are not finding trees which are local optima.

Questions?

Bootstrapping

Some resources related to this talk

A Mendeley Group of papers related to topology testing on trees.

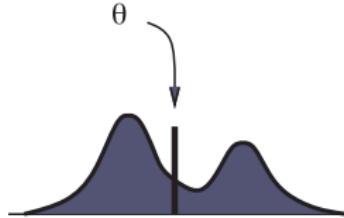
A <http://phylo.bio.ku.edu/woodshole/> has the beginnings of an annotated bibliography and some other notes.

The source for all the documents for my talk is at:

<https://github.com/mtholder/TreeTopoTestingTalks>

The bootstrap

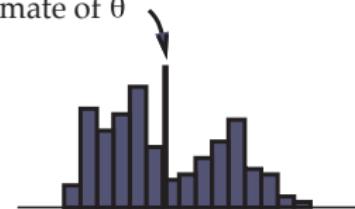
(unknown) true value of



(unknown) true distribution

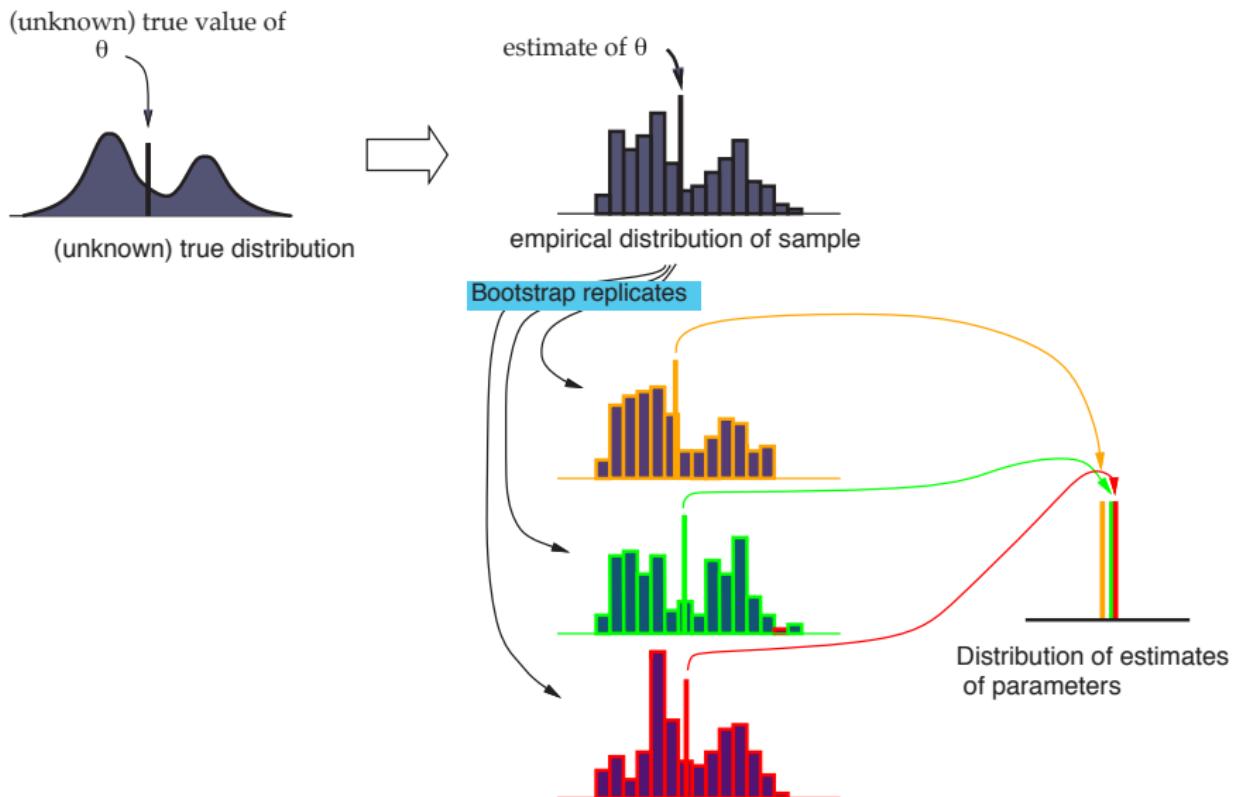


estimate of θ



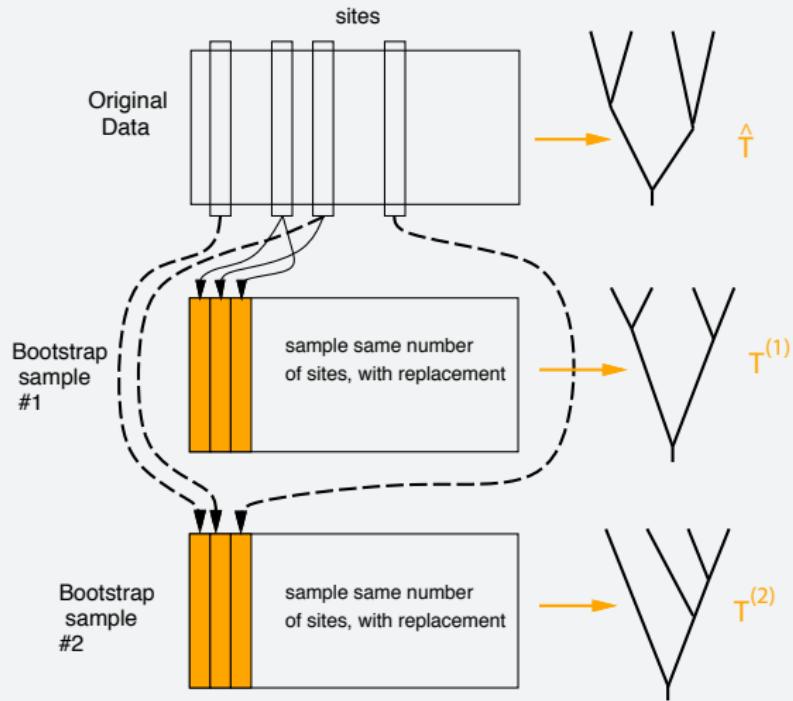
empirical distribution of sample

The bootstrap



Slide from Joe Felsenstein

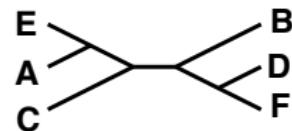
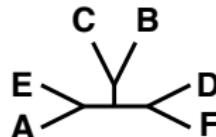
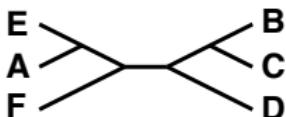
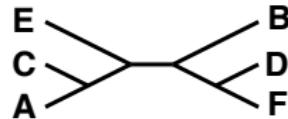
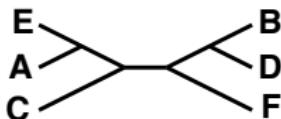
The bootstrap for phylogenies



Slide from Joe Felsenstein

The majority-rule consensus tree

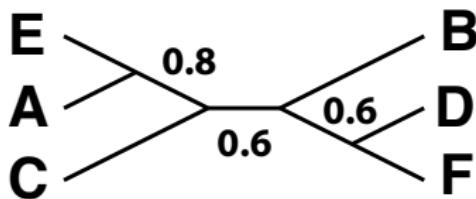
Trees:



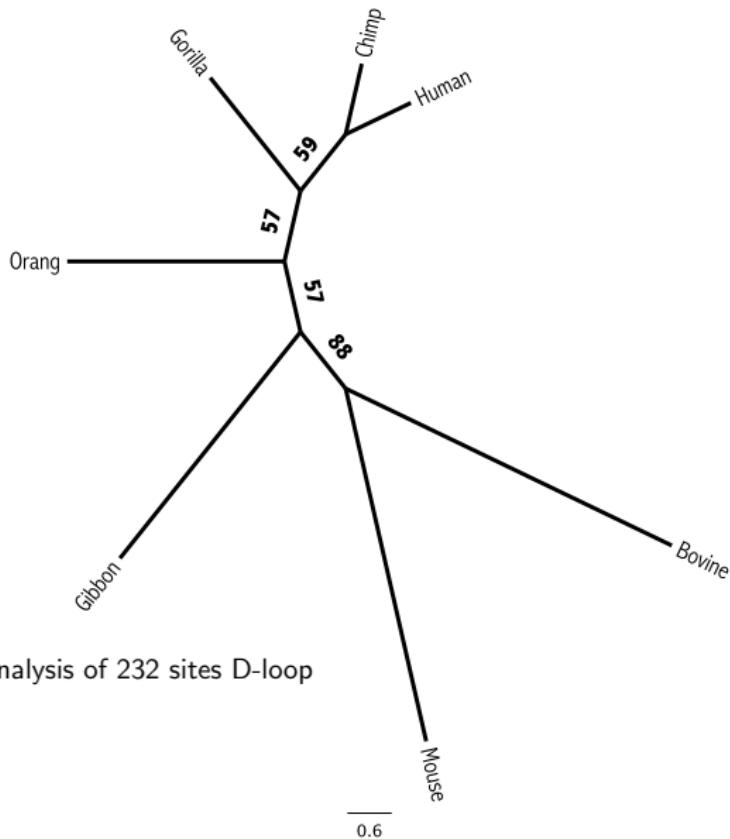
How many times each partition of species is found:

AE BCD	4
ACE BDF	3
ACEF BD	1
AC BDEF	1
AEF BCD	1
ADEF BC	2
ABCE DF	3

4
3
1
1
1
2
3



Slide from Joe Felsenstein



From Hasegawa's analysis of 232 sites D-loop

Bootstrapping for branch support

- Typically a few hundred bootstrap, pseudoreplicate datasets are produced.
- Less thorough searching is faster, but will usually artificially lower bootstrap proportions (BP). However, Anisimova et al. (2011) report that RAxML's rapid bootstrap algorithm may inflate BP.
- “Rogue” taxa can lower support for many splits – you do not have to use the majority-rule consensus tree to summarize bootstrap confidence statements.

a brief digression into file formats

Newick

- ▶ Parenthetical tree format
- ▶ Rooted vs. unrooted trees are not differentiated
- ▶ Some programs interpret polytomy at root as 'unrooted'
- ▶ Branches and nodes not well differentiated
- ▶ A name can contain any characters except blanks, colons, semicolons, parentheses, and square brackets

Nexus

- ▶ Starts with `#nexus`
- ▶ Can contain blocks of alignments, trees, commands, and more!
- ▶ Blocks between 'begin' and 'end'
- ▶ Trees in Newick format, prepended with `[&U]` unrooted or `[&R]` rooted

Nexus

```
#nexus
...
begin taxa;
  dimensions ntax=5;
  taxlabels
    Giardia
    Thermus
    Deinococcus
    Sulfolobus
    Haobacterium
  ;
end;

#nexus
...
begin data;
  dimensions ntax=5 nchar=54;
  format datatype=dna missing=? gap=-;
  matrix
    Ephedra      TTAAGCCATGCATGCTAAGTATGAACTAATTCCAACGGTGAACACTGCGGATG
    Gnetum        TTAAGCCATGCATGTCTATGTACGAACTAATC-AGAACGGTGAACACTGCGGATG
    Welwitschia   TTAAGCCATGCACGTGAAGTATGAACTAGTC-GAACGGTGAACACTGCGGATG
    Ginkgo         TTAAGCCATGCATGTGAAGTATGAACTCTTTACAGACTGTGAAACTGCGAATG
    Pinus          TTAAGCCATGCATGCTAAGTATGAACTAATTGCAAGCTGTGAAACTGCGGATG
    [-----+--10|-----+--20|-----+--30|-----+--40|-----+--50|-----]
  ;
end;
```

http://hydrodictyon.eeb.uconn.edu/eebedia/index.php/Phylogenetics:_NEXUS_Format

Nexus

```
#nexus
...
begin trees;
    translate
        1 Ephedra,
        2 Gnetum,
        3 Welwitschia,
        4 Ginkgo,
        5 Pinus
    ;
    tree one = [&U] (1,2,(3,(4,5));
    tree two = [&U] (1,3,(5,(2,4));
end;
```

```
#nexus
...
begin sets;
    charset trnL_intron = 562-4226;
    taxset gnetales = Ephedra Gnetum Welwitschia;
end;
```

http://hydrodictyon.eeb.uconn.edu/eebedia/index.php/Phylogenetics:_NEXUS_Format

NeXML

- ▶ Phylogenetic data as XML
- ▶ Can capture all information from Nexus
- ▶ Full semantic annotation
- ▶ Easily extensible

NeXML

Computer readable, but not very human readable

```
<otu about="#otu99" id="otu99" label="Parupeneus barberinoides">
  <meta datatype="xsd:string" property="ot:originalLabel" xsi:type="nex:LiteralMeta">Parupeneus
  <meta datatype="xsd:int" property="ot:ottId" xsi:type="nex:LiteralMeta">758968</meta>
  <meta datatype="xsd:string" property="ot:ottTaxonName" xsi:type="nex:LiteralMeta">Parupeneus b
</otu>
</otus>
<trees about="#trees1" id="trees1" otus="otus1">
  <tree about="#tree1" id="tree1" label="Untitled (tree1)" xsi:type="nex:FloatTree">
    <meta datatype="xsd:string" property="ot:branchLengthDescription" xsi:type="nex:LiteralMeta"/>
    <meta datatype="xsd:string" property="ot:branchLengthMode" xsi:type="nex:LiteralMeta">ot:undef
    <meta datatype="xsd:string" property="ot:curatedType" xsi:type="nex:LiteralMeta">Bayesian infe
    <meta datatype="xsd:string" property="ot:inGroupClade" xsi:type="nex:LiteralMeta">node2</meta>
    <meta datatype="xsd:string" property="ot:nodelabelMode" xsi:type="nex:LiteralMeta"/>
    <meta datatype="xsd:string" property="ot:nodelabellTimeUnit" xsi:type="nex:LiteralMeta"/>
    <meta datatype="xsd:string" property="ot:outGroupEdge" xsi:type="nex:LiteralMeta"/>
    <meta datatype="xsd:string" property="ot:specifiedRoot" xsi:type="nex:LiteralMeta">nodel1</meta
    <meta datatype="xsd:boolean" property="ot:unrootedTree" xsi:type="nex:LiteralMeta">false</meta
    <node about="#node1" id="node1" root="true"/>
    <node about="#node2" id="node2"/>
    <node about="#node144" id="node144"/>
    <node about="#node145" id="node145"/>
    <node about="#node146" id="node146"/>
    <node about="#node147" id="node147"/>
    <node about="#node148" id="node148"/>
    <node about="#node149" id="node149"/>
    <node about="#node150" id="node150"/>
    <node about="#node151" id="node151"/>
    <node about="#node152" id="node152"/>
    <node about="#node153" id="node153"/>
    <node about="#node154" id="node154"/>
    <node about="#node155" id="node155" otu="otu72">
      <meta datatype="xsd:boolean" property="ot:isLeaf" xsi:type="nex:LiteralMeta">true</meta>
    </node>
    <node about="#node156" id="node156" otu="otu73">
      <meta datatype="xsd:boolean" property="ot:isLeaf" xsi:type="nex:LiteralMeta">true</meta>
    </node>
    <node about="#node157" id="node157" otu="otu74">
      <meta datatype="xsd:boolean" property="ot:isLeaf" xsi:type="nex:LiteralMeta">true</meta>
    </node>
    <node about="#node158" id="node158"/>
    <node about="#node159" id="node159" otu="otu75">
      <meta datatype="xsd:boolean" property="ot:isLeaf" xsi:type="nex:LiteralMeta">true</meta>
    </node>
    <node about="#node160" id="node160" otu="otu76">
      <meta datatype="xsd:boolean" property="ot:isLeaf" xsi:type="nex:LiteralMeta">true</meta>
    </node>
```

Phylip (sequence data format)

- ▶ First line must be two integers: <number of taxa> <number of sites>
- ▶ Sequence ID followed by spaces up to 10 char.
- ▶ No duplicate names
- ▶ Relaxed phylip up to 250 characters followed by a space

5 42

Turkey	AAGCTNGGGC	ATTCAGGGT	GAGCCGGGC	AATACAGGGT	AT
Salmo	gairAAGCCTTGGC	AGTGCAGGGT	GAGCCGTGGC	CGGGCACGGT	AT
H. Sapiens	ACCGGTTGGC	CGTTCAGGGT	ACAGGTTGGC	CGTTCAGGGT	AA
Chimp	AAACCCTTGC	CGTTACGCTT	AAACCGAGGC	CGGGACACTC	AT
Gorilla	AAACCCTTGC	CGGTACGCTT	AAACCATTGC	CGGTACGCTT	AA

Phylip interleaved

5 42
Turkey AAGCTNGGGC ATTCAGGGT
Salmo gairAAGCCTTGGC AGTGCAGGGT
H. SapiensACCGGTTGGC CGTTCAGGGT
Chimp AAACCCTTGC CGTTACGCTT
Gorilla AAACCCATTGC CGGTACGCTT

GAGCCCGGGC AATACAGGGT AT
GAGCCGTGGC CGGGCACGGT AT
ACAGGTTGGC CGTTCAGGGT AA
AAACCGAGGC CGGGACACTC AT
AAACCATTGC CGGTACGCTT AA

Phylip sequential

5 42
Turkey AAGCTNGGGC ATTCAGGGT
GAGCCCGGGC AATACAGGGT AT
Salmo gairAAGCCTTGGC AGTGCAGGGT
GAGCCGTGGC CGGGCACGGT AT
H. SapiensACCGGTTGGC CGTTCAGGGT
ACAGGTTGGC CGTTCAGGGT AA
Chimp AAACCCTTGC CGTTACGCTT
AAACCGAGGC CGGGACACTC AT
Gorilla AAACCCATTGC CGGTACGCTT
AAACCATTGC CGGTACGCTT AA

Fasta (sequence data format)

- ▶ Description line before each sequence starts with (">") symbol in the first column

```
>AB000263 |acc=AB000263|descr=Homo sapiens mRNA for prepro cortistatin like peptide, complete cds.|len=368  
ACAAGATGCCATTGTCCCCCGGCCTCTGCTGCTGCTCTCCGGGGCACGGCCACCGCTGCCCTGCC  
CCTGGAGGGTGGCCCCACCGGCCGAGACAGCGAGCATATGCAGGAAGCGGCAGGAATAAGGAAAAGCAGC  
CTCCTGACTTTCTCGCTTGGTGGTTGAGTGGACCTCCCAGGCCAGTGCCGGGCCCCCTCATAGGAGAGG  
AAGCTCGGGAGGTGGCAGGCAGGAAAGGCGCACCCCCCCCAGCAATCCGCGCGCCGGGACAGAACGCC  
CTGCAGGAACCTCTCTGGAAGACCTTCTCCTCTGCAAATAAACCTCACCATGAATGCTCACGCAAG  
TTTAATTACAGACCTGAA
```

Computer lab takehomes:

- ▶ Perform ML phylogenetics search
- ▶ Compare searches
- ▶ Work with variety of phylogenetic software and file formats
- ▶ Bootstrapping
- ▶ Consequences of model misspecification
- ▶ Analyzing data on shared cluster

- Kozlov, A. M., Aberer, A. J., and Stamatakis, A. (2015). ExaML version 3: a tool for phylogenomic analyses on supercomputers. *Bioinformatics*, 31(15):2577–2579.
- Nguyen, L.-T., Schmidt, H. A., von Haeseler, A., and Minh, B. Q. (2015). IQ-TREE: A Fast and Effective Stochastic Algorithm for Estimating Maximum-Likelihood Phylogenies. *Molecular Biology and Evolution*, 32(1):268–274.
- Price, M. N., Dehal, P. S., and Arkin, A. P. (2009). FastTree: Computing Large Minimum Evolution Trees with Profiles instead of a Distance Matrix. *Molecular Biology and Evolution*, 26(7):1641–1650.
- Stamatakis, A. (2006). RAxML-VI-HPC: maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models. *Bioinformatics*, 22(21):2688–2690.
- Zhou, X., Shen, X.-X., Hittinger, C. T., and Rokas, A. (2017). Evaluating Fast Maximum Likelihood-Based Phylogenetic Programs Using Empirical Phylogenomic Data Sets. *bioRxiv*, page 142323.

Zwickl, D. J. (2006). GARLI—genetic algorithm for rapid likelihood inference. See <http://www.bio.utexas.edu/faculty/antisense/garli/Garli.html>.