A Statistical Test for Clades in Phylogenies

By
Thurston Hou Yeen Dang

A thesis submitted in partial satisfaction of the requirements for the degree of Master of Arts in Statistics in the Graduate Division of the University of California, Berkeley

Committee in charge:
Professor Elchanan Mossel, Chair
Professor Allan Sly
Professor Satish Rao

Spring 2013
Abstract

A Statistical Test for Clades in Phylogenies

by

Thurston Hou Yeen Dang

Master of Arts in Statistics

University of California, Berkeley

Professor Elchanan Mossel, Chair

This thesis investigates testing the likelihood of a phylogenetic tree by comparison to its subtree pruning and regrafting (SPR) neighbors, with or without re-optimizing branch lengths. This is inspired by aspects of Bayesian significance tests, and the use of SPRs for finding maximum likelihood trees.

Through a number of simulations with the Jukes-Cantor model on various topologies, it is observed that the SPR tests are informative, and reasonably fast compared to searching for the maximum likelihood tree. This suggests that the SPR tests would be a useful addition to the suite of existing statistical tests, for identifying potential issues with topologies.
# Table of Contents

Introduction .................................................................................................................. 1
Phylogenies .................................................................................................................. 1
A statistical test for clades .......................................................................................... 2
Heuristic Bayesian argument ....................................................................................... 2
Subtree pruning and regrafting .................................................................................... 3
Limitations .................................................................................................................. 7
Method ....................................................................................................................... 8
Implementation .......................................................................................................... 8
Scenarios .................................................................................................................... 8
Accuracy of the SPR tests .......................................................................................... 8
Speed of the SPR tests ............................................................................................... 9
Neighborhoods ......................................................................................................... 9
Results ....................................................................................................................... 10
4-species tree with unit length branches .................................................................... 10
C₂ (6-species) .......................................................................................................... 11
C₄ (18-species) ......................................................................................................... 13
Larger trees ............................................................................................................... 14
Neighborhoods ....................................................................................................... 17
Discussion ............................................................................................................... 18
References .............................................................................................................. 20
Appendices .............................................................................................................. 22
Non-monotonic likelihood for the 4-species topology .............................................. 22
Oscillation in likelihood of recovering the correct tree ........................................... 22
Oscillation in indistinguishability ............................................................................. 23
Infinite length branch leads to indistinguishability ................................................ 23
Oscillation of likelihood of having an infinite length branch .................................. 24
Implementation ....................................................................................................... 28
Printing the canonical form ...................................................................................... 28
Tree SPRs ................................................................................................................. 29
Acknowledgements

Thank you to the many people who have helped and inspired me on my computational biology journey, especially Richard Buckland, Rupert Shuttleworth, Catherine Suter, Jennifer Cropley, Simon Keam, Vanessa Venturi, Miles Davenport, Hui Yee Greenaway, Monica Kurniawan, and Elchanan Mossel.

I love you, Beth Cooper.
Introduction

Phylogenies

“A phylogenetic tree is the only figure in On the Origin of Species, evidence of the central importance of such trees to evolutionary biology.”

-- p. 1, Barton, Briggs, Eisen, Goldstein, & Patel (2007)

Phylogenetic trees show the evolutionary relationships between species (e.g., Barton et al., 2007). Phylogenies can be evaluated by various criteria, such as maximum parsimony or minimum evolution, though Felsenstein (1981) advocated for a maximum likelihood approach. For example, consider the tree in Figure 1. Let:

- $S_X$ be the nucleotides at node $X$
- $\pi_{S_X}$ be the prior probability of the bases $S_X$
- $P_{S_XS_A}(v_{XA})$ be the probability of mutating from the bases $S_X$ to $S_A$, given the branch length ($v_{XA})$ between $X$ and $A$

![Figure 1. A simple topology with four species.](image)

As per Felsenstein (1981), the likelihood of the tree is:

$$L = \pi_{S_X} P_{S_XS_A}(v_{XA}) P_{S_XS_Y}(v_{XY}) P_{S_YS_B}(v_{YB}) P_{S_YS_Z}(v_{YZ}) P_{S_ZS_C}(v_{ZC}) P_{S_ZS_D}(v_{ZD})$$

Since we do not know the sequences for the interior nodes $X$, $Y$, or $Z$, we marginalize over all possible assignments of bases:

$$L = \sum_{S_X} \sum_{S_Y} \sum_{S_Z} \pi_{S_X} P_{S_XS_A}(v_{XA}) P_{S_XS_Y}(v_{XY}) P_{S_YS_B}(v_{YB}) P_{S_YS_Z}(v_{YZ}) P_{S_ZS_C}(v_{ZC}) P_{S_ZS_D}(v_{ZD})$$

or more efficiently:

$$L = \sum_{S_X} \pi_{S_X} P_{S_XS_A}(v_{XA}) \left( \sum_{S_Y} P_{S_XS_Y}(v_{XY}) P_{S_YS_B}(v_{YB}) \left( \sum_{S_Z} P_{S_YS_Z}(v_{YZ}) P_{S_ZS_C}(v_{ZC}) P_{S_ZS_D}(v_{ZD}) \right) \right)$$
There are variety of likelihood-based tests for phylogenies (see Goldman, Anderson, & Rodrigo, 2000 for a review). These are useful because, when the topology is questionable, there are often “several reasonable trees” with similar likelihoods (Yang, Goldman, & Friday, 1995).

Furthermore, these tests are practical, because it is possible to obtain such comparison trees through minor modifications to the tree of interest (Foster, 2001). For example, FastTree applies the Shimodaira-Hasegawa test on the three alternate topologies obtained by nearest-neighbor interchanges (NNIs) (Price, n.d.), while PhyML (Guindon, 2010) performs an approximate likelihood ratio test (aLRT), comparing the inferred topology with the second-best topology obtained by an NNI.

A statistical test for clades

Heuristic Bayesian argument

Let:

- \( T \) be the “true” tree
- \( D \) be the data generated from the tree
- \( \hat{T} \) be the tree obtained by maximum likelihood from \( D \) (Figure 2)
- \( S \) be all possible tree topologies that have the same number of species as \( T \)
- \( R \subseteq S \)
- \( g \) be the branch lengths
- \( G(g) \) be a prior over the branch lengths
- \( g' = \arg \max_g P(D|\hat{T}, g) \)

![Figure 2. The true tree, data generated according to the true tree, and the maximum likelihood tree obtained from the data.](image)

The probability that a tree is correct given the sequence data can be calculated with Bayes’ rule (e.g., Velasco, 2008):

\[
P(T = \hat{T}|D) = \frac{\int_G P(D|\hat{T}, g)P(\hat{T})}{\sum_{t \in S} \int_G P(D|t, g)P(t)}
\]
Velasco (2008) noted that almost every published paper assumes a uniform prior over trees:

\[ P(T = \hat{T} | D) = \frac{\int_G P(D | \hat{T}, g)}{\sum_{t \in S} \int_G P(D | t, g)} \]

One approach for calculating the posterior probability is to use Markov Chain Monte Carlo (MCMC), integrating over nuisance parameters such as branch lengths (e.g., Huelsenbeck & Imennov, 2002). We instead restrict our comparison trees to a subset:

\[ P(T = \hat{T} | D) \leq \frac{\int_G P(D | \hat{T}, g)}{\sum_{t \in R} \int_G P(D | t, g)} \]

and assume we have a lot of data, hence the integrated likelihood is similar to the maximum:

\[ \approx \frac{G(g')P(D | \hat{T}, g')}{\sum_{t \in R} G(g')P(D | t, g')} \]

\[ = \frac{P(D | \hat{T}, g')}{\sum_{t \in R} P(D | t, g')} \quad (1) \]

if we are using the same branch lengths \( g' \) for all trees.

The branch lengths \( g' \) are not optimal for every topology. Hence, we can get a better bound by re-optimizing the branch lengths for every tree in the denominator i.e.,

\[ P(T = \hat{T} | D) \leq \frac{G(g')P(D | \hat{T}, g')}{\sum_{t \in R} \max_g G(g)P(D | t, g)} \]

\[ = \frac{P(D | \hat{T}, g')}{\sum_{t \in R} \max_g P(D | t, g)} \quad (2) \]

if we assume a uniform prior over the branch lengths.

**Subtree pruning and regrafting**

The extraordinary number of possible topologies \(((2n - 5)!!)\); e.g., Felsenstein, 1978), which made it impractical to exhaustively search for the maximum likelihood topology, also means that our set of comparison topologies, \( R \), must be drastically smaller.

To serve as the “several reasonable trees” (Yang, Goldman, & Friday, 1995), we propose using the topologies obtained by subtree pruning and regrafting (SPRs). For example, in Figure 3 (top), one possible subtree to prune is \{Z,E,F\}. The subtree can be regrafted onto any of the six remaining edges, such as DY, WX or AW (Figure 3, bottom).
When the substitution model is reversible, the tree is unrooted (e.g., Felsenstein, 1981). This means we could also prune the subtree consisting of all nodes to the left of the midpoint YZ, onto the edges EZ or FZ.

![Figure 3. A topology (top) and three topologies that can be obtained via subtree pruning and regrafting once (bottom).](image)

Using SPRs has a number of advantages:

- The resulting statistical test has an intuitive interpretation: the alternative topologies are obtained by moving a “clan” (a clade/subtree under some possible rooting; e.g., Zhu, Degnan, & Steel, 2011) to another location. For example, in Figure 4, where the subtree “below” Z is attached to BC with a long branch, the likelihood may be similar or even improved when spliced to another edge such as AB, CD, DE or EF.

Matching this intuition, many phylogenetic analysis programs rely on a series of SPRs to “intensively search the tree space” (Hordijk & Gascuel, 2005).

- SPRs model recombination or horizontal gene transfer (e.g., Allen & Steel, 2001).
- SPRs are tractable: they are quadratic in the number of species (see Table 1). This is far fewer than the number of possible topologies (Table 2).

| Type | Splicing to an edge ... | # topologies | # unique, new topologies (ignoring branch lengths) |
|------|-------------------------|--------------|-------------------------------------------------|
| i    | adjacent to the cut edge | 6(n-2)       | 0                                               |
| ii   | one away from the cut edge (NNIs) | 8(n-3)       | 2(n-3)                                          |
| iii  | more than one away from the cut edge | 4(n-3)(n-4) | 4(n-3)(n-4)                                     |
| Total|                          | (2n-3)(2n-4) | 2(n-3)(2n-7)                                    |

Table 1. From Theorem 2.1 (Allen & Steel, 2001). Type ii SPRs are equivalent to nearest neighbor interchanges (NNIs).
| # species | # unique topologies | # new, unique SPRs | Type II + III SPRs |
|-----------|--------------------|--------------------|--------------------|
| 3         | 1                  | 0                  | 0                  |
| 4         | 3                  | 2                  | 8                  |
| 5         | 15                 | 12                 | 24                 |
| 6         | 105                | 30                 | 48                 |
| 7         | 945                | 56                 | 80                 |
| 8         | 10395              | 90                 | 120                |
| 9         | 135135             | 132                | 168                |
| 10        | 2027025            | 182                | 224                |
| 18        | 191898783962511000 | 870               | 960                |
| 34        | 1.12275575285571E+044 | 3782          | 3968                |
| 66        | 1.64749260436028E+107 | 15750        | 16128                |

Table 2. The number of possible topologies and SPRs for n species.

Figure 4. The clan Z is only distantly related to BC.

Let \( SPR(X) \) be \{all new topologies reachable from \( X \) by exactly one SPR operation\} \( \cup \{X\} \). Accordingly, our statistics corresponding to equations (1) and (2) are:

- **SPR\(_{plain} \)** statistic: \( \frac{P(D|\tilde{T},g)}{\sum_{t \in SPR(T)} P(D|t,g)} \)
- **SPR\(_{opt} \)** statistic: \( \frac{P(D|\tilde{T},g)}{\sum_{t \in SPR(T)} \max_{g} P(D|t,g)} \)

For the SPR\(_{plain} \) statistic, since the branch lengths are not re-optimized for the trees in the denominator, there are four copies of each Type (ii) topology (when ignoring branch lengths), with possibly different branch lengths (Figure 5). Since we have no prior information, we weight
each of those topologies by $\frac{1}{4}$. Similarly, since we do not know where along the destination edge to regraft the tree, we use the midpoint of the destination edge.

![Tree Diagram]

Figure 5. The tree “((A,B),(C,D));” (top) and four Type (ii) SPRs (bottom), the latter of which are equivalent to each other. However, the distances between species are not identical.

Both statistics are only intended to provide an approximate upper bound on the likelihood that the topology is correct.

Although SPR_{opt} gives a better bound than SPR_{plain}, and some authors have cautioned against not re-optimizing parameters (Goldman, Anderson, & Rodrigo, 2000), it may be less desirable due to the extra run-time incurred (although partially offset by only needing $\frac{1}{4}$ of the Type (ii) topologies). We therefore evaluate both SPR_{opt} and SPR_{plain} for accuracy and speed.

Our statistics are perhaps most similar to Aris-Brosou’s (2003) Bayes significance test (BST):

$$\log BF_{T,l} \approx \frac{1}{K} \sum_{j=1}^{K} \log p(X|T_j) - \log p(X|T_l)$$

However, the BST uses some different approximations (e.g., a geometric average in place of an arithmetic average, which allowed for the summation of log-likelihoods), is intended to provide decisive evidence if the Bayes factor ($BF_{T,l}$) is sufficiently large, and does not specify the nature of the comparison trees $T_j$. 

6
**Limitations**

Suppose the likelihood of all trees is shown as per Figure 6, and that $SPR(R) = \{M, ..., W\}$, $SPR(F) = \{A, ..., J\}$.\(^1\)

![Figure 6. Hypothetical likelihood of 25 trees.](image)

Ideally, the likelihood for $R$ would be calculated as $\frac{\text{Pr}(D|\text{Tree}_R)}{\sum_{i \in \{A, ..., W\}} \text{Pr}(D|\text{Tree}_i)} = 0.348$. Our approximation using only the SPR neighborhood yields $\frac{\text{Pr}(D|\text{Tree}_R)}{\sum_{i \in \{M, ..., W\}} \text{Pr}(D|\text{Tree}_i)} = 0.455$.

Similarly, the true likelihood for $F$ would be 0.139; using only the SPR neighbors gives 0.595, higher than for the maximum likelihood tree. While this does not violate the property that our SPR statistics provide only an upper bound on the likelihood of correctness, it is undesirable.

For this example, $F$ is the only pathological case, since every other tree would have $F$ and/or $R$ as an SPR neighbor (by symmetry), and therefore have low likelihood according to the test. However, we would expect larger topologies to have many light-tailed local optima: Guindon (2010) had previously observed that the aLRT (which is based on NNIs) performed poorly if a good ML topology had not been found.

---

\(^1\) For concreteness, we refer to SPR neighborhoods, but the arguments apply to any symmetric neighborhood.
Method

Implementation
We implemented the SPR tests with Perl and shell scripts, using the following primitives:

a) parsing a Newick format tree into a tree data structure: BioPerl’s (v. 1.6.901; Stajich, et al., 2002) TreeIO module
b) generating a dataset from a tree: Seq-Gen (Rambaut & Grass, 1997)
c) finding the maximum likelihood tree: DNAML, which is part of Felsenstein’s (2005) PHYLIP (Phylogeny Inference Package).
d) evaluating the likelihood of a tree: DNAML, or with our own implementation.
e) printing the canonical form of a tree
For example, the trees in Figure 5 (bottom) are equivalent when ignoring branch length. However, if we had a naïve tree printing algorithm (Excerpt 1), then the serializations would be “((A,B),(C,D))”, “(((A,D),B),C)”, “(B,(C,(A,D)))”, “((A,(C,B),D))”, “((A,(C,B)),D)”. Thus, we cannot directly compare two trees (such as the original tree vs. the tree returned by DNAML) unless we canonicalize the representations.
f) calculating the SPRs of a tree
d) and e) are non-trivial because our phylogenetic trees are unrooted, whereas BioPerl represents all trees as rooted. Our approach is described in the Appendix.

```
sub printTree ($) {
  my ($node) = @_;
  if ($node->is_leaf ())
    return $node->id ();
  else
    return   '(' . $node->leftChild->id () . ', '
              . $node->rightChild->id () . ')';
}
```

Excerpt 1. Pseudo-code for printing a tree in simplified Newick format.

Scenarios
For all scenarios, we assumed the Jukes-Cantor model for both data generation and phylogenetic inference. With the Jukes-Cantor model the equilibrium base probabilities

\[ \pi_A = \pi_C = \pi_G = \pi_T, \text{ and } P_{ij}(v) = \begin{cases} \frac{1}{4} + \frac{3}{4} e^{-\frac{1}{3}v} & \text{if } i = j \\ \frac{1}{4} - \frac{1}{4} e^{-\frac{1}{3}v} & \text{if } i \neq j \end{cases} \] (e.g., Felsenstein, 2008).

Accuracy of the SPR tests
Define \( B_i \) to be the fully balanced binary tree with \( 2^i \) species. Let \( C_i \) be \( B_i \) but with an additional two species sub-tree, attached to the root via a branch of length two. All branch lengths not otherwise specified are 0.05. \( C_1, C_2, \) and \( C_3 \) are shown in Figure 7.
We used the following test trees to generate data:

- 4-species tree (Figure 5, top) with unit length branches: 100,000 simulations
- $C_2$ (6-species): 10,000 simulations
- $C_2$ but where the longer branch is 2.5: 10,000 simulations
- $C_4$ (18-species): 1000 simulations

The numbers of simulations stated above were used for calculating the average likelihood of obtaining the correct tree by ML, and the SPR\textsubscript{plain} statistics. Since the SPR\textsubscript{opt} statistics are slower to calculate, they are generally averaged over 1000 simulations.

**Speed of the SPR tests**
To provide a rough estimate of the speed of the phylogenetic analyses and SPR tests, we ran them on $C_3$, $C_4$, $C_5$, and $C_6$ for a single simulation. This also provides some anecdotal evidence of the behavior of the SPR test for larger trees.

**Neighborhoods**
We noted in the Introduction that using the SPR neighborhood, rather than all topologies, as the denominator could result in some non-ML topologies receiving a high test score. This phenomenon is less likely to occur in small trees because the SPR neighborhood is a substantial subset of all topologies (with 4 species, the SPR neighborhood is all other topologies; with 5 species, the SPR neighborhood contains 12 of the 14 other topologies); however, it is computationally infeasible to investigate large trees. Thus, we investigated the 6-species topology $C_2$, for which the SPR neighborhood contains 30 of the 104 other topologies.

If the sequences are extremely short, then most topologies will be of roughly equal likelihood and no topology will have a high test score, while if the sequences are extremely long, then the maximum likelihood topology will be unambiguously selected. We therefore chose a sequence
length of 512 (based on the results in Figure 10), so that there might be multiple highly plausible topologies.

For each of the 105 possible 6-species topologies (denoted $T_i \in \{1..105\}$), we optimized the branch lengths, and ran the SPR tests with that topology in the numerator i.e.,

- SPR\text{plain} statistic for $T_i$: $\frac{P(D|T_i,g'_i)}{\sum_{t \in SPR(T_i)} P(D|t,g_t)}$, where $g'_i = \arg\max_g P(D|T_i,g)$

- SPR\text{opt} statistic for $T_i$: $\frac{P(D|T_i,g'_i)}{\sum_{t \in SPR(T_i)} \max_g P(D|t,g)}$

This simulates the situation where a phylogenetic analysis program chooses one of these topologies, even if it is not actually the maximum likelihood topology. We also recorded the average likelihood for each of the topologies ($P(D|T_i,g'_i)$).

**Results**

Note: In all results below, $I(T=T')$ is calculated using DNAML, which does not necessarily (and probably does not, for very large trees) find the true maximum likelihood tree.

**4-species tree with unit length branches**

The likelihood of correctly recovering the correct topology ranges from roughly chance when the sequence length is 1, to almost perfect when there are 1024+ nucleotides (Figures 8 and 9). Curiously, this is non-monotonic: it is harder to recover the tree with sequences of length four sequences than of length two. This is a reliable difference, and is discussed in the Appendix.

Both the SPR\text{plain} and SPR\text{opt} statistics are higher for correct than incorrect topologies, and both statistics approach 1 for the correct topology when the sequences are very long, because the likelihood of the alternative (incorrect) topologies decreases.

However, when the SPR\text{plain} statistic is applied to incorrect topologies, it is a weaker bound. This is because the branch lengths of the topology in the numerator are optimized, but the branch lengths of the topologies in the denominator are not. We would expect that with longer sequences, the SPR\text{plain} statistic would further decrease for incorrect topologies; however, we did not run the simulations because the time required to gather enough data would be excessive (in part due to the longer sequences, but mostly since it becomes extremely rare to find an instance where the ML topology is incorrect).
Figure 8. The likelihood of recovering the correct tree, and SPR\textsubscript{plain} test scores, for various sequence lengths. "SPR\textsubscript{plain} | T=T'" means the average SPR\textsubscript{plain} statistic, conditional on having identified the correct topology. Error bars show one standard deviation.

Figure 9. The likelihood of recovering the correct tree, and SPR\textsubscript{opt} test scores, for various sequence lengths.

C\textsubscript{2} (6-species)
Perhaps the most striking feature of Figure 10 is the non-monotonic likelihood function. In this case, this is likely because of overly short branches in the true tree (Yang, 1995). We share their opinion that “Since there is no hope of recovering the true tree with such short sequences, we can restrict our discussion to relatively large n” (p. 693).

The relations between the test scores are similar to the previous 4-species tree: the statistics are higher for the correct topologies compared to the incorrect topologies; and the SPR\textsubscript{plain} statistics are higher than the corresponding SPR\textsubscript{opt} statistic.
The SPR\textsubscript{opt} statistics are similar for both correct and incorrect topologies for sequence lengths < 256, but they begin to diverge for $n \geq 512$ (i.e., once the likelihood is well-behaved).

Figure 10. The likelihood of recovering the correct tree, and SPR test scores, for various sequence lengths.

When the long branch is 2.5, the results are also similar, though our test sequences are not long enough to reliably recover the correct topology (Figure 11).
Figure 11. The likelihood of recovering the correct tree, and SPR test scores, for various sequence lengths.

C₄ (18-species)

With this topology, the likelihood of obtaining the correct topology appears much more mundane. The only unusual result is that for sequences of length n ∈ {128, 256, 512}, it appears that the SPR test scores are higher than incorrect than correct topologies (Figure 12). It may be due to random sampling, though a genuine difference would not be impossible (as illustrated in the “Neighborhoods” section).
Figure 12. The likelihood of recovering the correct tree, and SPR test scores, for various sequence lengths.

**Larger trees**

Note: the scores and timings in this section are noisy because they are based on a single simulation. Furthermore, computational resources (e.g., CPU, disk and network bandwidth) were shared between multiple users. All timings are based on a single core of an AMD Opteron Processor 8384 (2.7 GHz).

For each topology, the SPR statistics tend to be higher when the sequences are longer (Figure 13). Additionally, for a fixed sequence length, the SPR scores are higher for the smaller topologies (i.e., larger topologies are harder to resolve).
Figure 13. The SPR test scores, for various sequence lengths.

The run-times for the SPR tests are shown in Figure 14. For any fixed topology, there is a large constant component to the run-time (for example, for $C_6$ with a sequence length of 1, it takes over 20 minutes), but this is an artifact of the current test implementation\(^2\). Ignoring the shorter sequence lengths, we can see that the asymptotic complexity of the SPR tests is $O(m^2n)$, where $n$ is the sequence length and $m$ is the number of species. This is expected: there are $O(m^2)$ topologies obtained by SPRs, and evaluating each topology takes $O(mn)$ time.

\(^2\) There is a large number of SPR trees generated, and the BioPerl Newick parser has a high constant factor in its run-time (~17 minutes for $C_6$). This means that DNAML is able to calculate the likelihood for all these trees (which includes its own parsing) in only a few minutes, implying that parsing need not be expensive.
Furthermore, DNAML’s branch optimization procedure is $O(mn)$ per tree, as the branches are each optimized separately (Felsenstein, 2008).

Figure 14. The run-time of the SPR tests, for various trees and sequence lengths.

The SPR_{plain} run-times compare favorably with the DNAML run-times (Figure 15), while the SPR_{opt} run-times, although higher, are still reasonable.
Figure 15. The run-time of DNAML, for various trees and sequence lengths.

**Neighborhoods**

Figure 16 plots the average likelihood of the data (always generated from topology $C_2$) for each of the 105 topologies. The topologies have been grouped based on nearest-neighbor interchange (NNI) distance from the correct topology: 0 NNIs means the correct topology ($C_2$), 2 NNIs means a Type (iii) SPR away from $C_2$, and 3 NNIs means they are a Type (iii) SPR away from an NNI neighbor of $C_2$. The graph shows that all topologies that are far removed (> 1 SPR) from the correct topology have very low likelihood i.e., the SPR neighborhood (indeed, even the NNI neighborhood) contains most of the likelihood, making it a good approximation of the total likelihood across all topologies.

Figure 16. Likelihoods based on NNI distance.
The average SPR statistics of the data, for each of the 105 topologies, is shown in Figure 17. This illustrates the weaker bound of the SPR$_{\text{plain}}$ statistic: $C_2$ is assigned a likelihood of 0.6, while four (incorrect) topologies are assigned a score > 0.3; in total, the SPR$_{\text{plain}}$ test assigns a total likelihood of 3.15 across all the topologies. In contrast, the SPR$_{\text{opt}}$ statistic has a total likelihood of 1.02.

The statistics show, reassuringly, that none of the 104 incorrect topologies have a higher score than $C_2$; furthermore, the likelihoods of the topologies that are more than an NNI away are very small.

![Figure 17. SPR statistics based on NNI distance. For clarity, the SPR-opt data points are shifted horizontally.](image)

Note that the test scores for $C_2$ (distance 0, Figure 17) cannot be directly compared to those in Figure 10. In Figure 10, “SPR$_{\text{plain}}$ | T=T” shows the likelihood score of $C_2$, when $C_2$ was the maximum likelihood tree. In contrast, Figure 17 shows the likelihood score of $C_2$, whereby we always select $C_2$, even if it is not of maximum likelihood.

**Discussion**

Our simulations show that the SPR tests – especially SPR$_{\text{opt}}$ – are informative, generally providing low scores for incorrect topologies. Importantly, this is true for moderate length sequences, where a statistical test is useful (whereas for extremely short or long sequences, the likelihood of obtaining the correct or incorrect topology respectively is all but assured). Our results also indicate that both SPR tests are tractable, and that the SPR neighborhood is a good source of alternative topologies.

The rigorousness of our experiments has been limited by the large number of possible phylogenetic trees. Firstly, we have only tested specific examples of n-species topologies, with particular branch lengths. Secondly, our “Neighborhoods” test was performed on a 6-species topology (again, only a specific topology): it is likely that our tree is too small to exhibit multiple meaningful maxima.
We were also unable to run multiple simulations for the “Larger trees” studies. However, this is not a concern for potential test users, since they would only need to run the test once per dataset.

It would also be instructive to apply the test on real datasets (for which the true topology is known by other means): Stamatakis (2005) has cautioned that simulated alignment data has an unrealistically strong phylogenetic signal, due to the absence of gaps or sequencing errors. Analyses of real world data would motivate the use of more complicated models of DNA evolution.

There are many other, newer phylogenetic analysis programs than DNAML, though not all of them will readily analyze data with our assumptions (Jukes-Cantor model etc.). Such programs would be better at identifying the “true” maximum likelihood tree, and/or be faster. More importantly, since these programs often efficiently compute SPRs, we can use their techniques to improve our statistical tests. Improving the speed of the test would help not only users, but also make it practical for us to perform more simulations to assess the accuracy of the test.

In Figure 16, we observed that NNIs contained most of the likelihood for the 6-species tree. Unfortunately, using NNIs instead of SPRs has only minor time savings for small trees, and would provide a much poorer bound for larger trees. However, it is possible to obtain most of the benefits of SPRs, with similar cost to NNIs. PhyML switched from NNIs to SPRs, but used a distance-based heuristic to discard poor SPRs, and estimates SPRs locally (Guindon, 2010). Similarly, FastTree2 only looks at O(n) of the O(n^2) best SPRs (Price, Dehal, & Arkin, 2010), and RAxML uses “lazy subtree rearrangements”, whereby only the three branches adjacent to the resplicing point are optimized (Stamatakis & Alachiotis, 2010).

Although we have not implemented these SPR optimizations, our results already provide bounds on the speed and accuracy of related tests. For example, using NNIs as the neighborhood without optimizing the branch lengths (ill-advised as this may be), will be faster than SPR_{plain}, but less accurate, while SPR_{plain} with lazy subtree rearrangements will have speed and accuracy in between that of SPR_{plain} and SPR_{opt}.

Guindon (2010) observed that the aLRT and bootstrap supports should both be used, since they each detect different problems with trees (e.g., aLRT does not consider alternative topologies that are far removed, while the bootstrap can give high scores when there are very short branches, even though these are not supported by substitutions). Similarly, our tests cannot provide a definitive statement of the correctness of a topology, but might often be able to cast doubt on an incorrect topology. Thus, we envision the SPRs not as a replacement, but a complement to existing phylogenetic tree tests; when used in concert with other tests, the SPR tests are an effective and affordable method for testing phylogenies.
References

Allen, B., & Steel, M. (2001). Subtree Transfer Operations and their Induced Metrics on Evolutionary Trees. *Annals of Combinatorics*, 5, 1-15.

Aris-Brosou, S. (2003). How Bayes tests of molecular phylogenies compare with frequentist approaches. *Bioinformatics*, 19, 618-624.

Barton, N., Briggs, D., Eisen, J., Goldstein, D., & Patel, N. (2007). *Phylogenetic Reconstruction*. Retrieved from http://evolution-textbook.org/content/free/contents/ch27.html

Congdon, C. (2001). Gaphyl: A Genetic Algorithms Approach to Cladistics. *Lecture Notes in Computer Science: Principles of Data Mining and Knowledge Discovery*, 2168, 67-78.

Felsenstein, J. (1978). The Number of Evolutionary Trees. *Systematic Zoology*, 27, 27-33.

Felsenstein, J. (1981). Evolutionary trees from DNA sequences: a maximum likelihood approach. *Journal of molecular evolution*, 17, 368-376.

Felsenstein, J. (2005). PHYLIP (Phylogeny Inference Package) version 3.6. Washington: Department of Genome Sciences, University of Washington.

Felsenstein, J. (2008). *Dnadist - Program to compute distance matrix from nucleotide sequences*. Retrieved from http://evolution.genetics.washington.edu/phylip/doc/dnadist.html

Felsenstein, J. (2008). *Restml - Restriction sites Maximum likelihood program*. Retrieved from http://evolution.genetics.washington.edu/phylip/doc/restml.html

Foster, P. (2001, July). *Likelihood in Molecular Phylogenetics*. Retrieved April 1, 2013, from http://bioinf.ncl.ac.uk/molsys/data/like.pdf

Goldman, N., Anderson, J., & Rodrigo, A. (2000). Likelihood-Based Tests of Topologies in Phylogenetics. *Systematic Biology*, 49, 652-670.

Guindon, S. D. (2010). New Algorithms and Methods to Estimate Maximum-Likelihood Phylogenies: Assessing the Performance of PhyML 3.0. *Systematic Biology*, 59, 307-321.

Hordijk, W., & Gascuel, O. (2005). Improving the efficiency of SPR moves in phylogenetic tree search methods based on maximum likelihood. *Bioinformatics*, 21, 4338-4347.

Huelsenbeck, J., & Imennov, N. (2002). Geographic Origin of Human Mitochondrial DNA: Accommodating Phylogenetic Uncertainty and Model Comparison. *Systematic Biology*, 51, 155-165.

Kim, J. (1998). Large-Scale Phylogenies and Measuring the Performance of Phylogenetic Estimators. *Systematic Biology*, 47, 43-60.

Price, M. (n.d.). *FastTree 2.1: Approximately-Maximum-Likelihood Trees for Large Alignments*. Retrieved from http://meta.microbesonline.org/fasttree/#Support
Price, M., Dehal, P., & Arkin, A. (2010). FastTree 2 – Approximately Maximum-Likelihood Trees for Large Alignments. *PLoS One, 5*.

Rambaut, A., & Grass, N. (1997). Seq-Gen: an application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. *Bioinformatics, 13*, 235-238.

Schmidt, H., Strimmer, K., Vingron, M., & von Haeseler, A. (2002). TREE-PUZZLE: maximum likelihood phylogenetic analysis using quartets and parallel computing. *Bioinformatics, 18*, 502-504.

Stajich, J., D., B., Boulez, J., Brenner, S., Chervitz, S., Dagdigian, C., et al. (2002). The Bioperl Toolkit: Perl Modules for the Life Sciences. *Genome Research, 12*, 1611-1618.

Stamatakis, A. (2005). An efficient program for phylogenetic inference using simulated annealing. *Parallel and Distributed Processing Symposium. IEEE*.

Stamatakis, A., & Alachiotis, N. (2010). Time and memory efficient likelihood-based tree searches on phylogenomic alignments with missing data. *Bioinformatics, 26*, 132-139.

Team, R. C. (2012). R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing.

Velasco, J. (2008). The prior probabilities of phylogenetic trees. *Biology and Philosophy, 23*, 455-473.

Yang, Z. (1995). Evolution of the ML Estimate of the Evolutionary Tree Topologies. *Molecular Evolution, 40*, 689-697.

Yang, Z., Goldman, N., & Friday, A. (1995). Maximum Likelihood Trees from DNA Sequences: A Peculiar Statistical Estimation Problem. *Systematic Biology, 44*, 384-399.

Zhu, S., Degnan, J., & Steel, M. (2011, October 5). *Clades and clans probability in Yule trees*. Retrieved from http://www.math.canterbury.ac.nz/~s.zhu/other_files/Joe_clades.pdf
Appendices

Non-monotonic likelihood for the 4-species topology
In Figure 9 we observed that the probability of DNAML recovering the correct tree was higher with 2 n.t. sequences than 4 n.t. sequences (39.3% vs. 38.3%). While this is a small effect size, it is reliable (results are averaged over 100,000 trials).

Oscillation in likelihood of recovering the correct tree
We repeated the analysis with sequence lengths from 1 to 31 (Figure 18). This showed oscillation, with sequences of \(4k\) nucleotides being harder to recover than sequences of \(4k - 1\) nucleotides. These trends persisted even when we used OpenSSL’s (v1.01) cryptographic random number generator for the sequence generation (seq-gen) and phylogenetic program (DNAML).

For further validation, we used TREE-PUZZLE (Schmidt, Strimmer, Vingron, & von Haeseler, 2002), a phylogenetic analysis program which was written independently of DNAML. While the results are noisier (and do not match our predictions for 20 or 24 n.t. sequences), they also hint at oscillation.

![Figure 18. The likelihood of DNAML or TREE-PUZZLE recovering the correct topology, for varying sequence lengths. TREE-PUZZLE is plotted on the secondary axis.](image)

This is surprising at first glance because it implies that discarding part of the data will lead to better results.

However, Kim (1998) noted that consistency does not imply monotonicity. They provided an elegant example: estimating whether a biased coin has \(p = \frac{1}{4}\) or \(\frac{3}{4}\) heads, by determining whether the observed fraction of heads is \(\leq \left(\frac{1}{4} + \varepsilon\right)\). If we toss a coin \((p = \frac{1}{4}) 4k\) times \((k \in \mathbb{Z})\), then this estimator is correct when the number of heads is \(\leq k\); if we toss a coin \(4k + 1\) times, the set of outcomes where the estimator is correct is still the same, but the extra coin toss has non-zero probability of changing a correct estimate (if there were exactly \(k\) heads) into an incorrect estimate. Hence, the overall probability of a correct estimate with \(4k + 1\) coin tosses is less than with \(4k\) coin tosses. Kim (1998) further remarked that it is unknown whether
consistent phylogenetic estimators have monotonicity, but noted that “the situation is similar to the simple example (estimation of a discrete parameter, the topology, using an interval” (pp. 45-56).

On the other hand, Yang (1995) speculated that the likelihood of the correct tree is monotonically increasing with \( n \), with two exceptions: for short sequences, the star tree could have spuriously high likelihood if there was a high proportion of constant site patterns in the original data; and when assuming the wrong model.

As our analyses below will show, we have identified a case where a consistent phylogenetic estimator is non-monotonic, without relying on constant site patterns or incorrect models.

**Oscillation in indistinguishability**

For each dataset \( D \), we evaluated the likelihood of the three possible 4-species topologies under optimized branch lengths (i.e., \( \max_y P(D|t, g) \)), and identified the set of topologies, \( \hat{T} \), with highest likelihood. This is related to the likelihood of recovering the correct tree, via:

\[
P(\hat{T} = t_1) = P(\hat{T} = \{t_1\}) + \frac{1}{2}(P(\hat{T} = \{t_1, t_2\}) + P(\hat{T} = \{t_1, t_3\})) + \frac{1}{3}P(\hat{T} = \{t_1, t_2, t_3\})
\]

As seen in Figure 19, when the sequence length is a multiple of 4, the likelihood of the correct topology uniquely being maximum likelihood drops, while the probability of all topologies being equally likely peaks.

![Figure 19](image_url)

**Figure 19.** The likelihood that the ML topologies are \{t_1\} (the correct topology) or \{t_1, t_2, t_3\} (all topologies are equally likely), for varying sequence lengths. The likelihoods that the ML topologies are \{t_2\}, \{t_3\}, \{t_1, t_2\}, \{t_1, t_3\}, or \{t_2, t_3\} are not shown.

**Infinite length branch leads to indistinguishability**

Consider all four-species topologies where one of the peripheral branches has infinite length; without loss of generality, let this be the branch AP (Figure 20).
Figure 20. The three possible four-species topologies, with the stipulation that the ‘A’ branch has infinite length.

All three topologies are equivalent to Figure 21, and hence with each other. Thus, when the branch lengths are optimized for each topology, they are all of equally high likelihood.

Figure 21. Another representation of the four-species topology where the ‘A’ branch has infinite length.

Oscillation of likelihood of having an infinite length branch
Firstly, some weak anecdotal evidence is that the average total branch length of the maximum likelihood tree peaks at multiples of 4 (Figure 22).

Figure 22. The average total branch length of the maximum likelihood tree, for varying sequence lengths.

Next, we consider the optimal branch lengths. Suppose the sequence length is \( m \), and two nodes have \( k \) nucleotides in common (Figure 23).
Under the Jukes-Cantor model, the likelihood for a branch length $x$ is:

$$f(x) = \left(\frac{1}{4} - \frac{1}{4} e^{-\frac{4x}{3}}\right)^{m-k} \left(\frac{1}{4} + \frac{3}{4} e^{-\frac{4x}{3}}\right)^k$$

We used R (R Core Team, 2012) to solve this numerically for various values of $m$ and $k$ (Table 3). We can see that, as expected, if $k = m$ (i.e., both nodes have all nucleotides in common), the optimal branch length is 0, and if $k = 0$ (both nodes have no nucleotides in common), the optimal branch length is infinity$^3$.

However, we also know that if the branch length is infinite, then the expected number of identical nucleotides is $\frac{1}{4}$, hence it is not surprising that when $k \leq \frac{m}{4}$, the optimal branch length is also infinity.

Since nucleotides are discrete — we cannot have, for example, 2.25 nucleotides in common$^4$ – this is equivalent to Kim’s (1998) coin-tossing example!

---

$^3$ The numerical optimization was run over the range [0, 10].

$^4$ The Jukes-Cantor model is defined only for {A,C,G,T}; it does not consider ambiguity characters (e.g., purines vs. pyrimidines).
Table 3. The optimum branch length when sequences are of length n, and have k nucleotides in common.

| n | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|----|----|----|
| 1 | 10.00 | 0.00 | | | | | | | | | | | |
| 2 | 10.00 | 0.82 | 0.00 | | | | | | | | | | |
| 3 | 10.00 | 1.65 | 0.44 | 0.00 | | | | | | | | | |
| 4 | 10.00 | 10.00 | 0.82 | 0.30 | 0.00 | | | | | | | | |
| 5 | 10.00 | 10.00 | 1.21 | 0.57 | 0.23 | 0.00 | | | | | | | |
| 6 | 10.00 | 10.00 | 1.65 | 0.82 | 0.44 | 0.19 | 0.00 | | | | | | |
| 7 | 10.00 | 10.00 | 2.28 | 1.08 | 0.64 | 0.36 | 0.16 | 0.00 | | | | | |
| 8 | 10.00 | 10.00 | 10.00 | 1.34 | 0.82 | 0.52 | 0.30 | 0.14 | 0.00 | | | | |
| 9 | 10.00 | 10.00 | 10.00 | 1.65 | 1.01 | 0.67 | 0.44 | 0.26 | 0.12 | 0.00 | | | |
| 10 | 10.00 | 10.00 | 10.00 | 2.03 | 1.21 | 0.82 | 0.57 | 0.38 | 0.23 | 0.11 | 0.00 | | |
| 11 | 10.00 | 10.00 | 10.00 | 2.62 | 1.42 | 0.97 | 0.70 | 0.50 | 0.34 | 0.21 | 0.10 | 0.00 | |
| 12 | 10.00 | 10.00 | 10.00 | 10.00 | 1.65 | 1.13 | 0.82 | 0.61 | 0.44 | 0.30 | 0.19 | 0.09 | 0.00 |
| 13 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 1.92 | 1.29 | 0.95 | 0.72 | 0.54 | 0.40 | 0.28 | 0.17 | 0.08 |
| 14 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 2.28 | 1.46 | 1.08 | 0.82 | 0.64 | 0.48 | 0.36 | 0.25 | 0.16 |
| 15 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 |

Finally, we can provide an analytical proof:

Multiply $f(x)$ by $4^k$ and substitute $v = e^{-\frac{4x}{3}}$, to obtain:

$$g(v) = (1 - v)^{m-k} (1 + 3v)^k$$

With routine differentiation:

$$g'(v) = \left[-(m - k)(1 - v)^{m-k-1}(1 + 3v)^k\right] + \left[(3k)(1 - v)^{m-k}(1 + 3v)^{k-1}\right]$$

$$= -(m - k)(1 + 3v)(1 - v)^{m-k-1}(1 + 3v)^{k-1}$$

$$+ (3k)(1 - v)(1 - v)^{m-k-1}(1 + 3v)^{k-1}$$

$$= (1 - v)^{m-k-1}(1 + 3v)^{k-1}[-(m - k)(1 + 3v) + (3k)(1 - v)]$$

$$= (1 - v)^{m-k-1}(1 + 3v)^{k-1}[4k - m - 3mv]$$

Since $v = e^{-\frac{4x}{3}}$, $v \in [0,1]$, hence $g'(v) = 0$ iff:

1. $v = 1$; or,
2. $4k - m - 3mv = 0$
Case 1 means \( x = 0 \) (branch length of zero). We know that this is optimal when the two sequences are identical \((k = m)\). This is trivial to prove: \( g(\nu) = (1 + 3\nu)^k \) is obviously maximal when \( \nu = 1 \). It is also covered by Case 2.

Case 2 means \( \nu = \frac{4k}{3m} - \frac{1}{3} \). As \( k \to m, \nu \to 1 \) (i.e., \( x \to 0 \): when the sequences become more similar, the optimal branch length decreases, which matches our intuition). When \( \nu = 0 \) \((x = \infty)\), \( k = \frac{m}{4} \) i.e., when the fraction of matching nucleotides is 25%, the optimal branch length is infinity. We also know that as the sequences become less similar, the optimal branch length should be non-decreasing, hence it is also the case that when \( k \leq \frac{m}{4} \), the optimal branch length is infinity.

The formula for the optimum value of \( \nu \) matches the values found numerically (Table 3).

Optimizing each branch length separately might not, in general, give the optimum set of branch lengths.\(^5\) However, we only require a weaker claim: if A and X have \( \leq 25\% \) nucleotides in common, then it is usually globally optimal for the branch AX to be \( \infty \). Since A is closer to X than B, C or D (see Figure 20), the number of matching nucleotides between A and B, C or D (individually) is, in expectation, less than or equal to the number of matching nucleotides between A and X. Thus, if the optimum branch length between A and X (computed locally) is \( \infty \), then the optimum distance between A and the other leaves is likely to be \( \infty \) as well.

\(^5\) Although, as earlier noted, DNAML does optimize each branch length separately.
Implementation

Printing the canonical form
We used Congdon’s (2001) canonical form: the first species is a child of the root, and every left subtree is no larger than its sibling. Printing the canonical form can be performed with explicit rotations (Figure 24):

![Figure 24. One way to print the canonical form is to 1) Parse a Newick tree into memory (using BioPerl) 2) Canonicalize the tree using rotations 3) Print the tree in Newick format (e.g., using the pseudo-code in the Method section).](image)

However, rotations are expensive and error-prone, so we instead have a more complicated tree printing function (Excerpt 2), which goes directly from an arbitrary tree in memory, to a canonicalized tree in Newick format (should we require the canonicalized tree in memory, we can later parse it). To canonicalize a tree, we create a node above the first species, then run canonicalizeTree on that node (with a NULL caller). This is similar to flooding on a spanning tree.

```perl
sub canonicalizeTree ($$) {
    my ($node, $caller) = @_; #
    if ($node->is_Leaf ()) {
        return $node->id ();
    } else {
        my %neighbors = ($node->leftChild => 1, $node->rightChild => 1, $node->parent => 1);
        delete $neighbors {$caller};
        foreach my $neighbor (keys %neighbors) {
            push @text, canonicalizeTree ($neighbor, $node);
        }
        @text = sort {length $a <=> length $b} @text;
        return '(' . (join ',', @text) . ')';
    }
}
```

Excerpt 2. Pseudo-code for printing the canonicalized form of a tree in simplified Newick format.
Tree SPRs
Figure 25 illustrates one SPR operation: the subtree to the “lower right” of X has been pruned, and regrafted onto the edge XC. We can perform this by creating a node on the midpoint of XC, and adding Y as a child.

![Figure 25. Pruning the subtree Y using the XY edge, and regrafting it to XC.](image)

Since the tree is unrooted, the subtree to the “left” of X also defines a clade, and thus we can regraft it onto, say, the edge DZ (Figure 26). This is more difficult as X already has a parent (W), hence we cannot add X as a child of DZ. We could, in principle, reroot the Y subtree at the midpoint of DZ, thus allowing the midpoint of DZ to be a child of X:

![Figure 26. Pruning the subtree X using the XY edge, and regrafting it to DZ.](image)

However, as with canonicalization, we wish to avoid rotations, and only need to be able to produce a serialized tree (rather than have a rotated tree in memory). We therefore adapted the canonicalization function, so that when considering the “children” of a node, it would recognize the pruned/regrafted edge: for example, in Figure 26, the children of X become \{W, C, DZ\} instead of \{W, C, Y\}. 

29