Karlsruhe Institute of Technology

Quantitave Analysis of Phylogenetic Quasi-Terraces

Master Thesis of

Paula Breitling

At the Department of Informatics
Institute of Theoretical Computer Science

Reviewer: Prof. Dr. Alexandros Stamatakis
Prof. Dr. Dennis Hofheinz
Advisor: Ben Bettisworth

Time Period: 01st January 2019 – 30th June 2019

Statement of Authorship

I hereby declare that this document has been composed by myself and describes my own work, unless otherwise acknowledged in the text.

Karlsruhe, 30th June, 2019

. . . . . .

(**Paula Breitling**)

Abstract

Terraces in phylogenetic tree space are, among other things, important for the design of tree space search strategies. While the phenomenon of phylogenetic terraces is already known for unlinked branch length models on partitioned phylogenomic data sets, it has not yet been studied if an analogous structure is present under linked and scaled branch length models. To this end, we analyze aspects such as the log-likelihood distributions, likelihood-based significance tests, and nearest neighborhood interchanges on the trees residing on a terrace and compare their distributions among unlinked, linked, and scaled branch length models. Our study will show that there exists a terrace-like structure under linked and scaled branch length models as well. We denote this phenomenon as quasi-terrace. Therefore quasi-terraces should be taken into account in the design of tree search algorithms as well as when reporting results on "the" final tree topology in empirical phylogenetic studies.

Deutsche Zusammenfassung

Terrassen im phylogenetischen Raum der Bäume sind unter anderem wichtig für die Implementierung von Suchstrategien für die Stammbaumrekonstruktion. Während das Phänomen der phylogenetischen Terrassen bereits für unverknüpfte Astlängenmodelle auf partitionierten phylogenomischen Datensätzen bekannt ist, wurde noch nicht untersucht, ob eine analoge Struktur unter verknüpfte und skalierten Astlängenmodellen existiert. Zu diesem Zweck analysieren wir Aspekte wie die Log-Likelihood-Verteilungen, wahrscheinlichkeitsbasierte Signifikanztests und die nächstgelegenen Nachbarschaftsveränderungen an den Bäumen, die sich auf einer Terrasse befinden, und vergleichen deren Verteilungen zwischen unverknüpften, verknüpften und skalierten Astlängenmodellen. Unsere Studie wird zeigen, dass auch bei verknüpften und skalierten Astlängenmodellen eine terrassenartige Struktur existiert. Wir bezeichnen dieses Phänomen als Quasi-Terrasse. Daher sollten Quasi-Terrassen bei der Entwicklung von Baumsuchalgorithmen sowie in der Darstellung der Ergebnisse einer Baumsuche in empirischen phylogenetischen Studien berücksichtigt werden.

List of Figures

List of Tables

Contents

1. Introduction

From the dawn of time scientists have been interested in where we come from and how we are related to each other. Therefore, the inference and analysis of phylogenetic trees is an important field of research. We will give a more detailed motivation in the following section.

1.1. Motivation

Phylogenetic trees are widely used to explain and visualize the evolutionary relationships of species. One approach to reconstructing phylogenies consists in using genome data assembled into a large data matrix that is typically divided into disjoint partitions, often representing genes.

A common problem with these large phylogenomic matrices are missing data. That is, a taxon can have no data present in a specific partition. This can be due to sampling problems, or that this particular partition of the genome is not present in the specific taxon. Often this can arise due to errors in the sequencing process or because some species simply do not have data in this partition (do not have a specific gene) or if the gene has not been sampled yet (Dobrin, Zwickl, & Sanderson, 2018; Sanderson, McMahon, Stamatakis, Zwickl, & Steel, 2015). This type of missing data complicates phylogenetic tree inference under likelihood-based criteria (maximum likelihood (ML) or Bayesian inference). For numerical reasons, the logarithm of the likelihood instead of the likelihood itself is typically calculated.

When distinct tree topologies have the same log-likelihood (henceforth denoted as: LnL) score, which indicates that they are equally 'good', they reside on a so-called terrace in tree space, a phenomenon first described by Sanderson, McMahon, and Steel (2011).

Mathematically terraces in tree space can only occur under unlinked branch length models. Beside this, there exist two alternative approaches to modeling branch lengths in phylogenomic analyses: scaled and linked branch length estimates.

As linked and scaled branch length models are less computational expensive, and also induce a substantially smaller number of free model parameters, they are frequently

used in large scale empirical phylogenomic analyses. To this end, the question arises if a phenomenon, that is analogous to terraces in the unlinked case, can also be observed under linked and scaled branch length models. We term this phenomenon a quasi-terrace.

1.2. Objectives

In this thesis we conduct a thorough study of 14 published empirical phylogenomic data sets to assess (i) if quasi-terraces exist and (ii) what their potential impact on phylogenomic analyses and the design of tree search algorithms is.

The advances in sequencing technologies produces an unprecedented data flood. Therefore speeding up the tree search algorithms is of interest. The function calculating the likelihood takes up to 95% of the execution time in current phylogenetic inference programs. Additionally, these programs have run times on the order of days to weeks. Hence if one can reduce the number of likelihood calculations, the search algorithm speeds up, saving significant amounts of time. The state of the art ML tree search algorithms start with a tree containing all taxa from the data set. This tree can be produced by fast, but inaccurate, methods. Alternatively, random starting trees can be used, although they normally have lower likelihood scores and so might not accelerate the inference process (Elloumi & Zomaya, 2011).

When searching for the best tree, any method must search through tree space. Unfortunately, tree space is poorly understood, and quite large (Felsenstein, 2004), growing exponentially with the number of taxa present on the tree. This makes total enumeration of tree space impractical, and so heuristics must be used. Identifying terraces, which are sections of tree space with identical likelihoods, offer an opportunity to evaluate many trees at once, thus allowing terrace aware methods to skip evaluation of trees present on the terrace. By skipping computation, terrace aware methods are able to search a much larger section of tree space, with little additional cost. Therefore, we want to evaluate, if a terrace-like structure is present under linked and scaled branch length model as well.

Now that we understand the importance of terraces and hence of finding and using quasi-terraces to reduce computational cost, we needed to develop an approach to find quasi-terraces. Therefore, we developed a pipeline, where all analytical steps are performed under all three branch length models (unlinked, linked, and scaled). With these analyses we want to evaluate if quasi-terraces exist and if they do, obtain further insights about them and their neighborhood in tree space. The pipeline steps include preprocessing of the data, LnL score calculation, significance tests, and neighborhood assessment.

Through all these steps we want to confirm our hypothesis about quasi-terraces. Their existence could be used to speed up all algorithms that search tree space, as only one tree on the terrace has to be computed and can then act as a representative for all trees on the terrace. We search in tree space to find the ML tree and the tree space grows exponentially with the number of taxa. Depending on the size of the terrace, the savings in computational expenses could be significant.

1.3. Structure

After we have stated the motivation and objectives of this thesis, we explain the fundamentals of phylogenetic trees in Chapter 2 to provide a basic understanding of the topic. In Section 2.1 we give the general structure of the data, a detailed explanation of the branch length models, as well as a definition of terraces and the related literature. Afterwards we review the Robinson-Foulds distance, significance tests, and nearest-neighbor interchanges. Finally in the background chapter we introduce the programs used for the later analysis. Section 2.2 also includes information about own scripts.

In Chapter 3 we describe the data sets which are used in this thesis (see Section 3.1) as well as the relevant file formats (see Section 3.2). Thereafter, the experimental setting is expounded by us in Section 3.3. This includes a description of the data preprocessing that we performed, followed by an explanation on how to infer ML trees. Afterwards we explain how the enumeration of trees on a terrace work as well as the calculation of LnL scores. The chapter ends with a section about further analyses.

The results of the different analyses are described and visualized in Chapter 4. After a general overview of the results in Section 4.1, we report the outcome of the significance tests (see Section 4.2). At the end of this chapter we provide the results of the nearest-neighbor interchange analysis in Section 4.3.

This thesis ends in Chapter 5, where we shortly review our work and conclude the thesis in Section 5.1. At the end we reveal possible avenues of future work (see Section 5.2).

2. Background and related work

To gain a better understanding of phylogenetic inference, we start this section with important definitions and explanations, related work as well as the programs used for our analyses.

Phylogenetics is the study of the evolutionary relationships and history of species. These relationships are often inferred by modeling the effects of evolution as a random process for a DNA sequence (Felsenstein, 2004).
As it is known that species are linked to each other by common ancestors, a phylogenetic tree can display those connections in form of a tree diagram (Robinson & Foulds, 1981). These phylogenetic trees are the basis for all further definitions and the analyses we conduct in this thesis.

2.1. Definitions and related literature

In this section we explain the general structure of our data, followed by a visualization of branch lengths. Thereafter, we give the definition of terraces in phylogenetic tree space and briefly review important related work. Then, the Robinson-Foulds distance, the significance tests, and the nearest-neighbor interchanges is explained.

2.1.1. General structure

Before starting with any definitions, we need to explain in general how the data we intend to analyze is represented. Figure 2.1 shows a simplified scheme. For each species (taxa) we have the information for the gene as a DNA (deoxyribonucleic acid) or protein sequence, where a single letter denotes one site. The alignment is given by the number of homologous sites (columns) and the number of taxa (rows), often also denoted as species or sequences. Hence, a partition is a subset of alignment sites. These large matrices (also called supermatrix or phylogenomic data set) are structured in rows representing the taxa and columns denoting the set of sites to be analyzed.

	P_1	P_2	P_3	P_4	P_5
S_1	ACGTA	CGTACA	AGTCAG	- - - - -	- - - -
S_2	CGTAA	- - - - - -	CATTGA	- - - - -	GTAG
⋮	⋮	⋮	⋮	⋮	⋮
S_n	- - - - -	- - - - - -	GCTGAG	TTAGC	ATCA

Figure 2.1.: Simplified data representation
$S_1, ..., S_n$*: species;* $P_1, ..., P_5$*: partitions*

2.1.2. Branch length

One important parameter of a phylogenetic tree is its branch lengths (also called edge length), which represent the distance between two nodes in the phylogenetic tree. The options for modeling branch lengths in a partitioned data set are linked, scaled, and unlinked, as shown in Figure 2.2. The branch lengths are estimated for the same underlying tree topology. The length of a branch represents the evolutionary distance (under maximum likelihood: the mean number of expected substitutions per site) between two nodes in the tree. Hence, the shorter the distance, the closer the two species are related, which are represented by the nodes.

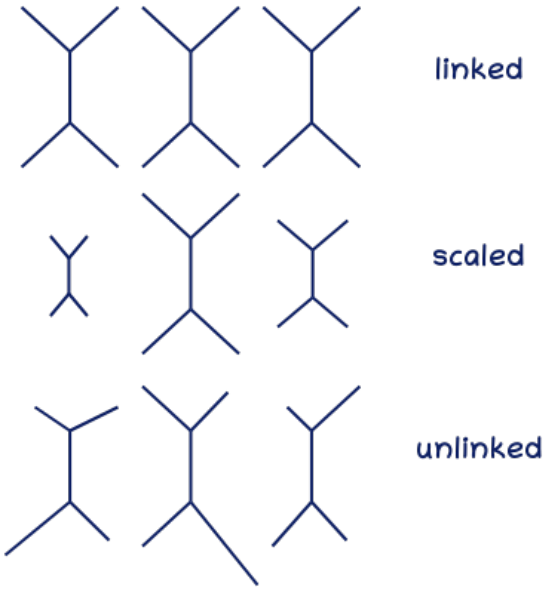


Figure 2.2.: Visualization of the three branch length options: linked, scaled, and unlinked

Mathematically terraces in tree space have only been shown to exist under unlinked branch length models. That is, when a separate independent set of branch lengths is estimated for each partition in the phylogenomic data set. Beside this, there exist two alternative approaches to modeling branch lengths in phylogenomic analyses: scaled and linked branch length estimates. Under a linked branch model a single set of branch lengths over all partitions is being estimated. The same underlying branch lengths are scaled via a single parameter for each partition under the scaled model.

As linked and scaled branch length models are less computationally expensive, and also induce a substantial smaller number of free model parameters, they are frequently used in large scale empirical phylogenomic analyses.

2.1.3. Terraces in phylogenetic trees

The concept of terraces in phylogenetic tree space was first implicitly used by Stamatakis and Alachiotis (2010) for accelerating ML calculations. In 2011 Sanderson et al. described the phenomenon in detail for unlinked branch lengths. Besides the mathematical definition, they define it as follows: ***"All trees on a terrace are distinct from each other, but they are indistinguishable in two important respects: They display the same set of subtrees, and they have the same optimal score"*** (Sanderson et al., 2011).

In the following years different aspects of terraces were analyzed, for example by Chernomor, Minh, and von Haeseler (2015); Chernomor, von Haeseler, and Minh (2016); Dobrin et al. (2018); Sanderson et al. (2015). Software libraries for detecting terraces and counting/enumerating the trees on the terraces were built (for example (Biczok et al., 2018)) as well as existing programs updated to take advantage of the terrace structure or to report the presence of terraces (for example (Biczok et al., 2018; Chernomor et al., 2016; Kozlov, Darriba, Flouri, Morel, & Stamatakis, 2019; Nguyen, Schmidt, von Haeseler, & Minh, 2014)). A description of the used library and programs is given in Section 2.2.

The paper by Sanderson et al. (2015) evaluates the influence of terraces on phylogenetic inference. According to the authors ambiguity is added through terraces as well as complexity, especially when sparse data sets yield the inference challenging. Further, the authors stated that due to the procedure of maximum likelihood approaches, one has to ensure that the neighborhood of the terraces does not contain trees with higher scores (Sanderson et al., 2015). Therefore, their findings are relevant to our analysis as our data sets contain missing data and are also partitioned. Moreover, the distinguishability between a terrace and its immediate neighborhood is a challenge for our analysis as well when using linked and scaled branch length models.

Chernomor et al. (2015) emphasize the importance of checking for terraces before evaluating the trees to save computational time by skipping the trees with identical scores. The total tree score is the addition of all the partition log-likelihoods evaluated with that tree. In focus are the rearrangements (changes of tree topology) of those trees which influence their topologies and hence their score. The authors investigated the following rearrangements/tree moves: nearest neighbor interchange, subtree pruning and regrafting, and tree bisection and reconnection (Chernomor et al., 2015). One of the rearrangements, nearest neighbor interchange, will be described in Section 2.1.6 as it is of further interest to this thesis. Subtree pruning and regrafting is briefly described in Section 5.2.

The second paper by the same authors a year later 2016, stresses again that terraces need to be considered during tree search in order to reduce the computational cost. They developed a data structure, which is aware of terraces, so that under partition models the analyses is efficient. They then implemented it in their program IQ-Tree and experimentally verified the respective time saving (Chernomor et al., 2016). The

paper emphasizes the need for further examination of terraces and related structures to accelerate the analysis of large data sets.

Dobrin et al. (2018) conducted a study of 26 data sets containing missing data. Their main question focused on the correlation between percentage of missing data and terrace size (Dobrin et al., 2018). Their results were the starting point for our analysis. Furthermore we used their collection of data sets, retrieved from twelve different publications, for our study. Detailed information about the sources and composition of the data sets used throughout this thesis can be found in Section 3.1.

2.1.4. Robinson-Foulds distance

One aspect of the later analysis, and generally when examining phylogenetic trees, is to calculate how similar two trees are. One method is the Robinson-Foulds (RF) distance: to calculate the distance between two trees, elementary operations (α and α^{-1}) are used to transform the first tree into the second one. The α operation, called contraction, removes those edges from the first tree, which are not present in the second. Then, the α^{-1} operation, called decontraction, adds the edges only present in the second tree. Hence the transformation of the first into the second tree is the sum of α and α^{-1} operations (Robinson & Foulds, 1981). The minimum number of those tree edit operations required to transform the first into the second tree is the Robinson-Folds distance between the trees.

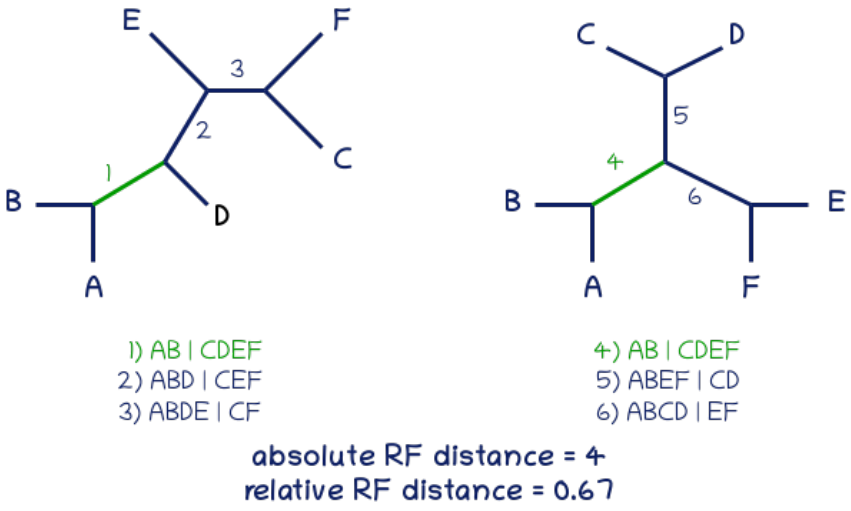


Figure 2.3.: Example of RF distance between two trees

Broadly speaking for both trees all possible bipartitions contained in the two trees are calculated. A bipartition is the decomposition of a taxa present on the tree into two distinct parts at a branch of the tree topology at hand. In Figure 2.3 we listed all possible bipartitions of the two given example trees, labeled with number 1 to 6, and the partition we represented with the pipe symbol (|). The absolute RF distance is calculated as follows: Sum up all bipartitions induced by the two trees (number 1 to 6 in our example), then subtract the duplicated bipartitions (in our example number 1 and 4). As this result (in our example the absolute RF distance is 4) can not be interpreted by itself, the relative RF distance is more commonly used. It puts into proportion the absolute RF distance with the number of inner

branches in the trees. Figure 2.3 gives a short example for the absolute and relative RF distance between two trees.

The results can range between zero and one. When interpreting the results, zero implies that the trees are identical. A number close to zero indicates a large similarity, while a number close to one reveals that the trees are very different. The concept of RF distance can be extended to compare more than a single pair of trees by simply calculating the average over all pairwise RF distances.

2.1.5. Significance tests

Later in the thesis, when we conduct further analyses, one part will be to examine whether the trees on the terrace are significantly different from each other with respect to their likelihood scores. Therefore, we first need to shortly explain log-likelihood based significance tests. For interested readers the relevant papers for each test are stated below. Note that we will only describe those tests used by the program later, even though other tests do exists. The most common tests are listed below:

1. **Bq-RELL test** (Bootstrap proportions using resampling estimated log-likelihood)
   This method performs a random sampling of the estimated log-likelihoods with replacement. Within this procedure we intend to obtain the probability, that the current tree is chosen as the best one out of all others, for each tree (Kishino, Miyata, & Hasegawa, 1990). In case of the program IQ-Tree, the test is performed with 10000 resamplings (Nguyen et al., 2014).

2. **Kishino-Hasagewa (KH) test**
   The KH-test can be performed one sided and weighted. In contrast to the bootstrap resampling, this method estimates the variance in the log-likelihood for the different trees. This leads to reduced computational expenses (Kishino & Hasegawa, 1989).

3. **Shimodaira-Hasagawa (SH) test**
   This test is a modification of the KH-test. As the KH-test was originally designed to compare only two trees, the KH-test is frequently used for the comparison of many trees. In this configuration an incorrect tree (i.e. not the optimal one) is often chosen due to overconfidence, which results from the overlooked the sampling error while selecting the topology. The SH-tests solves the problem, by automatically correcting for this bias (Shimodaira & Hasegawa, 1999).

4. **Approximately unbiased (AU) test**
   This test overcomes the selection bias in the KH-test, but is not as conservative as the SH-test. A newly developed multiscale bootstrap technique is used for selecting the maximum likelihood tree (Shimodaira, 2002). The developers of IQ-Tree therefore recommend to replace the both, the KH- and the SH-test with the AU test.

5. **Expected likelihood weight (ELW) test**
   Trees and their information such as substitution model are assumed to be correct by many tests. Without verifying this information, other tests can produce conflicting results. The ELW test in comparison uses expected likelihood weights when inferring the confidence of a tree (Strimmer & Rambaut, 2002).

At the end it should be stated, that the bq-RELL-, KH- and SH-test extend the ideas from Felsenstein (1981), whereas the theory of Efron, Halloran, and Holmes (1996) is the basis for the AU-test.
The output can be diveded into two different groups: p-values, which KH-, SH- and AU test return, and weights, which the bq-RELL and the ELW test emit. The weights of all tested trees sum up to one.

2.1.6. Nearest-neighbor interchange

Nearest-neighbor interchange (NNI) is a tree rearrangement operation which can be used in heuristic tree searches. Via small rearrangements of the branches of an existing tree one intends to obtain a better tree with respect to its score. NNI is applied to an inner branch of the tree and exchanges two neighbors adjacent to that branch (Felsenstein, 2004). In Figure 2.4 a short example visualizes the method. In the later analysis NNI trees will be used to obtain more insights about the trees surrounding a terrace. To be more precise, for each tree on the terrace we will calculate all possible NNI trees, which then is our NNI neighborhood.

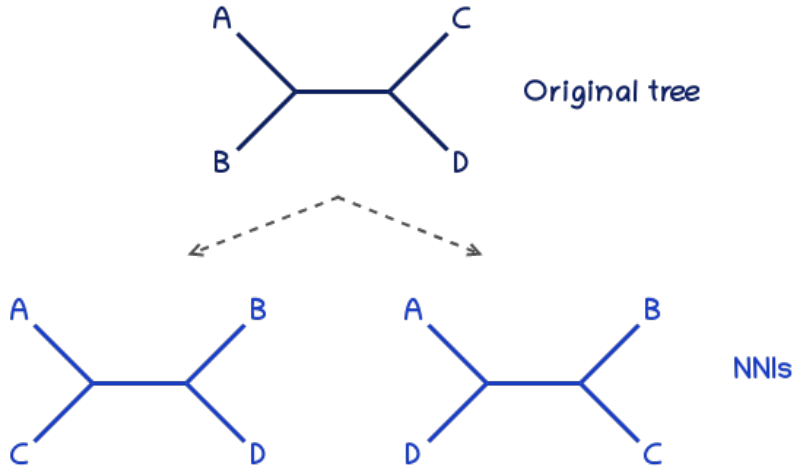


Figure 2.4.: Example of NNI for a tree consisting of four subtrees (A-D)

2.2. Programs used for analysis

There are various programs to conduct different phylogenetic analyses. For our analyses we used RAxML-NG (Randomized Axelerated Maximum Likelihood-Next Generation) by Kozlov et al. (2019) for several parts of the pipeline. For our purpose we only used the following RAxML-NG functions: inferring ML trees, scoring trees on a terrace as well as calculating the RF distance.

Terraphast I (Biczok et al., 2018) was used to count and enumerate all trees on a terrace. It outputs the number of trees on the terrace and additionally enumerate all trees in Newick format.

To perform significance tests and validate some results we used IQ-Tree (Chernomor et al., 2016; Nguyen et al., 2014). IQ-Tree is the successor of *IQ*PNNI and *TREE*-PUZZLE. It has a similar range of functionality as RAxML-NG. Additionally it also offers some functions (e.g., significance tests), which are not yet available in RAxML-NG.

In addition we wrote some Python scripts to automate our analysis process. With one of the scrips we could convert the NEXUS files into the formats needed for further analysis. To receive files containing a comprehensive taxon, which will be explained in Section 3.3.1, we build another Python script. The next script can create the NNI trees of a data set.

3. Experimental setting

Now that we have a general understanding of the topic and related definitions from the previous chapter, we now will describe our experimental setting. We first present our data sets and then the existing file formats used in our analyses. Following this we will specify each step of our pipeline, from the beginning until further analyses, in an own section.

3.1. Data sets

The data sets used for our analyses are from two papers which have already addressed different aspects of terraces in tree space (Chernomor et al., 2016; Dobrin et al., 2018).

Dobrin et al. (2018) assembled data sets from numerous studies (Burleigh, Kimball, & Braun, 2015; Meredith et al., 2011; Miadlikowska et al., 2014; Misof et al., 2014; Rabosky, Donnellan, Grundler, & Lovette, 2014; Shi & Rabosky, 2015; Soltis et al., 2013; Springer et al., 2012; Tolley, Townsend, & Vences, 2013; Wickett et al., 2014; Yang et al., 2015; Zanne et al., 2014) to assess the prevalence of terraces in published empirical phylogenomic studies. Out of the 26 data sets, 23 contain DNA data and 3 contain protein data. Dobrin et al. (2018) found that terrace sizes vary between a single tree and 1.30×10^{388} trees for the data sets they studied.

Three of the data sets in Chernomor et al. (2016) are protein alignments, while the others are DNA alignments. The authors also collected published empirical data sets from a plethora of studies (Bouchenak-Khelladi et al., 2008; Dell'Ampio et al., 2013; Fabre, Rodrigues, & Douzery, 2009; Hinchliff & Roalson, 2012; Nyakatura & Bininda-Emonds, 2012; Pyron et al., 2011; Springer et al., 2012; Stamatakis & Alachiotis, 2010; Van Der Linde, Houle, Spicer, & Steppan, 2010). Some data sets from Chernomor et al. (2016) are excessively large for our analysis, as their terrace sizes exceed 850,000 trees. Hence, they exceed the 24 hour time limit job run time on the cluster that was at our disposal.

In total, we analyzed 38 phylogenomic data sets. For 14 out of those 38 we were able to complete all analysis steps of our pipeline (see Section 3.3). Of those 14 fully

analyzed data sets, 11 contain DNA and 3 contain protein data. Table 3.1 provides an overview of the data sets. For the data sets from Dobrin et al. (2018) we got the species, as they are written directly into the paper. Regarding the data sets from Chernomor et al. (2016) we could only detect the data type but not the species. Therefore we can state, that from the known data sets 8 are plants, one is a lizard and one is a chameleon.

Table 3.1.: Overview of data sets

Data set	Data type	#Taxa	#Partitions	#Sites	Reference
Dobrin et al. (2018)					
Asplenium*	DNA	133	3	4,782	(Zanne et al., 2014)
Eucalyptus*	DNA	136	4	6,205	(Zanne et al., 2014)
Euphorbia*	DNA	131	6	9,154	(Zanne et al., 2014)
Iris*	DNA	137	4	5,815	(Zanne et al., 2014)
Primula*	DNA	185	5	7,321	(Zanne et al., 2014)
Rabosky.scincids	DNA	213	6	5,373	(Rabosky et al., 2014)
Ranunculus*	DNA	170	6	10,799	(Zanne et al., 2014)
Rhododendron*	DNA	117	5	7,321	(Zanne et al., 2014)
Szygium*	DNA	106	4	5,815	(Zanne et al., 2014)
Tolley.chameleons	DNA	202	6	5,054	(Tolley et al., 2013)
Chernomor et al. (2016)					
d128_34	DNA	128	34	29,198	(Stamatakis & Alachiotis, 2010)
d69_31	PROTEIN	69	31	8,546	(Dell'Ampio et al., 2013)
d70_35	PROTEIN	70	35	11,789	(Dell'Ampio et al., 2013)
d72_51	PROTEIN	72	51	12,548	(Dell'Ampio et al., 2013)

**: Subsampled data sets. Data sets were subsampled until they contained a comprehensive taxon.*
See Section 3.3.1 for details

3.2. File formats

Before we start describing our experimental setup, we initially describe the basic file formats we used in our analyses.

3.2.1. NEXUS

The NEXUS alignment file format is commonly used in bioinformatics. We will describe the parts which are used in our pipeline. A NEXUS file starts with general information about the dimensions of the subsequent alignment: the number of taxa and the number of sites are listed. The next line contains the data type, which is either DNA or PROTEIN, and the gap symbol, used to represent missing data. Afterwards the taxa are listed with their names followed by their sequence. At the end of the file one can find the partition information. We show an example in Figure 3.1

3.2.2. PHYLIP

Another alignment file format is PHYLIP. In Figure 3.2 we show an example. It is structured similar to the NEXUS format, but contains less information. The first line only holds two numbers, without any label: number of taxa and number of sites. Thereafter the taxon names and their sequences are listed.

```
#NEXUS
Begin data;
    Dimensions ntax=117 nchar=12632;
    Format datatype=dna gap=-;
    Matrix
Rhododendron_hippophaeoides ATGCATGTGTAAGTATGAACT-AATTCAGACTGTGAAACTGCGAAT-
    .
    .
Rhododendron_yedoense   ------------------------------------------------
;
End;
BEGIN SETS;
    CHARSET 18S = 1-1833;
        .
        .
    CHARSET trnLtrnF = 11463-12632;
END;
```

Figure 3.1.: Example NEXUS file (shortened Rhododendron data set)

```
117 12632
Rhododendron_hippophaeoides    ATGCATGTGTAAGTATGAACT-AATTCAGACTGTGAAACTGCGAAT-
    .
    .
Rhododendron_yedoense          ------------------------------------------------
```

Figure 3.2.: Example PHYLIP file (shortened Rhododendron data set)

3.2.3. NEWICK

Another file format is NEWICK. In contrast to the two formats we described above, a NEWICK file specifies a tree structure. Commas separate the nodes of the tree, in our case the species. The tree structure is given through brackets, where each level in the tree is encapsulated by brackets: the further down the node, the more brackets are around it. Figure 3.3 shows an example.

```
(,,(((((,((,Rhododendron_hippophaeoides),)),),),),(,((,(,((,(,(,(((,(,(((,Rhododendron_yedoense),),))),),)))),)))),)))));
```

Figure 3.3.: Example NEWICK file (shortened Rhododendron data set)

3.3. Experimental setup

For the quantitative analysis we first designed a data preparation and analysis pipeline, so that every data set is prepared and analyzed in exactly the same way to obtain comparable results. For readability we introduce several acronyms: linked branch model (LB), unlinked branch model (UB), and scaled branch model (SB) as well as unlinked branch model for the tree search and linked or scaled model in the LnL calculation (UB-LB respective UB-SB).

We will first outline the basic steps of our pipeline and subsequently will discuss each of them in an own section in detail. After each section we state the main command, where we use `[name]` as replacement for the data set name. For readability we include only the UB model, except for one analysis, which was only performed under LB model. The whole code of the pipeline can be found in the appendix in Figures A.1 and A.2. We conduct the following steps for each data set:

1. Data preprocessing: Create a PHYLIP formatted alignment, a binary presence/absence matrix (indicating which species has data for which partition), and a partition file from the original NEXUS input files
2. Conduct ML tree searches under LB, SB, and UB branch length models with RAxML-NG
3. Calculate if best-scoring ML tree for LB, SB, and UB from these tree searches resides on a terrace and enumerate trees on that terrace with Terraphast I
4. Calculate the LnL scores for all trees on the respective terrace with RAxML-NG under LB, SB, and UB models as well as for UB-LB and UB-SB
5. Further analyses:
    a) Calculate RF-distance between the best ML trees under LB, SB, and UB models from ML searches with RAxML-NG
    b) Significance tests of LnL scores under the LB model using IQ-Tree
    c) LnL calculation for the set of a NNI trees generated from the trees *on* the terrace under UB, UB-LB, and UB-SB
    d) Significance tests of NNI tree set and trees on the terrace under UB, UB-LB, and UB-SB (and additionally for a random Yule–Harding tree (Harding, 1971))

As terraces only occur under UB, we applied all of the above analysis steps using UB, to obtain a reference for comparison with the respective tree scores under LB and SB. We used RAxML-NG (Kozlov et al., 2019) for scoring the trees on a terrace, Terraphast I (Biczok et al., 2018) to enumerate all trees on a terrace, and IQ-Tree (Chernomor et al., 2016; Nguyen et al., 2014) to conduct statistical significance tests. We visualized the whole process in Figure 3.4 at the end of this chapter. In this representation we excluded the further analyses to improve readability.

3.3.1. Data preprocessing

As we had a collection of NEXUS data files, but required other formats for analyses we initially transformed our data accordingly. To analyze terraces we required a binary presence/absence matrix, which describes for which species we have data in which partitions.

An important aspect for the downstream analysis of terraces is that all data sets need to comprise a comprehensive taxon. In other words, each data set needs to contain at least one taxon that has data for *all* partitions (Biczok et al., 2018). However, not all of our data sets comprised such a comprehensive taxon a priori. Therefore, we needed to reduce some data sets by systematically removing partitions, until the reduced data set comprised at least one comprehensive taxon. We conduct this reduction as follows: we count the number of partitions containing data for all taxa and save the one taxon which has the most. Once such a taxon is selected, we remove all partitions from the data set for which this taxon does not have data which yields a smaller data set. This process is repeated until at least one comprehensive taxon is present in the data set. When such a reduction is applied, we need to propagate it to all subsequent files used within the analysis pipeline.

```
nexconvert [name].nex
reducer [name].prematrix [name].prepartition [name].prephy
raxml-ng --msa [name].phy --model [name].partition --prefix [name].parse
```

3.3.2. Inference of ML trees

We inferred best-known ML trees under LB, UB, and SB models with RAxML-NG. For each data set and branch length model configuration, we performed 20 independent tree searches, and recorded the best tree of all searches. We used 10 random trees and 10 randomized stepwise addition order parsimony trees as starting trees.

```
raxml-ng --msa [name].phy --model [name].partition --prefix [name].unlinked
  --brlen unlinked
```

3.3.3. Enumeration of trees on the terrace

Using Terraphast I, we computed the number of trees and enumerated the trees on a terrace for all best-known ML trees under LB, SB, and UB models. Apart from the ML tree file Terraphast I also requires the aforementioned binary presence/absence matrix as input. Terraphast I then outputs the number of trees on the terrace and enumerates all trees on that terrace in Newick format. As we only require these Newick trees for further downstream analyses, we extracted these from the output file and deleted the file header that only contains information about the number of trees on the terrace.

```
terraphast [name].unlinked.raxml.bestTree [name].matrix >
  [name].unlinked.terraphastoutput
tail -n +4 [name].unlinked.terraphastoutput >
  [name].unlinked.treesOnTerrace
```

3.3.4. Calculation of log-likelihood scores

Applying the corresponding tree scoring option, we used RAxML-NG to calculate the LnL scores of all trees on the respective terraces. The input is a PHYLIP alignment file, the partition file, and the terrace tree file created for the UB ML tree. In addition to this analysis for UB terrace trees (later denoted as: UB-LB, UB-SB), we also computed the LnL scores of terrace trees originating from ML trees under the same branch length model (later denoted as: UB, LB, SB). Note that the best-known ML trees per data set from tree searches under UB, LB, and SB models are often not identical. For analyzing quasi-terraces we used the UB, UB-LB, and UB-SB results and for significance testing the LB results.

```
raxml-ng --evaluate --msa [name].phy --tree [name].unlinked.treesOnTerrace
  --model [name].partition --prefix [name].unlinked.loglhs --brlen unlinked
  > [name].unlinked.loglhscoresOutput
```

3.3.5. Further analyses

To better understand quasi-terraces, we performed additional analyses. These include RF distance, LnL-based significance tests as well as NNI analysis.

3.3.5.1. Robinson-Foulds distance

We computed the RF distance between the best ML trees under each branch length model, to assess their similarity. To archive this RAxML-NG calculates the pairwise relative RF distance between all possible two-tree combinations and the average accordingly.

```
cat [name].linked.raxml.bestTree [name].unlinked.raxml.bestTree
  [name].scaled.raxml.bestTree> [name].allBestTrees
raxml-ng --rfdist --tree [name].allBestTrees --prefix [name].RF
```

3.3.5.2. Significance tests

We also conducted significance tests with IQ-Tree. As input we used the partition file, the terrace trees, and the respective best-known ML tree. For this analysis both, the terrace trees as well as the best tree file were analyzed under the LB model. We did so for LB only due to time saving, as the results for SB are expected to be similar.

```
iqtree -s [name].phy -spp [name].partition -te [name].linked.bestTree -z
  [name].linked.treesOnTerrace -zb 10000 -au -zw -pre
  [name].linked.iqtree.sigtest
```

IQ-Tree performs the following common LnL-based significance tests:

- Bootstrap proportions using the RELL method (Kishino et al., 1990)
- Kishino-Hasegawa test (one sided and weighted) (Kishino & Hasegawa, 1989)
- Shimodaira-Hasegawa test (weighted and unweighted) (Shimodaira & Hasegawa, 1999)
- Expected likelihood weight (Strimmer & Rambaut, 2002)
- Approximately unbiased (AU) test by Shimodaira (Shimodaira, 2002)

A short explanation of all the tests can be found in Section 2.1.5.

3.3.5.3. NNI analysis

To explore the trees in the surrounding of a (quasi-) terrace, we additionally performed NNI analyses. Therefore, we applied all possible NNIs to the trees on the terrace and compared them with trees which reside on the terrace. We applied this to all three models (UB, UB-LB, and UB-SB). We then calculated the LnL scores for the NNI trees and calculated the range between the respective maximum and minimum LnL of the trees on the terrace. We also computed the fraction of NNI trees which fall within this minimum-maximum range of LnLs of the trees *on* the terrace and also

recorded NNI trees that had a better LnL than any tree *on* the terrace. With the first we wanted to evaluate, if the NNI trees can reach the good scores of the terrace trees and if so, how many are as good as them. Furthermore we were interested, if NNI trees can even achieve better scores than the trees on the terrace. Additionally, we conducted significance tests (as described in the previous paragraph) on an extended tree set, containing, both trees *on* the terrace and the NNI trees.

```
NNI -f [name].unlinked.treesOnTerrace > [name].unlinked.NNI
raxml-ng --evaluate --msa [name].phy --tree [name].unlinked.NNI --model
  [name].partition --prefix [name].unlinked-scaled.iqtree.NNIloglhs --brlen
  scaled > [name].unlinked-scaled.iqtree.NNIloglhscores
iqtree -s [name].phy -q [name].partition -z [name].unlinked.terraceAndNNI
  -zb 10000 -au -zw -pre [name].linked.terraceNNI.sigtest
```

With this analysis, we intended to obtain further insights on how trees on a terrace differ from those in the NNI neighborhood of the terrace. Especially if there is a similarity in the results compared with the previous ML trees results or if terraces and their neighborhood differed when not dealing with ML trees. Therefore, we generated a random Yule-Harding tree for one of our data sets (Rhododendron) with IQ-Tree, which has as input the partition and PHYLIP alignment file. The random tree is generally not a ML tree but can also lay within a terrace. Hence we performed the same steps as in the aforementioned analyses, to gain deeper knowledge of terraces and their neighborhood developed without ML criterion.

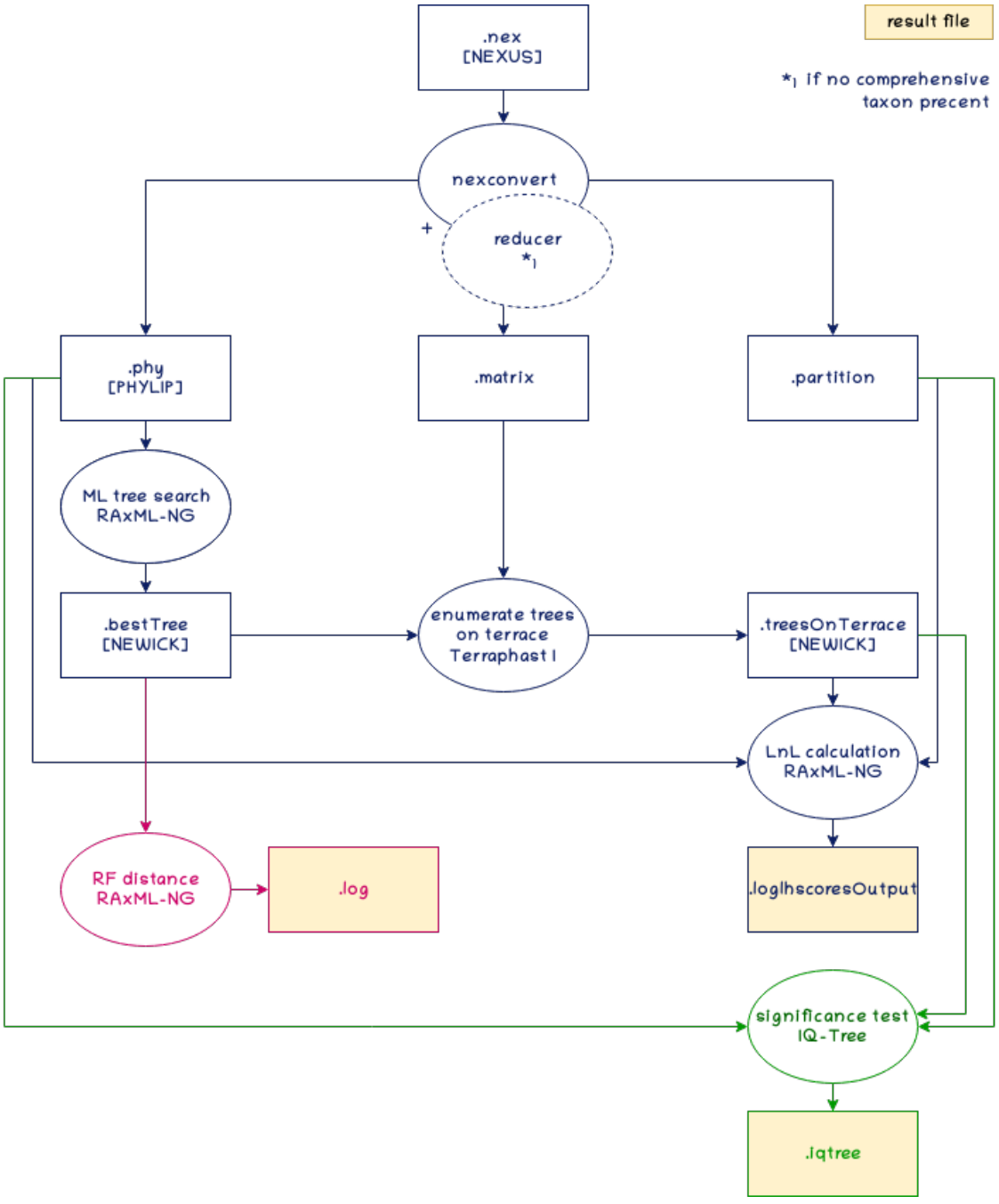


Figure 3.4.: Graphical process overview

4. Results of analyses

After we described the experimental setting of this thesis, we now will present the results from our various analyses. Therefore we first give an overview including the terrace size, average LnL, standard derivation, and relative RF distance. Next the outcome of the significance tests will be described. The last section of this chapter contains the different results from the NNI analysis.

4.1. Overview of results

In Table 4.1 we summarize the results of our analyses. Regarding the average LnL (Avg. LnL), we clearly see that the scores under UB are always better than for SB and in turn for LB. This was expected as the number of free model parameters decreases from UB to SB to LB. For the data sets which have a terrace size of one (i.e., the ML tree does not reside on a terrace) the standard LnL variation (Std. Dev.) is not applicable. For the remaining eight data sets the LnL Std. Dev. is near zero for all data sets under UB and also for four data sets under UB-LB and UB-SB. For the UB-LB and UB-SB case of Eucalyptus and Euphorbia the LnL standard deviation is considerably larger (between 2.14 and 4.58). The same holds for Primula and Rabosky.scincids (between 1.21 and 1.78). The relative RF distance (Rel. RF) between the best tree under LB, UB, and SB is smaller than 0.25 for 93%, and even smaller than 0.1 for 29% of the data sets. The rel. RF ranges between 0.06 and 0.25, except for Eucalyptus where we observe the highest value with 0.5.

For data sets with a terrace size larger than one, we created graphs which enumerate the trees on the x axis by the order in which they appear in the tree file and their LnL on the y axis. The blue line shows the average LnL for each branch length model.

In Figure 4.1 we show an example for the Asplenium data set under all three branch length models. Figure 4.1a shows the UB LnL scores which should, in principle, all be exactly identical. The slight deviations are due to numerical rounding errors associated with floating point numbers. The maximal difference amounts to 0.0012 LnL units. However, for UB-SB in Figure 4.1b and UB-LB in Figure 4.1c the differences are also comparatively small (0.0036 and 0.0028 LnL units, respectively).

Table 4.1.: Overview of results

Dateset	Terrace Size	Avg.∼LnL UB	Avg.∼LnL UB-LB	Avg.∼LnL UB-SB	Std.∼Dev. UB	Std.∼Dev. UB-LB	Std.∼Dev. UB-SB	Rel. RF
(Dobrin et al., 2018)								
Asplenium*	261	-19,900.7	-19,999.6	-19,951.1	0.0003	0.0005	0.0007	0.17
Eucalyptus*	267	-10,902.7	-11,263.9	-11,098.0	0.0120	3.2669	2.1400	0.50
Euphorbia*	1,863	-39,580.3	-41,201.1	-40,173.6	0.0009	4.5787	3.7331	0.15
Iris*	1	-23,493.5	-24,808.8	-24,295.6	N/A	N/A	N/A	0.21
Primula*	1,125	-37,034.1	-38,491.0	-38,084.1	0.0006	1.2102	1.2684	0.15
Rabosky.scincids	3	-125,550.6	-128,649.7	-127,973.0	0.0006	1.7823	1.607	0.13
Ranunculus*	9	-28,709.8	-29,674.1	-29,447.7	0.0009	0.0922	0.1452	0.18
Rhododendron*	27	-17,830.8	-18,216.8	-18,181.8	0.0005	0.1619	0.1644	0.25
Szygium*	5	-11,931.6	-12,214.3	-12,125.0	0.0003	0.1604	0.0155	0.18
Tolley.chameleons	1	-183,197.7	-187,550.7	-185,807.9	N/A	N/A	N/A	0.12
Chernomor et al. (2016)								
d128_34	1	-770,304.1	-810,956.9	-804,359.8	N/A	N/A	N/A	0.06
d69_31	1	-179,745.7	-186,100.3	-185,475.6	N/A	N/A	N/A	0.08
d70_35	1	-249,464.4	-258,459.8	-257,418.1	N/A	N/A	N/A	0.07
d72_51	1	-329,019.2	-339,845.6	-339,099.4	N/A	N/A	N/A	0.10

**: Subsampled data sets. Data sets were subsampled until they contained a comprehensive taxon*

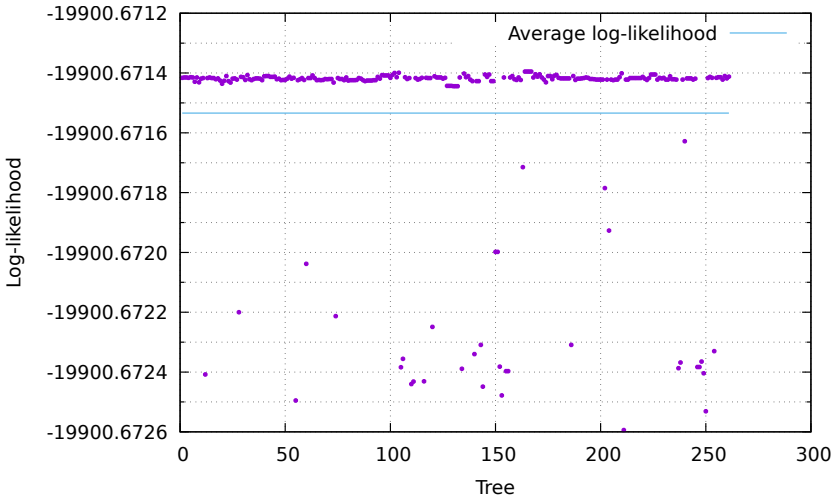
The same is the case for Ranunculus, Rhododendron, and Szygium, where the LnL differences under all branch length models are near zero. In the following paragraph we describe the results for the Eucalyptus data set. For the remaining three data sets (Euphorbia, Primula, and Rabosky.scincids) the differences for UB-SB and UB-LB are larger and range between 3.4 and 22.4 LnL units.

We present another data set, Eucalyptus, in Figure 4.2. Even though it looks more distributed than the Asplenium data set, the numbers are within a range we expected in regards to numerical round of error propagation. For UB, shown in Figure 4.2a, the difference between minimum and maximum LnL amounts to 0.0568 LnL units. In contrast to UB-LB (Figure 4.2c) and UB-SB (Figure 4.2b), where the difference is 7.4991 and 15.6402 LnL units. Besides the numerical results, the Eucalyptus data set has a clearly visible pattern with several peaks under UB. But for UB-LB and UB-SB this pattern is visible accordingly, even though it is fuzzier. As we plotted the LnL in order of trees in the tree file, this pattern comes from the algorithm implemented in Terraphast I.

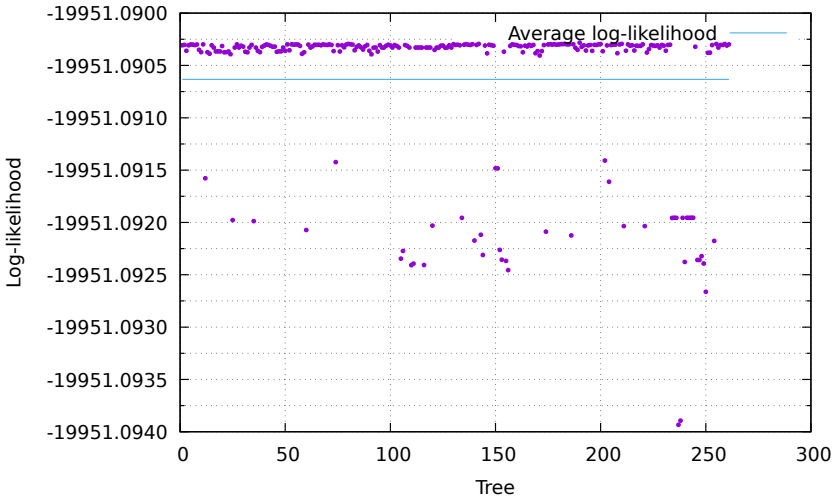
To verify the somewhat structured LnL variations that are visible in the plots, we also performed the calculation of LnL scores with IQ-Tree. The results are so similar, that we do not show them in separate graphs.

4.2. Significance tests

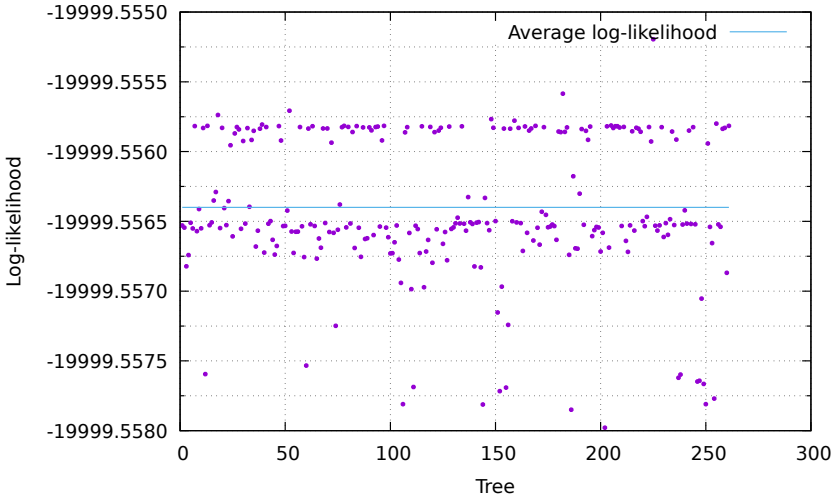
Table 4.2 shows the results of the significance tests with IQ-Tree. As the tests compare the best ML tree on the terrace to all the trees on that terrace under LB, we only included those data sets with more than one tree on the terrace in the results. For each data set, we counted the number of significant and non-significant LnL based differences using a 95% confidence cutoff. For Rabosky.scincids, Ranunculus, and Szygium all trees are not significantly different to each other. For the Primula and Rhododendron data sets six, respectively seven tests show non-significance and for Eucalyptus and Euphorbia the results are mixed. There is no single data set where



(a) Asplenium unlinked

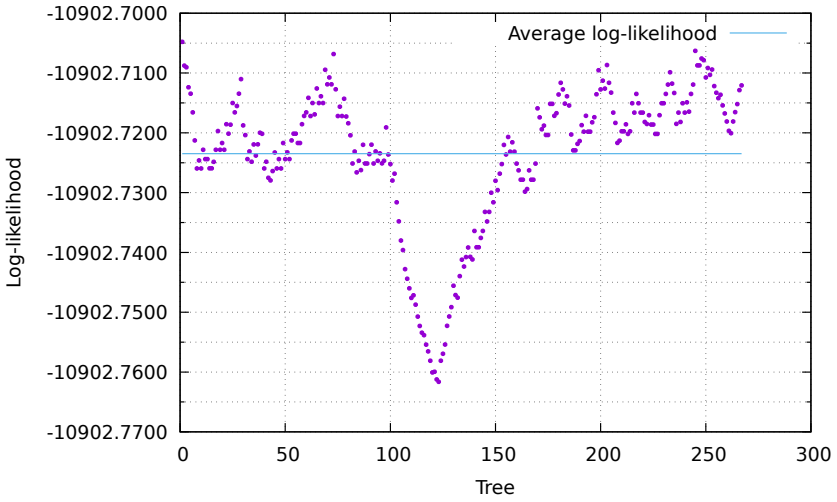


(b) Asplenium unlinked-scaled

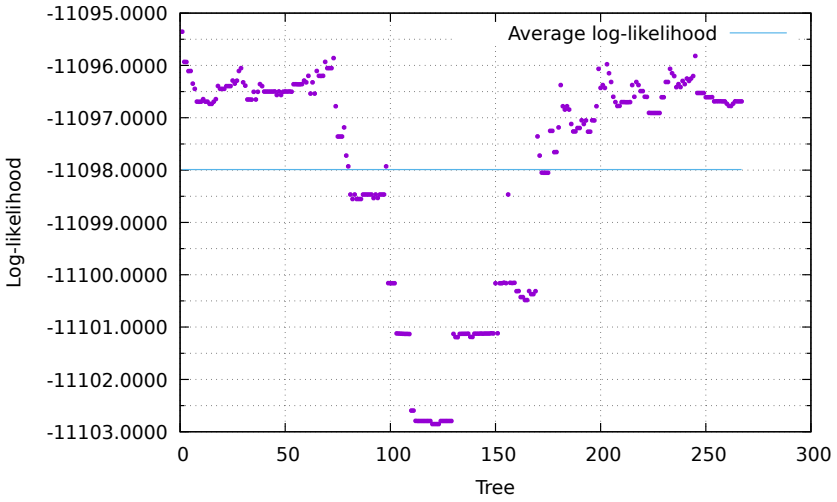


(c) Asplenium unlinked-linked

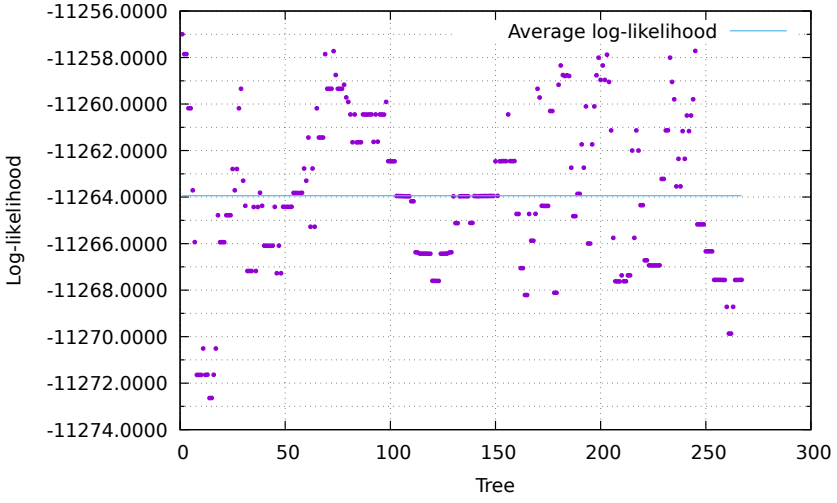
Figure 4.1.: Log-likelihoods for the data set Asplenium under 4.1a UB, 4.1b UB-SB, and 4.1c UB-LB model



(a) Eucalyptus unlinked



(b) Eucalyptus unlinked-scaled



(c) Eucalyptus unlinked-linked

Figure 4.2.: Log-likelihoods for the data set Eucalyptus under 4.2a UB, 4.2b UB-SB, and 4.2c UB-LB model

the trees on the terrace predominantly show significant LnL differences compared to the best ML tree on this terrace under LB.

Table 4.2.: Overview IQ-Tree results of significance tests

(Dobrin et al., 2018)	**bp-RELL**	**p-KH**	**p-SH**	**p-WKH**	**p-WSH**	**c-ELW**	**p-AU**
Eucalyptus	27.0%	43.4%	82.0%	21.7%	51.7%	68.2%	24.7%
Euphorbia	1.4%	26.1%	63.4%	24.5%	98.1%	32.8%	41.9%
Primula	19.0%	97.7%	100%	96.1%	100%	77.9%	98.8%
Rabosky.scincids	100%	100%	100%	100%	100%	100%	100%
Ranunculus	100%	100%	100%	100%	100%	100%	100%
Rhododendron	57.8%	100%	100%	93.3%	97.8%	95.6%	100%
Szygium	100%	100%	100%	100%	100%	100%	100%

Results in percent of trees with non-significant differences

4.3. NNI analysis

With the NNI analysis we intend to explore the neighborhood of the terrace. Due to time limits for the computations on the cluster we used to perform the analyses, the NNI analysis was only performed for the Rhododendron data set. This data set contains sufficient trees on the terrace (27) but is, at the same time, still small enough in terms of taxa (117) such that the resulting NNI trees (6141 unique trees) can be analyzed within a reasonable time frame.

Nevertheless, for the comparison of the LnL of the trees on the terrace and the NNI trees, we only used a subsample of the NNI trees, due to time limits (UB: 2389 corresponds to 38.9%, UB-LB: 3720 corresponds to 60.58%, and UB-SB: 3705 corresponds to 60.33%). For all branch models we calculated the fraction of NNI trees whose LnL fall within the range between the maximum and minimum LnL of the trees *on* the terrace: 15.8% for UB, 30.2% for UB-LB and 27.9% for UB-SB. In addition, we also computed the fraction of NNI trees with a LnL that is better than the maximum LnL from the trees on the terrace. The respective tree sets are very small with only 3.9% for UB, 1.4% for UB-LB, and 2.2% for UB-SB.

We performed the same analysis for a random Yule-Harding tree generated with IQ-Tree from the Rhododendron data set. This random tree yields a terrace of size 9, a NNI neighborhood of 2048 trees (duplicates already removed) and hence a total 2057 trees to be analyzed. Here we found that 18.9% NNI trees fall within the maximum-minimum LnL range of trees *on* the terrace for UB, 30.3% for UB-LB, and 51.5% for UB-SB, which was similar to the results for the ML tree of the Rhododendron data set. We did so to further understand the neighborhood of the terrace. And again, as for the original data set, we calculated the NNI trees with a better LnL than the trees *on* the terrace. The results are: 38.2% for UB, 36.5%, and 18.2% UB-SB, which was between 16 and 35.1 percent higher than in the former analysis.

For all three models we performed another significance test with IQ-Tree. As in the previous significance tests, we counted the number of significant and non-significant LnL based differences using a 95% confidence cutoff. Table 4.3 shows the results. The first 27 trees in the input file are the trees on the terrace (denoted as terrace

Table 4.3.: Results of significance test on data set Rhododendron

		bp-RELL	p-KH	p-SH	p-WKH	p-WSH	c-ELW	p-AU
	All trees	6.89%	76.46%	97.37%	59.34%	81.71%	59.57%	80.89%
UB	Terrace trees	0.00%	100.00%	100.00%	0.00%	0.00%	100.00%	100.00%
	NNI trees	6.92%	76.36%	97.36%	59.60%	82.07%	59.39%	80.80%
	All trees	6.29%	78.75%	96.50%	48.98%	87.01%	60.30%	83.04%
UB-LB	Terrace trees	0.00%	100.00%	100.00%	0.00%	55.56%	100.00%	100.00%
	NNI trees	6.32%	78.65%	96.48%	49.19%	87.15%	61.12%	82.97%
	All trees	6.91%	79.12%	96.50%	40.24%	84.14%	61.69%	83.80%
UB-SB	Terrace trees	0.00%	100.00%	100.00%	0.00%	74.07%	100.00%	100.00%
	NNI trees	6.94%	79.03%	96.48%	40.42%	84.19%	61.52%	83.73%

Results in percent of trees with non-significant differences

trees in the Table 4.3). The following 6141 trees are the NNI trees (denoted as NNI trees in the Table 4.3). This results in a total of 6168 trees for the tests (denoted as all trees in the Table 4.3). The results are mixed, as three tests (bq-RELL, p-WKH, and p-WSH) yield predominantly significant differences and four tests (p-KH, p-SH, c-ELW, and p-AU) predominantly non-significant differences for the trees on the terrace. The NNI trees follow the same pattern for the test results as the results for the trees *on* the terrace and this is the case for all three models.

Table 4.4.: Results of significance tests on a random tree based terrace for the Rhododendron data set

		bp-RELL	p-KH	p-SH	p-WKH	p-WSH	c-ELW	p-AU
	All trees	1.94%	4.23%	19.54%	4.38%	40.25%	2.43%	7.00%
UB	Terrace trees	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
	NNI trees	1.95%	4.25%	19.63%	4.39%	40.43%	2.44%	7.03%
	All trees	1.56%	3.89%	13.08%	2.04%	37.92%	1.99%	2.92%
UB-LB	Terrace trees	0.00%	0.00%	0.00%	0.00%	22.22%	0.00%	0.00%
	NNI trees	1.56%	3.91%	13.13%	2.05%	37.99%	2.00%	2.93%
	All trees	1.51%	3.60%	14.05%	2.14%	47.59%	1.94%	2.92%
UB-SB	Terrace trees	0.00%	0.00%	0.00%	0.00%	55.56%	0.00%	0.00%
	NNI trees	1.51%	3.61%	14.11%	2.15%	47.56%	1.95%	2.93%

Results in percent of trees with non-significant differences

Additionally we performed the significance analysis again for the random tree from the Rhododendron data set. We show the results in Table 4.4. The annotation is the same as for the preceding table and description. In this case we counted 9 terrace trees, 2048 NNI trees, and hence used a total of 2059 trees. Here, the results are clearer as we predominately observe significant different results under all models and all tree sets. Only the results of the WSH test showed up to 47.59% of non-significant differences. We want to emphasize that, in contrast to the results for the best-known ML trees, the trees on the terrace are 100% significant different, except for the WSH test under UB-LB and UB-SB.

5. Conclusion and future work

In this last chapter we summarize our work, provide a conclusion and discuss possible avenues of future work.

The question which motivated our research was if there exists a terrace-like structure under LB and SB length models, which we call a quasi-terrace. We first explained the necessary background in phylogenetics. Thereafter we described the experimental setting of our analysis pipeline. Step-by-step we prepared our data and then performed several analyses to explore structures under LB, SB, and UB model. Throughout the analysis steps we gained the insight, that there indeed exists a terrace-like structure under all three branch length models. Our results thus support the presence of quasi-terraces. We visualize and present our results in Chapter 4.

5.1. Conclusion

Our analysis of 14 data sets indicates that there exists a terrace-like structure under the linked as well as scaled branch length models. We conclude so, based on the computations performed by our analysis pipeline, that is, resulting LnLs under UB-LB and UB-SB, significance tests as well as neighborhood analyses. The average LnL as well as the standard variation are within the same range under all three models. The significance tests under the linked model in Table 4.2 show that most trees are not significantly different from each other. In addition, the NNI evaluation for the Rhododendron data set shows that there is an additional structure in the trees on a quasi-terrace. They are generally better than the surrounding NNIs, even if the subsequent significance tests do not show an uniform result.

Overall, we do observe a quasi-terrace like pattern under linked and scaled branch length models in our empirical test data sets. Even though we could not entirely distinguish the trees on the quasi-terrace from their NNI neighborhood, there is a clear quasi-terrace structure. Therefore, we recommend to further investigate the structure of quasi-terraces, in particular to improve the efficiency of tree search algorithms that should only evaluate one tree per quasi-terrace.

5.2. Future work

Current analysis suggests that quasi-terraces might be helpful for speeding up tree inferences by considering only one representative tree from it. An extensive analysis of more and larger data sets would solidify this notion. Additionally, a mathematical characterization of quasi-terraces would be desirable. We need, for instance, new mathematical tools to better distinguish a quasi-terrace from its neighborhood.

We saw in Table 4.4 that the trees on the terrace and the NNI trees are predominantly significant different. In contrast the results from the ML tree, we showed in Table 4.3, where at least four test show that the trees on the terrace are not significantly different. Hence we conclude that trees on the terrace and the surrounding NNI trees are more diverse, if not generated from a ML tree. On the other hand, terraces and the surrounding NNI trees generated and calculated from ML trees appear to be more similar. But as we only performed this analysis for one data set and one random tree, this should be repeated with more data sets and more random trees to verify this trend.

Regarding to the distinctness of terraces and their surrounding neighborhood, we suggest to use another method called subtree pruning and regrafting (SPR), as this would result in a larger neighborhood. The size of the SPR neighborhood is also the reason why we did not use these moves in our analyses. From such an analysis we hope to obtain additional insights about terraces and their surroundings as well as maybe a clearer distinguishability of them.

Bibliography

Biczok, R., Bozsoky, P., Eisenmann, P., Ernst, J., Ribizel, T., Scholz, F., ... Stamatakis, A. (2018, 05). Two C++ libraries for counting trees on a phylogenetic terrace. *Bioinformatics*, *34*(19), 3399–3401. Retrieved from https://doi.org/10.1093/bioinformatics/bty384 doi: 10.1093/bioinformatics/bty384

Bouchenak-Khelladi, Y., Salamin, N., Savolainen, V., Forest, F., van der Bank, M., Chase, M. W., & Hodkinson, T. R. (2008). Large multi-gene phylogenetic trees of the grasses (Poaceae): progress towards complete tribal and generic level sampling. *Molecular phylogenetics and evolution*, *47*(2), 488–505. Retrieved from http://www.sciencedirect.com/science/article/pii/S1055790308000584 doi: https://doi.org/10.1016/j.ympev.2008.01.035

Burleigh, J. G., Kimball, R. T., & Braun, E. L. (2015). Building the avian tree of life using a large-scale, sparse supermatrix. *Molecular Phylogenetics and Evolution*, *84*, 53–63. Retrieved from http://www.sciencedirect.com/science/article/pii/S1055790314004217 doi: https://doi.org/10.1016/j.ympev.2014.12.003

Chernomor, O., Minh, B. Q., & von Haeseler, A. (2015). Consequences of Common Topological Rearrangements for Partition Trees in Phylogenomic Inference. *Journal of Computational Biology*, *22*(12), 1129–1142. Retrieved from https://doi.org/10.1089/cmb.2015.0146 (PMID: 26448206) doi: 10.1089/cmb.2015.0146

Chernomor, O., von Haeseler, A., & Minh, B. Q. (2016, 04). Terrace Aware Data Structure for Phylogenomic Inference from Supermatrices. *Systematic Biology*, *65*(6), 997–1008. Retrieved from http://dx.doi.org/10.1093/sysbio/syw037 doi: 10.1093/sysbio/syw037

Dell'Ampio, E., Meusemann, K., Szucsich, N. U., Peters, R. S., Meyer, B., Borner, J., ... Misof, B. (2013, 10). Decisive Data Sets in Phylogenomics: Lessons from Studies on the Phylogenetic Relationships of Primarily Wingless Insects. *Molecular Biology and Evolution*, *31*(1), 239–249. Retrieved from https://doi.org/10.1093/molbev/mst196 doi: 10.1093/molbev/mst196

Dobrin, B. H., Zwickl, D. J., & Sanderson, M. J. (2018, April). The prevalence of terraced treescapes in analyses of phylogenetic data sets. *BMC Evolutionary Biology*, *18*(1), 46. Retrieved from https://doi.org/10.1186/s12862-018-1162-9 doi: 10.1186/s12862-018-1162-9

Efron, B., Halloran, E., & Holmes, S. (1996). Bootstrap confidence levels for phylogenetic trees. *Proceedings of the National Academy of Sciences*, *93*(23), 13429–13429. Retrieved from https://www.pnas.org/content/93/23/13429 doi: 10.1073/pnas.93.23.13429

Elloumi, M., & Zomaya, A. Y. (2011). *Algorithms in computational molecular biology: techniques, approaches and applications* (Vol. 21). John Wi-

ley & Sons. Retrieved from https://onlinelibrary.wiley.com/doi/book/10.1002/9780470892107 doi: 10.1002/9780470892107

Fabre, P. H., Rodrigues, A., & Douzery, E. J. P. (2009). Patterns of macroevolution among Primates inferred from a supermatrix of mitochondrial and nuclear DNA. *Molecular Phylogenetics and Evolution*, *53*(3), 808–825. Retrieved from http://www.sciencedirect.com/science/article/pii/S1055790309003169 doi: https://doi.org/10.1016/j.ympev.2009.08.004

Felsenstein, J. (1981, Nov 01). Evolutionary trees from dna sequences: A maximum likelihood approach. *Journal of Molecular Evolution*, *17*(6), 368–376. Retrieved from https://doi.org/10.1007/BF01734359 doi: 10.1007/BF01734359

Felsenstein, J. (2004). *Inferring phylogenies* (Vol. 2). Sunderland, Mass.: Sinauer Assoc. Retrieved from http://www.gbv.de/dms/hebis-darmstadt/toc/103801863.pdf

Harding, E. F. (1971). The probabilities of rooted tree-shapes generated by random bifurcation. *Advances in Applied Probability*, *3*(1), 44–77. Retrieved from https://doi.org/10.2307/1426329 doi: 10.2307/1426329

Hinchliff, C. E., & Roalson, E. H. (2012, 12). Using Supermatrices for Phylogenetic Inquiry: An Example Using the Sedges. *Systematic Biology*, *62*(2), 205–219. Retrieved from https://doi.org/10.1093/sysbio/sys088 doi: 10.1093/sysbio/sys088

Kishino, H., & Hasegawa, M. (1989, Aug 01). Evaluation of the maximum likelihood estimate of the evolutionary tree topologies from DNA sequence data, and the branching order in hominoidea. *Journal of Molecular Evolution*, *29*(2), 170–179. Retrieved from https://doi.org/10.1007/BF02100115 doi: 10.1007/BF02100115

Kishino, H., Miyata, T., & Hasegawa, M. (1990, Aug 01). Maximum likelihood inference of protein phylogeny and the origin of chloroplasts. *Journal of Molecular Evolution*, *31*(2), 151–160. Retrieved from https://doi.org/10.1007/BF02109483 doi: 10.1007/BF02109483

Kozlov, A. M., Darriba, D., Flouri, T., Morel, B., & Stamatakis, A. (2019). RAxML-NG: A fast, scalable, and user-friendly tool for maximum likelihood phylogenetic inference. *bioRxiv*. Retrieved from https://www.biorxiv.org/content/early/2019/03/05/447110 doi: 10.1101/447110

Meredith, R. W., Janečka, J. E., Gatesy, J., Ryder, O. A., Fisher, C. A., Teeling, E. C., ... Murphy, W. J. (2011). Impacts of the Cretaceous Terrestrial Revolution and KPg extinction on mammal diversification. *Science*, *334*(6055), 521–524. Retrieved from https://science.sciencemag.org/content/334/6055/521 doi: 10.1126/science.1211028

Miadlikowska, J., Kauff, F., Högnabba, F., Oliver, J. C., Molnár, K., Fraker, E., ... Stenroos, S. (2014). A multigene phylogenetic synthesis for the class Lecanoromycetes (Ascomycota): 1307 fungi representing 1139 infrageneric taxa, 317 genera and 66 families. *Molecular Phylogenetics and Evolution*, *79*, 132–168. Retrieved from http://www.sciencedirect.com/science/article/pii/S1055790314001298 doi: https://doi.org/10.1016/j.ympev.2014.04.003

Misof, B., Liu, S., Meusemann, K., Peters, R. S., Donath, A., Mayer, C., ... Zhou, X. (2014). Phylogenomics resolves the timing and pattern of insect evolution. *Science*, *346*(6210), 763–767. Retrieved from https://science.sciencemag.org/content/346/6210/763 doi: 10.1126/science.1257570

Nguyen, L.-T., Schmidt, H. A., von Haeseler, A., & Minh, B. Q. (2014, 11). IQ-TREE:

A Fast and Effective Stochastic Algorithm for Estimating Maximum-Likelihood Phylogenies. *Molecular Biology and Evolution*, *32*(1), 268–274. Retrieved from https://doi.org/10.1093/molbev/msu300 doi: 10.1093/molbev/msu300

Nyakatura, K., & Bininda-Emonds, O. R. P. (2012, February). Updating the evolutionary history of Carnivora (Mammalia): a new species-level supertree complete with divergence time estimates. *BMC biology*, *10*(1), 12. Retrieved from https://doi.org/10.1186/1741-7007-10-12 doi: 10.1186/1741-7007-10-12

Pyron, R. A., Burbrink, F. T., Colli, G. R., de Oca, A. N. M., Vitt, L. J., Kuczynski, C. A., & Wiens, J. J. (2011). The phylogeny of advanced snakes (Colubroidea), with discovery of a new subfamily and comparison of support methods for likelihood trees. *Molecular Phylogenetics and Evolution*, *58*(2), 329–342. Retrieved from http://www.sciencedirect.com/science/article/pii/S105579031000429X doi: https://doi.org/10.1016/j.ympev.2010.11.006

Rabosky, D. L., Donnellan, S. C., Grundler, M., & Lovette, I. J. (2014, 3). Analysis and Visualization of Complex Macroevolutionary Dynamics: An Example from Australian Scincid Lizards. *Systematic Biology*, *63*(4), 610–627. Retrieved from https://doi.org/10.1093/sysbio/syu025 doi: 10.1093/sysbio/syu025

Robinson, D. F., & Foulds, L. R. (1981). Comparison of phylogenetic trees. *Mathematical Biosciences*, *53*(1-2), 131–147. Retrieved from http://www.sciencedirect.com/science/article/pii/0025556481900432 doi: https://doi.org/10.1016/0025-5564(81)90043-2

Sanderson, M. J., McMahon, M. M., Stamatakis, A., Zwickl, D. J., & Steel, M. (2015, 05). Impacts of Terraces on Phylogenetic Inference. *Systematic Biology*, *64*(5), 709–726. Retrieved from https://doi.org/10.1093/sysbio/syv024 doi: 10.1093/sysbio/syv024

Sanderson, M. J., McMahon, M. M., & Steel, M. (2011). Terraces in Phylogenetic Tree Space. *Science*, *333*(6041), 448–450. Retrieved from http://science.sciencemag.org/content/333/6041/448 doi: 10.1126/science.1206357

Shi, J. J., & Rabosky, D. L. (2015). Speciation dynamics during the global radiation of extant bats. *Evolution*, *69*(6), 1528–1545. Retrieved from https://onlinelibrary.wiley.com/doi/abs/10.1111/evo.12681 doi: 10.1111/evo.12681

Shimodaira, H. (2002, 05). An Approximately Unbiased Test of Phylogenetic Tree Selection. *Systematic Biology*, *51*(3), 492–508. Retrieved from https://doi.org/10.1080/10635150290069913 doi: 10.1080/10635150290069913

Shimodaira, H., & Hasegawa, M. (1999, 08). Multiple Comparisons of Log-Likelihoods with Applications to Phylogenetic Inference. *Molecular Biology and Evolution*, *16*(8), 1114–1114. Retrieved from https://doi.org/10.1093/oxfordjournals.molbev.a026201 doi: 10.1093/oxfordjournals.molbev.a026201

Soltis, D. E., Mort, M. E., Latvis, M., Mavrodiev, E. V., O'Meara, B. C., Soltis, P. S., ... Rubio de Casas, R. (2013). Phylogenetic relationships and character evolution analysis of Saxifragales using a supermatrix approach. *American Journal of Botany*, *100*(5), 916–929. Retrieved from https://bsapubs.onlinelibrary.wiley.com/doi/abs/10.3732/ajb.1300044 doi: 10.3732/ajb.1300044

Springer, M. S., Meredith, R. W., Gatesy, J., Emerling, C. A., Park, J., Rabosky, D. L., ... Murphy, W. J. (2012, November). Macroevolutionary Dynamics

and Historical Biogeography of Primate Diversification Inferred from a Species Supermatrix. *PLOS ONE*, *7*(11), 1-23. Retrieved from https://doi.org/10.1371/journal.pone.0049521 doi: 10.1371/journal.pone.0049521

Stamatakis, A., & Alachiotis, N. (2010, 06). Time and memory efficient likelihood-based tree searches on phylogenomic alignments with missing data. *Bioinformatics*, *26*(12), i132–i139. Retrieved from http://dx.doi.org/10.1093/bioinformatics/btq205 doi: 10.1093/bioinformatics/btq205

Strimmer, K., & Rambaut, A. (2002). Inferring confidence sets of possibly misspecified gene trees. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, *269*(1487), 137–142. Retrieved from https://royalsocietypublishing.org/doi/abs/10.1098/rspb.2001.1862 doi: 10.1098/rspb.2001.1862

Tolley, K. A., Townsend, T. M., & Vences, M. (2013). Large-scale phylogeny of chameleons suggests African origins and Eocene diversification. *Proceedings of the Royal Society B: Biological Sciences*, *280*(1759), 20130184. Retrieved from https://royalsocietypublishing.org/doi/abs/10.1098/rspb.2013.0184 doi: 10.1098/rspb.2013.0184

Van Der Linde, K., Houle, D., Spicer, G. S., & Steppan, S. J. (2010). A supermatrix-based molecular phylogeny of the family Drosophilidae. *Genetics Research*, *92*(1), 25–38. Retrieved from https://www.cambridge.org/core/journals/genetics-research/article/supermatrixbased-molecular-phylogeny-of-the-family-drosophilidae/8C9158886CC27DA191816203BA3462DA doi: 10.1017/S001667231000008X

Wickett, N. J., Mirarab, S., Nguyen, N., Warnow, T., Carpenter, E., Matasci, N., ... Leebens-Mack, J. (2014). Phylotranscriptomic analysis of the origin and early diversification of land plants. *Proceedings of the National Academy of Sciences*, *111*(45), E4859–E4868. Retrieved from https://www.pnas.org/content/111/45/E4859 doi: 10.1073/pnas.1323926111

Yang, Y., Moore, M. J., Brockington, S. F., Soltis, D. E., Wong, G. K.-S., Carpenter, E. J., ... Smith, S. A. (2015, 04). Dissecting Molecular Evolution in the Highly Diverse Plant Clade Caryophyllales Using Transcriptome Sequencing. *Molecular Biology and Evolution*, *32*(8), 2001–2014. Retrieved from https://doi.org/10.1093/molbev/msv081 doi: 10.1093/molbev/msv081

Zanne, A. E., Tank, D. C., Cornwell, W. K., Eastman, J. M., Smith, S. A., FitzJohn, R. G., ... others (2014). Three keys to the radiation of angiosperms into freezing environments. *Nature Publishing Group*, *506*(7486), 89. Retrieved from https://doi.org/10.1038/nature12872 doi: 10.1038/nature12872

Appendix

A. Pipeline

```
1  #!/bin/bash
2  #SBATCH -o ThirdRun_%j.out
3  #SBATCH -N 1
4  #SBATCH -B 2:8:1
5  #SBATCH -t 24:00:00
6
7  #source /etc/profile.d/modules.sh
8  module load gompi/2017b
9  source /home/breitlpa/shortcuts.sh
10
11 #Preprocessing: Create PHYLIP, matrix and partition file from NEXUS, sanity check
   with RAxML
12 ../nexconvert [name].nex
13 ../reducer [name].prematrix [name].prepartition [name].prephy
14 ../raxml-ng --msa [name].phy --model [name].partition --prefix [name].parse --
       threads 16 --force
15
16 #Step 1: Create linked, unlinked and scaled trees with RAxML
17 ../raxml-ng --msa [name].phy --model [name].partition --prefix [name].linked --brlen
        linked --threads 16 --force
18 ../raxml-ng --msa [name].phy --model [name].partition --prefix [name].unlinked --
       brlen unlinked --threads 2 --force
19 ../raxml-ng --msa [name].phy --model [name].partition --prefix [name].scaled --brlen
        scaled --threads 16 --force
20
21 #Step 2: Calculate trees on terrace with terraphast
22 ../terraphast [name].linked.raxml.bestTree [name].matrix > [name]
       .linked.terraphastoutput
23 ../terraphast [name].unlinked.raxml.bestTree [name].matrix > [name]
       .unlinked.terraphastoutput
24 ../terraphast [name].scaled.raxml.bestTree [name].matrix > [name]
       .scaled.terraphastoutput
25
26 #Step 3: Crop terraphast output to only trees (delete first 3 lines)
27 tail -n +4 [name].linked.terraphastoutput > [name].linked.treesOnTerrace
28 tail -n +4 [name].unlinked.terraphastoutput > [name].unlinked.treesOnTerrace
29 tail -n +4 [name].scaled.terraphastoutput > [name].scaled.treesOnTerrace
30
31 #Step 4: Calculate likelihoodscores for all trees on terrace (for later plotting)
32 ../raxml-ng --evaluate --msa [name].phy --tree [name].linked.treesOnTerrace --model
       [name].partition --prefix [name].linked.loglhs --brlen linked --threads 16 --
       force > [name].linked.loglhscoresOutput
33 ../raxml-ng --evaluate --msa [name].phy --tree [name].unlinked.treesOnTerrace --
       model [name].partition --prefix [name].unlinked.loglhs --brlen unlinked --
       threads 2 --force > [name].unlinked.loglhscoresOutput
34 ../raxml-ng --evaluate --msa [name].phy --tree [name].scaled.treesOnTerrace --model
       [name].partition --prefix [name].scaled.loglhs --brlen scaled --threads 16 --
       force > [name].scaled.loglhscoresOutput
35
```

Figure A.1.: Code of pipeline, line 1-35

```
36  #Step 4b: Calculate likelihoodscores under unlinked tree
37  ../raxml-ng --evaluate --msa [name].phy --tree [name].unlinked.treesOnTerrace --
        model [name].partition --prefix [name].unlinked-scaled.loglhs --brlen scaled --
        threads 16 --force > [name].unlinked-scaled.loglhscoresOutput
38  ../raxml-ng --evaluate --msa [name].phy --tree [name].unlinked.treesOnTerrace --
        model [name].partition --prefix [name].unlinked-linked.loglhs --brlen linked --
        threads 16 --force > [name].unlinked-linked.loglhscoresOutput
39
40  #Step 5: Calculate RF-distance
41  cat [name].linked.raxml.bestTree [name].unlinked.raxml.bestTree [name]
        .scaled.raxml.bestTree> [name].allBestTrees
42  ../raxml-ng --rfdist --tree [name].allBestTrees --prefix [name].RF --threads 16 --
        force
43
44  #----------------------------------------Included
    later----------------------------------------------------------------
45  #Step 6: Preparation and execution of significance test with IQ-Tree
46  ../findBestTree [name].linked.results [name].linked.treesOnTerrace
47  ../iqtree -s [name].phy -spp [name].partition -te [name].linked.bestTree -z [name]
        .linked.treesOnTerrace -zb 10000 -au -zw -pre [name].linked.iqtree.sigtest
48
49  #Step 7: Second analysis of likelihoodscores under unlinked tree with IQ-Tree
50  # -q = linked / -spp = scaled / -sp = unlinked
51  ../iqtree -s [name].phy -sp [name].partition -z [name].unlinked.treesOnTerrace -pre
        [name].unlinked.iqtree.loglhs -n 0
52  ../iqtree -s [name].phy -q [name].partition -z [name].unlinked.treesOnTerrace -pre [
        name].unlinked-linked.iqtree.loglhs -n 0
53  ../iqtree -s [name].phy -spp [name].partition -z [name].unlinked.treesOnTerrace -pre
         [name].unlinked.scaled.iqtree.loglhs -n 0
54
55  #Step 7b: Crop IQ-Tree output to only include loglikelihoodscores
56  awk -F " " '{print $4}' [name].unlinked.iqtree.loglhs.trees | awk -F "=" '{print $2}
        ' > [name].unlinked.iqtree.lhscores
57  awk -F " " '{print $4}' [name].unlinked-linked.iqtree.loglhs.trees | awk -F "=" '
        {print $2}' > [name].unlinked-linked.iqtree.lhscores
58  awk -F " " '{print $4}' [name].unlinked-scaled.iqtree.loglhs.trees | awk -F "=" '
        {print $2}' > [name].unlinked-scaled.iqtree.lhscores
59
60  #Step 8: Creat NNI of all trees on terrace under unlinked model and recalculate
    likelihoodscores
61  ../NNI -f [name].unlinked.treesOnTerrace > [name].unlinked.NNI
62  ../raxml-ng --evaluate --msa [name].phy --tree [name].unlinked.NNI --model [name]
        .partition --prefix [name].unlinked.iqtree.NNIloglhs --brlen unlinked > [name]
        .unlinked.iqtree.NNIloglhscores
63  ../raxml-ng --evaluate --msa [name].phy --tree [name].unlinked.NNI --model [name]
        .partition --prefix [name].unlinked-scaled.iqtree.NNIloglhs --brlen scaled > [
        name].unlinked-scaled.iqtree.NNIloglhscores
64  ../raxml-ng --evaluate --msa [name].phy --tree [name].unlinked.NNI --model [name]
        .partition --prefix [name].unlinked-linked.iqtree.NNIloglhs --brlen linked > [
        name].unlinked-linked.iqtree.NNIloglhscores
65
66  #Step 9: Significance tests with IQ-Tree on all three models for terrace+1NNI
67  ../iqtree -s [name].phy -sp [name].partition -z [name].unlinked.terraceAndNNI -zb
        10000 -au -zw -pre [name].unlinked.terraceNNI.sigtest
68  ../iqtree -s [name].phy -q [name].partition -z [name].unlinked.terraceAndNNI -zb
        10000 -au -zw -pre [name].linked.terraceNNI.sigtest
69  ../iqtree -s [name].phy -spp [name].partition -z [name].unlinked.terraceAndNNI -zb
        10000 -au -zw -pre [name].scaled.terraceNNI.sigtest
```

Figure A.2.: Code of pipeline, line 36-69

B. Reducer

```
#!/usr/bin/python
import sys
import os

def process(argc, argv):
    if (argc < 3):
        print "Usage: reducer.py name.prematrix name.prepartition name.prephy"
        return

    matrixFile = argv[1]
    partitionFile = argv[2]
    phyFile = argv[3]

    #read in binary matrixFile
    matLines = [line.strip() for line in open(matrixFile) if len(line.strip()) > 0]
    numTaxa = matLines[0].split()[0]

    #find columns to be removed in order to have a comprehensive taxon (find maximum of 1s)
    maximum = 0
    posZero = []
    for line in matLines:
        counter = countOnes(line)
        if (counter > maximum):
            posZero = getPositionOfZeros(line)
            maximum = counter

    #reduction only needed if no comprehensive taxon available
    if (len(posZero) > 0):
        #produce reduced matrix
        matfile = open(matrixFile.replace(".prematrix", ".matrix"), "w")
        for pos, line in enumerate(matLines):
            if (pos == 0):
                matfile.write(numTaxa + " " + str(maximum) + "\n")
            else:
                newline = ""
                for index, char in enumerate(line.split(" ")):
                    if (index not in posZero):
                        newline += char + " "
                matfile.write(newline.strip() + "\n")

        #read in partition file and produce reduced partition file
        parLines = [line.strip() for line in open(partitionFile) if len(line.strip()) > 0]
        dif = 0
        partitionsOld = []
        parFile = open(partitionFile.replace(".prepartition", ".partition"), "w")
        newSeqLength = 0
        for index, line in enumerate(parLines):
            parRange = line.split("=")[1]
            name = line.split("=")[0].strip()

            #partition to keep with old numbering
            start = int(parRange.split("-")[0])
            end = int(parRange.split("-")[1])
            partition = Partition(start, end)
            if (index not in posZero):
                partitionsOld.append(partition)
                parFile.write(name + " = " + partition.sub(dif).toString() + "\n")
                newSeqLength = str(partition.sub(dif).end)
            else:
                dif += partition.length
```

Figure B.3.: Code of reducer script, line 1-60

```
61
62          #read in phylip file and produce reduced phylip file
63          phyLines = [line.strip() for line in open(phyFile) if len(line.strip()) > 0]
64          phyFile = open(phyFile.replace(".prephy", ".phy"), "w")
65          for index, line in enumerate(phyLines):
66              if (index == 0):
67                  phyFile.write(phyLines[0].split(" ")[0] + " " + newSeqLength + "\n")
68              else:
69                  species = line.split()[0]
70                  sequence = line.split()[1]
71                  whitespaces = ""
72                  newline = species + whitespaces.ljust(line.count(' '))
73                  for partline in partitionsOld:
74                      newline += sequence[partline.start:partline.end]
75                  phyFile.write (newline.strip() + "\n")
76          print "Reduction finished!"
77
78      #if no reduction needed only rename files
79      else:
80          os.rename(matrixFile, matrixFile.replace(".prematrix", ".matrix"))
81          os.rename(partitionFile, partitionFile.replace(".prepartition", ".partition"))
82          os.rename(phyFile, phyFile.replace(".prephy", ".phy"))
83          print "No reduction needed!"
84
85
86  #outsourced functions
87  def getPositionOfZeros (line):
88      result = []
89      for index, char in enumerate(line.split()):
90          if (char == "0"):
91              result.append(index)
92      return result
93
94  class Partition:
95      def __init__(self, start, end):
96          self.start = start - 1
97          self.end = end
98          self.length = self.end - self.start
99
100     def sub (self, dif):
101         return Partition(self.start - dif + 1, self.end - dif)
102
103     def toString (self):
104         return str(self.start + 1) + "-" + str(self.end)
105
106 def countOnes (line):
107     arg = line.split()
108     counter = 0
109     for index in arg:
110         if (index == "1"):
111             counter += 1
112     return counter
113
114 process(len(sys.argv), sys.argv)
```

Figure B.4.: Code of reducer script, line 61-114