FINDING STRUCTURE IN THE PHYLOGENY SEARCH SPACE

by

Seyed Amin Khalafvand

Submitted in partial fulfillment of the
requirements for the degree of
Master of Computer Science

at

Dalhousie University
Halifax, Nova Scotia
August 2015

I dedicate this work to my beloved mother

Table of Contents

List of Tables

List of Figures

Abstract

A phylogenetic tree is a graphical representation of inferred evolutionary relationships between a set of species or taxa. Phylogenetic trees play an important role in diverse research fields, including molecular biology, ecology, and physiology. Inferring the optimal phylogenetic tree using the maximum likelihood optimality criterion (a popular optimality criterion for phylogenies), is an NP-hard problem. Therefore, use of heuristics and optimization algorithms is necessary to solve this problem. Here, I offer some insights into the structure of the phylogeny search space by analysing novel ruggedness measures. I use a variety of nine-taxon and larger datasets as well as Subtree Prune and Regraft (SPR) and Nearest Neighbour Interchange (NNI) tree rearrangements to characterize and capture the ruggedness of the resulting phylogeny search spaces. Finally, inspired by my analysis of the structure of phylogeny search space, I propose two randomized algorithms to find the optimal tree in the phylogeny search space.

Acknowledgements

Many thanks go to all people who have helped and inspired me during working on this thesis. This work would never have been completed without their support.

I first would like to express my gratitude to my supervisors, Dr. Christian Blouin and Dr. Norbert Zeh, for their useful comments, supervision, and guidance on my thesis.

I would also like to take this opportunity to thank my family and my girlfriend for giving me support and inspiration during the thesis work.

Chapter 1

Introduction

1.1 Introduction

A *phylogenetic tree* (also *phylogeny* or *evolutionary tree*) is a graphical representation which shows the evolutionary history between a set of species or taxa during a specific time. In a phylogenetic tree, leaves (external nodes) represent different extant species or taxa and internal nodes show inferred ancestors of species [10]. Figure 1.1 shows a phylogenetic tree with 7 species. The history of phylogeny dates back to the $19th$ century; in 1859, Charles Darwin introduced one of the first representations of a phylogenetic tree in his book "On the Origin of Species" [8].

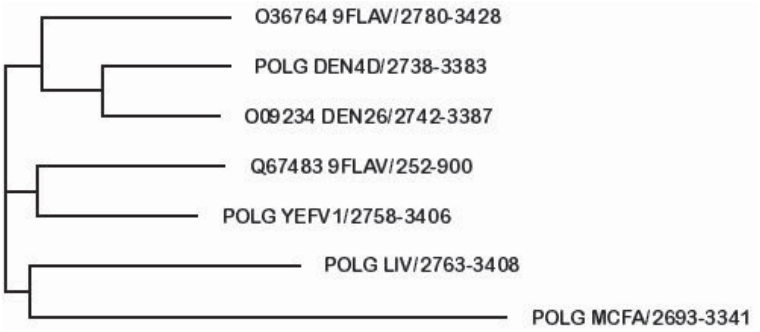

Figure 1.1: Flavivirus RNA-directed RNA polymerase phylogeny from PFAM [2].

Much research has focused on the study of phylogenetic trees since the $19th$ century, and today phylogenies play an important role in different scientific areas, ranging from molecular biology to physiology [37]. Advances in molecular biology have provided the tools needed to infer phylogenetic trees from the DNA or protein sequence data of the modelled taxa or to evaluate how well a phylogeny captures these data. However, due to error and difficulty inherent to the inference approaches, no method can guarantee that the inferred tree structure represents the "true" evolutionary relationships between a set of organisms in the sequence data. Moreover, inferring the optimal phylogenetic trees from DNA or protein sequence data using maximum likelihood, one of the most

popular optimality criteria for phylogenetic trees [10], is NP-hard [5]. The problem of finding the optimal tree structure from sequence data is known as the *phylogeny inference problem*.

An enormous amount of research has focused on tackling the phylogeny inference problem [29, 38, 12, 27, 30, 31, 22, 20, 34] ranging from the use of genetic algorithms to randomized approaches. Effort to develop new algorithms, heuristics, and approaches to address the phylogeny inference problem is still ongoing. The inference approaches in some work such as [38] and [29] are based on a permutation or rearrangement of phylogenetic trees in order to generate the next iteration of trees to search for the optimal tree in the entire set of phylogenetic trees. Nearest Neighbour Interchange (NNI), Subtree Prune and Regraft (SPR), and Tree Bisection and Reconnection (TBR) are typical tree rearrangements for phylogenies. Here, I focus on NNI and SPR tree rearrangements.

Maximum likelihood, parsimony, Bayesian, and minimum evolution methods are typical approaches to evaluate how well a phylogenetic tree explains DNA or protein sequence data. These methods provide a quantitative quality measure of a phylogenetic tree based on DNA or protein sequence data. In this thesis, I focus on maximum likelihood and parsimony approaches. The datasets I use include both DNA and protein sequence data.

In this thesis, I approach the phylogeny inference problem by first analysing the structure of a considerable number of phylogenetic search spaces using different ruggedness measures, and then proposing two randomized algorithms based on the gained insight into the structure of the phylogeny search space.

1.2 Optimality Criteria for Phylogenies

There are different types of methods to find a quantitative quality measure for phylogenetic trees based on DNA or protein sequence data. Maximum likelihood, parsimony, Bayesian, and minimum evolution are typical optimality criteria used to assess how well a phylogenetic tree topology describes the sequence data. Since maximum likelihood is the most popular optimality criterion for phylogenies [10], and many scientific papers (e.g. [29]) use parsimony as optimality criterion for phylogenetic trees, I focus on these two optimality criteria in this thesis.

1.2.1 Maximum Likelihood

In phylogeny, the *maximum likelihood* (ML) (also *likelihood*) optimality criterion is the process of finding the tree topology along with its branch lengths that provides the highest probability observing the sequence data [10]. The phylogenetic tree with the highest likelihood is preferred [10].

Definition 1.1. The *likelihood* of a dataset, D, is the probability of the dataset, given a hypothesis, θ. The hypothesis includes different parameters, such as a phylogenetic tree and a model of evolution. I denote the likelihood of a dataset, D, given a hypothesis, θ as

$$L = P(D|\theta). \tag{1.1}$$

Compared to other optimality criteria for phylogeny, maximum likelihood is statistically well founded, and also has a lower variance (i.e. estimation approach hardly influenced by sampling error). However, it is very time consuming to compute the likelihood of a given tree, and its result is dependent on the selected evolutionary model [10].

1.2.2 Parsimony

In phylogeny, *parsimony* optimality criterion is the fewest number of state-evolutionary changes required for a phylogenetic tree to explain the sequence data. The phylogenetic tree with lowest parsimony is preferred [10].

The simplicity of the parsimony method led many scientists to use it to infer and analyse phylogenetic trees. However, it is not statistically consistent under certain conditions. One of those conditions is when there are long branches in a phylogeny. This problem is known as long branch attraction [10].

Compared to maximum likelihood, parsimony is less time-consuming to compute, but maximum likelihood is more accurate and statistically consistent [10]. There are many hypotheses about the relationship between parsimony and maximum likelihood. As a part of this thesis, I shed some light on these relationships.

1.3 Definitions and Terminology

1.3.1 Graphs

A *graph*, G, includes a set of *edges*, $E(G)$, and a set of *vertices* or *nodes*, $V(G)$. Each edge in a graph is a pair of vertices, called its *endpoints*. Two nodes in a graph are *adjacent* if and only if there is an edge between them. The *degree* of a node in a graph is the number of edges incident to the node.

Graphs can be directed or undirected; In an *undirected graph*, edges do not have any direction, that is, (u, v) and (v, u) are considered to be the same edge. In a *directed graph*, edges are directed, that is, (u, v) and (v, u) denote two different edges. A *path* in a graph, G, is a sequence of vertices $x_0, x_1, ..., x_n$ such that, for all $1 \leq i \leq n$, (x_{i-1}, x_i) is an edge of G.

1.3.2 Phylogenetic Trees

A *phylogenetic tree* is an unordered tree which has two types of nodes: *leaves* or *external nodes* and *internal nodes*. An external node in a tree has degree one, and a node with degree > 1 is an internal node. An unordered tree does not specify an order of the children of a specific node. External nodes are labelled with the names of distinct taxa or species. Internal nodes are unlabelled.

Phylogenetic trees may be rooted or unrooted trees. A *rooted tree* has a distinguished root node and its edges are directed away from the root. Directed paths in such a tree lead from ancestor taxa to their descendants. An *unrooted tree* has no root and edges are undirected and simply represent the relatedness of taxa. Figure 1.2 shows an example of a rooted phylogenetic tree.

Another distinction that is made is based on the degrees of the internal nodes of the tree. A tree is *bifurcating* if every interior node is of degree 3. A tree is *multifurcating* if its interior nodes may have degree greater than 3 [10]. Note that every bifurcating tree is also multifurcating.

In this thesis, I use $E(T)$, $V(T)$, and $L(T)$ to denote the edge set, vertex set (node set), and leaf set of a phylogenetic tree, respectively, and I focus on unrooted bifurcating phylogenetic trees.

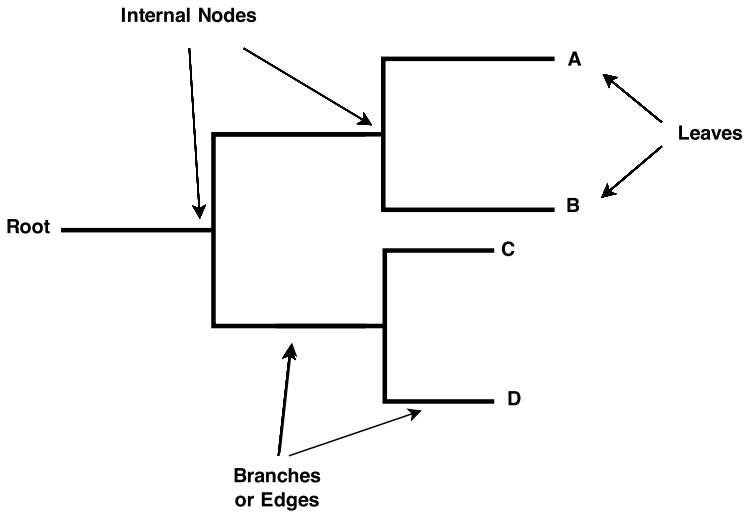

Figure 1.2: A rooted phylogenetic tree with 4 leaves.

1.3.3 Phylogeny Search Space

An n-taxon *phylogeny search space* or *phylogeny landscape* is the set of all possible phylogenetic trees with n leaves along with using a tree rearrangement to define the neighbourhood of each tree. There are $\frac{(2n-3)!}{2^{n-1}(n-1)!}$ different unrooted bifurcating phylogenetic trees in the n-taxon phylogeny search space [10]. I describe phylogenetic tree rearrangements to define the neighbourhood of each tree in the next section.

I use the stepwise addition algorithm [39] described in Section 2.3 of this thesis to enumerate all possible phylogenetic trees in the phylogeny landscape.

1.3.4 Operators on Trees

To produce the next iteration of potential trees during the search for finding the best tree in a phylogeny search space, a method to permute or rearrange a phylogenetic tree is needed. Nearest Neighbour Interchange (NNI), Subtree Prune and Regraft (SPR), and Tree Bisection and Reconnection (TBR) are typical tree rearrangements for phylogenies. For a given phylogenetic tree, NNI, SPR, and TBR tree rearrangements, produce $O(n)$, $O(n^2)$, and $O(n^3)$ neighbours, respectively [10]. In this thesis I focus on NNI and SPR tree rearrangements in order to achieve a computationally reasonable exploration of the phylogeny landscape.

1.3.4.1 Nearest Neighbour Interchange (NNI)

Nearest neighbour interchange (NNI) is a phylogenetic tree rearrangement that constructs a new tree by removing an internal edge and its four incident edges and then reconnects the resulting 4 subtrees using one of the three unrooted topologies on 4 leaves. Figure 1.3 shows the three possible unrooted trees with 4 leaves. Figure 1.4 shows the NNI tree rearrangement on a tree with 5 leaves.

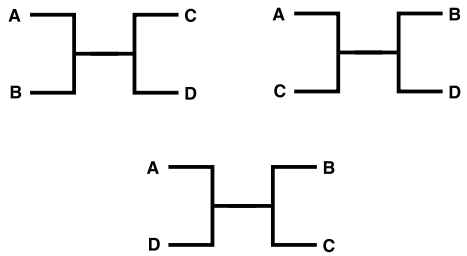

Figure 1.3: The three possible unrooted phylogenetic trees with 4 leaves.

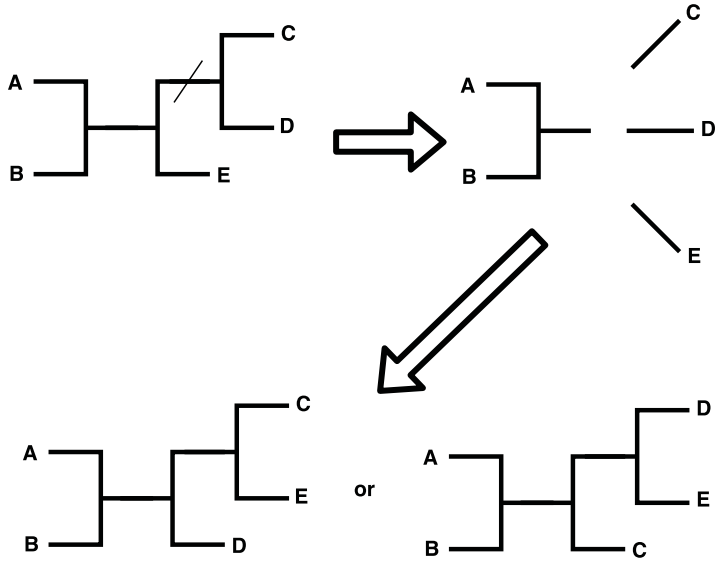

Figure 1.4: An example of NNI tree rearrangement on the crossed internal edge of the tree located in the top-left part of the figure.

There are $n-3$ internal edges in an unrooted phylogenetic tree with n leaves, and for each of these edges, NNI rearrangement creates two different phylogenetic trees (excluding the original one). Therefore, there are $2(n-3)$ NNI neighbours for

each phylogenetic tree. In this thesis, I use $NNI(T)$ to denote the set of all NNI neighbours of the tree T.

1.3.4.2 Subtree Prune and Regraft (SPR)

The *subtree prune and regraft* (SPR) operator cuts an edge (u, v) of T, thereby creating two subtrees T_u and T_v that contain u and v, respectively; creates a new node v' by subdividing an edge in T_v; adds an edge (u, v'); and finally suppresses v. Figure 1.5 shows the SPR tree rearrangement on a phylogenetic tree. There are $O(n^2)$ SPR neighbours for each particular phylogenetic tree [10]. I use $SPR(T)$ to denote the set of all SPR neighbours of the tree T in this thesis.

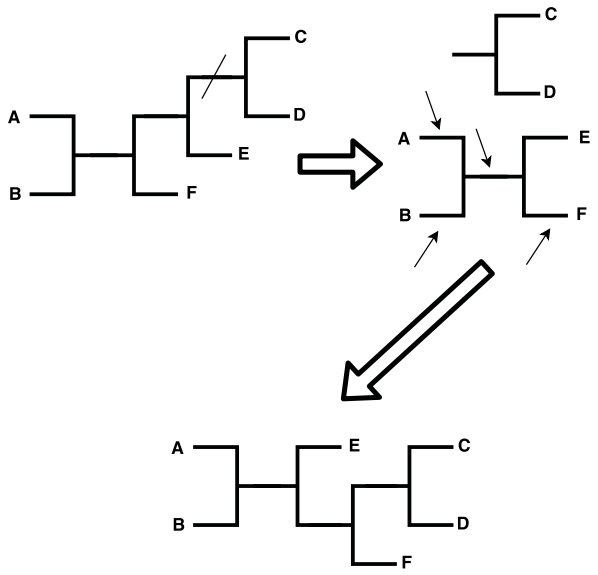

Figure 1.5: An SPR tree rearrangement on the crossed internal edge of the tree located in the top-left of the figure. The top-right part of the figure shows the two subtrees resulting from removing this edge. Arrows denote the potential regrafting place for the pruned subtree. The tree in the bottom part shows the tree obtained by regrafting the pruned subtree on the external edge of leaf F.

1.4 Review of Heuristics for Phylogeny Inference

There are many approaches to search for the best tree in a phylogenetic search space; There are examples of using evolutionary algorithms such as [22] and [20], and also use of stepwise and randomized approaches such as [29], [38], and [27]. I explain

fastDNAml [27], one of the earliest and most basic heuristics for phylogeny inference, and also RAxML [38], a cutting-edge software for phylogeny tree search in this section.

Before explaining the ideas behind these tools, let's first define what do I mean by "moving a subtree by a distance of specified nodes" in these description. Moving a subtree with a distance of specified nodes means that traversing a subtree by crossing a specified number of nodes and reinsert it to a neighbouring branch. Figure 1.6 shows rearrangement of a distance of one node for subtree ST5.

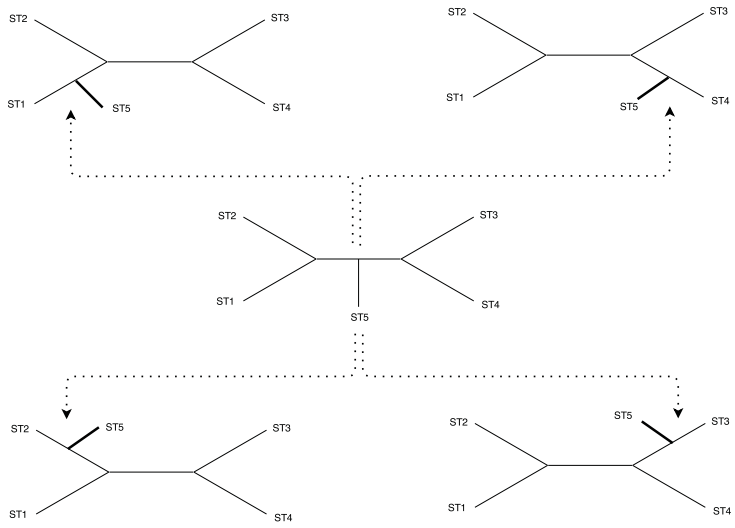

Figure 1.6: An example of moving a subtree to a distance of one node. In this figure, subtree ST5 is moved [38].

1.4.1 fastDNAml

fastDNAml [27] is a tool for phylogeny inference using the maximum likelihood optimality criterion. fastDNAml is inspired by dnaml [29], and its goal is to obtain the same answer as dnaml but faster.

fastDNAml allows users to select a minimum and a maximum distance of nodes specified to be crossed during the tree rearrangement; The default values for both distances are 1.

Suppose I search for the best tree in an n-taxon phylogeny landscape. The idea behind fastDNAml is outlined as follows (the focus tree is the currently best tree

found by fastDNAml):

1. Construct an initial three-taxon tree (there is only one topology) and select it as the focus tree.

2. Pick a new taxon to be added, and attach it to each possible branch of the focus tree. For each of the generated phylogenies, calculate the likelihood and choose a tree with the highest likelihood as the focus tree.

3. If the focus tree includes all n taxa, then go to step 5. Otherwise, proceed to step 4.

4. Perform a tree rearrangement with up to the minimum distance of nodes specified on every subtree in the focus tree. Compute the likelihood for each generated tree. Update the focus tree if there is a phylogeny with a higher score than the focus tree. Return to step 2.

5. Perform a tree rearrangement with up to the maximum distance of nodes specified on every subtree in the focus tree. Update the focus tree if there is a phylogenetic tree with a higher likelihood. Repeat this step until the tree rearrangement can not find a better phylogeny than the focus tree. Then report the focus tree as the best tree.

The difference between step 4 and 5 is that fastDNAml makes an attempt to explore a larger number of trees to find the best phylogeny in the phylogenetic landscape. However, both steps are hill climbing optimization, and use a simple tree rearrangement strategy to produce new candidates for the best phylogeny search in the landscape. In order to infer the best tree in the landscape faster than dnaml, fastDNAml uses Newton-Raphson method instead of EM algorithm to compute the branch lengths of a given tree during the likelihood calculation; It also uses parallel computing to find tree's branch lengths with a higher throughput.

1.4.2 RAxML-III

RAxML-III [38] is a cutting-edge maximum likelihood-based tool to infer large phylogenetic trees. RAxML-III is inspired by fastDNAml [27].

RAxML-III builds a parsimony tree using PHYLIP [29] as a starting tree in order to search for the best tree in the phylogeny landscape. PHYLIP constructs a parsimony tree from a multiple sequence alignment and its goal is to find the best parsimony tree to satisfactorily describe the dataset. The purpose behind building a parsimony tree as a starting tree is that one can expect a parsimony tree to have a higher likelihood score compared to a random starting tree or a neighbour joining tree [38]. After building the parsimony tree, RAxML-III marks the parsimony tree as the currently best tree topology; I call the currently best tree the focus tree.

After building a parsimony tree and choosing it as the focus tree, RAxML-III performs a tree rearrangement to find the best phylogeny. Similar to fastDNAml, it uses a minimum and a maximum distance of nodes specified for moving the subtrees. The tree rearrangement in RAxML includes removing all possible subtrees of the focus tree and reinserting them to minimum and up to maximum distance of nodes. During the tree rearrangement, if a tree has a higher likelihood score than the focus tree, RAxML-III marks it as the focus tree and continues the tree rearrangement on the new focus tree. The default settings for minimum and maximum distance of nodes are 1 and 5 respectively. In the case of no improvement in the likelihood of the focus tree after each step of the tree rearrangement, RAxML-III increases the values of minimum and maximum distance of nodes by 5.

In order to infer the best phylogeny in a timely fashion, computing the likelihood during the tree rearrangement in RAxML-III includes only optimizing the three branches which are adjacent to the insertion point in the tree. However, during the tree rearrangement, RAxML-III saves the best 20 tree topologies with highest likelihoods. After completing one step of the tree rearrangement, RAxML-III performs global branch length optimization on the best 20 tree topologies, and chooses the tree with the highest score as the focus tree.

Once the value of maximum distance of nodes becomes equal to or greater than 21, RAxML-III terminates the tree rearrangement and reports the focus tree as the best tree.

The authors of RAxML-III executed experiments on both real and synthetic datasets, and compared its result with PhyML [12] and MrBayes [34]. RAxML-III performed worse than both PhyML and MrBayes on synthetic datasets but outperforms

both on real datasets in terms of execution time and likelihood score of the final phylogeny.

1.5 Contributions

In this thesis, I aim to develop a better understanding of the structure of phylogeny search space by analysing different ruggedness measures. Since the number of all possible trees in the phylogeny landscape is large, and the landscape's features are dataset-dependent, I construct a variety of phylogeny search spaces from nine-taxon datasets and also use a sampling strategy to explore larger search spaces. Characterization of both nine-taxon and larger landscapes using different ruggedness measures leads me to a better understanding of the structure of the phylogenetic landscapes. Based on this insight, I propose two new randomized algorithms to address the phylogeny inference that are slower than standard heuristics but find the optimal tree.

Chapter 2 presents work on characterizing nine-taxon phylogeny search spaces. I use a variety of nine-taxon empirical and simulated datasets to generate the phylogeny landscapes. I define the neighbourhood of each tree in the landscapes using SPR and NNI tree rearrangements and score trees with likelihood optimality criterion. I explore the landscapes exhaustively and use different ruggedness measures to characterize their structures. I employ different graph-theoretic attributes to design these ruggedness measures, in order to capture different structural properties of the phylogenetic landscape. I use these measures to compare landscapes with different properties, including different protein models and tree rearrangements.

In Chapter 3, I extend my analysis to larger phylogeny landscapes. Since these landscapes are too big to explore exhaustively, I use a sampling strategy to gain insight into their structure. My analysis is based on an evaluation of 30 empirical datasets using the SPR and NNI tree operators to define the neighbourhood of each tree. I show that there is high correlation between parsimony and likelihood optimality criteria using the Pearson correlation coefficient. Therefore, in order to be able to score more trees in the landscape, I use parsimony instead of likelihood as the optimality criterion in my experiments. I use the same ruggedness measure as in Chapter 2 in this chapter. Since I use a sampling strategy to create larger landscapes, the landscapes are not

complete and I can only estimate the values of the ruggedness measures. In order to determine the number of samples sufficient to obtain good enough estimates of the different ruggedness measures, I generated 4 larger landscapes and evaluated each using an increasing number of sample points to determine at which point my estimates start to stabilize.

Chapter 4 presents two randomized algorithms to find the optimal tree based on the characterization and analysis of phylogenetic search spaces in Chapters 2 and 3.

I present conclusions and a discussion of future work in Chapter 5.

Chapter 2

Characterization of Enumerated Phylogenetic Landscapes

2.1 Introduction

In this chapter, I aim to explore the phylogeny search space exhaustively, and develop ruggedness measures to characterize it. I characterize the phylogeny search space for two purposes: to assist in comparing algorithms and heuristics for phylogeny inference, and to establish an environment to compare different attributes of the phylogeny landscape (e.g., different input data, such as separate amino acid or nucleotide models). I use the likelihood optimality criterion, a common optimality criterion for phylogenies [10], to evaluate the phylogenetic trees based on DNA or protein sequence data.

In practice, the largest phylogeny landscape to be generated exhaustively using the likelihood optimality criterion is the nine-taxon landscape; It includes 135,135 different nine-taxon phylogenetic trees. I enumerate a wide range of nine-taxon phylogeny search spaces using different DNA and protein datasets. I use NNI and SPR tree rearrangements to define the neighbourhood of trees in these landscapes. I explore the landscapes exhaustively and develop novel ruggedness measures to characterize them. With these, I also compare the phylogeny landscapes with different properties including different protein models and different types of sequence data.

2.2 Related Work

The work by Money and Whelan [25] is the first paper which characterizes phylogeny search spaces using ruggedness measures. They enumerate the entire phylogeny search space for eight-taxa phylogenetic trees. They exhaustively explore the whole landscapes using SPR and NNI tree rearrangements for different datasets. They then use SPR and NNI search methods implemented in RAxML [38] and PhyML [12] to explore the search space of 20- and 40-taxa phylogenies. They only use the number of optima and the relative size of optima as ruggedness measures for characterizing

phylogeny landscapes. Number of optima is the number of trees which hill climbing converges to them during the tree-search. The size of a local optimum is defined as the fraction of trees which converge to this local optimum when performing the hill climbing algorithm. They use these two ruggedness metrics to compare different phylogenetic landscapes considering the type of amino acid models, the score of the best maximum likelihood tree, and properties of the genes. Afterwards, based on these features they conclude that NNI tree rearrangement performs poorly on real datasets. They also conclude that phylogeny inference software, such as RAxML and PhyML cannot guarantee to estimate the best tree in phylogenetic landscapes.

2.3 Enumeration of the Phylogeny Search Space

Enumerating all possible phylogenetic tree topologies is the first step in building phylogeny search spaces. I use the stepwise addition algorithm [39] to perform this enumeration. The process starts from an initial unrooted phylogenetic tree with three leaves (there is only one possible unrooted topology with three leaves) and then iteratively adds a new leaf to each possible edge. After the first iteration, all tree topologies with four leaves are created. In the second step, the algorithm iteratively continues adding the fifth new leaf to all possible branches in all four-leaf tree topologies, producing fifteen different phylogenetic trees. Continuing this process until adding the ninth leaf generates the whole nine-taxon phylogenetic landscape, which contains 135,135 topologies. Pseudocode 1 describes the process of enumeration.

2.4 Description of Datasets

My datasets include empirical and simulated multiple sequence alignments. I describe both datasets below.

2.4.1 Empirical Data

The set of nine-taxon empirical alignments consist of 100 gapped amino acid alignments taken from the PhylomeDB database [14]. In this database, there are two versions for each alignment, raw and clean. I use both versions of the alignments. In this thesis, I use original and curated instead of raw and clean versions of the alignments,

Pseudocode 1 Enumeration(int N, LinkedTree $Tree$)

Require: N : target number of leaves in tree, and $Tree$: an initial tree

Ensure: All possible phylogenetic tree topologies with N leaves

```
if the initial Tree is null then
  Tree = Three-Taxon-Topology()
end if
treeSize ← number of leaves in Tree
if treeSize is equal to N then
  Add current Tree to list of possible tree topologies
else
  for i ← 1 to treeSize do
    Add a new leaf adjacent to leaf i in Tree, and update Tree
    Enumeration(N, Tree)
    Remove the leaf which is adjacent to leaf i in Tree, and update Tree
  end for
end if
```

respectively from now on.

The process of curating the alignments in the PhylomeDB database is explained as follows: "Positions in the alignment with gaps in more than 10 percent of the sequences were eliminated using trimAl [4] before phylogenetic analysis, unless this procedure removed more than one-third of the positions in the alignment. In such cases, the percentage of sequences with gaps allowed was automatically increased until at least two-thirds of the initial positions were conserved" [14].

Since many scientific papers (e.g. [12]) use LG [19] and WAG [42] protein models for phylogeny inference, I use these models to evaluate the likelihood for each individual tree in the search space. Therefore, each dataset is scored twice.

There are different ways of curating the alignments. None of them is a perfect curating method. Based on the fact that curated alignments in the PhylomeDB database are trimmed using a simple procedure, I use the BMGE software [6] to obtain alternate curated version of PhylomeDB's original alignments. I use an entropy of 0.7 and the BLOSUM30 model as parameters to curate the alignments. Only the LG model is used to calculate the likelihood for phylogenetic trees in the landscapes built

based on these curated alignments.

2.4.2 Simulated Data

I use six different balanced nine-taxon phylogenetic trees (root is considered to be a taxon as well) taken from [28] and then use the Seq-Gen software [32] to produce six nine-taxon ungapped nucleotide alignments with sequences of length 1000. The six balanced nine-taxon phylogenetic trees which I use are shown in Figure 2.1. Finally, I applied the HKY model to produce six simulated phylogeny landscapes. I use likelihood optimality criterion to score each tree in these landscapes.

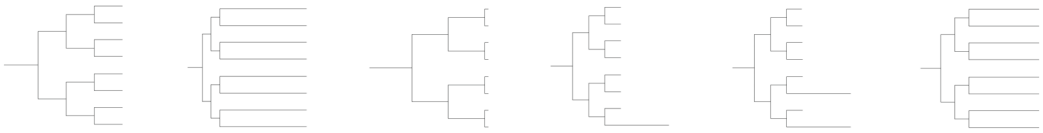

Figure 2.1: The six nine-taxon balanced phylogenetic trees used to generate the simulated datasets. Internal and external branch length pairs, (L_i,L_e), from left to right respectively are: (0.25,0.25), (0.05,0.5), (0.5,0.05), (0.25,0.25(1)), (0.25,0.25(1)), (0.05,0.25). Two of the trees have external branches with different lengths; In order to denote those external branches, I used extra parentheses in (L_i,L_e) beside L_e.

2.5 Hill Climbing Strategy

I use the hill climbing strategy to walk in the search space and employ it to characterize the phylogenetic landscape. The strategy includes starting from an initial tree in the landscape and mark it as the focus tree. Afterwards, it explores the neighbourhood of the focus tree and selects a tree from this neighbourhood as the new focus tree. This process continues until a tree that has no better solution in its neighbourhood is found. At this point, the reached tree is labelled as a local optimum. I apply the hill climbing strategy for each individual tree to find all local optima in a landscape. I mark the local optimum with highest score as the global optimum.

2.6 Description of Ruggedness Measures

I define different measures to capture the ruggedness of the phylogeny landscape. I explain each measure below.

2.6.1 Number of Local Optima (NLO)

The number of local optima (NLO) is the number of trees that do not have a neighbour with higher score than itself. This is determined by starting from each tree and using hill climbing to explore the phylogeny landscape. A lower NLO shows a less rugged or a smoother landscape.

2.6.2 Relative Basin Size of Local Optimum (BS)

The relative basin size of a local optimum (BS) is the fraction of trees that deterministically converge to this optimum through hill climbing. I am mainly interested in the basin size of the global optimum, which I call it BSGO. This value is represented as a fraction of the size of whole phylogeny search space. A higher BSGO shows a less rugged or a smoother landscape.

2.6.3 Number of Unfavourable Moves (NUM)

I define the number of unfavourable moves (NUM) for each local optimum as the number of moves to a lower scoring tree along a shortest path between that local optimum and the global optimum in the landscape. This is done by uniformly sampling k shortest paths between each local optimum and global optimum using breadth-first search. I then calculate the average of unfavourable moves along the k shortest paths for each local optimum and represent it as the NUM.

2.6.4 NUM-Superbasin (SB-NUM)

I identify the NUM-Superbasin (SB-NUM) as merging the basin of a local optimum into the basin of the global optimum if and only if the NUM of the local optimum is one. I perform this process for all local optima in the landscape, and represent it as the SB-NUM. A higher SB-NUM shows a less rugged or a smoother landscape.

2.6.5 AU-Superbasin (SB-AU)

The Approximately Unbiased (AU) test is a technique to evaluate the confidence set of phylogenetic trees, the set of trees that are not rejected to be the true tree topology based on a sequence data [35]. Here, I use the maximum-likelihood based AU-test to

statistically compare two phylogenetic trees. The test determines whether the two trees are statistically the same or not. I use the standard cutoff of 0.05 for the p-value in the test. PAML [47] along with CONSEL [36] are used to perform the AU test.

The AU-Superbasin (SB-AU) is obtained by merging the basin of a local optimum into the basin of the global optimum if the AU-test determines that the local optimum is statistically the same as the global optimum. I perform this process for all local optima in the landscape.

2.6.6 Weighted Average Number of Unfavourable Moves (WANUM)

The weighted average number of unfavourable moves (WANUM) is an extension of the NUM measure. Its main purpose is to summarize the concept of both NLO and relative basin size of each local optimum into one measure. The WANUM for a phylogeny search space is defined as:

$$WANUM = \sum_{i=1}^{NLO} BS_i \times NUM_i \tag{2.1}$$

BS_i and NUM_i are the basin size and NUM of local optimum i in the landscape, respectively. A lower WANUM shows a smoother or a less rugged landscape.

2.7 Empirical Datasets Results

I build the entire nine-taxon phylogenetic landscape and exhaustively explore it using different empirical datasets, phylogenetic tree rearrangements, and evolutionary models. To score the whole landscape, I used the likelihood optimality criterion. In this section, I present my results of characterizing phylogeny landscapes from empirical datasets using my metrics. Before presenting my results, let's first define what do I mean by "a smooth landscape". Once the NLO in a landscape is one, it means that all the tree topologies converge to a same local optimum (the global optimum), BSGO and SB-NUM are 100%, and WANUM is zero. I describe this type of landscape as a smooth landscape.

2.7.1 Number of Local Optima (NLO)

Figure 2.2 compares the NLO in landscapes obtained using SPR and NNI neighbourhoods for 250 different empirical amino acid datasets. In the SPR-based search spaces, the NLO ranges from 1 to 12. In NNI-based landscapes, it ranges from 1 to 550.

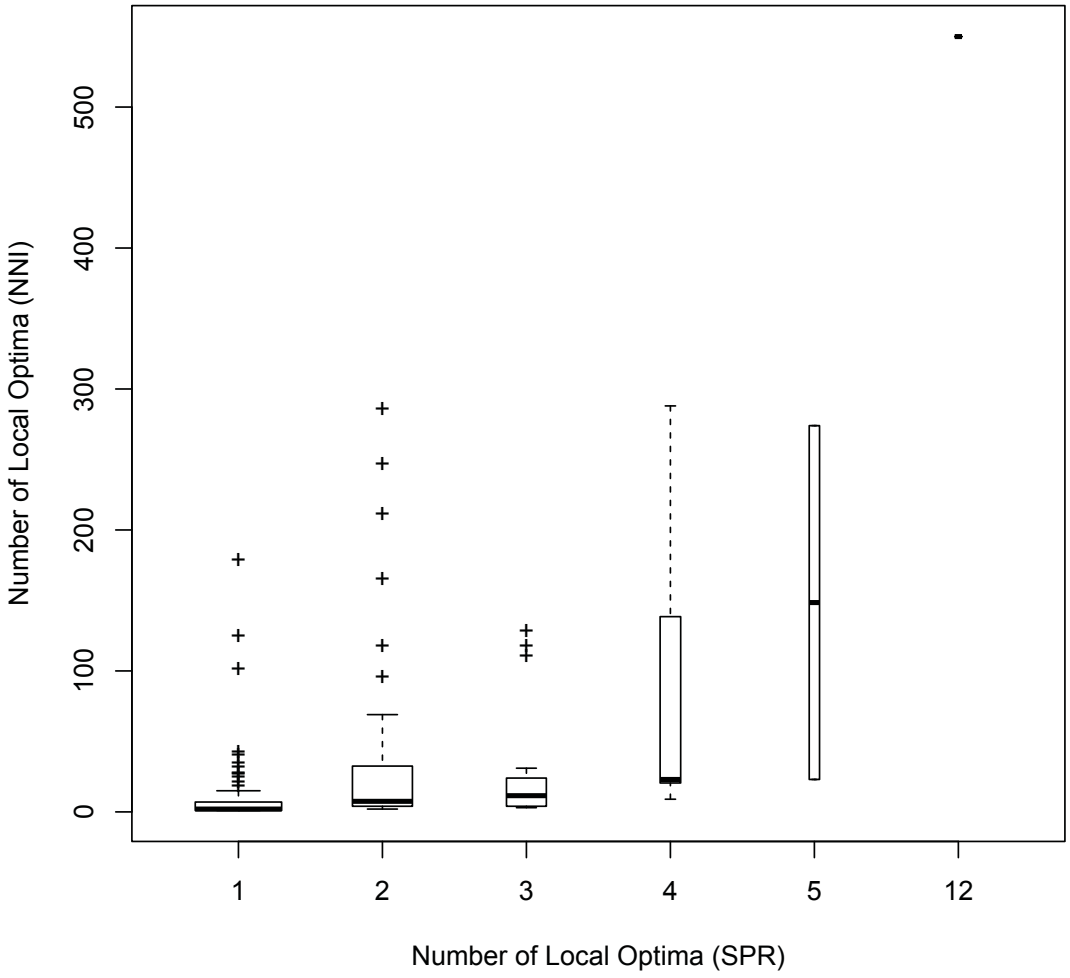

Figure 2.2: Comparing the number of local optima (NLO) in SPR-based search spaces with NNI-based search spaces using 250 different empirical amino acid datasets with both LG and WAG protein models. The width of each box is proportional to the number of observations in each group.

It should be noted that the width of the boxes in Figure 2.2 is proportional to the number of samples in each box. The box corresponding to one local optimum has the

largest width. This simply means that using SPR tree rearrangement often makes the nine-taxon phylogenetic landscape smooth. The median of observations in this box is one, which means more than half of the observations in the box have only one local optimum also for the NNI landscapes. This shows that NNI tree rearrangement often also leads to smooth landscapes if the number of taxa is small.

There is only one SPR landscape with the NLO of 12. It is interesting that this landscape is built from an alignment which is curated by BMGE software.

2.7.2 Relative Basin Size of Global Optimum (BSGO)

Figure 2.3 compares the basin size of the global optimum (BSGO) in SPR-based landscapes versus NNI-based landscapes. As can be seen, lots of datasets are below the diagonal line (solid line) and also the linear regression line is below the diagonal line. These mean that the BSGO in SPR-based landscapes is often larger than BSGO in NNI-based landscapes. It is noticeable in the figure that the BSGO is 100% for many of the SPR-based landscapes. This suggests that using SPR to define the neighbourhood of each tree in the landscape results in a smoother landscape than using NNI. Since the size of the neighbourhood of each tree using SPR is n times bigger than the neighbourhood using NNI, this is expected.

2.7.3 NUM-Superbasin (SB-NUM)

Figure 2.4 compares the SB-NUM metric to the BSGO measure. In more than half of the datasets the basin size of the global maximum gets increased by taking one unfavourable step away from a local optimum (SB-NUM metric). It is interesting that SB-NUM is 100% for a considerable number of the datasets I considered.

2.7.4 AU-Superbasin (SB-AU)

Figure 2.5 compares SB-AU with BSGO in SPR-based search spaces. Only in 26% of the datasets, the basin size of the global optimum gets improved using SB-AU. However, it is interesting to note that there are datasets with BSGO less than 60% but an SB-AU of 100%. This means that all local optima in those landscapes are statistically as good as the global optimum.

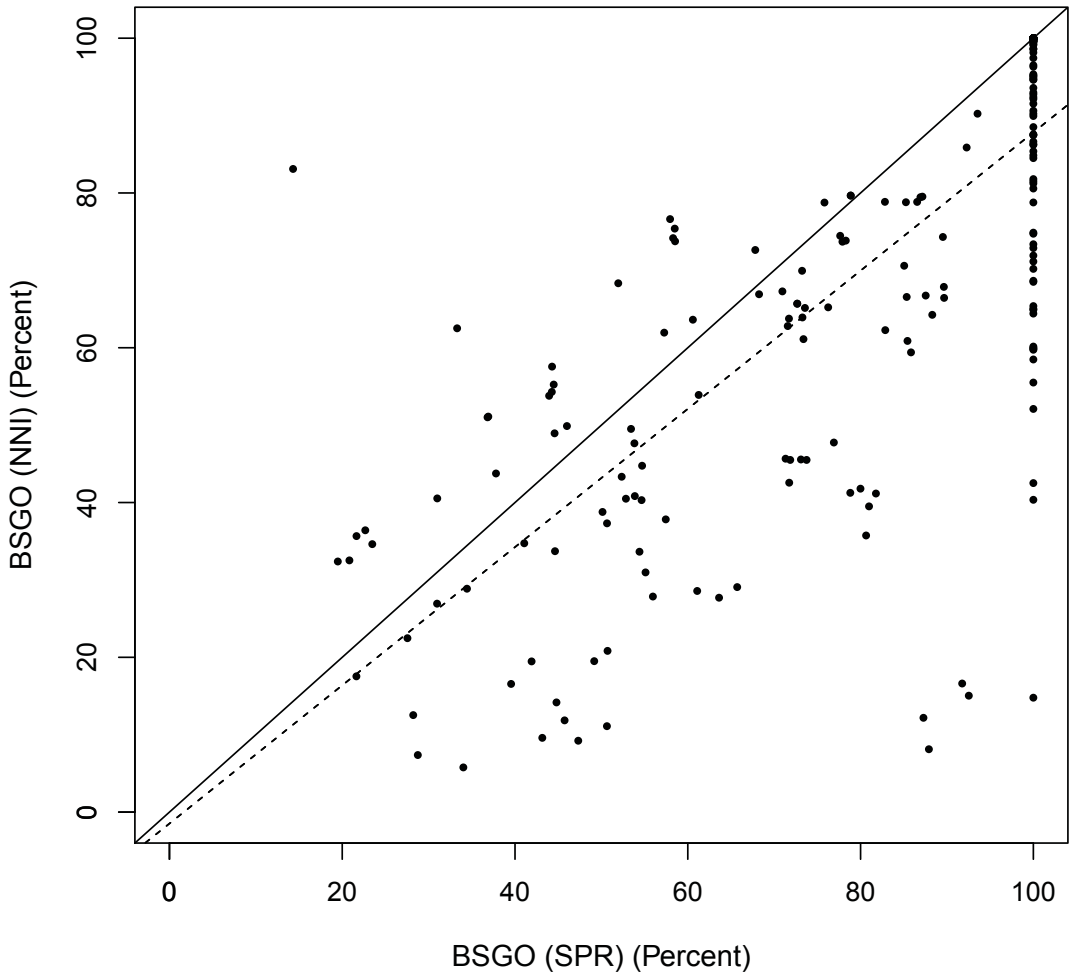

Figure 2.3: The relative basin size of the global optimum (BSGO) in SPR-based landscapes versus NNI-based landscape in all empirical amino acid datasets. The solid line is the diagonal $y = x$. The dashed line is the linear regression line.

2.7.5 Weighted Average Number of Unfavourable Moves (WANUM)

Figure 2.6 shows the comparison between the WANUM in SPR-based search spaces versus NNI-based search spaces. A lower WANUM describes a smoother landscape. It can be seen in Figure 2.6 that for a significant number of datasets the SPR-based search spaces have lower WANUM than the corresponding NNI-based landscapes. This shows that SPR-based landscapes tend to be smoother than NNI-based search spaces.

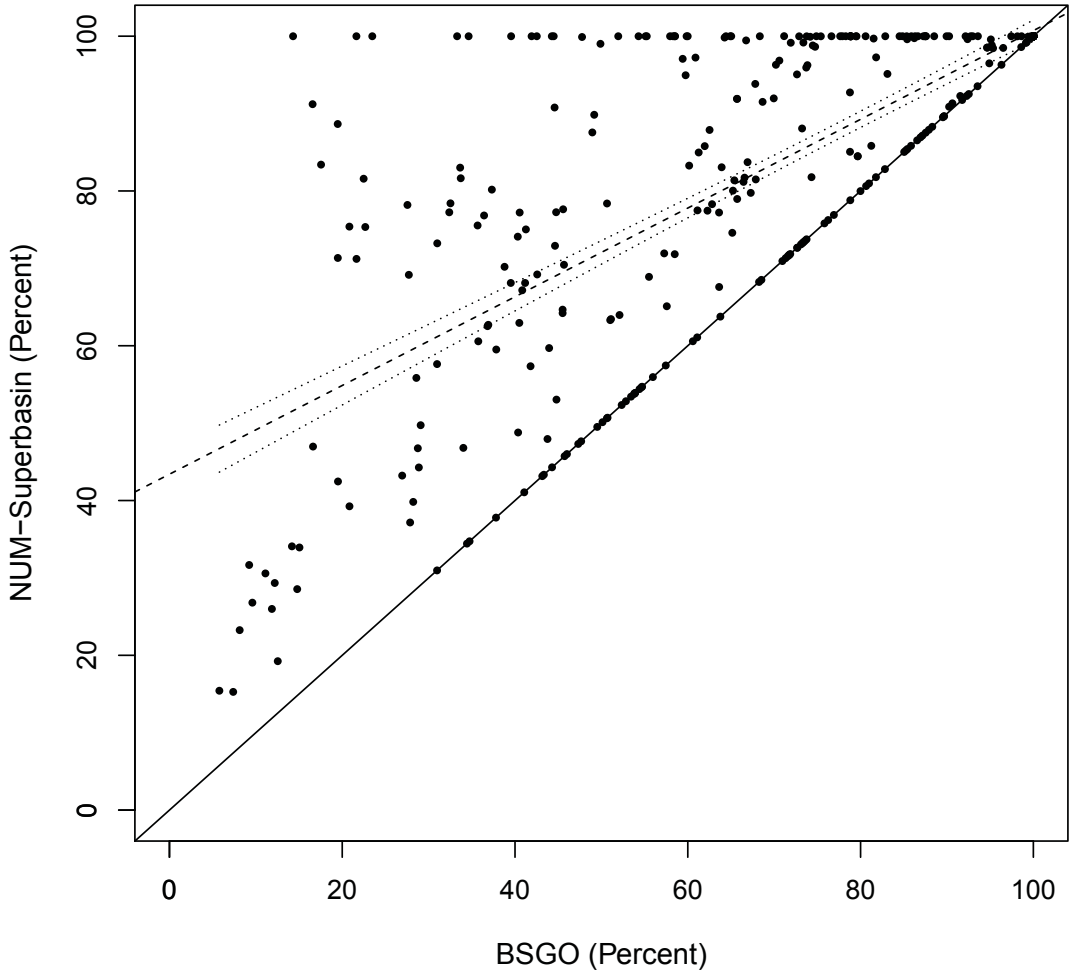

Figure 2.4: Comparing SB-NUM with BSGO. The solid line shows the diagonal $y = x$, the dashed line is linear regression line, and the dotted lines show the linear regression confidence interval. The plot includes all empirical SPR- and NNI-based phylogenetic landscapes.

2.8 Simulated Datasets Results

In order to compare the structure of phylogeny landscapes from the empirical datasets with the simulated datasets, I built six nine-taxon phylogenetic search spaces from simulated datasets using both SPR and NNI tree rearrangements. Table 2.1 represents different ruggedness measures in these phylogeny landscapes. SPR-based landscapes are smooth, as the NLO is always one, BSGO and SB-NUM include all trees in

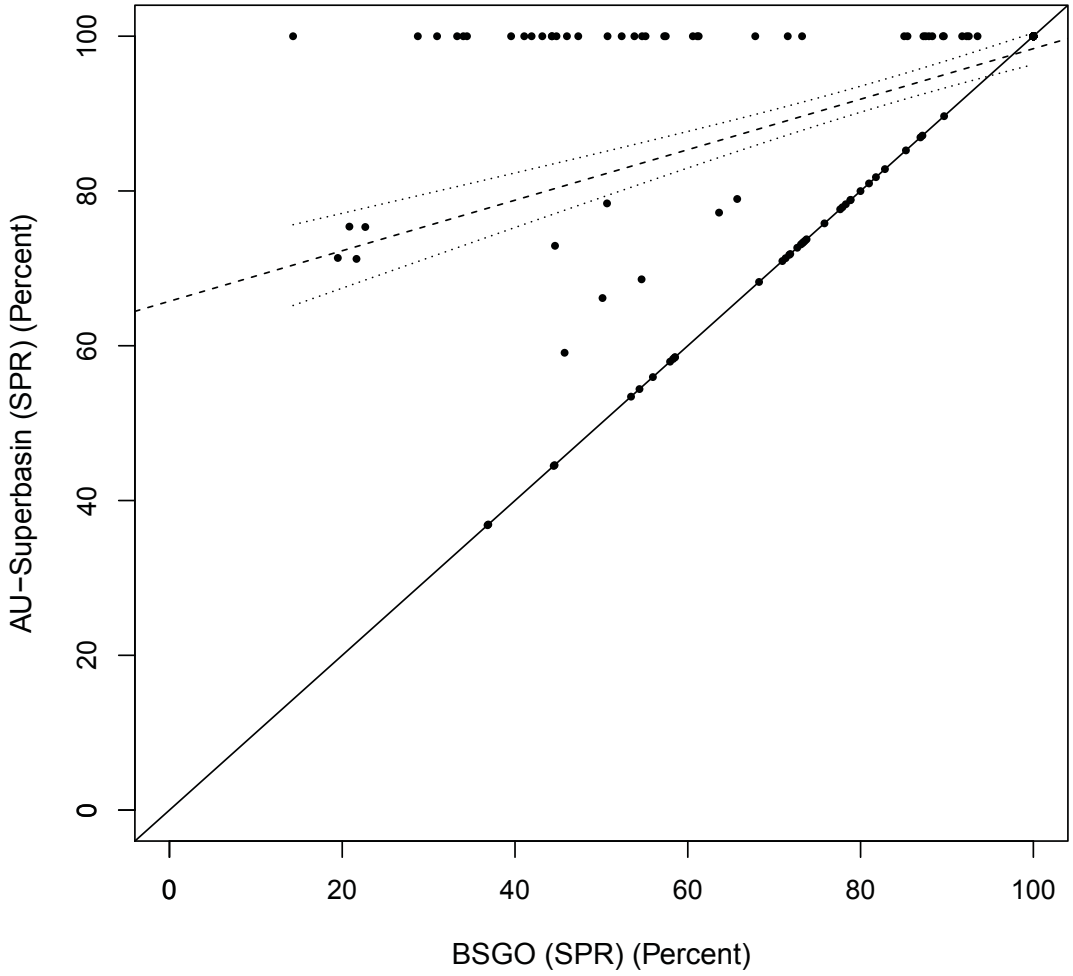

Figure 2.5: Comparing SB-AU and BSGO. The solid line shows the diagonal line $y = x$. The dashed line is the linear regression line, and the dotted lines show the confidence interval of the linear regression. The plot includes all empirical SPR-based phylogenetic landscapes.

the search space, and WANUM is always zero. Despite the fact that in NNI-based landscapes NLO is high, the BSGO and SB-NUM are close to 100%, and WANUM is close to zero. Therefore, it can be concluded that NNI-based landscapes are fairly smooth as well.

Compared to empirical datasets result, both SPR and NNI landscapes from simulated datasets are smoother in terms of a higher BSGO and SB-NUM, and a

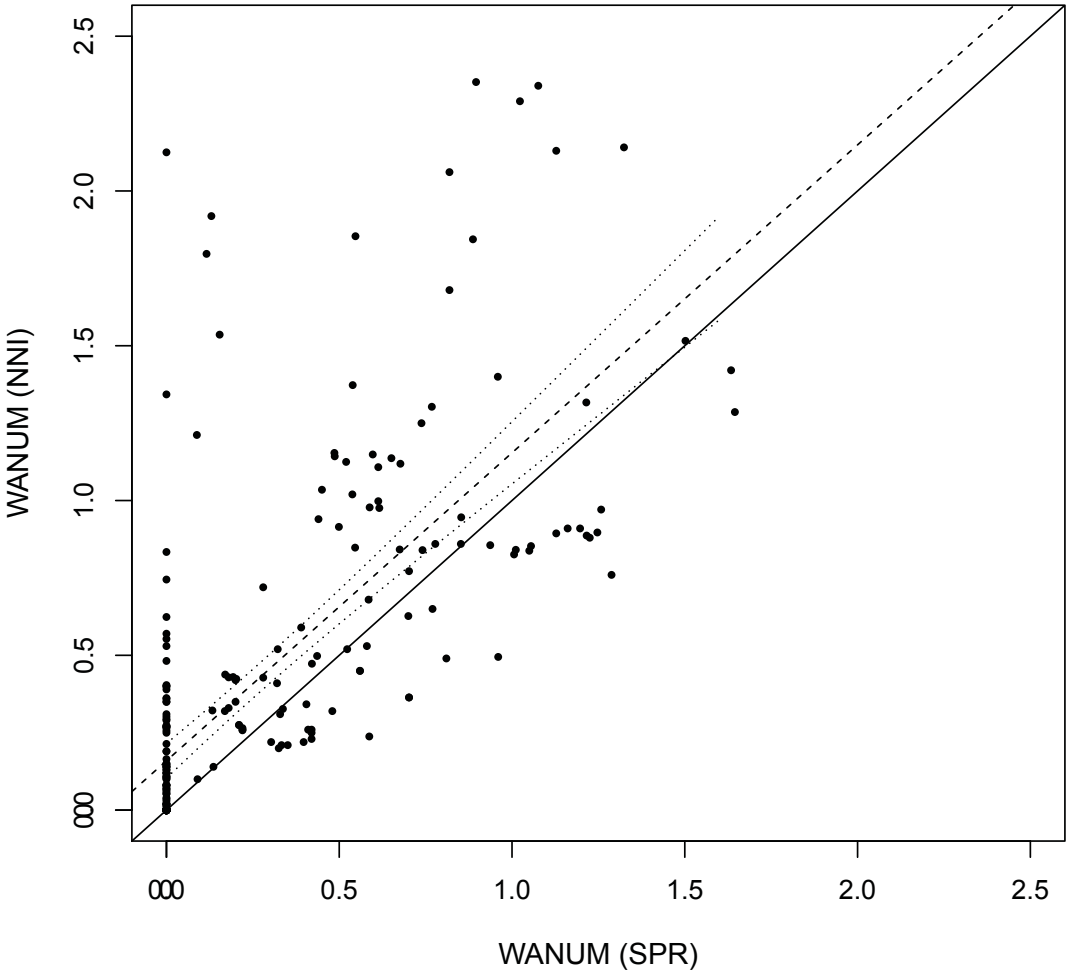

Figure 2.6: WANUM in SPR landscapes vs. NNI landscapes. The solid line shows the diagonal line $y = x$, the dashed line is the linear regression line, and the dotted lines show the confidence interval of the linear regression. The plot includes all empirical amino acid datasets.

lower WANUM.

2.9 Analysing of the Landscape based on Amino Acid Models

In this section, I aim to compare the phylogeny landscapes based on different protein models. The purpose of this comparison is to study the structure of these landscapes, and also to determine if using a specific protein model makes the phylogeny search

Table 2.1: Simulated Datasets Result

Trees	Tree Rearrangement	NLO	BSGO	WANUM	SB-NUM
1	SPR	1	100.00	0.0	100.00
1	NNI	27	99.65	0.0036	99.74
2	SPR	1	100.00	0.0	100.00
2	NNI	11	99.83	0.0025	99.88
3	SPR	1	100.00	0.0	100.00
3	NNI	64	99.04	0.0119	99.43
4	SPR	1	100.00	0.0	100.00
4	NNI	81	99.06	0.0116	99.32
5	SPR	1	100.00	0.0	100.00
5	NNI	15	99.94	0.0007	99.96
6	SPR	1	100.00	0.0	100.00
6	NNI	5	99.99	0.0001	99.99

space smoother. I use the LG-based and WAG-based phylogeny landscapes (both SPR and NNI) from empirical datasets of this chapter.

Figure 2.7 compares the resulting phylogeny landscapes using the WANUM. The linear regression line (dashed) and the diagonal line (solid) are very close to each other and also the 95% confidence interval lines' region (the area between the dotted lines) includes both linear regression and diagonal lines. This indicate that there is a significant correlation between LG-based and WAG-based landscapes. Examining the WANUM of LG-based and WAG-based landscapes suggests that the choice of LG or WAG protein models has no significant impact on the ruggedness of the landscape.

2.10 Conclusion

I examined a variety of empirical and simulated datasets to build the nine-taxon phylogeny search spaces. The SPR and NNI tree operators were used to define the neighbourhood of each tree in the landscapes. I developed different ruggedness measures and compared different properties of phylogeny search spaces. The main conclusions for this chapter are: (i) Based on all ruggedness measures, SPR-based landscapes are often smoother than NNI-based landscapes. (ii) Using one unfavourable step away from a local optimum considerably increases the chance of finding the global optimum compared to exploring the landscape using only different starting trees. (iii)

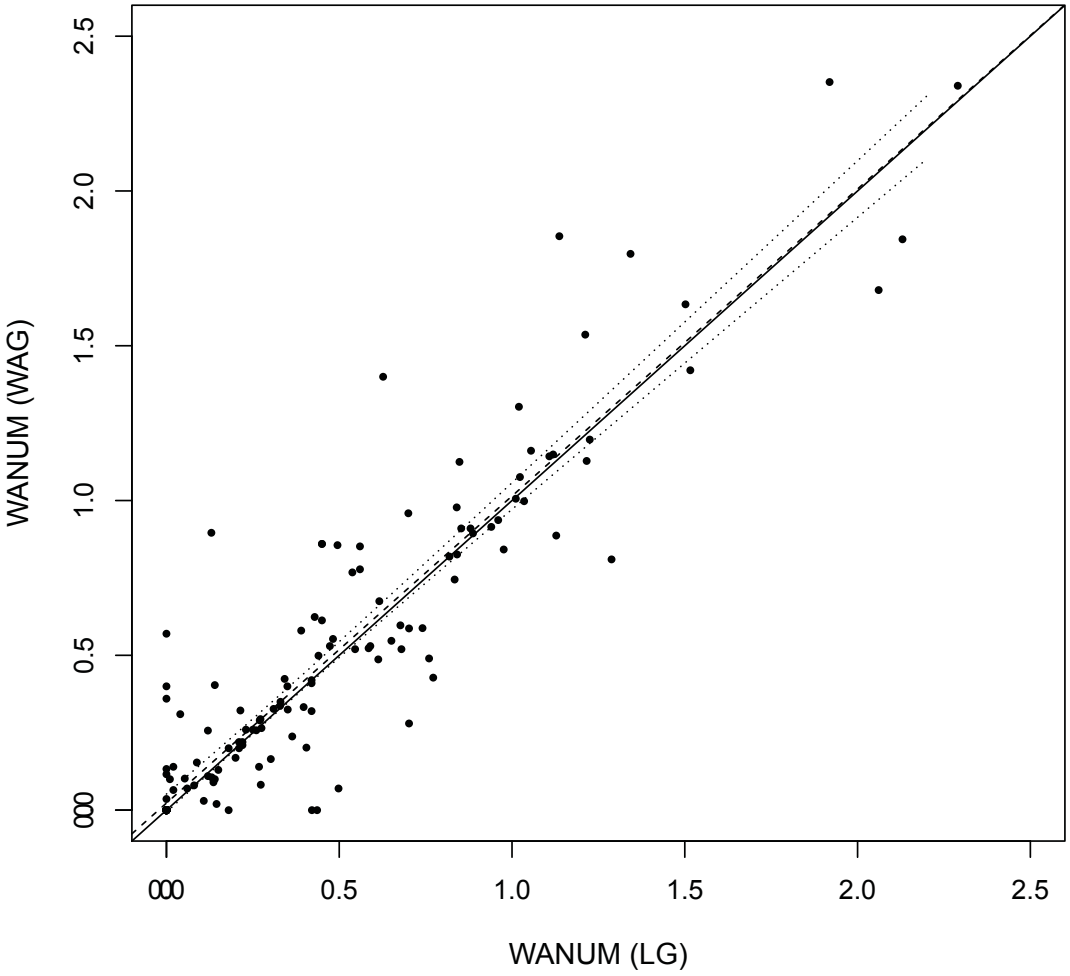

Figure 2.7: WANUM in LG landscape vs. WAG landscape. The solid line shows the diagonal line, the dashed line is the linear regression line, and the dotted lines show the linear regression confidence interval.

Examining the the WANUM of nine-taxon LG-based and WAG-based landscapes suggests that the choice of LG or WAG protein models has no significant impact on the ruggedness of the landscape.

Chapter 3

Characterization of Larger Phylogeny Search Spaces using Sampling

3.1 Introduction

In this chapter, I extend my analysis to larger phylogeny landscapes from the previous chapter. I use a strategy to sample and estimate the ruggedness of larger landscapes, as these landscapes are too big to explore exhaustively. I calculate the correlation between parsimony and likelihood optimality criteria. My results show that there is a significant correlation between these optimality methods. Therefore, in order to score more trees and sample a larger proportion of the search space, I use parsimony instead of likelihood in this chapter.

3.2 Datasets

I use 30 empirical amino acid multiple sequence alignments [12]. These datasets contain 14 to 46 taxonomic units. I score these datasets using the parsimony optimality criterion to build SPR- and NNI-based phylogeny search spaces. I also use nine-taxon datasets from Chapter 2 to compare the parsimony and likelihood optimality criteria.

3.3 Uniformly Sampling the Phylogeny Search Space

As I mentioned in Chapter 2, there are $\frac{(2n-3)!}{2^{n-1}(n-1)!}$ trees in the phylogeny search space (n is the number of taxa). It is impractical to generate the whole phylogeny landscape for $n > 14$ using any optimality criterion. Therefore, sampling is the only practical strategy to characterize larger phylogeny search spaces.

I use the tree topology sampling method described in Appendix A to sample uniform random phylogenetic trees in the phylogeny search space. This method was provided to me by Dr. Norbert Zeh.

3.4 Ruggedness Measures

I use the same ruggedness measures identified in Chapter 2 to characterize larger phylogeny landscapes, but in this chapter I am only able to estimate these measures. Some ruggedness measures have been adjusted to account for changes in landscape attributes. The main change is that in some cases hill climbing finds multiple globally optimal trees with the same score in the search space (in other words, the best discovered trees so far). This affects the definition of NUM, BSGO, WANUM, and NUM-Superbasin (SB-NUM), since the definition of these metrics assumes a single global optimum. I adjust these ruggedness measures so that they take into account a single global optimum or multiple globally optimal trees.

3.4.1 Number of Global Optima (NGO)

As I mentioned, in some cases, hill climbing converges to more than one global optimum in large phylogeny search spaces. I define a new metric of Number of Global Optima, NGO, to capture this.

3.4.2 BSGO

I use the same definition for the basin size of the global optima, BSGO, as in Chapter 2, but since there are multiple global optima, I define the BSGO to be sum of the basin sizes of all global optima.

3.4.3 NUM

Calculating the number of unfavourable moves, NUM, requires to calculate the shortest path between two trees in the phylogeny landscape. I use the Maximum Agreement Forest of two trees from [43] as a basis for finding the shortest path. Section 3.4.4 describes the algorithms used to find the shortest path. The algorithm can only calculate the shortest path in the phylogeny landscape built by SPR tree rearrangement.

After finding the shortest path between two trees, I count the number of unfavourable moves in the shortest path, and represent it as the NUM. It should be

noted that, if two adjacent trees along the path have the same parsimony score, I don't consider the move between these trees to be an unfavourable move.

Since in some phylogeny landscapes, there are several global optima (in other words, the several best discovered trees so far), I calculate the NUM from each local optimum to every global optimum. The NUM score of the local optimum is then the minimum of these values. WANUM and NUM-Superbasin (SB-NUM) are defined as in Chapter 2.

Therefore, each local optimum has multiple NUM values; Afterwards, I identify the minimum value of NUM for each specific local optimum to be considered as the NUM for that local optimum. Using this definition of NUM, the representation of WANUM and NUM-Superbasin (SB-NUM) are exactly same as the definition in Chapter 2.

3.4.4 Computing the SPR Shortest Path between Two Trees

I compute the shortest path between two phylogenetic trees in the SPR landscape in order to calculate the ruggedness measures. I use the rSPR package [43] to compute the Maximum Agreement Forest (MAF) of two trees, T_1 and T_2.

Each MAF has different components; Since the MAF is efficient, if I move subtrees of T_1 based on the components provided by an MAF and also corresponding place in T_2, I generate all trees along the SPR shortest path between two trees.

I label the edges of T_1 and T_2 as belonging to MAF components using the following rules. Before that, I need to define the meaning of finished and unfinished components. A component C is finished if all of the nodes in the components has been visited, otherwise is unfinished.

- If node x in the tree is a leaf, then corresponding edge, e_x, belongs to an MAF component which includes x.

- If node x in the tree is an internal node, then corresponding edge, e_x, belongs to one of the components of its children. If only one of its children's component is unfinished, e_x belongs to that component, otherwise I pick one of the children's component arbitrary.

Considering the component C, I identify the parent's component as follows:

- $\exists\ e \in C$ such that e's parent belongs to a different component.
- If there is no such edge, C's parent is root.

The algorithm generates the shortest path by starting from one of the trees and move its edges to corresponding edges' place in the other tree. An edge is moveable if the component it moves to is not a descendent. Claim 3.1 proves that there is always a moveable edge.

Claim 3.1. There is always a moveable edge.

Proof. If I consider the component C which is containing root, every components that is a child of C in T_2 is moveable. □

If I move the moveable edges in a top-down traversal of one of the trees, the algorithm generates the SPR shortest path between two trees.

3.5 Parsimony vs. Likelihood

I compare parsimony and likelihood as scoring methods for a number of landscapes built with nine-taxon datasets as well as larger datasets. The comparison uses the Pearson correlation coefficient as a test of correlation.

3.5.1 Correlation Coefficient

I use the scores provided by both parsimony and likelihood to score the nodes (which are trees in this case) in a phylogenetic landscape. I also label the edges of the landscapes built using SPR tree rearrangement with the difference of the scores of its endpoints.

Figure 3.1 shows the Pearson correlation coefficient between the likelihood and parsimony approaches on nodes and edges in landscapes built from the empirical datasets in Chapter 2. As shown in Figure 3.1, more than 75 percent of the datasets have correlation higher than -0.8 for both node and edge scores. This shows a strong correlation between these two methods. All these correlations are statistically significant since their p-values are less than 0.05.

I also calculate the correlation between node and edge scores in larger phylogeny search spaces from the datasets used in this chapter. My strategy uniformly samples

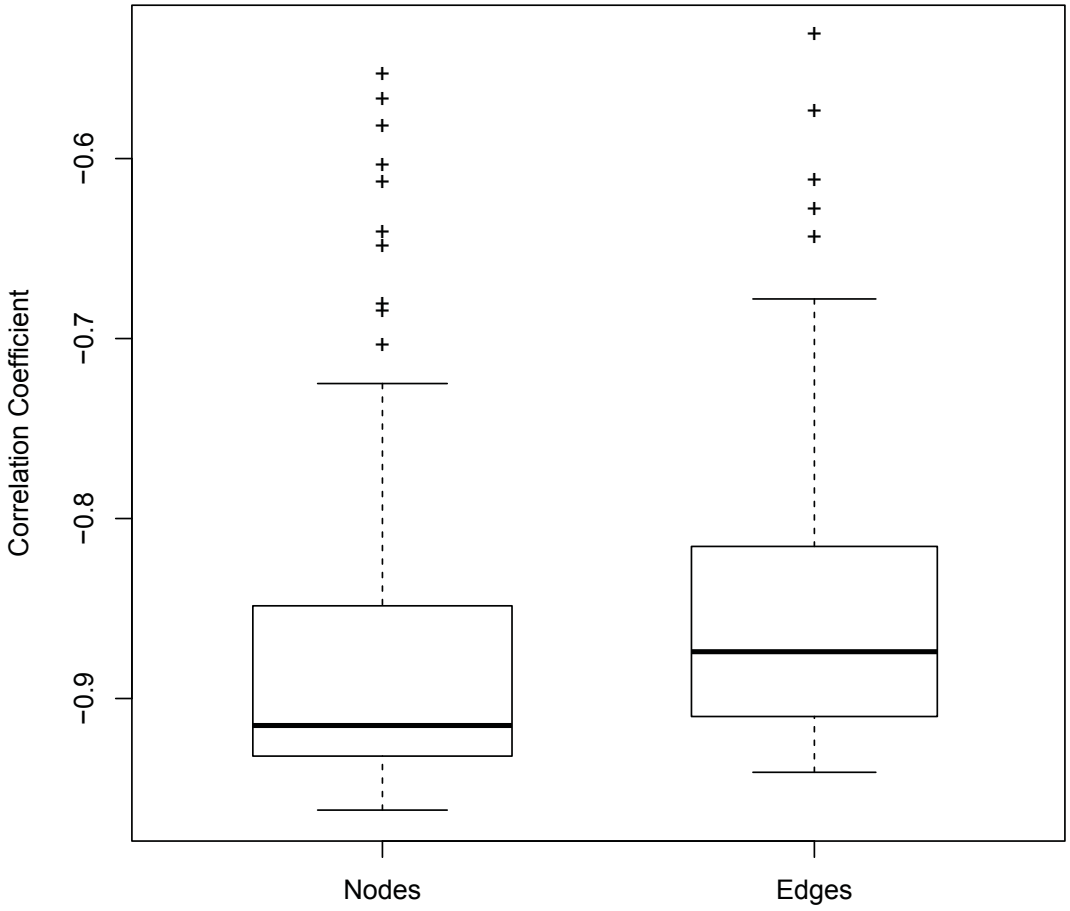

Figure 3.1: Correlation between likelihood and parsimony scores of nodes and edges in nine-taxon phylogeny search spaces.

100 different starting trees and explores the landscape using hill climbing. Figure 3.2 shows the correlation between the likelihood and parsimony methods on nodes and edges in these larger search spaces. As Figure 3.2 shows, for more than 99 percent of the datasets, the correlation between likelihood and parsimony node and edge scores is higher than -0.85, which is a strong correlation. All these correlations are statistically significant since their p-values are less than 0.05.

This work [41] states that maximum likelihood and parsimony are related to each other under simple evolutionary models, and also my results show that there is a

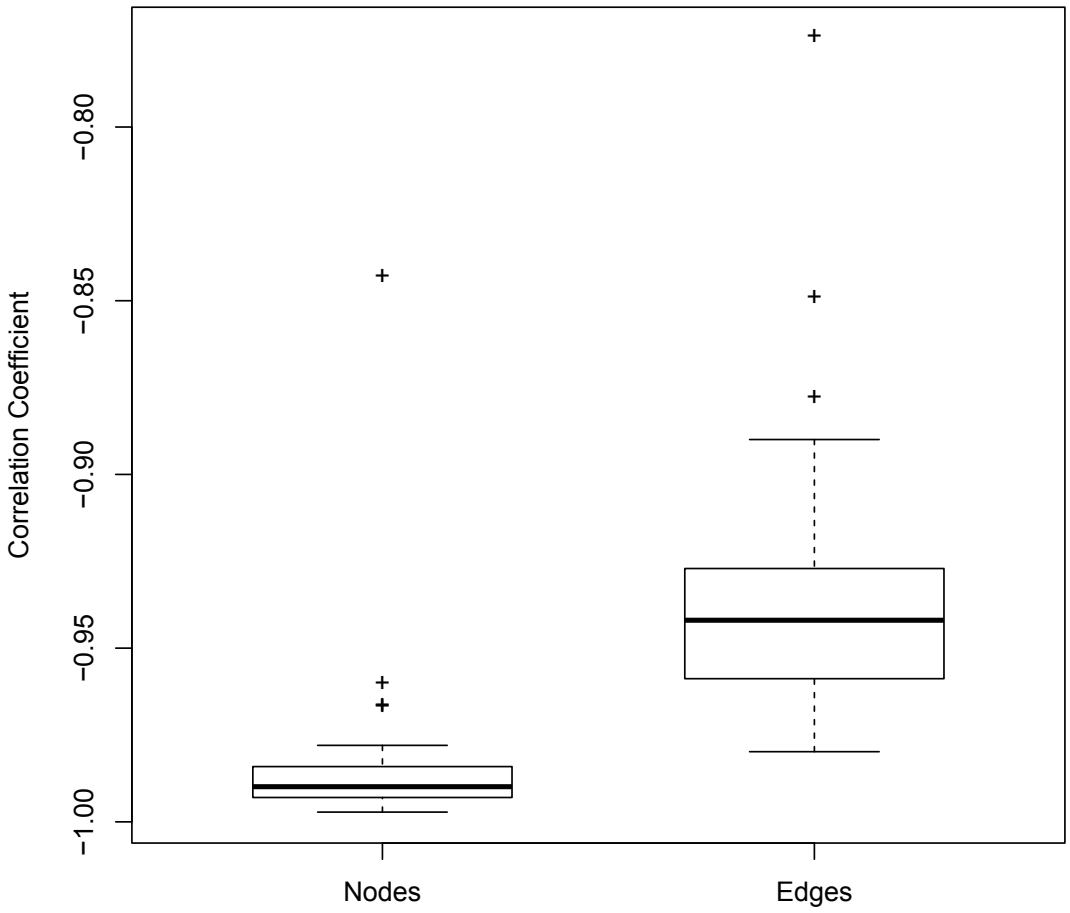

Figure 3.2: Correlation between likelihood and parsimony scores of nodes and edges in larger phylogeny landscapes.

strong correlation between maximum likelihood and parsimony scores. Therefore, due to the high cost of computing maximum likelihood I use parsimony instead of maximum likelihood in this chapter

3.6 Results

I built a variety of larger phylogeny search spaces using 30 empirical datasets and different phylogenetic tree rearrangements. I explore each landscape using hill climbing

starting from 500 uniformly sampled trees. In this section, I present the characterization of larger phylogeny landscapes based on various ruggedness measures estimated from these samples.

3.6.1 NLO and NGO

Figure 3.3 shows the NLO in the SPR-based landscapes versus NNI-based landscapes. The linear regression line (dashed line) is very close to the diagonal line (solid line), and the 95% regression confidence interval (dotted lines) includes both the linear regression and diagonal lines. This shows that there is a significant correlation between SPR- and NNI-based landscapes in terms of the NLO.

Figure 3.4 shows the NGO in the SPR-based landscapes versus the NNI-based landscapes. The linear regression line (dashed line) being above the diagonal line (solid line) indicates that SPR-based landscapes tend to have lower NGO compared to NNI-based search spaces.

3.6.2 BSGO

Figure 3.5 compares the basin size of the global optima (BSGO) in the SPR-based landscapes versus NNI-based ones. Similar to Figure 3.3, the linear regression and diagonal lines are close to each other and the confidence interval include both lines. This indicates that there is a correlation between the BSGO of SPR- and NNI-based landscapes.

3.6.3 SB-NUM

Figure 3.6 compares the BSGO to the SB-NUM for SPR-based landscapes. The plus signs and solid circles represent the SB-NUM and BSGO, respectively. The linear regression line for SB-NUM (dotted line) is above the linear regression line for BSGO. This shows that taking one unfavourable step away from local optima and then returning to hill climbing increases the chance of finding the global optimum in the phylogeny search space.

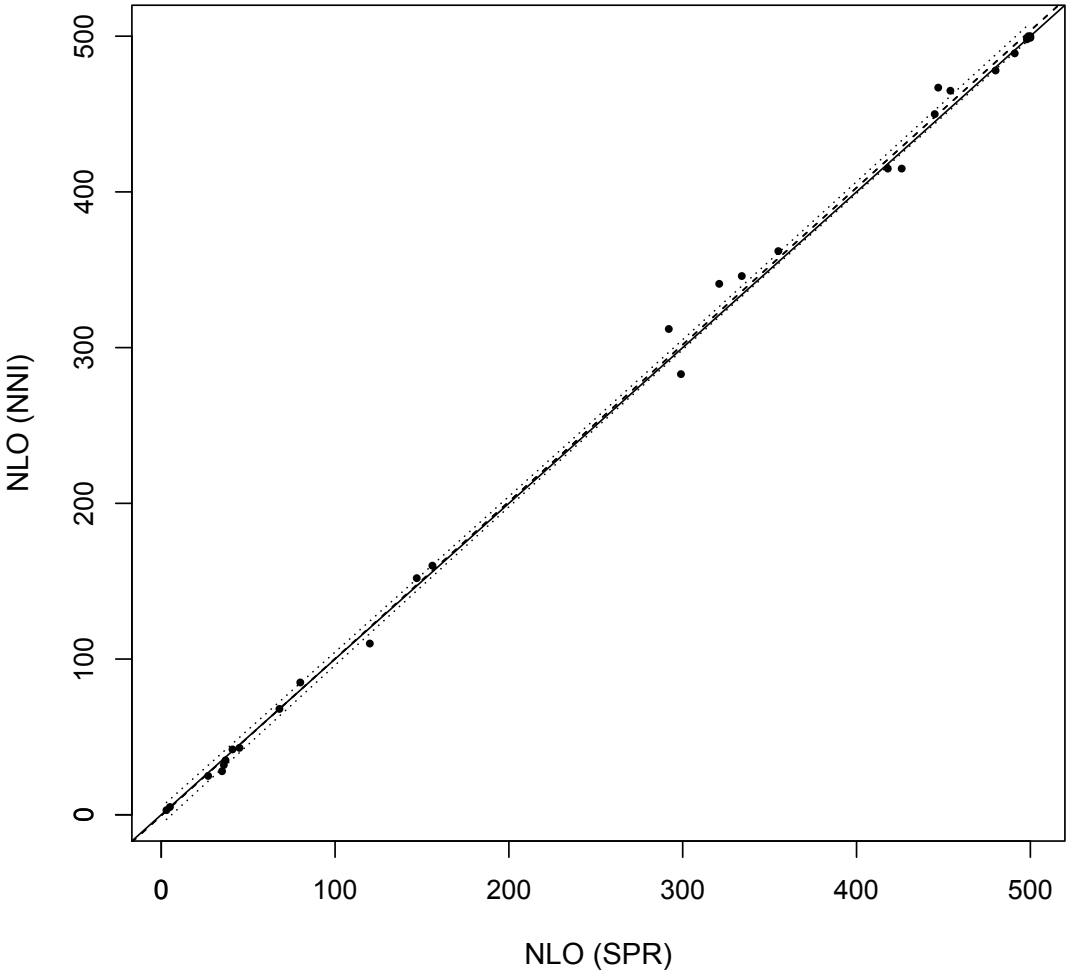

Figure 3.3: NLO in SPR-based landscapes versus NNI-based landscapes. The solid line is the diagonal line $y = x$, the dashed line is the linear regression line, and dotted lines show the regression confidence interval. The plot includes all larger empirical phylogenetic landscapes.

3.6.4 WANUM

Figure 3.7 shows the WANUM for SPR-based landscapes. The upper bound for WANUM in SPR-based nine-taxon landscapes in Chapter 2 was 1.6 and it is interesting that more than 75 percent of datasets in Figure 3.7 have values lower than 4. (based on the third quartile in Figure 3.7).

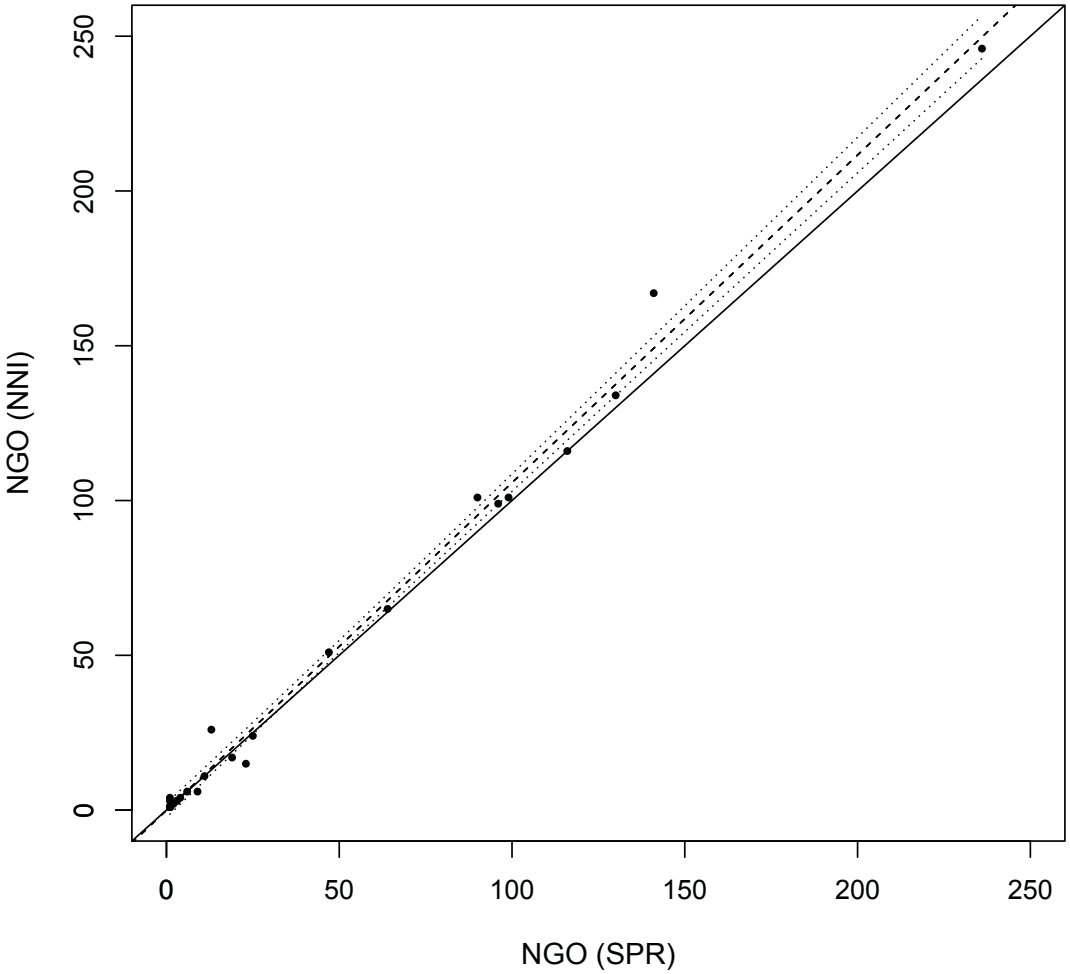

Figure 3.4: NGO in SPR-based landscapes versus NNI-based landscapes. The solid line is the diagonal line $y = x$, the dashed line is the linear regression line, and the dotted lines show the regression confidence interval. The plot includes all larger empirical phylogenetic landscapes.

3.6.5 Stability Plots

In order to determine number of random starting trees sufficient to obtain good approximations of the different ruggedness measures, I uniformly sampled 10000 trees, and sample subsets of increasing size from these trees, starting with 10 trees. Each subset is chosen to be a superset of the previous, smaller subsets. For each subset, I apply hill climbing to obtain estimates of the different ruggedness measures.

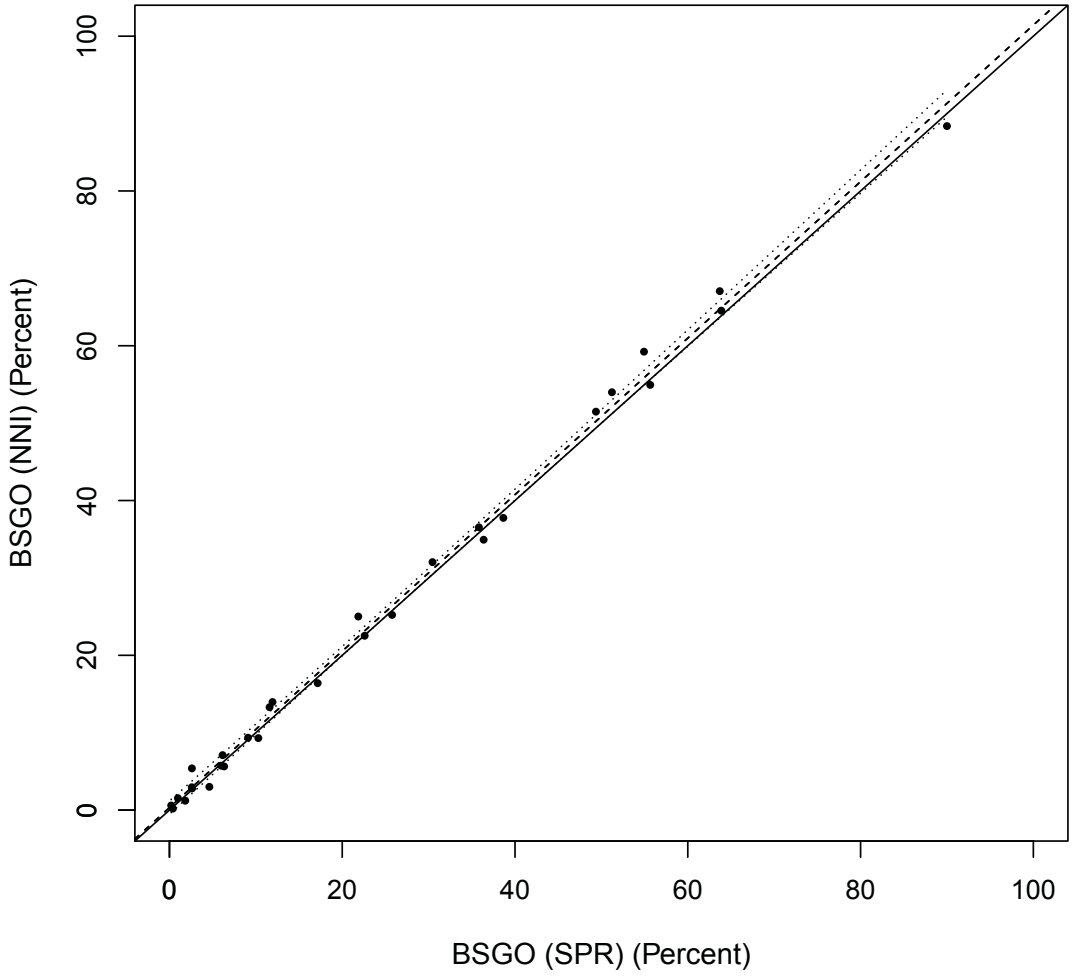

Figure 3.5: BSGO in SPR-based landscapes versus NNI-based landscapes. The solid line is the diagonal line $y = x$, the dashed line is the linear regression line, and the dotted lines show the regression confidence interval. The plot includes all larger empirical phylogeny landscapes.

I use 4 multiple sequence alignments in this chapter containing 19 to 23 taxonomic units to build SPR-based phylogeny landscapes, and explore them using the above strategy. For each chosen subset, I calculate the NLO, BSGO, SB-NUM, and WANUM. Figures 3.8, 3.9, 3.10, and 3.11 plot these ruggedness measures as functions of the number of sampled trees for 4 different SPR landscapes.

In Figure 3.8, NLO tends to increasingly change, but in Figures 3.9, 3.10, and

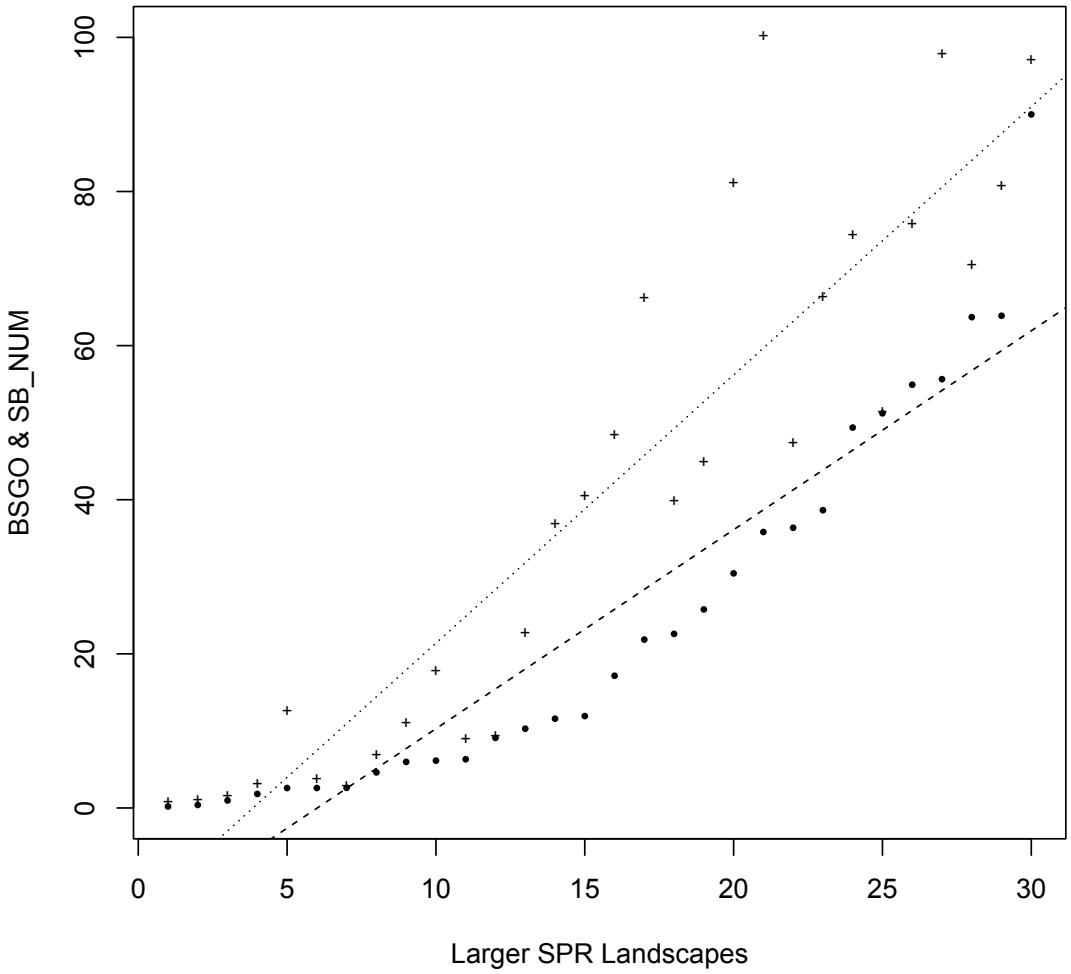

Figure 3.6: SB-NUM and BSGO in larger datasets (SPR) (both SB-NUM and BSGO values are sorted increasingly based on BSGO). The plus signs show the SB-NUM. Solid circles represent the BSGO. The dotted line is the linear regression line for the $SBNUM$ and the dashed line is the linear regression line for BSGO.

3.11 NLOs have lower variations, and also lower values. However, irregular changes of NLOs in all these figures cannot guarantee that one can obtain a good approximate of NLO by exploring the landscape with a specific number of uniformly sampled starting trees.

In Figures 3.8 and 3.9, there are lower variations in BSGOs and SB-NUMs after about the 2000th exploration, and in Figures 3.10 and 3.11 BSGOs and SB-NUMs

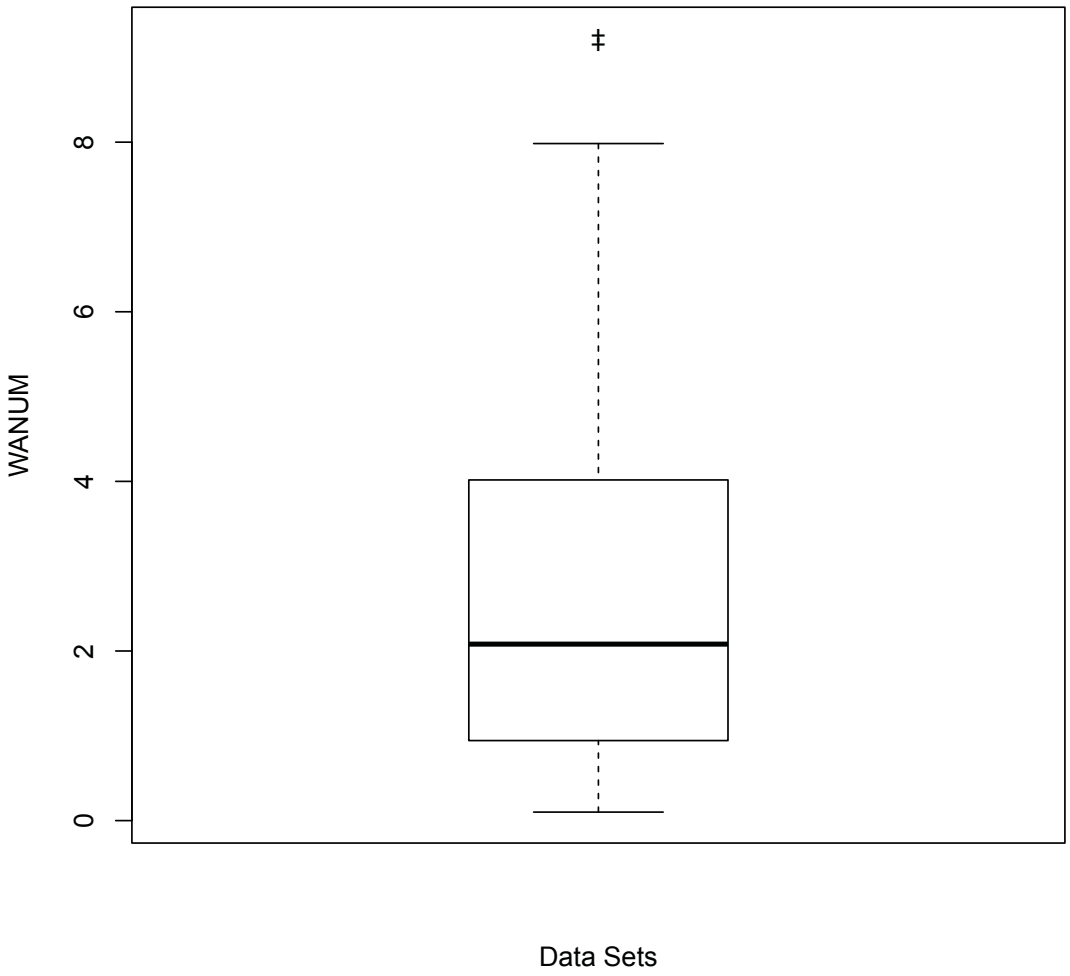

Figure 3.7: WANUM in larger SPR-based landscapes.

tend not to have irregular changes after about the 1000th exploration compared to changes in previous explorations. WANUMs in all these figures tend to have a lower variation after about the 500th exploration of the landscapes using uniformly random starting trees compared to irregular changes in the previous explorations.

In Figure 3.10, around the 4000th exploration, hill climbing does not detect any new local optima (in other words, the NLO measure remains constant), but there is variation in the estimate of the values of WANUM. Based on the definition of WANUM, if there is not any new local optimum, the WANUM remains unchanged. In

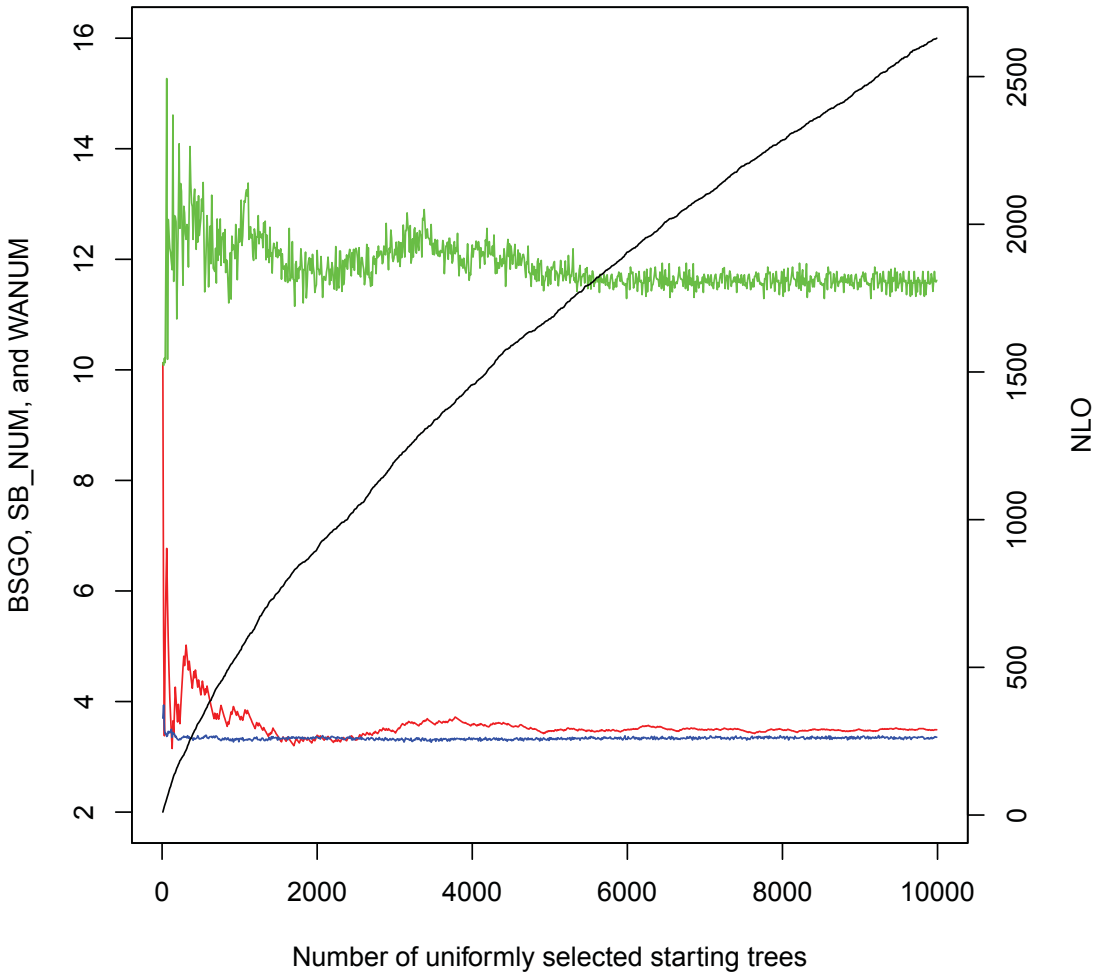

Figure 3.8: Estimates of the different ruggedness measures from an increasing numbers of sample trees in an SPR-based phylogeny landscape. The NLO is shown in black, the BSGO is shown in red, the SB-NUM is shown in green, and the WANUM is shown in blue. I used a multiple sequence alignment containing 23 taxonomic units from this chapter to build the SPR-based phylogeny landscape.

order to explain this variation, I need to point out the fact that MAF given by rSPR package [43] for calculating the path between two trees changes per each different run of the package. This causes to infer different paths and consequently different NUM. Since I use the NUM to evaluate WANUM, I observe the irregular changes in WANUMs.

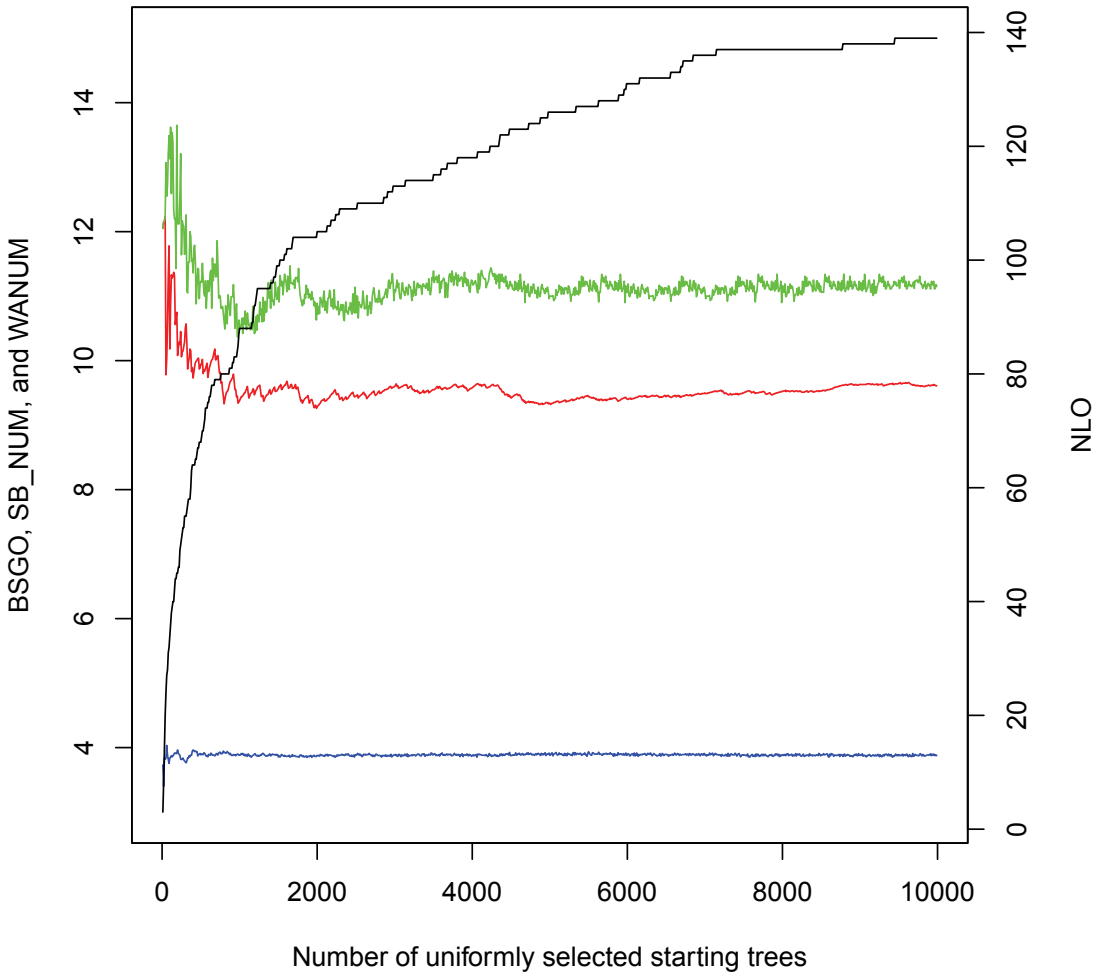

Figure 3.9: Estimates of the different ruggedness measures from an increasing numbers of sample trees in an SPR-based phylogeny landscape. The NLO is shown in black, the BSGO is shown in red, the SB-NUM is shown in green, and the WANUM is shown in blue. I used a multiple sequence alignment containing 19 taxonomic units from this chapter to build the SPR-based phylogeny landscape.

3.7 Conclusion

I examined a variety of empirical datasets of sizes from $n = 14$ to $n = 46$, different phylogenetic tree rearrangement, parsimony optimality criterion, and a sampling strategy to explore the larger phylogeny landscapes. I used different ruggedness measures to estimate the shape of large phylogenies and compare search spaces

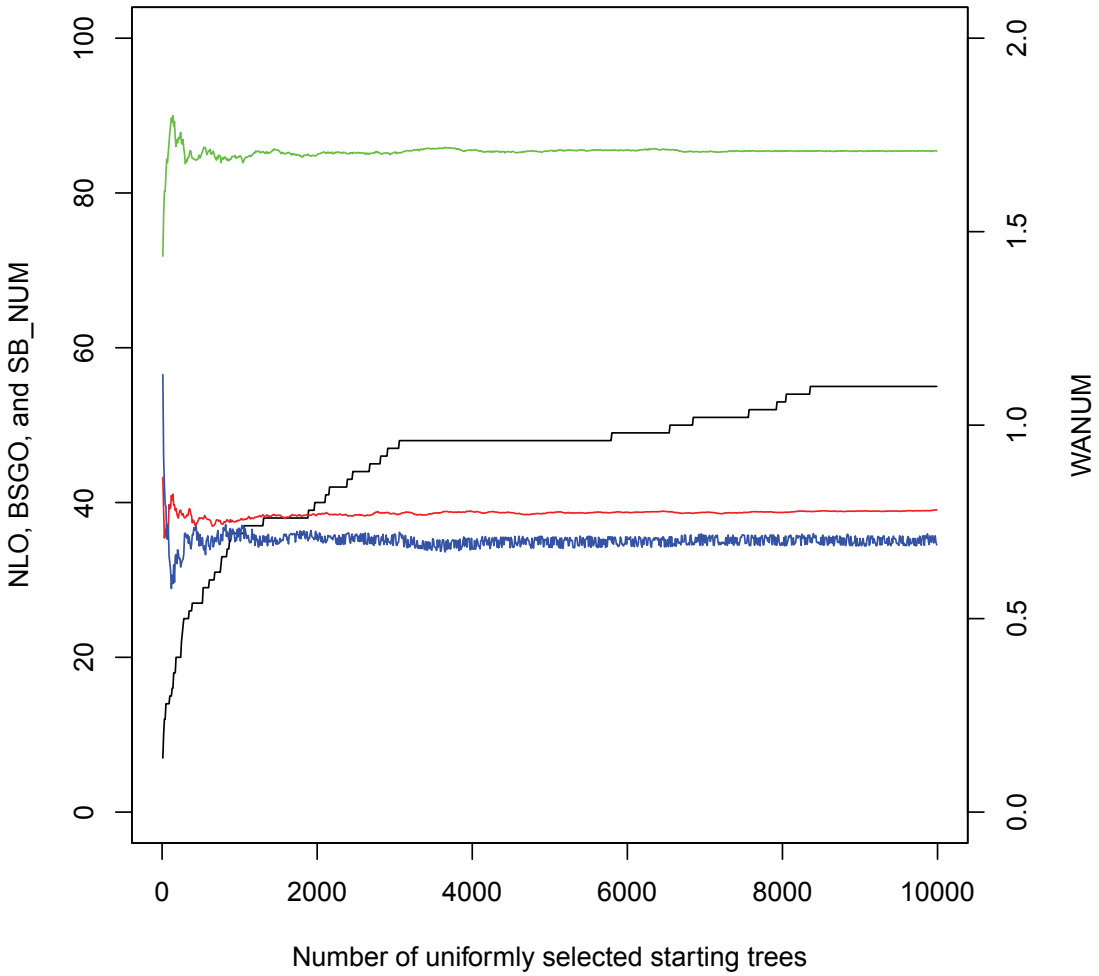

Figure 3.10: Estimates of the different ruggedness measures from an increasing numbers of sample trees in an SPR-based phylogeny landscape. The NLO is shown in black, the BSGO is shown in red, the SB-NUM is shown in green, and the WANUM is shown in blue. I used a multiple sequence alignment containing 22 taxonomic units from this chapter to build the SPR-based phylogeny landscape.

attributes.

The main conclusions of this chapter are: (i) There is a strong correlation between likelihood and parsimony optimality criteria. (ii) Based on the NLO and BSGO, there is a correlation between SPR- and NNI-based landscapes. (iii) As in Chapter 2, using one unfavourable step away from local optima increases the chance of finding global optimum tree in a phylogenetic landscape.

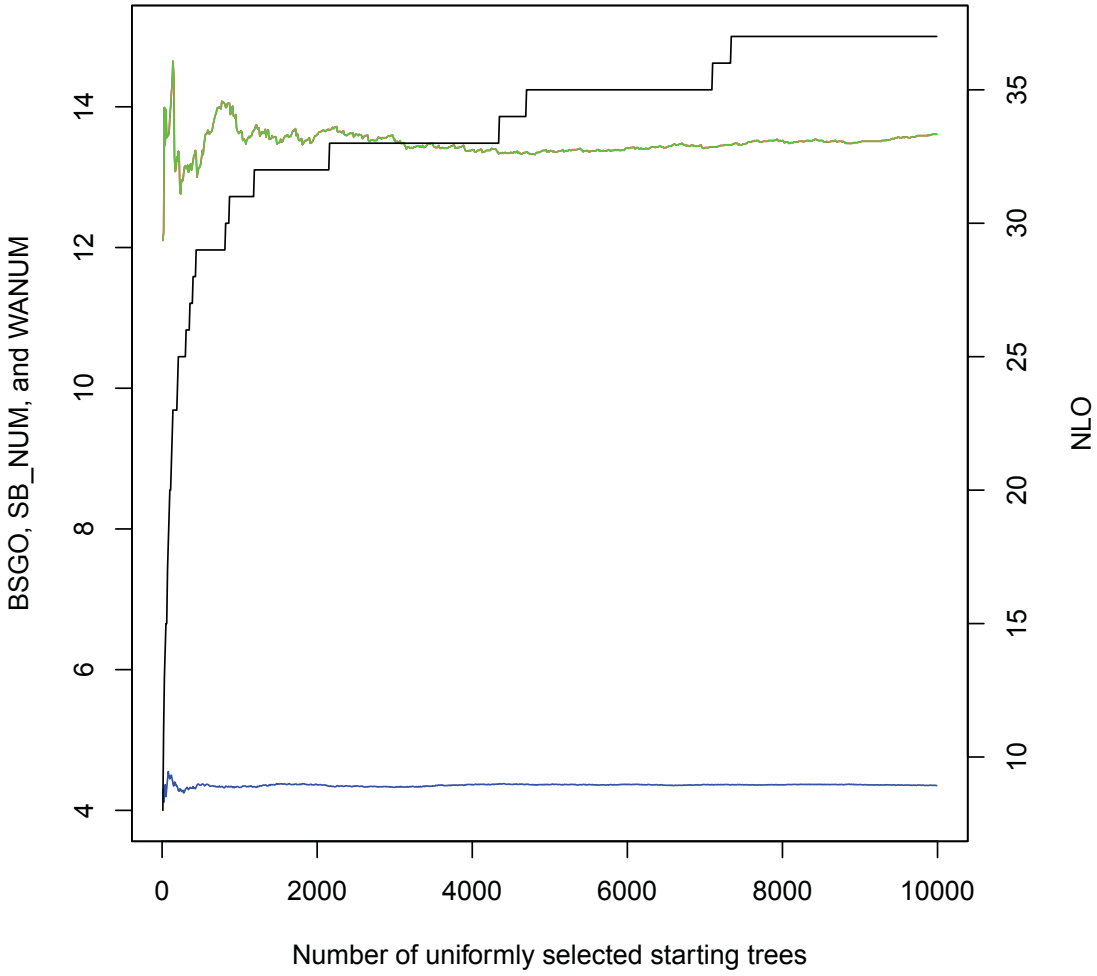

Figure 3.11: Estimates of the different ruggedness measures from an increasing numbers of sample trees in an SPR-based phylogeny landscape. The NLO is shown in black, the BSGO is shown in red, the SB-NUM is shown in green, and the WANUM is shown in blue. I used a multiple sequence alignment containing 23 taxonomic units from this chapter to build the SPR-based phylogeny landscape.

Chapter 4

Randomized Algorithms for Phylogeny Inference

4.1 Introduction

In this chapter, I propose two randomized algorithms to address the phylogeny inference problem. The algorithms are based on the BSGO and SB-NUM metrics. These two measures represent the probabilities of finding the global optimum in the search space starting from a uniform random phylogenetic tree and using hill climbing and hill climbing combined with one unfavourable move, respectively. I compare two algorithms in the SPR-based and NNI-based landscapes by computing the time complexity and probability of finding the global optimum. Afterwards, I apply the proposed randomized algorithms on a variety of amino acid multiple sequence alignments.

4.2 Estimation of Hill Climbing Time Complexity

In this section, I estimate the time complexity of hill climbing in both SPR and NNI landscapes. I examine the number of steps taken by hill climbing to converge to local optima in nine-taxon datasets in Chapter 2 as well as larger datasets in Chapter 3. Let n be the number of taxonomic units in phylogenetic trees in a landscape. In all these datasets, hill climbing takes at most $2n$ steps to discover a local optimum for both NNI and SPR landscapes. In most datasets the number of steps is less than n. Therefore, I estimate the upper bound for moves taken by hill climbing in both SPR and NNI landscapes to be $O(n)$ based on empirical observations.

Based on the fact that each tree in the SPR landscape has $O(n^2)$ neighbours, and given that hill climbing requires $O(n)$ moves to converge to a local optimum, I estimate the time complexity of hill climbing in the SPR landscape to be $O(n^3)$.

There are $O(n)$ neighbours for each tree in the NNI search space, and since the hill climbing needs $O(n)$ moves to reach to a local optimum, I can estimate the complexity

of hill climbing to be $O(n^2)$ in the landscape based on NNI tree rearrangement.

It should be mentioned that since I discovered plateaus in the phylogeny search space during the characterization of the landscape, I defined hill climbing in Chapter 2 in such a way that if it reaches a tree in a plateau, this tree is identified as a local optimum. Therefore, hill climbing never performs further search on plateaus.

4.3 The Randomized Algorithms

To address the phylogeny inference problem, I propose two different randomized algorithms based on the probability of finding the global optimum in the phylogeny landscape. The first algorithm uses the concept of BSGO, which represents the probability of converging to the global optimum using uniformly random starting trees. The second algorithm uses the SB-NUM to tackle the phylogeny inference problem. It should be mentioned that the SB-NUM is always equal to or greater than the BSGO.

I use $Prob(FGO)$ instead of probability of finding the global optima in the landscape, and also $Prob(FGO - SPR)$ and $Prob(FGO - NNI)$ instead of probability of finding the global optima in the SPR and NNI landscapes, respectively.

4.3.1 Algorithm I

The first algorithm is a randomized algorithm based on the BSGO ruggedness measure, which is the probability of converging to the global optimum using hill climbing from a random starting tree. The algorithm starts with generating k uniformly random phylogenetic trees and uses hill climbing strategy to explore the landscape from these trees. Pseudocode 2 shows the algorithm. Equation 4.1 represents the probability of finding the global optimum using algorithm I with k uniform random starting trees.

$$Prob(FGO) = 1 - (1 - BSGO)^k \tag{4.1}$$

The algorithm performs hill climbing for each individual tree. The time complexity of hill climbing in the SPR landscape is $O(n^3)$ and in the NNI landscape is $O(n^2)$. This implies that the time complexities of the algorithm in the SPR and NNI landscapes are $O(kn^3)$ and $O(kn^2)$, respectively.

Pseudocode 2 A randomized algorithm for phylogeny inference

Require: n : number of taxa in trees

$LocalOptimaList \leftarrow null$

Generate k uniform random phylogenetic trees $(Tree_1, \ldots, Tree_k)$

for $i \leftarrow 1$ to k **do**

Performing hill climbing starting with $Tree_i$

Add the local optimum found by hill climbing to $LocalOptimaList$

end for

$GlobalOptima \leftarrow Max(LocalOptimaList)$

4.3.2 Algorithm II

Now consider the SB-NUM. As I observed in previous chapters, taking one unfavourable step out of a local optimum and then returning to hill climbing can increase the chance of discovering the global optimum as reflected by the difference between SB-NUM and BSGO for the landscape in question. I use this observation to propose the second algorithm.

The algorithm produces k uniform random trees and uses hill climbing to explore the phylogeny landscape. After finding each local optimum, a hill climbing search is performed for each tree in the neighbourhood of the local optimum. Pseudocode 3 shows this algorithm. Equation 4.2 represents the probability of finding the global optimum with k uniform random starting trees using Algorithm II.

$$Prob(FGO) = 1 - (1 - SB_{NUM})^k \tag{4.2}$$

The algorithm first uses k uniform random starting trees, and performs hill climbing for each individual tree. The time complexity of this part is $O(kn^3)$. Afterwards, for each neighbour of each local optimum that is found, the algorithm explores the landscape with hill climbing. Since there are $O(n^2)$ neighbours for each tree in the SPR landscape, and the algorithm repeats this part k times, the complexity of this part is $O(kn^2 \cdot n^3) = O(kn^5)$. Therefore, the time complexity of the algorithm is $O(kn^5)$ in an SPR landscape.

I now want to estimate the time complexity of the algorithm for the NNI landscape. Based on the fact that there are k uniform random starting trees, and the algorithm

Pseudocode 3 A randomized algorithm for phylogeny inference

Require: n : number of taxa in trees

```
LocalOptimaList ← null
Generate k uniform random phylogenetic trees (Tree_1, ..., Tree_k)
for i ← 1 to k do
    Performing hill climbing starting with Tree_i
    Add the local optimum found by hill climbing to LocalOptimaList
    for j ← 1 to size of neighbourhood of the local optimum do
        Performing hill climbing starting with Tree_j
        Add the local optimum found by hill climbing to LocalOptimaList
    end for
end for
GlobalOptima ← Max(LocalOptimaList)
```

performs hill climbing for every individual tree, and since the complexity of hill climbing in the NNI landscape is $O(n^2)$, the complexity of the first part of the algorithm is $O(kn^2)$. The algorithms continues with performing hill climbing for every neighbour of each individual local optimum. Since there are $O(n)$ neighbours for each tree in the NNI landscape, and I repeat the second part once for each of the k local optima I found, the complexity of this part is $O(kn^3)$. Thus, the time complexity of the algorithm is $O(kn^3)$ in an NNI landscape.

4.3.3 Comparing the Probabilities of Finding the Global Optimum using Algorithms I and II

In this section, I compare the two randomized algorithms in terms of their probabilities of finding the global optimum. Suppose I choose the k starting trees in Algorithm I and k' starting trees in Algorithm II.

It should be noted that, I only consider SPR-based landscapes because I cannot calculate the SB-NUM in the NNI landscapes (since I am not able to generate the NNI shortest paths using rSPR package).

The worst BSGO observed for the SPR landscapes of all larger empirical datasets was 0.002 (0.2 percent), and the SB-NUM for the same dataset is 0.1129 (11.29 percent).

Using k starting trees, Algorithm I finds the global optimum with probability

$$Prob(FGO) = 1 - (1 - 0.002)^k. \tag{4.3}$$

Aiming for a success probability of 99 percent or larger using Algorithm I, thus requires $k \geq 2098$ starting trees.

Using k' starting tree, Algorithm II finds the global optimum with probability

$$Prob(FGO) = 1 - (1 - 0.1129)^{k'}. \tag{4.4}$$

Aiming for a success probability of 99 percent or larger using Algorithm II, thus requires $k' \geq 36$ starting trees.

I now aim to perform the average case analysis on both algorithms. The averages of BSGO and SB-NUM in all larger empirical datasets are 0.2448 (24.48 percent) and 0.3341 (33.41 percent), respectively. Using equations 4.1 and 4.2, Algorithms I and II converge to the global optimum with 99 percent of accuracy for $k \geq 15$ and $k' \geq 11$, respectively.

The estimated time complexities of Algorithms I and II in SPR landscapes are $O(kn^3)$, and $O(k'n^5)$, respectively. I observed the $k \geq 2048$, and $k' \geq 36$ starting trees suffice for Algorithms I and II find the global optimum with 99 percent of accuracy given the worst observed BSGO and SB-NUM values in my experiments. This implies that Algorithm II is faster only for landscapes with $n \leq 7$. For larger landscapes, Algorithm I is faster. In the average case, Algorithm I is always faster than Algorithm II in finding the global optimum with probability of 99 percent or larger.

4.3.4 Comparing the Probabilities of Finding the Global Optimum in the SPR Landscape versus the NNI Landscape

I now aim to compare the probabilities of finding the global optima in SPR landscapes versus NNI landscapes using the presented randomized algorithms.

I previously estimated the time complexities of Algorithms I and II in both SPR and NNI search spaces, and I demonstrated that applying the randomized algorithms in NNI landscapes is less time-consuming. I use Algorithm I for the comparison, since I cannot compute the SB-NUM in NNI-based search spaces.

Let's suppose k to be the number of starting trees in the SPR landscape and k' to be the number of starting trees in the NNI search space. I aim to find the minimum values of k and k' to guarantee a 99 percent probability of finding the global optimum.

The worst BSGO value observed in the larger NNI landscapes is 0.002 (0.2 percent). The BSGO of the corresponding SPR landscape is 0.004 (0.4 percent). I now aim to calculate the number of starting trees k and k' so that the global optimum is found with 99 percent probability in the SPR and NNI landscapes, respectively.

$$Prob(FGO - SPR) = 1 - (1 - 0.004)^k. \tag{4.5}$$

Thus, $k \geq 1048$ trees suffice to ensure the global optimum is found with probability of at least 99 percent in the SPR landscape.

$$Prob(FGO - NNI) = 1 - (1 - 0.002)^{k'}. \tag{4.6}$$

Thus, $k' \geq 2098$ trees suffice to ensure the global optimum is found with probability of at least 99 percent in the NNI landscape.

As I discussed, the time complexities of Algorithm I in the SPR and NNI landscapes are $O(kn^3)$ and $O(k'n^2)$, respectively. In this section, I also observed that for $k \geq 1048$ and $k' \geq 2098$, Algorithm I finds the global optimum with 99 percent probability these landscapes respectively. Therefore, Algorithm I is faster in finding the global optimum with high probability on NNI landscapes only for $n \geq 2098$.

It should be mentioned that the averages of BSGO in larger SPR and NNI landscapes are 0.2448 (24.48 percent) and 0.2507 (25.07 percent), respectively. Thus, on average, Algorithm I is just as likely to find the global optimum using a given number of starting trees in the both types of the landscapes and it is significantly faster on the NNI landscapes.

4.4 Applying Algorithms I and II on Different Datasets

I now aim to apply Algorithms I and II on different amico acid datasets. I compare their results in the SPR landscape and the NNI landscape. I also compare one result to PhyML and RAxML.

4.4.1 Datasets

I use the datasets from Chapter 3 to compare Algorithm I with Algorithm II in both SPR and NNI landscapes using parsimony optimality criterion.

4.4.2 Results

I apply Algorithms I and II on 30 amino acid multiple sequence alignments scored with parsimony optimality criterion using both SPR and NNI tree rearrangements. I generate 500 different uniform random starting trees to explore the landscapes. In 29 cases, Algorithms I and II converged to a same global optimum (the minimum parsimony tree discovered) in both SPR and NNI landscapes. In only one of the datasets, Algorithm II (in both SPR and NNI landscapes), and Algorithm I (in the SPR search space) discovered a better phylogenetic tree comparing to Algorithm I applying in the NNI landscape.

4.5 Conclusion

In this chapter, I proposed two randomized algorithms based on the concept of the BSGO and SB-NUM ruggedness measures. I compared the algorithms based on time-complexity and probability of finding the global optimum in both SPR and NNI landscapes. Afterwards, I applied the algorithms on different multiple sequence alignments and compared their results with each other and PhyML and RAxML. The main conclusion for this chapter are (i) Applying both Algorithms I and II in the NNI-based landscapes is less time consuming. (ii) To address the phylogeny inference problem, selecting one of the proposed randomized algorithms, and choosing the SPR or the NNI search space for explore the phylogenetic landscape depends on the input size. (iii) In the average case, Algorithm I is always faster than Algorithm II in finding the global optimum with probability of 99 percent or larger. (iv) On average, Algorithm I is just as likely to find the global optimum using a given number of starting trees in the both SPR and NNI landscapes and it is significantly faster on the NNI landscapes.

Chapter 5

Conclusion

The goal of this thesis was to develop a better understanding of the structure of the phylogeny search space by characterizing the landscape ruggedness employing different measures. I examined a wide range of nine-taxon and larger phylogenetic landscapes using SPR and NNI tree rearrangements as well as maximum likelihood and parsimony optimality criteria. Analysing both nine-taxon and larger landscapes provided insight into the shape of the phylogeny landscape,and highlighted differences between the phylogeny landscapes created using different types of evolutionary models and tree rearrangements. Based on these insights, I proposed two randomized algorithms to solve the phylogeny inference problem.

5.1 Contribution

I first exhaustively built a variety of nine-taxon phylogenetic landscapes using different empirical and simulated datasets, evolutionary models, and also SPR and NNI tree rearrangements. I used maximum likelihood optimality criterion to score the trees in the landscapes. Afterwards, I studied the structure of the landscapes using exhaustive exploration by hill climbing and compared the landscape attributes, such as different evolutionary models and different tree rearrangements, employing different ruggedness measures. Based on all ruggedness measures, SPR-based landscapes are often less rugged than NNI-based landscapes. Using either LG or WAG protein models to build the phylogeny landscape yielded identical landscapes in terms of ruggedness. Taking one unfavourable step away from local optima considerably increased the chance of discovering the optimal tree in the phylogenetic landscape.

In order to study larger phylogeny search spaces, I used a sampling strategy to explore larger phylogenetic landscapes, since these landscapes are too big to explore exhaustively. I built the larger landscapes using 30 empirical datasets of size from $n = 14$ to $n = 46$, and also used SPR and NNI tree rearrangements to define the

neighbourhood of each tree in the landscapes. I demonstrated that there is a strong correlation between maximum likelihood and parsimony optimality criteria, and then used parsimony optimality criterion instead of maximum likelihood to score more trees in the larger landscapes. I showed that there is a correlation between SPR- and NNI-based landscapes based on NLO and BSGO ruggedness measures. In order to find the number of samples sufficient to obtain good estimates of the ruggedness measures, I used a large number of samples, and calculated the ruggedness measures for subsets of these samples of increasing size. The chance of finding the optimal tree in the phylogeny landscapes noticeably increased by taking one unfavourable step away from local optima.

Finally, based on the knowledge provided by ruggedness measures, I proposed two randomized algorithms to find the optimal tree in the phylogeny search space. I applied these algorithms on different multiple sequence alignments as well as SPR and NNI tree rearrangements, and compared their results. I discussed the estimated time complexities of these algorithms and also compared their probabilities to find the optimal tree on SPR- and NNI-based phylogeny search spaces. I demonstrated that the choice of the algorithm and of the type of the landscape to be explored in order to find the optimal tree with high probability depends on the input size. I showed that applying both Algorithms I and II in the NNI-based landscapes is less time consuming, and, on average, Algorithm I is faster than Algorithm II in finding the global optimum with probability of 99 percent or larger.

5.2 Future Work

I describe future work for this thesis as

- Using TBR tree rearrangement to create the phylogeny landscape, and comparing the structure of the landscape with SPR- and NNI-based landscapes.

- Considering the maximum likelihood based optimality criterion in [38], in which it only optimizes the three branches connected to the regraft node during the tree rearrangement; Using this optimality criterion instead of maximum likelihood to characterize the phylogeny landscape.

- Computing the time complexity of hill climbing in SPR and NNI landscapes by finding the upper bound for the length of the long path in these landscapes.
- Using parallel computing techniques to speed up the proposed randomized algorithms.
- To reduce the time complexity of the randomized algorithms, design a new tree rearrangement with a linear number of permutation on trees, but time complexity of hill climbing in the landscape created by new tree rearrangement being less than the NNI-based landscape.
- Considering to uniformly sample trees up to a specific distance from Neighbour Joining tree, and use that to feed the proposed randomized algorithms instead of uniform random sampling of trees.

Bibliography

[1] Michael D. Atkinson and J-R. Sack. Generating binary trees at random. *Information Processing Letters*, 41(1):21–23, 1992.

[2] Alex Bateman, Lachlan Coin, Robert D. Finn, Volker Hollich, Sam Griffiths-Jones, and Ajay Khanna. The PFAM protein families database. *Nucleic Acids Research*, 32(suppl 1):D138–D141, 2004.

[3] Alan Joseph J. Caceres, Samantha Daley, John DeJesus, Michael Hintze, Diquan Moore, and Katherine St John. Walks in phylogenetic tree space. *Information Processing Letters*, 111(12):600–604, 2011.

[4] Salvador Capella-Gutiérrez, José M. Silla-Martínez, and Toni Gabaldón. trimAl: A tool for automated alignment trimming in large-scale phylogenetic analyses. *Bioinformatics*, 25(15):1972–1973, 2009.

[5] Benny Chor and Tamir Tuller. Maximum likelihood of evolutionary trees is hard. *Bioinformatics*, 21(suppl 1):i97–i106, 2005.

[6] Alexis Criscuolo and Simonetta Gribaldo. BMGE (Block Mapping and Gathering with Entropy): A new software for selection of phylogenetic informative regions from multiple sequence alignments. *BMC Evolutionary Biology*, 10(1):210, 2010.

[7] Jeffrey HF Cullis. A Framework for the Construction, Visualization, and Characterization of Phylogeny Search Space. Master's thesis, Dalhousie University, 2008.

[8] Charles Darwin. *On the Origin of the Species by Natural Selection*. Murray, London, 1859.

[9] David Eppstein. Finding the k shortest paths. *SIAM Journal on Computing*, 28(2):652–673, 1998.

[10] Joseph Felsenstein. *Inferring Phylogenies*. Palgrave Macmillan, 2004.

[11] George W. Furnas. The generation of random, binary unordered trees. *Journal of Classification*, 1(1):187–233, 1984.

[12] Stephane Guindon and Olivier Gascuel. A simple and fast and accurate algorithm to estimate large phylogenies by maximum likelihood. *Systematic Biology*, 52(5):696–704, 2003.

[13] John Hershberger, Matthew Maxel, and Subhash Suri. Finding the k shortest simple paths: A new algorithm and its implementation. *ACM Transactions on Algorithms*, 3(4):45, 2007.

[14] Jaime Huerta-Cepas, Anibal Bueno, Joaquín Dopazo, and Toni Gabaldón. PhylomeDB: A database for genome-wide collections of gene phylogenies. *Nucleic Acids Research*, 36(suppl 1):D491–D496, 2008.

[15] Jaime Huerta-Cepas, Salvador Capella-Gutierrez, Leszek P. Pryszcz, Ivan Denisov, Diego Kormes, Marina Marcet-Houben, and Toni Gabaldón. PhylomeDB v3.0: An expanding repository of genome-wide collections of trees, alignments and phylogeny-based orthology and paralogy predictions. *Nucleic Acids Research*, 39(suppl 1):D556–D560, 2011.

[16] Jaime Huerta-Cepas, Salvador Capella-Gutierrez, Leszek P. Pryszcz, Marina Marcet-Houben, and Toni Gabaldón. PhylomeDB v4: zooming into the plurality of evolutionary histories of a genome. *Nucleic Acids Research*, page gkt1177, 2013.

[17] Víctor M. Jiménez and Andrés Marzal. Computing the k shortest paths: A new algorithm and an experimental comparison. *In Algorithm Engineering*, Springer Berlin Heidelberg:15–29, 1999.

[18] Richard M. Karp. An introduction to randomized algorithms. *Discrete Applied Mathematics*, 34(1):165–201, 1991.

[19] Si Quang Le and Olivier Gascuel. An improved general amino acid replacement matrix. *Molecular Biology and Evolution*, 25(7):1307–1320, 2008.

[20] Alan R. Lemmon and Michel C. Milinkovitch. The metapopulation genetic algorithm: An efficient solution for the problem of large phylogeny estimation. *Proceedings of the National Academy of Sciences*, (16):10516–10521, 2002.

[21] Kazmier Leonard. *Schaum's Outline of Business Statistics*. McGraw Hill Professional, 2003.

[22] Paul O. Lewis. A genetic algorithm for maximum-likelihood phylogeny inference using nucleotide sequence data. *Molecular Biology and Evolution*, 15(3):277–283, 1998.

[23] Michael Mitzenmacher and Eli Upfal. *Probability and computing: Randomized algorithms and probabilistic analysis*. Cambridge University Press, 2005.

[24] Erkki Mokinen. Generating random binary trees: a survey. *Information Sciences*, 115(1):123–136, 1999.

[25] Daniel Money and Simon Whelan. Characterizing the phylogenetic tree-search problem. *Systematic Biology*, 61(2):228–239, 2012.

[26] Rajeev Motwani and Prabhakar Raghavan. *Randomized Algorithms*. Chapman and Hall/CRC, 2010.

[27] Gary Olsen, Hideo Matsuda, and Ross Overbeek. fastDNAml: A tool for construction of phylogenetic trees of DNA sequences using maximum likelihood. *Computer Applications in the Biosciences*, 10(1):41–48, 1994.

[28] Shirley Pepke, Davin Butt, Isabelle Nadeau, Andrew J. Roger, and Christian Blouin. Using Confidence Set Heuristics During Topology Search Improves the Robustness of Phylogenetic Inference. *Journal of Molecular Evolution*, 64(1):80–89, 2007.

[29] D. O. T. R. E. E. Plottree and D. O. T. G. R. A. M. Plotgram. PHYLIP-phylogeny inference package (version 3.2). *Cladistics*, (5):163–166, 1989.

[30] Morgan N. Price, Paramvir S. Dehal, and Adam P. Arkin. FastTree: computing large minimum evolution trees with profiles instead of a distance matrix. *Molecular Biology and Evolution*, (7):1641–1650, 2009.

[31] Morgan N. Price, Paramvir S. Dehal, and Adam P. Arkin. FastTree 2approximately maximum-likelihood trees for large alignments. *PloS One*, (3):e9490, 2010.

[32] Andrew Rambaut and Nicholas C. Grass. Seq-Gen: an application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. *Computer Applications in the Biosciences*, 13(3):235–238, 1997.

[33] Jean-Luc Rémy. Un procédé itératif de dénombrement dárbres binaires et son application à leur génération aléatoire. *RAIRO Inform. Théor*, 19(2):179195, 1985.

[34] Fredrik Ronquist and John P. Huelsenbeck. MrBayes 3: Bayesian phylogenetic inference under mixed models. *Bioinformatics*, (12):1572–1574, 2003.

[35] Hidetoshi Shimodaira. An approximately unbiased test of phylogenetic tree selection. *Systematic Biology*, 51(3):492–508, 2002.

[36] Hidetoshi Shimodaira and Masami Hasegawa. CONSEL: for assessing the confidence of phylogenetic tree selection. *Bioinformatics*, 17:1246–1247, 2001.

[37] Douglas E. Soltis and Pamela S. Soltis. The role of phylogenetics in comparative genetics. *Plant Physiology*, 132(4):1790–1800, 2003.

[38] Alexandros Stamatakis, Thomas Ludwig, and Harald Meier. RAxML-III: a fast program for maximum likelihood-based inference of large phylogenetic trees. *Bioinformatics*, 21(4):456–463, 2005.

[39] David L Swofford, Gary J Olsen, Peter J Waddell, and David M Hillis. Chapter 11: Phylogenetic Inference. *Molecular Systematics*, 2, 1996.

[40] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2013. ISBN: 3-900051-07-0.

[41] Chris Tuffley and Mike Steel. Links between maximum likelihood and maximum parsimony under a simple model of site substitution. *Bulletin of Mathematical Biology*, 59(3):581–607, 1997.

[42] Simon Whelan and Nick Goldman. A general empirical model of protein evolution derived from multiple protein families using a maximum-likelihood approach. *Molecular Biology and Evolution*, 18(5):691–699, 2001.

[43] Chris Whidden, Robert Beiko, and Norbert Zeh. Fixed-parameter algorithms for maximum agreement forests. *SIAM Journal on Computing*, 42(4):1431–1466, 2013.

[44] Chris Whidden and Norbert Zeh. *A Unifying View on Approximation and FPT of Agreement Forests*. In: WABI 2009. LNCS, vol. 5724, pp. 390401. Springer-Verlag, 2009.

[45] Chris Whidden, Norbert Zeh, and Robert Beiko. Supertrees based on the subtree prune-and-regraft distance. *Systematic Biology*, 63(4):566–581, 2014.

[46] Philipp Woelfel. *Randomized Algorithms*. University of Calgary, Lecture Notes, 2013.

[47] Ziheng Yang. PAML 4: phylogenetic analysis by maximum likelihood. *Molecular Biology and Evolution*, 24(8):1586–1591, 2007.

Appendix A

Generating Uniform Random Phylogenetic Trees

A.1 The Number of Ordered Binary Trees

As a basis for the correctness proof of the sampling method, it is necessary to determine the number of distinct ordered binary trees with n nodes. This number is known as the nth Catalan number

$$C_n = \binom{2n}{n}\frac{1}{n+1}.$$

Theorem A.1. *There are C_n distinct ordered binary trees with n nodes.*

Proof. First observe that there exists a bijection between ordered binary trees with n nodes and triangulations of a convex polygon with $n+2$ vertices: To obtain the triangulation of a convex polygon P corresponding to an ordered binary tree T, choose one of the edges of the polygon P as the polygon's "root edge" and number the vertices of P as $v_1, v_2, \ldots, v_{n+2}$ in clockwise order, starting at one endpoint of P and ending at the other endpoint. Now let L and R be the left and right subtrees of T, respectively. Add diagonals v_1v_{r+2} and $v_{r+2}v_{n+2}$ to the triangulation, where $r = |R|$. This splits P into two sub-polygons P_l and P_r with root edges v_1v_{r+2} and $v_{r+2}v_{n+2}$, respectively. Construct the final triangulation by applying this procedure recursively to L and P_l, and to R and P_r. It is not hard to see that, given the same root edge, this construction can be reversed and the original tree T can be obtained from the triangulation constructed from T. Thus, this is a bijection.

Given this bijection, it now suffices to count the number of distinct triangulations of a convex polygon with $n+2$ vertices. Let T_n be the number of such triangulations. We claim that $T_n = C_n$. For $n = 1$, there exists exactly one triangulation of the polygon with 3 vertices, that is, $T_n = 1 = C_n$. For $n > 1$, it follows by induction that $T_n = \frac{1}{n+1}\binom{2n}{n} = C_n$ if it can be shown that $T_{n+1} = \frac{4n+2}{n+2}T_n$. To this end consider two types of augmented triangulations of convex polygons. For a convex polygon P with $n+2$ vertices, a fixed root edge r, and a triangulation $\mathfrak{T}$ of P, a *splittable* triangulation

is obtained from $\mathfrak{T}$ by choosing any of its edges, say edge uv, marking uv and giving uv a direction. For a convex polygon P' with $n+3$ vertices, a fixed root edge r, and a triangulation $\mathfrak{T}'$ of P', a *collapsible* triangulation is obtained from $\mathfrak{T}'$ by choosing any of the boundary edges of P and marking it. Each triangulation of P gives rise to $4n+2$ splittable triangulations, and each triangulation of P gives rise to $n+2$ collapsible triangulations. It remains to establish a bijection between these marked triangulations of polygons with n and $n+1$ vertices, as this would imply the desired equality $(4n+2)T_n = (n+2)T_{n+1}$.

For each splittable triangulation $\mathfrak{T}$, let $\phi(\mathfrak{T})$ be the collapsible triangulation defined as follows: Let uv be the marked edge of $\mathfrak{T}$, oriented from u to v. Replace v with two vertices v_1 and v_2, replace edge uv with two edges uv_1 and uv_2, add an edge v_1v_2, and mark this edge. A reverse transformation is obtained by defining $\phi^{-1}(\mathfrak{T}')$ as follows, for each collapsible triangulation $\mathfrak{T}'$. Let v_1v_2 be the marked edge of $\mathfrak{T}'$. Now replace v_1 and v_2 by a single vertex v, remove edge v_1v_2, replace edges uv_1 and uv_2 by a single edge uv, mark uv, and orient uv from u to v. It is not hard to verify that $\phi^{-1}(\phi(\mathfrak{T})) = \mathfrak{T}$, for every splittable triangulation $\mathfrak{T}$, and that $\phi(\phi^{-1}(\mathfrak{T}')) = \mathfrak{T}'$, for every collapsible triangulation $\mathfrak{T}'$. Thus, $\phi(\cdot)$ is a bijection between all splittable triangulations on n vertices and all collapsible triangulations on $n+1$ vertices. □

A.2 Generating Uniformly Random Phylogenetic Trees

Theorem A.2. *There are* $C_{n-1}\frac{n!}{2^{n-1}} = \frac{(2n-2)!}{2^{n-1}\cdot(n-1)!}$ *distinct phylogenies with* n *leaves.*

Proof. Let an *ordered phylogenetic tree* be a tree obtained from a phylogenetic tree by declaring one of the children of each internal node to be the left child, and the other the right child. For a given n-leaf phylogeny, there are exactly 2^{n-1} corresponding ordered phylogenies because there are $n-1$ internal nodes, each of which gives rise to a binary choice which of the two children is the left child. Thus, there are $P_n \cdot 2^{n-1}$ ordered phylogenies, where P_n is the number of n-leaf phylogenies. Now observe that every ordered phylogenetic tree is also defined uniquely by a pair (T, π), where T is an ordered binary tree with $n-1$ nodes and π is a permutation of the elements $1, 2, \ldots, n$: First replace every missing child of a node in T with a new leaf. This makes all nodes in T internal nodes of the resulting tree T', and T' has n leaves. Then

assign the elements of π to the leaves of T' from left to right. It is obvious that each such pair (T, π) defines a unique ordered phylogeny and that each ordered phylogeny is generated by such a pair. Thus, since there are C_{n-1} ordered binary trees with $n-1$ nodes and $n!$ permutations of the elements $1, 2, \ldots, n$, this shows that there are $C_{n-1} \cdot n!$ ordered n-leaf phylogenies. Since the number of ordered n-leaf phylogenies is also equal to $P_n \cdot 2^{n-1}$, this gives $C_{n-1} \cdot n! = P_n \cdot 2^{n-1}$, and the lemma follows. □

Rémy's algorithm [33] for generating a phylogeny with n leaves can be specified as follows. Assume w.l.o.g. that the label set of the tree is $\{1, 2, \ldots, n\}$.

If $n = 1$, return the unique 1-node tree with label set $\{1\}$.

If $n > 1$, recursively generate a uniformly random phylogenetic tree T with label set $\{1, 2, \ldots, n-1\}$. This tree has $2n-3$ nodes. Choose one of these nodes uniformly at random. Let v be the chosen node, and let p be its parent. Then create two new nodes u and w, define $p_u := p$, $p_v := p_w := u$, and label w with the leaf label n.

Theorem A.3. *Rémy's algorithm generates each n-leaf phylogeny with probability $1/P_n$, where $P_n := \frac{(2n-2)!}{2^{n-1} \cdot (n-1)!}$ is the number of distinct phylogenies with n leaves.*

Proof. By induction on n.

For $n = 1$, the unique 1-leaf phylogeny is generated with probability 1 and $P_1 = \frac{0!}{2^0 \cdot 0!} = 1$.

For $n > 1$, consider a particular n-leaf phylogeny T, and let T' be the $(n-1)$-leaf phylogeny obtained by detaching the leaf w with label n and suppressing the resulting internal degree-2 node or degree-1 root node. Let v be w's sibling in T. Observe that the algorithm generates T if and only if its recursive call generates T' and it then chooses v as the sibling of the new node w with label n in T. By the inductive hypothesis, the former happens with probability $1/P_{n-1}$. The latter happens with probability $1/(2n-3)$ because the algorithm has $2n-3$ potential siblings in T' to choose from, and each is chosen with equal probability. Thus, T is generated with probability

$$\frac{1}{P_{n-1}} \cdot \frac{1}{2n-3} = \frac{2^{n-2} \cdot (n-2)!}{(2n-4)!} \cdot \frac{1}{2n-3} = \frac{2^{n-1} \cdot (n-1)!}{(2n-2)!} \cdot \frac{2n-2}{2 \cdot (n-1)} = \frac{2^{n-1} \cdot (n-1)!}{(2n-2)!} = \frac{1}{P_n}$$

□