Phylogenetic Networks: Simulation, Characterization, and Reconstruction

by

Monique Marlene Morin

A.A., Cottey College, 1990

S.B., Massachusetts Institute of Technology, 1992

DISSERTATION

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Doctor of Philosophy

Computer Science

The University of New Mexico

Albuquerque, New Mexico

December 2007

Dedication

A ma famille – les générations passées, présentes, et futures.

Acknowledgments

As this work is the culmination of my graduate career, I would like to extend my thanks to certain individuals who have contributed to this path. First and foremost are my parents, Diane and Frank Morin. From our days of playing blackjack and cribbage, little did we envision a Ph.D. would be in my future. I wish to thank them for teaching me the invaluable skill of typing at a young age and most importantly, instilling in me the determination to accomplish any task to which I set my mind.

My “scientific upbringing” was fostered by many influential individuals. These included: Sid Lycan (“you’re right, there are not a lot of women in science” – chemistry, Palm Desert High School), Leroy Sikes (“you can do whatever you want with math” – Cottey College), John Molitoris (“have you considered applying as a transfer to MIT?” – Lawrence Livermore National Lab), Sy Friedman (logic-man – MIT), Jan Batteux (“a clearance means a low security risk, not a high I.Q.” – Lawrence Livermore National Lab), David Clifton (my first computer-geek role-model – SRA, Virginia), Dana Richards (“you could apply to graduate school now” – George Mason University), members of the CS-Reentry program (University of California at Berkeley), and George Luger, Cris Moore, and Bernard Moret (University of New Mexico).

I was fortunate to have a variety of support throughout my graduate school experience. Thanks goes to fellow lab-mates, Eric Gottlieb and Nick Pattengale, who acted as my practice audience more than once, and Krister Swenson who gave me free reign of his machine whenever I needed it, thanks for literally “being there.” I thank Gabriela Barrantes and Leigh Fanning for saving my sanity with their technical advice on Linux and Macs, along with invaluable words of wisdom on topics ranging from physicists to motherhood. I wish to note the valuable contributions of my committee – David Bader, Stephanie Forrest, Randy Linder, and Bernard Moret, along with Lynne Jacobsen, for making the logistics of this “extended dissertation committee” work.

To my family and friends who have seen me along part or all of this journey, I express “une mille fois mercis.” All probably have been wondering when this phase of my life would be over. They include: Aunt Betty and Uncle Dave, Aunt Jann and Uncle Milton, Uncle Joe, Grandma, Jim and Nancy (and families), Mr. and Mrs. Hansen, Mom and Dad Lanier, Jennifer Blessing, Beryl Castello, Patricia Daughtry, Jed Dennis, Elsie Elster, Ernesto Gonzalez, Emeline Picart, Amy Weaver, and many

SRA friends – may you once again receive correspondence from me that does not include a progress report on my graduate work.

Most importantly, I wish to thank my immediate family for tolerating the saga of my graduate career. I was blessed to have such wonderful support. Generous with her "grandma-time," my mom's babysitting provided me with many daytime work hours during the final months of this effort. To my extraordinary husband Nick, I am indebted for all of his listening and learning about phylogenetic networks – certainly more than he ever wanted, or should have been obliged. Although his contributions were many, the one I will always remember fondly is the many hours he logged driving throughout both of our undergraduate and graduate careers. Finally, I *sign* a big "thank you" to Carina Marie, our amazing daughter who brightened many of my late-night-turned-mornings with her grins and giggles.

Phylogenetic Networks: Simulation, Characterization, and Reconstruction

by

Monique Marlene Morin

ABSTRACT OF DISSERTATION

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Doctor of Philosophy
Computer Science

The University of New Mexico

Albuquerque, New Mexico

December 2007

Phylogenetic Networks: Simulation, Characterization, and Reconstruction

by

Monique Marlene Morin

A.A., Cottey College, 1990

S.B., Massachusetts Institute of Technology, 1992

Ph.D., Computer Science, University of New Mexico, 2007

Abstract

Phylogenetic trees are topological depictions of evolutionary histories that involve birth and death events. This tree model inherently assumes that active lineages evolve independently from all others. However, interspecific events such as hybridization and lateral gene transfer are known to occur, resulting in an intermingling of DNA information. In essence, this creates a bridge between two previously independent lineages. By definition, these processes are fundamentally dependent on other contemporary species and lead to a more complex history, which is often referred to as a phylogenetic network. Although tree topology software is abundant and mature, algorithms and applications that permit and address issues specific to phylogenetic networks are less common and still in their infancy.

The primary contributions of this work include modelling phylogenetic networks and algorithms for their reconstruction and characterization. This involved the development of three new applications: NetGen for simulating source topologies, NetReconstruct for inferring histories, and NetMeasure for quantitatively comparing and categorizing network features. Phylogenetic networks are created by extending the traditional birth-death model to include hybridizations. This approach is unique in that the DNA sequences are evolved in conjunction with the topology, thereby permitting the outcome of hybrid events to be influenced by genetics. By utilizing the sequence information from NetGen's final active lineages, a proposed history containing a single-diploid-hybrid event is inferred. This reconstruction process segments the species into sets for which subtrees are created and then merged to form the final topology. Finally, hybrid events in a topology can be quantitatively characterized with three new measures and two networks can be compared using the existing tripartition method.

Our results show that tripartition scores for this reconstruction model improve when the number of current day species increases and the branch lengths of the topology shorten. Additionally, topologies containing a single ancient hybridization event (one occurring early in time) result in reconstructions more analogous to their source histories as compared to those with a hybrid event having occurred more recently. This work supports the ongoing effort of phylogenetic reconstruction in fields such as pharmacology, genetics, and systematics.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This first chapter imparts the preliminaries for the research presented in this document. Section 1.1 gives a cursory overview of phylogenies and in particular, phylogenetic networks. Scope and related work, which serve to motivate and further the context of this effort, are outlined in Section 1.2. Finally, Section 1.3 contains the overview of this dissertation.

1.1 Phylogenetic Overview

Phylogenies are evolutionary histories inferred and studied by systematists,biologists who study biological diversity [25]. However, the use of phylogenies is not limited to the field of systematics and applications range from studying pathogens (e.g. [16]) to human genetics (e.g. [65]). A sample phylogeny, based upon [80], is annotated in Figure 1.1 and illustrates the well known fact that chimpanzees are closer primate relatives to humans than gorillas.

Traditionally, phylogenies at the interspecific (across species) level, such as the example in Figure 1.1, are accepted to have a tree structure – meaning lineages are

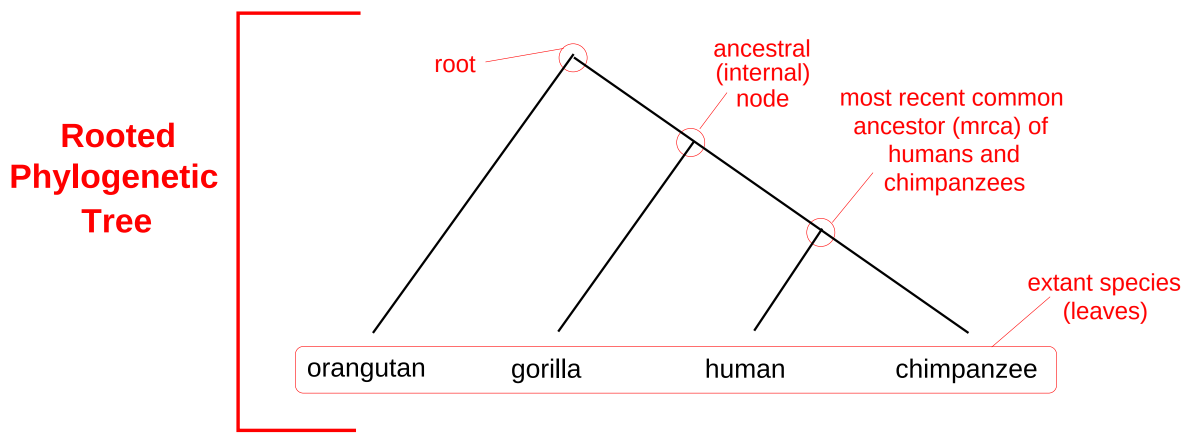


Figure 1.1: A sample rooted phylogenetic tree of primates based upon information from [80]. The term "tree" refers to the complete figure where the top node is the "root" and the extant species are the bottom "leaves." The most recent common ancestor (mrca) refers to the nearest common ancestor of any number of nodes. Branching patterns depicted in this type of representation provide information about evolutionary proximity and relationships among the species.

independent of each other and are arranged in a hierarchical fashion. However, it is known that in certain domains this assumption of independence is inaccurate, as interactions among two or more lineages do occur [9, 14, 27, 29, 35, 41]. Broadly, these interactions are referred to as "reticulate events" and at the interspecific level can be further categorized as "lateral gene transfer" or "hybridization." Typically the process of lateral gene transfer is found in prokaryotes (organisms whose cells do not contain a separate nucleus, such as bacteria) and its process is characterized by one lineage "donating" DNA to another. With hybridization (occurring in eukaryotes such as plants), two different lineages combine their DNA in a manner such that the offspring forms a new lineage that is often capable of sexually reproducing.

Topologies containing one or more of these reticulate situations (see Figure 1.2) are frequently referred to as phylogenetic *networks*. This terminology comes from graph theory and is applied to graphs containing at least one node with more than one "inbound" edge. Biologists dedicated to population genetics have an intraspe-

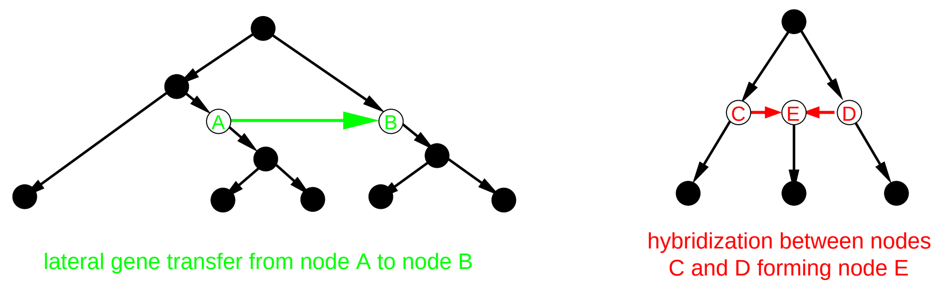


Figure 1.2: Two simple phylogenetic networks. On the left the green edge depicts a lateral gene transfer event and a hybridization is shown on the right.

cific, within species, focus and have a head start on techniques for lineage interaction, as their domain has historically included the reticulate evolutionary behavior of individual organisms (e.g. chromosomal recombination [54] and mating [32][1]). As the mechanisms by which reticulation occurs for "within" versus "across" species differ, models and methods are usually designed for either the intraspecific or interspecific level. Since trees are the common representation for interspecific evolutionary histories, there is a need for algorithms, techniques, and tools to address reticulate evolution at this broader level, which is the focus of this dissertation.

By reconstructing phylogenies, biologists gain two important pieces of information – order of speciation and branch lengths. However, it is difficult to know these facts with certainty for real biological data. Therefore, when developing and assessing reconstruction methods, it is common practice to work with a framework such as that depicted in Figure 1.3. Using a "source" history (either simulated or constructed by hand), the resulting extant taxa[2] are provided as input to a reconstruction algorithm, which in turn generates a "proposed" evolutionary history by inferring a tree from the

[1]Chromosomal recombination is also known as meiotic recombination. The terms sexual recombination and pedigree are often used in lieu of the term mating.

[2]The term "taxa" here is used to mean a generic collection of items. Depending on the scope of a phylogeny, a taxon may represent a single *species* or a single *individual* of a species. Recall from Figure 1.1 that the term "extant" refers to present-day taxa as opposed to ancestral ones.

"leaves" to the "root" with each internal node representing an ancestral species. The two phylogenies can then be quantitatively compared to evaluate how topologically similar they are.

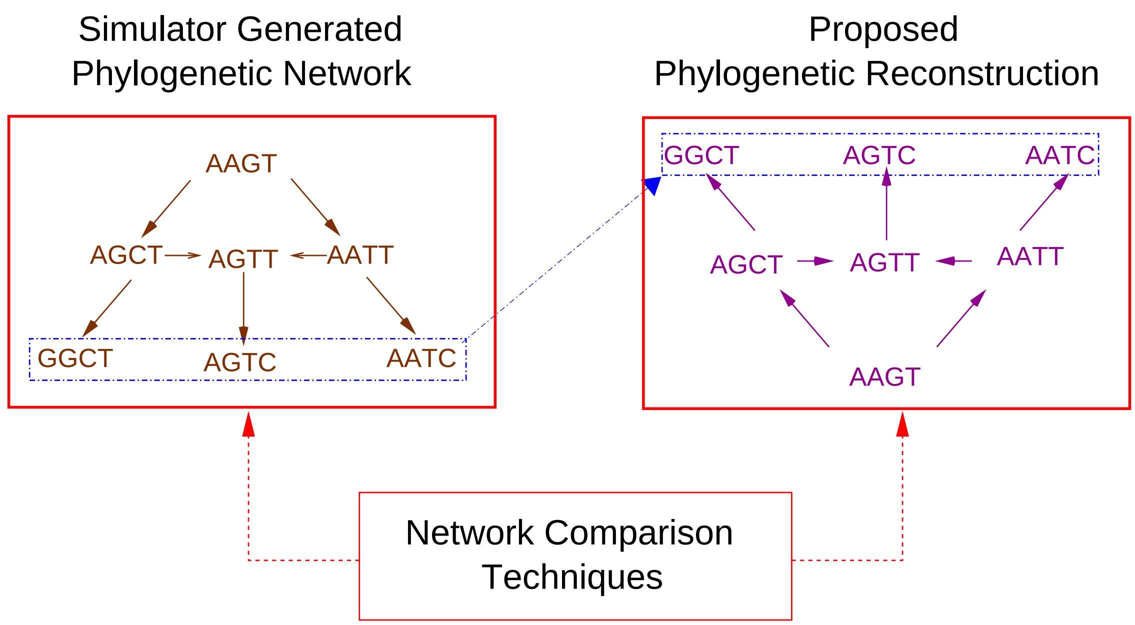


Figure 1.3: The three phases of phylogenetic algorithm development – simulation, reconstruction, and comparison.

The direct benefit of using this framework is that the source topology, evolutionary model, and branch lengths are known. Hence, quantitative and objective comparisons between the two topologies can be performed, in addition to developing and exploring reconstruction algorithms for a potentially diverse set of well-characterized sources. However, the weakness of this technique is that a good reconstruction is only a measure of how well one can reproduce simulated networks. Typically, a reconstruction algorithm that is capable of inferring histories similar to simulated source topologies, is further validated by biologists using an extant data set for which the relationships are accepted or known.

There are two general weaknesses of artificial topologies – inadequate models of sequence evolution and the generation of branches and their lengths. Both of these topics as they relate to trees are discussed further in Subsection 1.2.2. Furthermore, simulations that do not include the possibility of reticulation, or poorly model it, will fail to produce realistic evolutionary histories for those domains where events such as hybridization and lateral gene transfer are known to occur. These potential limitations motivate designing a simulator that generates networks arising from real-world factors. Thus, reconstruction algorithms that recreate these simulated networks will arguably yield better results when attempting to infer networks from true biological data.

1.2 Scope and Related Work

Phylogenetic networks with hybrids are the focus of this work. Although an effort was made to develop algorithms and software applicable to hybrids in general, portions of the research demanded a more narrow scope. In these situations, diploid hybridizations, which are defined in Subsection 1.2.1, were chosen. Subsection 1.2.2 covers existing simulation, comparison, and reconstruction techniques, which are frequently oriented to tree topologies. Finally, previous reticulate research and how it relates to this work, is reviewed in Subsection 1.2.3.

1.2.1 Hybridization

Hybridization is an evolutionary event known to occur in many groups of organisms and has been studied for centuries. In his 1959 article entitled "The role of hybridization in evolution" [74], Stebbins states that Kölreuter conducted the first systematic hybrid experiments from 1761 to 1806. Although hybrids do occur in the animal kingdom (e.g. [14]), biologists working with plants are fortunate to have multiple instances that either occur naturally or can be created in the laboratory [62, 81]. With this abundant supply of sources and new laboratory techniques with increased data throughput [81], knowledge about the nature of hybrids can be advanced.

The term “hybrid” refers to the sexual crossing of two different lineages that produce a new offspring, the first generation of which is called “F_1.” Within biology, the general term hybridization can be used for both “within” and “between” species crosses. Hybrid *speciation* is a more specific term used to describe lineages that are not only the result of two different species, but have the capability of reproducing and existing distinctly from their parents beyond the F_1 generation [35]. It is these interspecific, persistent lineages, resulting from hybrid speciation that are the focus of this work. However, for the sake of simplicity, we will frequently refer to them as “hybrids” throughout this document.

Two types of hybridizations are diploid and polyploid, referring to the ploidy level of the produced offspring. The number of sets of chromosomes found in an organism defines its ploidy level. (For example, humans are diploid organisms having 23 *pairs* of chromosomes, for a total count of 46 chromosomes.) Each chromosome is a strand of DNA and can be represented as a string of nucleotide bases using the letters A, G, C, and T for adenine, guanine, cytosine, and thymine respectively [72]. The chromosomes of diploid organisms are paired, and each of the two members is often referred to as a chromosomal homologue [37].

A diploid hybrid is the offspring formed from two different, diploid parental species. Despite the differing parental lineages, the regular sexual process proceeds, where each parent randomly contributes one chromosome from each of its pairs to the new offspring. Since each parent only contributes one chromosome from each of its chromosome pairs, both parents must have the same number of chromosomes. If they have different numbers of chromosomes, at least one chromosome in the hybrid will be unpaired, which will almost always produce a sterile hybrid even if it is viable [37]. In contrast, polyploid hybrids inherit all of the chromosomes from the two different parents. It is possible in this scenario for the parents to have different numbers of chromosomes. As each parent provides their complete set of chromosomes to the hybrid, different levels of polyploidization are possible. For example, polyploid hybridization between two diploid species results in a tetraploid, and polyploidization between a tetraploid and a diploid species results in a hexaploid [37]. Examples of both types of sequence inheritance, when the two different parents have a single pair of chromosomal homologues, are illustrated in Figure 1.4. Note that for diploid hybrids, the chromosome contributed by each parent to form the new pair of homologues in the offspring is random. Therefore, for each new pair of homologues in the diploid hybrid, there are four possible combinations.[3]

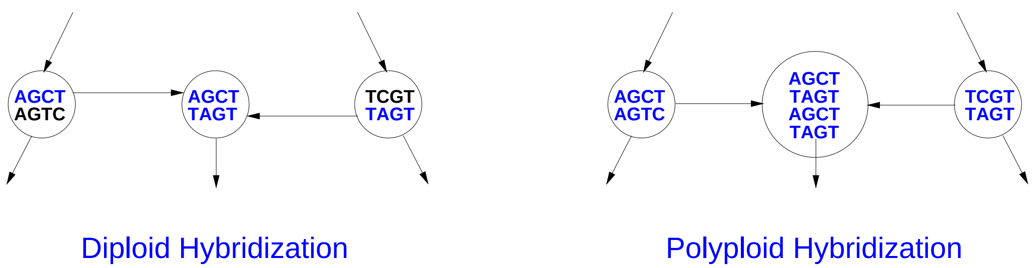


Figure 1.4: In the diploid case, each parent randomly contributes one of its chromosomes, from every pair, towards the formation of the hybrid's set of chromosomes. In contrast, polyploid hybrids inherit all of the chromosomes from both parents.

Although both types of hybridizations are known to occur, there are many more documented cases of polyploid than diploid events [23, 50]. Nonetheless, we chose to focus our immediate efforts on diploid hybridization. As compared to polyploids, diploids are more constrained and more closely parallel the typical mating process found at the intraspecific level. These features reduce the complexity and allow for the possibility of comparing and extending our work to intraspecific reticulation.

[3] Even more than four possible outcomes for a pair of new homologues exist when other evolutionary processes such chromosomal recombination ("crossing over") are taken into account.

1.2.2 Existing Phylogenetic Techniques

Simulation models are often classified as deterministic versus stochastic [67] and continuous-time versus discrete-event [5]. Deterministic models are characterized by fixed relationships [67]. This approach is beneficial when studying a system that has no random components and changes over a period of time. Stochastic simulations allow for variation of behavior due to the use of random numbers [59]. With a simulator of this type, one can produce a new outcome for the same time period and set of input parameters, by simply changing the random number seed. This permits the investigation of average behavior over multiple executions. Discrete-event simulations lend themselves to systems where there is a finite number of changes over a period of time [5, 67], but the different types of events and their potential interactions may be sufficiently complex to warrant a method other than one based on equations. These are in contrast to continuous-time models where differential equations are used to model an infinite number of changes in a limited time span [5]. When studying the behavior of a system with distinct event types, a stochastic, discrete-event based simulator is often a reasonable approach in terms of modular implementation and desired analysis of average behavior.

The birth-death model is a common population model used by biologists [59]. In searching for patterns to explain the number of species per genus in real data sets, Yule [85] developed a birth-only, exponential model for speciation [1]. Despite Yule's assumptions [21, 85], Yule states that the agreement of the predicted and actual data is "better than one has any right to expect" [85]. Whether this representation is used for studying the number of species or individuals, two limitations are that: 1) predicted growth cannot continue indefinitely and 2) an assumption of relative independence exists [21, 59]. Death/extinction events are commonly integrated with the Yule process, thus forming the well-established birth-death model. Each lineage is modelled as a Poisson process, which is characterized by being independent with exponentially distributed intervals [18, 66]. For the birth-death model, the expected number of lineages at time t can be derived as $m(t) = n_0 e^{(B-D)t}$. The mean of the size at time t is $m(t)$ and n_0 is the initial size. The birth and death rates are B and D respectively. This mathematical result is one technique for validating an implementation of the birth-death model. Although not without limitations, the birth-death model does provide, with a minimal amount of mathematical complexity, a reasonable estimate of the change in the number of species, when studying average behavior.

When working with a process that begins with a single species or individual and evolves into many entities, the notions of a root and start time are straightforward. However, when commencing with many extant taxa and working backwards in time, a root is typically identified by use of an outgroup. By definition, an outgroup is one or more taxa known to have a common ancestor with the other extant taxa. However, the outgroup species is known to have diverged prior to the evolution of the most recent common ancestor for the other extant taxa. This establishes a common point of origin, which in turn defines the divergence history for the extant taxa [15].

In addition to the establishment of a root, there are two separate notions of time associated with a phylogeny – evolutionary and clock based. The passage of seconds, years, decades, centuries, etc. is known as clock time and assuming a single origin of species, the amount of time elapsed from the origin until each current day species is the same. However, the extent of evolutionary change a species has undergone is referred to as evolutionary time and can vary across species. For example, in [17] it is reported that rodents and humans, which both originate from a common ancestor, have very different rates of evolutionary change.[4] Topologies where the clock and evolutionary time are considered equal, or even proportional to one another, are referred to as ultrametric. Non-ultrametric topologies are where the clock time is

[4]Humans are shown to evolve more slowly than rodents based upon amino acid substitutions [17].

consistent across lineages, as it must, but evolutionary time is allowed to vary.

A related, but separate, issue related to ultrametricity is event rates. Typically phylogenetic models employ a constant rate approach where all lineages share the same rate for a given event. However, some families of species are known to speciate more frequently than others, which cannot be captured with a model requiring all lineages to have the same birth rate. For example, [40] contrasts the cichlids (a type of fish) known to have speciated many times in the last 12,000 years to the species of skunk cabbage found in Asia and North America that were separated millions of years ago and have not speciated at anywhere near the same rate. Variable rate models such as [22] permit the investigation of such phenomena.

Although phylogenies can be simulated or reconstructed using morphological data (such as shape and form), the current trend is to use DNA sequences. Chromosomes are comprised of DNA nucleotide sequence information with each location in the string being referred to as a site. Although it is possible to study sequence evolution at higher levels such as protein and gene order [78], our interest is at the nucleotide level.

Given an evolutionary branch length and a starting DNA sequence, there are a variety of models that can be used to predict the resulting sequence. Some of the most common models include: Jukes-Cantor (JC), Kimura two-parameter model (K2P), Hasegawa-Kishino-Yano model (HKY85), and Felsenstein 1984 (F84) [78]. Two limiting assumptions with these models are independent sites and a consistent rate across sites. This means they assume that each site within a DNA sequence evolves at the same rate as all the others and without being influenced by their own evolutionary history or that of other sites [15].[5] They also do not account for

[5]Note that even if a distribution is used to allow for individual sites in the same sequence to evolve at different rates, a "consistent rates across sites" limitation is still present if the same distribution is used for all sequences in the model. For example, if the employed distribution favors the sites located in the first half of a sequence to evolve more slowly

insertions and/or deletions of sites (commonly called "indels") to/from the sequence. Although [15, 82] list more complex models of evolution that compensate for these assumptions, and at least one such model is implemented [75], it is typically the simpler models that are implemented as components of simulation or reconstruction applications [10, 26, 58, 84].

Phylogenetic trees are often compared by calculating the Robinson-Foulds (RF) distance [63]. This metric captures the number of incorrect edges between two topologies. It is covered in Chapter 3 as it is the basis for the Extended Robinson-Foulds measure [43], which we make use of in our reconstruction experiments. Another type of comparison used in phylogenetic analysis is the distance between two sequences. When substitutions, and not other operations such as insertions and deletions, are the only option for altering a string, the Hamming distance is a common distance measure. It is defined as the number of indices that do not agree (see Figure 1.5).[6]

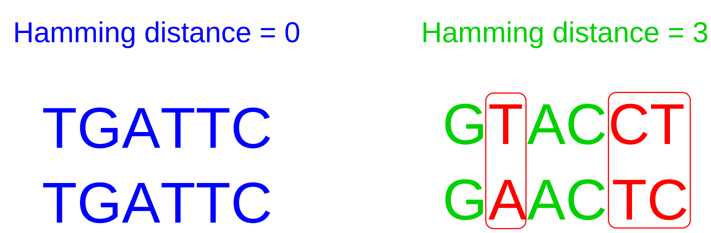


Figure 1.5: Hamming distances for two pairs of DNA sequences. The pair on the left has a Hamming distance of 0 indicating the sequences match and the pair on the right has a Hamming distance of 3.

than those in the second half, this behavior will exist for all branches, thus leading to the same problem as before, but at a higher level of data [82].

[6]Hamming originally applied this pairwise distance to sequences of 0's and 1's representing vertices of an n-dimensional, unit cube [20]. However, the definition he provided in the seminal paper [20]: "Thus we define the distance $D(x, y)$ between two points x and y as the number of coordinates for which x and y are different." does not preclude other alphabets (characters) for the strings. For example, page 214 of [15] states "...Hamming distance (the observed uncorrected number of differences between two sequences)..." It is this latter, less restrictive definition of Hamming distance that we will use in this work.

Sequence distance data can be used as the basis of tree reconstruction algorithms. Neighbor joining is a common distance method where taxa are repeatedly paired to form ancestral nodes of the inferred tree according to the minimum distances of their sequences. Due to its simplistic approach and underlying assumptions, this method does not assign sequences to the internal nodes and is not appropriate under all circumstances, however it is one of the least CPU intensive methods available [19, 78].

The two other broad categories of reconstruction techniques, in addition to that of distance data, are maximum parsimony and maximum likelihood.[7] The assumption that evolutionary changes occur in the most efficient manner, namely the fewest alterations to the DNA sequences, underlies maximum parsimony approaches. Finding the most parsimonious tree is NP-hard [72]. As the number of possible tree topologies grows rapidly for a given number of extant taxa (e.g. over 2x10^6 possibilities for 10 extant taxa and 2x10^{182} for 100 extant taxa [19]), identifying the most parsimonious one(s) is a computationally expensive task, even with the aid of good heuristics. However, this approach does generate the sequence information for the proposed internal/ancestral nodes, which is an advantage for some biological studies [42]. A related problem is to assign sequences to internal nodes of a given topology in the most parsimonious fashion. This is known as the Fitch small parsimony problem [11, 15, 78] and is solvable in polynomial time.

The third category of techniques falls under maximum likelihood. These methods are based on the statistical assessment of how likely the provided aligned sequences are to be observed, under an assumed model of sequence evolution [78]. On average, finding the most likely tree is more computationally expensive than finding the most parsimonious one [42], and like maximum parsimony, is NP-hard [64]. However for small sets of extant taxa, this approach is preferred often by biologists because it performs as well as or better than other methods, even when using shorter sequence

[7]The reader is referred to [15] and [78] for more detailed, yet still accessible, reviews of reconstruction techniques.

lengths [78].

With maximum parsimony and maximum likelihood optimally solvable for only small numbers of extant taxa [15, 19], and as greater quantities of DNA information become available, the need for reconstruction algorithms addressing larger sets of species and/or longer periods of evolutionary time has become more imperative.[8] Three separate research efforts addressing these questions are GRAPPA [4], GARLI [86], and RAxML [73]. The GRAPPA work is focused on reconstructing evolutionary histories using gene-order data, which is a higher level of DNA information as compared to protein strings or individual bases at the nucleotide level. The benefit of this approach is that rearrangements of DNA sequences usually evolve at a slower pace than at the nucleotide level and greater amounts of evolutionary time can be investigated [78]. The GARLI and RAxML tools perform maximum likelihood reconstructions and both have sequential and parallel implementations. Various algorithmic and performance improvements have allowed them to be successful on data sets with over 500 extant taxa [73, 86].

Techniques that merge multiple topologies and quantify the amount of confidence in subtrees are also critical to phylogenetic efforts. As maximum likelihood and maximum parsimony techniques in particular can return more than one tree topology with an optimal, or near-optimal score, there is a need to be able to synthesize the information into a single topology. Common consensus tree algorithms are majority-rule and strict consensus [15]. However alternative techniques that seek to identify common structure across reconstructed trees have also been investigated [13, 52]. A separate technique for evaluating a reconstructed topology is bootstrapping. By sampling with replacement on the columns of DNA data and repeating the reconstructions process, insight into which subtrees and evolutionary relationships persist

[8] Recent advances in techniques are allowing for reconstructions of larger extant taxon sets [73], but have not yet reached the scale being pursued by projects such as AToL [3] and CIPRES [6].

can be gained [19, 15].

Fortunately, there is a wide selection of tools to accomplish many of the algorithmic tasks discussed above. Felsenstein maintains an extensive list of phylogenetic software – `http://evolution.genetics.washington.edu/phylip/software.html`. Although most of the software is oriented towards tree topologies, many are capable of providing reasonably good results when applied appropriately. For aspects of our simulation and reconstruction efforts that could be accomplished with tree-based techniques, we considered and often chose to use existing tools. This provided the convenience of not rewriting applications, in addition to affording potential users a level of reliability and a sense of familiarity.

1.2.3 Prior Reticulate Work Related to this Effort

The term phylogenetic network is often loosely applied to any phylogenetic structure that is non-tree like and can arise for reasons other than an interspecific reticulate history. With our goals being the simulation, measurement, and reconstruction of interspecific networks containing hybrids, we first consider in this section other efforts that address or support this immediate pursuit, and then turn to intraspecific recombination and the detection of reticulate histories.

A key component for our effort is a realistic network simulator. To date, most interspecific, reticulate simulators have been primitive. For example, a tree topology is created first in its entirety and then reticulate branches are added manually (e.g. [47]), before any sequences are evolved. Furthermore, it is important to define event rates and employ a statistical model that are appropriate for an interspecific focus. Namely, events should be permitted to occur at any point during the simulation, not be restricted to any certain pattern, and yield reasonable numbers of lineages.[9] As a suitable generator did not exist, we have created a new one.

No single, all-encompassing metric for quantifying the similarity between phylogenetic networks exists. However, the tripartition measure [43] is a leading technique to compare the topologies of two networks. It is an extension of the well-known Robinson-Foulds measure for trees and captures the percentage of differing internal edges. This is a meaningful value when performing reconstructions and is applicable to all networks whether arising from hybridizations or lateral gene transfer events. The tripartition measure became our primary method for evaluating reconstructions.

Initial approaches for reconstructing phylogenetic networks at the interspecific level have been adaptations of existing methods normally applied to trees. Two techniques motivated by the concept of maximum parsimony have had different levels of success. In [79], a tree topology was first reconstructed and then potential network edges were added post-facto using a parsimony approach. According to Tholse [79], this "heuristic appears ill-suited" as the addition of extraneous reticulate edges is promoted by the parsimony criterion. However, Hein [24] proposed a different strategy for applying the parsimony criterion to networks. When restricting the scope to cases of lateral gene transfer, with a known underlying tree, favorable results using Hein's approach have been reported [29]. Maximum likelihood based techniques have also been applied to scenarios with lateral gene transfer when the underlying tree structure is known [30]. Unfortunately, with our focus on eukaryotes and hybrid speciation, having an underlying/starting tree topology is not a realistic assumption and necessitated the development of a more complex algorithm for reconstruction.

Population geneticists work at an *intra*specific level, and although they work with different models and parameters, they must also deal with reticulate events. Two

[9] Whereas an annual reproduction cycle may exist for a single species, thus motivating a different type of model, set of rates, and expected growth for a population level study.

such events are sexual and chromosomal recombination. Sexual recombination refers to mating where each parent donates half of its sequences towards the creation of each offspring. Chromosomal recombination encompasses a reciprocal exchange of DNA known as "crossing over" and a non-reciprocal form called "gene conversion" [15]. Though operating under different model assumptions from us, Xu [83] worked to incorporate reticulations using a least-squares approach for reconstruction purposes at the population level. Schierup and Hein [71], as well as Posada and Crandall [56], examine the impact of recombination, at the intraspecific level, on phylogenetic trees. Although these efforts had different goals than ours, they did provide the seeds for our new timing and diversity measures, as well as some of our experiments.

Simply detecting the occurrence of reticulation before attempting to reconstruct or analyze data is an active area of research and multiple techniques have been developed at both the *intra* [54] and *inter* [27, 70] specific levels. Posada and Crandall in [54] state that the most common approach found in the literature for detecting recombination is based on phylogenies. One such tool is SplitsTree [27], which identifies when DNA sequence data is inconclusive for forming a tree and displays what is known as "splits" for a given set of extant taxa. Often referred to as phylogenetic networks, these topologies can be caused by a variety of sources including a history of reticulate events, but also error in sampling the DNA sequences or side effects from interpreting the data. Although an important area of research, this category of phylogenetic network analysis is separate from our efforts.

1.3 Overview of Dissertation

This work focuses on simulating, reconstructing, and characterizing similarities between phylogenetic networks with diploid hybrids. Our first task was to develop a network simulator for creating the topologies that would act as the initial/source, phylogenetic networks. As *trees* are comprised of independent lineages, the tasks of creating a topology and evolving sequences on it can be separated without any impact on the result. However, in the real world, sequences evolve over time and become intermingled across species due to reticulate events. Hence, it became a priority to generate both a topology and its sequences in a simultaneous fashion. Unable to find a simulator meeting these requirements we have created NetGen [46], which is based on the traditional birth-death model [59], but incorporates hybrid events as well as simultaneous sequences for the topology. The model design and primary NetGen features, along with experimental results for its validation and characterization purposes are presented in Chapter 2.

Chapter 3 reviews the Robinson-Foulds distance measure for trees and its extension to phylogenetic networks [43] along with introducing three new quantitative measures for describing reticulate nodes within a phylogenetic network. Reticulate timing captures whether a hybrid is ancient or recent with respect to the overall height of the network. Calculating the percent of extant taxa that are descendant from a hybrid node, defines the reticulate impact measure, which is an integral part of our reconstruction algorithm. And finally, the reticulate diversity measure quantifies the relative topological placement of the lineages involved in the reticulate event as either being close or divergent. Experimental results and analysis highlighting the behavior and interaction of these measures are also discussed. The final phase of this effort required developing a reconstruction algorithm. Restricting the effort to networks with a single-diploid-hybrid event, NetReconstruct was designed and implemented. The initial results are promising and Chapter 4 reviews the algorithm, software details, and experiments exploring the model's performance and sensitivity to a variety of parameters.

With the foundation established for simulating, measuring, and reconstructing networks from the earlier chapters, a case study, presented in Chapter 5, was undertaken to investigate what types of reticulate nodes have the greatest potential to impede reconstruction efforts. Following the lead of Posada and Crandall [56] to categorize and examine topologies with a single reticulate event, we have created single-diploid-hybrid networks, categorized them according to our quantitative measures of timing and diversity, and inferred histories using our reconstruction algorithm. We found at the interspecific level, as Posada and Crandall reported for the intraspecific level, that generated networks with an ancient reticulate node, one that occurred early in the simulation, were more accurately reconstructed than those with a recent, divergent event. Finally Chapter 6 presents the conclusions from this work and future directions for research in the area of phylogenetic networks are covered in Appendix C.

Chapter 2

Simulating Phylogenetic Networks

NetGen [46] is a phylogenetic simulator capable of producing tree and network topologies. Moreover, the DNA sequences are generated simultaneously with the topology. This is novel because, as in the real-world environment, sequences evolve over time and are allowed to impact the outcome of reticulate events, whereas past simulators assign sequences in a post-processing step [47, 49]. Creation of a realistic network generator is a necessary prerequisite to developing and evaluating future network-based reconstruction techniques.

In this chapter, we provide the principal motivation and design details (Section 2.1) for our simulator, NetGen. Section 2.2 summarizes results of key validation tests, and the model's behavior with hybrids is characterized in Section 2.3. Finally, this extension to networks required a new text representation for capturing hybrid nodes in topologies, which is discussed in Section 2.4.

2.1 Network Simulator – NetGen

Traditionally, phylogenetic networks are created in a multi-phase approach depicted in Figure 2.1. The first step is to create a tree topology using a tool such as R8S [69] (pronounced "rates"). Next, reticulate edges and nodes (e.g. hybrids) are added to the topology by hand. Finally, sequences are evolved to the fixed topology [47]. Another technique is to combine the first two steps into one model, which if chosen correctly may provide a more realistic placement of reticulate events [49]. However, neither of these cases allows for the evolving sequences to influence the subsequent reticulate topology.

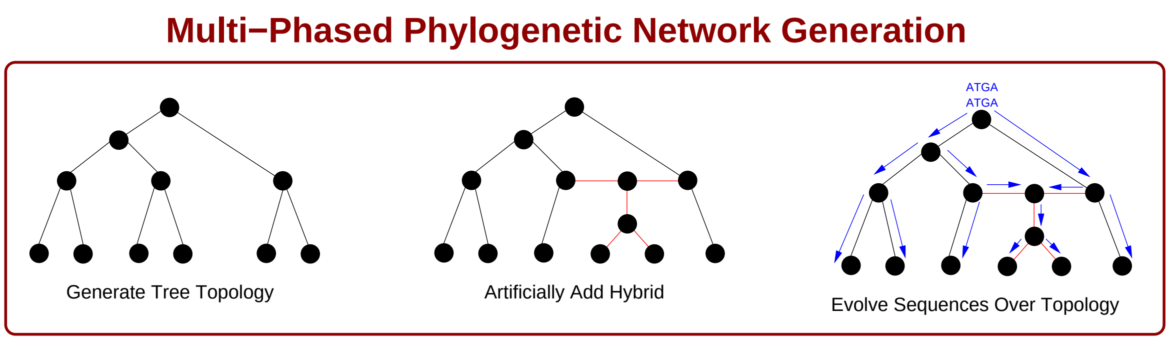


Figure 2.1: The traditional multi-phased approach: generation of the topology, introduction of hybrid events, and evolution of sequences.

Evolutionary distances are known to play a significant role in hybridization and two lineages will hybridize only if their genetic makeups are compatible.[1] One method for assessing hybridization compatibility is to calculate sequence distance, motivating the incorporation of DNA sequence information into the simulation. However, factors other than DNA sequences do also influence and constrain hybrid events. For example, two parent lineages cannot hybridize unless they both co-exist at the same time and reside in geographic proximity. Since it is not possible to incorporate all

[1]As discussed in Chapter 1, the hybrids of interest for this work are ones that are distinct from their parental lineages and continue to thrive over an extended period of time.

potential influences for hybridization into a simulator, time and DNA sequences were given priority.

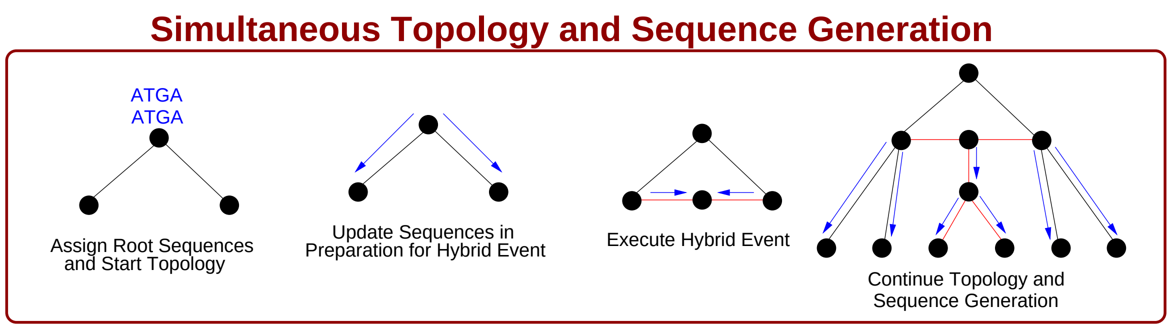


Figure 2.2: The simultaneous approach implemented in NetGen where the sequences are evolved as the topology is created as well as influence the outcome of the topology as the hybrid decision is based upon up to date sequence information.

In an effort to create a tool that produces realistic phylogenetic networks, we have developed NetGen. As discussed in Chapter 1, the birth-death ($B - D$) model [59] generates tree topologies by assuming events occur according to a Poisson model. The implication of this is that the birth and death events are independent and have interarrival times that are exponentially distributed. NetGen extends this model by introducing a new event, H, for hybridization. Prior to a lineage undergoing a hybridization, updated sequence information is used to identify a suitable second parent lineage from other lineages that are currently active. Moreover, the updated, real-time sequence information is available to influence the evolving topology. Although the interarrival times for all three events – births, deaths, and hybrids – remain exponentially distributed, the model can no longer be considered strictly Poisson because the lineages do interact, thus violating the independence requirement. In contrast to Figure 2.1, the simultaneous approach to generating phylogenetic networks is illustrated in Figure 2.2.

Although based on the birth-death model, the source code for the simulator is completely new and employs a modular design. As there are multiple models for evolving sequences and tools already exist for accomplishing this task with trees, we chose to write an interface that interacts with Seq-Gen [58], a well-established tool for this purpose. The event management and the hybridization portions of the model were given extra attention as these are the novel aspects of the software. Drawing from features of existing tree simulators, we also chose to incorporate such options as variable rates, non-ultrametricity, and outgroups to allow for the further customization of phylogenetic networks and greater exploration of how such parameters impact a simulation of this nature.

2.1.1 NetGen Executable Structure

NetGen is modular and is comprised of three executables : NG, NSGW, and a nucleotide sequence generator. NG is the primary simulation code, which creates the phylogenetic network by processing birth, death, and hybrid events. NSGW interfaces between NG and the sequence generator, and is responsible for processing and responding to sequence requests. The third piece is an independent nucleotide sequence generator, which provides an evolved sequence given an initial DNA sequence and a branch length (period of time). For this task, we have chosen Seq-Gen [58], which is a well-established sequence generator capable of employing different models of evolution.[2] These three pieces, NG, NSGW, and Seq-Gen, operate separately and communicate via pipes. With this configuration, it is easy to swap in/out new versions of Seq-Gen as they become available and if one desires, a different nucleotide sequence evolution tool can be employed with only the interface code requiring modification. Moreover, NSGW is capable of assigning sequences to a predefined topology independent of requests from NG. Although initially developed for testing purposes, this functionality provides the capability to perform the final

[2]Customization options available for Seq-Gen can be specified as input to NG and will be subsequently used when processing the sequence request.

step of a traditional, multi-phased network generation, or simply repeated sequence simulations for a single topology. Figure 2.3 illustrates the three executables that comprise NetGen, their pipe interaction, and primary responsibilities.

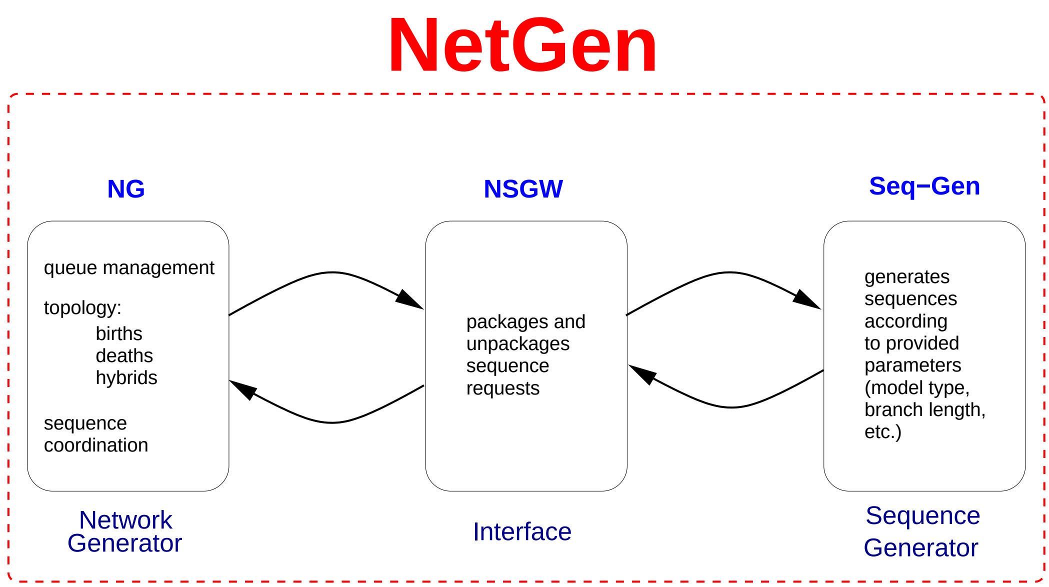


Figure 2.3: Components of NetGen and the flow of sequence requests and responses among them. Seq-Gen [58] is a well-established sequence generation tool provided by a group at the University of Oxford while the other two executables (NG and NSGW) were developed for this research and are available in C source code, under GNU General Public License, from the author at `http://www.cs.unm.edu/~morin/`.

In general, branch lengths and sequences for a lineage are updated at event time. Birth and death events require only an update for the specific individual lineage being acted upon as these events occur independently of other lineages. However, for hybridizations, the branch lengths and sequences for all active lineages must be updated in order to ensure that the second lineage is chosen based upon current information. Branch lengths are revised internally and sequences are altered with external calls. NG requests a sequence update by providing a start sequence and a branch length to NSGW. In turn, NSGW communicates with Seq-Gen and then relays the response back to NG. Updates may occur multiple times along a single lineage and NetGen tracks these intermediate sequences and branch lengths in order to guarantee that subsequent requests are performed using the correct information.

2.1.2 Simulation Events and Termination

NetGen tracks events for simulation using a global queue. An event is a generic structure which has a type, scheduled time, and lineage to which it is assigned. These events are processed sequentially according to their time-stamps in the queue, and the simulation progresses as the events are completed. The queue is updated throughout the simulation. When the lineage undergoes a birth, two new lineages are formed and random draws, based upon the user-specified birth/death/hybrid rates, determine their future events. These new events are placed in the global queue in accordance with their scheduled times. Event times follow a Poisson model by having interarrival times drawn from an exponential distribution [18]. With a global event queue, model modification and customization is simplified when new functionality is desired because additional event types[3] can be implemented in a modular fashion.

Traditionally, two options are employed when ending a simulation. The first involves a user-specified, predetermined end time. Given that NetGen is an event-driven simulator, the generated networks all evolve at random rates, and fixing a predetermined end time would result in networks with differing numbers of extant taxa. The second option is to stop the simulation after the number of active lineages reaches a specified size. In our case, the final end time is randomly determined to be between the last event and the next one on the queue. This technique is most appropriate for an event-driven simulator, but one must ensure all the branch lengths and sequences for the remaining active lineages are evolved to the final end time. If not, they will experience an artificial shortening in that their branch lengths and

[3]Some events that are not included in NetGen at this time include mass extinction and lateral gene transfer.

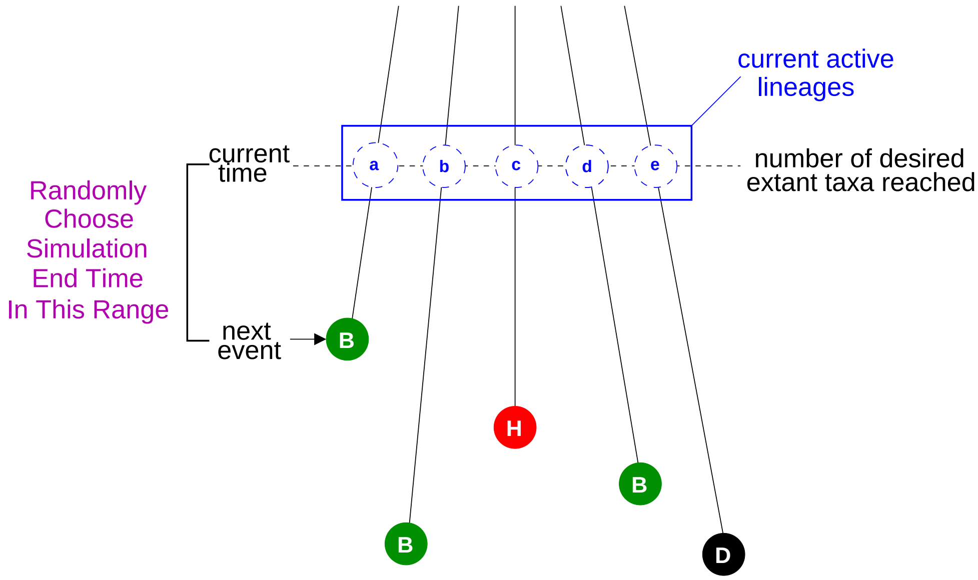


Figure 2.4: The simulation end time is determined once the desired number of extant taxa are reached. It is chosen randomly between the last executed event and the next scheduled event.

DNA sequence information will correspond to the time when they were created, and not the time at which the simulation ends. This approach is presented in Figure 2.4 and reflects the real world in that the time at which a biologist is sampling their extant taxa for a phylogeny is really a point in time between two evolutionary events. If death events are allowed to occur, it is possible the simulation will end prematurely, before the desired extant count is reached, as there may be no active lineages remaining.

2.1.3 Hybridization Implementation

Extending the original birth-death model to include hybridizations required the tracking of two separate notions of branch lengths – clock/elapsed time and evolutionary time. In order for two lineages to hybridize, they must co-exist at the same clock time. However, as discussed in Chapter 1, clock time is not necessarily equivalent, or even proportional, to evolutionary time and the two lineages may have experienced different amounts of evolutionary change since descending from the root. Evolutionary time dictates the amount of DNA change and is used in requesting sequences. Sequences for our networks are generated using SEQ-GEN [58]. The interface, NSGW, coordinates the generation of one sequence along one branch at a time between the requesting code (NG) and the supplying code (SEQ-GEN).

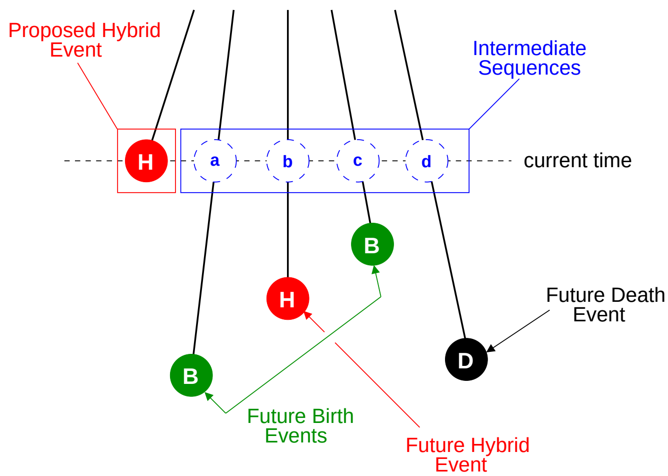


Figure 2.5: In preparation for the hybridization, all active lineages receive updated sequences based on the amount of evolutionary time since the last update.

When a hybridization event is scheduled, it is placed in the queue and associated with a specific lineage. When it is time to execute the hybridization, branch and sequence information for all active lineages is updated. It is this information that is used to identify a suitable second parent, see Figure 2.5. If the constraints are met, and a suitable second parent is found, a hybrid offspring is created and a future event for this new lineage is added to the queue. The lineage originally identified as having the hybrid event receives a new event for its lineage that propagates, while the second parent keeps its previously assigned and scheduled event on the queue (Figure 2.6). If for some reason the hybridization event does not occur, the event is marked as processed, but not executed, and the original lineage is assigned a new event which is added to the queue.

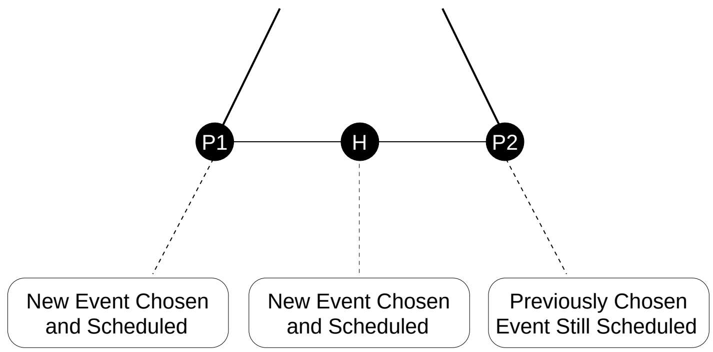


Figure 2.6: The result of a hybrid event are three active lineages (one from each of the two parents and the hybrid itself). The lineage initially scheduled for the hybridization (P1) and the newly created hybrid lineage (H) each receive new events, while the second parent (P2) keeps its previously scheduled event already present on the queue.

The user can customize many constraints concerning the hybridization event. As part of the input process, the user declares the number of chromosomes and ploidy level for the start of the simulation. Then when specifying the hybrid rate, the user can provide further values to designate the proportions of diploid and polyploid hybridizations, thereby allowing both types to occur in the same simulation. The two parental lineages for a diploid hybridization must have the same number of chromosomes and an even ploidy level. However, polyploid hybridizations involve the merging of sequences from both parents regardless of whether the number of chromosomes are the same. This type of event has the potential to yield tetraploids, hexaploids, etc. Although the focus of our work here is on diploid hybridization, it was important to provide the flexibility for the polyploid option as it is the most common category of hybridization for eukaryotes [37, 50].

There are four user options for how the second parent in hybridization event is chosen. Two of the options are based on sequences – minimum Hamming distance and an exponential function based on Hamming distance. The minimum Hamming distance option finds the lineage with the closest set of sequences to the first parent. In the case of a tie, one of the lineages with the minimum distance is chosen at random to be the second parent. Although this option does not leave room for deviation, it is motivated by the biological premise that parental lineages need to be similar. The second option is the most realistic in that the Hamming distance is the dominant factor, but statistical variation is allowed in order to capture other influences (e.g. geographical proximity and tendency to hybridize), which are also known to impact the hybrid choice. Although it is not realistic to explicitly model these additional factors, we use a Hamming distance function approach based upon the truncated exponential. The user specifies the average Hamming distance that has a $1/e$ probability (approximately 37%) of being the distance that the second parent will have from the first. This value defines a curve, from which a random number is chosen from the probability distribution, that in turn defines a target distance. Candidates with the target Hamming distance are identified and one is chosen at random. If no potential second parent with the specified distance is found, the search is incrementally expanded until a user specified bound is reached. Figure 2.7 shows a sample truncated exponential function where the specified value (H) is 250 and the maximum average Hamming distance is 500.

The third and fourth options for selecting the second parent of a hybrid are random and minimum evolutionary distance. Although these techniques are not based on sequences, they do offer the possibility of performing alternative hybrid investigations. With random, as the name implies, a second parent is chosen at random without consideration of the sequences involved. Although not biologically realistic, this option is used for analysis purposes as a contrast case and to speed the simulation when features not dependent on sequences (e.g. population growth rate) are studied. Finally, the minimum evolutionary distance option searches for a second parent with a minimum evolutionary branch length distance between the first

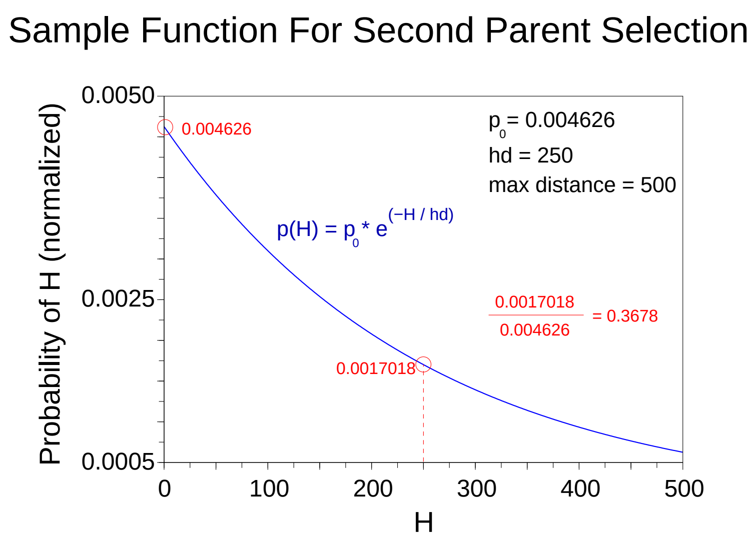


Figure 2.7: Sample function for second parent selection based upon a user-specified value (H) of 250. The H value identifies the curve based upon H being the average Hamming distance between the two parents of the hybrid with a $1/e$ probability. A random value from this distribution is chosen as the target average Hamming distance. Hence, the target distance is likely to be minimal, but not necessarily to the extreme.

parent and all other active lineages. As evolutionary distance is another measure of lineage compatibility, this is a biologically motivated alternative to using sequence information for selecting a hybrid's second parent.

As previously mentioned, occurrences of hybrids can be restricted by a user-specified threshold. For the exponential function, the threshold limits how far from the initially selected target Hamming distance to expand the search above and below the target value. For the minimum Hamming and evolutionary distance options, the threshold is a value above which hybridizations cannot occur. Furthermore the user can "cap" the number of hybrids for a simulation, thus influencing the location of the hybrid(s). For example, having a high rate of hybridization with a limit on the number of hybrids will lead to hybrid events occurring early on in the simulation.

2.1.4 Additional Features

In an effort to make NetGen a robust and versatile tool, other features were implemented to permit customization and a wide variety of experimentation. A seed for the random number generator can be specified on the command line when the simulator is executed and there are input file parameters that address: the number and type of sequences generated, constant and variable event rates, ultrametric and non-ultrametric branch lengths, and outgroup generation.

NetGen requires a stream of pseudorandom numbers for sequence generation (e.g. seeds provided to Seq-Gen and making choices in "tie" situations) and choosing values from a pre-defined distribution for event time selection. Although it is recommended that the user specify a seed on the command line for repeatability purposes, the code will create one from the clock if not explicitly provided.[4] The random number generator employed is known as Mersenne Twister [39] (version MT19937 February 2002) and is known for its high periodicity. Source code was available on the web and was used as a component in both NG and NSGW.

The user can set the number of chromosomes and ploidy level, as well as the length to be used for the root node. If desired, one can specify the DNA sequences to be assigned to the root node as well. The evolution of sequences is handled by Seq-Gen, and options that specify certain evolutionary models and parameters to that software can be specified as input to the NetGen simulation.

The default rate assignment is constant, meaning that all lineages in the simulation will have the same set of rates (birth, death, and hybrid) as specified on input. However, the user can opt for variable rates where each lineage is assigned its own set based on averages and variances provided. Another option is to create

[4]Whether specified by the user or generated based on the clock, the random number seed is reported in the output summary.

a non-ultrametric network, which means the extant taxa will not all be equidistant from the root in terms of evolutionary branch lengths. (The default is an ultrametric network where evolutionary branch lengths are equidistant.) This is achieved by multiplying each edge by a gamma distributed random variable, whose parameters are specified as part of the input file.

Finally, we provide a meaningful generation for an outgroup taxon if the user so desires. As discussed in Chapter 1, an outgroup is often used in reconstruction algorithms to root a network. This requires having a species that is common enough to the other extant species to have a reasonably close common ancestor, yet removed/different enough to not be involved directly with any of the ingroup taxa under study. Figure 2.8 shows an outgroup for a tree.

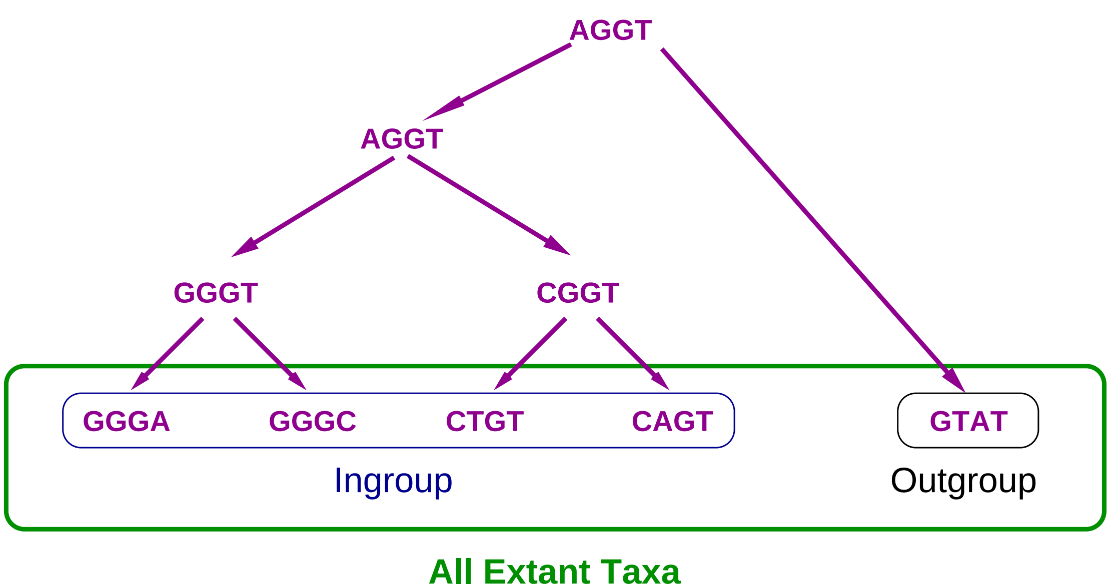


Figure 2.8: Outgroups, which are used to root phylogenies, are often carefully picked by biologists to be similar, yet distant from the ingroup. NetGen allows the user to specify sequence similarity/dissimilarity bounds so that a meaningful outgroup can be created to correspond with the simulated ingroup. The number of attempts at finding such an outgroup is limited by the user and program to avoid excessive processing.

By allowing the user to specify values for outgroup sequence similarity and dissimilarity, a more meaningful outgroup, with respect to the ingroup, will be generated and hopefully of use in reconstruction algorithms. A set of potential outgroup sequences is generated according to the evolutionary branch length between the root and outgroup taxon. The proposed outgroup is then scored based on Hamming distance, which is calculated between it and each of the ingroup taxa. If the candidate outgroup meets the similarity/dissimilarity bounds with respect to each member of the ingroup, the proposed sequences are assigned to the outgroup and the processing is complete. If the bounds are not strictly satisfied, another set of candidate sequences for the outgroup is generated and tested. This approach can be demanding on the processing time as it requires the potential generation and scoring of multiple candidates. Therefore, the number of tries is limited and if a candidate that meets the criteria in absolute terms is not found, a best alternative based on averages is chosen.

2.2 NetGen Validation

In an effort to gain confidence in NetGen results, experiments were designed to validate the code and its performance. The common approach for checking a birth-death implementation is to examine population growth, which is covered in Section 2.2.1. A discussion of branch length distributions is provided in Section 2.2.2. Section 2.2.3 confirms that the technique of incrementally updating a branch's sequence, necessitated by the addition of hybrids to the model, does not impact the overall sequence behavior at the branch or network level.

2.2.1 Population Growth

The traditional birth-death model has a very well characterized growth pattern [59]. Specifically at any point in time, the population size either stays the same, or gains or loses one lineage. The equation for growth is: $m(t) = n_0 e^{(B-D)t}$. Where $m(t)$ is the mean population at time t with n_0 as the initial population size and B and D are the birth and death rates respectively. By running NetGen to generate birth-only and birth-death trees (no hybrid events) for a variety of rate combinations, we see (Figure 2.9 and Table 2.1) that the growth pattern is matched and are therefore confident the implementation of the base model is correct.

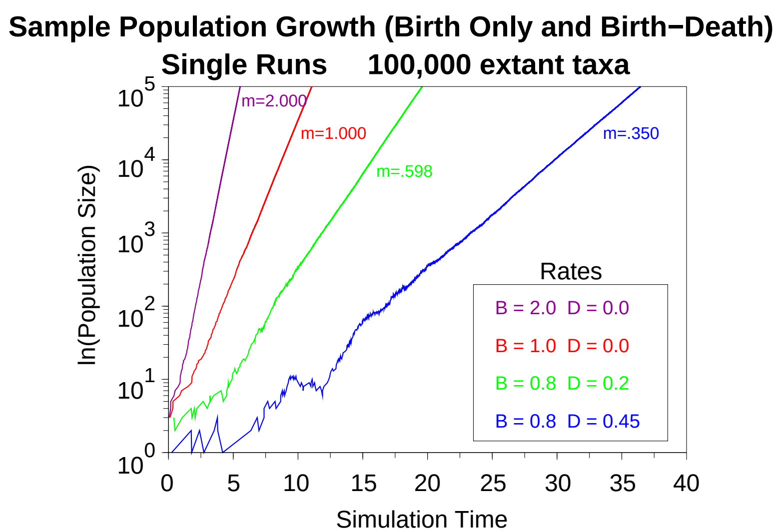


Figure 2.9: Population growth data for four different scenarios. Each line illustrates the number of lineages (natural log scale) over time for a single run of 100,000 extant taxa.

2.2.2 Branch Length Distribution

When NetGen terminates with the specified number of extant taxa there are three sets of branches:

Multiple NetGen Runs – 100,000 extant taxa		
Runs	Rates	Average Slope
10	B = 2.0 D = 0.0	1.997 ± .009
10	B = 1.0 D = 0.0	0.998 ± .004
8	B = 0.8 D = 0.2	0.598 ± .004
7	B = 0.8 D = 0.45	0.349 ± .004

Table 2.1: Average population growth for the same four scenarios presented in the table above. Calculating the average slope (natural log of the population size) shows consistency between the implementation of the model and the underlying mathematics (namely the slope being equal to the difference of the birth and death rates).

(i) *completed* branches, which are internal to the topology as their start and end points occurred during the simulation,

(ii) *future* branches, which started during the simulation, but were prematurely terminated when the simulation ended due to the desired number of extant taxa being reached, and

(iii) the *total* (combined) set of future and completed branches comprising all the *chosen* random variables that were selected to be branch lengths.

The first and third case are of general interest, with the third being the most straightforward to analyze. By recording the completed and future events, a list of all branch lengths can be compiled and then a histogram showing the distribution of branch lengths constructed. Figure 2.10 shows the normalized histograms for two runs (birth-only and birth-death) of 250,000 extant taxa as well as their fits.

Both distributions are exponentials with the form, $r_1 * e^{(-r_2 * t)}$. This is consistent with Poisson processes, which are characterized by events having exponential inter-arrival times [18]. In the birth-only (b=8) case on the left-hand side of the figure, the fits for r_1 and r_2 are 7.990 and 7.988; for the birth-death (b=9, d=3) case on the right, the fits are 12.007 and 11.999 which is the sum of the birth and death rates.

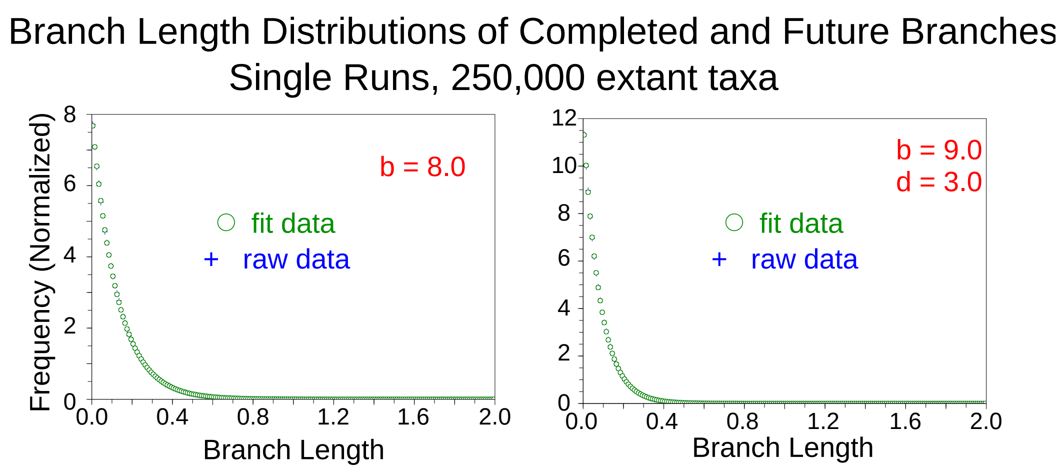


Figure 2.10: The branch length distribution in both cases is exponential. Fits are conducted with the *datafit* function in SCILAB [28] which employs a least squares approach.

In the latter case, this sum (as opposed to the difference used for population growth earlier) makes sense as this scenario is the superposition of two Poisson processes (with rates λ_1, λ_2) which is known to yield a Poisson process itself with an expected value of $\lambda_1 + \lambda_2$ [31].

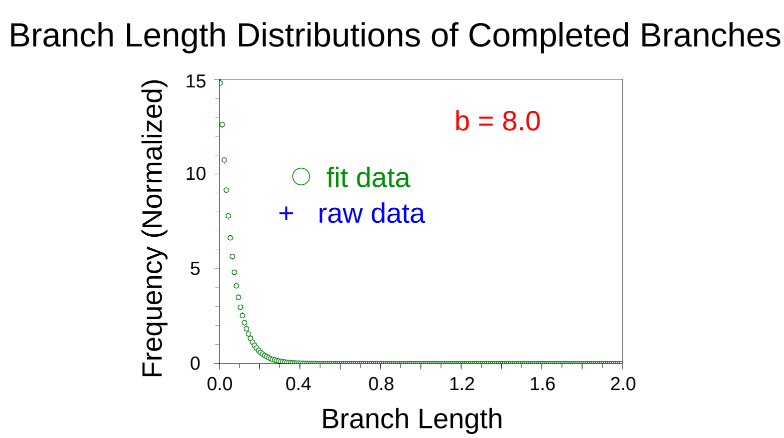


Figure 2.11: The completed branch length distribution and its fit for the birth-only case.

The other branch set of interest is that of (i) where the completed branch lengths are indicative of what is referred to in the field of branching processes as the age distribution. Using the same birth-only run from before, Figure 2.11 presents the data for the *completed* branches and the fit. The general fit comes from page 151 of Harris [21] which provides the age distribution formula as:

$$A(x) = \frac{\alpha \int_0^x e^{-\alpha t}[1 - G(t)]dt}{1 - \frac{1}{m}}$$

where for our purposes, α is the birth rate, m is 2 for the number of children produced at each birth, and $G(t)$ is the generating function $1 - e^{-\alpha t}$.[5] Solving this equation in a general manner for our birth-only instances, yields a cumulative distribution function whose corresponding probability density function is $2\alpha e^{-2\alpha X}$, approximately the shape of the histogram. The fit of the data supports this with r_1 and r_2 being 16.031 and -16.032 (each twice the birth rate of 8).

2.2.3 Incremental Sequences

When hybrid events are included in the simulation, branch lengths and their corresponding sequences are evolved incrementally by the model. The first step of a hybridization event is to update the branch lengths and sequences of all active lineages, which may mean multiple intermediate updates of these attributes for one or more lineages. Although NetGen requests the update using the modified branch length and sequence (see Figure 2.12), it was important to verify such an approach did not alter the expected Hamming distance for the overall/total branch in such a situation. (Note that the expected Hamming distance is not necessarily equivalent to the product of the branch length and the sequence length. This is because bases are free to change and then revert back to their original state along a single branch.)

[5] See Chapter 6 of Harris[21] for more details on these parameters.

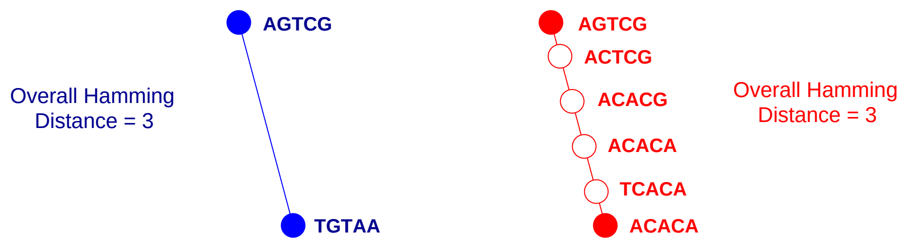


Figure 2.12: Two approaches to update branch lengths and sequences. On the left, a single update for the total branch is shown, while four intermediate updates are shown on the right. Although two identical start sequences are unlikely to yield the exact same finish sequences, the average overall Hamming distance for the two cases should be statistically similar.

An experiment was conducted using $1,000$ randomly generated start sequences (length = 2,000), a constant total branch length of 0.1, and the evolutionary model choice of HKY for Seq-Gen. The control case was where one request (for each start sequence) was given to Seq-Gen. For the second case, the total branch length was broken into $1,000$ calls, yielding separate branch lengths of 0.0001 each and using the returned result as the intermediate start sequence for the next call to Seq-Gen, mimicking the approach of NetGen. It is important to continue the "same" lineage for the purpose of hybridization and not simply regenerate a new random sequence for each update. The overall branch Hamming distances in each scenario resulted in normal curves as expected. The average and standard deviation for each are shown in Figure 2.13 and were statistically identical.

Although the previous experiment was important to show that Seq-Gen's performance was not altered by the use of multiple calls, it is also prudent to show that NetGen's behavior was similar under the two scenarios. Using a common set of input parameters (birth only tree with birth rate = 48, one chromosome with a pair of homologues each having length 1,000, and 1,000 extant taxa) ten runs (per scenario) using NetGen were executed. The first scenario operated in default mode where sequences were assigned only once to a *single* lineage when its birth event

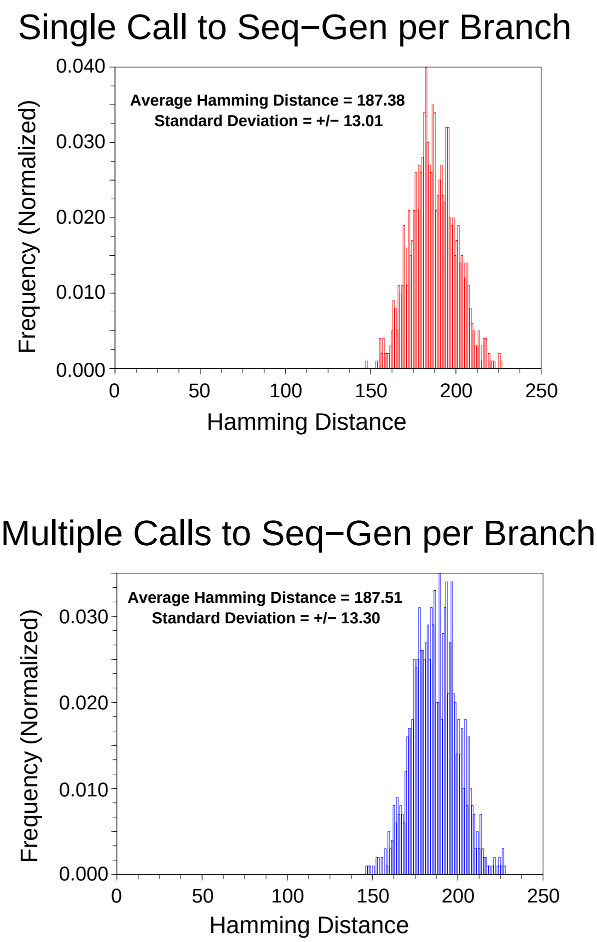


Figure 2.13: Similar results were obtained regardless of single or multiple update calls to Seq-Gen. The above histograms present the Hamming distance for the overall branch whether the sequence was created by a single call to Seq-Gen (top) or multiple shorter length calls to Seq-Gen (bottom).

occurred. In the second scenario, an additional control parameter invoked updates to both the branch lengths and their sequences for *all* active lineages whenever a birth event occurred anywhere in the topology – leading to frequent, intermediate updates of lineages. The resulting parsimony score was used to gauge the results.

The parsimony score of a tree or network is the Hamming distance along every

Parsimony Scores of NetGen Runs 1,000 extant taxa, birth rate = 48.0, total sequence length 2,000		
	Single Sequence Assignment	Multiple Updates to Sequence
	38855	40956
	40341	40607
	40556	42654
	42732	43297
	43699	39978
	41245	43721
	41804	41840
	41784	39310
	39461	41281
	42403	41784
avg	41288	41543
stdev	+/- 1504	+/- 1411

Table 2.2: Parsimony scores for the ten experimental birth-only trees generated by NetGen are similar under the two scenarios of branch and sequence updates (single vs. multiple calls to Seq-Gen).

branch of the topology for the sequence(s). In the case of a tree, this is calculated by summing the Hamming distance for each sequence along every branch.[6] For each scenario, Table 2.2 shows the parsimony scores for ten topologies and Figure 2.14 contains the Hamming distance distributions from a single run. The results indicate that the measures are not significantly impacted by the method of branch length and sequence updates.

2.3 Hybrid Model Characterization

As hybrids were a new event type added to the model, it was important to examine the behavior when such events were included in the simulation. Using two approaches

[6]For a network with diploid hybrids, not every homologue propagates along every branch, so some extra bookkeeping is necessary, but the idea remains the same.

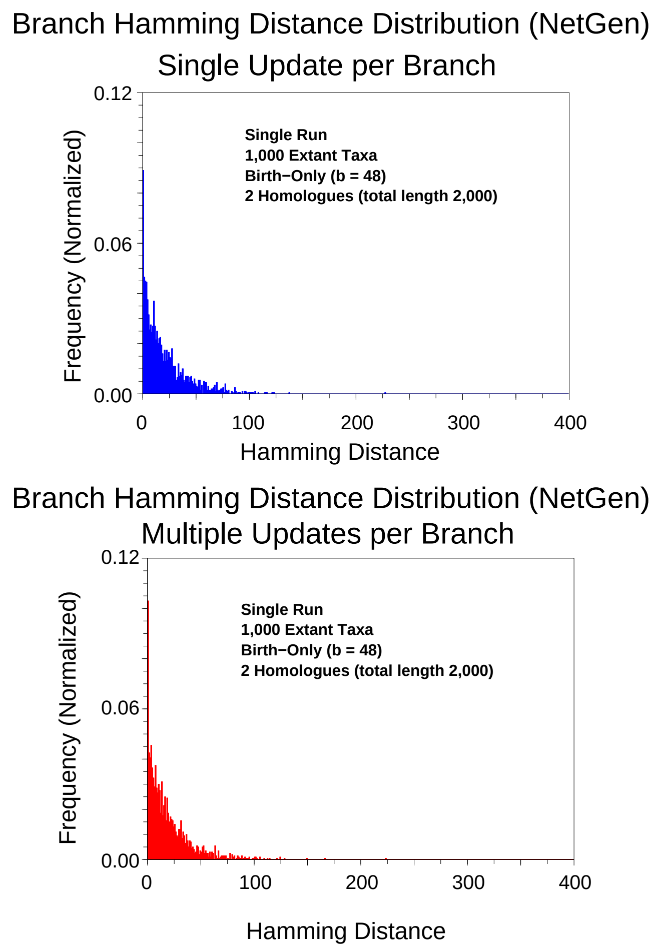


Figure 2.14: Similar results were obtained regardless of single or multiple update calls to Seq-Gen. The above histograms present the Hamming distances for the overall branches whether the sequence was created by a single call to Seq-Gen (top) or multiple shorter length calls to Seq-Gen (bottom).

from the validation section, scenarios were designed to examine the growth pattern (Section 2.3.1) and branch length distributions (Section 2.3.2). Being the most complex of the three events, hybridizations dominate the theoretical run time, details of which are presented in Appendix A.

2.3.1 Population Growth with Hybrids

The growth data (see Table 2.3 and Figure 2.15) show that the hybrid rate impacts the population growth in a manner similar to the birth rate presented in Section 2.2.1. This makes intuitive sense as the result of a hybridization is one new lineage created, just like a birth, and all previously scheduled events (as in the case of the second parent) are allowed to occur as previously planned.

Multiple NetGen Runs – 5,000 extant taxa		
Runs	Rates	Average Slope
10	B = 48 D = 0.0 H = 12	59.872 ± 0.872
10	B = 48 D = 5.0 H = 12	55.000 ± 1.663
10	B = 0 D = 0.0 H = 50	49.461 ± 1.103
10	B = 0 D = 0.0 H = 12	11.873 ± 0.264
10	B = 0 D = 0.0 H = 0.5	0.495 ± 0.011

Table 2.3: Average population growth for multiple runs of the single run scenarios presented in the figure below. Calculating the average slope as the natural log of the population size, labelled here as m, shows that birth and hybrid events impact the population growth of the model in the same manner. (Note that all simulations start with two active lineages originating from the root and the B, D, and H rates refer to all subsequent events.)

2.3.2 Branch Length Distributions with Hybrids

With respect to branch length distributions, shown for the base model in Section 2.2.2, we see a similar behavior with the addition of hybrids. Specifically the hybrid rate influences the *total* and *completed* branch length distributions as one would expect another birth rate would. Figure 2.16 shows both scenarios and the exponential fits for the two sets of branches. The fitted values, 7.997 and -7.996 for the total (completed + future) case and 16.049 and -16.040 for the completed-only scenario follow the same pattern discussed earlier. This indicates the model is behaving as expected.

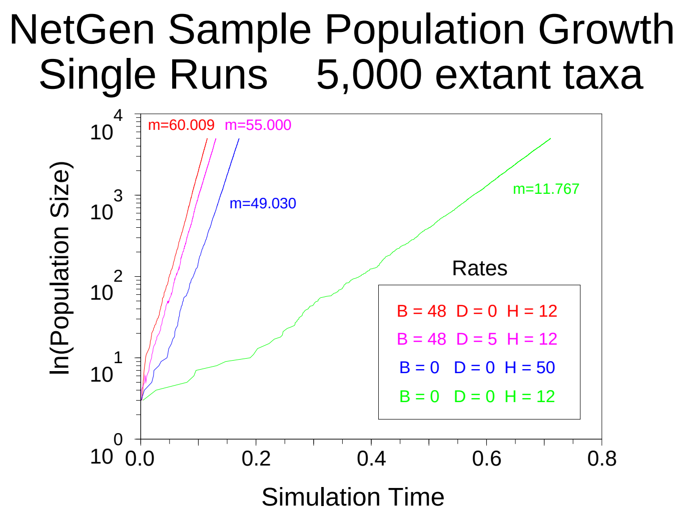


Figure 2.15: Growth statistics for a variety of runs that contain hybrids. (The run from Table 2.3 with the hybrid rate of 0.5 is omitted here as its slope is difficult to display alongside the others given it grows much more slowly over time.)

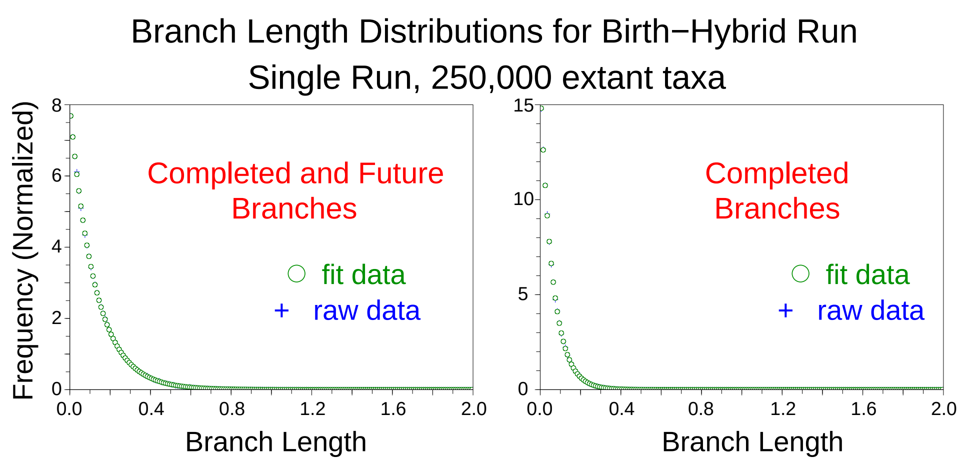


Figure 2.16: The hybrid rate contributes to the branch length distributions as a second birth rate in both the *total* (left) and *completed* (right) sets.

2.4 Representing Phylogenetic Topologies

The Newick format is a well-known text representation employed for phylogenetic trees. It employs nested parentheses to capture topological relationship information [2]. A post-order traversal of a tree topology is typically used to generate the string of text referred to as the “Newick format.” Taxa are separated by commas, and parentheses group siblings. The roots are implicitly identified by right parentheses and may also be explicitly named. Although this is a robust method for representing tree topologies, there are no provisions for addressing reticulate nodes and edges which are found in phylogenetic *networks*.

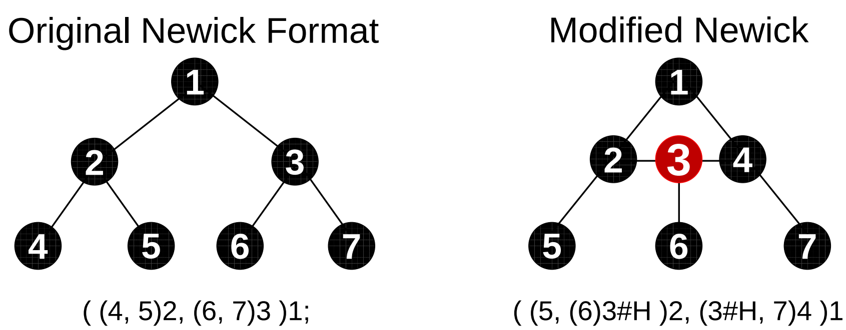


Figure 2.17: Original versus modified Newick formats. The phylogeny on the right contains the hybrid node (3) denoted by $\#H$ in the text string.

Motivated by a need to communicate topological information among our applications, we extended the Newick format to annotate reticulate nodes. Figure 2.17 provides examples of both the original and our *modified* Newick formats. Specifically a reticulate node, whether from a hybridization or a lateral gene transfer, is annotated with a `#H` or `#LGT` to indicate the nature of its reticulation. Although such a node appears only once in a topology, the format lists it multiple times – as a child for each of its parents. This permits a parser to identify the lineages being combined at a reticulate node and provides the necessary information for a network data structure. This format also lends itself to reticulate events deriving from any number of parents. In the case of a tree topology, the format is identical to that which the original Newick notation would produce. This should facilitate the adoption of the extension by software packages that currently process and display tree phylogenies.

A typical implementation of depth-first search (DFS), like the one found in [7], can be used to produce a post-order traversal. Adding constant requirements to print the necessary regular and modified Newick symbols as part of the traversal does not alter the order of the complexity. The complexity of DFS is $O(V + E)$, where V and E respectively refer to the number of vertices and edges found in the topology [7]. Appendix A provides the details of deriving upper bounds on these values for a given set of simulation inputs. The result is that DFS, and thus the regular and modified Newick formats, can be accomplished for NetGen topologies with a complexity of $O(n)$, where n is the number of extant taxa.

Chapter 3

Measuring Phylogenetic Networks

Having quantitative techniques to capture the characteristics of phylogenetic networks is essential for analyzing and comparing topologies that include reticulate events. The amount of similarity between a reconstructed network and its simulated (or source) network cannot be measured when methods assuming a tree topology break down or are incapable of dealing with the complexity of reticulation. Furthermore, experiments such as those undertaken in Chapter 5 are enhanced when networks can be quantitatively identified and categorized according to their reticulate properties.

Few measures have been developed for interspecific phylogenetic networks, as the study of these topologies is relatively new. One such measure is the tripartition, which was developed by Moret et. al. [43] as an extension of the well-established Robinson-Foulds bipartition measure for trees [63]. These measures capture the topological accuracy between either two trees (bipartition) or two networks (tripartition). Both of these techniques are reviewed in Section 3.1. In Section 3.2, we present three new measures that capture important properties of reticulate nodes which are useful for characterization purposes. Finally data illustrating the experimental behavior of these measures is discussed in Section 3.3.

The network measures presented in this chapter are generic in that they can be used for networks containing lateral gene transfer or hybridization nodes, although the focus of this current work is on the latter. Software called NetMeasure, implementing these measures, has been developed and is available from the author, under GNU General Public License, at `http://www.cs.unm.edu/~morin/`.

3.1 Robinson-Foulds (RF) Distances

The Robinson-Foulds (RF) distance is a well-established and accepted method for comparing two phylogenetic trees [63]. The measure is a topological one, calculating the similarity of the branches between two topologies. Due to how the measure is defined, both topologies must be trees and have the same set of extant taxa.

In theoretical terms, a tree can be analyzed as a graph comprised of nodes and edges. When a single edge is removed, the graph becomes separated into two pieces, and is called a bipartition . An edge that bipartitions a graph causing one node to be isolated from all the others, is often called a "tip." With the RF measure, the set of tips for each graph will be identical, as the two topologies must have the same set of extant taxa. Therefore these edges are considered trivial, and are not part of the distance calculation. However, all other edges are part of the internal structure and are regarded as non-trivial. The bipartitions induced from the removal of these non-trivial edges form the basis of the RF measure, which can be computed in linear time using what is known as Day's algorithm [8].

The definition of RF distance used for our purposes is calculated by the following steps:

1. For each tree, list the bipartition induced for every non-trivial edge. Namely,

identify the two sets of extant taxa that are created by removing the edge.

2. Compare the two lists of bipartitions to identify which edges have a matching counterpart in the other topology. A "matched" edge is one that creates the same bipartition in each topology.

3. Count the number of edges that exist in the first tree, but do not have a match in the second (false negatives).

4. Count the number of edges that exist in the second tree, but do not have a match in the first (false positives).

5. Divide the false negative and false positive counts each by the number of internal edges for their respective tree – this yields the rates of each.

6. The RF distance is calculated as the average of these two rates (sum the rates and divide by 2).

Figure 3.1: A one non-trivial edge case of Robinson-Foulds distance. The bipartition of extant taxa induced when the red edge of the tree on the left is compared to that of the blue edge of the tree on the right yielding an RF distance of 1.0.

Figure 3.1 shows a simple, one edge case. Here there is only one non-trivial edge in each tree to consider (identified in red and blue). When the edge is removed, the bipartition of extant taxa in the left tree is $(A, B) vs. (C, D)$. Whereas in the right tree, the bipartition is $(A, C) vs. (B, D)$.[1] In this case, there are no matches for either edge in the other topology – leading to an RF distance of 1.0.

[1]When the topologies are unrooted, as in the example, the order of sets for the bipartition induced by an edge removal does not matter (e.g. $(A, B) vs. (C, D)$ is a match for $(C, D) vs. (A, B)$). However with rooted topologies, the order does matter as it indicates

This definition of RF distance is based on the one employed by [43] and results in a range of values from 0.0 to 1.0. A value of 0.0 for a pair of trees indicates isomorphism and all edges in one topology have a counterpart in the other topology. A value of 1.0 on the other hand, means that there are no edges in common between the two topologies. It should be noted that some authors (e.g [12, 51, 76]), and one of the seminal papers [63], use slightly different definitions of the RF distance. Some define the distance as a simple count of false positives and false negatives (not converted into rates) and may or may not divide by two to average the value. While these definitions are valid, the value ranges are not necessarily constrained to the 0.0-1.0 interval which is desired for compatibility purposes with our other measures.

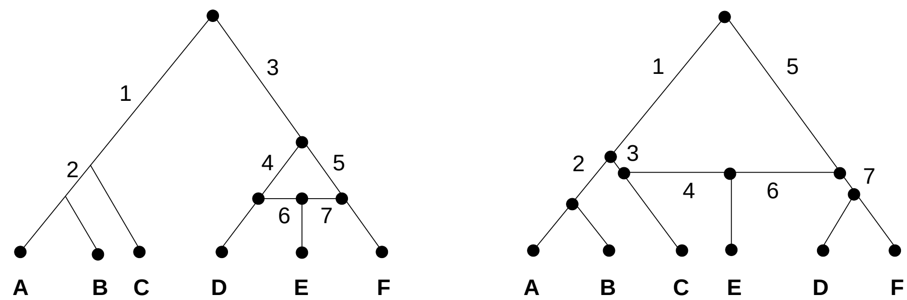
The tripartition measure, developed by Moret et. al. [43], is an extension of the Robinson-Foulds distance from trees to networks. Like the original RF distance, this measure attempts to match corresponding edges between two topologies. However, rather than the removal of each edge inducing a bipartition, a tripartition of extant taxa is created. The three sets induced by each edge removal are:

- extant taxa reachable only from the root via the selected edge,
- extant taxa reachable from the root using a path via the selected edge, and using a path *not* via the edge, and
- extant taxa *not* reachable from the root via the selected edge.

Figure 3.2 is an example based upon [43] and shows two networks with their corresponding tripartition listing. The measure is defined as $\frac{FN+FP}{2}$ where:

$$FN = \frac{\text{Number of edges in } N_1 \text{ that do not have matches in } N_2}{\text{Number of non-trivial edges in } N_1} \tag{3.1}$$

which set is disjoint from the root and which is connected to it (e.g. $(A,B) vs. (C,D)$ is not a match for $(C,D) vs. (A,B)$).



Network N_1		Network N_2	
Edge	Tripartition	Edge	Tripartition
1	$\langle\{A,B,C\},\emptyset,\{D,E,F\}\rangle$	1	$\langle\{A,B,C\},\{E\},\{D,F\}\rangle$
2	$\langle\{A,B\},\emptyset,\{C,D,E,F\}\rangle$	2	$\langle\{A,B\},\emptyset,\{C,D,E,F\}\rangle$
3	$\langle\{D,E,F\},\emptyset,\{A,B,C\}\rangle$	3	$\langle\{C\},\{E\},\{A,B,D,F\}\rangle$
4	$\langle\{D\},\{E\},\{A,B,C,F\}\rangle$	4	$\langle\emptyset,\{E\},\{A,B,C,D,F\}\rangle$
5	$\langle\{F\},\{E\},\{A,B,C,D\}\rangle$	5	$\langle\{D,F\},\{E\},\{A,B,C\}\rangle$
6	$\langle\emptyset,\{E\},\{A,B,C,D,F\}\rangle$	6	$\langle\emptyset,\{E\},\{A,B,C,D,F\}\rangle$
7	$\langle\emptyset,\{E\},\{A,B,C,D,F\}\rangle$	7	$\langle\{D,F\},\emptyset,\{A,B,C,E\}\rangle$

Figure 3.2: An example of tripartition information for two network topologies based upon [43]. The internal edges labelled in the topologies correspond to the table at the bottom. The tripartitions induced by each edge removal are listed in order of the sets discussed previously. The final tripartition value is $\frac{4}{7}$.

$$FP \quad = \quad \frac{\text{Number of edges in } N_2 \text{ that do not have matches in } N_1}{\text{Number of non-trivial edges in } N_2}. \tag{3.2}$$

For the example in Figure 3.2:

- edges 1, 3, 4, and 5 of the left network have no corresponding edges in the network on the right, and
- edges 1, 3, 5, and 7 of the right network have no corresponding edges in the network on the left.

This leads to:

$$FN = \frac{4}{7}, \quad FP = \frac{4}{7}, \quad \text{and a tripartition measure of: } \frac{4}{7}. \tag{3.3}$$

In general, tripartition scores can range from 0.0 to 1.0, like the RF distance. At the extremes, a value of 0.0 indicates two isomorphic (or "indistinguishable" as defined in [43]) networks and 1.0 results when there are no non-trivial edges in common. A tripartition score of $0.x$ means that $x\%$ of the branches are mismatched.

3.2 New Topological Measures

In this section, we present three new measures: reticulate timing, reticulate impact, and reticulate diversity. Each is designed to capture topological features about a single reticulate node, and when multiple reticulate nodes exist in a network, the measure is repeated on an individual node basis. Expected run times for these measures are discussed in Appendix A.

3.2.1 Reticulate Timing

Figure 3.3 illustrates the concept of the reticulate timing measure on a simple topology containing one hybrid node. To capture whether a reticulate node happened relatively early or late in a given phylogeny, a ratio of the time at which the event occurred and the height of the network is calculated.

This measure is defined using clock time, as opposed to evolutionary time, because clock time progresses for all the lineages at the same pace, whereas evolutionary time can vary if the network is non-ultrametric and/or contains lineages with varying rates.

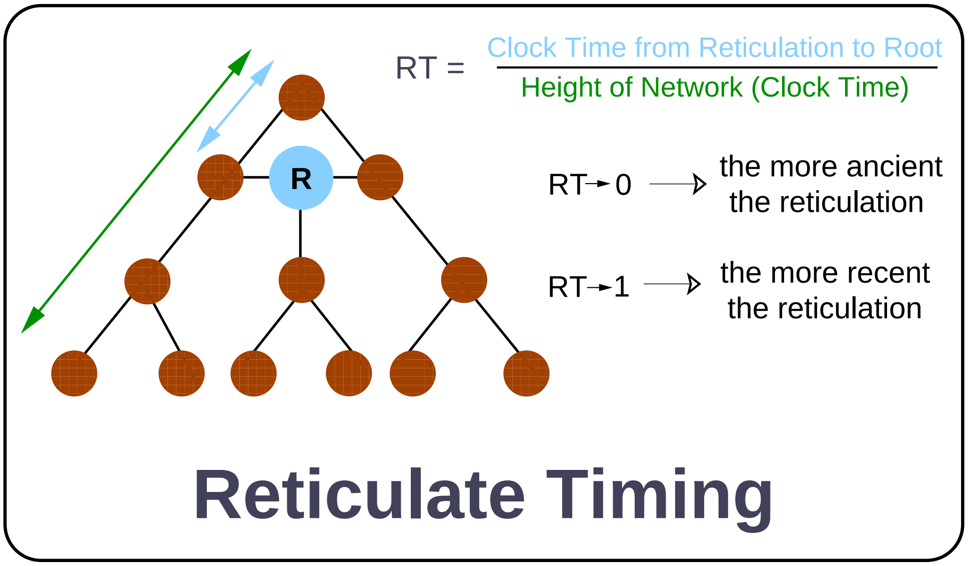


Figure 3.3: Reticulate timing is a ratio of two clock values – the time at which the reticulation occurred and the height of the network. The limits of this value are 0.0 (most ancient occurrence) to 1.0 (most recent occurrence).

This measure is useful when characterizing existing topologies as one does for source networks in a study such as Chapter 5.

3.2.2 Reticulate Impact

The impact measure was developed to provide a sense of how much influence a specific reticulate node has on the given topology. This measure, illustrated in Figure 3.4, is defined as a ratio of the number of extant taxa that are descendants of the hybrid node to the total number of extant taxa.

Of the three new measures, reticulate impact has the greatest potential to be a component of phylogenetic reconstruction efforts. If the extant offspring of a hybridization speciation is characterized by a specific trait, leading to the identification of the impacted extant taxa set, the reticulate impact score can be calculated. The

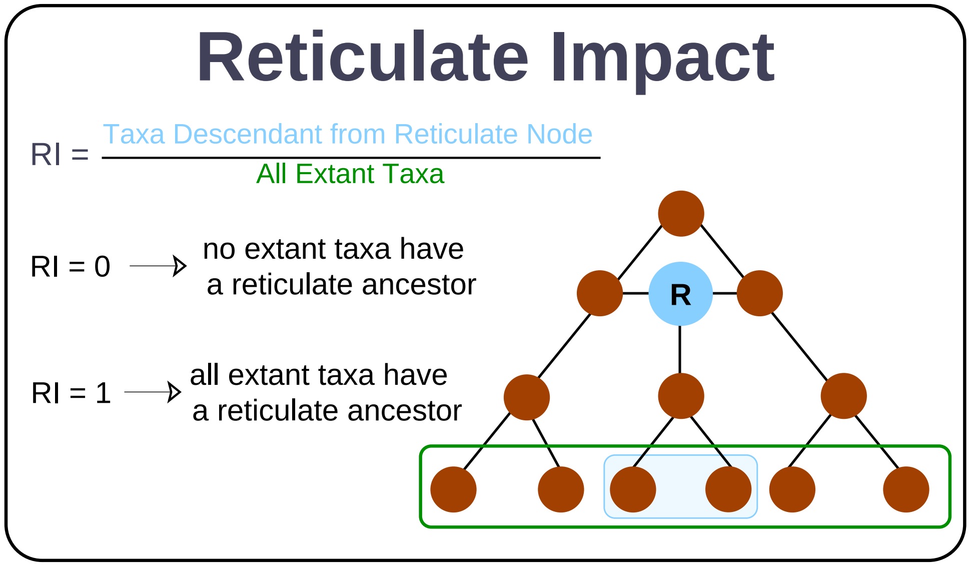


Figure 3.4: Reticulate impact measure is a ratio of extant taxa descendant from the hybrid node and the number of extant taxa in the phylogeny. The value can range from 0.0 to 1.0.

impact information can then be capitalized on for reconstruction purposes, as is the case with our NetReconstruct algorithm in Chapter 4.

3.2.3 Reticulate Diversity

Diversity was the most difficult topological notion to capture quantitatively. This goal required finding a value that at one extreme would indicate sister taxa as the parents of a hybrid node, while the other end of the spectrum would point to parents as diverse as possible within the given topology.

By combining counts from the concepts of impact (found in the previous subsection) and the most recent common ancestor (mrca)[2] for the parent nodes, it is

[2]Note that clock branch lengths are the most accurate measure for determining the mrca, as clock time remains constant for all lineages, unlike evolutionary time which may vary from lineage to lineage.

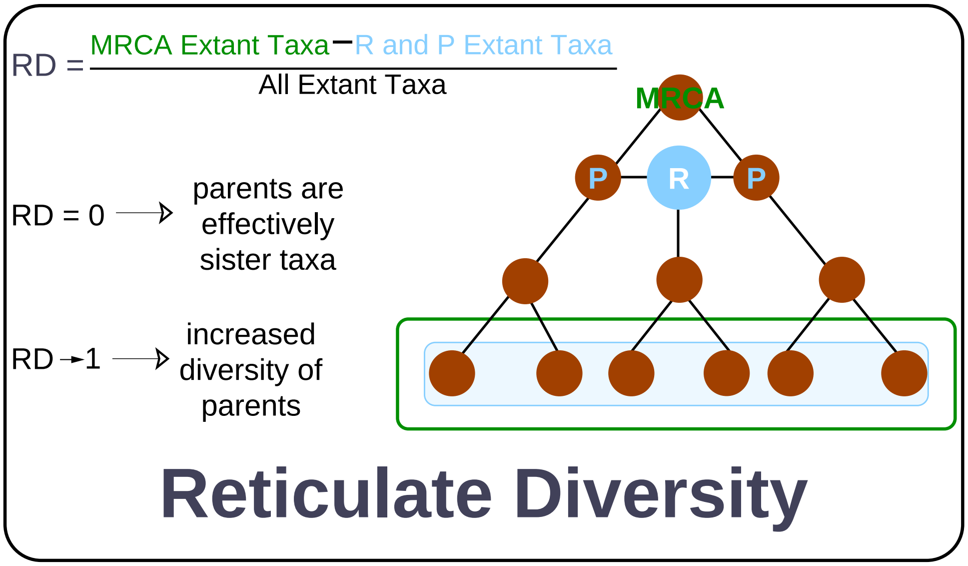


Figure 3.5: The reticulate diversity measure involves counting the difference in extant taxa between the most recent common ancestor (mrca) and the hybrid and its parents which is then scaled (divided by) the number of extant taxa in the topology. The resulting value can range from 0.0 to 1.0, where 0.0 indicates the parents are essentially sister taxa and 1.0 reflects extremely diverse parents.

possible to capture the extent to which the location of the parents differ. Figure 3.5 shows a case where the parents are sister taxa and a diversity value of 0.0 results. Like the timing measure, this quantity is most useful when characterizing source networks for which the topology is known, perhaps as part of a study such as the one presented in Chapter 5.

3.3 Experimental Behavior of Measures

Experimental results are presented in this section that characterize the measures described in the last section. Section 3.3.1 focuses on bipartition and tripartition scores for random topologies in order to establish a baseline against which future reconstruction algorithm results can be compared. Interdependencies among our three new measures (timing, impact, and diversity) are examined in Section 3.3.2. It is useful to perform these analyses and review their results before proceeding to the new reconstruction technique presented in Chapter 4.

3.3.1 Characterization of Random Robinson-Foulds Distances

Without a baseline, it is difficult to assess quantitatively the performance of a reconstruction algorithm. For example if a new technique consistently yields, an average tripartition score of 0.5, is that to be considered good, bad, or just average? In order to address this situation, characterization experiments were conducted on well-specified, yet random, topologies.

Our first set of experiments captured the Robinson-Foulds distances for trees created by NetGen. One hundred and fifty random birth-only trees were created and then the RF distance was calculated for every pair (11,175 unique pairs avoiding symmetric ones). As the bipartition measure requires that the two topologies have the same set of extant taxa, it was necessary to relabel the extant taxa of every topology. This was accomplished by going through the Newick representation and changing each extant taxon label to a value between one and the total number of extant taxa. Runs were executed for three very different birth rates (0.2, 1.3, and 48) and two sizes of extant taxa (8 and 50) to ensure that these parameters did not alter the outcome. Figure 3.6 contains the histograms for these six cases and all show a distribution space that is exponential towards one. The results for the 50 extant case were so dramatic that the bin size for the histogram had to be reduced by a factor of 10 (0.1 to 0.01) in order to illustrate its exponential nature.

We repeated the above experiment for networks with exactly one hybrid. The topologies were created using NetGen and relabelled appropriately, but the experiment was doubled by using two different hybrid rates to assess any impact this

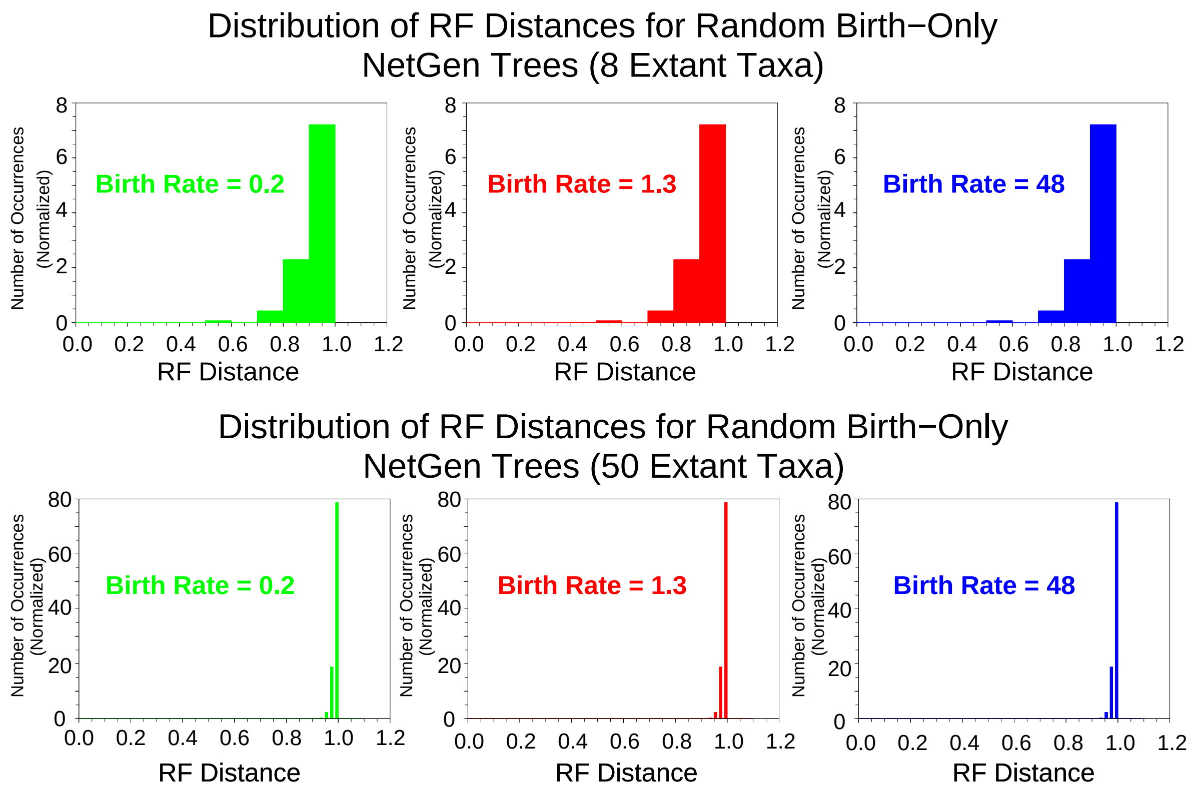


Figure 3.6: Frequency of bipartition scores for 150 random birth-only trees as generated by NetGen and measured with NetMeasure. The top row is for 8 extant taxa, while the bottom row shows runs with 50 extant taxa.

parameter would have on the results. The hybrid rates were chosen as 12 and 100 (with a maximum hybrid count of 1) and 200 topologies were generated in an attempt to ensure a sufficient number of pairs on which to compute the measure. In the case with the lowest birth and hybrid rates (b = 0.2 and h = 12), 153 of the 200 topologies contained one hybrid leading to 11,628 pairs used for computing the tripartition measure. With the other rate combinations, all 200 topologies met the one-hybrid criteria, and therefore there were 19,900 non-symmetric pairs on which to complete the extended-RF distance. The results shown in Figure 3.7 are similar to those of the bipartition score with an exponential bias towards 1.0 indicating no branches in common. Once again, the 50 extant taxa case required a greatly refined bin size to show the nature of the distribution, but did not reveal any differences and that data is omitted here.

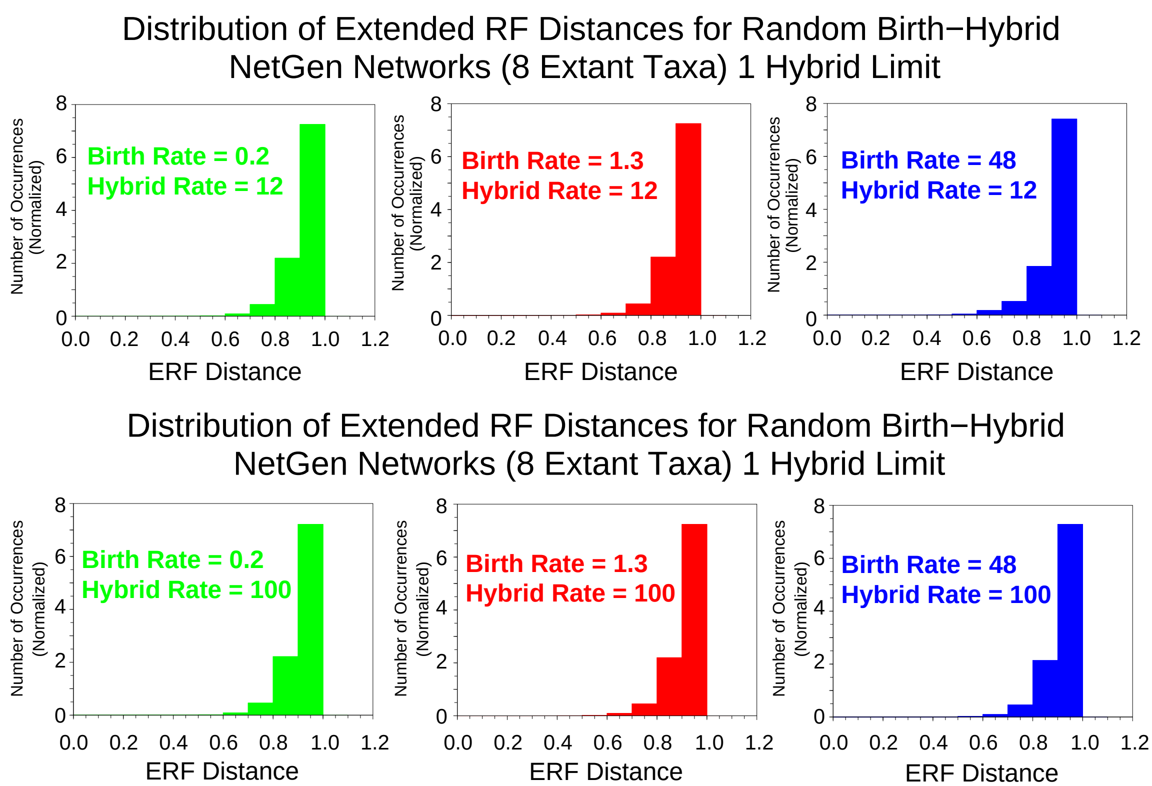


Figure 3.7: Frequency of tripartition scores for 200 random birth-hybrid networks with exactly one hybrid as generated by NetGen and measured with NetMeasure. The top row is for runs with a hybrid rate of 12, while the bottom row shows runs with a hybrid rate of 100.

3.3.2 Diversity, Timing, and Impact Dependencies

Although our new measures were motivated by the desire to characterize reticulate nodes in a topological fashion, it seemed probable that such properties would exhibit dependencies especially in scenarios that did not contain extinct lineages (death rate equal to 0.0). The dependencies explored were: impact and timing, diversity for different methods of second parent selection, and diversity and timing.

The data set for the first experiment was birth-hybrid networks generated by NetGen with exactly one hybrid, where the second hybrid parent was chosen by the minimum Hamming distance option. The parameters for size of extant taxa (50) and birth rate (b = 1.0) were kept constant across the scenarios, while we varied the hybrid rate (0.05, 0.1, 0.5, and 0.9) to promote the placement of the hybrid in different regions of the topology. Although 10,000 topologies were generated, some rate combinations did not produce the full set of networks meeting the one hybrid requirement, but at a minimum there were 9,000 in each scenario.

The contour plots of Figure 3.8 show there is a connection between reticulate impact and timing measures. The earlier the hybrid occurs, the greater impact it is likely to have on the extant taxa. One can see that the peak of the hybrid timing moves along the x-axis from the right to the left as the hybrid rate gets larger, causing the hybrid to occur sooner in the simulation. These results make intuitive sense as one would expect hybrids that occur earlier, as opposed to later, in the simulation to have a better chance of producing more offspring, thus increasing the reticulate impact score.

The second experiment was to examine in what manner the diversity measure was affected by how the second parent for a hybridization was chosen. The two extremes (minimum Hamming distance and random) were examined. In order to enhance any behavior that might be exhibited, the extant taxa size was set to 500 and the rate combination was chosen to be birth = 1.0 and hybridization = 0.05 with a limit of one hybrid per network. This rate combination has the hybrids occurring later in the simulation where there is more of a difference to be noticed in the choice of a second

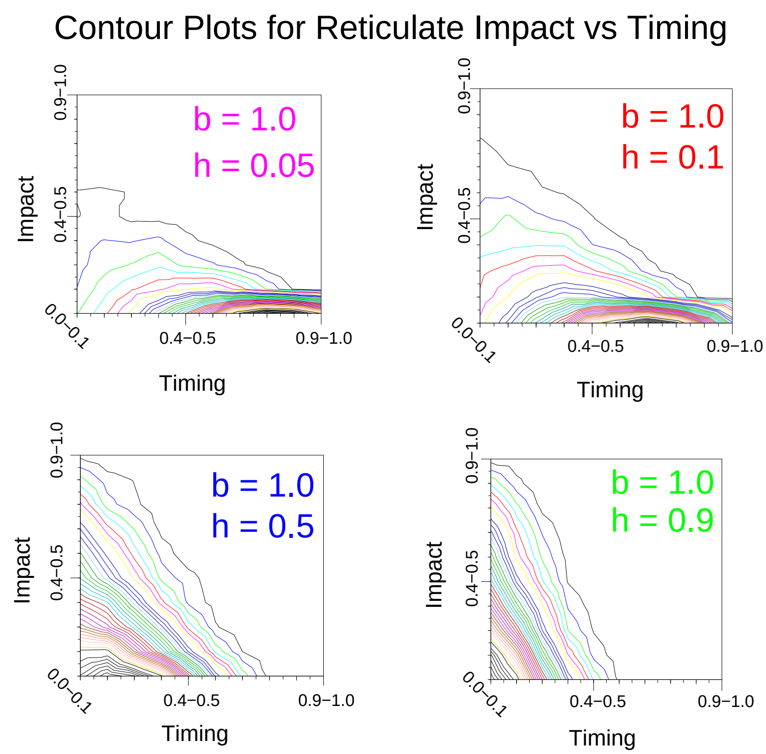


Figure 3.8: Contour plots for reticulate impact vs. reticulate timing measures from birth-only (b = 1.0) networks with one hybrid, but varying hybridization rates. The relationship of impact being greater for early hybrids can be seen across all four plots.

parent. (In the case of only one hybrid occurring early in the simulation, the choice for a second parent is limited regardless of the method employed.) As expected, Figure 3.9 shows the diversity scores in the random case have a much greater spread, whereas the minimum Hamming distance chooses second parents topologically close to the first.

The third and final experiment was to look at the interaction between the diversity and timing measures for the same two hybrid parent choices as above (minimum Hamming distance and random). Using the same parameters (500 extant taxa,

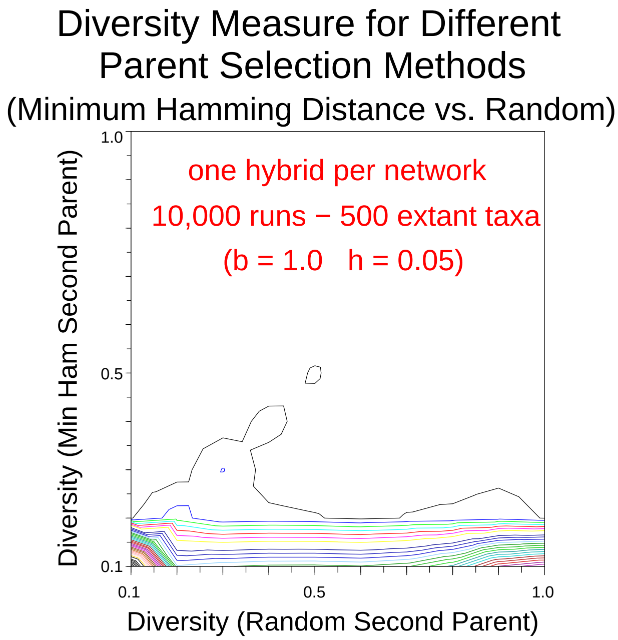


Figure 3.9: This contour plot shows the two dimensional histogram for diversity scores. The y-axis contains diversity scores from NetGen runs where the second hybrid parent is chosen according to the method of minimum Hamming distance and the x-axis scores comes from runs where the random method was utilized. Not surprisingly, the results are much tighter and have smaller values in the minimum Hamming distances case.

10,000 runs, b = 1.0, h = 0.05, limit of one hybrid per topology), NetMeasure was used to calculate the diversity and timing measures from runs for the two types of parent selection methods. The plot on the left of Figure 3.10 shows the expected result that when the second parent is chosen on the basis of Hamming distance, the diversity is likely to be small. When the second parent was chosen randomly, the data behavior is more complex. If the hybridization occurs relatively early in the simulation, indicated by a small value on the x-axis, the diversity is likely to also be small. This makes intuitive sense, because although the second parent is chosen randomly, there is not much diversity early in the simulation. At the other extreme of timing, with hybridizations occurring relatively late in the simulation, the diversity score is high. This is consistent with the fact that at this point of the simulation, the odds are high that the second parent will be in a completely different subtree than the first, causing the most recent common ancestor to be the root, thus yielding a high diversity score. In the middle ground, two factors are competing – time and diversity. While there is certainly plenty of diverse second parents to choose (and the odds are high such a choice will be made) there is still plenty of time for the hybrid offspring to propagate, which helps to overcome the gap in how many of the final extant taxa are impacted by the most recent common ancestor and the hybrid and its parents – thus decreasing the overall diversity score.

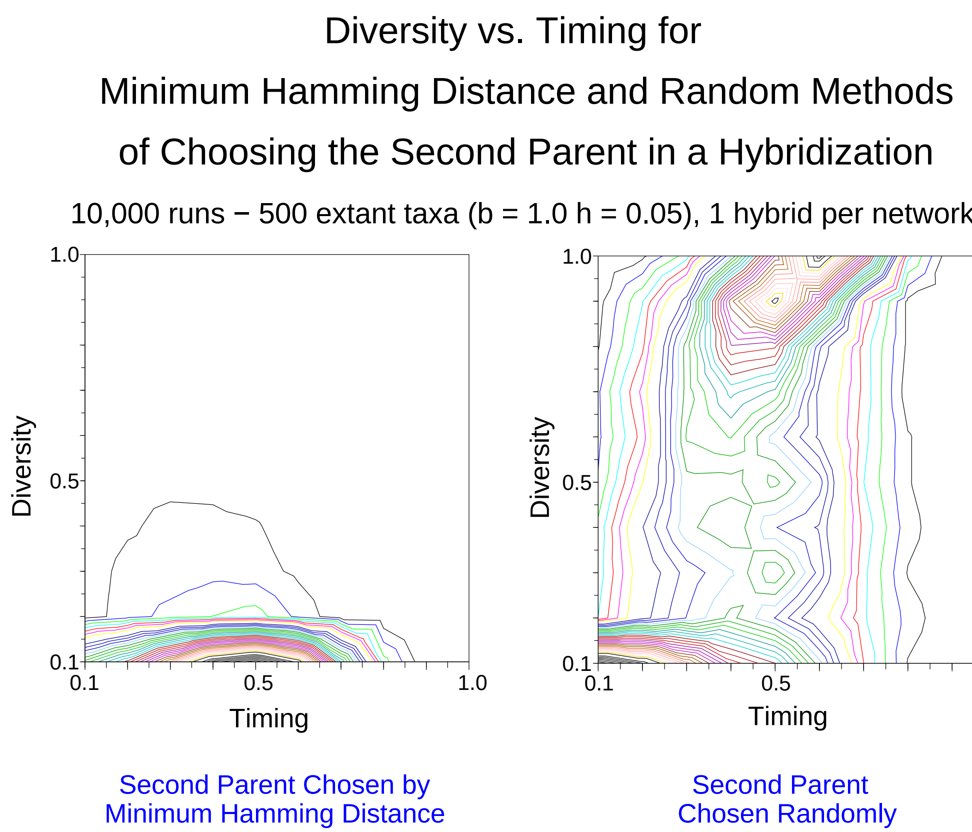


Figure 3.10: The left plot illustrates that regardless of the timing, the minimum Hamming distance option for selecting the second parent restricts the diversity scores as one would expect. The low-density bump reflects the fact that most hybrids have a mid-range score, and as time has advanced enough for lineages to be diverse, there are a few instances where the diversity score is higher than average. The effect of random selection for the second hybrid parent is illustrated here in the right plot. Early in the simulation, it is difficult to have extremely different parents and towards the end of the simulation the odds make it difficult to have similar parents. These influences result in the high-density regions found at the top and bottom of the graph. The middle area shows the trade-off in the diversity measure between having a high likelihood of choosing a substantially different second parent, but also time to generate more offspring decreasing the difference in the number of extant taxa impacted.

Chapter 4

Reconstructing Phylogenetic Networks with Single-Diploid-Hybrid Events

A primary goal of this research is to reconstruct phylogenetic networks, as well as to generate and measure them. Given the complexity that multiple reticulate nodes contribute to a network, we have decided to focus our efforts on topologies with one diploid-hybrid event. Building upon existing ideas and software (e.g. Fitch small parsimony [11] and PHYLIP [10]) for tree reconstructions, we developed an algorithm to infer a phylogenetic network with a single-diploid-hybrid event from a given set of extant taxa, an outgroup, and hybrid-impact information. Our reconstruction software, NetReconstruct is available from the author, under GNU General Public License, at `http://www.cs.unm.edu/~morin/`.

In Section 4.1, we present the design of the reconstruction algorithm. Details pertaining to the implementation and use of NetReconstruct are provided in Section 4.2, and experimental results on the performance of the algorithm are given in Section 4.3.

4.1 Reconstruction Algorithm

Our reconstruction algorithm is comprised of three subtree stages and builds a network with a single-diploid-hybrid event. At a minimum, tree reconstruction algorithms require a set of extant taxa and their sequences, and an outgroup if a rooted topology is desired. In addition to these standard inputs, NetReconstruct requires a hybrid-impacted set that identifies the extant taxa believed to be descended from the ancestral hybrid. This requirement was inspired by the reticulate impact measure presented in Chapter 3. The foundation for this approach is that biologists are often able to identify an extant taxon or taxa as having hybrid origins and are sometimes able to propose the parental lineages that formed the hybrid [33, 34, 77]. However, easily determining whether a group of extant taxa descended from a single hybridization event is not yet possible, although biologists are constantly developing new techniques that have the potential to provide this information [23, 61]. In the interim, alternative ideas for determining a set of hybrid-impacted extant taxa, to be used with NetReconstruct, are provided in Appendix B. Although it may be tempting to use the results of any reconstruction algorithm as a final answer, it is important to remember that techniques such as NetReconstruct are designed for simulation and algorithm research. Hence, the intention is that they be used iteratively and in conjunction with other biological data and knowledge.

The first stage of the reconstruction requires generating the “hybrid subtree.” Starting with the sequences of the extant taxa of the hybrid-impact set and an outgroup, a call is made to PHYLIP to infer a tree. When the topology is returned, sequences are assigned to all the internal nodes using the technique known as Fitch small parsimony.[1] The outgroup is removed, and the root of this tree is labelled as the hybrid. The last step of this stage is to add the structure surrounding the hybrid. Taking the sequences assigned to the hybrid, we randomly split the homologues (sequences) to the two newly created parent nodes. The major steps of this stage are illustrated in Figure 4.1.

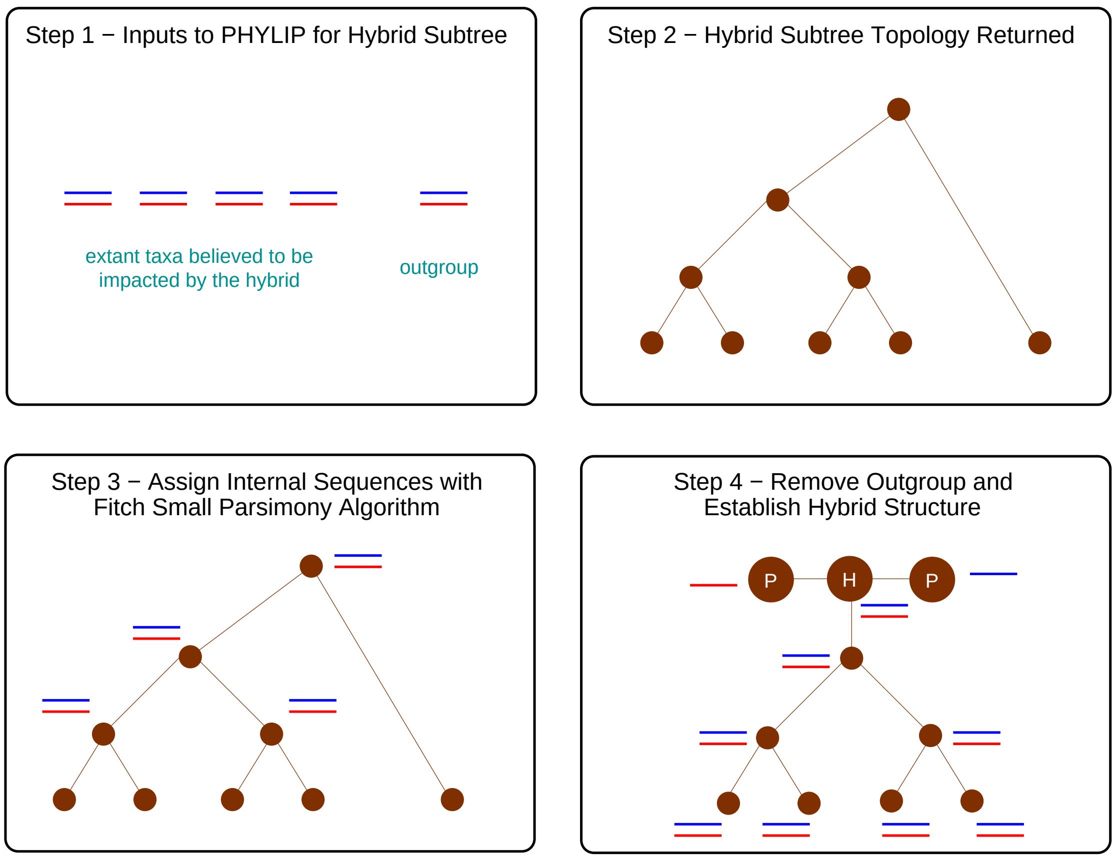


Figure 4.1: The primary steps of Stage 1 for the reconstruction algorithm. Starting from the hybrid-impacted extant taxa and an outgroup, the hybrid subtree topology, sequences and related structure are inferred.

[1]Only the preliminary phase of the small parsimony algorithm is utilized as the final rule phase allows the generation of *all* parsimonious assignments and adds realism to internal nodes provided specific evolutionary assumptions are applicable. As these assumptions may or may not be appropriate for our reconstructions and the sequences of the root do not change in this phase, only the preliminary one is required.

Constructing the "parental subtree" is the second stage of the algorithm and is depicted in Figure 4.2. With an outgroup and the identification of which extant taxa will be the descendants of the hybrid's parents, a single subtree for these taxa is inferred once again using PHYLIP (though sequences for these internal nodes are not assigned until later). After removing the topological structure related to the rooting of this subtree, one or more subtrees will remain. For each of these subtrees, it must be determined from which parent to descend. This is decided by calculating an overall average Hamming distance between all extant taxa in the subtree and each of the two parents of the hybrid node. The closest parent inherits the subtree as one of its children.

Currently there are four different approaches for identifying the set of extant taxa that will be affiliated with the hybrid's parents – custom, extreme custom, closest neighbor, and closest $2x$ hybrid. The two custom options (custom and extreme custom) are best-case scenarios where the parental extant taxa are provided as input (and further split into two parental sets in the extreme case), thus meaning the three sets of extant taxa (hybrid-impacted, descendants from hybrid's parents, and remaining) are all known a priori. Although it is not expected that these approaches can be used in real-world cases, they do provide a good baseline comparison when dealing with reconstructions from known source networks and analyzing results. The closest neighbor option, which is the default, identifies the extant taxon with the smallest Hamming distance, for all extant taxa that are *not* impacted by the hybrid. If an extant taxon's closest neighbor is descendant from the hybrid, the original taxon is added to the parental descendant group, otherwise it will become a part of the third (remainder) subtree. The final option is the closest $2x$ hybrid approach where x refers to the number of hybrid-impacted extant taxa. This method was designed with the intuition that in a perfect binary network and the idealized case of the same number of levels of offspring reproduction for both the parents and the hybrid, there should be twice as many extant taxa descendant from the parents as there are from

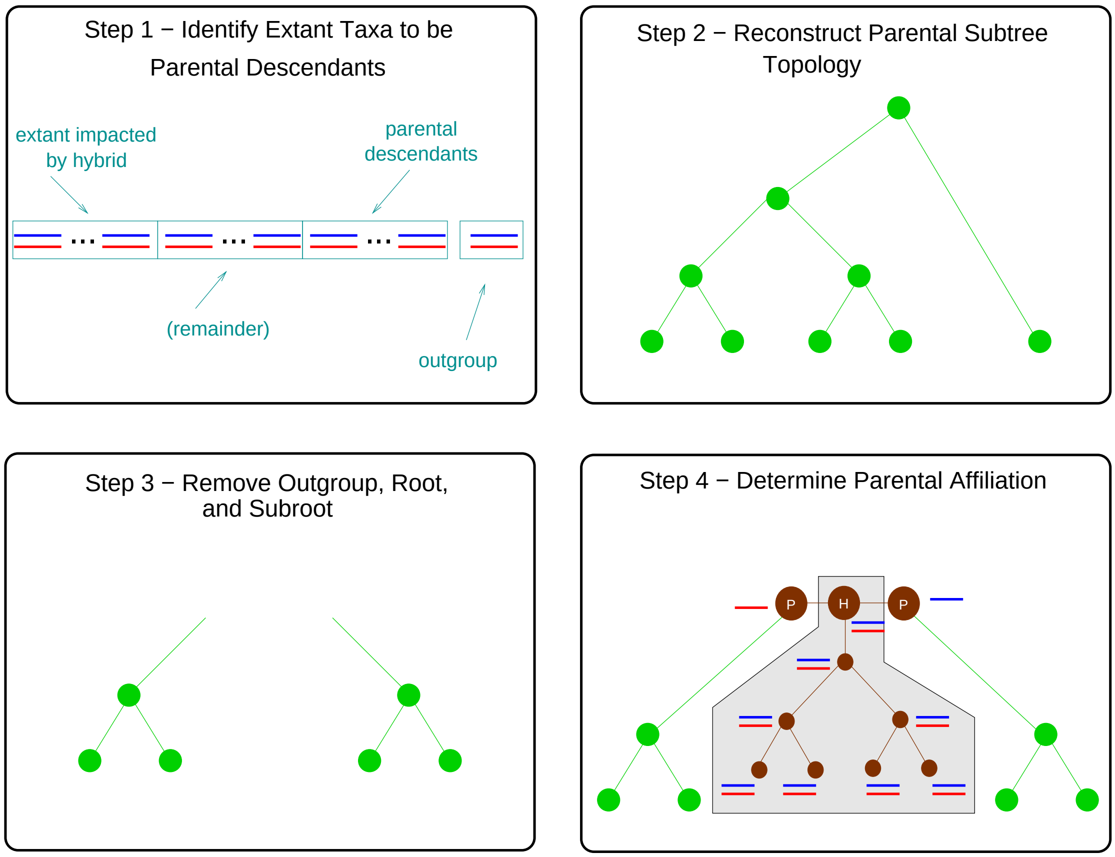


Figure 4.2: The primary steps of Stage 2 for the reconstruction algorithm. After parental descendants are identified, a subtree for those extant taxa are reconstructed. The resulting subtrees are assigned to one of the parents of the hybrid – connecting the subtree from Stage 1 and this one. The greyed portion of the figure was completed in Stage 1.

the hybrid node itself. For this method, Hamming distances are used again, but this time they are calculated between all those extant taxa not impacted by the hybrid and the hybrid itself (not its descendants), whose sequences were determined in Stage 1. Then the $2x$ extant taxa with the smallest Hamming distances are labelled as parental descendants.[2]

[2]In order for this approach to work, there must be a sufficient quantity of extant taxa not impacted by the hybrid.

In the last stage of the reconstruction, the remainder of the network and final sequence assignments are pieced together. Using all the remaining extant taxa not assigned to the hybrid and parental subtrees, in addition to the outgroup and the hybrid's parental nodes, PHYLIP is called one last time to reconstruct a tree. At this point, the topology is set and what remains to be completed is the sequence assignment for the internal nodes of Stage 2, Stage 3, and the hybrid's parents (which are missing half of their sequences). (The hybrid and its offspring already had their sequences set in Stage 1.) These last steps are shown in Figure 4.3.

4.2 NetReconstruct Software Details

An outgroup is an important part of a reconstruction effort because it helps root the topology. NetReconstruct allows for the input of two, possibly different, outgroups. The first is a general outgroup that is used for reconstructing the parental and remaining subtrees in stages 2 and 3. A second outgroup must also be specified for the hybrid subtree inferred in Stage 1. While it is most likely that the same extant taxon will be used for both outgroups, the option of specifying a separate outgroup for the hybrid subtree seemed useful for domains where there is considerable knowledge about the extant taxa impacted by the hybrid.

Another consideration when involving an outgroup is whether the topology is considered to be rooted or not (see Figure 4.4). As phylogenetic networks must be rooted to ensure that parents of any hybrid event exist at the same time, all of our results are considered rooted. However, during its process, NetReconstruct can encounter unrooted subtrees (e.g. those returned by PHYLIP that when regarded as rooted, appear to have a polytomy at the root node, instead of the expected root and sub-root). With the outgroup known, this situation is automatically adjusted (though an input option to override it can be invoked) to avoid misinterpreting

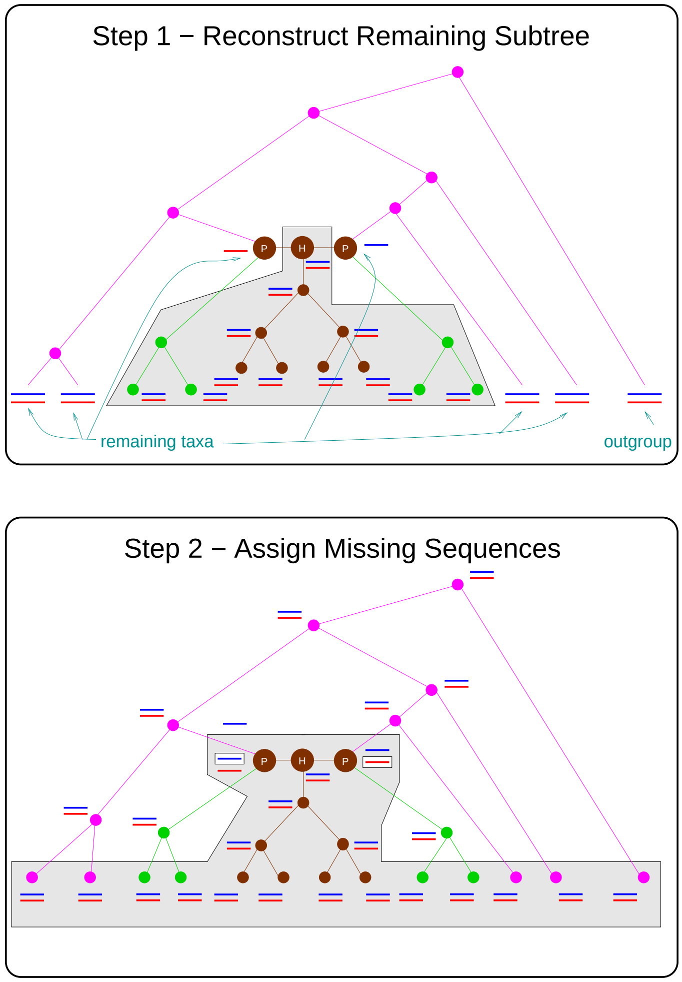


Figure 4.3: The primary steps of Stage 3 for the reconstruction algorithm. Once the remaining topology is constructed missing sequences are assigned, including the halves for the two parents of the hybrid. The greyed portions of the diagrams indicate work completed in the prior stages and is not altered further.

topology information.

As mentioned in the previous section, PHYLIP was chosen to perform the *tree* reconstructions at each of the three stages of our algorithm. However, a simple baseline tree reconstruction technique was also desired in order to assess any improvement our algorithm would show over a random approach. Therefore we have

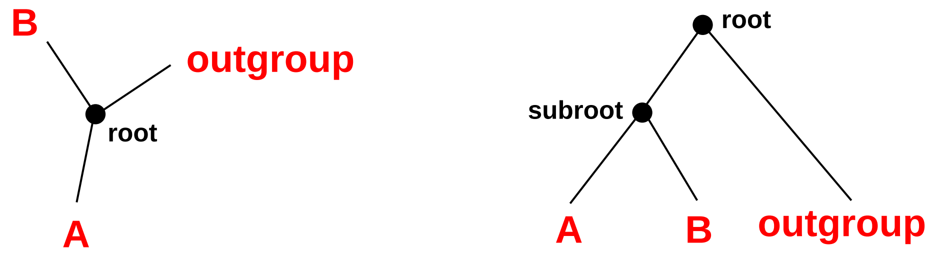


Figure 4.4: An unrooted (left) and rooted (right) topology for three extant taxa where one leaf is known to be the outgroup. The tree on the right has one extra edge and a sub-root node.

created a simple random inference package (RIP), which can be called in lieu of PHYLIP to recreate the tree topologies at each stage.[3] RIP takes a list of input extant taxa and an outgroup (basically the same input file used for PHYLIP) and randomly pairs active taxa to infer ancestors in the manner of neighbor joining, except no sequence information is used for the pairing decision/choice. The declared outgroup is reserved from pairing until the last step when it is connected to the root as NetReconstruct requires, so that it can be stripped from the subtrees in stages 1 and 2. Figure 4.5 shows this approach for a four extant (plus outgroup) example.

PHYLIP is a well-established and widely used software program providing a variety of routines related to phylogenies for those working with DNA data (e.g. sequences, proteins, and gene frequencies). NetReconstruct uses a threshold to determine which method of PHYLIP routines to employ. The default is nine extant taxa[4] and a subtree whose extant taxa count (not including the outgroup) falls above this threshold will be reconstructed using a neighbor joining approach, while instances less than or equal to the threshold, a maximum parsimony based approach is employed. As the three subtrees for a single reconstruction can vary in

[3]Unless specified to the contrary, NetReconstruct uses the tree reconstructions performed by PHYLIP, not RIP.

[4]PHYLIP documentation recommends not exceeding 10 or 11 for the DNAPENNY package, therefore we set the threshold at nine plus an outgroup.

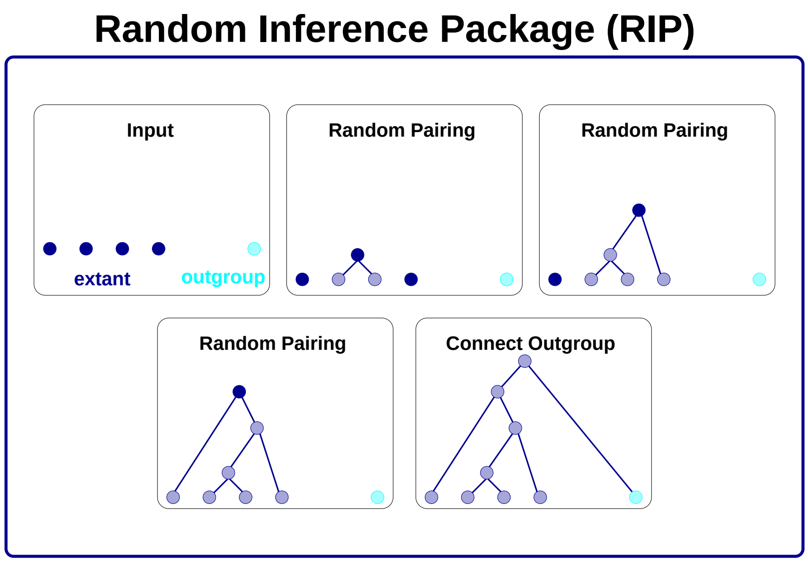


Figure 4.5: A four-taxon (plus outgroup) example of how the RIP software reconstructs a tree topology. The specific choices for pairings are dependent on random number calls, however the outgroup is always reserved and forced as part of the last pair. The dark solid colored nodes represent an active status for pairing purposes, while the lighter nodes are inactive.

number of extant taxa, it is possible that both PHYLIP approaches will be utilized during one run.

For the maximum parsimony scenario, when the taxa set is small, DNAPENNY[5] is used to find *all* the most parsimonious topologies (by using a branch and bound algorithm). As there are often multiple most parsimonious topologies, the CONSENSE[5] routine is subsequently used to consolidate the reconstruction results using what is known as the extended majority rule.[6] In the larger scenarios, DNADIST[5] is run

[5]DNAPENNY, CONSENSE, DNADIST, NEIGHBOR are all routines contained in PHYLIP [10].

[6]A consensus tree is a single phylogenetic tree output made from a set of input trees. There are different 'rules' (or types/methods) that can be employed. The extended majority rule approach starts by including topological features that appear in more than 50% of the input trees. Then, more structure is added (according to frequency of appearance) as long as it is compatible with the existing topology.

prior to NEIGHBOR[5] to create the distance matrix, which is needed as input for the neighbor joining algorithm. A PHYLIP input parameter for the DNADIST routine is a model of sequence evolution indicating assumptions of how the sequences were formed. As the SEQ-GEN default when simulating networks with NETGEN is the Jukes-Cantor model [78], the same default is included in NETRECONSTRUCT. If one of the other three sequence models (with default parameters) is desired, it can be requested or a custom parameters file for DNADIST can by provided as part of the input to NETRECONSTRUCT.

The core data structures of NETRECONSTRUCT are capable of representing networks with multiple hybrids and polytomies which may occur in the topology as PHYLIP is not limited to returning only binary trees. Like in NETGEN (see 2.1.4), the MERSENNE TWISTER [39] is used when random numbers are required for dividing sequences between the hybrid's parents, breaking ties, and providing PHYLIP random number seeds. The final topological structure reported by NETRECONSTRUCT is done in the modified Newick format that was presented in Section 2.4.

4.3 Experimental Results and Analysis

The evaluation of NETRECONSTRUCT's performance required an extensive series of tests. Experiments were conducted where the following parameters were varied: parental identification methods, PHYLIP thresholds, models of sequence evolution, and topological parameters (e.g. extant taxa size, event rates, ultrametricity, and second parent choice for the hybrid). Unless otherwise specified, the single-diploid-hybrid topologies were inferred with an outgroup, contained two homologues, each of length 1,000, were ultrametric, and had the second parent for the hybrid event chosen with the minimum Hamming distance option. The tripartition score was selected as the primary performance measure since it captures the similarity of two networks, in this case, the source topology created by NetGen and the one produced by NetReconstruct.

4.3.1 Methods for Identifying Extant Taxa of the Hybrid's Parents

Outlined in Section 4.1, there are four methods for determining which extant taxa can be traced back to the hybrid's parents. They are:

- extreme custom – extant taxa are assigned to each subtree and each parent of the hybrid,
- custom – extant taxa are assigned to each subtree, though separate assignments to each parent of the hybrid are not made,
- closest neighbor – if an extant taxon's closest neighbor is identified as being impacted by the hybrid, the taxon itself is assigned to the parental subtree, and
- closest $2x$ hybrid – twice the quantity of hybrid-impacted taxa are assigned to the parental subtree based upon average proximity to the hybrid node.

NetGen generated 2,000 source networks each with 50 extant taxa and birth/hybrid rates of 48 and 12 respectively. Then starting with the extant taxa, outgroup, and hybrid-impacted taxa, networks were reconstructed using the four techniques above. The results are shown in Figure 4.6. As one would expect the two custom options, with their ideal input information, are the best performers. However, it is encouraging to see that the closest neighbor option,which is the most likely to be used for real applications, also exhibits good results.

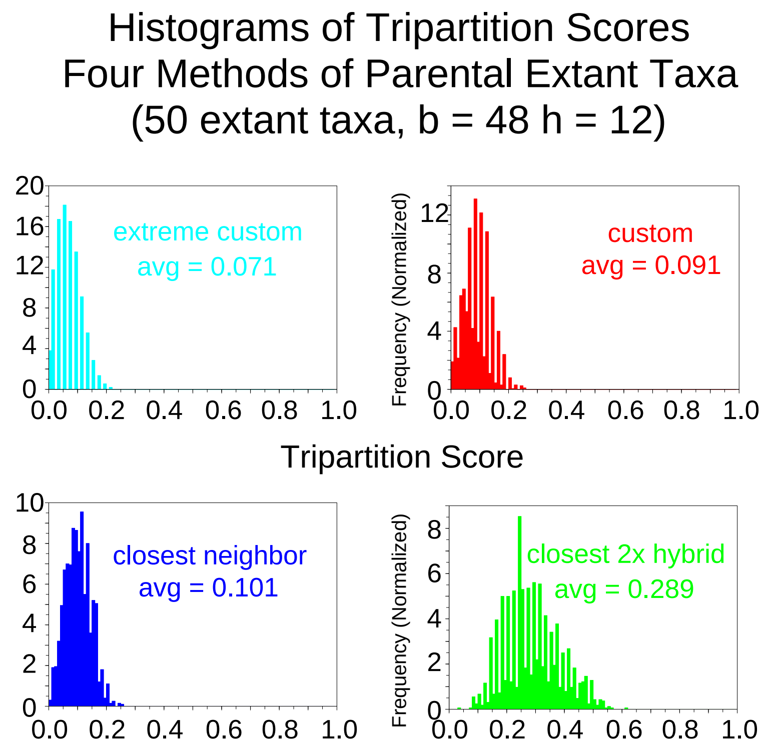


Figure 4.6: Tripartition scores for the four different methods of identifying parental extant taxa. The experiments were conducted with 2,000 source networks (though in the case of closest $2x$ hybrid only 1642 of those runs were eligible for reconstruction as the number of extant taxa impacted by the hybrid was so great that there were not enough remaining extant taxa for a parental set).

In order to confirm that the above results were not an anomaly based upon the general location of the hybrid, the experiment was repeated with a birth rate of 240 and hybrid rate of 12. This new birth:hybrid ratio of rates (20:1 here, versus 4:1 previously) means that usually the one hybrid will occur much later in the topology. The birth rate was scaled up to achieve the greater ratio, instead of decreasing the hybrid rate, due to concerns about extant taxa similarity. In general, small rates lead to large inter-event times causing long branch lengths and relatively high amounts of evolution and dissimilarity among extant taxa.

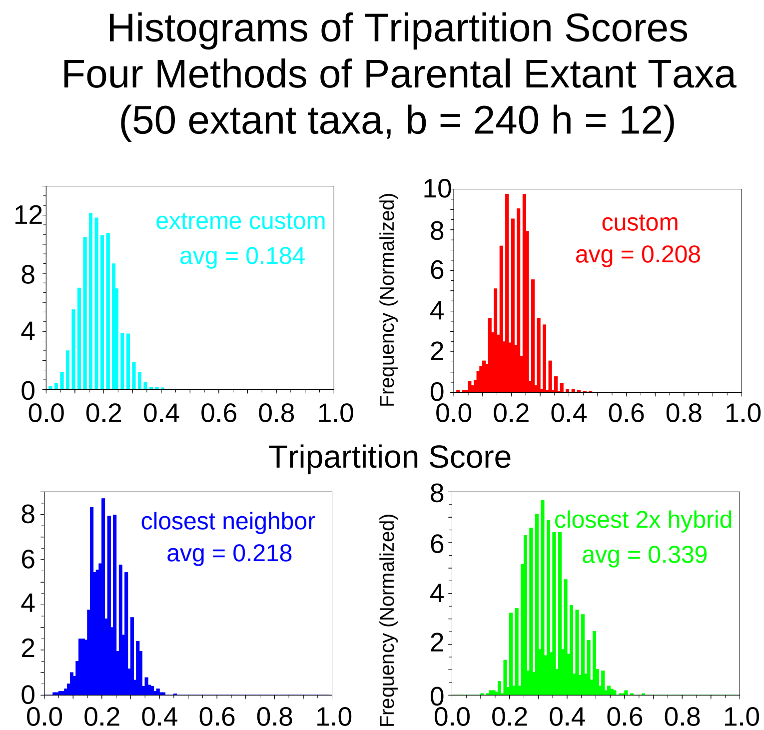


Figure 4.7: Tripartition results for the four different possible methods of identifying the extant taxa impacted by the hybrid's parents. Each scenario contained 2,000 runs, though most (approximately 1800), but not all of the sources met the one hybrid requirement. And due to set sizes, like in the previous experiment, approximately 1670 of the runs were capable of having the closest $2x$ work. Although shifted to the right, when compared to Figure 4.6 results (indicating slightly higher tripartition scores on average), the same trends appear to hold, with the customs being the best, closest neighbor being similar, and closest $2x$ hybrid having the largest scores on average.

We observe from Figure 4.7 that the results are similar, where the custom options performed best (avg $= 0.184 \pm 0.062$ and avg $= 0.208 \pm 0.065$ for the extreme and regular respectively), followed by closest neighbor being similar (avg $= 0.218 \pm 0.063$), and finally closest $2x$ hybrid (avg $= 0.339 \pm 0.087$). Although the behavioral trend is the same, the average tripartition scores are not as low as the first set of experiments. The influence that the rate variations have on reconstruction performance is further discussed in Subsection 4.3.4. Another intriguing feature of this data is that all scenarios, except for the extreme custom, appear to contain a “short” and “tall” histogram. As this behavior was consistently present in multiple experiments, it was investigated and is explained in the last subsection of this chapter (4.3.5).

From these experiments, we conclude that although the custom options are consistently better, the closest neighbor option is a good, realistic alternative when it comes to choosing an identification method for parental extant taxa. As the closest $2x$ hybrid method did not do well in either of the two cases, it may not be necessary to pursue extensively and/or analyze this option in the future.

4.3.2 Maximum Parsimony versus Neighbor Joining

The next step in the investigation was to determine what type of influence, if any, the PHYLIP threshold would have on the reconstruction performance. As explained in Section 4.2, this threshold determines whether a given subtree is reconstructed using maximum parsimony (PHYLIP’s DNAPENNY and CONSENSE) or neighbor joining (PHYLIP’s DNADIST and NEIGHBOR) routines. A threshold of nine extant taxa is NETRECONSTRUCT’s default as the PHYLIP documentation recommends restricting the use of the DNAPENNY routine to when there are ten or fewer taxa.[7] Distance methods such as neighbor joining have their drawbacks [15, 78], however they do provide reasonable results quickly under many circumstances. On the other hand, maximum parsimony is a computationally hard problem to solve optimally and without the use of heuristics, requires significantly more processing time.[8]

These experiments were conducted on ultrametric, single-hybrid topologies with 15 extant taxa, plus an outgroup. This number was kept small to ensure that the computationally intensive maximum parsimony processing would be completed in a

[7] The threshold is based on a count *without* the outgroup.
[8] A run-time analysis of the NETRECONSTRUCT algorithm is presented in Appendix A.

Tripartition Scores for Various Thresholds 1,000 runs, b=48 h=12, one hybrid, 15 extant taxa				
number eligible runs	parental taxa method	threshold=2	threshold=9	threshold=21
945	extreme custom	0.098 ± 0.085	0.108 ± 0.085	0.112 ± 0.086
945	custom	0.160 ± 0.102	0.172 ± 0.101	0.177 ± 0.103
945	closest neighbor	0.187 ± 0.100	0.198 ± 0.095	0.197 ± 0.100
814	closest $2x$ hybrid	0.346 ± 0.133	0.349 ± 0.131	0.351 ± 0.131
approximate wall run-time		1.25 hours	1.5 hours	17 hours

Table 4.1: These tripartition results indicate that NetReconstruct's ability to reproduce these types of topologies, small height, ultrametric, and small number of extant taxa, is not significantly affected by the choice of maximum parsimony or neighbor joining techniques.

reasonable amount of time. At one extreme, the threshold was set to two extant taxa, which forced subtree construction using the neighbor joining technique. The other extreme guaranteed maximum parsimony by setting the PHYLIP threshold to 21 extant taxa.

Table 4.1 shows the detailed results for 1,000 run scenarios. Although some of the averages appear slightly better with the larger thresholds, this is a false impression because when the standard deviation is considered, there is no significant improvement. All runs were performed on a dual processor linux machine (*Intel Core 2 CPU* at 2.4 GHz) and approximate wall clock times are given for each run. These results indicate that spending the additional time for a 21 threshold case, for a slight improvement in average performance, is not warranted, at least for a 15 extant taxa case with birth and hybrid rates of 48 and 12 respectively.

4.3.3 Sensitivity to Sequence Evolution Models – Jukes-Cantor, K2P, HKY/F84

Although the majority of our experiments were conducted using the Jukes-Cantor model of sequence evolution, other models (e.g. F84, Kimura two parameter (K2P), HKY, and GTR) exist [15, 78], and tools that either generate sequences or perform reconstructions typically offer options for some subset of them. The Jukes-Cantor model was chosen as the default for our tools since it is the simplest model, does not require additional parameters, and is an option available in both of our support tools (Seq-Gen and PHYLIP). We chose two models in addition to the standard Jukes-Cantor with which to compare for this experiment. The other two models were K2P (with an expected transition/transversion ratio of 2.0) and a HKY/F84 (with base frequencies of 0.1, 0.2, 0.3, and 0.4 for A, C, G, and T respectively, in addition to an expected transition/transversion ratio of 2).[9] This latter model scenario was selected as it was used by Posada and Crandall in [56] which is the basis for our case study in Chapter 5.

By specifying the model parameters for Seq-Gen as part of NetGen's input, 1,000 source topologies were made for each of the three scenarios. Other parameters such as rates and size were kept constant at 50 extant taxa, b=48 h=12, second parent of the hybrid chosen by minimum Hamming distance, closest neighbor used for selecting parental extant taxa, etc. Then for the reconstruction phase, the correct model of sequence evolution was provided to NetReconstruct in order to pass it to PHYLIP's dnadist routine. In order to ensure the model information would be used in the reconstruction, the threshold for determining a maximum parsimony or distance based reconstruction was lowered to two. Hence no reconstructions would

[9]The expected transition/transversion ratio refers to the ability for altering the frequency of transitions (base changes *within* the categories of purines (A and G) and pyrimidines (C and T) versus transversions, which are defined as changes which cross these categories.

be done with maximum parsimony. As the HKY model is available with Seq-Gen, but not PHYLIP's distance algorithms, the F84 (which is known to be similar to HKY [78]) was selected for the reconstruction portion of the third scenario. Shown in Figure 4.8, the average tripartition scores for the different scenarios are similar. This implies that for at least this set of parameters, the model of sequence evolution has minimal influence on NetReconstruct's performance.

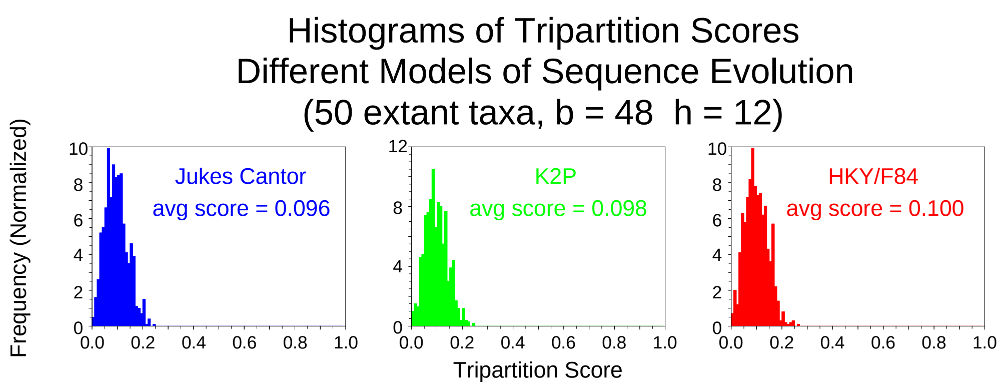


Figure 4.8: Tripartition scores for the different scenarios of sequence models. All three scenarios have very similar results indicating that under these parameters, the reconstruction algorithm is not sensitive to this influence.

4.3.4 Influence of Topological Factors on Reconstruction Performance

As the results in Section 4.3.1 clearly showed a difference in average tripartition score due to a rate change, this group of experiments was designed to investigate the influence of topological factors on the performance of NetReconstruct. Similar analyses have been conducted for trees (e.g. [44, 48, 68]). The factors chosen for examination here were: number of extant taxa, birth/hybrid event rates, ultrametricity/molecular clock deviation, and second parent choice for the hybrid event.

Extant Taxa Number

For the number of extant taxa study, we chose the three sizes of 15, 50, and 250. Although our immediate goal is at the 15-taxon range, it was important to see if larger scale attempts would result in poorer performance. The first round of 1,000 NetGen runs were conducted all with the same b=48, h=12 rates for all sizes and two homologues each with length 1,000. The topologies were ultrametric and limited to one hybrid and the PHYLIP threshold was set to nine. The results are shown in Figure 4.9.

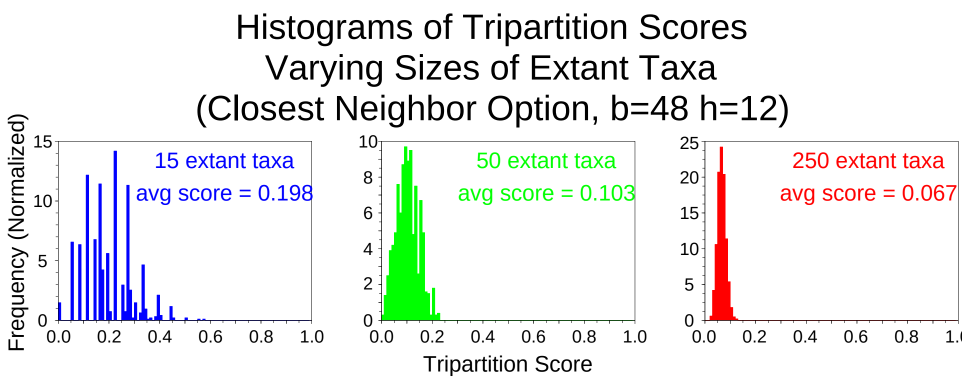


Figure 4.9: Tripartition scores for custom neighbor option with constant rates and varying extant taxa size. While the scores improve with taxa size, the results are potentially biased as the heights of the networks vary as a function of size.

It should be noted that the average heights (0.043 ± 0.014, 0.068 ± 0.014, and 0.101 ± 0.014) for these three cases differ because the rates were the same (generating on average the same size branches, thus taking more branches/level to achieve greater extant taxa counts and leading to greater values of clock heights for the networks). Therefore in order to make a fair assessment of how the amount of extant taxa influenced the tripartition results, it was necessary to adjust rates and rerun the experiments in order to achieve similar clock heights for the three cases of size. Using the previous 250 extant taxa case as a starting point, two new scenarios were 50 extant taxa (b=36, h=9, average height = 0.090 ± 0.019), and 15 extant taxa (b=24, h=6, average height = 0.086 ± 0.028). With these three heights being statistically close (within 1.5%) to each other, the variable height factor is removed and we conclude NetReconstruct's behavior, as measured by tripartition scores, does indeed improve with the number of extant taxa, as shown in Figure 4.10.

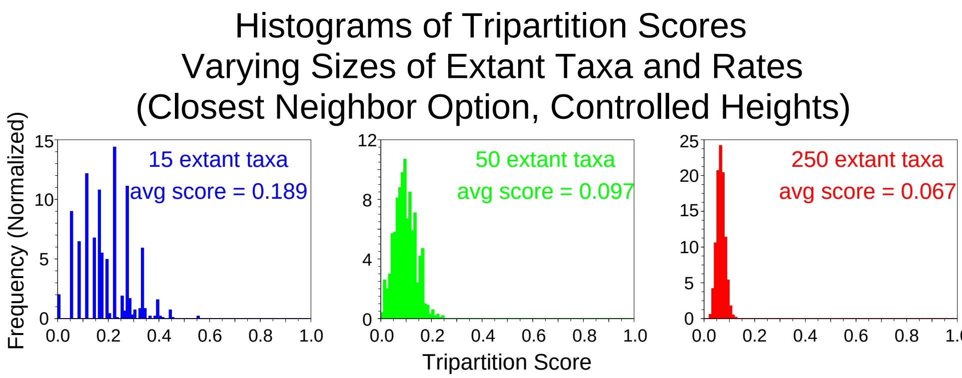


Figure 4.10: With rates adjusted to fairly compare differing sizes of extant taxa, there is evidence of a trend for better performance with greater sizes of extant taxa.

Event Rate Influence

Another factor which warranted examination was how the birth and hybrid rates influenced the tripartition results. We chose three different pairs of rates to examine, and following the approach of the previous experiment, the number of extant taxa was altered in this case to achieve a fair comparison of networks with similar heights.[10] While experimenting with PHYLIP the characteristic that was found to cause the most difficulty in reconstruction efforts was how similar/dissimilar the extant taxa were, which is determined not only by the rates, but also the number of extant taxa and sequence length. Thus an average percent Hamming distance [11] (including the

[10] As in previous sections, these experiments were conducted with ultrametric, one hybrid networks, 1,000 NetGen runs with two homologues, each of length 1,000.

[11] For each run, an average Hamming distance for all unique pairs of extant taxa is calculated, capturing how much of a spread exists among extant taxa. This representative value is calculated and tracked per run and averaged over all runs for the final statistic.

outgroup) was calculated and reported for these runs. Hamming distances in general are not an ideal measure, as a site can flip one or more times (possibly reversing earlier changes) during a simulation before arriving in an extant taxon sequence. However, it was tracked in these cases to see if there would be a significant difference in the values for different rate combinations.

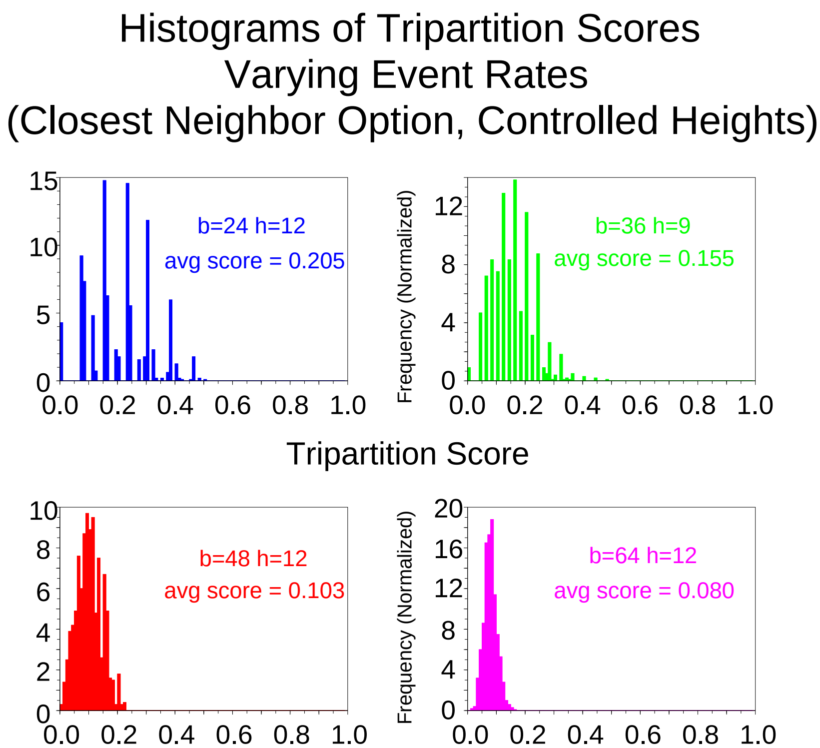


Figure 4.11: Tripartition scores for the closest neighbor option with varying rates. Clock heights for the network were controlled by altering the number of extant taxa. Results tend to improve as the event rates are higher.

As displayed in Figure 4.11, there is a clear trend that larger event rates have better performance with respect to tripartition scores. The full set of data for the closest neighbor option is presented in Table 4.2. It is interesting to observe that there seems to be a difference between the two cases which have the same (4:1) ratio of rates (b=48 h=12 and b=36 h=9). This makes intuitive sense because higher rates mean more frequent events and shorter branch lengths leading to easier,

Tripartition Score Dependencies for Event Rates Minimum 950 Runs, Closest Neighbor Option, Ultrametric Networks				
event rates	num extant	height	avg % hd extant	tripartition score
b=24 h=12	10	0.066	196 $\pm$ 64	0.205 $\pm$0.108
b=36 h=9	22	0.068	194 $\pm$ 45	0.155 $\pm$0.074
b=48 h=12	50	0.068	198 $\pm$ 34	0.103 $\pm$0.043
b=64 h=12	150	0.068	206 $\pm$ 26	0.080 $\pm$0.023

Table 4.2: Tripartition data for four different rate combinations. The fact that the second and third cases have different results for the same ratio of rates indicates that a higher rate value (leading to shorter branch lengths) in general aids the reconstruction effort.

more accurate subtree reconstructions by PHYLIP. There is no statistical difference across the cases for the average percent Hamming distance.

Molecular Clock Deviations

For this set of experiments we compared three different branch length deviation values for topologies of extant taxa size 250, all with birth/hybrid rates of 48 and 12 respectively. As in the previous sections, there were 1,000 source runs, with single-hybrid topologies, and two homologues, each of length 1,000. The first case was the standard ultrametric approach where evolutionary branch lengths are equal to the clock branch lengths leading to the all extant taxa being equidistant from the root with respect to evolutionary time. The goal of the other two scenarios was to deviate the evolutionary branch lengths resulting in non-ultrametric networks where the extant taxa have different *evolutionary* distances from the root. Deviations to the evolutionary branch lengths are achieved by multiplying each branch with a randomly selected value from a pre-defined gamma distribution. The distributions used for these experiments were gamma(3,2) and gamma(3,9), where the first term, η, of gamma(η,λ) is the shape parameter and λ dictates the scale. For a gamma distribution, the expected value is η/λ, with a variance of η/λ^2, leading to a standard

Tripartition Scores for Clock Variations 250 extant taxa, b=48 h=12, minimum 800 runs	
Ultrametric – Avg Evolutionary Height = 0.101 ± 0.014	
extreme custom	0.059 ± 0.015
custom	0.064 ± 0.016
closest neighbor	0.067 ± 0.016
closest $2x$ hybrid	0.225 ± 0.082
Gamma (3,2) – Avg Evolutionary Height = 0.278 ± 0.062	
extreme custom	0.073 ± 0.018
custom	0.078 ± 0.019
closest neighbor	0.081 ± 0.019
closest $2x$ hybrid	0.241 ± 0.071
Gamma (3,9) – Avg Evolutionary Height = 0.062 ± 0.014	
extreme custom	0.134 ± 0.024
custom	0.138 ± 0.024
closest neighbor	0.141 ± 0.024
closest $2x$ hybrid	0.293 ± 0.070

Table 4.3: Average tripartition scores under varying molecular clock assumptions for all four NetReconstruct methods of identifying parental extant taxa.

deviation of $\sqrt{\eta}/\lambda$ [18].

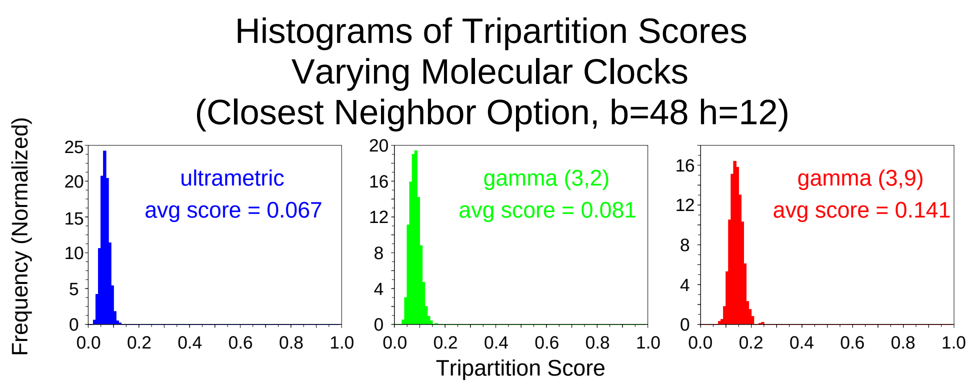


Figure 4.12: Tripartition scores for differing clock options. While the ultrametric scenario performs the best, the two deviation scenarios still demonstrate reasonable tripartition scores. The data results from 1,000 runs using the closest neighbor option for identification of parental extant taxa.

Figure 4.12 provides histograms for these three deviation scenarios under the closest neighbor method of parental extant taxa identification. These results indicate the deviation does have a noticeable impact. The results for all four parent options, including the average of the resulting evolutionary height, are presented in Table 4.3 and show similar trends to the closest neighbor histograms.

Second Parent Selection Methods for Hybrids

A final topological factor to consider was the method by which second parents for the hybrids in the source networks were chosen. For the previous experiments, the default criteria of minimum Hamming distance was used. Although simplistic, the algorithm is premised on the biological fact that parental lineages need to be similar. NetReconstruct's design implicitly favors a minimum Hamming distance approach. Recall that the parents of the hybrid are created by randomly splitting and assigning the DNA sequences of the hybrid to two parents and subsequent reconstruction of those two parental nodes is performed as part of the Stage 3 (remaining subtree) which employs either maximum parsimony or neighbor joining depending on the set size.

Four different settings for choosing the second parent were specified and the results for the closest neighbor reconstruction are shown in Figure 4.13. Although the standard deviations are not small enough to make absolute conclusions, it is interesting that the average scores follow an intuitive trend. Namely, the extreme best and worst scenarios are when the second parent for the hybrid is chosen by the minimum Hamming distance and random methods respectively. The two scenarios where the exponential function is used fall in between the extremes, with the smaller setting doing better on average. Recall from Section 2.1.3 that a truncated exponential function is defined by the user specifying a Hamming distance having a $1/e$ probability of occurring. A value for the desired Hamming distance between the first and second

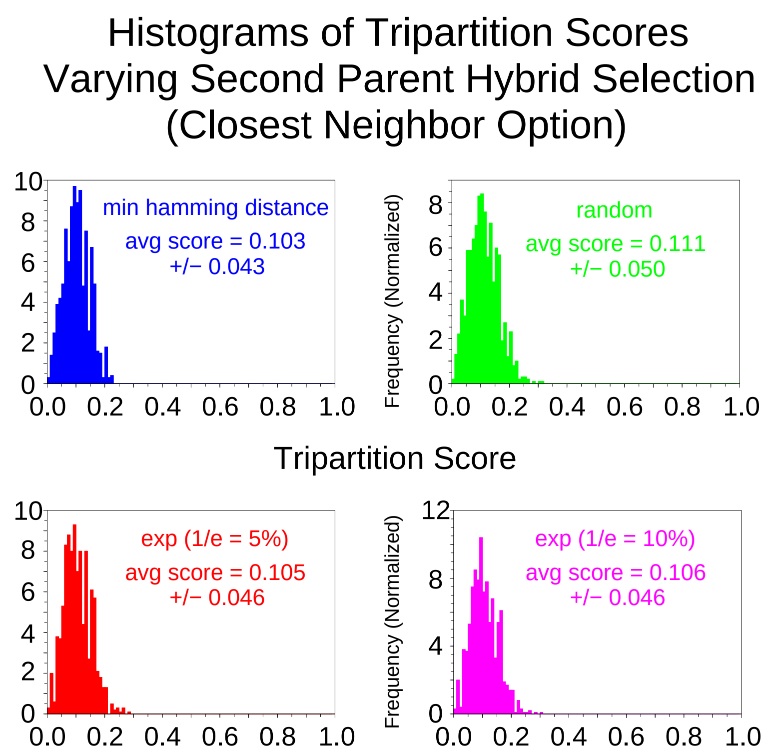


Figure 4.13: Histograms of the tripartition scores for varying scenarios of second parent selection. The two top plots show that the minimum Hamming distance and the random techniques yield the most diverse results while the bottom graphs fall in between. The exponential function (described in Section 2.1.3) appears to have a small, though not significant, impact on average.

parents is then chosen from the distribution. If no lineages meet this criteria, it is expanded as needed until the threshold is reached. In this case, the two scenarios set the $1/e$ values of 5% and 10%, and both utilized thresholds of 50%. These values were chosen to allow small variations in the second parent, but purposely not large as it is known that very diverse species do not hybridize. Under these circumstances, no significant influence on reconstruction quality was observed.

4.3.5 Data Characterization

While reflecting upon the results presented in the prior subsections, three questions have surfaced. First, how well could one expect NetReconstruct to perform with its dependency on PHYLIP, and in particular neighbor joining, for sufficiently large subtrees? Second, to what extent did the topological structure versus the sequence information contribute to the observed success of the reconstruction efforts? Finally, what was the cause of the appearance of two distributions in many of the tripartition histograms?

The first question is concerned with performance, however, it indirectly raises the issue of whether the subtree reconstruction method is a weakness of the overall model. For this PHYLIP dependency experiment, we created 1,000 source trees with 50 extant taxa using NetGen, having a birth rate of 48 and then used PHYLIP's neighbor joining algorithm to reconstruct trees. The similarity was then measured using the bipartition measure and provides a boundary notion for how well the subtrees for NetReconstruct can be made using these parameters.

Figure 4.14 shows that PHYLIP is capable of recreating very similar trees of this nature.[12] This outcome indicates that the earlier results for the idealized, extreme custom option are approaching the limit. As neighbor joining recreates reasonable trees given these parameters, this suggests that algorithmic improvements to NetReconstruct should be sought before changing the subtree reconstruction software support.

The next aspect to examine was the extent to which the topology assumptions ver-

[12] As NetGen yields rooted trees and PHYLIP returns unrooted trees, it was necessary to "unroot" the source trees as failing to do so would create an unfair comparison when computing the bipartition score since rooted trees contain an additional non-trivial edge. This is not a factor when the trees are used in NetReconstruct as an adjustment for the outgroup is made as discussed in Section 4.2.

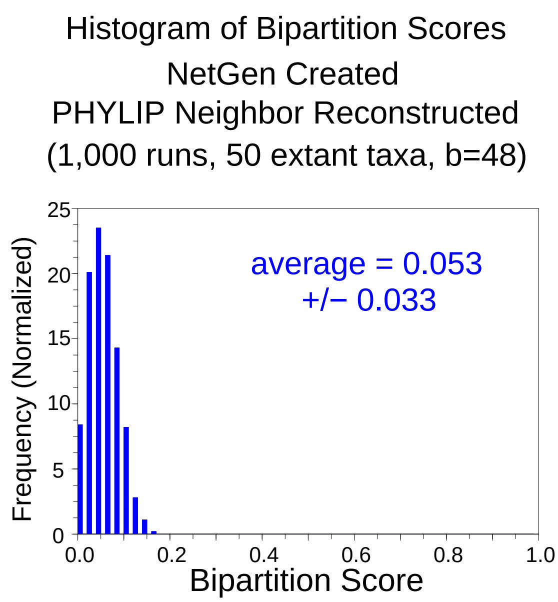


Figure 4.14: PHYLIP's neighbor joining reconstructions of *trees* perform very well for these sets of parameters, implying that one can expect good subtree reconstructions for the purposes of NetReconstruct.

sus the sequence information for NetReconstruct affected the tripartition results. The experiment for this area was to look at the results of calculating the tripartition measure on pairs of random networks (initially presented in Section 3.3.1) against results of NetReconstruct run with regular subtrees (using PHYLIP) and random subtrees using RIP. When using the combination of RIP and the extreme custom option, whose input specifies separate sets for the parental offspring, only topological aspects such as outgroups and the structure of a hybrid are preserved when executing NetReconstruct.

Figure 4.15 illustrates the intuitive result that the sequence information relied upon by PHYLIP for the subtree reconstructions has the most influence on creating results with low tripartition scores. However, it is interesting that the overall topological structure of the three subtrees provided by NetReconstruct does contribute favorably to the algorithm's performance as compared to the purely random

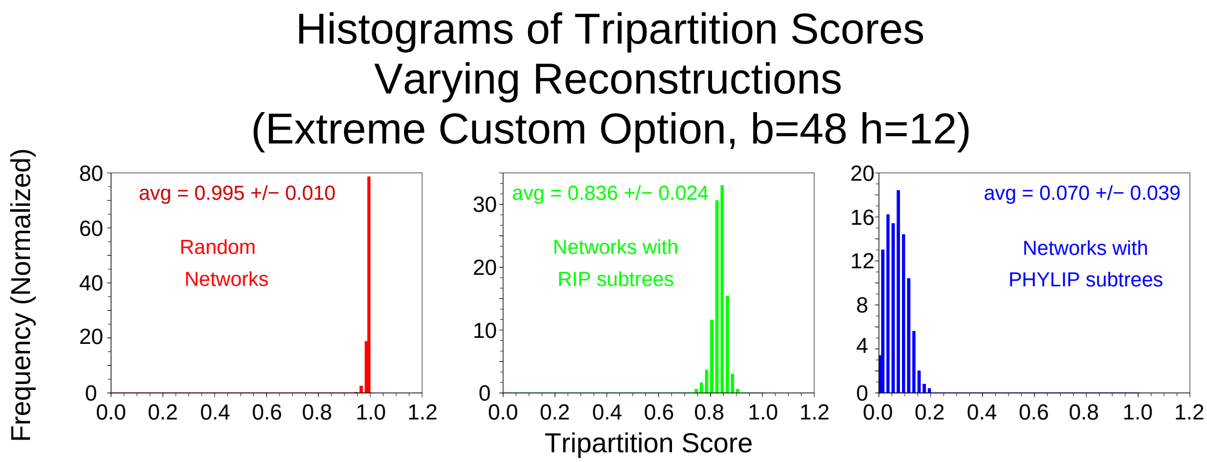


Figure 4.15: Tripartition histograms for networks reconstructed in three different manners, purely random, topological (no sequence) constraints, and full information. Clearly the last option performs the best, though topology does seem to reduce the scores slightly.

case.

The last issue was the appearance of more than one distribution in some of the tripartition results. For example, Figure 4.16 provides both the normalized and raw histograms from the 15 extant taxa case (b=48, h=12) where the closest neighbor option was used. After studying the reconstruction details for the closest neighbor case, it was determined that the closest neighbor option often underestimates the number of extant offspring descending from the hybrid's parents. On average, the number of parental extant offspring were 6.02 versus 0.29 for source and reconstructed networks respectively. This leads to these extant taxa being joined to the network as part of the remaining subtree, although often right next to the parents therefore preserving much of the original structure, and artificially increasing the number of non-trivial edges. This increase from 17 to 18.72 on average, impacts the tripartition calculation and thus the histogram.

The variation in the parental extant descendants from the closest neighbor re-

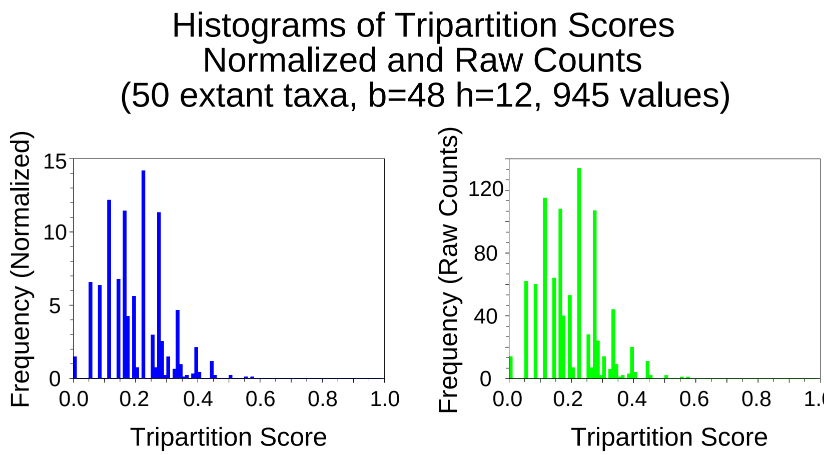


Figure 4.16: Tripartition histograms (normalized and not normalized) for the 15 extant taxa case. The data are comprised of at least two disjoint distributions.

construction technique is not present when using the custom option as the user is required to identify the parental extant taxa. Even with the number of parental extant taxa fixed, forcing the non-trivial edge counts to be equal, the custom option continued to exhibit the multi-distribution feature (see leftmost plot Figure 4.17). The one factor that was still allowed to differ in this scenario was whether NetReconstruct gave all the parental subtrees to one or both of the hybrid's parents. Hence we split the data based on this criteria and cleanly found this to be the underlying cause. The histogram on the right of Figure 4.17 shows the two sets – 674 cases where one parent has no descendants other than the hybrid (red) and 271 cases where both parents have extant offspring in addition to the hybrid node (blue). Furthermore, this explains why the multiple distribution effect was not visible with the extreme custom results.

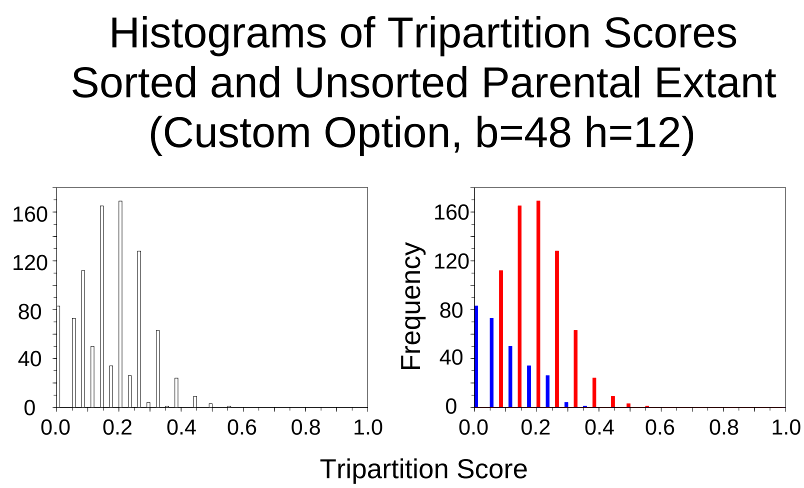


Figure 4.17: Tripartition histograms for the custom option. There are 945 data points in the leftmost plot and rightmost contains two sets (674 in the red and 271 in the blue) sorted according to whether one or both parents have extant taxa.

Chapter 5

Case Study – Reticulate Node Influence on Phylogenetic Reconstructions

Multiple studies have attempted to assess how the inclusion of extant taxa, descendant from an ancestral hybrid, affects the outcome of phylogenetic reconstruction [81]. In fact, some biologists concerned about possible loss of topological accuracy, will exclude any extant taxon believed to be the progeny of a hybrid [62]. Hence, understanding the influence of reticulate nodes on inferred phylogenies is an important pursuit by itself. However, when true topologies are known, an additional question can be asked – "What characteristics about the reticulate nodes themselves tend to hinder reconstruction efforts?".

Posada and Crandall conducted such an examination for small topologies that contained a single instance of chromosomal recombination [56], a different kind of reticulation event than hybridization. Although different in scope from our work, their study provided some ideas for how to explore a hybrid's influence on reconstruction at the interspecific level. Section 5.1 reviews pertinent aspects of the Posada and Crandall study that influenced our investigations. Our experimental design is outlined in Section 5.2, and the results in addition to conclusions complete the chapter in Section 5.3.

5.1 Relevant Components of the Posada and Crandall Study

Posada and Crandall considered topologies of size eight extant taxa with one reticulate node of chromosomal recombination. Although DNA is mingled across lineages by chromosomal recombination, this type of reticulation is unlike hybridization because individual chromosomes are recombined to become mixtures of different lineages. For simulation purposes, reciprocal chromosomal recombination can be modelled as two single strings with a defined breakpoint at which the remainder of the sequences are exchanged. For topological representation, such an event can be shown as an intersection of two lineages. In contrast, hybridizations, as we have defined them, receive chromosomes from their parental lineages, but the chromosomes are not recombined. Namely, each homologous chromosome consists of only the sequence inherited from one parent, not a mixture of the two parents. Diploid hybrids are modelled as the formation of a new lineage with complete strings. Thus, when depicted topologically, hybridization is illustrated with a new lineage in addition to the two existing parental ones. Nonetheless, both hybridizations and chromosomal recombinations are non-tree like events, which can be studied to assess their influence on phylogenetic analyses. Given our interest in diploid hybridizations, where each parent of the hybrid contributes half of its DNA sequences, the most applicable portion of Posada and Crandall's study is the reciprocal, 50/50 breakpoint scenario. [1]

The technique chosen by Posada and Crandall for modelling an evolutionary history containing a single, reciprocal recombination event was the composition of two separately generated trees. With this approach, the topology of the first tree provides the history of sequence evolution for the left portion of a single DNA string, up until the breakpoint index, where the history from the second tree is then used for the latter half of the sequence. Based upon an example from [56], Figure 5.1 shows a simple example of this concept for a four-taxon tree. Posada and Crandall also restricted their tree topologies to three shapes: 1) balanced (strictly binary tree structure), 2) unbalanced (all but the last birth event results in exactly one extant taxon and the subroot for the next event, sometimes referred to as a "caterpillar" tree), and 3) intermediate (a fixed topology containing balanced and unbalanced components).

Once source topologies using the HKY model were created[2] along with their corresponding sequences,[3] they were segregated into sets of 100 based upon whether the type of reticulate event was ancient, recent-divergent, or recent-close. Then, assuming no reticulate history, tree reconstructions were performed. Reconstruction techniques included: maximum parsimony, maximum likelihood, and minimum evolution. The resulting tree histories were assessed according to the percentage of subtrees found and the number of exact topology matches between the reconstructed topology and either of the source trees.

Although various tree reconstruction methods, mutation rates, and sequence lengths were tested, these particular parameters were not found to have a signifi-

[1] Reciprocal recombination is is also known as "crossing over" and is defined with two lineages interchanging DNA sequences at the specified breakpoint, which refers to the index of the base pair where the break occurs.

[2] The topology heights were chosen as 0.3 and 0.6.

[3] Source sequences were generated using the HKY model with a transversion ratio of 2.0 and A, C, G, and T base frequencies of 0.1, 0.2, 0.3, and 0.4 respectively.

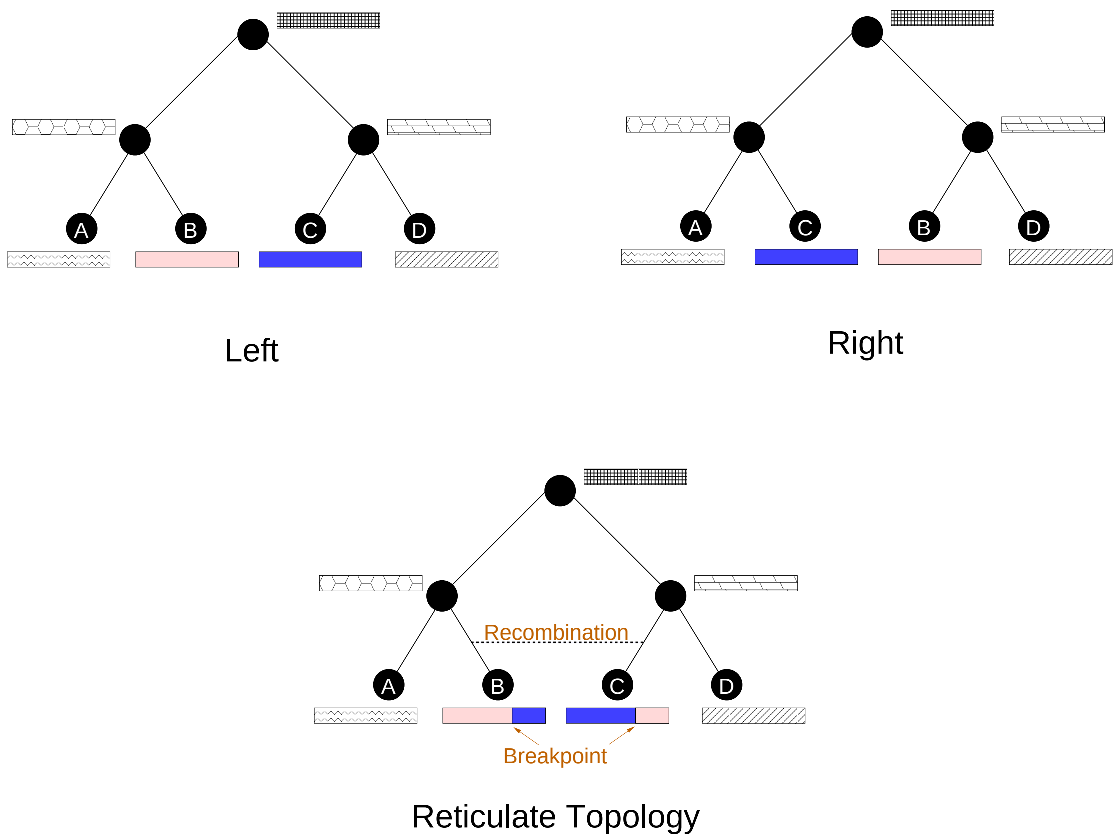


Figure 5.1: Posada and Crandall merged two underlying trees to mimic a reticulate topology resulting from a chromosomal recombination event. The two topologies on top are for the sequence evolution on either side of the breakpoint. The bottom topology is the resulting recombination network. (This figure is based upon an example found in [56].)

cant influence on their results. For the 50/50 reciprocal cases, Posada and Crandall showed that when the chromosomal recombination was assigned to have occurred recently between divergent lineages, the reconstructed tree was consistently different from the two underlying source trees. However, when the event was ancient or recent, but between close lineages, one of the underlying source trees was recovered almost all of the time [56]. There was a third category of results falling between the two extremes, for capturing when the reconstructed tree was close to one of the underlying trees, but not an exact match. However, none of the 50/50 reciprocal scenarios were described by this type of result.

5.2 Experimental Design for Single-Diploid-Hybrid Topologies

We adapted some experiments from Posada and Crandall to be applicable for our interspecific, diploid-hybrid focus, and designed others to capture network reconstruction performance data. As mentioned in the previous section, hybrid topologies fundamentally differ from the chromosomal recombination ones, as the sequences of the parental lineages remain unchanged and a new lineage that contains the intermingled DNA is formed. Even with these differences however, both types of topologies can be regarded as a single network or multiple, underlying trees. Although we used NetGen to create our source networks, and not underlying trees as Posada and Crandall, we chose to break them into their tree components in order to perform the match analysis like the chromosomal recombination study. An example of a single-hybrid topology decomposed into its two underlying trees is shown in Figure 5.2.

The first step in our experiments, was to use NetGen for creating large quantities of source networks containing single-diploid-hybridization events. For consistency with Posada and Crandall, we chose two sets of base rates (b=4.85, h=1 and b=2.43, h=0.5) that would on average yield heights of 0.3 and 0.6 for the eight extant taxa case. In addition to these small scenarios, we expanded the study to include topologies with an ingroup of 50 extant taxa.[4] The decomposition of source networks into their underlying subtrees, as described in the previous paragraph, was performed as a post-processing step to network generation.

[4]Note that the final sizes were actually one greater in each case as we used an outgroup for both generation and reconstruction.

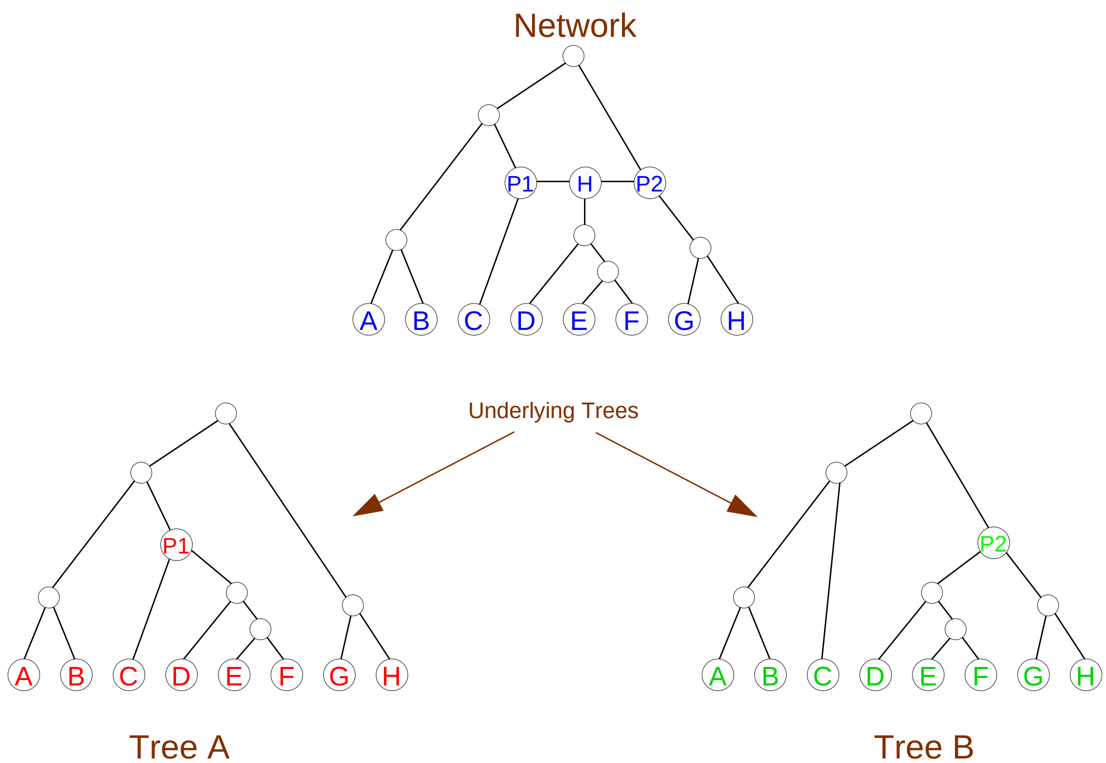


Figure 5.2: A network with a single hybrid event can be broken into two underlying trees. The top topology is the network with one hybridization. The two topologies on the bottom are the corresponding, underlying subtrees. The one on the left (red extant taxa), represents the evolutionary history for the chromosomes contributed by parent 1 (P1) to the descendants (D,E,F) while the topology on the right with green extant taxa shows the same information for parent 2 (P2).

With respect to sequence evolution, we created source topologies using the same model and parameters as Posada and Crandall (HKY, expected transition/transversion ratio = 2.0, base frequencies A=0.1, C=0.2, G=0.3, and T=0.4). As diploid hybrids use multiple chromosomes by definition, we worked with two, each of length 1,000. In order to generate networks for all of the reticulate node categories, we chose the second parents for the hybrids with the exponential function option where the distribution defining ($1/e$) values were set to 10% and 50% of the chromosome length of 1,000, allowing for diversity to range.

As eight extant taxa networks are small, Posada and Crandall segregated their source topologies into three categories of recombination: ancient, recent-close, and recent-divergent. Using our measures for reticulate timing and diversity we quantitatively established six categories, a subset of four which was used with the smaller, eight-taxon case. The categories were:

(i) ancient with low diversity,

(ii) ancient with high diversity,

(iii) mid-range with low diversity,

(iv) mid-range with high diversity,

(v) recent with low diversity, and

(vi) recent with high diversity.

The diversity component is divided into low, when scores are less than 0.5, and high, when scores are greater than or equal to 0.5. With eight extant taxa, timing was evenly split between ancient, scores less than 0.5, and recent indicating a value of greater than or equal to 0.5. For the 50 extant taxa, timing was segregated into three categories – ancient ($0 <= x <= 0.333$), mid-range ($0.333 < x <= 0.666$), and recent ($0.666 < x <= 1.0$). Given the simulation parameters, it was possible to generate source networks for all categories, though often disproportionately. Therefore, after source networks were segregated, a fixed number were randomly chosen for each category.

Both *tree* and *network* reconstructions were performed for each source network. NetReconstruct, with the extreme custom option, was used for the network reconstructions, and the threshold for the subtree algorithm choice between maximum parsimony versus neighbor joining was set to nine extant taxa. When neighbor joining was required by NetReconstruct, the sequence model parameters from above were provided as input to PHYLIP [10]. However, as HKY is not an option available with PHYLIP, the similar F84 model [78] was substituted in these situations.

In order to assess overall performance, tripartition scores were calculated for each pair of source and inferred networks across the reticulate node categories. It was also desirable to investigate the accuracy with which the hybrid nodes were located in the reconstructed networks, by comparing timing and diversity categories for the source and reconstructed networks. Unfortunately, a perfectly parallel comparison cannot be made as clock branch lengths are used by both the timing and diversity measures (see Chapter 3), as opposed to the evolutionary branch lengths produced by reconstruction algorithms. However, given the timing measure is defined in relative, not absolute terms, and that the networks are generated with the assumptions of constant rates and ultrametricity, this difference in type of branch lengths was not anticipated to have a significant influence on the results. By making both the overall and the more refined category assessments, insight into the behavior of the reconstruction algorithm was achieved.

For the straight tree portion of our study, trees were reconstructed using either the maximum likelihood technique, which was computationally reasonable for the eight-taxon scenarios, or neighbor joining at the fifty-taxon size. PHYLIP [10] was used for neighbor joining, again with the aforementioned sequence model parameters and RAxML's GTRGAMMA model [73] was used for the smaller scenarios. Bipartition scores were calculated and examined for exact and close matches between the two underlying source trees, T_A and T_B, and the reconstructed one, T_R.

At first glance, it may appear that one tree reconstruction, yielding a single tree, T_R, and two bipartition calculations for comparing T_R to each of the underlying source trees, T_A and T_B, would be sufficient to mimic the Posada and Crandall match analysis. However, this is incorrect as our networks produce a diploid hybrid by randomly selecting a homologous chromosome from each pair of homologues present in the two parents. This mimics what happens in the real-world when diploid hybrids are formed. However, as tree reconstructions use a single aligned sequence as input for each taxon, there are two possibilities for concatenating the two homologues from our source networks. Therefore it was necessary to perform a second tree reconstruction where the extant descendants of the hybrid had their chromosomes concatenated in the opposite order. This resulted in a total of four bipartition calculations to search for matches between the reconstructed trees, T_{R1} and T_{R2}, and the underlying trees, T_A and T_B.

Finally, an overall estimation of the performance for tree versus network reconstructions was conducted. A set of source topologies was generated and two sets of inferred networks were created – one using the closest neighbor option for selecting the parental descendants and the other using the extreme custom technique. By comparing these results to the bipartition scores of underlying source and reconstructed trees, conclusions were drawn as to whether network reconstructions have the potential to infer more accurate topologies than tree approaches.

5.3 Results and Conclusions for Single-Diploid-Hybrid Networks

Before addressing the results of the experiments, it is important to highlight what is being measured and the applicability of the results. Recall that each reconstruction was executed based on a complete set of extant taxa, including those lineages descended and not descended from the reticulate event. For the specified parameters, the results capture: 1) if, and how much, a reconstructed topology is altered by a reticulate node, and 2) what categories of reticulate nodes are more or less problematic for the chosen reconstruction techniques. The data reflect the performance of the selected tree and network reconstruction algorithms, from which one can assess the capabilities of the algorithms. The data do not reflect the difficulty, ease, or accuracy with which other algorithms and/or laboratory techniques may perform when attempting to detect or reconstruct the same reticulate sources.[5]

The data in Table 5.1 indicate that the tripartition scores improve with larger numbers of extant taxa and higher rates. Given the results of Chapter 4, this performance is consistent with NetReconstruct's general behavior. Upon closer observation, the more intriguing result is that within each set of experiments, the tripartition score follows a clear trend according to the category defined by the timing and diversity of the hybrid. Namely, the reconstructed networks with lower tripartition scores are most probable when the hybrid is ancient with low diversity. Poorer reconstructions are found when the topology has a recent and high diversity hybrid.

One finds a similar trend when examining the timing and diversity scores presented in Table 5.2. Using the same set of reconstructed and source networks from above, the timing and diversity scores for the reconstructed hybrids were calculated in order to determine if the reconstruction placed them in the same range as their sources. The percentages in Table 5.2 indicate how many reconstructed hybrids in the set possessed comparable timing and diversity scores relative to the category. The data here show that in general, NetReconstruct tends to infer hybrids as being ancient and occurring between parental lineages with low diversity. The timing results are the most significant of the two, with drastic differences between the two ends of the spectrum (ancient versus recent). It is possible that using clock versus evolutionary branch lengths, as discussed in the previous section, influenced these results, but nevertheless these values motivate future investigation. The fact that the

[5]For example, a traditional tree analysis may be performed if a biologist has pairs of homologous chromosomes that are known to come from separate parents and are not suspected of having undergone recombination. In this scenario, each member of the paired homologues is treated as a separate extant taxon and a tree reconstruction is performed to determine if the different members of a pair associate with a different subtree.

Average Tripartition Scores by Category (Reconstructions Performed with Extreme Custom Option)				
birth rate = 4.85, hybrid rate = 1.0				
	8 extant taxa		50 extant taxa	
Category timing/diversity	Tripartition Score	Set Size	Tripartition Score	Set Size
ancient low	0.048 ±0.066	500	0.063 ±0.033	500
ancient high	0.132 ±0.092	125	0.082 ±0.035	500
middle low	—	—	0.090 ±0.038	500
middle high	—	—	0.115 ±0.041	500
recent low	0.162 ±0.105	500	0.122 ±0.046	500
recent high	0.231 ±0.122	500	0.159 ±0.049	250
birth rate = 2.43, hybrid rate = 0.5				
	8 extant taxa		50 extant taxa	
Category timing/diversity	Tripartition Score	Set Size	Tripartition Score	Set Size
ancient low	0.056 ±0.072	500	0.086 ±0.051	500
ancient high	0.149 ±0.091	125	0.099 ±0.045	500
middle low	—	—	0.129 ±0.084	500
middle high	—	—	0.140 ±0.070	500
recent low	0.168 ±0.108	500	0.182 ±0.128	500
recent high	0.237 ±0.122	500	0.212 ±0.111	250

Table 5.1: Average tripartition scores for the four scenarios sorted by hybrid category. Reconstruction scores improve when the hybrid occurs early during the simulation and its parents are from closely related lineages.

diversity scores exhibit a similar, but slightly different behavior in a less significant fashion, leads to the suspicion that the sequence evolution model may play a role in these results.

Depending on the sequence data and analysis techniques employed, one might expect that a hybridization between diverse parents in recent history would be the easiest kind to detect because its differences should be significant, and there would not have been sufficient time for it to blend into the evolutionary history. However, a very different trend is found in Tables 5.1 and 5.2, with the topologies containing

Reconstructed Hybrids with the Same Range of Timing and Diversity Scores as their Source Category (Reconstructions Performed with Extreme Custom Option)				
birth rate = 4.85, hybrid rate = 1.0				
	8 extant taxa		50 extant taxa	
Category timing/diversity	% Timing Correct	% Diversity Correct	% Timing Correct	% Diversity Correct
ancient low	74.6	98.8	78.6	90.4
ancient high	95.2	76.0	98.0	52.0
middle low	—	—	6.0	66.2
middle high	—	—	2.2	49.6
recent low	49.6	79.4	0.0	51.6
recent high	8.4	75.4	0.0	42.4
birth rate = 2.43, hybrid rate = 0.5				
	8 extant taxa		50 extant taxa	
Category timing/diversity	% Timing Correct	% Diversity Correct	% Timing Correct	% Diversity Correct
ancient low	72.2	99.0	75.5	91.6
ancient high	92.0	76.0	87.8	49.6
middle low	—	—	18.2	65.2
middle high	—	—	16.8	52.2
recent low	48.8	78.2	4.4	48.4
recent high	8.0	74.8	4.4	48.4

Table 5.2: The data indicates that NetReconstruct has a tendency to place reconstructed hybrids as being ancient and occurring between low diversity parents. It is interesting to note that the rate of decline across the categories between timing and diversity is significantly different.

an ancient hybridization scoring better than the recent ones. This trend is a reflection of how the reconstruction algorithm performs on simulated data. A simple, initial explanation for this tendency is that perhaps NetReconstruct performs better whenever sequences are more close than diverse. A more intriguing question is whether NetGen contains biases that cause the recent and diverse hybrids to be too diverse leading to reconstruction difficulties. This issue is addressed further in relationship to the results of the next experiment.

In addition to examining the tripartition scores, we also conducted a match analysis like the one performed by Posada and Crandall. For this measure, the bipartition score between the inferred trees, T_{R1} and T_{R2}, and each of the underlying source trees, T_A and T_B is calculated. Recall that a bipartition result of 0.0 for a pair of trees indicates the topologies are isomorphic. Table 5.3 shows the bipartition information for the scenarios examined. The percentages are reported for comparison purposes because it was not computationally reasonable to have the same number of source networks in each category.

Case Study Reconstructed vs. Underlying Tree Matches by Category						
birth rate = 4.85, hybrid rate = 1.0						
	8 extant taxa			50 extant taxa		
Category timing/diversity	Match Count	Set Size	%	Match Count	Set Size	%
ancient low	429	500	85.8%	26	500	5.2%
ancient high	86	125	68.8%	20	500	4%
middle low	—	—	—	19	500	3.8%
middle high	—	—	—	15	500	3%
recent low	311	500	62.2%	15	500	3%
recent high	272	500	54.4%	15	250	3.6%
birth rate = 2.43, hybrid rate = 0.5						
Category timing/diversity	Match Count	Set Size	%	Match Count	Set Size	%
ancient low	392	500	78.4%	11	500	2.2%
ancient high	80	125	64%	15	500	3%
middle low	—	—	—	5	500	1%
middle high	—	—	—	9	500	1.8%
recent low	324	500	64.8%	5	500	1%
recent high	234	500	46.8%	1	250	0.4%

Table 5.3: Percentage of matches between the reconstructed trees and the underlying trees of the source network. Although topologies with eight extant taxa are often matched, this behavior occurs significantly less often for the larger scenarios.

The higher frequency of perfect matches with eight extant taxa makes intuitive sense as there are fewer internal edges that must be correctly reconstructed. This is supported by examining a simple case analysis for birth only trees. One thousand trees of two different sizes (rates are kept constant) are created by NetGen and then inferred using RAxML. The histograms in Figure 5.3 indicate that although the average bipartition score is similar for both scenarios, the one on the left with fewer extant taxa has three distinct bins – each one corresponding to whether zero, one, or two edges are incorrect. Furthermore, it is interesting to note the eight taxon case on the left implies an upper bound of 79.3% for how often, on average, one can expect to have a perfect tree reconstruction.

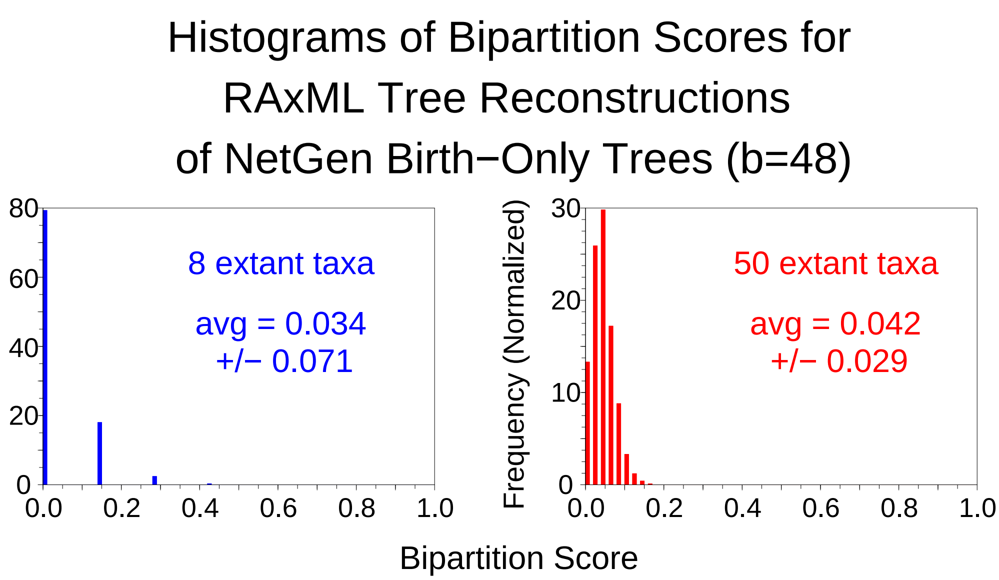


Figure 5.3: Histograms reflecting the spread of bipartition scores for two differing sizes of simulated/reconstructed birth-only trees. Specifically, the scores for the smaller topology are more discretized.

Although it is encouraging that our study yields high percentages of matching for the eight-taxon case, it remains unclear for certain why Posada and Crandall found almost 100% matches for their ancient, reciprocal 50/50 breakpoint scenario. We hypothesize the cause is related to their limited tree shapes, as that is the primary difference found between the experimental setups. Furthermore, our values for matching in the recent diverse cases are much higher than their results which is puzzling. Even with potential confounding factors between the chromosomal recombination and single-diploid-hybrid studies, we see the re-emergence of the earlier trend where ancient hybrids are more easily reconstructed than recent and diverse ones. The appearance of this trend in both the tree and network studies motivates considering whether the behavior is real or an artifact of the algorithms employed by both Posada and Crandall and us for generating sequences.

This sequence evolution issue may be more pronounced with our diploid hybrid model than others, as there is an element of randomness when selecting and assigning the homologues from the parental lineages to the hybrid. Recall that the multiple possible outcomes for the hybrid sequences required the reconstruction of a second tree for the underlying tree experiments. NetGen models homologues as individual sequences that evolve over time. Although it is realistic that a pair of homologous chromosomes do not match identically, it may be desirable to maintain some level of genetic similarity between them throughout the simulation. However, there is no current capability to ensure this, and for topologies with greater evolution, the similarity of homologues will diminish over time. Thus for hybrids occurring later in the simulation, it is quite likely the combined homologues donated by each of the parents will exhibit even less genetic similarity. However, Hamming distance, or minimum evolutionary distance, is typically a factor for selecting the second parent of the hybrid. Therefore, although the resulting hybrid lineage may contain dissimilar paired homologues, they cannot be too disparate, otherwise the second parent would not have been a candidate to participate in the hybrid event. This issue of extended sequence divergence certainly warrants future work. If it is found to be a real limitation, it would be interesting to see if other simulators exhibit a similar weakness. As the Posada and Crandall results demonstrated the same trend across the reticulate node categories for reconstruction performance, it is possible the difficulty is a general limitation of evolutionary sequence models.

The final experiment investigated how network and tree reconstructions compare to each other. Ideally one would like to quantify the topological similarity/dissimilarity between a reconstructed tree and a source network. This would facilitate a comparison of two scores – one for a reconstructed tree/source network and another for a reconstructed network/source network. It would then be possible to gauge which reconstruction algorithm is better suited for inferring the evolutionary history.

Unfortunately, the ways in which the bipartition and tripartition measures are defined, attempting to score a tree and a network topology results in incomparable edge lists. Additionally, even with the same number of extant taxa, networks inherently have more internal edges due to the reticulate event(s). This fact makes it more difficult to achieve a perfect match between two network topologies than two trees, as there are more internal edges under consideration. However, having a single edge incorrect between two topologies has a smaller cost for a network than a tree due to the extra internal edges present. Furthermore, comparing bipartition and tripartition scores does not capture the benefit of the hybrid information (e.g. location of hybrid in the history, relationship of hybrid's parents to each other, etc.) produced by a network reconstruction algorithm. Given these shortcomings and restrictions, the best available option at this time is to compare network and tree reconstructions by examining their respective scores (tripartition and bipartition) for the appropriate corresponding source. Namely, we compare tripartition scores of reconstructed and source networks to bipartition scores of reconstructed and underlying source trees.

These experiments examined the same four size/rate sets as previously discussed. In order to achieve an overall performance assessment, each scenario contained 1,000 single-diploid-hybrid networks from all categories (in proportion to their frequency of occurrence during the generation process). To review, the four sets of simulation parameters were:

- 8 extant taxa, birth = 4.85, hybrid = 1.0,
- 8 extant taxa, birth = 2.43, hybrid = 0.5,
- 50 extant taxa, birth = 4.85, hybrid = 1.0, and
- 50 extant taxa, birth = 2.43, hybrid = 0.5.

The three reconstruction scenarios examined were:

1. Tripartition score of reconstructed network, N_{R-CN} (NetReconstruct with closest neighbor option), and the source network, N_S. This calculation is:

   $$tripart(N_{R-CN}, N_S).$$

   This corresponds to a possible real-world situation where a single hybrid and its extant descendants are identified and a network reconstruction is performed.

2. Bipartition score of a reconstructed tree and an underlying tree of the source network.[6] In this situation we perform both tree reconstructions (T_{R1} and T_{R2}) possible (where the sequences have been flipped for hybrid impacted taxa) and compute all four bipartition scores with the two underlying source trees (T_A and T_B). The average of T_{R1}'s and T_{R2}'s best bipartition score with respect to T_A and T_B is then calculated, namely:

   $$average(X, Y),$$

   where $X = min(bipart(T_{R1}, T_A), bipart(T_{R1}, T_B))$, and

   where $Y = min(bipart(T_{R2}, T_A), bipart(T_{R2}, T_B)))$.

[6] For the smaller size of eight extant taxa the tree reconstructions are performed with the maximum likelihood technique implemented in RAxML. Topologies with 50 extant taxa are inferred with PHYLIP's neighbor joining application.

This is representative of a situation where the extant taxa are known, but a hybrid is not suspected to have occurred and therefore a tree reconstruction with a single sequence is performed. However, not knowing the order of the sequences or which underlying tree is more likely, the above calculated value is used.

3. Tripartition score of reconstructed network, N_{R-EC} (NetReconstruct with extreme custom option), and the source network, N_S. As expressed by:

$$tripart(N_{R-EC}, N_S).$$

This is an ideal situation where all the extant details are known and it provides an upper bound on NetReconstruct's potential performance.

The first set of histograms, Figure 5.4, indicates that for birth and hybrid rates of 4.85 and 1.0 respectively, the tree reconstructions are topologically more close to their underlying source trees than are the reconstructed networks to their source networks. However, the difference between the extreme custom and tree reconstructions is small for both extant sizes and indicates that network reconstructions have the potential of outperforming tree efforts.

For the second set of histograms (b = 2.43 h = 0.5) in Figure 5.5, the tripartition scores for the extreme custom option are lower on average than the bipartition values in both size scenarios. The fact that the smaller rates, which on average result in longer branches and more evolutionary change, performed better than the first set of rates is not consistent with the findings of Chapter 4 where tripartition scores improved with larger rates. We hypothesize that the differing proportions of the hybrid categories represented in each scenario may be influencing these results and further experiments would be required before drawing any global conclusions. The results do indicate however, that in addition to hybrid information provided by network reconstructions, the potential to acquire a more accurate topology with a network instead of tree reconstruction does exist.

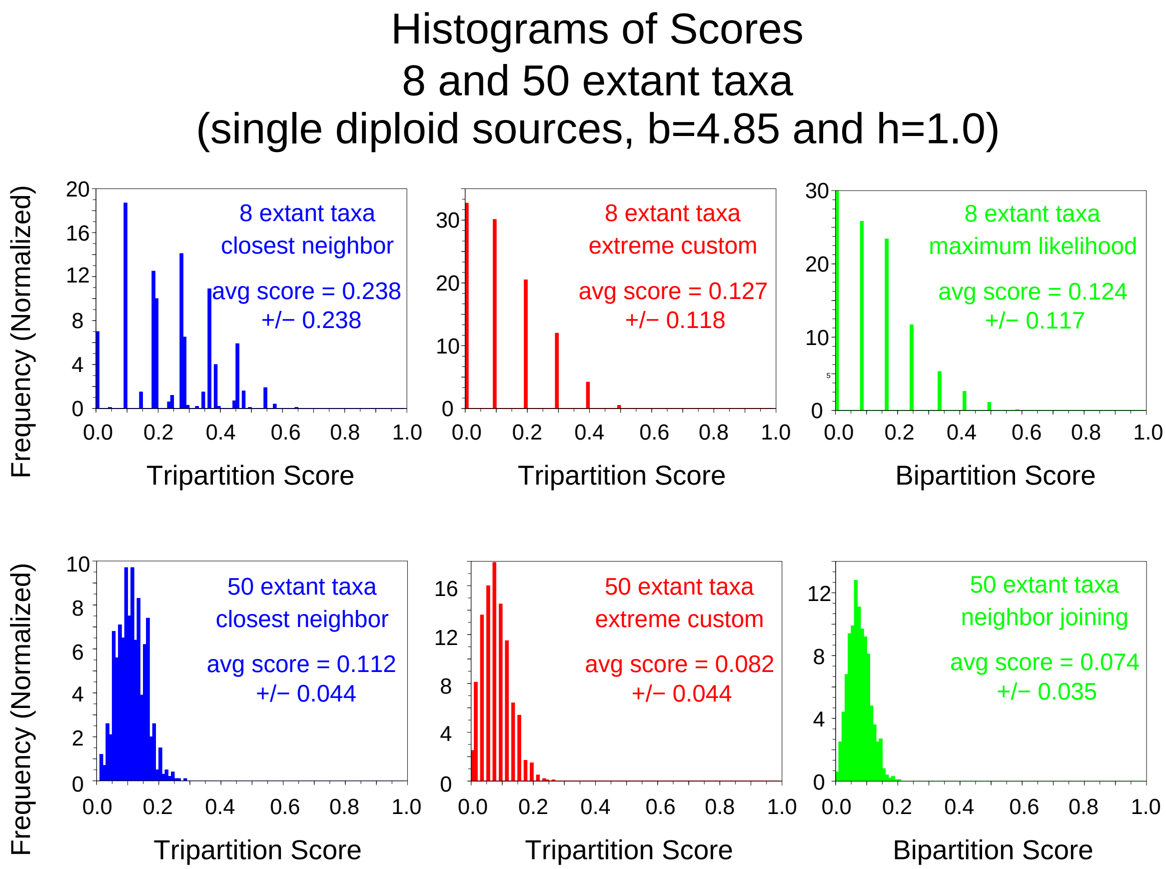


Figure 5.4: Tripartition and bipartition scores for topologies based upon single-diploid-hybrid source networks. The bipartition scores of the reconstructed trees and underlying source trees perform the best on average for this set of rates.

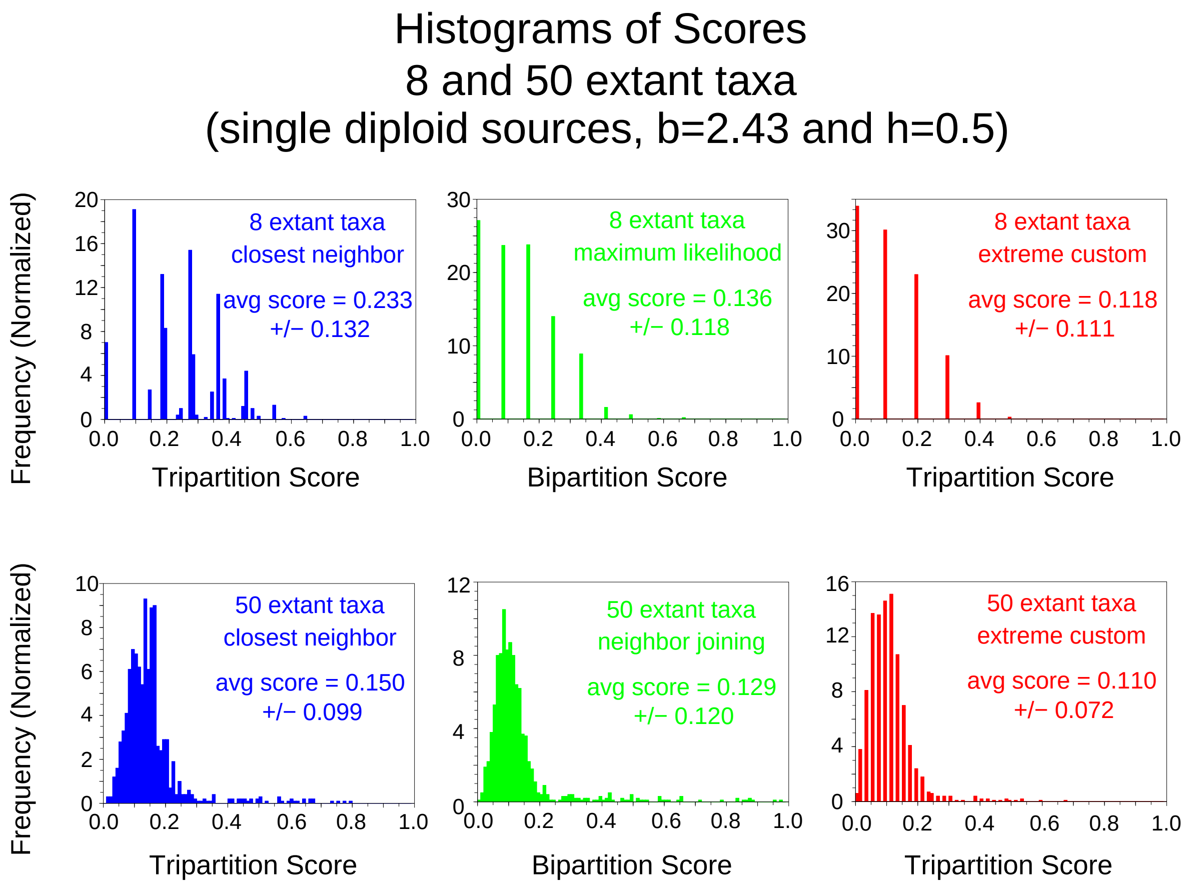


Figure 5.5: Tripartition and bipartition scores for topologies based upon single-diploid-hybrid source networks. On average, the tripartition scores resulting from the extreme custom reconstructions and source networks perform the best out of the three scenarios examined for this set of rates. This indicates a network reconstruction algorithm is capable of providing a more topologically accurate reconstruction than a tree technique under certain conditions.

Chapter 6

Conclusions

This work contributes to both the fields of computer science and bioinformatics. A direct benefit for bioinformatics is the three new pieces of publicly available software, NetGen, NetMeasure, and NetReconstruct. Although not designed to address every question pertaining to reticulate events, these programs do provide a framework for exploring modelling issues and promoting the development of new techniques relevant to phylogenetic networks. The new node-based measures of timing, diversity, and impact, should aid communication between biologists and computer scientists when discussing the characterization of reticulate nodes in graphs. Additionally, NetGen's underlying models of hybridization and simultaneous sequence evolution represent significant advances for the simulation community. Moreover the techniques related to subtree reconstruction, which were developed for NetReconstruct, yield new applied algorithms. Finally, the modified Newick format answers an immediate need of biologists and computer scientists by representing reticulate phylogenies.

The dissertation proposal [45] for this work put forth the following research goals:

(i) provide reasonable simulation data for network reconstruction benchmarks,

(ii) assuming there is a significant meaning to the region and/or type of hybrid, develop a new measure to capture this information,

(iii) gain insight into the impact of hybrid data on phylogenetic trees, and

(iv) assess the impact and propose a reconstruction method that will place a single hybrid event in an appropriate region of the reconstructed network.

At the time of the proposal, it was believed that an existing network simulator from our extended laboratory group could be modified to incorporate simultaneous sequences and would provide reasonable source data for our research. Upon closer inspection, it was found to produce an expected number of lineages proportional to $ln(2)$. Although a valid model, it was not appropriate for our interspecific efforts, where the exponential growth of the birth-death model (reviewed in Chapter 1), is more biologically realistic. Therefore, as presented in Chapter 2, a new model was created and NetGen developed. Experiments validating the sequence design of the software, as well as the population growth behavior and branch length distributions, were conducted. This provided confidence in the code's primary functionality. With an extensive list of input parameters, this simulator is capable of providing a wide array of source networks containing diploid and/or polyploid hybrids.

An unanticipated contribution from the initial phase of research was the development of the modified Newick format which emerged from a software input/output requirement [46]. Using the well-established Newick format as a foundation, supplemental notation was formulated to annotate hybrid nodes and their descendants in an efficient manner. The parsing and internal structure for topologies in NetMeasure and NetReconstruct were facilitated by having a common representation combined with a graph traversal algorithm. Since a standard format for representing networks is lacking in the phylogenetic community, and as this format reverts to the generally accepted Newick in the case of trees, it is hoped the modified Newick will lend itself to being incorporated with existing analysis and visualization tools that are already established for phylogenetic trees.

In our experiments, we found the pre-existing tripartition measure [43] was a good metric for capturing the topological similarity/dissimilarity of two networks. When reviewing background literature, the measures that could be used to characterize reticulate nodes in a quantitative manner were found to be lacking. Guided by our reconstruction algorithm development, the new measures, reticulate timing, impact, and diversity were developed to meet this need and provided structure to our case study.

NetReconstruct is a new reconstruction algorithm that employs a subtree approach. Various experiments were conducted to assess the model's performance under different simulated scenarios. The novel component of the algorithm, and the primary contributor to its favorable performance, is the input data identifying the extant offspring of the hybrid. For simulated source topologies, this information is known. However with real DNA sequence information, a hybrid-impacted set may not be easily identified at this time. It is believed that as DNA sequence analysis techniques for detecting hybrid origins are advanced, it will be possible to derive these sets and during the interim, the alternative ideas from Appendix B for conjecturing such sets can be explored and employed.

The case study brought all of our applications NetGen, NetMeasure, and NetReconstruct, together for the purpose of investigating the influence of a single-diploid hybrid on reconstruction efforts. We found, as did Posada and Crandall [56] in their chromosomal recombination study, that reconstruction algorithms perform better with networks containing an ancient reticulate event formed by non-diverse parents than those with a recent hybridization of divergent lineages. In addition, we found evidence that network reconstruction algorithms have the potential to infer more accurate topologies than tree techniques, as well as providing hybrid information. A by-product of our case study was that our newly developed timing and diversity measures provided a solid quantitative structure for categorizing networks and investigating their behaviors. Furthermore, through these experiments we have observed how various models (e.g. hybrid creation, sequence evolution, and subtree structure) can interact to possibly produce artificial behaviors.

In summary, this work provides a practical framework for researching phylogenetic network algorithms. It is hoped, that with a realistic simulator such as NetGen and characterization techniques as contained in NetMeasure, NetReconstruct will be the first of many reconstruction algorithms developed for phylogenetic networks with hybrids. From this effort, there are many directions of research that can be pursued in this burgeoning area. We anticipate this work will prove to be one small, yet serviceable step, along those paths.

Appendix A

Run-Time Analyses

When developing algorithms, it is beneficial to assess their theoretical computational time. Such analyses can provide insight into the complexity of the algorithm and how its run time will increase as the input size grows. In Section A.1, we first calculate the average number of events that will be executed in a simulation. Using this value, an upper bound on conducting a depth-first-search of a simulation's topology is then calculated. With these two components complete, complexities can be assessed for the new simulation (Section A.2), node measures (Section A.3), and reconstruction algorithms (Section A.4). Finally, Section A.5 provides some experimental run-time results for NetGen, as it is the most complex of these applications.

A.1 Topological Preliminaries

Our first step is to estimate the average number of events that will occur in a simulation given the number of extant taxa specified, in addition to the birth, hybrid, and death rates. We define input and output variables as:

$$
\begin{array}{lll}
Inputs: & & \\
 & B & \text{birth rate} \\
 & H & \text{hybrid rate} \\
 & D & \text{death rate} \\
 & n & \text{number of extant taxa} \\
Outputs: & & \\
 & ne & \text{average total number of events in a simulation.}
\end{array}
$$

First we calculate the proportions of birth, hybrid, and death events with respect to all events. These values are defined as:

$$
\begin{array}{lll}
fb & = \frac{B}{B+D+H} & \text{fraction of birth events on average,} \\
fh & = \frac{H}{B+D+H} & \text{fraction of hybrid events on average, and} \\
fd & = \frac{D}{B+D+H} & \text{fraction of death events on average.}
\end{array}
$$

A net increase of one in the number of extant taxa, n, is achieved for every birth and hybrid event. (The one exception to this rule is that the initial birth event for the root node actually increases the number of extant taxa from 0 to 2.) However, each death event decreases n by one. Thus, in general, the number of events in the simulation, ne, will be greater than simply n, in order to account for the death events reducing the final number of extant taxa. An average value for ne can be calculated by using the fractions of average birth, hybrid, and death events defined above. The equation, including +1 to account for the root's additional contribution beyond that of a regular birth, is:

$$((fb + fh) * ne) - ((fd) * ne) + 1 = n,$$

which can be rewritten as:

$$ne = \frac{n-1}{fb + fh - fd}$$

to yield the desired output value ne, the average total number of events in a given simulation. Note that the order for this result is simply $O(n)$.

Once the average number of events is known for a simulation, it is trivial to place an upper bound on the average number of vertices (nodes) and edges in a corresponding topology. A single birth or death event is topologically represented as a single vertex. Hybrids however require at least three vertices, and possibly four if the offspring lineage is created and then terminated as an extant leaf. Assuming every event in a given topology is a hybrid, the upper bound on the number of vertices is $4ne$. The bound on the number of edges can be estimated in a similar manner. A death event only involves the single inbound edge to the vertex where the lineage dies. There are three edges associated with a birth vertex, one inbound connecting a vertex to its parent, and two outbound edges providing the connections to its children. Hybrid events however can involve up to seven edges – four to connect the parents to their parents and their non-hybrid offspring in addition to the 3 edges associated with the hybrid node itself. Note that by counting inbound and outbound edges for events, we are over-estimating the number of edges by double counting edges that serve as outbound and inbound connectors. The only implication of this fact is that the upper bound is not as tight as possible. Thus, assuming all events are hybrid events requiring seven edges we have an upper bound on the number of edges for the topology as $7ne$.

As depth-first search (DFS) is a common algorithm required for our subsequent analyses, we also calculate that value now. From [7], the complexity for DFS is $O(V + E)$. Using our results for the vertex and edge counts from above, we find that a DFS of an average network topology will be $0(11ne)$, which reduces to simply

$O(ne)$. With respect to simulation input parameters, the DFS complexity further reduces to $O(n)$, where recall n is the number of extant taxa.

A.2 Network Simulation Model

Generating a network, with the model presented in Chapter 2, is comprised of five main components: birth events, death events, hybrid events, end of simulation work, and outgroup generation. Thus we write:

$$Generation = BRT + DRT + HRT + EOSRT + OGRT$$

where the variables correspond to the run time for each component. Input parameters, in addition to those already defined in Section A.1, that influence the run time include:

sl	sequence length for each homologue
nhg	number of homologous groups of chromosomes per lineage (sometimes referred to as haploid number)
pld	ploidy level
OG	1 if outgroup requested, 0 if not
$MOGT$	maximum number of outgroup tries

$SPRT$ second parent choice is one of the following:
rand when second parent for hybrid(s) chosen randomly,
mhd when second parent for hybrid(s) chosen according to minimum hamming distance,
exphd when second parent for hybrid(s) chosen with exponential function based on hamming distance, or
evd when second parent for hybrid(s) chosen by minimum evolutionary branch distance.

From the previous values, we can derive the following:

ala	average number of active lineages	$\frac{ne}{2}$
tsl	total sequence length for one node/lineage	$nhg * pld * sl$.

Finally we define SG as the time for SEQ-GEN, the software we use to evolve sequences, to process the sequences for one lineage. As the evolutionary models implemented by SEQ-GEN assume independent sites, the run time associated with this effort should be linearly proportional to the total sequence length (tsl).

Having established the above values, we can now calculate the run times for the five main components. The birth and death components can be represented as:

$$
\begin{aligned}
BRT = &\quad fb * ne * SG \Longrightarrow O(n) * O(tsl) \quad \text{and} \\
DRT = &\quad fd * ne * SG \Longrightarrow O(n) * O(tsl).
\end{aligned}
$$

The run time for each of these two pieces is simply the expected number of events multiplied by the SEQ-GEN effort. Both result in run times of $O(n * tsl)$.

The hybrid events are the most complicated and the run time associated with these events can be described as:

$$HRT = (fh * ne) * ((ala * SG) + SPRT)$$
$$\text{where } SPRT = \begin{cases} n & \text{if } rand \Longrightarrow O(ne), \\ n & \text{if } mhd \Longrightarrow O(ala * tsl), \\ n^2 & \text{if } exphd \Longrightarrow O((ala * tsl) + (ala * sl)), \\ n^2 & \text{if } evd \Longrightarrow O(ne^2). \end{cases}$$

This reduces to an overall, worst-case run time of:

$$HRT = O(n) * ((O(n) * O(tsl)) + O(n^2)) = O(n^3).$$

For each hybrid event, all active lineages require an update to their branch lengths and sequences before a second parent can be chosen. The cost of this component is dictated by how the second parents are chosen. The *rand* option is the most simple and could be executed in constant time theoretically, but NetGen first identifies a candidate pool of active lineages (from all lineages) prior to selecting the second parent, thus its execution time is proportional to the number of events. Total sequence length contributes to the *mhd* option as the Hamming distance must be calculated for each possible pair between the known first parent and all active lineages as potential second parents. The *exphd* option is also influenced by total sequence length due to its Hamming distance component. However, it has the additional complication that if a second parent with the calculated target distance is not found, the bounds are expanded. Hence, there may be an incremental search up until the maximum sequence length for a homologue is reached. Finally, Dijkstra's algorithm for finding shortest paths, with a quadratic run time [7], dominates the *evd* option. Thus, when one of the more complicated options is chosen, the overall run time for this component is dominated by $O(n^3)$.

When the simulation reaches the desired number of extant taxa, all the final lineages must have their sequences updated and the outgroup, if requested, is created. The outgroup processing includes generating candidate sequences (limited by $MOGT$) and scoring them with respect to the n extant taxa. The run times for these components are:

$$\begin{aligned} EOSRT &= n * SG \Longrightarrow O(n) * O(tsl) \text{ and} \\ OGRT &= MOGT * (SG + (n * tsl) + n) \Longrightarrow O(n) * O(tsl). \end{aligned}$$

Thus the average simulation run time where n is the number of extant taxa, reduces to:

$$Generation = \begin{cases} O(n^2) & \text{when the second parent option is simple, and} \\ O(n^3) & \text{when the second parent option is complex.} \end{cases}$$

A.3 Reticulate Node Measures

The reticulate node measures of timing, impact, and diversity described in Chapter 3 are topological in nature. All of the measures can be performed with depth-first searches. The diversity measure is the most complicated, and yet requires only four such searches. Therefore, using the DFS result from Section A.1, we find that the reticulate timing, impact, and diversity measures all have a complexity of $O(n)$, where n is the number of extant taxa.

A.4 Reconstruction Algorithm

The reconstruction algorithm presented in Chapter 4 involves: inferring three subtrees, using the Fitch small parsimony algorithm [11] twice, and performing a constant amount of work for creating the single-diploid-hybrid structure. This is expressed as:

$$Reconstruction = 3 * TR + 2 * FSP + C.$$

The subtree reconstructions (TRs) are performed either with maximum parsimony or neighbor joining. As maximum parsimony is NP-hard [72], it is used with only small sets of extant taxa.[1] The exhaustive version of maximum parsimony, where every possible tree is examined, has $O(n!)$ running time, where n is the number of extant taxa. In contrast, the neighbor joining algorithm, again with n extant taxa, has a run time of $O(n^3)$. This low polynomial complexity comes from the n rounds of pairwise comparisons between taxa, where each round involves n^2 or less distance calculations.

The Fitch small parsimony algorithm (FSP) requires making two separate passes of the subtree for the purpose of assigning sequences. These traversals are proportional to the number of taxa in the subtree, which is linearly proportional to the number of extant taxa.

Therefore the run time of the reconstruction algorithm for a single-diploid hybrid is:

$$Reconstruction = \begin{cases} O(n!) + O(n) + O(1) & \text{if } (n_1 \text{ or } n_2 \text{ or } n_3) \leq 10 \\ O(n^3) + O(n) + O(1) & \text{if } (n_1 \text{ and } n_2 \text{ and } n_3) > 10 \end{cases}$$

and reduces to:

$$Reconstruction = \begin{cases} O(n!) & if (n_1 \; or \; n_2 \; or \; n_3) \leq 10 \\ O(n^3) & if (n_1 \; and \; n_2 \; and \; n_3) > 10 \end{cases}$$

where n refers to the total number of extant taxa and n_i refers to the number of extant taxa affiliated with each of the three subtrees.

[1] The default threshold for using maximum parsimony is 10 extant taxa, including the outgroup. If the number of extant taxa exceeds the threshold, neighbor joining is employed.

A.5 Experimental Run-Time Results for NetGen

As the simulation model had the most interesting theoretical analysis due to its multiple components, we decided to conduct experiments using NetGen for those parameters that are most likely to be varied. The following experiments were conducted on a Dell power server.[2]

For the first round of experiments the performance measure was simple elapsed time between the beginning and the end of each simulation run, which is commonly referred to as wall-clock time [53]. Three experiments were conducted as deviations from a birth-only base case. The parameters for the base case were: birth rate = 0.6, 10 extant taxa, 10 pairs of homologous chromosomes, each of length 1,000 (for a total sequence length of 20,000), and no outgroup generation. Run-time dependencies were investigated for varying sequence length, number of extant taxa, and maximum number of outgroup tries. Each point of data is an average wall-clock time based upon 50 executions of NetGen. The standard deviation for each point is plotted, but is often so small that it is not visible. As predicted by the prior analysis and shown in Figures A.1, A.2, A.3, all three of these parameters exhibited a linear behavior.[3]

The second round of experiments examined the performance of NetGen with respect to hybrids. Specifically we wanted to confirm the predicted quadratic and cubic behaviors for the minimum Hamming distance (mhd) and evolutionary distance (evd) approaches of selecting the second parents for the hybridizations. In order to capture this data, CPU time in seconds was measured for each execution of the appropriate selection routine and summed for each run. The numbers of extant taxa

[2] This machine has two dual-core Xeon 5140 processors, 4MB L1 cache, 4GB main memory and runs Debian Linux, kernel 2.6.17.

[3] All lines were fitted with Scilab's [28] *datafit* function, which employs a least squares approach.

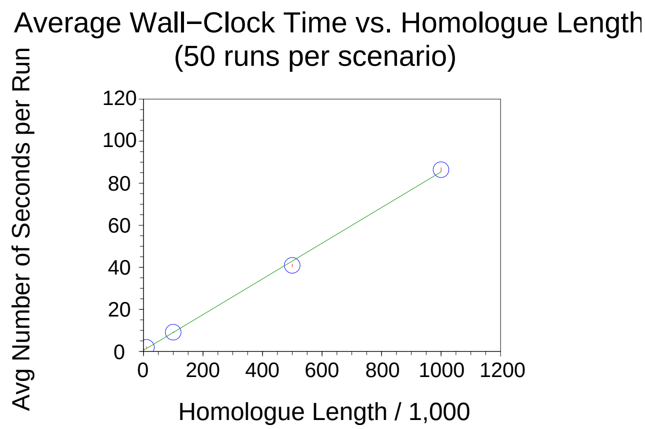


Figure A.1: The sequence length for the 20 individually tracked homologues were varied between 1,000 and 1,000,000 leading to a range of total sequence lengths of 20,000-20,000,000. As predicted, a linear behavior of the run time is exhibited as the sequence length changed.

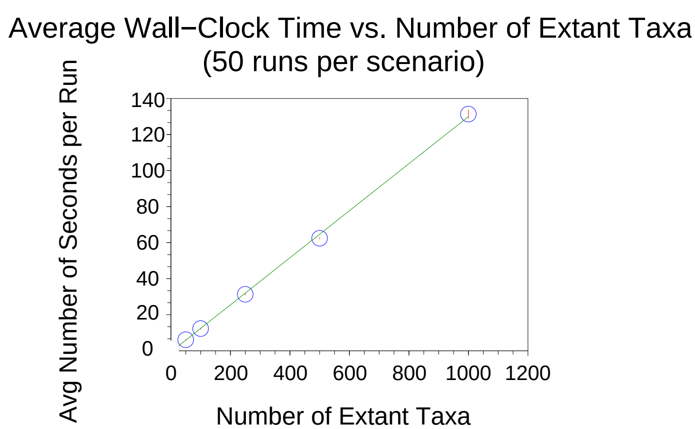


Figure A.2: Varying the number of extant taxa in the birth-only case leads to a linear behavior of the run time, as expected.

were: 250, 500, 1,000, 1,500, and 2000, which were normalized for data analysis purposes. The base case parameters were: birth rate = 0.0, hybrid rate = 0.6, 1 pair of homologous chromosomes (each of length 1,000), and no outgroup generation. These are artificial scenarios in that all events, except for the initial birth at the root, are hybridizations and the sequence length is short compared to the number

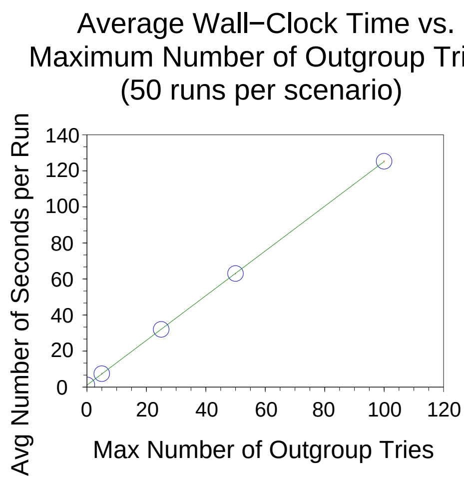


Figure A.3: For all five scenarios, the outgroup similarity range was set to 50-55% (min/max similarity values). This being a narrow range, the maximum number of outgroup tries for each scenario influenced the run time in a linear fashion as expected.

of extant taxa. However, these experiments were designed to highlight the run time for hybrids (HRT) discussed in Section A.2. Figure A.4 illustrates the respective cubic and quadratic behaviors of the evd and mhd options. Although the $O(n^3)$ term from second parent selection dominates the theoretical and these experimental results, one would anticipate actual run times to be influenced the most by the linear factors investigated in the first round of experiments. This is due to the expectation that the number of hybrids in a simulation would be much smaller than the sequence length and/or number of extant taxa.

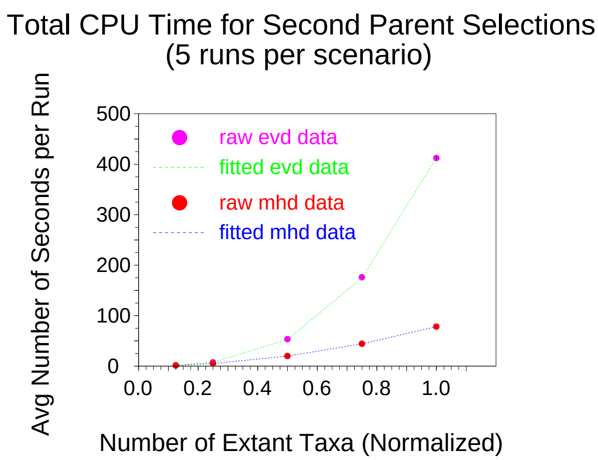


Figure A.4: As predicted by the theoretical analysis, the evolutionary distance (evd) method for selecting second parents behaves in an $O(n^3)$ manner and $O(n^2)$ describes the results for the minimum Hamming distance (mhd) option. The fits were performed with SCILAB's datafit function using $y = a_1 + m_1 * x^3$ and $y = a_2 + m_2 * x^2$ for the evd and mhd options respectively. The values for the cubic function were $a_1 = 1.24$ and $m_1 = 411.57$. The quadratic function values were $a_2 = 0.35$ and $m_2 = 77.84$.

Appendix B

Alternative Ideas for Determining Hybrid-Impacted Extant Taxa

A set of extant taxa believed to be descended from a hybridization event is required for the NetReconstruct algorithm presented in Chapter 4. Although biological research has yielded much information about the nature of plant hybridizations [60, 62, 81], no single, universal characteristic has been found at this time to indicate if a species is a descendant of a single hybridization event [23, 61]. Although it is believed that advances will permit the identification of these sets in the future, at present alternative techniques are required. Biologists currently have a variety of techniques to investigate hypothesized origins for a single extant taxon suspected of being a hybrid progeny [23, 62]. Given a starting taxon of this nature, one can construct a proposed set of hybrid-impacted taxa for use with NetReconstruct.

Clustering is a common approach in computer science. As the name implies, items of a similar nature are placed together. For this situation, the single extant taxon, believed to be of hybrid origin, would be compared and scored against all the other extant species under investigation. Taxa scoring favorably would be considered neighbors of the original taxon and placed in the set. One would expect the scoring technique to be customized for the domain of interest. Hamming distance, for a subsection or the total length of the DNA string(s), or even morphological data, depending on biological insight, are potential scoring mechanisms. A complementary approach would be to start with the complete set of extant taxa and repeatedly segregate the taxa into subsets based upon a score. When the scores between sets were similar, the set containing the taxon believed to be the hybrid descendant would be taken as the hybrid-impacted set.

A more involved approach would incorporate a spectrum of information into the decision of which species should be placed in the same set as the originally identified hybrid descendant. If the hypothesized parents of the hybrid are known to not be extremely divergent,[1] a pre-processing tree reconstruction can be performed to gain some general information about the evolutionary history. Using the reconstructed tree as a guide, an expansion around the initial hybrid descendant could be considered by choosing subroots that include progressively more descendants. At each level, as shown in Figure B.1, a biologist could make an informed guess as to whether the included taxa are similar enough to be in the same impacted set. One method for deciding set inclusion would be to determine potential hybrid origins for one or more of the proposed constituents. If evidence was found for origins similar to the initial hybrid progeny, the set could be expanded; if not, the set could be restricted.

Other pieces of information that could be used in combination to contribute to the set selection process include examining factors such as novel characters and similar chemical compounds. It was found by [62] that many later generation hybrid descendants do not have morphologies common to a blend of parental lineages, but rather taxa tend to have novel characters, perhaps as a result of such factors as the initial instability of a hybrid, or an increased rate of mutation, among other poten-

[1] Tree reconstructions of extant taxa sets containing hybrid descendants are less disrupted when the hybrid's parents are similar [61].

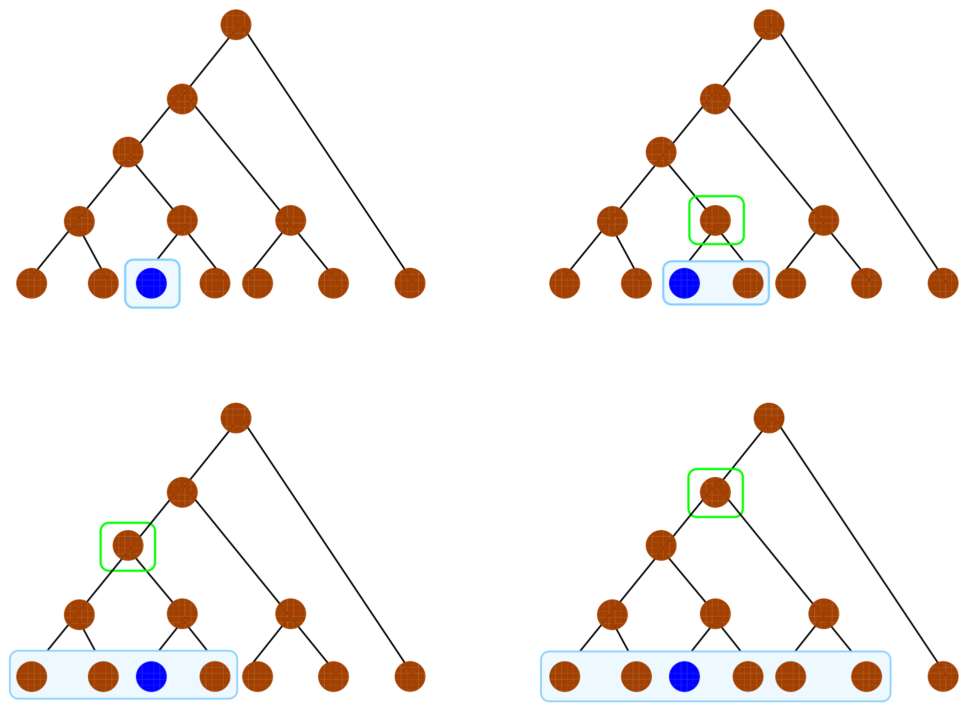


Figure B.1: Proposed hybrid-impacted sets (identified in the blue boxes) increase in size as subroots closer (identified in green boxes) to the root are considered.

tial explanations. Also found by [62] was the tendency for first generation hybrids to contain a mixture of chemical compounds found in the two parental lineages, which may be useful if working with a recent hybridization. Finally, a tool such as SplitsTree [27] could be used to analyze the complete set of extant taxa for sequence data incongruencies, possibly indicating a reticulate history.[2] If a group of extant taxa were found to be closely related with splits, one could choose them as the set of hybrid-impacted taxa.

[2]Tools that detect data incongruencies are known to propose reticulate histories even when none are present.

Appendix C

Future Work

As NetGen provides a variety of parameters that impact the height, shape, size, and branch lengths of a tree or network, further experiments could be tailored for different types of topologies. For example, non-ultrametric topologies with extinctions and/or varying lineage rates, all of which can be generated by the current version of NetGen, but have not been explored in this work, may be a priority when considering a wide span of extant taxa. Another potentially interesting investigation would be to compare single-diploid-hybrid networks generated with different options for selecting the second parent of the hybrid. Or idealized phylogenies with few parameters might be used to explore upper bounds on new reconstruction techniques.

Two tasks that could be undertaken to extend the current functionality of NetGen are the generation of: 1) outgroup subtrees and 2) a function-based option for minimum evolutionary distance, when selecting the second parent of a hybridization. The first improvement would involve generating a subtree of possible extant taxa to serve as an outgroup instead of a single taxon, as is currently performed. A long evolutionary distance would still exist, now between the root of the topology and each of the extant taxa in the outgroup subtree, thus leading to a set of potential taxa from which to choose the outgroup. This may prove beneficial to biologists interested in exploring how outgroup selection influences reconstruction efforts [38], as it is known to do [15]. Another area of the model's behavior that biologists may wish to explore is the influence of how the second parent for a hybrid is selected. By adding an option for a truncated exponential function based on minimum evolutionary distance, similar to the Hamming distance one already implemented, the model's behavior with respect to the second parent selection could be explored. Specifically an investigation could be made into how reticulate topologies and the hybrid progeny are altered when different techniques for second parent selection are employed [37].

A more complex undertaking would be to include homologous sequence evolution in the underlying model of NetGen. Specifically, this would involve coupling a homologue's evolution to the other sequences, possibly both within and across lineages. From a biological perspective, these sequences are expected to evolve over time, but also maintain some similarity with one another. The degree of similarity is determined by population genetic processes, controlling similarity within a species, and natural selection, constraining sequence evolution across lineages. Presently, the NetGen approach of completely independent sequence evolution, does not retain the genetic similarity between and/or across homologues. As simulation time progresses, sequences can diverge significantly from each other. The problem is compounded when a diploid hybrid event occurs. The currently implemented, biological model of randomly selecting homologues from each pair of chromosomes found in the parents, can lead to diverse and potentially unrealistic combinations for the new offspring. Subsequent hybrid events could compound the problem further when the second parent for a hybrid is selected according to Hamming distance.[1] If homologous sequence evolution is modelled, one could also consider modelling homologous,

[1] Although one could artificially redefine Hamming distance across lineages to be calculated between the correct homologues when searching for the second parent, it does not resolve the underlying issue of a hybrid's homologous sequences from being potentially unrealistically diverse.

chromosomal recombination (e.g. crossing over and gene replacement [15]). Although it is not clear what the best modelling approaches for these problems are, an improvement to homologous sequence evolution would result in NetGen producing more realistic sequences and hybrid topologies.

A major extension of this work would be to develop NetReconstruct so that more than one hybrid could be reconstructed. The concept of combining multiple reconstructed subtrees for each hybrid event may appear straightforward on the surface, however the complexity increases drastically when considering the details. The most challenging aspect would be how to deal with overlap between, or among, extant taxa descendant from hybrids. Figure C.1 illustrates a simple occurrence of this difficulty, where an algorithm would be required to merge the two trees since there are extant taxa that exist in both sets.

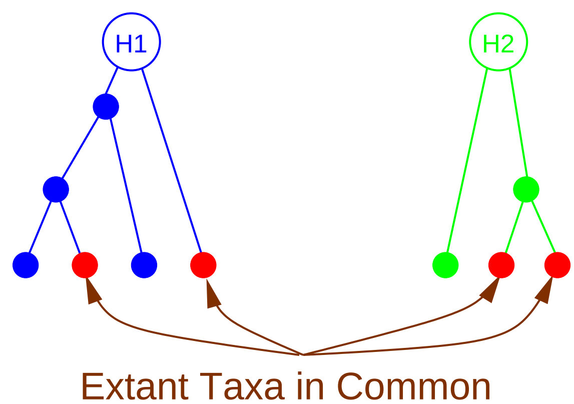


Figure C.1: Overlap of extant taxa in hybrid subtrees. The red extant taxa are impacted by both hybrids and would therefore complicate a merge operation.

As the number of hybrids events increase and the set relationships among extant taxa become more complicated, the task of merging becomes even more daunting (e.g. three impacted sets A, B, and C and there exist extant taxa belonging to each possible *pair* of sets as well as all three). With or without intertwined hybrid subtrees (i.e. networks that have hybridizations events involving hybrids), there is another complication of how the subtrees are arranged chronologically with respect to each other (see Figure C.2).

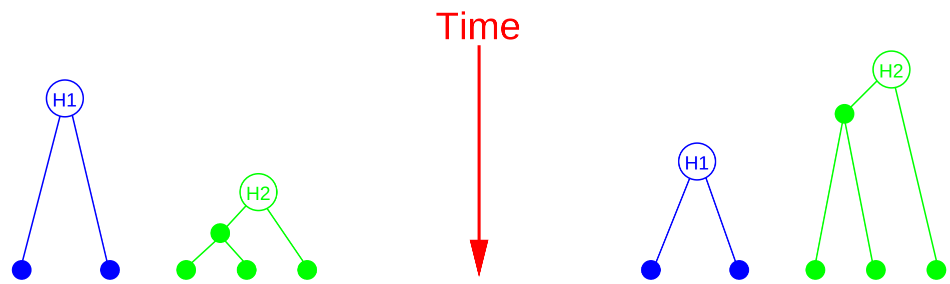


Figure C.2: The axis in middle relates to clock time and highlights that a decision must be made regarding the relative placement of the hybrid subtrees.

As the hybrid impact measure can be calculated from NetReconstruct's input data and the connection between it and timing (presented in the analysis of Chapter 3), a stochastically based method for resolving this issue would be an initial path to investigate. Moreover the tasks of identifying and assigning extant offspring to parents of a hybrid would need to be adapted for the multiple hybrid scenario. Although a version of NetReconstruct with multiple hybrids would be a worthwhile project benefiting the community, these unfortunately are not trivial issues to resolve.

At a smaller scale, modifications could be made to the current one-hybrid version of NetReconstruct. As identified in Section 4.3.5, the closest neighbor option, although resulting in reconstructed topologies with good tripartition scores, tends to underestimate the number of offspring that are descendant from the hybrid's parents, resulting in additional internal edges and slight variations from the source topology. A new algorithm allowing for variation in this identification (e.g. examining a node's closest 3 out of 5 neighbors for example) might yield better results. Furthermore, there is room for improvement with how the parental subtree(s) is(are) assigned to each parent in order to address the tendency of the current technique to assign all parental descendants to only one of the two parent nodes. Modifications in these areas are worthwhile, regardless of whether or not the model is expanded to more than one reticulate node.

As explained in Section 1.2.2, there are three categories of tree reconstruction techniques – distance methods, maximum parsimony, and maximum likelihood. Currently NetReconstruct employs one of the first two approaches, depending on the provided threshold. As analyzed in Section 4.3.5, the neighbor joining approach produces good subtrees for our reconstruction effort and improvements are more likely to be found in ways other than changing the supporting software. However, depending on the application, there may be a need to use a maximum likelihood technique for reconstructing the subtrees. As a preliminary test we generated 1,000 NetGen trees, each having 50 extant taxa and an outgroup (birth rate = 48), and then attempted reconstruction using PHYLIP's neighbor joining [10] and RAxML [73], a tool that performs maximum likelihood based inferences. The performance was measured by computing the bipartition score for 1) source and PHYLIP reconstruction and 2) source and RAxML reconstruction.[2] Figure C.3 shows that the averages are rather close, with no significant difference between the two methods for this set of NetGen source trees. This result further confirms the finding of Section 4.3.5 that a different subtree reconstruction tool was not a priority at this stage of development. It also is consistent with two of the known strengths of maximum likelihood, which are robustness against DNA sampling errors and good performance with shorter sequences [78]. Although not influential factors for this work, they may arise in the future if NetReconstruct is used with real biological data.

Lateral gene transfer is the other common reticulate event found at the inter-

[2]While all the trees were made (either simulated or reconstructed) with an outgroup specified, PHYLIP's neighbor joining routine returns unrooted trees. As the value of nontrivial edges (which is used in the bipartition calculation) changes depending on whether a tree is rooted or not, we adjusted the NetGen source trees and the reconstructed RAxML trees to be unrooted as well in order to facilitate a fair comparison of bipartition scores.

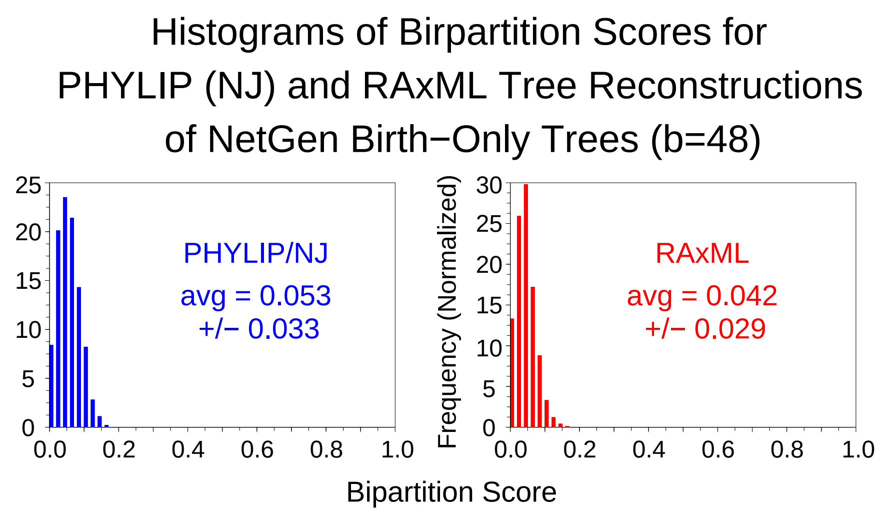


Figure C.3: The bipartition scores for the two tree reconstruction tools (PHYLIP on the left and RAxML on the right) do not differ significantly in this case and thus switching to a maximum-likelihood based method for NetReconstruct's subtrees is not a priority at this time.

specific level. While the focus of this work was on hybridization, the new measures presented in Chapter 3 are applicable not only to hybrid nodes, but to topologies containing reticulate nodes that donate or exchange DNA information without the formation of a separate lineage as well. An event based simulator such as NetGen permits the addition of new events, therefore a new module could be added for generation. However, if reconstructions containing one or more lateral transfer events were desired, a new reconstruction algorithm would be required. A subtree approach, similar to that of NetReconstruct's for inferring hybrids, may prove useful for reconstructing other types of ancestral reticulate events.

Motivated by the phylogenetic tree perspective, another potential research area for networks is bootstrapping [35], which is used to assess the reliability of reconstruction results. For this technique, reconstructions are repeated multiple times (after altering the columns) to determine how "strong" the subtree relationships are in the original, inferred topology [19]. We are not aware of any research that has developed an equivalent for phylogenetic networks. An obvious option would be to do traditional bootstrapping for the three subtrees created by NetReconstruct that would provide values for most, but not all, edges of the topology. A more robust approach would be to implement the underlying principles of bootstrapping for networks. Most likely this would start with a program such as seqboot (available as part of the PHYLIP software [10]) to make multiple sets of extant taxa with randomly selected and rearranged column data [19]. Then, NetReconstruct would have to be executed for each data set and some type of network consensus would be required to summarize the results for each edge. Another option for improving the performance of NetReconstruct and/or extending its applicability to real data would be to offer permutations on the hybrid-impacted extant taxa. Pattengale et. al. has begun to investigate how “rogue” taxa affect subtree mergers of disk-covering methods [52]. This motivates an idea where slight permutations (e.g. adding or dropping one or two taxa) to the set of extant taxa descendant from the hybrid, may prove useful for reconstructing more reliable hybrid subtrees. This would potentially compensate for errors or inconsistencies in the identification of impacted taxa, which is not an exact process as NetReconstruct currently assumes. Subsequently, experiments assessing how perturbations in the hybrid-impacted set affect NetReconstruct's performance would be of benefit to the biological community interested in using the software.

Finally, modifications to the underlying reconstruction approach could be also explored. In a presentation by Linder and Moret [36], the problems of data loss, inadequate sampling, and levels of reticulation were highlighted as issues confounding reticulate reconstruction. Linder proposed that increasing the number of samples and markers are the most likely methods for correcting these deficits. In general, any reconstruction can be subject to these problems, so all are areas for improvement. As part of a private communication following the presentation, Linder and the author discussed ideas motivated by population genetics, which both had pondered independently. Essentially, instead of a single sample representing a whole species (as is common for many interspecific efforts), sets could be implemented. If dealing with DNA sequences collected from the field, multiple samples of the same species could be used, or in the case of simulated data, slight permutations in the sequences could be made by a program such as Seq-Gen. Then reconstruction efforts could be generated either as a consensus of multiple runs using different or randomly chosen individuals from each set during a single run, or a "composite individual" could be created prior to a single reconstruction run. It is believed these two approaches have the potential to minimize the odds of common genes being overlooked and unusual ones having a disproportionate influence on any reconstruction results.

References

[1] D. J. Aldous. Stochastic models and descriptive statistics for phylogenetic trees, from yule to today. *Statistical Science*, 16:23–34, 2001.

[2] J. Archie, W. Day, J. Felsenstein, W. Maddison, C. Meacham, F. Rohlf, and D. Swofford. The newick tree format, 1986. Summarized online at `http://evolution.genetics.washington.edu/phylip/newicktree.html`.

[3] AToL. Assembling the Tree of Life. http://www.phylo.org.

[4] D.A. Bader, B.M.E. Moret, T. Warnow, S.K. Wyman, and M. Yan. *GRAPPA (Genome Rearrangements Analysis under Parsimony and other Phylogenetic Algorithms)*. `www.cs.unm.edu/~moret/GRAPPA/`.

[5] Francois Cellier. *Continous System Modeling*. Springer-Verlag, 1991.

[6] CIPRES. Cyberinfrastructure for Phylogenetic Research. http://www.phylo.org.

[7] Cormen, Leiserson, and Rivest. *Introduction to Algorithms*. MIT Press, 1990.

[8] W.H.E. Day. Optimal algorithms for comparing trees with labeled leaves. *Journal of Classification*, 2(1):7–28, 1985.

[9] W.F. Doolittle. Phylogenetic classification and the universal tree. *Science*, 284:2124–2129, 1999.

[10] J. Felsenstein. *Phylogenetic Inference Package (PHYLIP), Version 3.6*. University of Washington, Seattle, 2004.

[11] W. M. Fitch. Toward defining the course of evolution: minimum change for a specific tree topology. *Systematic Zoology*, 20(4):406–416, 1971.

[12] G. Ganapathy, G.B. Goodson, R. Jansen, V. Ramachandran, and T. Warnow. Pattern identification in biogeography. In *Proc. 5th Int'l Workshop Algs. in Bioinformatics (WABI'05)*, pages 116–127, 2005.

[13] G. Ganapathysaravanabavan, , and T. Warnow. Finding a maximum compatible tree for a bounded number of trees with bounded degree is solvable in polynomial time. In *Proc. 1st Int'l Workshop Algs. in Bioinformatics (WABI'01)*, volume 2149 of *Lecture Notes in Computer Science*, pages 156–163, 2001.

[14] P.R. Grant. Hybridization of Darwin's finches on Isle Daphne Major, Galapagos. *Philosophical Transactions: Biological Sciences*, 340(1291):127–139, 1993.

[15] Dan Graur and Wen-Hsiung Li. *Fundamentals of Molecular Evolution*. Sinauer Assoc., Sunderland, Mass., 2000.

[16] B.T. Grenfell, O.G. Bybus, J.R. Gog, J.L.N. Wood, J.M. Daly, J.A. Mumford, and E.C. Holmes. Unifying the epidemiological and evolutionary dynamics of pathogens. *Science*, 303:327–332, 2004.

[17] X. Gu and W. Li. Higher rates of amino acid substitution in rodents than humans. *Molecular Phylogentics and Evolution*, 1(3):211–214, 1992.

[18] G. J. Hahn and S. S. Shapiro. *Statistical Models in Engineering*. John Wiley and Sons, 1994.

[19] B.G. Hall. *Phylogenetic Trees Made Easy*. Sinauer Assoc., Sunderland, Mass., 2001.

[20] F.W. Hamming. Error detecting and error correcting codes. *The Bell System Technical Journal*, 26(2):147–160, 1950.

[21] Theodore E. Harris. *The Theory of Branching Processes*. Dover Publications, 1963.

[22] S.B. Heard. Patterns in phylogenetic tree balance with variable and evolving speciation rates. *Evol.*, 50:2141–2148, 1996.

[23] M.J. Hegarty and S.J. Hiscock. Hybrid speciation in plants: new insights from molecular studies. *New Phytologist*, 165:411–423, 2005.

[24] J. Hein. Reconstructing evolution of sequences subject to recombination using parsimony. *Math. Biosciences*, 98:185–200, 1990.

[25] D.M. Hillis, B.K. Mable, and C. Moritz. *Molecular Systematics*. Sinauer Assoc., Sunderland, Mass., 1996.

[26] J. P. Huelsenbeck and F. Ronquist. MrBayes: Bayesian inference of phylogeny. *Bioinformatics*, 17:754b, 2001. Available at `morphbank.ebc.uu.se/mrbayes/`.

[27] D.H. Huson and D. Bryant. Application of phylogenetic networks in evolutionary studies. *Molecular Biology and Evolution*, 23(2):254–267, 2006.

[28] INRIA. `Scilab` software package. Available online from http://www.scilab.org/.

[29] G. Jin, L. Nakhleh, S. Snir, and T. Tuller. Efficient parsimony-based methods for phylogenetic network reconstruction. *Bioinformatics*, 23:e123–e128, 2006. Proceedings of the European Conference on Computational Biology (ECCB 2006).

[30] G. Jin, L. Nakhleh, S. Snir, and T. Tuller. Maximum likelihood of phylogenetic networks. *Bioinformatics*, 22(21):2604–2611, 2006.

[31] J.F.C. Kingman. *Poisson Processes*. Oxford:Clarendon, 1993.

[32] Dmitry A. Konovalvo, Clint Manning, and Michael T. Henshaw. KINGROUP: a program for pedigree relationship reconstruction and kin group assignments using genetic markers. *Molecular Ecology Notes*, 4:779–782, 2004.

[33] J.A. Koontz, P.S. Soltis, and S.J. Brunsfield. Genetic diversity and tests of the hybrid origin of the endangered yellow larkspur. *Conservation Biology*, 15(6):1608–1618, 2001.

[34] J.A. Koontz, P.S. Soltis, and D.E. Soltis. Using phylogeny reconstruction to test hypotheses of hybrid origin in Delphinium section Diedropetala (Ranunculaceae). *Systematic Botany*, 29(2):345–357, 2004.

[35] C. R. Linder and L. H. Rieseberg. Reconstructing patterns of reticulate evolution in plants. *American Journal of Botany*, 91(10):1700–1708, 2004.

[36] C.R. Linder and B.M.E. Moret. Network (reticulated) evolution: Biology, models, and algorithms, 2004. talk given at the DIMACS Workshop on Reticulated Evolution, Piscataway, NJ, USA, slides available at http://dimacs.rutgers.edu/Workshops/Reticulated/slides/slides.html.

[37] R. Linder, 2007. Private communication with author on the topic of hybridization terminology.

[38] R. Linder, 2007. Private communication with author on the topic of outgroups.

[39] M. Matsumoto and T. Nishimura. Mersenne twister: A 623-dimensionally equidistributed uniform pseudorandom number generator. *ACM Trans. on Modeling and Computer Simulation*, 8(1):3–30, 1998.

[40] Ernst Mayr. *What evolution is.* Basic Books, A Member of the Perseus Books Group, 2001.

[41] L. McDade. Hybrids and phylogenetic systematics I: Patterns of character expression in hybrids and their implications for cladistic analysis. *Evol.*, 44:1685–1700, 1990.

[42] B.M.E. Moret. Phylogenetic reconstruction from gene order data, 2003. talk given at the Mathematics of Evolution and Phylogeny workshop, Paris, France, slides available at http://www.lirmm.fr/ guindon/ihp/index.html.

[43] B.M.E. Moret, L. Nakhleh, T. Warnow, C.R. Linder, A. Tholse, A. Padolina, J. Sun, and R. Timme. Phylogenetic networks: Modeling, reconstructibility, and accuracy. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1(1):13–23, 2004.

[44] B.M.E. Moret, U. Roshan, and T. Warnow. Sequence length requirements for phylogenetic methods. In R. Guigo and D. Gusfield, editors, *Proc. 2nd Int'l Workshop Algs. in Bioinformatics (WABI'02)*, volume 2452 of *Lecture Notes in Computer Science*, pages 343–356. Springer-Verlag, 2002.

[45] M.M. Morin. Reconstruction of phylogenetic networks, 2003. PhD proposal, available from the author.

[46] M.M. Morin and B.M.E. Moret. NetGen: generating phylogenetic networks with diploid hybrids. *Bioinformatics*, 22(15):1921–1923, 2006.

[47] L. Nakhleh, G. Gin, F. Zhao, and J. Mellor-Crummey. Reconstructing phylogenetic networks using maximum parsimony. In *Proc. 2005 IEEE Computational Systems Bioinformatics Conference (CSB'05)*, pages 93–102, 2005.

[48] L. Nakhleh, B.M.E. Moret, U. Roshan, K. St. John, and T. Warnow. The accuracy of fast phylogenetic methods for large datasets. In *Proc. 7th Pacific Symp. on Biocomputing (PSB'02)*, pages 211–222. World Scientific Pub., 2002.

[49] L. Nakhleh, J. Sun, T. Warnow, R. Linder, B.M.E. Moret, and A. Tholse. Towards the development of computational tools for evaluating phylogenetic network reconstruction methods. In *Proc. 8th Pacific Symp. on Biocomputing (PSB'03)*, pages 315–326. World Scientific Pub., 2003.

[50] S. Otto and J. Whitton. Polyploid incidence and evolution. *Annual Review of Genetics*, 34:401–437, 2000.

[51] N.D. Pattengale, E.J. Gottlieb, and B.M.E. Moret. Efficiently computing the robinson-foulds metric. Accepted to appear special isse on best papers from RECOMB '06, 2006.

[52] N.D. Pattengale, K.M. Swenson, M.M. Morin, and B.M.E. Moret. Higher fidelity subtree merging for disk-covering methods, 2006. poster at Algorithmic Biology.

[53] D.A. Patterson and J.L. Hennessy. *Computer Organization and Design The Hardware/Software Interface*. Morgan Kaufmann Publishers, Inc., 1998.

[54] D. Posada and K.A. Crandall. Evaluation of methods for detecting recombination from DNA sequences: Computer simulations. *Proc. Nat'l Acad. Sci., USA*, 98:13757–13762, 2001.

[55] D. Posada and K.A. Crandall. Intraspecific gene geneaologies: trees grafting into networks. *Trends in Ecol. and Evol.*, 16(1):37–45, 2001.

[56] D. Posada and K.A. Crandall. The effect of recombination on the accuracy of phylogeny estimation. *J. Mol. Evol.*, 54(3):396–402, 2002.

[57] D. Posada, K.A. Crandall, and E.C. Holmes. Recombination in evolutionary genomics. *Annu. Rev. Genet.*, 36:75–97, 2002.

[58] A. Rambaut and N. Grassly. Seq-Gen: an application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. *Comput. Appl. Biosci.*, 13:235–238, 1997.

[59] E. Renshaw. *Modelling Biological Populations in Space and Time*. Cambridge University Press, 1991.

[60] L. H. Rieseberg. Hybrid origins of plant species. *Annual Review of Ecology and Systematics*, 28:359–389, 1997.

[61] L.H. Rieseberg. The role of hybridization in evolution—old wine in new skins. *American Journal of Botany*, 82(7):944–953, 1995.

[62] L.H. Rieseberg and N.C. Ellstrand. What can molecular and morphological markers tell us about plant hybridization? *Critical Reviews in Plant Sciences*, 12(3):213–241, 1993.

[63] D.R. Robinson and L.R. Foulds. Comparison of phylogenetic trees. *Math. Biosciences*, 53:131–147, 1981.

[64] S. Roch. A short proof that phylogenetic tree reconstruction by maximum likelihood is hard. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 3(1):92–94, 2006.

[65] S. Rootsi, C. Magri, T. Kivisild, G. Benuzzi, H. Help, M. Bermisheva, I. Kutuev, L. Barac, M. Pericic, O. Balanovsky, and et. al. Phylogeography of Y-chromosome haplogroup I reveals distinct domains of prehistoric gene flow in europe. *American Journal of Human Genetics*, 75(1):128–137, 2004.

[66] Sheldon Ross. *Simulation*. Academic Press, 1997.

[67] R.Y. Rubinstein and B. Melamed. *Modern Simulation and Modeling*. John Wiley and Sons, 1998.

[68] N. Saitou and M. Nei. The neighbor-joining method: A new method for reconstructing phylogenetic trees. *Mol. Biol. Evol.*, 4:406–425, 1987.

[69] M.J. Sanderson. `r8s:` inferring absolute rates of molecular evolution and divergence times in the absence of a molecular clock. *Bioinformatics*, 19(2):301–302, 2003.

[70] T. Sang and Y. Zhong. Testing hybridization hypotheses based on incongruent gene trees. *Syst. Biol.*, 49(3):422–434, 2000.

[71] M.H. Schierup and J. Hein. Consequences of recombination on traditional phylogenetic analysis. *Genetics*, 156:879–891, 2000.

[72] J. Setubal and J. Meidanis. *Introduction to Computational Molecular Biology*. PWS Publishing Company, 1997.

[73] Alexandros Stamatakis. RAxML-VI-HPC: Maximum likelihood-based phylogenetic analysis with thousands of taxa and mixed models. *Bioinformatics*, 22(21):2688–2690, 2006.

[74] G. L. Stebbins. The role of hybridization in evolution. *Proceedings of the American Philosophical Society*, 103(2):231–251, 1959.

[75] J. Stoye, D. Evers, and F. Meyer. Rose: generating sequence families. *Bioinformatics*, 14(2):157–163, 1998.

[76] S. Sul and T.L. Williams. A randomized algorithm for comparing sets of phylogenetic trees. Technical Report TR-CS-2006-9-3, Texas A&M Univ., 2006.

[77] K. Sun, X. Chen, R. Ma, C. Li, Q. Wang, and S. Ge. Molecular phylogenetics of Hippophae L. (Elaeagnaceae) based on the internal transcribed spacer (ITS) sequences of nrDNA. *Plant Systematics and Evolution*, 235:121–134, 2002.

[78] D.L. Swofford, G.J. Olsen, P.J. Waddell, and D.M. Hillis. Phylogenetic inference. In D.M. Hillis, B.K. Mable, and C. Moritz, editors, *Molecular Systematics*, pages 407–514. Sinauer Assoc., Sunderland, Mass., 1996.

[79] A. Tholse. Master's thesis: Phylogeneitc networks, 2003. University of New Mexico, Department of Computer Science.

[80] Tree of Life web project. Hominidae phylogeny. http://tolweb.org/tree/.

[81] B. Vriesendorp and F.T. Bakker. Reconsctructing patterns of reticulate evolution in angiosperms: what can we do? *Taxon*, 54(3):593–604, 2005.

[82] T. Warnow. Detecting language contact in indo-european, 2005. slides available at http://www.cs.utexas.edu/users/tandy/warnow-harvard.ppt.

[83] S.Z. Xu. Phylogenetic analysis under reticulate evolution. *Mol. Biol. Evol.*, 17(6):897–907, 2000.

[84] Z. Yang. PAML Phylogenetic Analysis by Maximum Likelihood, 2004. Available online from http://abacus.gene.ucl.ac.uk/software/paml.html/.

[85] G.U. Yule. A mathematical theory of evolution, based on the conclusions of Dr. J. C. Willis, F.R.S. *Philosophical Transactions of the Royal Society of London. Series B, Containing Papers of a Biological Character*, 213:21–87, 1925.

[86] D. Zwickl. Genetic algorithm approaches for the phylogenetic analysis of large biological sequence datasets under the maximum likelihood criterion, 2006. Ph.D. dissertation, The University of Texas at Austin.