Algorithms, load balancing strategies, and dynamic kernels for large-scale phylogenetic tree inference under Maximum Likelihood

zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften

von der KIT-Fakultät für Informatik
des Karlsruher Instituts für Technologie (KIT)

genehmigte
Dissertation

von

Benoit Morel
aus Nice, France

Tag der mündlichen Prüfung:	27.10.2021
Erster Gutachter:	Prof. Dr. Alexandros Stamatakis
Zweiter Gutachter:	Prof. Dr. Bertil Schmidt

Hiermit erkläre ich, dass ich diese Arbeit selbständig angefertigt und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie die wörtlich oder inhaltlich übernommenen Stellen als solche kenntlich gemacht habe. Ich habe die Satzung des Karlsruher Institutes für Technologie (KIT) zur Sicherung guter wissenschaftlicher Praxis beachtet.

Heidelberg, 27.10.2021

. . . . . .

(**Benoit Morel**)

Zusammenfassung

Phylogenetik, die Analyse der evolutionären Beziehungen zwischen biologischen Einheiten, spielt eine wesentliche Rolle in der biologischen und medizinischen Forschung. Ihre Anwendungen reichen von der Beantwortung grundlegender Fragen, wie der nach dem Ursprungs des Lebens, bis hin zur Lösung praktischer Probleme, wie der Verfolgung von Pandemien in Echtzeit. Heutzutage werden Phylogenetische Bäume typischerweise anhand molekularer Daten über wahrscheinlichkeitsbasierte Methoden berechnet. Diese Verfahren suchen nach demjenigen Stammbaum, welcher eine Likelihood-basierte Bewertungsfunktion unter einem gegebenen stochastischen Modell der Sequenzevolution maximiert.

Die vorliegende Arbeit konzentriert sich auf die Inferenz Phylogenetischer Bäume von *Arten* sowie *Genen*. Arten entwickeln sich durch Artbildungs- und Aussterbeereignisse. Gene entwickeln sich durch Ereignisse wie Genduplikation, Genverlust und horizontalen Gentransfer. Beide Ausprägungen der Evolution hängen miteinander zusammen, da Gene zu Arten gehören und sich innerhalb des Genoms der Arten entwickeln. Man kann Modelle der *Gen*-Evolution einsetzen, welche diesen Zusammenhang zwischen der Evolutionsgeschichte von Arten und Genen berücksichtigen, um die Genauigkeit phylogenetischer Baumsuchen zu verbessern. Die klassischen Methoden der phylogenetischen Inferenz ignorieren diese Phänomene und basieren ausschließlich auf Modellen der *Sequenz*-Evolution.

Darüber hinaus sind aktuelle Maximum-Likelihood-Verfahren rechenaufwendig. Dies stellt eine große Herausforderung dar, zumal aufgrund der Fortschritte in der Sequenzierungstechnologie immer mehr molekulare Daten verfügbar werden und somit die verfügbare Datenmenge drastisch anwächst. Um diese Datenlawine zu bewältigen, benötigt die biologische Forschung dringend Werkzeuge, welche schnellere Algorithmen sowie effiziente parallele Implementierungen zur Verfügung stellen.

In dieser Arbeit entwickle ich neue Maximum-Likelihood Methoden, welche auf einer expliziten Modellierung der gemeinsamen Evolutionsgeschichte von Arten und Genen basieren, um genauere phylogenetische Bäume abzuleiten. Außerdem implementiere ich neue Heuristiken und spezifische Parallelisierungsschemata um den Inferenzprozess zu beschleunigen.

Mein erstes Projekt, ParGenes, ist eine parallele Softwarepipeline zum Ableiten von Genstammbäumen aus einer Menge genspezifischer Multipler Sequenzalignments. Für jedes Eingabealignment bestimmt ParGenes zunächst das am besten geeignete

Modell der Sequenzevolution und sucht anschließend nach dem Genstammbaum mit der höchsten Likelihood unter diesem Modell. Dies erfolgt anhand von Methoden, welche dem aktuellen Stand der Wissenschaft entsprechen, parallel ausgeführt werden können und sich einer neuartigen Lastverteilungsstrategie bedienen.

Mein zweites Projekt, SpeciesRax, ist eine Methode zum Ableiten eines gewurzelten Artenbaums aus einer Menge entsprechender ungewurzelter Genstammbäume. Berücksichtigt wird die Evolution eines Gens unter Genduplikation, Genverlust und horizontalem Gentransfer. SpeciesRax sucht den gewurzelten Artenbaum, der die Likelihood-basierte Bewertungsfunktion unter diesem Modell maximiert. Darüber hinaus führe ich eine neue Methode zur Berechnung von Konfidenzwerten auf den Kanten des resultierenden Artenbaumes ein und eine weitere Methode zur Schätzung der Kantenlängen des Artenbaumes.

Mein drittes Projekt, GeneRax, ist eine neuartige Maximum-Likelihood-Methode zur Inferenz von Genstammbäumen. GeneRax liest als Eingabe einen gewurzelten Artenbaum sowie eine Menge genspezifischer Multipler Sequenz-Alignments und berechnet als Ausgabe einen Genstammbaum pro Eingabealignment. Dazu führe ich die sogenannte *Joint Likelihood*-Funktion ein, welche ein Modell der Sequenzevolution mit einem Modell der Genevolution kombiniert. Darüber hinaus kann GeneRax die Abfolge von Genduplikationen, Genverlusten und horizontalen Gentransfers abschätzen, die entlang des Eingabeartenbaums aufgetreten sind.

Abstract

Phylogenetics, the study of evolutionary relationships among biological entities, plays an essential role in biological and medical research. Its applications range from answering fundamental questions, such as understanding the origin of life, to solving more practical problems, such as tracking pandemics in real time. Nowadays, phylogenetic trees are typically inferred from molecular data, via likelihood-based methods. Those methods strive to find the tree that maximizes a likelihood score under a given stochastic model of sequence evolution.

This work focuses on the inference of *species* as well as *gene* phylogenetic trees. Species evolve through speciation and extinction events. Genes evolve through events such as gene duplication, gene loss, and horizontal gene transfer. Both processes are strongly correlated, because genes belong to species and evolve within their genomes. One can deploy models of *gene* evolution and to exploit this correlation between species and gene evolutionary histories, in order to improve the accuracy of phylogenetic tree inference methods. However, the most widely used phylogenetic tree inference methods disregard these phenomena and focus on models of *sequence* evolution only.

In addition, current maximum likelihood methods are computationally expensive. This is particularly challenging as the community faces a dramatically growing amount of available molecular data, due to recent advances in sequencing technologies. To handle this data avalanche, we urgently need tools that offer faster algorithms, as well as efficient parallel implementations.

In this thesis, I develop new maximum likelihood methods, that explicitly model the relationships between species and gene histories, in order to infer more accurate phylogenetic trees. Those methods employ both, new heuristics, and dedicated parallelization schemes, in order to accelerate the inference process.

My first project, ParGenes, is a parallel software pipeline for inferring gene family trees from a set of per-gene multiple sequence alignments. For each input alignment, it determines the best-fit model of sequence evolution, and subsequently searches for the gene family tree with the highest likelihood under this model. To this end, ParGenes uses several state-of-the-art tools, and runs them in parallel using a novel scheduling strategy.

My second project, SpeciesRax, is a method for inferring a rooted species tree from a set of unrooted gene family trees. SpeciesRax strives to find the rooted

species tree that maximizes the likelihood score under a dedicated model of *gene* evolution, that accounts for gene duplication, gene loss, and horizontal gene transfer. In addition, I introduce a new method for assessing the confidence in the resulting species tree, as well as a novel method for estimating its branch lengths.

My third project, GeneRax, is a novel maximum likelihood method for gene family tree inference. GeneRax takes as input a rooted species tree as well as a set of (per-gene) multiple sequence alignments, and outputs one gene family tree per input alignment. To this end, I introduce the so-called *joint likelihood* function, which combines both, a model of sequence evolution, and a model of gene evolution. In addition, GeneRax can estimate the pattern of gene duplication, gene loss, and horizontal gene transfer events that occured along the input species tree.

Acknowledgments

First and foremost, I would like to thank my primary supervisor, Prof. Dr. Alexandros Stamatakis, who guided me throughout this research project, and always offered me his unconditional support, for both, scientific, and personal matters. I would also like to thank my co-supervisor, Prof. Dr. Bertil Schmidt, who kindly agreed to review this work.

Furthermore, I wish to thank my colleagues and friends at the Exelixis Lab for all the great moments and inspiring conversations we had together: Diego Darriba, Tomáš Flouri, Paschalia Kapli, Alexey Kozlov, Lucas Czech, Pierre Barbera, Sarah Lutteropp, Rudolf Biczok, Dora Serdari, Ben Bettisworth, Lukas Hübner, Dimitri Höhler, and Julia Schmid. I am grateful to my collaborators: Bastien Boussau, Éric Tannier, Celine Scornavacca, Karen Meusemann, Sebastian Schlag, Paul Schade, Sebastian Gornik, and Tom Williams. I especially wish to thank Gergely Szöllősi, whose wise advice considerably influenced this project.

On a more personal note, I want to express my infinite gratitude to my wife, Judit, for having constantly been on my side, and to my three children, Amélie, Olivia, and Flora. I also wish to thank my parents and brother for their support, and Guillaume Bérard for always being there despite living on the other side of the globe.

Finally, I would like to thank the *Klaus Tschira Foundation* and the *Deutsche Forschungsgemeinschaft* for founding me, and the *Heidelberg Institute for Theoretical Studies* and its entire staff for providing such an excellent working place.

Contents

List of Figures

List of Tables

List of Acronyms

AIC	Akaike Information Criterion
AU	Approximately Unbiased
BIC	Bayesian Information Criterion
CCP	Conditional Clade Probabilities
CLV	Conditional Likelihood Vector
CTMC	Continuous-Time Markov Chain
DNA	Deoxyribonucleic Acid
DTL	Duplication, Transfer, and Loss
EQPIC	Extended Quadripartition Internode Certainty
GFT	Gene Family Tree
GTR	General Time Reversible
HGT	Horizontal Gene Transfer
ILS	Incomplete Lineage Sorting
LCA	Lowest Common Ancestor
MCMC	Markov Chain Monte Carlo
ML	Maximum Likelihood
MPI	Message Passing Interface
MQSST	Maximum Quartet Support Species Tree
MSA	Multiple Sequence Alignment
MSCM	Multi-Species Coalescent Model
NJ	Neighbor Joining
PLF	Phylogenetic Likelihood Function
QPIC	Quadripartition Internode Certainty
RF	Robinson-Foulds
RNA	Ribonucleic Acid
SPR	Subtree Prune and Regraft
SPV	Species Probability Vector
SQ	Speciation-driven Quartet
STA	Species Tree Aware

1. Introduction

1.1 Motivation

The theory of evolution has played an exceptional role in our understanding of the world. This change of perspective has not only influenced our culture, but has also helped us to answer important biological questions. We now know that the diversity of the contemporary living species is the result of complex evolutionary mechanisms that take place at the genome level. The phenotype of a species is to a large extent determined by its genetic material, which is inherited from generation to generation. This genetic material evolves over time: genes evolve within species genomes, and undergo biological events such as gene duplication, gene loss, and Horizontal Gene Transfer (HGT). At the same time, the Deoxyribonucleic Acid (DNA) sequences that form those genes also evolve, through mutation events such as nucleotide insertion, deletion, and substitution.

Phylogenetic trees represent the evolutionary history of a group of species or of a gene. They play a central role in many fields of biology. Examples of their applications include explaining the origin of life [157, 158], understanding the underlying mechanisms of evolution [90, 121], tracking pandemics [61, 106], or predicting protein functions [68, 119]. The phylogenetic tree of a group of species is called a *species tree*, and the phylogenetic tree of a group a genes that evolved from the same ancestral gene is called a *Gene Family Tree (GFT)*.

Species tree and GFT topologies differ from each other, partly because of gene events such as gene duplication, gene loss, and HGT [145]. Species tree and GFT *reconciliation* is the process of explaining this apparent conflict between species and gene evolutionary histories, by identifying the aforementioned gene evolutionary events [17, 37]. A precondition for accurate species tree and GFT reconciliation is the ability to accurately infer the species tree and the GFT topologies: indeed, wrong (species or gene) tree topologies can lead to overestimating the number of gene events, because tree inference error tends to artificially increase the topological differences between the species tree and the GFT [56].

Both, species trees, and GFTs are nowadays inferred from molecular data, that is, from either protein or DNA sequences, that are first assembled into a Multiple Sequence Alignment (MSA). Then, Maximum Likelihood (ML) approaches strive to find the tree that maximizes the probability of observing this MSA under a given stochastic model of sequence evolution. However, by solely relying on the sequences, those methods ignore the relationship between species and gene family histories. Understanding and exploiting this relationship between the species tree and the GFTs currently constitutes one of the most important challenges in phylogenetics [17].

Furthermore, in the last decades, the throughput of DNA sequencing has dramatically increased, while its cost has dramatically dropped at the same rate. As a result, the quantity of available sequence data has been rising at an exponential rate [141]. This data avalanche causes a considerable computational challenge for phylogenetic analyses. For instance, the 1KP Transcriptomes Initiative [67] sequenced the transcriptomes of 1,124 plant species, and the 10KP Genome Sequencing Project [26] plans to extend this number to 10,000 before the year 2023. In addition, it is now common to conduct phylogenetic analyses on datasets with thousands of gene families [21, 49, 158]. To process those increasingly large datasets, there is an urgent need for tools that implement faster algorithms as well as more efficient parallelization strategies.

The goal of this thesis is to develop new ML methods for both, species tree, and GFT inference, that can exploit the intricate relationship between species and gene evolutionary histories, and that can process thousands of gene families in a reasonable amount of time.

1.2 Scientific contribution

My main contribution consists in the development of three tools (ParGenes, GeneRax, and SpeciesRax) that address some of the challenges described above. The inputs and outputs of those tools are shown in Figure 1.1.

First, I developed ParGenes, a parallel pipeline for GFT inference that can simultaneously process multiple gene families. It takes as input one MSA per gene family. For each gene family, ParGenes first runs Modeltest [31] to determine the model of sequence evolution that best fits the MSA. Then, it runs RAxML-NG [76], in order to search for the GFT that maximizes the probability of observing the input MSA, under the selected model of sequence evolution. In addition, ParGenes can also involve RAxML-NG in a way such as to compute the bootstrap support values [46] for the output GFTs. To efficiently and simultaneously process multiple gene families of heterogeneous sizes, ParGenes deploys both, a novel parallelization scheme, and a novel scheduling strategy. ParGenes was described in a *Bioinformatics* application note [103].

Secondly, I implemented SpeciesRax, a parallel tool that infers a rooted species tree from a set of GFTs (typically inferred using ParGenes). SpeciesRax strives

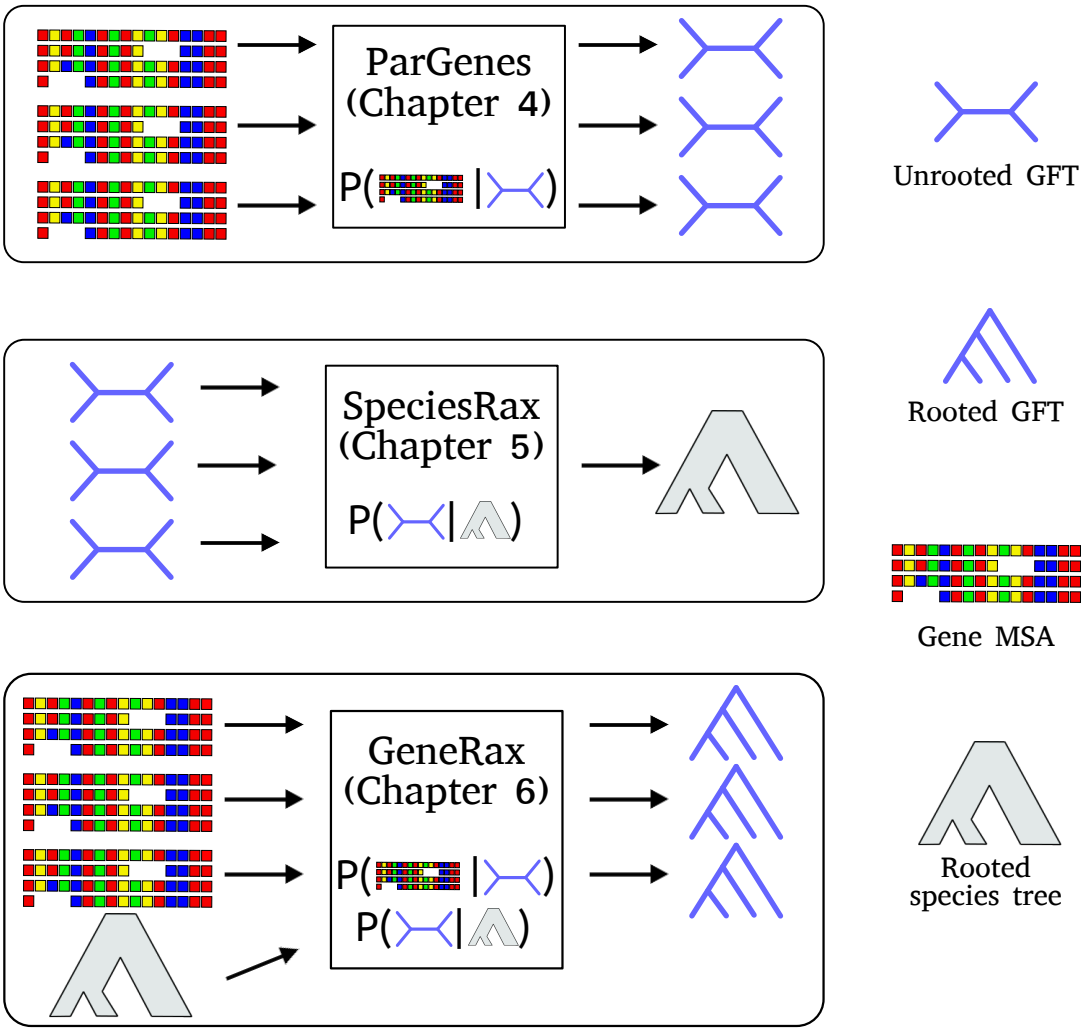


Figure 1.1: Inputs and outputs of ParGenes, SpeciesRax, and GeneRax. ParGenes takes as input a set of MSAs, and returns a set of unrooted GFTs by optimizing the phylogenetic likelihood, that is, the probability of observing an MSA given an unrooted GFT. SpeciesRax takes as input a set of unrooted GFTs, and returns a rooted species tree by optimizing the reconciliation likelihood, that is, the probability of observing the GFTs given the species tree. GeneRax takes as input a set of MSAs and a rooted species tree, and returns a set of reconciled as well as rooted GFTs by maximizing the joint likelihood, that is, the product of the phylogenetic likelihood and the reconciliation likelihood.

to find the species tree that maximizes a likelihood score under a model of gene evolution that accounts for gene duplication, gene loss, and HGT events.

Finally, I released GeneRax, a parallel tool for species tree aware GFT inference. GeneRax takes as input a rooted species tree (typically inferred using SpeciesRax), and one MSA per gene family. GeneRax infers one rooted GFT per gene family by taking into account both the input species tree and the input gene family MSA. To this end, I implemented a *joint likelihood* function, which is the product of two likelihood functions: first, the probability of observing the MSA given the GFT, under any standard model of sequence evolution, and second, the probability of observing the GFT given the species tree, under a dedicated model of gene evolution. For each gene family, GeneRax returns a rooted GFT and its most likely reconciliation with the input species tree. GeneRax was described in the journal *Molecular Biology and Evolution* [105].

Those three software tools and algorithms (ParGenes, SpeciesRax, and GeneRax) are open source and publicly available on GitHub. In the course of my thesis, I also contributed to several projects which resulted in peer-reviewed publications, but have not been included here.

First, I integrated the *site repeats* technique [74] into our libpll library. This technique accelerates the Phylogenetic Likelihood Function (PLF) (defined in Section 2.3.3) computation, which represents $90 - 95\%$ of the execution time in several standard ML phylogenetic tools, such as RAxML-NG [76], ModelTest-NG [31], and EPA-NG [11]. My implementation of site repeats improved the overall runtime of those tools by a factor of 1.2 up to 2.0. Thanks to this contribution, I co-authored several peer-reviewed publications describing the tools that now use site repeats: RAxML-NG was published in *Bioinformatics* [76], ModelTest-NG in *Molecular Biology and Evolution* [31], and EPA-NG in *Systematic Biology* [11]. Furthermore, the site repeats technique can introduce load imbalance when the PLF computation is parallelized over several computational cores. Therefore, I designed a novel data distribution algorithm to account for site repeats. I presented this method at the *2017 IEEE 19th International Conference on High Performance Computing and Communications* [102]. I also co-supervised a group of students that further improved this data distribution strategy by using a judicious hypergraph partitioning [15] approach. Their method was presented at the *2019 IEEE International Parallel and Distributed Processing Symposium Workshops* [9].

Secondly, with Pierre Barbera (a lab member), I co-developed a pipeline for conducting phylogenetic analyses on SARS-CoV-2 data. Our team used this pipeline to demonstrate that inferring reliable phylogenetic trees on such data is difficult. We concluded that the results of phylogenetic analyses from large numbers of SARS-CoV-2 sequences should be interpreted with extreme caution. In addition, we presented several recommendations for conducting such analyses. In particular, we proposed two *tree thinning* approaches, aimed to reduce the number of sequences to analyze in a "reasonable" way. We published our results in the journal *Molecular Biology and Evolution* [104]. My own contribution consisted in setting up the pipeline (with

Pierre Barbera), and in developing and testing one of the two proposed *tree thinning* methods.

Furthermore, I contributed to the development of TreeRecs, a further GFT correction tool. TreeRecs produces several candidate GFT solutions using a parsimony criterion, and subsequently computes their joint likelihood score (as defined above) to select the best solution. My contribution consisted in integrating the joint likelihood function implemented in GeneRax into TreeRecs. TreeRecs was published as an application note in *Bioinformatics* [27].

Finally, I participated in several empirical data analysis projects. I executed several large-scale phylogenetic analysis for a project aimed to resolve the phylogeny of Antliophora (a clade of insects). The results of this analysis are available on *biorXiv* [95]. In addition, I ran an analysis with GeneRax in order to classify a group of opsin (a protein involved in vision) genes belonging to a group of Cnidaria (a phylum of aquatic animals) species. The results of this study were published in the journal *Molecular Biology and Evolution* [53].

1.3 Structure of the thesis

This thesis is structured as follows: Chapter 2 introduces different evolutionary mechanisms that are relevant to this work, and presents the probabilistic models that are used to describe these mechanisms. Chapter 3 provides an overview of the existing methods for inferring GFTs and species trees. In Chapter 4, I present ParGenes, a parallel tool for simultaneously inferring GFTs from multiple MSAs. In Chapter 5, I describe SpeciesRax, a tool for inferring a rooted species tree from a set of unrooted GFTs in the presence of paralogy. In Chapter 6, I present GeneRax, a tool for species tree aware GFT correction and reconciliation. Finally, I conclude and discuss future work in Chapter 7.

2. Preliminaries: Models of evolution

2.1 Species and evolution

Although a *species* is one of the most fundamental units of biology, its exact definition is subject to controversy among biologists [91, 148]. In the scope of this thesis, we consider a species to be a group of individuals that share a common gene pool and that are able to interbreed.

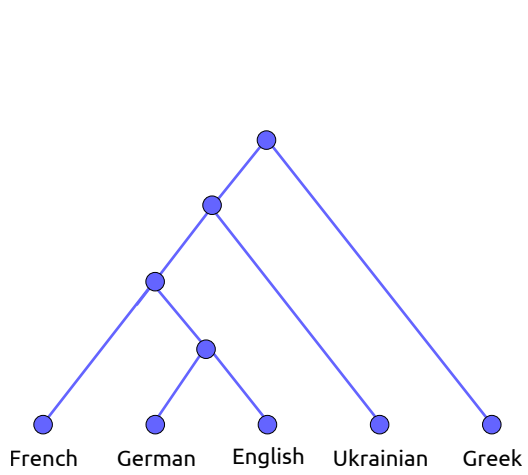
Species are subject to evolutionary forces such as *mutation* and *natural selection*. As a consequence, they evolve over time, both from a molecular and morphological perspective. In particular, a species can either split into two new species that subsequently evolve separately (*speciation event*) or go extinct (*extinction event*). These events form a branching pattern than can be represented by a tree structure, sometimes called the *tree of life*.

It is worth noticing that such a tree representation does not allow for reticulation events such as hybridization or genetic recombination. *Phylogenetic networks* [152] have been proposed to model such events. However, because of their simplicity, trees are still the most widely used structure to represent phylogenetic relationships.

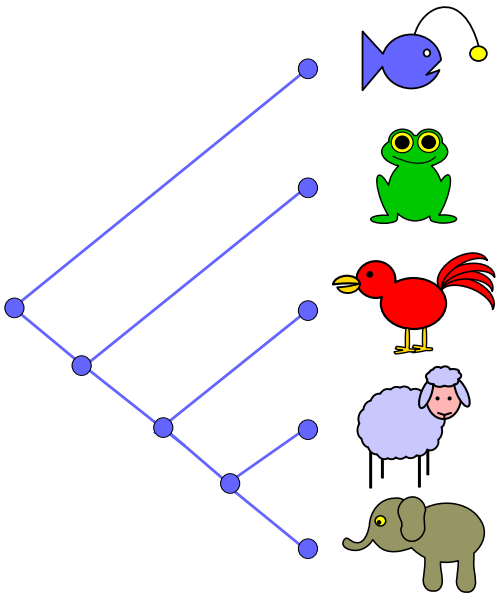
2.2 Phylogenetic trees

A *phylogenetic tree* is a tree structure that represents the evolutionary history of a set of entities. For instance, a *phylogenetic species tree* or *species tree* describes a hypothetical pattern of speciation events that occurred in the past. In a species tree, the internal nodes represent the putative speciation events, and the terminal nodes (*leaves* or *tips*) represent living (*extant*) species, labelled with the corresponding species names. Figure 2.1 shows two examples of phylogenetic trees.

Phylogenetic trees can be either *rooted* or *unrooted*. In rooted trees, one and only one internal node is tagged as *root node* and represents the first speciation event.



(a) Language tree



(b) Species tree

Figure 2.1: Examples of rooted phylogenetic trees. (a) A phylogenetic tree of the languages spoken in our lab, extracted from [54]. (b) A cartoon representation of a vertebrates species tree.

Unrooted trees do not have a root node and thus do not contain information about the chronological order of the speciation events.

A *bifurcating node* is an internal node of degree three or a root of degree two. A *multifurcating node* or *polytomy* is an internal node that is not bifurcating. A *binary* or *bifurcating* tree is a tree that does not contain any polytomy. A *multifurcating* tree is a tree that is not bifurcating. Throughout this thesis, trees are always assumed to be binary, unless stated otherwise.

Every branch b in a tree defines two subtrees with two complementary sets of leaves L and $\bar{L}$. Such a pair $(L, \bar{L})$ is called the *bipartition* or *split* induced by b. A bipartition induced by a branch adjacent to a terminal node is said to be *trivial* because it is induced by all trees that share the same leaf set. Let T_1 and T_2 be two trees with the same leaf set, and let B_1 and B_2 be the sets of non-trivial bipartitions of T_1 and T_2, respectively. The *Robinson-Foulds (RF) distance* [123] between T_1 and T_2 is defined as follows:

$$RF(T_1, T_2) = |B_1 \cup B_2| - |B_1 \cap B_2|$$

An unrooted tree with n taxa induces $n - 3$ non-trivial bipartitions. Therefore, the maximal RF distance between to trees (if no partitions are shared) is $2(n - 3)$. The *relative RF distance* can be computed as follows:

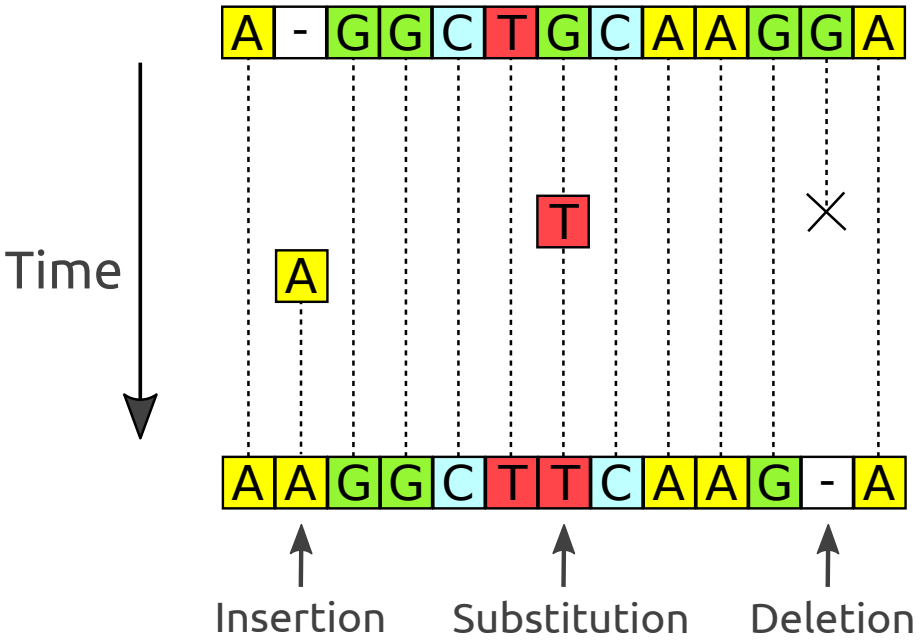


Figure 2.2: Illustration of DNA sequence evolution.

$$nRF(T_1, T_2) = \frac{|B_1 \cup B_2| - |B_1 \cap B_2|}{2(n-3)}$$

2.3 Molecular sequence evolution

Molecular sequences such as DNA or protein sequences evolve over time. This section provides an overview over molecular sequence evolution. First, I introduce the concept of a sequence. Then, I present common models of sequence evolution. Finally, I derive the formula for computing the so-called *phylogenetic likelihood function* under those models.

2.3.1 Sequences and mutations

In this work, *sequences* are string representations of DNA or protein molecules. For instance, a DNA molecule is a succession of *nucleotides* which can be represented by a string formed by the four characters A, C, G, T. Similarly, a protein molecule is a succession of *amino acids*. There are 20 different amino acids, and thus the alphabet for protein sequences comprises 20 characters.

DNA and protein sequences are subject to *mutations* (see Figure 2.2). A *substitution* replaces a base (e.g., A) by another (e.g., T). An *insertion* inserts a base at a given position in the sequence. A *deletion* removes a base at a given position in the sequence. Those mutations accumulate from one generation to another and represent one of the most important driving forces of evolution.

Homologous sequences are sequences that evolved from the same ancestral sequence. Because insertions and deletions change the sequences, it is often necessary to *align*

homologous sequences before comparing them, by inserting gaps ($-$ characters) into the sequences. The result of this operation is a bi-dimensional array called an MSA, in which the rows are the sequences with inserted gaps and the columns (*sites*) are the nucleotides that have evolved from the same ancestral nucleotide.

2.3.2 Substitution models

Sequence evolution can be modeled as a Continuous-Time Markov Chain (CTMC), where the bases are the states and the base substitutions are the transitions. For instance, DNA sequence evolution is modeled via a CTMC with four states corresponding to each of the four nucleotides A, C, G, and T. The values q_{ij} of the Q-matrix Q of the CTMC represent the *instantaneous transition rates* between base pairs i and j. The diagonal values of Q are set such that each row sums to 0. For a given positive real number t and for the bases i and j, let $p_{i,j}(t)$ be the probability that i mutates to j in time t. The elements $p_{i,j}(t)$ form the *transition probability matrix* $P(t)$ and can be computed by matrix exponentiation of Q: $P(t) = e^{Qt}$.

Let π_i denote the stationary frequency of the base i, that is, the equilibrium distribution to which the process converges for large values of t. A substitution model is said to be *time reversible* if $\pi_i q_{ij} = \pi_j q_{ji}$, $\forall i \neq j$. Time reversibility is a crucial property because it allows mathematical simplifications that allows to compute values efficiently. Throughout this thesis, I assume that all substitution models are time reversible.

The substitution models used for phylogenetic inference mainly differ by the constraints imposed to Q. For instance, the General Time Reversible (GTR) model [80, 125] is the most general time reversible substitution model. For DNA models, its transition probability matrix has the following form:

$$Q_{GTR} = \mu \begin{pmatrix} q_{A,A} & a\pi_C & b\pi_G & c\pi_T \\ a\pi_A & q_{C,C} & d\pi_G & e\pi_T \\ b\pi_A & d\pi_C & q_{G,G} & f\pi_T \\ c\pi_A & e\pi_C & f\pi_G & q_{T,T} \end{pmatrix} \tag{2.1}$$

$$q_{i,i} = -\sum_{i \neq j} q_{ij}, i, j \in A, C, G, T \tag{2.2}$$

where μ is the *mean instantaneous substitution rate* that controls how frequently substitutions occur. The variables a, b, c, d, e, and f are *relative base parameters* and control the relative rate of each possible substitution. Simpler DNA substitution models impose more constraints to Q, and have thus less parameters. For instance, the Jukes-Cantor (JC or JC69) model [70] has no free parameter. It imposes equal stationary frequencies for all bases and equal relative rates:

$$Q_{JC} = \begin{pmatrix} -\frac{3}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & -\frac{3}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & -\frac{3}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & -\frac{3}{4} \end{pmatrix} \tag{2.3}$$

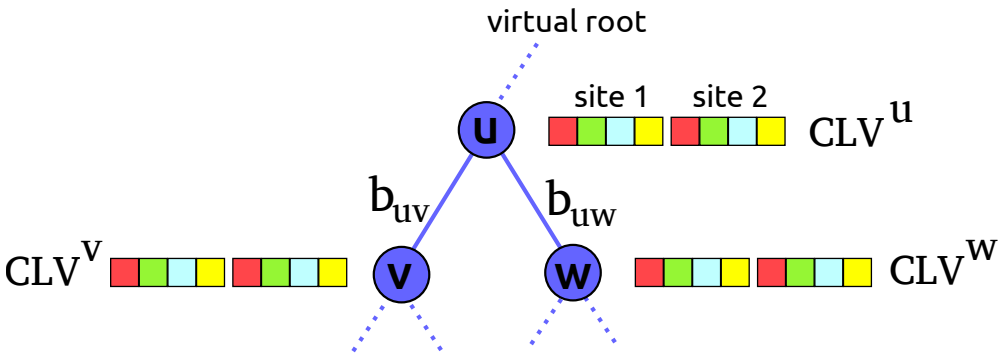


Figure 2.3: Conditional likelihood vectors illustration. Three nodes and there CLVs are represented. u is the parent node of v and w. In this example, the MSA contains DNA data (with four states) and two sites. Dashed lines represent potential subtrees. The Felsenstein pruning algorithm visits the tree via a post-order traversal, and therefore computes the CLVs of nodes v and w before computing the CLV of u.

For protein substitution models, the Q matrix is substantially larger than for DNA data: Q has $20^2 = 400$ elements and thus substantially more parameters that need to be estimated. For instance, the GTR model for proteins has 208 free parameters: $(400-20)/2-1 = 189$ free parameters for the relative base parameters and $20-1 = 19$ free parameters for the stationary frequencies. To avoid over-parametrization, it is common to use preexisting matrices estimated from large collections of empirical data, such as WAG [154] or LG [82].

All these models assume that all sites evolve at the same rate. However, this assumption is often violated, because some regions of the DNA are under higher evolutionary pressure than others. Some more advanced models such as the Γ *model* [160] and the *free rates* model [162] account for this rate heterogeneity among sites. However, for the sake of simplicity, we assume throughout the rest of this chapter a constant evolutionary rate among sites.

2.3.3 The phylogenetic likelihood function

2.3.3.1 Definition

Let G be an unrooted phylogenetic tree with a length assigned to each of its branches. Let A be an MSA from which each sequence is located at one leaf of G. Let N be the set of possible base states (for instance, $N = \{A, C, G, T\}$ for DNA). Let θ be the set of parameters associated to a given model of sequence substitution. The *Phylogenetic Likelihood Function (PLF)* is the probability of observing A given G and θ:

$$L_A(G, \theta) = P(A|G, \theta) \tag{2.4}$$

2.3.3.2 The Felsenstein pruning algorithm

The *Felsenstein pruning algorithm* [45] evaluates the PLF of the tree G by recursively traversing G in a post-order fashion, that is, from the tips toward the root. Since

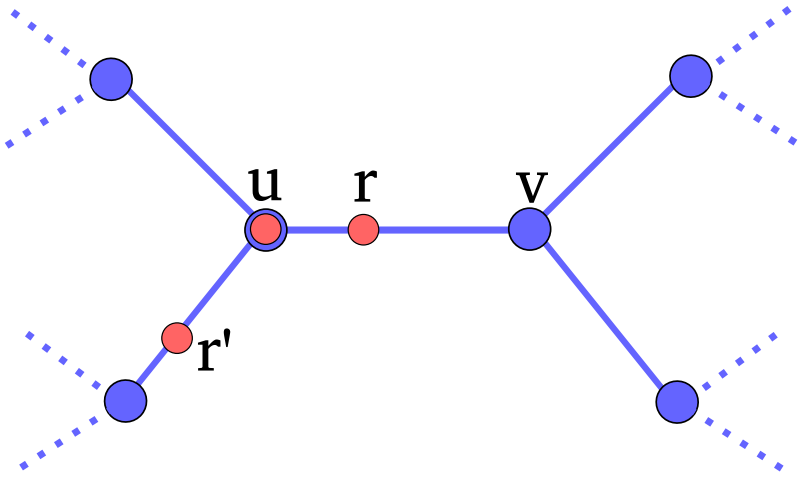


Figure 2.4: The virtual root placement does not affect the PLF. The pink nodes represent different possible placements of the virtual root to evaluate the PLF. For instance, the root can be placed between u and v (r placement), on u (u placement), or on any other branch (r' placement). The likelihood score is the same for any placement. In practice, the virtual root typically coincides with a node (for instance u in this example).

G is unrooted, a *virtual root* is added to G. I discuss the virtual root placement in Section 2.3.3.3. At each step of the traversal, the algorithm fills the so-called Conditional Likelihood Vector (CLV) of each node of G. The variable CLV^u denotes the CLV of node u. The elements $CLV^u_{s,c}$ of CLV^u represent the probability that u is in state c at site s, conditional on the subtree topology and branch lengths.

The algorithm initializes the CLVs of the tips of G by setting $CLV^u_{s,c} := 1$ if the sequence assigned to tip u is in state c at site s, and $CLV^u_{s,c} := 0$ otherwise.

Now, let u be an inner node and let v and w be its child nodes. The values of CLV^u can be computed after CLV^v and CLV^w have been filled (see Figure 2.3): let r be the evolutionary rate and b_{uv} be the length of the branch between u and v . The estimated time between two nodes u and v is equal to $t = r \cdot b_{uv}$. The probability of a transition from a state i to a state j along the branch b_{uv} is thus equal to $p_{i,j}(r \cdot b_{uv})$. The entries of CLV^u can be computed using the recursive formula:

$$CLV^u_{s,c} = \left(\sum_{j \in N} p_{c,j}(r \cdot b_{uv}) \cdot CLV^v_{s,j} \right) \left(\sum_{k \in N} p_{c,k}(r \cdot b_{uw}) \cdot CLV^w_{s,k} \right)$$

The recursion is applied for every state $c \in N$ and every site of the alignment A, for all nodes in G, from the tips to the virtual root, until all CLVs have been computed.

2.3.3.3 Evaluation at the root

Under a time reversible model (see Section 2.3.2), the value of the PLF does not depend on the virtual root placement position. Thus, it can be placed into any branch,

and at any position on that branch (see Figure 2.4). To simplify computations, it is typically placed at one of the two adjacent nodes of the selected branch. Once the values of the CLVs have been filled, the likelihood of an MSA site s can be computed at the virtual root level as follows:

$$L_s = \sum_{i \in N} \sum_{j \in N} CLV^u_{s,i} \cdot \pi_i \cdot p_{i,j}(r \cdot b_{uv}) \cdot CLV^v_{s,j} \tag{2.5}$$

Under the assumption that all sites evolve independently, the PLF is then the product of the per-site likelihoods over the sites of the MSA A:

$$L_A(G, \theta) = \prod_{s \in A} L_s \tag{2.6}$$

In practice, the logarithm of the likelihoods is computed to avoid numerical underflow. Equation 2.6 then becomes:

$$L^*_A(G, \theta) = log(L_A(G, \theta)) = \sum_{s \in A} log(L_s) \tag{2.7}$$

2.4 Gene family evolution

This section introduces the concept of a Gene Family Tree (GFT), and how to compute the probability of observing a GFT given a species tree under the so-called UndatedDTL model. The UndatedDTL model is highly relevant for this thesis, because it allows to study the relationships between species trees and GFTs. In particular, it can be used to perform both species tree inference and GFT inference.

2.4.1 Definitions

The definition of a *gene* is recurrently challenged with new genetic discoveries [50, 115]. In the scope of this thesis, a gene is a basic unit of heredity that belongs to a species. A *gene locus* is a specific, fixed position on a chromosome where a particular gene is located. A *gene sequence* is a molecular sequence associated to a gene. Different gene sequences associated to the same gene locus for the same species are called *alleles* (see Figure 2.5). A set of *homologous genes* is a set of genes that evolved from a common ancestral gene. A *gene family* is a set of homologous genes. I use the term *gene copies* to designate several homologous genes that belong to the same species. A gene family with at least two genes that belong to the same species is a *multiple-copy* gene family. A gene family that is not a multiple-copy gene family is a *single-copy* gene family.

A *GFT* is a phylogenetic tree that represents the evolutionary history of a set of homologous genes (see Figure 2.6(b)). Its terminal nodes correspond to the sequenced genes and its internal nodes correspond to hypothetical ancestral genes. Each leaf in a GFT is mapped to a species. In the case of multiple-copy gene families, several leaves in a GFT can be mapped to the same species.

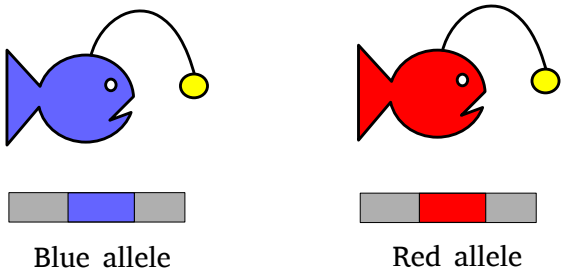


Figure 2.5: Illustration of a multi-allele gene locus. Two fish individuals that belong to the same species but that have different skin colors (blue and red). The grey rectangles represent a region of the chromosomes of each individual. The red and blue rectangles represent the two different alleles located at the same gene locus, responsible for the skin color of the fishes. Those two gene alleles have different DNA sequences and encode different proteins, resulting (in this case) in a different phenotype.

2.4.2 Speciation

When an ancestral species undergoes a speciation event (see Section 2.1), it gives rise to two new species. Every gene in the ancestral species is passed down to each of the new children species, and then starts to evolve separately (see Figure 2.7). A speciation event corresponds to an internal node in a GFT (see Figure 2.6).

2.4.3 Gene duplication and gene loss

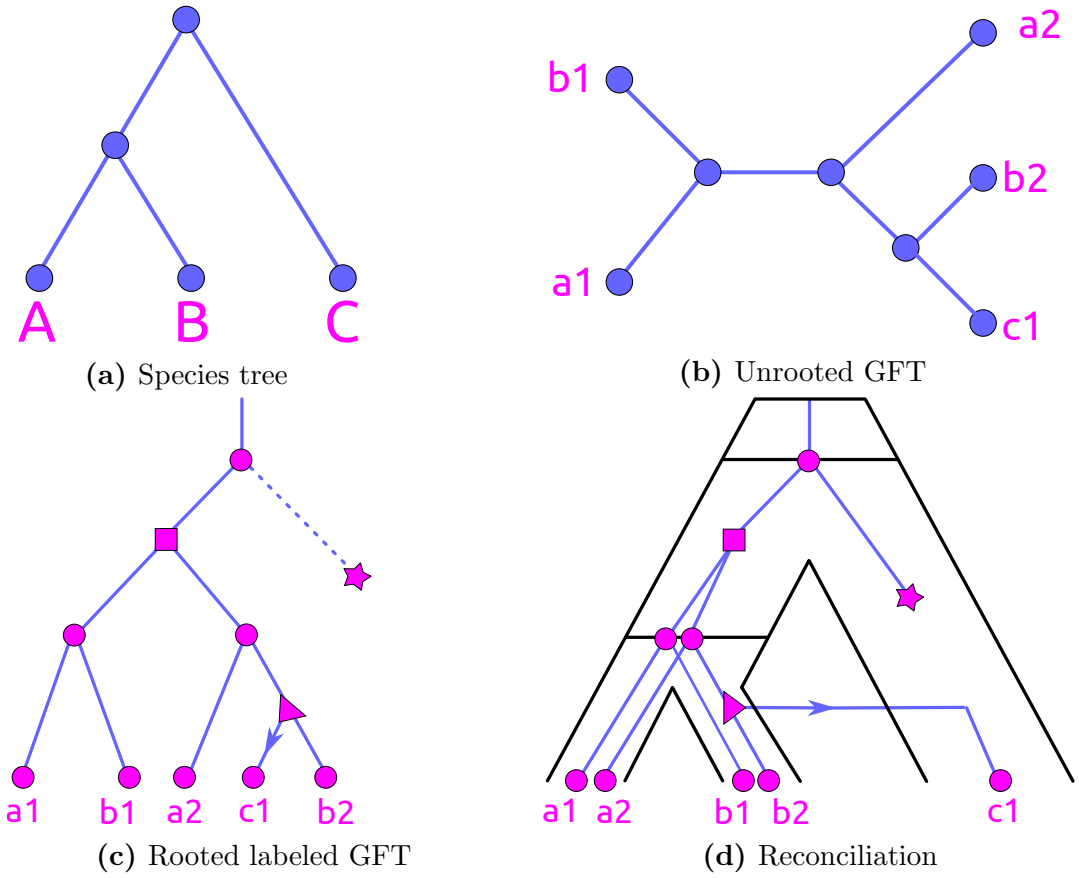
Gene duplication is a mechanism through which a region of DNA that contains a gene is duplicated (see Figure 2.8(a)). Gene duplication events can either duplicate several genes, or occasionally an entire genome [35, 52, 130]. However, in this thesis, the term *gene duplication* designates an event that only duplicates a single gene. Gene duplication is considered to another major force of evolution, because it allows genomes to grow and new functions to emerge [90, 167].

Gene loss is a mechanism through which a region of DNA that contains a gene is lost (see Figure 2.8(b)). Such losses can occur and subsist through generations when a gene is dispensable or redundant [4]. Surprisingly, gene loss might have played a major role in adaption and diversification: gene loss that reduces the fitness of an organism in the short term might sometimes allow for the emergence of new, alternative genes that yield a stronger fitness than the lost genes [60]. In this thesis,the term *gene loss* always designates the loss of a single gene.

In a GFT, a duplication event corresponds to an internal node, and a gene loss is generally not represented (see Figure 2.6)

2.4.4 Horizontal gene transfers

Horizontal Gene Transfer (HGT) is a mechanism through which a gene is transferred from one species to another (see Figure 2.9). It is orthogonal to *vertical* gene events that transmit DNA material from a parent to its offspring. A HGT event corresponds to an internal node in a GFT (see Figure 2.6).



(a) Species tree **(b)** Unrooted GFT

(c) Rooted labeled GFT **(d)** Reconciliation

Figure 2.6: Illustration of a GFT evolving in a species tree. (a) The species tree represents the evolutionary history of three species: A, B, and C. (b) The GFT of five homologous genes: a1 and a2 (belonging to species A), b1 and b2 (belonging to species B), and c1 (belonging to species C). (c) The same GFT, but rooted and labeled by gene events. Circles represent speciations, squares represent gene duplications, stars represent gene losses, and triangles represent HGTs. (d) The reconciliation of the GFT with the species tree. The species tree is represented in black and the GFT in blue.

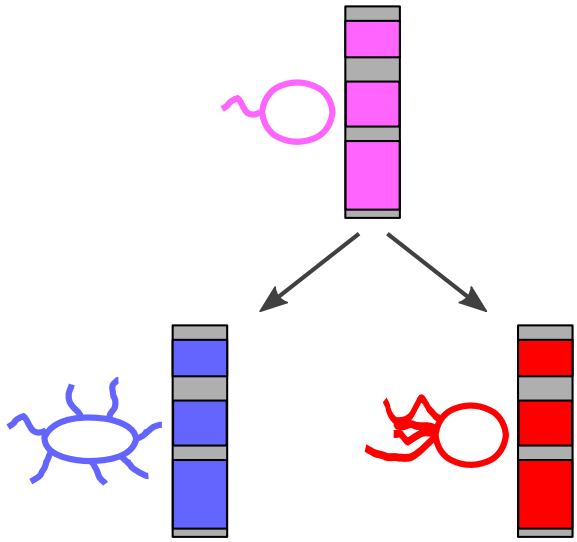
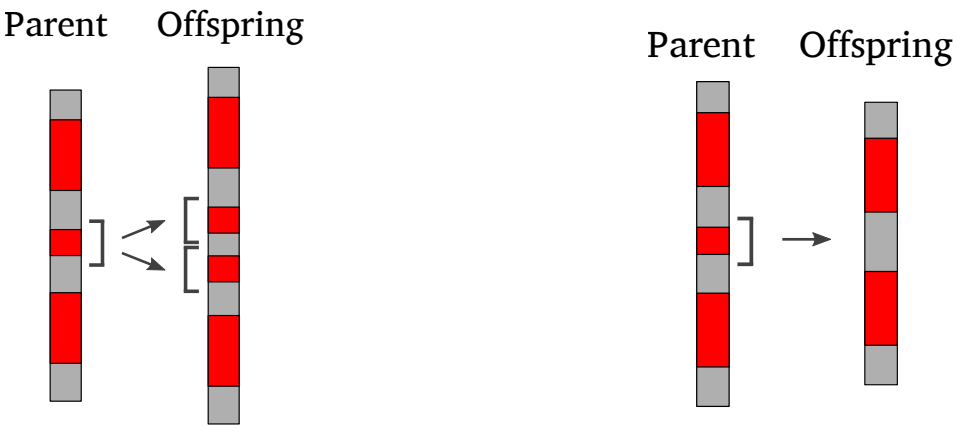


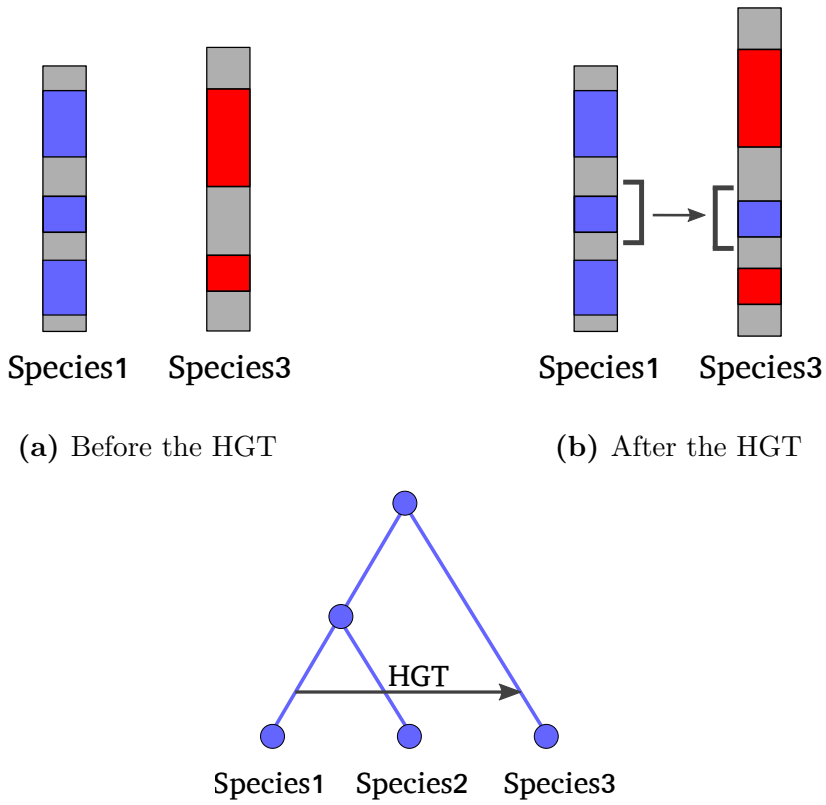
Figure 2.7: Illustration of speciation event at the gene level. An ancestral species (pink color) splits into two new species (blue and red colors). The rectangles represent a fragment of the genomes for each species, and the colored rectangles (pink, blue and red) represent genes. Every gene from the ancestral species is transmitted to both new species.



(a) Gene duplication

(b) Gene loss

Figure 2.8: Illustration of gene duplication and gene loss events. On each figure, the grey rectangles represent fragments of the genomes of an individual and of its offspring. Red regions represent genes. In this example, the duplication and loss events only affect one gene.



(a) Before the HGT

(b) After the HGT

(c) HGT on the species tree

Figure 2.9: Illustration of a HGT. (a) Chromosome regions of two contemporary species (Species1 and Species3) before a HGT event. Blue regions represent genes from Species1 and red regions represent genes from Species3. (b) The same regions after a HGT event from Species1 to Species3. After this event, Species3 has a copy of a gene coming from Species1 in its genome. (c) The same HGT event represented on a species tree with three species. Species2 is not affected by the HGT event between Species1 and Species3.

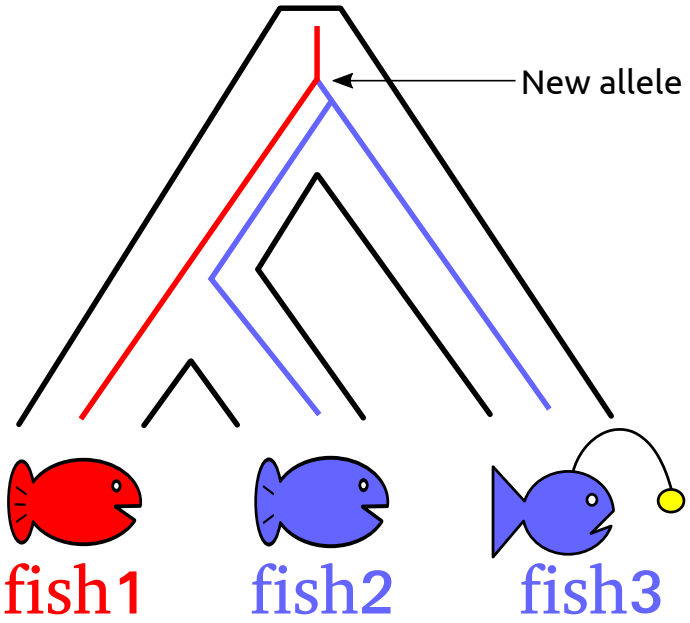


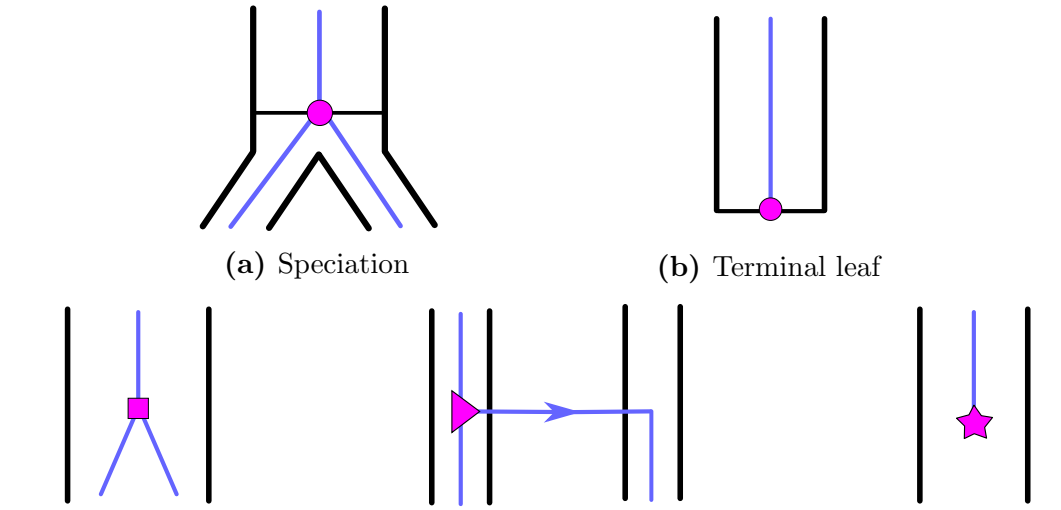
Figure 2.10: Illustration of incomplete lineage sorting. The species tree is represented in dark color and the allele tree in blue and red colors, corresponding to the alleles coding for the blue and red skin, respectively.

2.4.5 Incomplete lineage sorting

Incomplete Lineage Sorting (ILS) is a consequence of gene polymorphism (the presence of several alleles in a population) and one of the main causes of gene and species history discordance [89]. Let us consider a population of red fishes and let us assume that one single gene is responsible for their skin color. In this fictional example, at a given time, a sequence mutation causes the emergence of a new allele that yields a blue skin. After this event, both alleles co-exist in the fish population. Let us now assume that after a large number of generations, the initial fish species evolved into three species (fish1, fish2, and fish3), and that each new species only retained one of the two alleles (blue or red). Figure 2.9 illustrates such a scenario where the blue allele survives in two species (fish2 and fish3) and the red allele survives in the other species (fish1). At the species level, fish1 and fish2 are evolutionarily more closely related to each other than to fish3. But at the gene level, the genes of fish2 and fish3 are more closely related to each other than to the gene from fish1. This phenomenon is called ILS and is often modeled via the so-called Multi-Species Coalescent Model (MSCM) [120].

2.4.6 The UndatedDTL model

The UndatedDTL model [105] is a discrete time Markov model, which starts with a single gene copy on a branch of a given species tree. Subsequently, gene copies evolve independently until, either all copies are observed at the leaves, or every gene copy becomes extinct. On an arbitrary branch of the species tree a gene copy (see also Figure 2.11):



(a) Speciation **(b)** Terminal leaf

(c) Gene duplication **(d)** HGT **(e)** Gene loss

Figure 2.11: UndatedDTL model events. The five possible events that can affect gene evolution under the UndatedDTL model. The black lines represent the species tree. The blue lines represent the GFT. Pink shapes represent the five possible events: a circle for a speciation or a terminal node, a square for a gene duplication, a triangle for a HGT, and a star for a gene loss.

- either duplicates and is replaced by two corresponding gene copies on the same branch (`D` event, with probability $p^{\mathtt{D}}$)
- a new copy is transferred to a random branch that is *not* ancestral to the donor branch, but otherwise drawn uniformly at random from the species tree, while a copy also remains at the donor branch (`T` event, with probability $p^{\mathtt{T}}$)
- is lost (`L` event, with probability $p^{\mathtt{L}}$)
- undergoes a speciation event on internal branches, in which case it is replaced by a copy on each descendant branch (`S` event, with probability $p^{\mathtt{S}} = 1-p^{\mathtt{D}}-p^{\mathtt{T}}-p^{\mathtt{L}}$)
- is observed for terminal branches, that is, arrives in the present and is observed, thus terminating the process (again with probability $p^{\mathtt{S}} = 1 - p^{\mathtt{D}} - p^{\mathtt{T}} - p^{\mathtt{L}}$)

2.4.7 The reconciliation likelihood

The *reconciliation likelihood* is the probability of observing a rooted GFT G under the UndatedDTL model as defined above. It can be calculated by summing over all possible series of `D`, `T`, `L`, and `S` events (henceforth called *scenarios*) that yield a rooted topology that is congruent with G. The sum over all possible scenarios can be computed in two steps [144]. The first step consists in calculating the extinction probability of a gene copy that was initially present on some branch of the species tree. The extinction probability is the sum over all scenarios that do not yield

descendants. The second step consists in summing over all reconciliations of G, where a reconciliation of G corresponds to a specific sequence of D, T, S, and gene copy extinction events, and its probability corresponds to the product over the specific sequence of events.

Let δ, λ, and τ denote the duplication, loss, and transfer intensity parameters that parametrize the D, T, L, and S event probabilities as follows:

$$p^{\mathtt{D}} = \delta/(1+\delta+\tau+\lambda) \tag{2.8}$$
$$p^{\mathtt{T}} = \tau/(1+\delta+\tau+\lambda) \tag{2.9}$$
$$p^{\mathtt{L}} = \lambda/(1+\delta+\tau+\lambda) \tag{2.10}$$
$$p^{\mathtt{S}} = 1/(1+\delta+\tau+\lambda). \tag{2.11}$$

To begin, let e be a branch of the species tree S, and let f and g be its descendant branches. Let $\mathcal{T}(e)$ be the set of species tree branches that can receive a gene via a HGT from e. Because the model does not assume any time information on the species tree other than the order of descent induced by the rooted tree topology, the set $\mathcal{T}(e)$ corresponds to all nodes that are not ancestors of e. The model allows transfers from e to its descendants, because a gene could have evolved along an extinct or unsampled lineage and could subsequently have been transferred back to a descendant of e [144].

The extinction probability E_e, that is, the probability that a gene copy observed on an internal branch e becomes extinct before being observed at the tips of the species tree is:

$$E_e = p^{\mathtt{L}} + p^{\mathtt{S}}\left(E_f E_g\right) + p^{\mathtt{D}}\left(E_e^2\right) + p^{\mathtt{T}}\left(E_e \bar{E}_e\right). \tag{2.12}$$

$$\bar{E}_e = \sum_{h \in \mathcal{T}(e)} \frac{E_h}{|\mathcal{T}(e)|} \tag{2.13}$$

The terms correspond to the i) loss probability, ii) speciation and subsequent extinction probability in both descending lineages (this term must be omitted for terminal branches), iii) duplication and subsequent extinction probability of both copies and finally iv) transfer and subsequent extinction probability of both, the donor copy on branch e, and the transferred copy on branch h.

In Equation 2.12, the value of E_e depends on $\bar{E}_e$, and thus on the extinction probabilities of all species in the species tree. One can estimate $\bar{E}_e$ and E_e for all nodes e in the species tree, by initializing $[E_e]^0 = 0$ and computing:

$$\begin{aligned}[E_e]^n =& p^{\mathtt{L}} + p^{\mathtt{S}}\left[E_f\right]^{n-1}\left[E_g\right]^{n-1} + p^{\mathtt{D}}([E_e]^{n-1})^2 \\ &+ p^{\mathtt{T}}\left[E_e\right]^{n-1} \sum_{h \in \mathcal{T}(e)} \left[E_h\right]^{n-1} / |\mathcal{T}(e)|\end{aligned} \tag{2.14}$$

The probability of observing a rooted GFT G given a rooted species tree S is the sum over the probabilities of all reconciliations of G with S. This includes all D, T,

S, and gene extinction events that may have generated the observed, rooted GFT along the species tree. The algorithm to calculate the sum over all reconciliation histories proceeds from the tips of the rooted species tree and rooted GFT toward their respective roots. Let v and w be descendants of u on G, and f as well as g be descendants of e on the species tree S. For calculating the recursive sum over reconciliations, consider $P_{u,e}$, as the sum over all reconciliations that generate the sub-tree below some internal node u of G starting from a single gene being present on the internal branch e of the species tree S. $P_{u,e}$ is calculated by enumerating *all* possible single D, T, and S events that can result from u on e:

$$\begin{aligned} P_{u,e} = & p^{\mathtt{S}} \left(P_{v,g}P_{w,f} + P_{w,g}P_{v,f}\right) + p^{\mathtt{S}} \left(E_f P_{u,g} + P_{u,f}E_g\right) \\ & + p^{\mathtt{D}} \left(P_{v,e}P_{w,e}\right) + p^{\mathtt{D}} \left(2P_{u,e}E_e\right) \\ & + p^{\mathtt{T}} \left(\bar{P}^e_w P_{v,e} + \bar{P}^e_v P_{w,e}\right) + p^{\mathtt{T}} \left(\bar{P}_{u,e}E_e + \bar{E}_e P_{u,e}\right), \end{aligned} \tag{2.15}$$

$$\bar{P}_{u,e} = \sum_{h \in \mathcal{T}(e)} \frac{P_{u,h}}{|\mathcal{T}(e)|}, \tag{2.16}$$

where $\mathcal{T}(e)$ denotes the branches of S that are *not* ancestors of e. If both e and u are terminal branches, $P_{u,e} = P^{\mathtt{S}}$.

Similar to the expression for the extinction probability, $P_{u,e}$ depends on itself. This can be solved through fixed point iteration analogously to (2.12). Apart from the self dependence, every other term involves either descendant branches in G (u and w), descendant branches in S (f and g), or both. This allows to devise a bottom-up dynamic programming recursion starting at the leaves. Thereby for the leaf g of the GFT and leaf s of the species tree $P(g, s) = 1$, if gene g maps to species s, and zero otherwise.

Given the above, to calculate the reconciliation likelihood, let G be a rooted GFT, r the root of G, S a rooted species tree, s the root of S, $V(S)$ the set of nodes of S, and $N = \{\delta, \tau, \lambda\}$ the set of DTL intensity parameters. The reconciliation likelihood can then be expressed as:

$$L(S, N|G) = \sum_{s \in V(S)} P_{r,s} / \sum_{s \in V(S)} (1 - E_s), \tag{2.17}$$

The division by $\sum_{s \in V(S)}(1 - E_s)$ conditions on survival, as extinct gene families cannot be observed.

2.5 Species and gene histories

In this section, I discuss why the evolutionary histories of species and genes seem to disagree with each other, and how these apparently discordant histories can be reconciled.

2.5.1 Species tree and GFT discordance

The evolutionary history of a set of homologous genes can disagree with the evolutionary history of the corresponding species [89]. In other words, when homologous genes are sampled from a set of species, the GFT related to these genes might differ from the species tree. First, in multiple-copy gene families, the number of genes can differ from the number of species. Secondly, for both multiple- and single-copy gene families, the species tree and GFT topologies can differ. Reasons for this discordance include gene duplication, gene loss, and HGT (see Figure 2.6) as well as ILS (see Figure 2.10). Note that another important source of discordance is GFT reconstruction inaccuracy, for instance because of a lack of signal in the gene sequences [103], or because finding the ML tree is NP-hard [69].

2.5.2 Reconciliation

Let `D`, `T`, `L`, and `S` denote gene duplication, HGT, gene loss, and speciation, respectively. We now assume that species tree and GFT discordance are due to `D`, `T`, and `L` events only (`S` events also occur but are not a cause for discordance). *Reconciling* a species tree and a GFT consists in labeling the GFT with gene events (`D`, `T`, `L`, and `S`) and mapping each node in the GFT to a node in the species tree (see Figure 2.6). Species tree and GFT reconciliation has many applications in biology, such as ancestral genome size estimation [73], gene event rates estimation [57], gene classification [53], or species tree root inference [106, 157].

3. Preliminaries: Methods for phylogenetic tree inference

In this chapter, I provide an overview over the state-of-the-art phylogenetic tree inference methods that are relevant for this thesis. I first present Neighbor Joining (NJ) [128], a generic method for inferring a tree from a distance matrix. Then, I outline a standard pipeline for GFT inference. Thereafter, I introduce several methods for species tree inference. Finally, I describe two classes of methods for GFT correction and reconciliation.

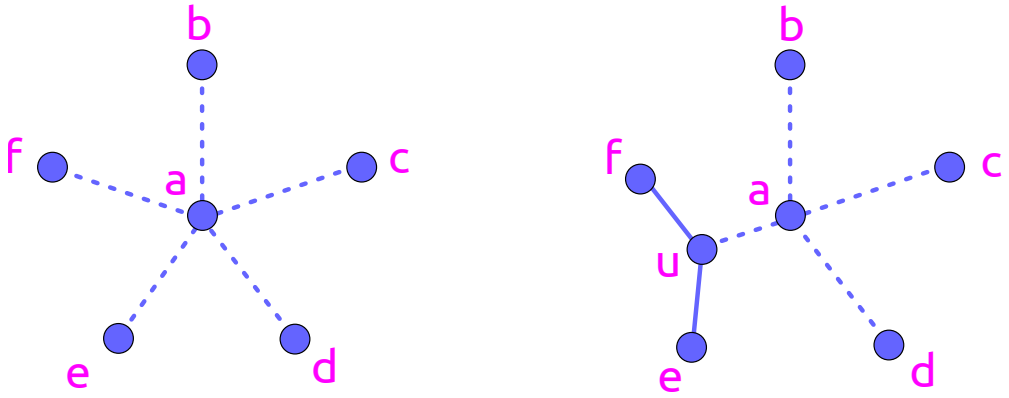
3.1 Neighbor joining

Neighbor Joining (NJ) [128] is a distance-based algorithm for phylogenetic tree inference. It takes as input a distance matrix D of size n, where the elements $D_{i,j}$ are the pairwise distances between taxa i and j where $1 \leq i, j \leq n$ and $i \neq j$. D is typically obtained by computing the pair-wise distances on the input sequences [63, 153] or from a set of GFTs [87]. The NJ algorithm first assigns each taxon to its own cluster, and then works bottom-up by iteratively joining the most similar pairs of clusters, until the whole tree has been built.

The algorithm starts from a *star* tree, in which all terminal nodes are connected to the same unique internal node (see Figure 3.1(a)). Each terminal node is initially assigned to a different cluster. At each iteration of the algorithm, a matrix Q is computed from the distance matrix D of size n as follow:

$$Q_{i,j} = (n-2)D_{i,j} - \sum_{k=1}^{n} D_{i,k} - \sum_{k=1}^{n} D_{j,k}$$

Then, let (e, f) be the pair of nodes such that $Q_{e,f}$ is the smallest element in Q. The nodes e and f are joined into a newly created node u which is connected to the central node (see Figure 3.1). The nodes e and f are removed from the list of



(a) Initialization **(b)** After the first step

Figure 3.1: Illustration of the NJ algorithm. (a) Initial star tree. Every cluster is connected to the central node a. Dash lines represent unresolved branches. (b) After the first step, e and f are joined to a new node u to form a new cluster, and u is connected to the central node a. The solid lines between u and e, and between u and f, represent resolved branches.

clusters and u is added to the list of clusters. Finally, D is updated by calculating the distances between the new cluster u and each of the remaining clusters k as follows:

$$D_{u,k} = \frac{1}{2}\left(D_{k,e} + D_{k,f} - D_{e,f}\right)$$

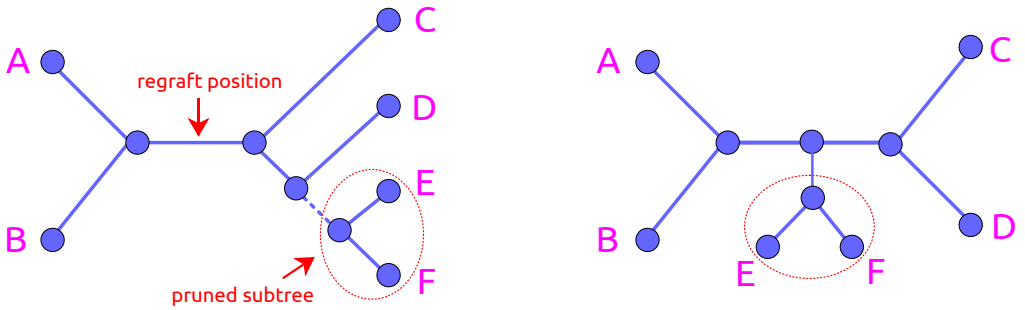
The NJ algorithm iterates until the tree is completely resolved (e.g., until it does not contain any polytomy). It has a cubic complexity with respect to the number of taxa.

3.2 GFT inference

This sections describes how one can infer a GFT from its corresponding MSA using an ML approach. I first describe how to select the most adequate substitution model for a given MSA. Then, I present the tree search heuristic that is commonly used to infer the best-known ML GFT. Finally, I explain how to assess confidence values for the inferred ML GFT.

3.2.1 Model selection

The choice of the substitution model is a crucial step for phylogenetic tree inference. A model with too many degrees of freedom can lead to over-parametrization, whereas a model that is too constrained can fail to capture the complexity of sequence evolution. In addition, some models might fail to represent the process being studied



(a) Before the move **(b)** After the move

Figure 3.2: Illustration of a SPR move of radius 1. The subtree containing the nodes E and F is pruned from the tree. Then, it is reattached one node away ($r = 1$) from its initial position.

(*model misspecification*). An inadequate model might negatively affect the quality of the subsequent GFT inference process.

In absence of reliable prior knowledge, *model selection* tools such as ModelTest-NG [31] or ModelFinder [71] automatically choose the best-fit model. These tools start by generating a plausible (i.e., non-random) tree for the MSA. Then, for each candidate substitution model, they optimize its parameters and the branch lengths in order to maximize the likelihood of this tree. Since the tested models have different degrees of freedom, their likelihoods can not be directly compared. Instead, criteria such as the Akaike Information Criterion (AIC) [2] or the Bayesian Information Criterion (BIC) [131] are used to select the best-fit model.

3.2.2 Maximum likelihood tree search

ML phylogenetic tree inference approaches aim to find the unrooted GFT that maximizes the likelihood score. It is worth noting that the number of distinct unrooted tree topologies $N(n) = \prod_{i=3}^{n}(2i-5)$ grows super-exponentially with the number of taxa n [47]. Therefore, trying all possible tree topologies is in most cases not possible. Furthermore, finding the ML tree has been shown to be an NP-hard problem [124].

A *tree search* is a heuristic that explores a promising subset of the tree topology space. It starts from an initial tree and iteratively alters it in order to incrementally improve its likelihood score. The initial tree can be either randomly generated, provided by the user, or inferred with a faster heuristic. A *Subtree Prune and Regraft (SPR) move* of radius r is an operation that consists in pruning a subtree from a tree and regrafting it at another position that is located r nodes away from its initial position (see Figure 3.2). The tree search strategy implemented in the tool RAxML-NG [76] consists in testing all possible SPR moves within a given radius and applying every SPR move that yields a likelihood improvement. The search stops when no better tree can be found. To avoid local maxima, the same procedure can be repeated from different starting trees.

3.2.3 Bootstrap support values

Heuristic ML tree searches are not guaranteed to find the true tree topology. First, because the search heuristic does not explore all possible tree topologies and might fail to find the best tree (w.r.t. the likelihood score). Second, because this best tree might not correspond to the true tree, for instance because of a lack of signal in the input MSA [103]. *Bootstrap support values* [46] were introduced, for instance, to assess the confidence in an inferred ML tree, or to build a consensus tree out of the bootstrap replicates, under the assumptions that the selected substitution model is well specified and that the sequences have been correctly aligned.

The method consists in randomly subsampling the sites of the input MSA with replacement and to infer a so-called *bootstrap tree* from the new re-sampled MSA using the standard ML search approach. The procedure is repeated several times to produce a set of bootstrap trees (typically 100 or 1000, but adaptive approaches to determine the number of required bootstrap replicates also exist [112]). Let X_{uv} be the bipartition induced by a branch uv. The support value of a branch uv of the inferred ML tree is defined as the fraction of bootstrap trees that induce X_{uv} (see Figure 3.3).

3.3 Species tree inference

This section describes the different methods used to infer a species tree. First, I present the so-called supermatrix approach, which concatenates several MSAs into a single MSA, from which the species tree is subsequently inferred, for instance using ML methods. Second, I describe several GFT-based approaches, which involve GFT inference as an intermediate step to infer the species tree.

3.3.1 Supermatrix methods

Supermatrix methods are currently still the most widely used methods to infer species trees. They take as input single-copy gene families, and concatenate all the per-gene MSAs into a single, large MSA called the supermatrix. Then, they infer a species tree from this supermatrix, typically using ML methods (see Section 3.2.2).

However, extracting single-copy gene families from multiple-copy gene families is a challenging step. When paralog genes (genes that evolved from a duplication event) are selected, the resulting MSA is likely to support a topology that disagrees with the true species tree topology, as explained in Section 2.5.1. *Orthology inference* [6] consists in selecting groups of genes that contain information about speciation events (*ortholog groups*).

In addition, supermatrix methods have been shown to be statistically inconsistent under the Multi-Species Coalescent Model (MSCM) [34, 77]: in presence of ILS (see Section 2.4.5), and under specific conditions [34], they have been shown to converge to an incorrect tree when the size of the available data grows asymptotically.

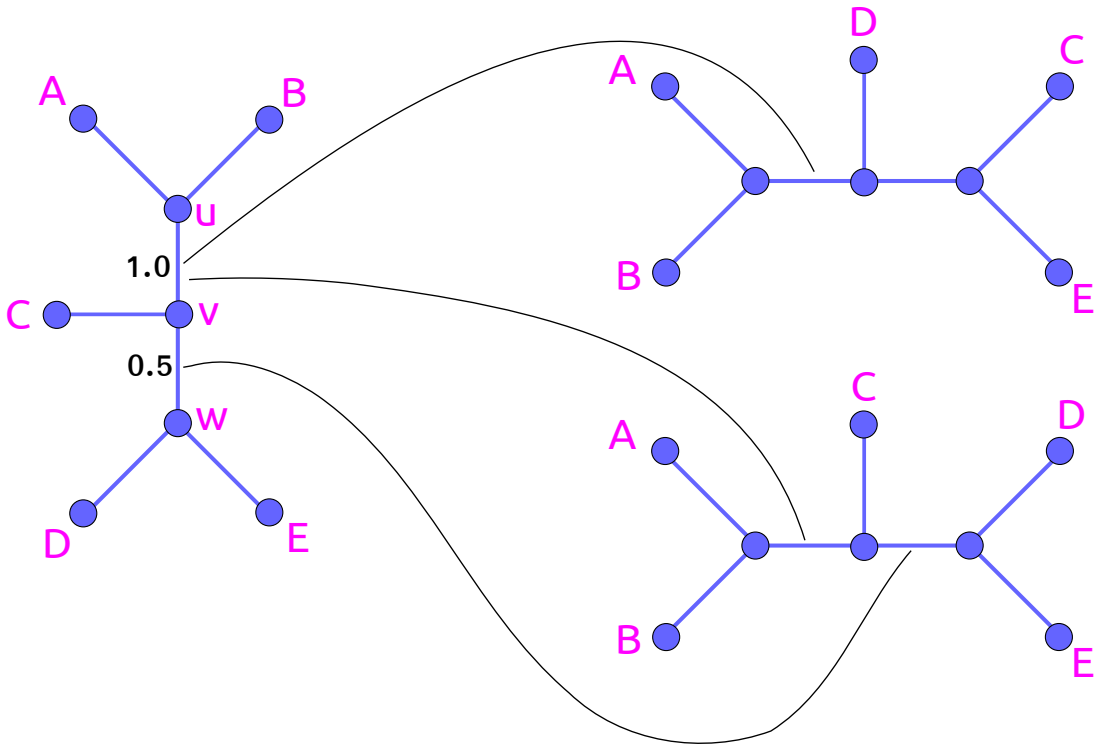


Figure 3.3: Illustration of bootstrap support value computation. In this example, two bootstrap trees (represented on the right) are generated to compute the bootstrap support values on the ML tree (represented on the left). Both bootstrap trees fully support the bipartition induced by the branch uv because they both contain a branch that induces the bipartition $X_{uv} = (A, B|C, D, E)$. Therefore, the normalized support value of uv is equal to 1 (maximum support). However, only one of the two bootstrap trees (the one on the top) supports the bipartition $X_{vw} = (A, B, C|D, E)$: there is no branch in the bottom bootstrap tree that splits the taxon set into A, B, and C on one side, and D and E on the other side. The normalized support value of the branch vw is thus 0.5. Note that the branches connected to terminal nodes are, by definition, always fully supported, and their support values are usually not shown.

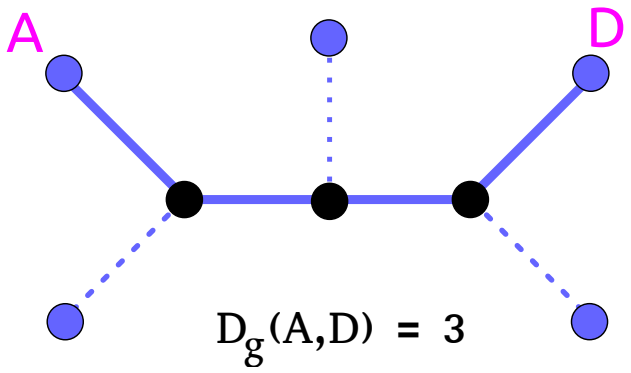


Figure 3.4: The internode distance. In this example, the internode distance between the terminal nodes A and D is $D_g(A, D) = 3$, because there are three internal nodes along the path connecting A to D.

3.3.2 GFT methods

GFT methods aim to alleviate the pitfalls of supermatrix approaches by accounting for the discordance between the species tree and the GFTs. Some GFT methods try to simultaneously estimate the GFTs and the species tree [18], while some others first estimate the GFTs from the per-gene MSAs, and then infer the species tree [87, 101, 151, 166]. In the following, I present some of these GFT methods.

3.3.2.1 NJst

NJst [87] initially computes a distance matrix from a set of unrooted GFTs and then applies the NJ algorithm (see Section 3.1) to reconstruct the species tree.

NJst defines the *internode distance* D_g such that $D_g(x, y)$ is the number of internal nodes on the path between the terminal nodes x and y in a given GFT (see Figure 3.4). NJst computes the distance between two species as the average over the internode distances between all pairs of gene copies mapped to those two species.

More formally, let a and b be two species. Let K be the number of GFTs. Let m_{ak} be the terminal nodes from GFT k mapped to species a. Let x_{iak} be the ith terminal node from the GFT k mapped to species a. NJst defines the distance matrix D_{NJst} as follows:

$$D_{\text{NJst}}(a, b) = \frac{\sum_{k=1}^{K} \sum_{i=1}^{m_{ak}} \sum_{j=1}^{m_{bk}} D_g(x_{iak}, x_{jbk})}{\sum_{k=1}^{K} m_{ak} m_{bk}} \tag{3.1}$$

The species tree is then obtained by applying the NJ algorithm to the matrix D_{NJst}.

3.3.2.2 Quartet methods

A *quartet topology* is an unrooted tree with four taxa (see Figure 3.5). Let G be a GFT and let (a, b, c, d) be four of its taxa. The quartet topology induced by (a, b, c, d)

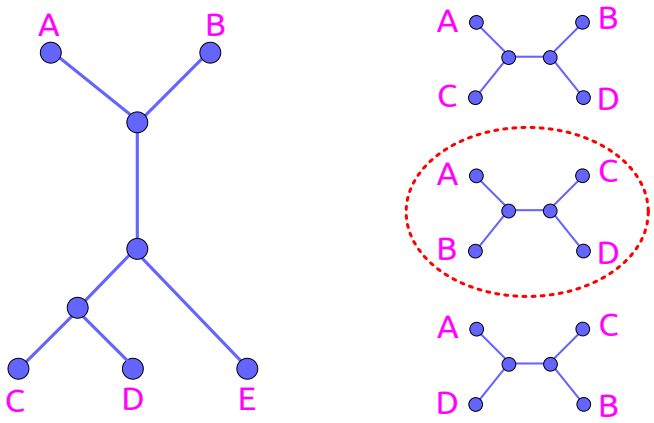


Figure 3.5: Illustration of quartets. On the left, we depict an unrooted tree with five taxa. On the right, the three possible quartet topologies for the set of taxa (A, B, C, D) are shown. The red dashed circle indicates the quartet topology that agrees with the unrooted tree.

from G is the quartet topology obtained by removing all other taxa from G. For a given set of four taxa, there are three possible quartet topologies. A quartet topology agrees with a species tree if the species tree induces this quartet topology. The Maximum Quartet Support Species Tree (MQSST) problem [99] consists in finding the unrooted species tree that agrees with the largest number of quartets induced by the set of GFTs. This problem has been shown to be NP-hard [69].

ASTRAL [98, 99, 165] is a quartet-based method for inferring a species tree from *single-copy* gene families. It implements a heuristic that approximately solves the MQSST problem in polynomial time by constraining the set of species trees to explore. Let $\mathcal{X}$ be the set of bipartitions induced by the input GFTs. Let $\mathcal{Q}$ be the set of species trees whose bipartitions belong to $\mathcal{X}$. ASTRAL-I [99] implements a dynamic programming approach to evaluate all the species trees in $\mathcal{Q}$ without having to explicitly enumerate all the quartet topologies. ASTRAL-II [98] and ASTRAL-III [165] improve the asymptotic runtime of ASTRAL by introducing several techniques, for instance to reduce the size of $\mathcal{Q}$, resulting in an overall time complexity of $O((n \cdot k)^{2.726})$ where n is the number of species and k is the number of families. ASTRAL is statistically consistent under the MSCM [99].

ASTRAL-PRO [166] extends ASTRAL to accept multiple-copy gene families as input and to account for paralogy. In a first step, it approximately roots the input GFTs and tags them with "duplication" and "speciation" nodes. Then, it adapts the original ASTRAL algorithm by enumerating only those quartets that are informative regarding speciation events, using the following definition: a quartet Q on a rooted tagged GFT is a speciation-driven quartet if and only if the Lowest Common Ancestor (LCA) of any three out of four leaves of Q is a speciation node.

3.3.2.3 Maximum likelihood methods

ML methods search for the species tree that maximizes a likelihood score under a given probabilistic model of evolution.

PHYLDOG [18] is an ML method that co-estimates the GFTs and the species tree. It introduces a custom model of gene evolution that accounts for gene duplication and gene loss (`DL`) events, and computes a reconciliation likelihood function $L(S, N|G)$ that describes the probability of observing a GFT G given a rooted species tree S and a set N of `DL` events under this model. PHYLDOG optimizes the so-called *joint likelihood* score, which is the product of the standard phylogenetic likelihood (see Section 2.3.3) and this specific reconciliation likelihood function. For a given gene family i, let S be a species tree, A_i be a gene MSA, G_i be a GFT, and N be the rates of `D` and `L` events. The joint likelihood of a family i is defined as:

$$L(G_i, S, N|A_i) = L(S, N|G_i) \cdot L(G_i|A_i) \tag{3.2}$$

Let $\mathcal{I}$ be a set of gene families, $\mathcal{G}$ be the corresponding set of GFTs, and A be the corresponding set of gene MSAs. The joint likelihood of $\mathcal{I}$ is obtained by multiplying the joint likelihoods over all gene families:

$$L(\mathcal{G}, S, N|A) = \prod_{i \in \mathcal{I}} L(G_i, S, N|A_i) \tag{3.3}$$

PHYLDOG implements a search strategy that explores the space of rooted species trees using SPR moves (see Section 3.2.2). For each candidate species tree obtained via an SPR move, it optimizes each GFT topology to maximize its joint likelihood using an SPR tree search heuristic. The candidate species tree is accepted if the optimized GFTs yield a better overall joint likelihood, and the procedure is repeated until not better species tree can be found.

3.3.2.4 Parsimony methods

Parsimony methods aim to find the rooted species tree S that requires the least number of gene events to *reconcile* (see Section 2.5.2) the input GFTs with S. For instance, DupTree [151] implements a heuristic to find the species tree with the lowest reconciliation cost, measured in terms of number of gene duplications. Similarly, DynaDUP [12] searches for the species tree that requires the lowest number of gene duplication and gene loss events.

3.3.2.5 Robinson-Foulds supertree methods

Let $\mathcal{G}$ be a set of (uniquely labelled) unrooted trees with the same label set and let RF be the RF distance function (see Section 2.2). The RF supertree problem [10] consists in finding the tree S that minimizes:

$$\sum_{G \in \mathcal{G}} RF(S, G)$$

FastMulRFS [101] is a method that extends the RF distance definition and the RF supertree problem to *multi-labelled* trees, that is, trees that can have several leaves with the same label. It provides an algorithm to compute the RF distance between a single-labelled tree (the species tree) and a multi-labelled tree (a GFT, in which each terminal gene node is labelled with the species it belongs to) in polynomial time. FastMulRFS uses a dynamic programming approach to evaluate all the species trees that belong to a constrained search space $\mathcal{X}$, similarly to the quartet-based approach described in Section 3.3.2.2, and solves the multi-labelled RF supertree problem on $\mathcal{X}$ in polynomial time.

3.4 Species tree aware GFT correction and reconciliation

GFTs are typically inferred from their gene MSAs via ML tree search heuristics, as described in Section 3.2.2. However, the lack of signal in the gene MSAs often leads to GFT reconstruction errors [103]. Species Tree Aware (STA) methods aim to exploit the relationship between the GFT and the species tree to leverage additional information for GFT inference. In addition, they can *reconcile* (see Section 2.5.2) the resulting GFT with the species tree, providing useful insights into the gene family history.

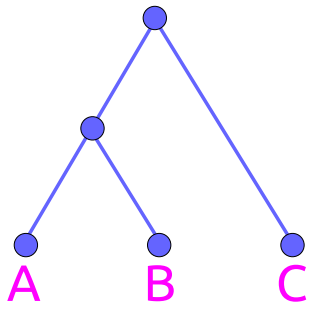
This section describes two standard classes of methods for STA GFT correction and reconciliation methods. First the parsimony methods, that aim to minimize the number of gene duplication, gene loss, and HGT events required to reconcile the GFT with the species tree. Secondly, the amalgamation methods, that sample GFTs under a joint model of sequence evolution and GFT-species tree evolution.

3.4.1 Parsimony methods

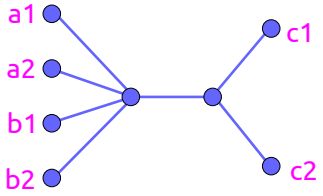
Parsimony methods take as input a rooted species tree and an unrooted, multifurcating GFT (see Figure 3.6(b)). They output a rooted binary GFT and its reconciliation with the input species tree. Parsimony methods aim to find the GFT that requires the lowest number of gene duplication, gene loss, and HGT events to be explained. The multifurcating input GFT is typically obtained by first inferring an ML GFT with its bootstrap support values, and by subsequently contracting the branches whose support value fall under a given arbitrary threshold. The next paragraphs introduce several definitions in order to describe the minimum parsimony problem in the context of GFT correction and reconciliation.

Let G be an unrooted multifurcating tree. An unrooted binary tree G' *agrees* with G if it induces all the bipartitions induced by G (see Figure 3.6(c)). A rooted tree G' *agrees* with G if its unrooted topology agrees with G.

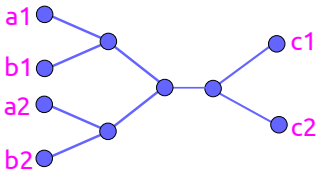
The *reconciliation cost* $C(R_{G',S})$ of a reconciliation $R_{G',S}$ between a rooted binary GFT G' and a rooted species tree S is the sum over the number of gene duplication ($n_{\mathtt{D}}$), gene loss ($n_{\mathtt{L}}$), and HGT ($n_{\mathtt{T}}$) events involved in this reconciliation, weighted by their respective costs $C_{\mathtt{D}}$, $C_{\mathtt{L}}$, and $C_{\mathtt{T}}$ (see Figure 3.6):



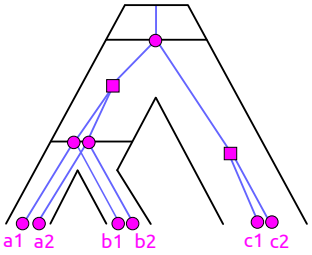
(a) Species tree



(b) Unresolved GFT



(c) Resolved GFT



(d) Most parsimonious reconciliation

Figure 3.6: Illustration of most parsimonious GFT reconciliation. (a) The rooted binary input species tree S. (b) The unrooted multifurcating input GFT G. The gene copies a_1 and a_2 belong to species A, b_1 and b_2 to species B, and c_1 and c_2 to species C. (c) An unrooted representation of G', a binary GFT that agrees with G'. (d) A reconciliation of G' with S: circles represent speciation events and squares represent duplication events. There are neither gene loss nor HGT events involved in this scenario: $n_{\mathtt{D}} = 2$, $n_{\mathtt{L}} = 0$, and $n_{\mathtt{D}} = 0$. With costs $C_{\mathtt{D}} = 10$, $C_{\mathtt{L}} = 1$ and $C_{\mathtt{T}} = \infty$, the overall cost of this reconciliation is 20. G' is the resolution of G that yields the most parsimonious reconciliation with S (note that another equally parsimonious solution can be obtained by exchanging the nodes a_1 and a_2).

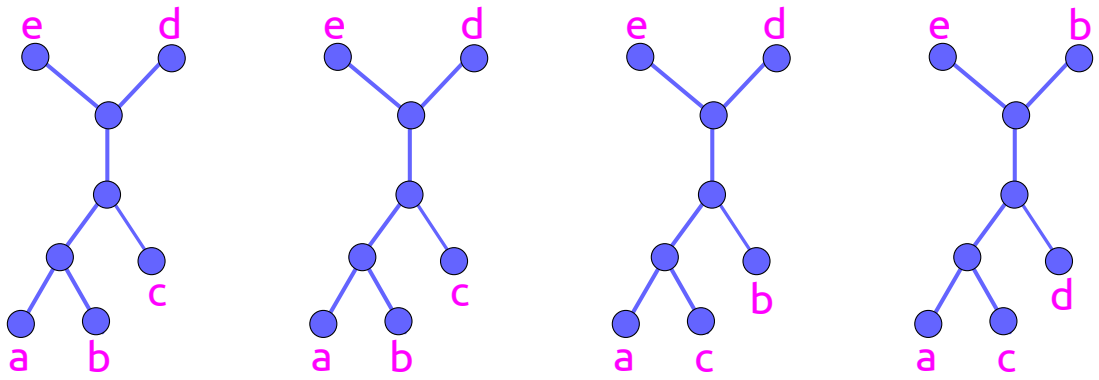


Figure 3.7: Estimating the conditional clade probabilities from a distribution of GFTs. The figure represents a distribution of four GFTs. The clade $\gamma = (A, B, C)$ is observed in the first three trees only. The two first trees split γ into the subclades $\gamma'_1 = (A, B)$ and $\gamma''_1 = (C)$. The third tree splits γ into the subclades $\gamma'_2 = (A, C)$ and $\gamma''_2 = (B)$. Therefore, $p(\gamma'_1, \gamma''_1|\gamma) = \frac{2}{3}$ and $p(\gamma'_2, \gamma''_2|\gamma) = \frac{1}{3}$

$$C(R_{G',S}) = n_{\mathtt{D}}C_{\mathtt{D}} + n_{\mathtt{L}}C_{\mathtt{L}} + n_{\mathtt{T}}C_{\mathtt{T}}$$

The costs $C_{\mathtt{D}}$, $C_{\mathtt{L}}$, and $C_{\mathtt{T}}$ are arbitrarily set by the user, and reflect prior expectations about the relative event frequencies. For instance, $C_{\mathtt{D}} = 10$, $C_{\mathtt{L}} = 1$ and $C_{\mathtt{T}} = \infty$ implies that HGTs are forbidden, and that gene losses are expected to happen more frequently than gene duplications.

The reconciliation cost $C(G', S)$ of a rooted binary GFT G' and a rooted species tree S is the minimum reconciliation cost over all possible reconciliations between G' and S. Let G be an unrooted multifurcating tree. Parsimony methods aim to find the rooted GFT G^* that agrees with G and that yields the minimum reconciliation cost.

One solution to this problem consists in recursively resolving each polytomy of the input GFT via a dynamic programming approach [111] [27].

3.4.2 Amalgamation methods

ALE [143] is a method that samples GFTs using a joint model of sequence evolution and GFT-species tree evolution. For a given gene family, it takes as input a distribution of GFTs $\mathcal{G}$ estimated from the input gene MSA A, and outputs a distribution of GFTs sampled under this joint model. The input distribution of GFTs $\mathcal{G}$ is typically obtained from tools such as MrBayes [127] or ExaBayes [1], which sample GFTs from an MSA proportionally to the posterior probability distribution under a given model of sequence evolution.

3.4.2.1 PLF approximation

Let G be a rooted tree. The *clade* induced by a node u in G is defined as the set of terminal nodes of G that descend from u. Note that the term *clade* is sometimes

used as a synonym of the term *subtree* in the literature, but here, it corresponds to a leaf set, without any topological information. Let u be an internal node of G and let v and w be its two child nodes. Let γ_u, γ_v, and γ_w be the clades induced by the nodes u, v, and w, respectively. The pair (γ_v, γ_w) is the *clade split* induced by the node u and its children v and w. The conditional probability $q_G(\gamma_u)$ of observing the subtree of G under u is defined as:

$$q_G(\gamma_u) = p(\gamma_v, \gamma_w|\gamma_u) q_G(\gamma_v) q_G(\gamma_w) \tag{3.4}$$

The term $p(\gamma_v, \gamma_w|\gamma_u)$ is the probability of observing the clade split (γ_v, γ_w) conditional of γ_u being observed. It can be estimated from the input GFT distribution $\mathcal{G}$ by computing the ratio between the number of trees in $\mathcal{G}$ that contain the clade split (γ_v, γ_w) and the number of trees in $\mathcal{G}$ that contain the clade γ_u (see Figure 3.7). Let Γ be the top clade induced by the root of G. The phylogenetic likelihood of G, that is, the probability of observing the MSA A given G, can be approximated by recursively applying Equation 3.4, starting from Γ and stopping the recursion at the leaves by setting $q_G(\gamma) = 1$ if γ is a clade with only one taxon:

$$P(A|G) \approx q_G(\Gamma) \tag{3.5}$$

3.4.2.2 GFT sampling

Let A be an MSA, S a rooted species tree, and $\mathcal{G}$ a distribution of GFTs estimated from A under a given model of sequence evolution. Let G be a GFT in $\mathcal{G}$. The reconciliation likelihood (see also Section 2.4.7) is defined as the probability of observing G given S under a given model of GFT-species tree sequence evolution (e.g., the UndatedDTL model introduced in Section 2.4.6). ALE defines the *joint likelihood* of A, S, and G as the product between the phylogenetic likelihood $P(A|G)$ and the reconciliation likelihood $P(G|S)$. The joint likelihood of $\mathcal{G}$ is obtained by multiplying over the GFTs in $\mathcal{G}$:

$$\begin{aligned} \mathcal{L}_{joint}(A, S) &= \sum_{G \in \mathcal{G}} P(A|G) \cdot P(G|S) \\ &\approx \sum_{G \in \mathcal{G}} q_G(\Gamma) P(G|S) \end{aligned} \tag{3.6}$$

In Section 2.4.7, I described an algorithm to evaluate the reconciliation likelihood $P(G|S)$ under the UndatedDTL model, by recursively computing the term $P_{u,e}$ for each node u of the GFT and each node e of the species tree. ALE calculates this joint likelihood by adapting this algorithm to compute the term $P_{\gamma,e}$ for each node e of the species tree and each clade γ induced by at least one of the GFTs of $\mathcal{G}$. Let s_γ be the set of clade splits (γ', γ'') of γ induced by at least one GFT in $\mathcal{G}$. For the

sake of simplicity, I only show how the duplication term of Equation 3.6 is adapted to compute $P(\gamma, e)$:

$$P_{u,e} = ... + p^{\mathtt{D}} \left(2P_{u,e}E_e\right) + p^{\mathtt{D}} P_{v,e}P_{w,e} + ... \tag{3.7}$$

becomes

$$P_{\gamma,e} = ... + p^{\mathtt{D}} \left(2P_{\gamma,e}E_e\right) + p^{\mathtt{D}} \sum_{\gamma',\gamma'' \in s_\gamma} p(\gamma', \gamma''|\gamma) P_{\gamma',e}P_{\gamma'',e} + ... \tag{3.8}$$

ALE samples reconciled GFTs under a joint model of sequence evolution and GFT-species tree evolution by stochastic backtracking along this sum, starting from $P_{\Gamma,R}$, where Γ is the clade containing all gene copies and R is the root of the species tree.

4. Gene family tree inference with ParGenes

This chapter is based on the following peer-reviewed application note:

> **Benoit Morel**, Alexey M. Kozlov and Alexandros Stamatakis. "ParGenes: a tool for massively parallel model selection and phylogenetic tree inference on thousands of genes." *Bioinformatics*, Volume 35, Issue 10, 15 May 2019, Pages 1771–1773. , https://doi.org/10.1093/bioinformatics/bty839

4.1 Introduction

The availability of genomic data for an increasing number of organisms allows to use thousands of gene families to infer evolutionary relationships between species. Species tree inference methods can be divided into supermatrix and Gene Family Tree (GFT) approaches. The former infer the species tree directly from a large concatenated MSA called the *supermatrix* (see Section 3.3.1), whereas the latter infer individual GFTs which are then reconciled into a species phylogeny (see Section 3.3.2). Supermatrix methods are widely used due to their simplicity and availability of efficient implementations [75, 110]. However, GFT methods gain popularity as they can model events such as ILS (e.g., [98]), gene duplication and loss (e.g., [8]), as well as HGT (e.g., [86]).

As input, GFT methods typically require a set of GFTs (potentially also including bootstrap trees) that shall be reconciled (e.g., [18]). Inferring this set of GFTs using ML methods is computationally intensive and requires the use of cluster computing resources.

While popular parallel tools for ML tree inference (e.g., RAxML [138], IQ-TREE [110]) can efficiently process large supermatrices, no dedicated parallel tool exists for inferring per-MSA GFTs on a large set of MSAs. In current studies users deploy ad hoc, and thus potentially error-prone or inefficient, scripts for submitting each individual GFT inference to a cluster as a single job. As common cluster configurations typically limit the number of sequential jobs a single user can execute in parallel, this can substantially increase the time-to-solution.

To this end, we have developed and made available a novel tool called ParGenes. It offers a simple command-line interface that allows to select the best-fit model, infer ML GFTs, and compute bootstrap support values on thousands of per-gene MSAs via a single parallel Message Passing Interface (MPI) run. ParGenes relies on ModelTest-NG [31] and RAxML-NG [76], to perform model selection and tree inference, respectively.

In Section 4.2, we list the features supported by ParGenes. Then, in Section 4.3, we describe our scheduling strategy. In Section 4.4 and Section 4.5, we describe the experimental setup that we used to benchmark ParGenes, and our results, respectively. Finally, we discuss these results in Section 4.6.

4.2 Features

ParGenes encapsulates all per-gene family calculations in one single MPI invocation. To improve load balancing and decrease time-to-solution, ParGenes schedules per-gene family inferences and allocates a *variable* number of cores to these inferences within its MPI runtime environment. In the following, we describe some of the key features.

4.2.1 Simultaneous processing of MSAs

Unlike standard tools for ML inference, ParGenes operates on multiple MSAs. Thus, the user needs to provide a directory containing all MSAs in PHYLIP or FASTA format. One can either specify global or MSA-specific options for both, RAxML-NG and ModelTest-NG. ParGenes initially pre-processes each MSA, to check that the file format is valid, compresses it, saves it in a binary file, and reads its number of taxa and unique site patterns (e.g., the number of non-identical columns in the MSA).

4.2.2 Model selection

ParGenes employs ModelTest-NG, a re-designed, substantially more efficient version of the widely used Modeltest tool [116], to select the best-fit model of evolution for a given MSA. If model testing is enabled in ParGenes, it will first execute ModelTest-NG on each MSA, and then use the best-fit model for the subsequent ML inferences.

4.2.3 ML searches and bootstrapping

ParGenes actively schedules the per-MSA inference jobs that are executed using RAxML-NG [76]. ParGenes allows to run multiple RAxML-NG tree searches per MSA from independent starting trees. This is recommended to better explore the tree search space. Then, it identifies the best-scoring ML tree for each gene. To increase job granularity and thereby improve load balancing, each independent tree search is scheduled separately. ParGenes can optionally conduct a user-specified number of bootstrap inferences. It schedules independent tree inferences of bootstrap replicates (10 bootstrap replicates per job), and subsequently concatenates the resulting trees into one per-MSA bootstrap tree file. Then, it executes RAxML-NG again to map support values to the best-scoring ML tree.

4.2.4 Checkpointing

Since ParGenes performs massively parallel and compute-intensive operations, it also offers a checkpointing feature that allows to resume calculations (e.g., if program execution was interrupted due to typical cluster run-time limitations of 24 or 48 hrs).

ParGenes keeps track of all jobs that have finished so far, and skips them upon restart from a checkpoint. A job typically consists of an individual per-gene ML search, a batch of 10 bootstrap replicate searches, or a ModelTest-NG run.

Furthermore, RAxML-NG and ModelTest-NG also have their own intrinsic checkpointing mechanisms: RAxML-NG writes a checkpoint after each inference step (e.g., model optimization, topological optimization cycle, etc.) of the tree search, and ModelTest-NG after each model it tests. ParGenes uses these checkpointing mechanisms as well, thereby allowing for fine-grained checkpointing.

4.2.5 Estimating the optimal number of cores

Given the dimensions of the input MSAs, ParGenes can calculate an *a priori* estimate of the number of overall cores that will yield 'good' parallel efficiency. This is important, as it is difficult for users to set this value prior to running the analysis.

4.3 Job Scheduling

ParGenes implements a scheduler that simultaneously executes independent jobs with a varying number of cores per job. A job is either a per-MSA RAxML-NG or ModelTest-NG run. We first outline the parallelization scheme, and then the scheduling strategy.

4.3.1 Parallelization scheme

For a typical use case, the input data will contain thousands of independent (per-gene) MSAs with hundreds to a few thousands sites each. While standard tools like RAxML-NG parallelize likelihood computations over MSA sites, ParGenes parallelizes the computations over the MSAs. Note that, the parallel efficiency of

the RAxML-NG parallelization is limited by MSA length (rule-of-thumb: 1,000 MSA sites per core). While most of input MSAs are small, their size distribution exhibits substantial variance with respect to both, the number of taxa, *and* sites (see Figure 4.1). Therefore, inferring trees on large per-gene MSAs on a single core has two drawbacks. First, the MSA size might exceed the available main memory per core. Second, this can decrease parallel efficiency as a long job on a large MSA might take longer to complete than all other jobs (see Figure 4.2(a)). To this end, ParGenes allocates several cores for the largest jobs (i.e., for the largest MSAs) by invoking the multi-threaded RAxML-NG executable (see Figure 4.2(c)). For each MSA, ParGenes first calls RAxML-NG in parsing mode to obtain the recommended number of cores for optimal parallel efficiency via the fine-grained parallelization of the likelihood function in RAxML-NG [139]. The actual number of cores assigned to a job is then rounded down to the next power of two to simplify scheduling. We also assign twice the number of recommended cores to the 5% MSAs with the largest number of taxa (we justify this choice in Section 4.5.1).

4.3.2 Scheduling strategy

ParGenes first sorts all jobs by (i) decreasing number of required cores and (ii) decreasing overall number of characters per MSA. As the number of cores per job is always a power of two (see Section 4.3.1), ParGenes can always keep all cores busy, as long as there are jobs left to process. This works because the MSAs requiring the largest number of cores are scheduled first.

4.4 Experimental Setup

4.4.1 Datasets

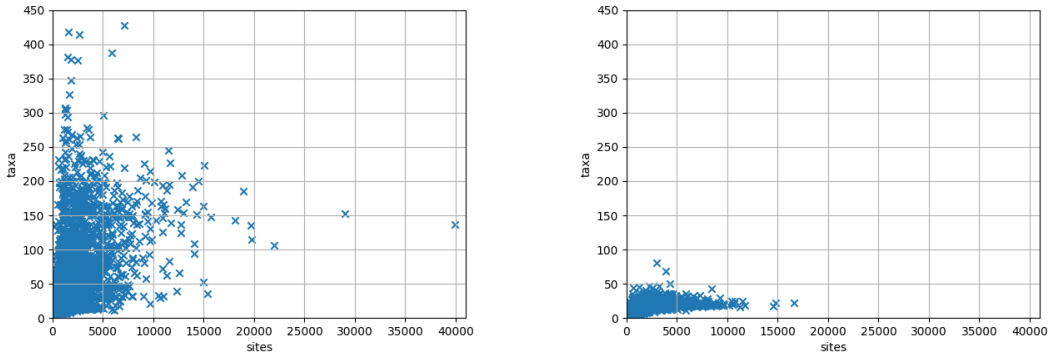
We benchmarked ParGenes using two empirical gene family datasets.

The first one[1] was initially used in [7] and was extracted from the Ensembl database [164]. It contains 8,880 gene families from 15 mammalian species. The second dataset[2] was obtained from the VectorBase database [51] and contains 12,000 gene families of 15 *Anopheles gambiae* species, the primary mosquito vector responsible for the transmission of malaria in large parts of sub-Saharan Africa.

Scheduling independent ML tree searches on these gene family MSAs is challenging, because of their varying dimensions, both, in terms of the number of sites, and number of taxa (see Figure 4.1). Note that, we count MSA lengths in terms of distinct MSA site patterns, as identical site patterns can be, and are compressed by all phylogenetic inference tools in a pre-processing step.

[1] Available at https://github.com/YoannAnselmetti/ADseq-Anopheles-APBC2018/blob/master/data/FASTA/MSA/CDS/MUSCLE.tar.gz

[2] Available at https://sco.h-its.org/exelixis/material/ensembl_8880_15.tar.gz



(a) Ensembl dataset (8880 families) **(b)** Vectorbase dataset (12000 families)

Figure 4.1: MSA dimensions in the Ensembl (top) and VectorBase (bottom) datasets. Each dot represents a per-family MSA, the x coordinate represents the number of unique site patterns, and the y coordinate represents the number of taxa.

4.4.2 Hardware

We executed our benchmarks on our institutional cluster that is equipped with 224 nodes with Intel Haswell CPUs (E5-2630v3) running at 2.40GHz. Each node has 2 CPUs and each CPU has 8 physical cores. The nodes have 64GB RAM and are connected via an Infiniband interconnect.

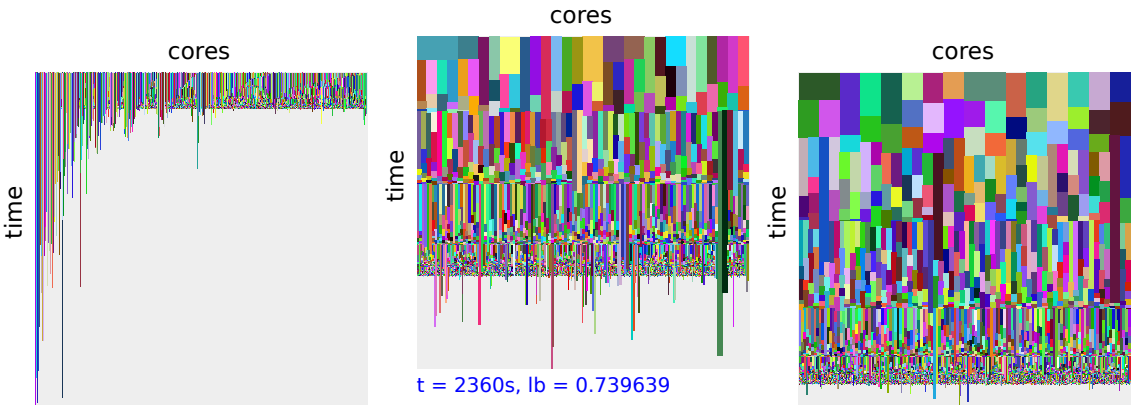
4.4.3 Benchmarks

We executed two distinct benchmarking runs.

- FULL Benchmark: for each gene family (each MSA), execute model testing, then 20 ML tree searches (starting from both, random, and parsimony starting trees) and 100 bootstrap tree inferences, select the best ML tree on the original MSA, and compute the bootstrap support values on that tree.

- FAST Benchmark: run a single ML tree search per gene family (per MSA).

The FULL Benchmark covers the complete feature set of ParGenes and thus represents the realistic default use-case. The load balancing of this benchmark is likely to be 'good', even using a naïve scheduling strategy, as it generates a comparatively large number of independent jobs (e.g., more than one million RAxML-NG runs for the 8800 gene family MSAs from Ensembl). While the FAST benchmark might also corresponds to a realistic use case (e.g., rapid initial data exploration), it generates substantially less inference jobs. This benchmark is thus more relevant for assessing the quality of our load balancing strategy.



(a) One core per job

(b) Variable number of cores per job (recommended by RAxML-NG)

(c) Top 5% heuristic

Figure 4.2: CPU core utilization diagrams for the three distinct scheduling strategies (FAST benchmark, Ensembl dataset, 8880 **gene families,** 512 **cores).** Each colored block represents a single per-gene family MSA job, with the number of cores allocated to the job (x-axis) and the jobs' execution time relative to the overall ParGenes' execution time (y-axis). The grey part depicts idle time. We also show execution time (t) and average CPU core utilization (lb) for each heuristic. Please note that, the time scales differ between the figures.

4.4.4 Parallel performance evaluation

To evaluate the parallel performance of ParGenes, we ran both the FAST and FULL benchmarks on the two empirical datasets with varying number of cores (from 32 up to 1024 cores for the FAST benchmark, and from 256 up to 1024 cores for the FULL benchmark). We then measured the *parallel efficiency* $E(N)$, defined as:

$$E(N) = \frac{N_m \cdot T_{N_m}}{N \cdot T(N)} \tag{4.1}$$

where N is the number of cores, N_m is the minimum number of cores (32 for the FAST benchmark and 256 for the FULL benchmark), and $T(X)$ is the runtime for X cores.

4.5 Results

4.5.1 Impact of Load Balancing Strategy

In this Section, we show that an appropriate scheduling strategy is required to attain 'good' load balancing. To this end, we analyzed our empirical datasets using the FAST benchmark (one ML search per family/MSA, see Section 4.4.3) under distinct scheduling strategies.

Cores	32	64	128	256	512	1024
VB. time	2713s	1327s	662s	339s	190s	143s
VB. efficiency	1.0	1.0	1.0	1.0	0.89	0.60
En. time	33150s	16660s	8342s	3899s	2050s	1330s
En. efficiency	1.0	1.0	0.99	1.06	1.01	0.78

Table 4.1: Benchmark FAST: execution times (time) and parallel efficiencies (efficiency) for both VectorBase (VB.) and Ensenbl (En.) datasets with different numbers of cores.

In a first naïve ParGenes implementation, we initially sorted the jobs by descending order of expected execution time. Then, we dynamically assigned each job to the available cores. Rather unsurprisingly, we observed that, some per-MSA inference jobs require substantially longer execution times than the average job, and that they continue running when most of the other jobs have completed. This resulted in a high proportion of inactive cores and, as a consequence, in poor load balancing (see Figure 4.2(a)).

To overcome this issue, we changed our strategy to assigning one or more cores (depending on the MSA size) to each job. However, assigning too many cores to a job reduces its parallel efficiency, and might thus induce a slowdown. We therefore determined the optimal trade-off between the parallel efficiency of an individual job and the overall load balancing of ParGenes. To this end, we empirically defined appropriate criteria (MSA width, substitution model used, sequence data type, i.e., DNA or protein data) to determine the optimal per-job core number. The corresponding function is implemented in RAxML-NG. As mentioned before, the jobs that require the highest number of cores are executed first. As shown in Figure 4.2(b), this heuristic improves the load balancing and reduces the overall execution time by almost a factor of 5 in the specific experiment.

However, we still observe some 'tails', that is, jobs that are still running when most others are done, even when the number of cores assigned to an individual MSA inference job is high. We observed that these tail jobs always correspond to per-MSA inferences on a large number of taxa. To shorten the tails, we assign twice the number of cores than recommended by RAxML-NG, to the top 5% of the per-MSA jobs with the largest number of taxa (top 5% heuristic). This is done, to (i) reduce their execution times and (ii) to also start executing them earlier (remember that jobs with more cores are executed first). On the one hand, these jobs will typically exhibit sub-optimal parallel efficiency. on the other hand, they only account for a small portion of the overall workload and the performance impact of tails is alleviated. As shown in Figure 4.2(c), the top 5% heuristic shortens the tails and decreases overall runtime (15% improvement in the specific experiment).

Cores	256	512	1024
Execution time	33000s	18000s	10800s
Efficiency	1.0	0.91	0.76

Table 4.2: Execution times and parallel efficiency for the FULL benchmark applied to the VectorBase dataset, with different number of cores.

4.5.2 Experimental results for FAST Benchmark

We executed the FAST benchmark (one ML search per gene family, see Section 4.4.3) on the VectorBase and the Ensembl datasets, using 32 up to 1024 cores. Our results (Table 4.1) show that, the ParGenes parallelization and scheduling strategy scale well up to 512 cores for both datasets. Using 512 cores ParGenes inferred the $8,880$ and $12,000$ ML trees in 35 minutes and 3 minutes, respectively, while attaining high parallel efficiency.

4.5.3 Experimental results for FULL Benchmark

In this section, we describe the results of the FULL benchmark (model testing, ML search with multiple distinct starting trees, and bootstrap replicates).

ParGenes executed the entire analysis on the Ensembl dataset in 25 hours using 1024 cores. It achieved an overall parallel efficiency above 99%, which indicates that this analysis is likely to scale with substantially more cores. Most of the time (more than 24 hours) is spent for computing the best ML trees and the bootstraps replicates. ModelTest-NG ran in 25 minutes. All the other intermediate steps took in total five minutes. We successfully restarted the computations after ParGenes stopped because of the 24-hours wall time limit of our cluster, using our checkpoint mechanism.

The VectorBase gene families contain much fewer genes per family than the Ensembl dataset (see Figure 4.1), and this dataset is thus faster to process. We executed the FULL benchmark on this dataset with different number of cores to show that it scales well (Table 4.2).

4.6 Conclusion

We have presented an efficient parallel tool for comprehensive phylogenetic inference of GFTs on thousands of MSAs via a *single* MPI invocation. Apart from being flexible with respect to the inference options, ParGenes also yields 'good' parallel efficiency via appropriate scheduling mechanisms. We expect that ParGenes will contribute to increasing throughput times and productivity in GFT/species-tree reconciliation studies.

5. Rooted species tree inference with SpeciesRax

This chapter is based on the following open-access publication:

> **Benoit Morel**, Paul Schade, Sarah Lutteropp, Tom A. Williams, Gergely J. Szöllősi, and Alexandros Stamatakis. "SpeciesRax: A tool for maximum likelihood species tree inference from gene family trees under duplication, transfer, and loss." *bioRxiv*, 2021, https://doi.org/10.1101/2021.03.29.437460

5.1 Introduction

Phylogenetic species tree inference constitutes a challenging computational problem. Nonetheless, accurate and efficient tools for species tree inference exhibit a substantial potential for obtaining novel biological insights.

The concatenation or supermatrix approach (see Section 3.3.1) has long been the gold standard for species tree inference. Here, per-gene sequences are first aligned into per-gene MSAs and subsequently concatenated into a single, large supermatrix. Then, statistical tree inference methods (ML [76, 97] or Bayesian inference [1, 127]) are applied to infer a tree on these supermatrices. The concatenation approach heavily relies on accurate orthology inference, which still constitutes a challenging problem [6]. In addition, concatenation methods were shown to be statically inconsistent under the Multi-Species Coalescent Model (MSCM) because of potential ILS [77, 94].

As GFT methods (see Section 3.3.2) can alleviate some of the pitfalls of the supermatrix approach, they are becoming increasingly popular. GFT methods can take into account that the evolutionary histories of the GFTs and the species tree

are discordant (see Section 2.5.1) due to biological phenomena such as ILS, gene duplication, gene loss, and HGT.

At present, the most commonly used GFT tools [16, 165] only model ILS and are limited to single-copy gene families. These methods also heavily rely on accurate orthology inference and discard large amounts of potentially informative data. Approaches that can handle multiple-copy gene families exist, but have not been widely adopted yet [18, 101, 151, 166]. Here, we focus on describing, evaluating, and making available a novel method for inferring reliable species trees from multiple-copy gene families in the presence of both paralogy and HGT. For instance, HGT is particularly challenging when analysing microbial data, because supermatrix analyses can be misled in unpredictable ways if HGTs are included in the concatenated supermatrix [36, 156].

One class of existing methods to infer species trees from multiple-copy gene families attempts to simultaneously estimate the GFTs and the species tree (see Section 3.3.2.3 and [18, 32]). However, these methods are computationally demanding and are limited to small datasets comprising less than 100 species.

Another class of existing methods handles the GFT inference and the species tree inference steps separately. As input they require a set of given, fixed GFTs and do not attempt to correct the GFTs during the species tree inference step. DupTree [151] and DynaDUP [12] (see Section 3.3.2.4) search for the species tree with the least parsimonious reconciliation cost, measured as the number of duplication events in DupTree, and the sum of duplication and loss events in DynaDUP. STAG [40] infers a species tree by applying a distance method to each gene family that covers *all* species, and subsequently builds a consensus tree from all these distance-based trees. However, STAG ignores a substantial fraction of signal by discarding gene families that do not cover all species. FastMulRFS [101] (see Section 3.3.2.5) extends the definition of the RF distance to multiple-copy GFTs and strives to minimize this distance between the species tree and all input GFTs. More recently, with Astral-Pro [166] (see Section 3.3.2.2) a promising improvement of Astral was released that can handle multiple-copy GFTs: Astral-Pro uses dynamic programming to infer the species tree that maximizes a novel measure of quartet similarity that accounts for orthology as well as paralogy. All of the above methods are non-parametric and do not deploy a probabilistic model of evolution. In addition, none of them explicitly models HGT.

Here, we present SpeciesRax, the first ML method for inferring a rooted species tree from a set of GFTs in the presence of gene duplication, gene loss, and HGT. SpeciesRax takes as input a set of MSAs and/or a set of GFTs. If MSAs are provided, SpeciesRax will infer one ML GFT tree per gene family using RAxML-NG [76]. Thereafter, SpeciesRax first generates an initial, reasonable (i.e., non-random) species tree by applying MiniNJ (which we also introduce in this chapter), our novel *distance* based method for species tree inference from GFTs in the presence of paralogy. MiniNJ shows similar accuracy as other non-parametric methods while being at least two orders of magnitude faster on large datasets. Finally, SpeciesRax executes a ML tree search heuristic under an explicit statistical gene loss, gene

duplication, and HGT model starting from the MiniNJ species tree. When the species tree search terminates, SpeciesRax calculates approximate branch lengths in units of mean expected substitutions per site. Furthermore, it quantifies the reconstruction uncertainty by computing novel quartet-based branch support scores on the species tree. We show that SpeciesRax is fast and at least as accurate as the best competing species tree inference tools. In particular, SpeciesRax is twice as accurate (in terms of relative RF distance to the true species trees) than all other tested methods on simulations with large numbers of paralogous genes.

5.2 Method

The *UndatedDTL model* (introduced in Section 2.4.6) describes the evolution of a GFT along a species tree through gene duplication, gene loss, speciation, and HGT events. The *reconciliation likelihood* (introduced in Section 2.4.7) is the probability of observing a set of GFTs $\mathcal{G} = (G_1, ..., G_n)$ given a rooted species tree S and the set Θ of duplication, loss, and HGT intensities:

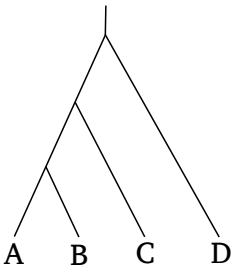
$$L(S, \Theta|\mathcal{G}) = \prod_{k=1}^{n} P(G_k|S, \Theta) \tag{5.1}$$

As already mentioned, SpeciesRax takes a set of unrooted GFTs as input. It starts its computations from an initial species tree that can either be randomly generated, user-specified, or inferred using our new distance method MiniNJ (Section 5.2.1). Then, it performs a tree search (Section 5.2.2) for the rooted species tree S and the model parameters Θ that maximize the reconciliation likelihood (Section 5.2.3) $L(S, \Theta|\mathcal{G})$. At the end of the search, it also calculates support values (Section 5.2.4) and branch lengths in units of expected mean number of substitutions per site (Section 5.2.5) for each branch of the inferred species tree from the GFTs. Furthermore, SpeciesRax adapts the likelihood score to better account for missing data and inaccurate assignment of sequences to gene family clusters (Section 5.2.6). All the above steps are parallelized using MPI (Section 5.2.7).

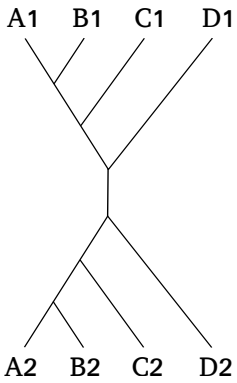
5.2.1 Computing a reasonable initial species tree with MiniNJ

Here, we introduce MiniNJ (Minimum internode distance Neighbor Joining), our novel distance based method for inferring an unrooted species tree in the presence of paralogy. MiniNJ is fast, that is, it is well-suited for generating an initial species tree for the subsequent maximum likelihood optimization. MiniNJ is inspired by NJst [87], a distance based method that performs well in the *absence* of paralogy. Initially, we briefly recall the NJst algorithm already introduced in Section 3.3.2.1, and subsequently describe our modifications.

NJst initially computes a distance matrix from the unrooted GFTs and then applies NJ to reconstruct the species tree. NJst defines the gene internode distance D_g such that $D_g(x, y)$ is the number of internal nodes between the terminal nodes x and y in



(a) Species tree



(b) Gene tree

	A	B	C	D
A	0	3.5	3.5	3.5
B		0	3.5	3.5
C			0	2.5
D				0

(c) NJstmatrix

	A	B	C	D
A	0	1	2	3
B		0	2	3
C			0	2
D				0

(d) MiniNJ matrix

Figure 5.1: An example where MiniNJ computes distances that better reflect the true species tree than NJst. (a) The true rooted species tree. (b) A GFT resulting from a gene duplication at the root of the species tree. (c) The distance matrix D_{NJst} computed with NJst, incorrectly suggesting that all species are equidistant, except for C and D. This is the result of distance overestimation due to paralogous genes: for instance, species A and B are neighbors in the species tree, but the gene copies $A2$ and $B1$ are very distant from each other in the gene tree, because they start diverging at an early duplication event (paralogous genes). (d) Distance matrix D_{MiniNJ} computed with MiniNJ. The gene internode distances correctly reflect the species distances, because MiniNJ successfully pruned pairs of paralogous genes, such as $A2$ and $B1$, and only takes into account orthologous genes, such as $A1$ and $B1$.

a GFT. NJst computes the distance between two species as the average over the internode distances between all pairs of gene copies mapped to those two species.

More formally, let a and b be two species. Let K be the number of GFTs. Let m_{ak} be the terminal nodes from the GFT k mapped to species a. Let x_{iak} be the ith terminal node from the GFT k mapped to species a. NJst defines the distance matrix D_{NJst} as follows:

$$D_{\text{NJst}}(a,b) = \frac{\sum_{k=1}^{K} \sum_{i=1}^{m_{ak}} \sum_{j=1}^{m_{bk}} D_g(x_{iak}, x_{jbk})}{\sum_{k=1}^{K} m_{ak} m_{bk}} \tag{5.2}$$

NJst has two drawbacks. First, it accounts for all pairs of gene copies, including paralogous gene copies that do not contain information about speciation events (see Figure 5.1). Secondly, it assigns very high (quadratic) weights to gene families comprising a high number of gene copies: for instance, a gene family k_1 with 5 gene copies in both species a and b will contribute 25 times to the distance between a and b, while a single-copy family k_2 will only contribute once. For instance, $\sum_{i=1}^{m_{ak}} \sum_{j=1}^{m_{bk}} D_g(x_{iak}, x_{jbk})$ is the sum over 25 gene internode distances for family k_1 and of only one gene internode distance for family k_2. Since the normalization by the number of gene internode distances is conducted after summing over all these quantities (with the denominator in Equation 5.2), the contributions of families k_1 and k_2 are unbalanced.

MiniNJ adapts Equation 5.2 to address these two issues. It attempts to discard pairs of paralogous gene copies by only considering the two closest GFT terminal nodes mapped to a pair of species for each family, according to the internode distance: let δ_{abk} be equal to 1 if gene family k contains at least one gene copy mapped to a and one gene copy mapped to b, and 0 otherwise. We define D_{MiniNJ}:

$$D_{MiniNJ}(a,b) = \frac{\sum_{k=1}^{K} \min_{i=1}^{m_{ak}} \min_{j=1}^{m_{bk}} D_g(x_{iak}, x_{jbk})}{\sum_{k=1}^{K} \delta_{abk}}$$

Note that for any two species a and b, all gene families that cover a and b contribute equally to $D_{MiniNJ}(a,b)$.

MiniNJ then infers an unrooted species tree from this distance matrix using the NJ algorithm (see Section 3.1). The distance matrix computation has time complexity $O(\sum_{k=1}^{K} |g_k|^2)$ where $|g_k|$ is the number of gene sequences in the family k. The NJ algorithm has time complexity $O(|S|^3)$ where $|S|$ is the number of species. The overall time complexity of MiniNJ is thus $O(|S|^3 + \sum_{k=1}^{K} |g_k|^2)$.

5.2.2 Tree search heuristic

Given a set $\mathcal{G}$ of unrooted GFTs, SpeciesRax implements a hill-climbing algorithm to search for the rooted species tree S and optimize the set of Duplication, Transfer, and Loss (DTL) intensities Θ that maximize the reconciliation likelihood $L(S, \Theta|\mathcal{G})$.

The search algorithm consists of four steps: ML species tree root inference, DTL intensities optimization, local SPR species tree search, and transfer-guided SPR species tree search. In this section, we first describe each of these four steps in detail, and subsequently describe in which order we apply them.

5.2.2.1 Maximum likelihood species tree root inference

To infer the ML root of a given species tree, SpeciesRax roots the species tree at several candidate positions, evaluates the reconciliation likelihood of each putative root position, and keeps the best one. In the *exhaustive* root search mode, SpeciesRax evaluates all possible putative root positions. In the *local* root search mode, SpeciesRax only explores putative root positions around a given radius of the current root (typically, all branches that are less than three nodes away from the current root).

5.2.2.2 DTL intensities optimization

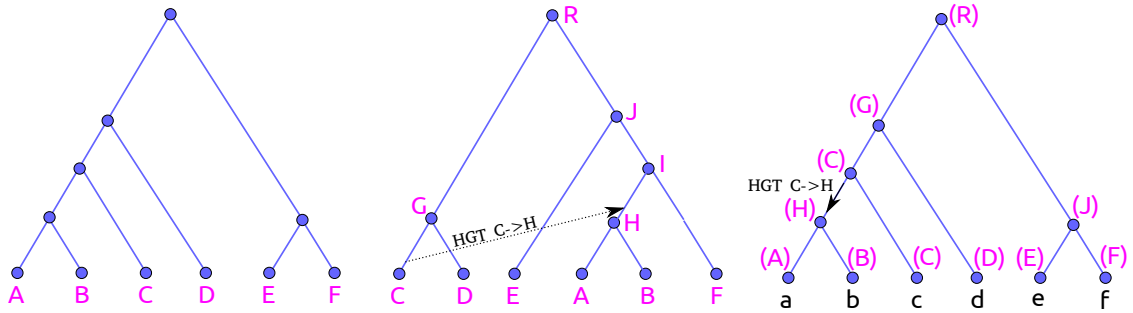
SpeciesRax optimizes the DTL intensities via a gradient descent method. SpeciesRax deploys two modes. In the *global* DTL intensities mode, all families share the same three (duplication, loss, and HGT) intensities. In the *per-family* DTL intensities mode, each gene family has its own set of DTL intensities to account for DTL rate heterogeneity among GFTs. In our experiments we observed that the choice of the mode does not significantly affect runtime and that the per-family DTL intensities mode yields slightly more accurate species trees.

5.2.2.3 Local SPR search

In the *local SPR search*, we explore all possible SPR moves (see Section 3.2.2) for a given subtree rearrangement radius (i.e., the number of nodes away from the subtree pruning position at which we attempt to re-insert the subtree again). The default rearrangement radius is set to 1. In our experiments, we observed that higher values did not improve the reconstruction accuracy. We directly keep the trees generated by SPR moves that improve the reconciliation likelihood. We stop this procedure when no better tree can be found.

5.2.2.4 Transfer-guided SPR search

In the *transfer-guided SPR search*, we assume that the most promising SPR moves with respect to improving the reconciliation likelihood are those SPR moves that reduce the number of HGT events that are necessary to reconcile the GFTs with the species tree (see Figure 5.2). A similar strategy has been previously applied in [142]. To this end, we infer the ML reconciliation between the GFT and the current species tree, and count the number of HGTs between each pair of nodes in the species tree.



(a) True species tree **(b)** Current species tree **(c)** True GFT

Figure 5.2: Illustration of the transfer-guided SPR search. (a) The true species tree topology. (b) The current species tree topology. Labels are assigned to the internal nodes. The subtree under node H is incorrectly placed next to species F. (c) The true GFT in the case where no DTL event occured. This GFT perfectly matches the true species tree topology. The GFT is reconciled with the *current* species tree, that is, each node of the GFT is assigned to a node in the current species tree (represented between parentheses). In this example, the ML reconciliation involves a HGT from species C to the ancestral species H. Note that this event does not correspond to a real HGT event, because the true scenario does not involve any HGT event. Instead, this HGT event was inferred because the current species tree is incorrect. A promising candidate species tree can be obtained by pruning H and regrafting it next to C, in order to (potentially) reduce the number of inferred HGT events. In this specific case, this candidate species tree is also the true species tree.

We then try the SPR moves between the pair of species (we regraft the node of the species tree corresponding to the receiving species lineage next to the node of the species tree corresponding to the source species lineage) that yields the highest numbers of HGTs, and apply those SPR moves that improve the reconciliation likelihood if any. We stop these attempts after n unsuccessful consecutive trials, where n is the number of species. After $k = max(15, n/4)$ non-consecutive successful trials, we re-infer the HGTs on the current species tree. We empirically determined 'good' values for n and k.

5.2.2.5 Species tree search overview

We now explain how we combined the previously described moves to infer a rooted ML species tree.

If the starting species tree was provided by the user or generated with MiniNJ, we optimize the DTL intensities and execute a local root search. If the starting species tree was randomly generated, we start using a default set of DTL intensities ($\delta := \tau := \lambda := 0.2$). Then, we apply the transfer-guided and local SPR searches in an alternating manner. After each (transfer-guided or local) SPR search, we conduct a local species tree root search and optimize the DTL rates. When no better species tree can be found, we run a final local species tree root search with an increased radius (5 by default instead of 3 for the previous ones) and stop the search. The aforementioned *exhaustive* species tree root search is optional and is not executed by default.

5.2.3 Reconciliation likelihood evaluation

Calculating the reconciliation likelihood under the UndatedDTL model represents the major computational bottleneck. To reduce its computational cost, we introduce several approximations that we describe in this subsection. The description of the exact reconciliation likelihood computation is provided in Section 2.4.7.

5.2.3.1 The HGT-Loss approximation

We first observe that Equation 2.15 is not an analytic formula but a system of equations, because the term for computing $P_{e,u}$ depends on itself and on $P_{h,u}$ for all nodes h in the species tree. This is due to the duplication-loss term $p^D\,(2P_{e,u}E_e)$ and to the HGT-loss term $p^T\left(\bar{P}_u^T E_e + \bar{E}^T P_{e,u}\right)$. While the first term can be computed analytically, we need to deploy numerical optimization routines to evaluate the second term.

In our HGT-loss approximation approach, we simply discard this second term $p^T \bar{E}^T P_{e,u}$ from the initial likelihood formula. Thus, we do not to account for scenarios involving a gene u being transferred from a species e to a species f and going extinct after e.

We ran the experiments for the current chapter with and without this approximation. We did not observe any difference in species tree reconstruction accuracy for SpeciesRax. However, the reconciliation likelihood evaluation runs three times faster than the exact evaluation.

5.2.3.2 Rooting the GFTs

The input GFTs are unrooted. Yet computing the reconciliation likelihood on a rooted GFT is substantially faster than for an unrooted GFT. First, this is because for unrooted GFTs, we need to iterate over all possible GFT root positions, and therefore have to compute every $P_{e,u}$ term three times (once for each possible orientation of the outgoing branches of the GFT node u toward a potential root) instead of once for a single GFT root. Second, in the following subsection we introduce the *double-HGT approximation* to accelerate the computation of $P_{e,u}$ for internal GFT nodes u that are far from the root (in terms of internode distance). Thus, iterating over all possible roots would substantially reduce the overall speedup that can be obtained via the double-HGT approximation.

For the above reasons, we only compute the likelihood for the ML root position of each GFT. To infer this ML root, we perform a local GFT root search for each new species tree candidate as follows: Let us assume that we know the best root position of a GFT G for a given species tree S. When evaluating the likelihood of a new species tree S' we a different tree topology, we evaluate its value for the previously best GFT root position and for placing the root into the neighboring branches. Note that the additional cost for exploring the neighboring putative root positions is negligible for large GFTs, as the majority of the recursive intermediate computations are redundant for all five root positions. If one of the four putative neighboring root positions yields a better likelihood, we set it as the new best root. Then we repeat the above procedure on the four neighboring branches of this new root until no better root position is found. We omit the computations for the root position that we have already tested. Note that this local root search is not guaranteed to find the globally optimal ML root of the GFT.

Let $\mathcal{G}$ be a set of GFTs, let S be the current species tree and let S' be a new candidate species tree. Let $L(S, \mathcal{G})$ be the likelihood of S. By L we denote the likelihood obtained for the globally optimal ML GFT root positions and by $\hat{L}$ we denote the likelihood of the best respective roots obtained via the local root search procedure described above. Obviously, $\hat{L}(S', \mathcal{G}) \leq L(S', \mathcal{G})$ because the local root search might not find the globally optimal ML roots on all GFTs. To approximately correct for this underestimation when comparing S and S', we test:

$$\hat{L}(S', \mathcal{G}) + \epsilon \geq L(S, \mathcal{G}) \tag{5.3}$$

where ϵ is twice the average underestimation for all previously accepted species trees Φ:

$$\epsilon = 2\frac{\sum_{S \in \Phi} L(S,\mathcal{G}) - \hat{L}(S,\mathcal{G})}{|\Phi|} \tag{5.4}$$

When the test in Equation 5.3 is positive, we exactly evaluate $L(S', \mathcal{G})$ via an exhaustive GFT root search for each gene family. We then accept S' as the new current species tree if:

$$L(S', \mathcal{G}) > L(S, \mathcal{G}) \tag{5.5}$$

To initialize ϵ, we skip the test in Equation 5.3 for the 20 first candidate species trees. We determined this value (20) empirically via computational experiments on simulated and empirical datasets.

5.2.3.3 The double-HGT approximation

Let S be a rooted species tree and let G be a rooted GFT. Computing the term $P(G|S)$ has time complexity $O(|S||G|)$ because it consists in evaluating $P_{e,u}$ for all species nodes e of S and all gene nodes u of G. We observe that under the UndatedDTL model, the probability $\frac{P_T}{|S|}$ of an HGT event between two species is typically substantially smaller than the probabilities of other events (P_S, P_L, and P_D). Hence, unlikely HGT events substantially decrease the reconciliation likelihood scores.

Further, we observe the following: an ancestral GFT node u is very unlikely to be observed in a species branch e that is not the ancestor of at least one terminal species in which at least one GFT terminal node descending from u is observed. This is because such a scenario would require at least two HGT events in order to be explained (one on each lineage of u) and would therefore penalize the reconciliation likelihood to a larger extent than an alternative scenario with one single HGT event prior to u.

Let $L(u)$ be the set of GFT terminal nodes that descend from u. Let $L_S(u)$ be the set of species that are mapped to the elements of $L(u)$. Let X_u be the lowest common ancestor of $L_S(u)$ in S. In our approximation, we only compute $P_{e,u}$ if e is either an ancestor or a descendant of X_u, and set $P_{e,u} := 0$ otherwise.

We do not attempt to formally estimate the speedup obtained via this approximation because it depends on the *G and S* tree topologies. We remark that the GFT nodes u where X_u is close to the leaves of the species tree will require few computations and that we can expect a large fraction of GFT nodes to be located close to the leaves as, in practice, most nodes in a binary rooted tree are closer to the leaves than to the root.

5.2.4 Support value estimation

Here, we describe how SpeciesRax calculates branch support values on the species tree from a set of unrooted GFTs $\mathcal{G}$. We first revisit the definition of a Speciation-driven Quartet (SQ). Then, we explain how we use the SQ frequency to estimate branch support values. Finally, we describe two alternative SQ-based scores, namely the Quadripartition Internode Certainty (QPIC) and the Extended Quadripartition Internode Certainty (EQPIC) scores.

We first briefly revisit the definition of a SQ [166]. Let $\hat{\mathcal{G}}$ be a set of rooted GFTs with internal nodes either tagged by "duplication" or "speciation" events as estimated from $\mathcal{G}$. A quartet from $\hat{G} \in \hat{\mathcal{G}}$ only contains information about the speciation events, if it includes four distinct species *and* if the LCA of any three out of the four taxa of this quartet is a speciation node. Such a quartet is called SQ. We refer to [166] for a more formal definition of the SQ count and for its computation from a set of unrooted and unlabelled GFTs.

We now introduce several notations in order to define the *SQ frequency* of a pair of internal nodes in the species tree. Let S be an unrooted species tree. Let (u, v) be a pair of distinct internal nodes in S. The nodes u and v define a *metaquartet* $M_{u,v} = (A, B, C, D)$, where A and B (resp. C and D) are the leaf sets under the left and right children of u (resp. v) with S rooted at v (resp. u). Let $z = (z_1, z_2, z_3)$ such that z_1 (resp. z_2 and z_3) is the *SQ count* in $\mathcal{G}$ corresponding to the metaquartet topology $AB|CD$ (resp. $AC|BD$ and $AD|BC$). Note that z_1 corresponds to the metaquartet topology that agrees with S ($(A, B|C, D)$) and that z_2 and z_3 correspond to the two possible alternative distinct metaquartet topologies ($AC|BD$ and $AD|BC$). Let $\hat{z} = (\hat{z}_1, \hat{z}_2, \hat{z}_3)$ such that $\hat{z}_i = \frac{z_i}{z_1+z_2+z_3}$ for $i \in (1, 2, 3)$. We define the *SQ frequency* of (u, v) in S given $\mathcal{G}$ as $SQF_{\mathcal{G}}(u, v) = \hat{z}_1$.

The SQ frequency represents how many SQs around u and v support the species tree topology. However, it does not always reflect if $(AB|CD)$ is the best supported of the three possible metaquartet topologies, in particular when $\frac{1}{3} < z_1 < \frac{1}{2}$. For instance, $\hat{z} = (0.4, 0.3, 0.3)$ suggests that $(AB|CD)$ is the correct topology, but $\hat{z} = (0.4, 0.6, 0.0)$ suggests that the alternative topology $(AC|BD)$ is better supported. Thus, the value of z_1 alone is not sufficiently informative to assess our confidence in a branch defined by u and v.

To overcome this limitation, we therefore also compute the QPIC and EQPIC scores introduced in [169]. Note that these scores were initially defined for single-copy gene families. Since SpeciesRax operates on multiple-copy families, we adapt the scores by only counting SQs instead of counting *all* quartets. Let (u, v) be two distinct nodes of S.

$$qpic'(u, v) = 1 + \hat{z}_1 log(\hat{z}_1) + \hat{z}_2 log(\hat{z}_2) + \hat{z}_3 log(\hat{z}_3) \tag{5.6}$$

$$QPIC(u, v) = \begin{cases} 0 & \text{if } z_1 = z_2 = z_3 = 0 \\ qpic'(u, v) & \text{if } z_1 = max(z_1, z_2, z_3) \\ -qpic'(u, v) & \text{otherwise} \end{cases} \tag{5.7}$$

In particular, if u and v are neighbors, we define the QPIC of the branch e between u and v as $QPIC(e) = QPIC(u, v)$. One limitation of the QPIC score is that it discards all SQs defined by nodes u and v that are not neighbors. The QPIC score is extends in [169] by defining the EQPIC score of a branch e:

$$EQPIC(e) = min_{\{u,v\} \in \mathcal{N}(e)}(QPIC(u, v)) \tag{5.8}$$

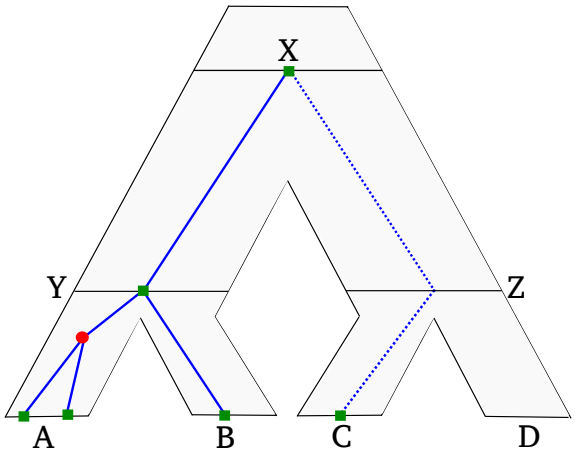


Figure 5.3: Illustration of relevant paths for the species tree branch length estimation. A GFT (in blue) is represented within a species tree (tree with grey background). Relevant paths are represented by solid lines, and non-relevant paths by dashed lines. The GFT has one Y-relevant path, one B-relevant path and two A-relevant paths (due to the duplication event along A). The gene branch that leads from species X to species C is not part of a relevant path because C is not a direct child of X: this branch goes through an unobservable speciation (the gene goes extinct in the branch of species D) and thus the time of the speciation at Z is unknown.

where $N(e)$ is the set of node pairs $\{u, v\}$ such that the branch e belongs to the unique path between u and v.

We remark that both QPIC and EQPIC scores range between -1 and 1. They take positive values when they support the relevant metaquartet topologies of the species tree S and negative values otherwise.

5.2.5 Branch length estimation

SpeciesRax infers the branch lengths of the rooted species tree in units of expected mean number of substitutions per site. We assume that the GFTs have been reconciled with the species tree and that their branch lengths are given in units of expected mean number of substitutions per site. Our method independently estimates the length of each species branch by averaging over the branch lengths between relevant speciation events in the reconciled GFTs.

Remember that both, the species tree, and the reconciled GFTs are rooted. For any rooted tree T, let $N(T)$ be the set of nodes in T. For any $x \in N(T)$, let $t(x)$ be the branch length of x. Let S be a species tree, let $\mathcal{G}$ be the set of GFTs, and let $G \in \mathcal{G}$. Let (s, σ) be the reconciliation function that maps each gene node $x \in G$ to a species node $s(x) \in N(S)$ and to an event label $\sigma(x) \in \{E_S, E_D, E_T\}$ (speciation, duplication, and HGT). We define a *path* p in G as a sequence of nodes in $N(G)$ such that each element in p is the child of its predecessor. We define the length $t(p)$ of a path p as the sum of the lengths of its elements.

We now introduce the concept of *f-relevant paths* to characterize gene paths that contain relevant information for estimating the branch length above a species node $f \in N(S)$. For a given $f \in N(S)$ and its parent node e, an *f-relevant path* $p = (x_1, x_2, ..., x_{|p|})$ in G is a path such that $\sigma(x_1) = \sigma(x_{|p|}) = E_S$, $s(x_1) = e$ and $s(x_{|p|} = f)$ (See example in Figure 5.3). By $\mathcal{P}_f(g)$ we denote the set of all f-relevant paths in G.

For each species node f, we compute its length as the weighted average over all f-relevant paths:

$$\hat{t}(f) = \frac{\sum_{G \in \mathcal{G}} \sum_{p \in \mathcal{P}_f(G)} w(G)t(p)}{\sum_{G \in \mathcal{G}} \sum_{p \in \mathcal{P}_f(G)} w(G)}$$

where $w(G)$ is a weight function associated to each GFT. If the MSAs are available, we set $w(G) = l_G r_G$, where l_G is the length of the MSA associated with G, and r_G is the proportion of characters that are neither undetermined nor gaps. If the MSAs are not available, we set $w(G) = 1$.

5.2.6 Accounting for missing data

We refer to *missing data* as gene copies that are absent from a gene family to which they should belong. This can occur, for instance, when some gene sequences have not been sampled or when the gene family clustering is inaccurate. Missing data is problematic for species tree estimation, in particular when the missing data pattern distribution is non-random [159]. In particular, reconciliation methods like SpeciesRax can be affected by missing gene copies: for instance, if sequences for a subset of the species under study have not been sampled for several families, the statistical reconciliation model will attempt to explain these missing gene copies via additional, yet incorrect extinction events. Thus, a candidate species tree that groups such a subset of species into one subtree will typically exhibit a better reconciliation likelihood score than the "true" species tree. This is the case, because only one loss event per family would be necessary to explain all missing gene copies. We alleviate this problem to a certain extent by deploying a *species tree pruning mode*: let G be a GFT and S a species tree. We replace the reconciliation likelihood term $L(S, G)$ by $L(S', G)$, where S' is obtained from S by pruning all species that are not covered by G *and* by removing internal nodes of degree 1 until the tree is bifurcating. Thus, if a species is not present in a family, the reconciliation likelihood of this family does not depend on the position of this species in the species tree.

A downside of this approach is that it can disregard some true gene loss events. On both, empirical, and simulated datasets, we observed that this does not seem to negatively affect the reconstruction accuracy though.

5.2.7 Parallelization

We parallelized SpeciesRax with MPI which allows to execute it using several compute nodes with distributed memory (e.g., compute clusters). We distribute the gene families among the available cores to parallelize the reconciliation likelihood computation.

Method	Type	Infers root	Ref.
NJst	distance matrix	No	[87]
DupTree	parsimony	Yes	[151]
FastMulRFS	distance to GFTs	No	[101]
Astral-Pro	quartet	No	[166]
MiniNJ	distance matrix	No	(this paper)
SpeciesRax	maximum likelihood	Yes	(this paper)

Table 5.1: Software used in our benchmark.

5.3 Experiments

5.3.1 Tested tools

In the following we describe the settings we used for executing all tools (summarized in Table 5.1) in our experiments. We ran DupTree, FastMulRFS, and MiniNJ with default parameters. Among the four outputs that FastMulRFS provides, we discarded the outputs that may contain multifurcating trees ("majority" and "strict"). Among the two remaining outputs ("greedy" and "single"), we selected "single" because it performed slightly better in our experiments.

We used our own (re-)implementation of NJst because the existing implementation written in `R` was too slow for completing our tests in a reasonable time.

We executed Astral-Pro using all available memory ("-Xms700G -Xmx700G") and a fixed random number seed ("– seed 692").

We executed SpeciesRax starting from a MiniNJ tree, with the UndatedDTL model, with per-family DTL rates. We also disabled all irrelevant steps such as gene tree optimization ("-s MiniNJ –optimize-species-tree –do-not-optimize-gene-trees –rec-model UndatedDTL –per-family-rates –skip-family-filtering –do-not-reconcile"). For the experiments on empirical datasets, we added the SpeciesRax option "–prune-species-tree" described in Section 5.2.6 to account for missing data. To analyze the empirical dataset that does not contain any multiple-copy gene families (Archaea364), we disabled the gene duplication events in the UndatedDTL model (option "–no-dup").

5.3.2 Hardware environment

We executed all experiments on the same machine with 40 physical cores, 80 virtual cores and 750GB RAM. Note that DupTree, FastMulRFS, NJst, and MiniNJ only offer a sequential implementation. In contrast, SpeciesRax and Astral-Pro provide a parallel implementation and were run using all available cores. We discuss the implications of this choice in the results section.

5.3.3 Simulated datasets

We generated simulated datasets with SimPhy [92] to assess the influence of the simulation parameters on the reconstruction accuracy of the methods.

Parameter name	Parameter value
Standard parameters	
Replicates number	100
Speciation rate	5×10^{-9}
Extinction rate	4.9×10^{-9}
Number of gene families	100
Number of species	25
Dup and loss rates	$\delta \times Log\text{-}\mathcal{N}(0,1)$, $\delta = 4.9 \times 10^{-10}$
HGT rate	$\tau \times Log\text{-}\mathcal{N}(0,1)$, $\tau = 4.9 \times 10^{-10}$
Population size	10
Species tree height	$Log\text{-}\mathcal{N}(21.25, 0.2)$
Global substitution rate	$Log\text{-}\mathcal{N}(-21.9, 0.1)$
Lineage specific rate gamma shape	$Log\text{-}\mathcal{N}(1.5, 1)$
Family specific rate gamma shape	$Log\text{-}\mathcal{N}(1.551533, 0.6931472)$
Gene tree branch specific rate gamma shape	$Log\text{-}\mathcal{N}(1.5, 1)$
Sequence length	$\nu \times Log\text{-}\mathcal{N}(0, 0.25)$, $\nu = 100(e^{-\frac{0.25^2}{2}})$
Sequence base frequencies	Dirichlet(A=36,C=26,G=28, T=32)
Sequence transition rates	Dirichlet(TC=16,TA=3,TG=5, CA=5,CG=6,AG=15)
Seed	$[3000, 3100[$
Varying parameters	
Dup and loss rates multiplier	0.5,1.0,2.0,3.0
HGT rates multiplier	0, 0.5, 1, 2.0, 4.0
Population size	10, 10^7, 10^8, 10^9
Number of species	15, 25, 35, 50, 75
Number of gene families	50, 100, 200, 500, 1000
Average number of sites	50, 100, 200, 300

Table 5.2: SimPhy parameters to simulate the SIMDL and SIMDTL datasets. In the varying parameters section, the rate multipliers are used to scale the constants λ for the dup-loss rates and τ for the HGT rates. For sequence length, ν is set to obtain 100 sites on average.

The parameters we studied are: the average number of sites per gene family MSA, the number of families, the size of the species tree, the average DTL rates and the population size. The population size is a parameter of the Multi-Species Coalescent Model (MSCM): when the population size increases, ILS happens more frequently. For each parameter we studied, we varied its value while keeping all other parameters fixed. We generated 100 replicates for each set of parameter values. We executed the entire experiment twice, once including HGTs (D**T**LSIM experiment) and once excluding HGTs (DLSIM experiment).

We reused the default parameters of the S25 experiment of [166] with some modifications that we list in the following. By default, we do not simulate ILS, which yields the species tree inference easier than in the original S25 experiment. To make the reconstruction more challenging and to reduce the computational cost of the entire experiment, we reduced the number of families from 1000 to 100. To increase the heterogeneity among gene families, we used a log-normal distribution for the sequence length and the DTL rates. In the D**T**LSIM experiment, we simulated under the distance-independent HGT model (i.e., the receiving species is uniformly sampled from all contemporary species) and we set the HGT rates equal to the duplication rates. We provide a detailed list of the SimPhy parameters and arguments used for the gene event rates in Table 5.2.

We inferred the GFTs with ParGenes [103], under the general time reversible model of nucleotide substitution with four discrete gamma rates (GTR+G4) [147, 160]. To limit the execution time (the GFT inference is the most computationally expansive step of our experiments), we performed only one RAxML-NG search on a single random starting tree per gene family. Then, we inferred the species trees from the inferred GFTs with every tool listed in Table 5.1. Finally, for each dataset, we assessed the species tree reconstruction accuracy by computing the average relative RF distance between each inferred species tree and the corresponding true species tree using the ETE Toolkit [65].

5.3.4 Empirical datasets

We used empirical datasets from various sources to cover a wide range of organisms including plants, fungi, vertebrates, bacteria, and archaea. We describe these datasets in Table 5.3. When the datasets included outgroups, we excluded them from the analysis, because SpeciesRax does not need any outgroup to root the species trees. For datasets where we pruned outgroups and for which alignments were available, we reinferred the GFTs from the alignments. This was done to avoid any potential bias in the tree reconstruction that could be caused by the outgroup [62]. In the following we describe in detail how we assembled each empirical dataset.

5.3.4.1 Primates13 and Vertebrates188 datasets

We extracted the alignments comprising 199 species from the Ensembl Compara database [164]. We removed 5 non-vertebrates species to obtain the Vertebrates188 dataset. Further, we extracted 13 primate species to obtain the Primates13 dataset. For both datasets, we inferred the GFTs with ParGenes under the GTR+G4 model with one random starting tree per RAxML-NG search.

Dataset	Families	Genes	GFTs	Gene data source
Primates13	16670	268338	inferred	Ensembl Compara
Cyanobacteria36	1099	41035	inferred	Hogenom
Vertebrates22	18829	1521587	extracted	PhylomeDB
Fungi16	7180	85866	extracted	[22]
Fungi60	5665	391471	inferred	PhylomeDB
Plants23	21469	1652464	inferred	PhylomeDB
Life92	41222	628747	interred	[158]
Archaea364	150	46801	inferred	[36]
Plants83	9237	1294695	extracted	1000k plants
Vertebrates188	31612	3725332	inferred	Ensembl Compara

Table 5.3: Description of the empirical datasets used in our benchmark. Dataset names are suffixed by the number of species in the respective dataset. Families is the number of input gene families. Genes is the total number of gene copies in the dataset. GFTs indicates if we inferred the GFTs ("inferred") or if we extracted them from the data source ("extracted"). Gene data source is the database or the project/publication from which the GFTs and/or gene family alignments were obtained.

5.3.4.2 Cyanobacteria36 dataset

We reused the protein alignments of a previous study [143] covering 36 cyanobacteria species to generate the Cyanobacteria36 dataset. We inferred the GFTs with ParGenes under the same substitution model used in the original study (LG+G4+I) with one random starting tree per RAxML-NG search.

5.3.4.3 Fungi16 and Plants83 datasets

The Fungi16 and Plants83 datasets respectively correspond to the Plant (1kp) and Fungal datasets studied in [166]. We downloaded the respective GFTs from `https://github.com/chaoszhang/A-pro_data`.

5.3.4.4 Fungi60, Plants23, and Vertebrates22 datasets

We extracted datasets from three different phylomes of the PhylomeDB [64] database: vertebrates (phylome ID = 200), fungi (phylome ID = 3), and plants (phylome ID = 84). We removed the two outgroup species (*Arabidopsis thaliana* and Human) from the fungi phylome to generate the Fungi60 dataset. We removed the five outgroup species (outgroups: human, Drosophilia, *Caenorhabditis elegans*, *Saccharomyces cerevisiae*, and *Plasmodium falciparum*) from the plant phylome to generate the plants21 dataset. We reinferred the GFTs of both, the fungi and plants datasets using ParGenes with best-fit model selection enabled (-m option) and one random starting tree per RAxML-NG search. We generated the Vertebrates22 dataset from the vertebrates phylome. Here we did not remove any outgroup and did therefore not re-infer the corresponding GFTs.

5.3.4.5 Life92 dataset

To compare to the supertrees inferred in the original study [158], we extracted the original GFTs covering 92 species from the Eukaryote and Archaea domains. To take advantage of the signal from duplications and transfers, we also inferred new homologous gene families from the genomes used in that study. To do so, we performed all-versus-all Diamond [19] searches, then clustered gene families using mcl [41] with an inflation parameter value of 1.4. As in the original study, sequences were aligned using MAFFT [72] and poorly-aligning positions removed using BMGE 1.12 [28] with the BLOSUM30 matrix.

5.3.4.6 Archaea364 dataset

We downloaded the MSAs of the marker proteins from the original study [36]. We inferred the GFTs with ParGenes using the LG+G4 substitution model.

5.4 Results

5.4.1 Accuracy on SimPhy simulations

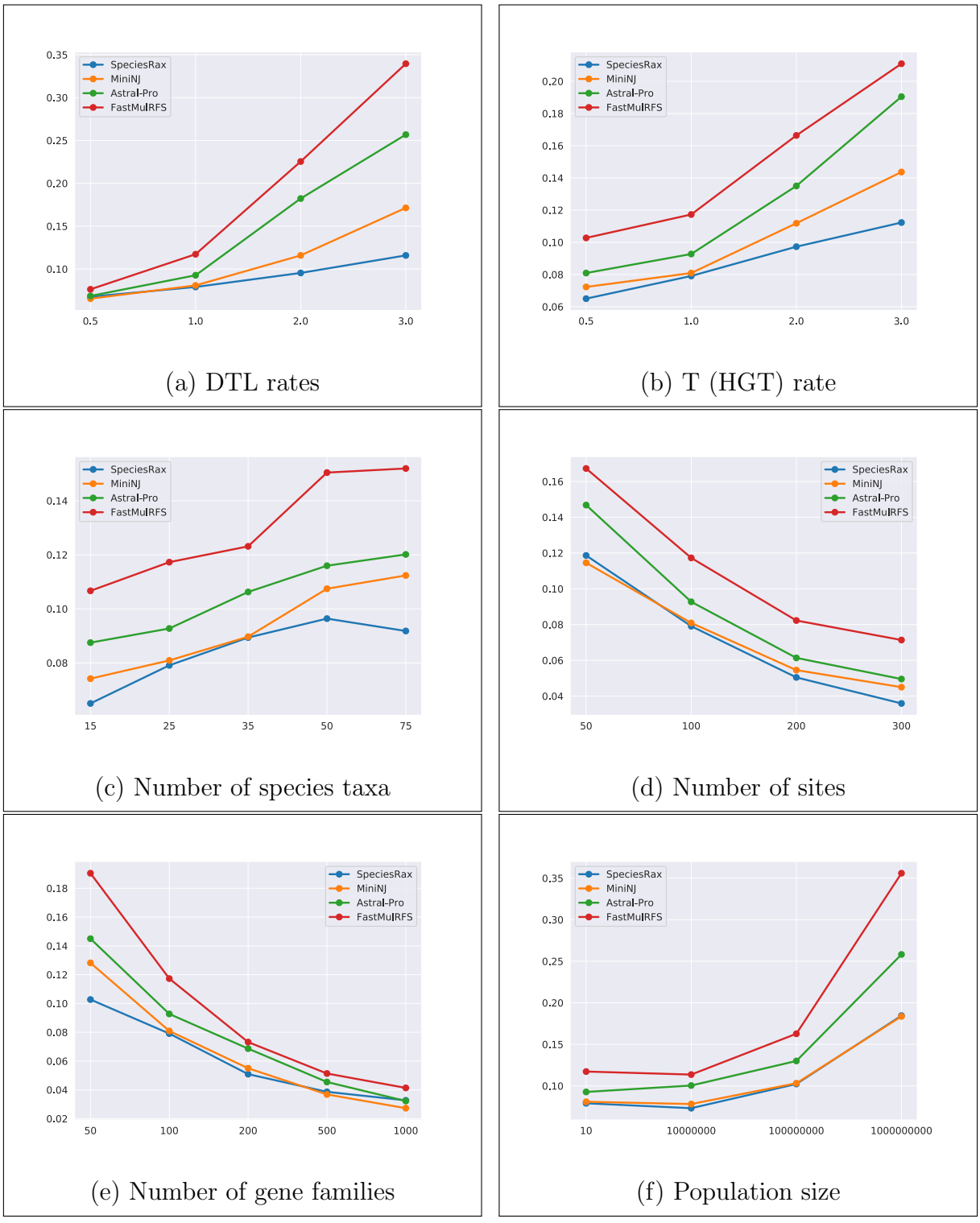
We summarize the accuracy of the different species tree reconstruction methods for the D**T**LSIM and DLSIM experiments in Table 5.4 and Table 5.5, respectively. We excluded DupTree and NJst from the D**T**LSIM plots and NJst from the DLSIM plots for the sake of an improved visual representation of the results because of their very high error rate.

In presence of HGTs (D**T**LSIM experiment), SpeciesRax performs better than the competing methods with an average relative RF distance of 0.082 (0.092 for MiniNJ, 0.115 for Astral-Pro, 0.143 for FastMulRFS, 0.409 for NJst and 0.447 for DupTree). In absence of HGTs (DLSIM experiment), we observe the same trend (0.059 for SpeciesRax, 0.063 for MiniNJ, 0.072 for Astral-Pro, 0.089 for FastMulRFS, 0.289 for NJst and 0.116 for DupTree).

As expected, all methods perform better when the phylogenetic signal (number of sites, number of families) increases and perform worse when the discordance between the GFTs and the species tree (ILS level, DTL rates) increases. We do not observe a clear correlation between the number of species and the reconstruction accuracy.

Compared to the other methods, SpeciesRax reconstruction accuracy seems to be less affected by increasing DTL rates and almost unaffected by increasing DL rates. We hypothesize that larger DTL rates increase the species tree - GFT discordance but also the signal as we obtain larger gene families. Therefore, the competing methods fail to exploit the putative increase in signal but are affected by the higher level of discordance.

Although SpeciesRax does not model ILS, its accuracy is not hampered to a larger degree by increasing population size than that of competing tools.



(a) DTL rates

(b) T (HGT) rate

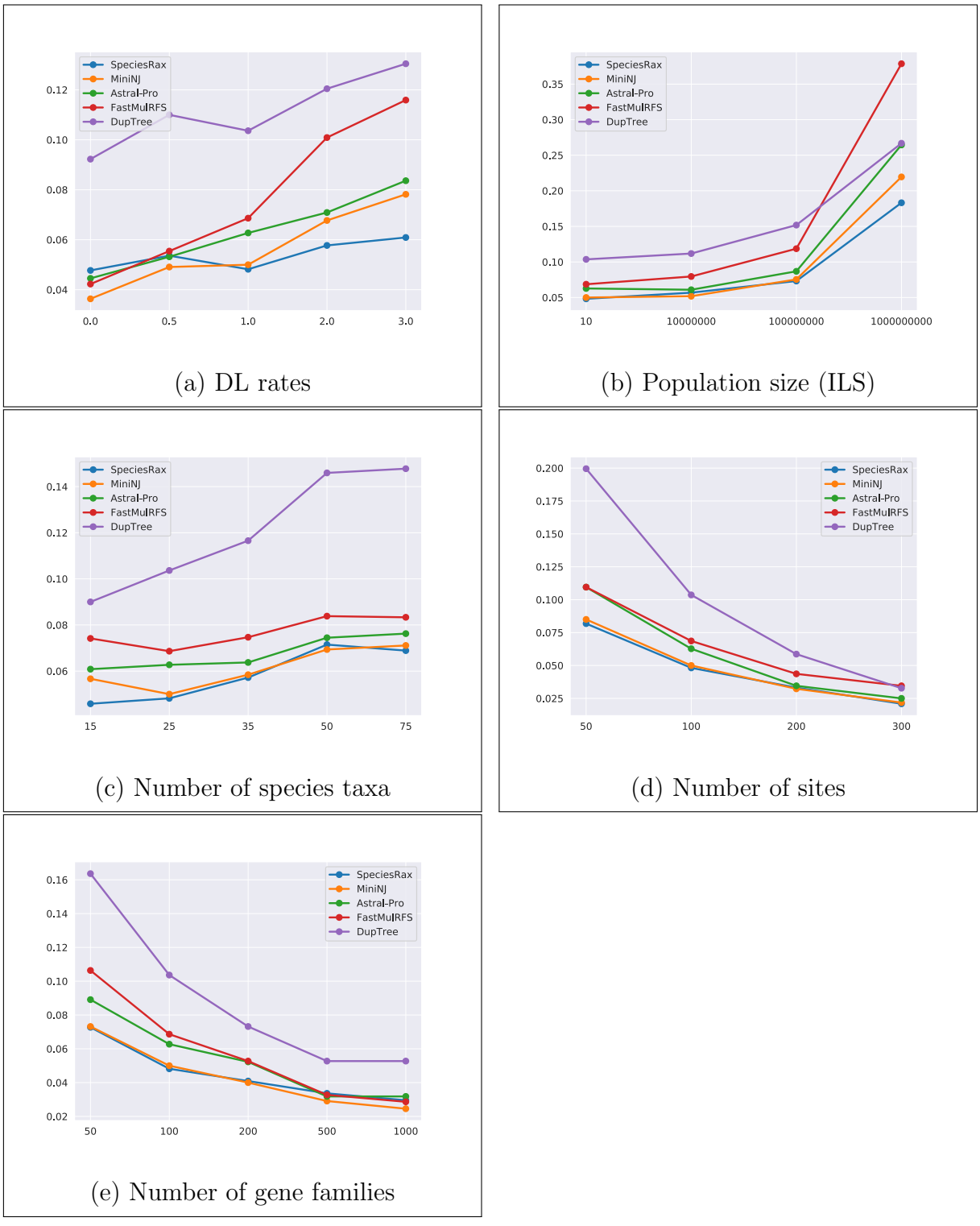
(c) Number of species taxa

(d) Number of sites

(e) Number of gene families

(f) Population size

Figure 5.4: Average unrooted RF distance between inferred and true species trees, in the presence of duplication, loss, and HGT.



(a) DL rates

(b) Population size (ILS)

(c) Number of species taxa

(d) Number of sites

(e) Number of gene families

Figure 5.5: Average unrooted RF distance between inferred and true species trees, in the presence of duplication and loss (no HGT).

5.4.2 Accuracy on empirical datasets

Here, we describe the results of species tree inferences on empirical datasets with Astral-Pro, DupTree, FastMulRFS , MiniNJ, and SpeciesRax. We excluded

NJst from this analysis because it performed poorly on most empirical datasets. Initially, we only compare *unrooted* topologies and defer the root placement analysis to a separate subsection.

5.4.2.1 Vertebrates188 dataset

All tested methods inferred a different species tree. We first counted the number of splits that differ between the inferred trees and the multifurcating NCBI taxonomy [44] tree. The SpeciesRax, Astral-Pro, and FastMulRFS tools disagree on 5 splits, MiniNJ on 6 splits, and DupTree disagrees on 20 splits.

Then, we focused on the five splits on which SpeciesRax disagrees with the NCBI taxonomy tree that we downloaded from the Ensembl Compara database. Among those discordant splits, the SpeciesRax tree seems to clearly violate only one well established phylogenetic relationship: the elephant shark is believed to have diverged before the split between *Actinopterygii* and *Sarcopterygii* [150], but SpeciesRax places it as sister to *Sarcopterygii*. Note that all tested methods (Astral-Pro, FastMulRFS, DupTree and MiniNJ) agree with SpeciesRax.

In the following we analyze the remaining four disagreements.

First, SpeciesRax (as well as all other competing tools) places *Cichliformes* as sister to *Ambassidae* while the taxonomy places *Pomacentridae* as sister to *Ambassidae*. Most studies we have found support the taxonomy [14, 66, 109]) but other studies are undecided about the resolution of these clades and present trees inferred using different inference methods that support the three alternative resolutions [42].

Another discordance with the taxonomy occurs within the avian subtree, between the *Estrildidae*, *Fringillidae*, and *Passerellidae* clades: the taxonomy groups the *Estrildidae* and *Fringillidae* together, while SpeciesRax, Astral-Pro, and FastMulRFS group Fringillidae and Passerellidae together. A recently published 363 taxon bird phylogeny [48] agrees with SpeciesRax on this split and perfectly matches the remaining 24 taxon avian subtree we inferred.

In addition, all tested tools place *Bos mutus* (yack) and *Bison bison* closer to each other than to *Bos taurus*, while the taxonomy places *Bos mutus* next to *Bos taurus*. To our knowledge, the literature agrees with our resolution [33, 79].

The last inconsistency between the taxonomy and the SpeciesRax tree occurs among the *Platyrrhini* (monkey suborder) when placing *Aotidae*, *Cebus/Saimiri*, and *Callitrichidae*. This split is perhaps more interesting because SpeciesRax disagrees with the competing methods: the taxonomy places *Cebus/Saimiri* and *Callitrichidae* together, SpeciesRax places *Aotidate* and *Callitrichidae* together. The Astral-Pro, FastMulRFS, MiniNJ, and DupTree tools all group *Aotidate* with *Cebus/Saimiri*. There exist studies that agree with the SpeciesRax [114, 137] but also with the Astral-Pro [43] resolutions of these clades.

5.4.2.2 Plants23 dataset

Both SpeciesRax and Astral-Pro species trees disagree with the literature by placing the *Malvales* as sister to *Malpighiales* (instead of sister to *Brassicales*

[5, 49]). The SpeciesRax species-driven quartet support scores, positively support our resolution, suggesting a potentially misleading signal from the GFTs. When investigating the GFTs, we observed that the *Brassicales* genes often diverge much earlier than they should and that they are often placed outside of the *Rosids* clade to which they should belong. A hypothesis for this misleading signal is the apparent overestimation of the gene family sizes during the gene family clustering performed in the original study [49] as many gene families contain 150 genes (the maximum family size cutoff used in the respective gene family clustering procedure). In addition, the GFTs exhibit clear clusters of genes covering all species separated by extremely long branches. We note however that DupTree and FastMulRFS correctly inferred the entire species tree.

5.4.2.3 Plants83 dataset

The unrooted topologies of the SpeciesRax, Astral-Pro, and FastMulRFS trees are in very good agreement with current biological opinion on the *Viridiplantae* phylogeny, recovering *Setaphyta* and the monophyly of *bryophytes* [59, 83, 117]. The SpeciesRax tree further agrees with several recent analyses [59, 83] in placing the *Coleochaetales* algae as the closest relatives of *Zygnematophyceae* and *Embryophyta* (land plants). The DupTree tree violates many well-established phylogenetic relationships.

5.4.2.4 Fungi60 dataset

All tools found a species tree that disagrees with the literature: they placed the clade formed by *Chytridiomycota* and *Zygomycota* between *Basidiomycota* and *Ascomycota*, which are typically grouped together [88, 93]. The positive EQPIC score computed with SpeciesRax along the relevant path shows that the quartets of the GFTs do support this incorrect split. We conclude that the GFTs presumably contain a misleading signal around this split. One possible explanation is that *Encephalitozoon cuniculi* is evolutionary very distant from the remaining species, potentially causing a long branch attraction effect [13]. Apart from this split, SpeciesRax, Astral-Pro, and FastMulRFS inferred the same tree, which agrees with the original species tree obtained via concatenation [93]. The tree inferred with DupTree differs from the SpeciesRax tree in one split.

5.4.2.5 Primates13, Cyanobacteria36, Vertebrates22, and Fungi16 datasets

All tools inferred the same species trees for the Primates13, Cyanobacteria36, and Fungi16 datasets and do not violate any well-established phylogenetic relationships. On the Vertebrates22 dataset, all tested methods inferred trees that agree with the multifurcating NCBI taxonomy, but the inferred bifurcating trees are nonetheless different: Astral-Pro, MiniNJ, and SpeciesRax inferred the same tree, which differs from the FastMulRFS tree by one split and the DupTree tree by two splits.

5.4.2.6 Archaea364

The original authors [36] suggested that one reason for the difficulty in resolving the archaeal tree was the presence of host-symbiont gene transfers in broadly-conserved marker genes, in which members of the DPANN Archaea sometimes grouped with their hosts in single gene phylogenies. Using the full set of marker genes, the SpeciesRax tree recovered a clan [155] of DPANN; that is, all DPANN Archaea clustered together on the tree. The unrooted SpeciesRax topology is congruent with several recent analyses of the archaeal tree [36, 122, 157].

5.4.2.7 Life92

SpeciesRax and Astral-Pro both recovered the major lineages of Archaea and Eukaryotes, including the *Euryarchaeota* and "TACK" Archaea (*Thaumarchaeota*, *Aigarchaeota*, *Crenarchaeota* and *Korarchaeota*) within the Archaea, and the SAR, *Archaeplastida* and *Amorphea* clades of Eukaryotes. Astral-Pro resolves the *Excavates* into two separate clades (*Discobans* and *Metamonads*, with *Trimastix* branching between them), while SpeciesRax unites them as sister groups, albeit with very weak statistical support (-0.03); previous work is equivocal as to whether these two lineages form a monophyletic *Excavata* clade [20, 58].

SpeciesRax recovers the monophyly of *Asgardarchaeota*, while Astral-Pro instead places one lineage, *Odinarchaeota*, with the TACK Archaea; the position recovered by SpeciesRax is the consensus view [163]. However, SpeciesRax recovers *Asgardarchaeota* as sister to the TACK Archaea, albeit with low support (-0.0075). This topology is incompatible with a specific relationship between Eukaryotes and *Asgardarchaeota*, as supported by analyses of conserved marker genes [136, 158, 163]. The unrooted tree inferred by Astral-Pro groups *Asgardarchaeota* (without *Odinarchaeota*) with Eukaryotes, and is therefore compatible with an origin of the eukaryotic host cell from within the *Asgardarchaeota*.

5.4.3 Rootings

In the following, we conduct an in depth assessment of the accuracy of the species tree root inference with SpeciesRax on the tested empirical datasets.

We first discuss the datasets on which SpeciesRax inferred a species tree root that agrees with the current literature. On the Primates13 dataset, SpeciesRax correctly places the species tree root between the *Strepsirrhini* and *Haplorhini* clades [24]. On the Fungi16 dataset, the root inferred with SpeciesRax correctly separates the *Candida* and *Saccharomyces* clades [22]. The root we inferred on the Vertebrates22 species tree correctly separates the *Actinopterygii* and *Sarcopterygii* clades [96]. On the Plants23 dataset, our species tree root correctly separates the *Chlorophyta* and *Streptophyta* clades [84]. On the Fungi60 dataset, we correctly find that *Encephalitozoon cuniculi* (*Microsporidia* clade) diverged earlier than the other clades contained in the dataset [108]. On the Vertebrates188 dataset, SpeciesRax infers a root that groups lampreys and hagfishes, on one side, and cartilaginous fishes, bony fishes, and tetrapodes on the other side. The position of the vertebrate root is

still controversial [100, 146] and our resolution complies with some of the plausible scenarios discussed in the literature [96, 100, 146].

On the Plants83 dataset, SpeciesRax agrees with the literature in placing *Embryophyta* (land plants) within the *Streptophyte* algae. However, the inferred root is three branches away from the consensus position, in the common ancestor of the *Chlorophyta* (*Volvox*, *Chlamydomonas* and *Uronema*).

On the Cyanobacteria36 dataset, the root placement inferred by SpeciesRax is one branch away from one of the three plausible roots inferred in a recent study [142].

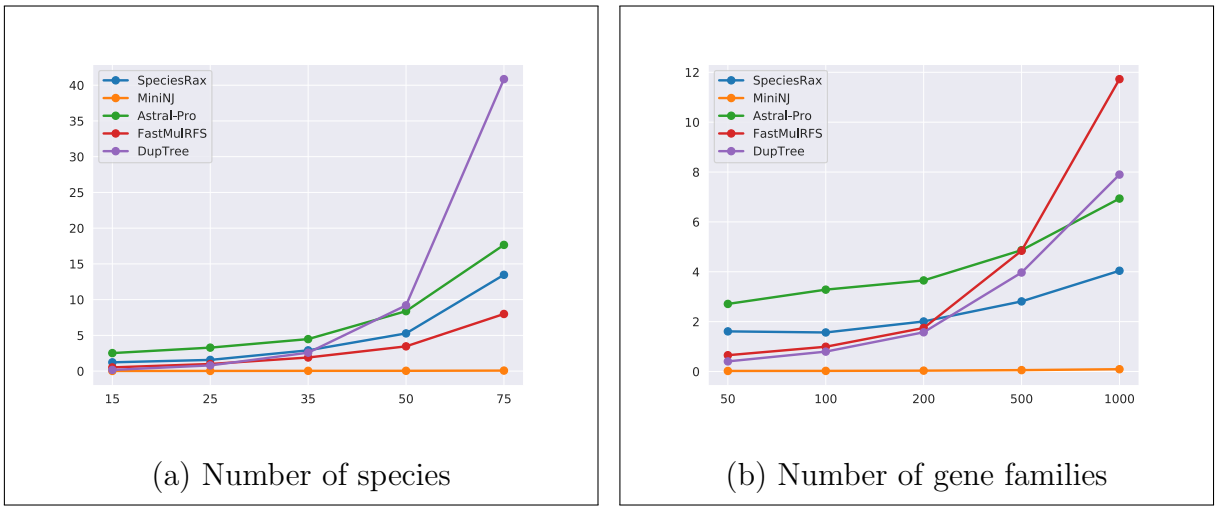
The Archaea364 dataset only contained single-copy gene families, and thus no gene duplications. As a result, the position of the root was uncertain. However the 95% confidence set of possible root placements obtained via the Approximately Unbiased (AU) test [134] was compatible with several recent suggestions in the literature, including a root between DPANN and all other Archaea [36, 157] and a root within the Euryarchaeota [122], among a range of other positions within and between the major archaeal lineages.

The root inferred by SpeciesRax on the Life92 dataset is biologically not plausible as it is located between *Viridiplantae* and all other taxa. One possibility is that root inference for these data is affected by large differences in gene content among the included taxa. For example, the *Viridiplantae* (and other *Archaeplastida*) have chloroplasts, and so possess an additional source of bacterial-origin genes compared to other Eukaryotes and Archaea. To evaluate the impact of major gene content differences, we performed another SpeciesRax analysis in which the gene families covering less than half of the species were removed. In this second analysis, the root was inferred to lie between the Eukaryotes and Archaea. This root position is compatible with a three-domains tree of life hypothesis. However, this should be interpreted with caution, because the branch separating Eukaryotes and Archaea is one along which major gene content changes occurred, including (but not limited to) the acquisition of a bacterial genome's worth of genes in the form of the mitochondrial endosymbiont [126].

5.4.4 Runtime

Before comparing runtimes, we emphasize again that we executed the experiments on a 40 core machine and that only SpeciesRax and Astral-Pro provide a parallel implementation. While this choice might appear to favor SpeciesRax and Astral-Pro, we argue that the absence of parallelization constitutes a substantial limitation of the remaining tools as completing an analysis in less than one day on a parallel system instead of having to wait for several weeks represents a strong advantage.

We also emphasize that SpeciesRax is the *only* tested tool that can be executed across several compute nodes with distributed memory in contrast to Astral-Pro that can only run on a single shared memory node. All tools, with the exception of MiniNJ, required huge amounts of memory for the largest dataset ($> 200GB$ on



(a) Number of species

(b) Number of gene families

Figure 5.6: Average runtime in seconds for species tree inference.

Dataset	FMR	DupTree	Astral-Pro	SpeciesRax	MiniNJ
Primates13	62s	18s	38s	14s	2s
Cyanobacteria36	19s	26s	12s	14s	< 1s
Fungi16	9s	7s	18s	7s	< 1s
Vertebrates22	9min	5min	2min 45s	2min 30s	7s
Fungi60	16min	17min	1min 30s	1min	2s
Plant23	9min	6min	1min 35s	2min	8s
Life92	6h	2h20min	31min	6min	2s
Archaea364	11min	6h	1min	14min	10s
Plants83	4h	2h 40min	1h 40min	8min	27s
Vertebrates188	14 days	3.5 days	12h	1h 5min	53s

Table 5.4: Species tree inference runtimes for all tested tools. FMR stands for FastMulRFS).

Vertebrates188) and can therefore not be executed on most common servers. The SpeciesRax MPI implementation allows to distribute the memory footprint over different compute nodes, which is not feasible with the other tools.

We show the runtimes for an increasing number of species and an increasing number of families on the simulated datasets in Figure 5.6. Our MiniNJ method requires less than 0.1 second for all parameter combinations and is the fastest method we tested. The runtimes of DupTree and FastMulRFS grow faster with increasing number of gene families, and DupTree runtime quickly raises with the number of species. The SpeciesRax and Astral-Pro runtimes are less affected by these parameters.

On almost all empirical datasets, MiniNJ and SpeciesRax are the fastest methods. On the two largest datasets (Plants83 and Vertebrates188), MiniNJ is at least one order of magnitude faster than SpeciesRax and SpeciesRax is at least one order of magnitude faster than all other methods. In particular, SpeciesRax only requires

one hour on 40 cores to infer the 188 vertebrate species tree with 188 species and 31612 gene families.

5.5 Discussion

5.5.1 A fast and accurate approach

We introduced two new methods for species tree inference from GFTs in the presence of paralogy. Our MiniNJ tool, is a distance based method that is faster than all tested methods while being at least as accurate as all other non-parametric methods on the majority of our simulated data experiments. In particular, MiniNJ inferred a species tree with 188 species in less than one minute from more than 30000 gene families. SpeciesRax, is a novel maximum likelihood tree search method that explicitly accounts for gene duplication, gene loss, and HGT events. Our SpeciesRax tool infers rooted species trees with branch lengths in units of mean expected substitutions per site. Further, to assess the confidence of the inferred species tree, we introduce several quartet based support measures.

In terms of accuracy, SpeciesRax is more accurate than its competitors on simulated datasets, and up to twice as accurate under high duplication, loss, and HGT rates. On empirical datasets, SpeciesRax is on par with or more accurate than its competitors. In addition, among the tested tools, SpeciesRax and DupTree are the only methods that can infer *rooted* species trees. SpeciesRax inferred the correct (biologically well-established) roots on 6 out of 10 empirical species trees, and found roots that are close to the plausible roots in 3 out of the remaining 4 datasets (Plants83, Archaea364 and Cyanobacteria). For the most challenging-to-root dataset (Life92), we managed to infer a plausible root by removing those gene families that only covered less than half of the species.

Despite being a compute-intensive maximum likelihood based tree search method, SpeciesRax is faster than all tested methods (except MiniNJ) on large empirical datasets. This is due to our fast method MiniNJ for inferring a reasonable starting tree and to our efficient reconciliation-aware search strategy. In addition, SpeciesRax provides a parallel implementation and can be run on distributed memory cluster systems. Thereby it facilitates conducting large-scale analyses.

5.6 Data availability

The code is available at `https://github.com/BenoitMorel/GeneRax` and data are available at `https://cme.h-its.org/exelixis/material/speciesrax_data.tar.gz`.

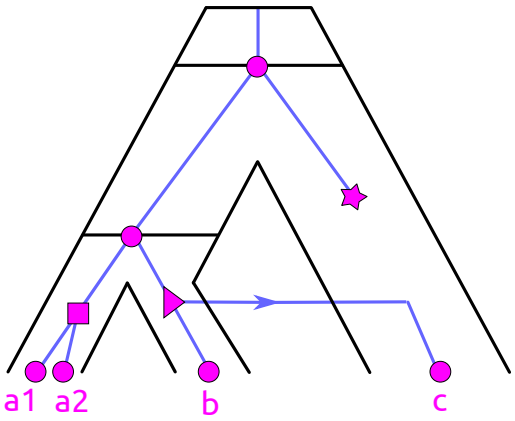
6. Species tree aware gene family tree inference and reconciliation with GeneRax

6.1 Introduction

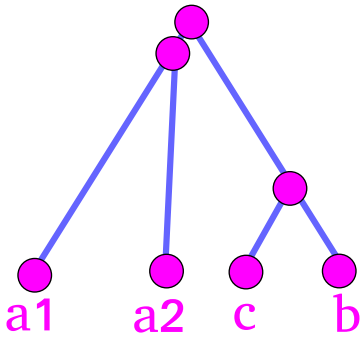
Reconstructing the evolutionary history of homologous genes constitutes a fundamental problem in phylogenetics, as Gene Family Trees (GFTs) play a prominent role in numerous biological studies. For instance, GFTs are essential to understand genome dynamics [149], to study specific traits [107], or to infer the species tree [18, 99].

Standard phylogenetic methods infer GFTs from per-gene MSAs, for instance using the ML criterion (see Section 3.2.2). Under the correct substitution model, ML methods are statistically consistent [161], that is, they converge to the true tree when the sequences are long enough. However, this condition is often violated for GFTs: typical per-gene MSAs are short (50 to 1000 sites) and can comprise a large number of sequences representing a large number of *taxa* (hundreds or thousands for large gene families). As a result, there is typically insufficient signal in the MSA to reconstruct a well supported phylogeny. In other words, the tree with the highest likelihood might not correspond to the true GFT.

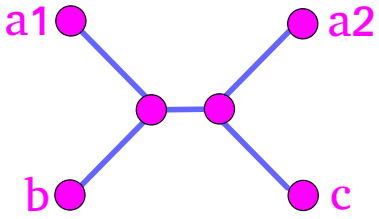
Species Tree Aware (STA) approaches (see Section 3.4) aim to compensate for this insufficient phylogenetic signal by relying on a putative species tree. Indeed, GFTs and the species tree exhibit an intricate relationship: genes evolve within a (species) genome and undergo biological processes such as gene duplication, HGT, gene loss, or speciation (Figure 6.1). Therefore, although GFTs can be incongruent with the species tree, their own evolutionary history is, to a substantial degree, determined by the species tree. STA methods exploit this relationship between the GFTs and the



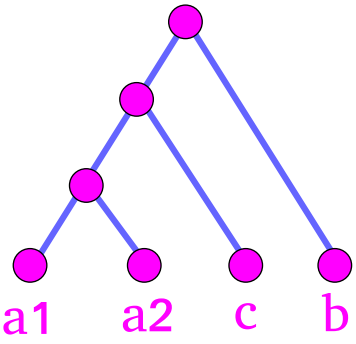
(a) Real scenario



(b) True GFT



(c) GFT inferred from the sequences only



(d) GFT inferred from the species tree only

Figure 6.1: Example of a reconciliation scenario and several possible inferred GFTs. (a) The true history. The GFT (blue lines) evolves within the species tree, and undergoes speciation (circle), gene duplication (square), gene loss (star), and HGT (triangle) events. (b) The true rooted GFT. (c) An unrooted GFT inferred using a sequence-aware method. The splits between gene lineages are very close in time, and there is not enough signal in the sequences to correctly infer the unrooted GFT topology. (d) Rooted GFT inferred from the species tree only (without taking into account the sequences), assuming that HGTs are less likely than gene duplications.

species tree to leverage additional information for GFT inference. In this chapter, we denote gene duplication, gene loss, and HGT events as *DTL events*.

A common approach used by STA methods [25, 111, 132] consists in contracting weakly supported GFT branches into polytomies, which are subsequently resolved using the species tree (see Section 3.4.1). These heuristics limit the set of GFTs explored to trees that can be obtained as combinations of alternative resolutions of the contracted branches. Most existing implementations [25, 111] are based on parsimony, and require an a priori specification of arbitrary DTL parsimony costs. This is particularly problematic if the substitution model is misspecified, or if it fails to adequately capture the complexity of the data. This is commonly the case for shorter gene alignments where parameter rich substitution models are more difficult to use. In addition, the user must define what a "low support value" for branch contraction is, often by setting an arbitrary threshold. In Treerecs [27] we addressed this last limitation by exploring several thresholds, and returning the GFT that maximizes a likelihood score that is based on both, the MSAs, and the species tree. Finally, obtaining branch support values usually requires a substantial amount of computational effort (e.g., 1-2 orders of magnitude more than for a simple ML tree search on the original MSA, if the classic Felsenstein Bootstrap procedure is used [46]).

Other STA methods utilize a hierarchical probabilistic model of sequence level substitutions and gene level events, such as gene duplication, HGT, and gene loss. This allows to define the *joint likelihood* as the product of the probability of observing the alignments given the GFTs (the *phylogenetic likelihood*, see Section 2.3.3) and the probability of observing the GFTs given the species tree (the *reconciliation likelihood*, see Section 2.4.7):

$$L(\mathcal{G}, S|\mathcal{A}) \propto \prod_{G_i \in \mathcal{G}} P(A_i|G_i)P(G_i|S) \tag{6.1}$$

where S is the species tree, $\mathcal{G}$ is the set of GFTs, and $\mathcal{A}$ the set of corresponding MSAs. PHYLDOG (see Section 3.3.2.3) co-estimates the GFTs and the species tree by conducting a tree search that is based on such a joint likelihood score. However, PHYLDOG does not model HGT. ALE (see Section 3.4.2) calculates the joint likelihood using a dynamic programming scheme that requires the phylogenetic likelihood to be approximated via conditional clade probabilities [81]. In order to calculate conditional clade probabilities, ALE requires a sample of GFTs as input that are typically obtained via Markov Chain Monte Carlo (MCMC) sampling. This approach has two shortcomings.

First, the conditional clade probability approximation inevitably limits the set of GFTs explored to trees that are comprised of clades observed in a tree sample, as the phylogenetic likelihood of all other trees is approximated to be zero [143]. While being less restrictive, conceptually this limitation is nonetheless analogous to those induced by the branch contraction methods discussed above. It is also similarly sensitive to model mis-specification and inadequacy.

Secondly, obtaining a tree sample, either via Bayesian phylogenetic MCMC methods or via bootstrap methods for a set of gene families is computationally expensive.

In addition, there is no method for assessing with certainty that an MCMC run converged. For an in depth review of GFT inference methods, see [39, 145].

Probabilistic frameworks to model both, sequence [45], and gene evolution events [3, 133, 144] can be found in the literature. However, no ML tool can currently directly infer GFTs from MSAs by simultaneously accounting for sequence substitutions and DTL events. We believe that such a method can substantially improve the accuracy of GFT inference. A common argument against using STA ML approaches is the amount of time and computational resources required to conduct such analyses [39]. However, a joint (phylogenetic and reconciliation likelihood) ML approach does not require expensive pre-processing and can therefore decrease the overall computational cost substantially, while increasing accuracy at the same time. Tree search heuristics are widely used to infer phylogenies from sequence data [76, 110] using the phylogenetic likelihood. Thus, extending these methods by joint likelihood calculations represents a natural way of improving the accuracy of GFT inference.

Here we introduce GeneRax, our novel software to infer ML reconciled GFTs, based on a joint reconciliation and phylogenetic likelihood. We use the term *reconciled GFT* to designate both, the GFT topology, and its reconciliation with the species tree. The input for GeneRax consists of a rooted, but undated (that is, branch lengths are not given) binary species tree, a set of per-family MSAs (DNA or amino-acid), and corresponding gene-to-species leaf name mappings. Several gene copies from the same gene family can be mapped to the same species. In addition, the user can provide initial GFTs, typically inferred via standard phylogenetic methods [76, 110]. GeneRax is easy to use, models DTL events, and can process multiple gene families in parallel. Employing a hierarchical probabilistic model allows it to simultaneously account for both, the signal from the gene family MSAs, and from the species tree. It estimates all substitution and DTL event intensity parameters, and does neither require any *ad hoc* threshold nor any arbitrary DTL event parsimony costs.

Nonetheless, one should keep in mind that ILS constitutes another important source of discordance between GFTs and the species tree. A recent study suggests that ILS can bias reconciliation inference [168]. To this end, we also assess the impact of ILS on the reconstruction accuracy of STA methods and discuss the limitations of GeneRax in the presence of ILS.

6.2 New Approaches

In this section, we outline the joint likelihood computation, our tree search algorithm, our GFT and species tree reconciliation algorithm, and our parallelization scheme.

6.2.1 Joint likelihood evaluation

GeneRax attempts to maximize the joint likelihood defined as:

$$L(\mathcal{G}, S, \Theta|\mathcal{A}) \propto \prod_{G_i \in \mathcal{G}} L(S, \Theta|G_i) L(G_i|A_i) \tag{6.2}$$

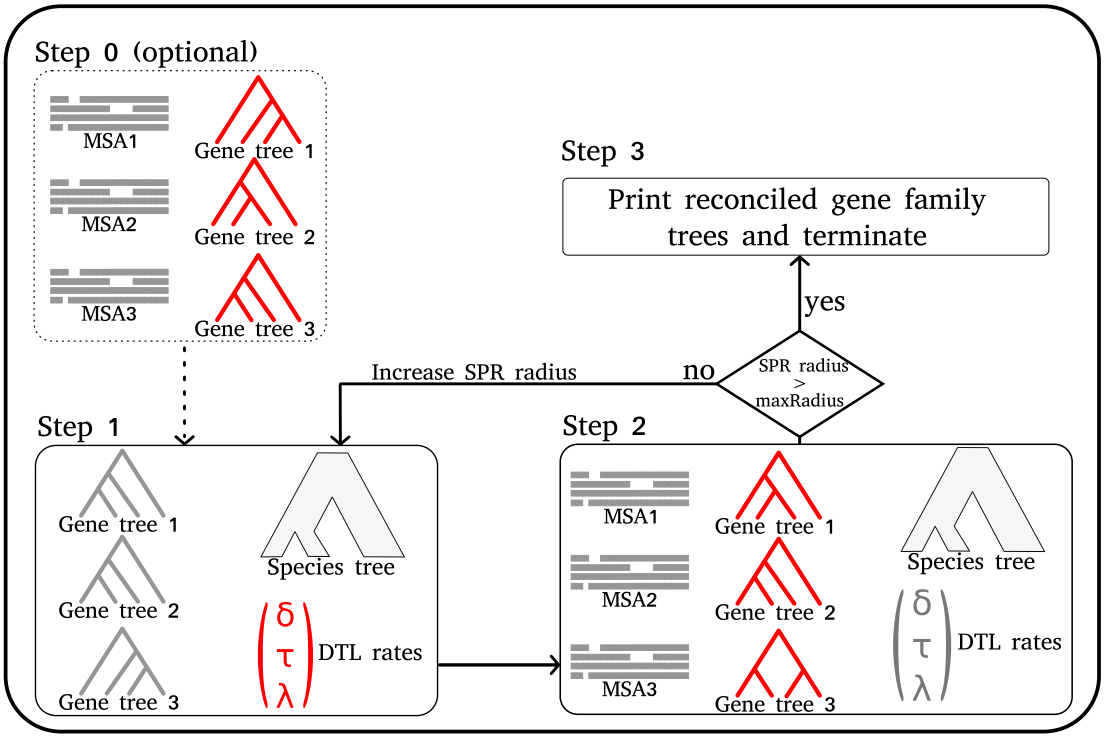


Figure 6.2: The GeneRax pipeline. In each step, we highlight in red the parameters that GeneRax optimizes, and in grey the fixed parameters that GeneRax uses to compute the likelihoods. GeneRax performs Step 0 only when starting from random GFTs, to infer ML GFTs from the MSAs. Step 1 optimizes the DTL event intensities from the GFTs and the species tree. Step 2 optimizes the GFTs from the MSAs, the species tree and the DTL intensities. GeneRax repeats Step 1 and Step 2 with increasing SPR radius, until it reaches the maximum radius. Then it applies Step 3 to reconcile the GFTs with the species tree.

where $\mathcal{G}$ is the set of GFTs, S is the species tree, Θ are the DTL event intensity parameters, and $\mathcal{A}$ is the set of gene family MSAs.

GeneRax estimates the reconciliation likelihood $L(S, \Theta|G_i)$ based on the dynamics programming recursion described in Section 2.4.7. It uses the highly optimized *pll-modules* library [30] to compute the phylogenetic likelihood $L(G_i|A_i)$. Hence, GeneRax offers all substitution models that are also supported by RAxML-NG [76].

6.2.2 Joint likelihood optimization

Given a set of MSAs and a species tree, GeneRax searches for the set of rooted GFTs and DTL intensity parameters that maximizes the joint likelihood. We illustrate the search procedure in Figure 6.2.

GeneRax either starts from user-specified GFTs or from random GFTs. Our joint likelihood search algorithm needs to start from GFTs with high phylogenetic

likelihood, preferably inferred with phylogenetic ML tools such as RAxML-NG [76]. We provide a rationale for this in Section 6.4. When starting from random GFTs, GeneRax performs an initial search (Step 0 in Figure 6.2) that solely maximizes the phylogenetic likelihood, without accounting for the reconciliation likelihood.

After this optional step, GeneRax starts optimizing the joint likelihood, by alternating between optimizing the GFTs and the DTL event intensity parameters.

When optimizing the GFTs (Step 1 in Figure 6.2), GeneRax processes each family independently, and applies a tree search heuristic to each of them separately: for a given tree, it tests *all* possible SPR moves within a given radius and subsequently applies the SPR move that yields the tree with the highest joint likelihood. Then it iterates by again applying SPR moves to this new tree, until the joint likelihood can not be further improved. At the end of the GFT optimization, GeneRax increases the SPR radius by one until a certain maximum value is reached (see further below).

GeneRax optimizes the DTL intensity parameters globally over all gene families (Step 2 in Figure 6.2). To this end, we apply the gradient descent method to find a set of DTL intensity parameters that maximizes the reconciliation likelihood over all gene families. We numerically approximate the gradient via finite differences.

The entire procedure stops when the SPR radius (starting from 1) exceeds a user-defined value. When the user does not define this maximum SPR radius, we set it to 5, as we did not observe any improvement above this value in our experiments.

6.2.3 GFT and species tree reconciliation

The reconciliation likelihood computation algorithm (described in Section 2.4.7) conducts a post-order traversal of both, the species tree, and the GFT, and sums over all possible scenarios at each step of the traversal. To infer the ML reconciliation (Step 3 in Figure 6.2), GeneRax keeps track of the maximum likelihood path during the traversal.

GeneRax can export the reconciled GFTs into both NOTUNG [25] and RecPhyloXML [38] formats (Figure 6.3).

6.2.4 Parallelization

Achieving 'good' parallel efficiency given a large number of gene families is challenging: the most straight-forward solution consists in assigning a subset of gene families to each core [18]. However, gene family MSAs are highly heterogeneous in terms of size, and are hence hard to evenly distribute over cores [103] such as to achieve 'good' load balance. In particular, large gene family MSAs can easily generate a parallel performance bottleneck. Our solution allows to split up individual inferences on such large gene family MSAs across several cores. Thus, we parallelize over, but also within gene families, in analogy to our ParGenes [103] tool. However, unlike ParGenes, GeneRax parallelizes individual GFT searches over the possible SPR moves and *not* over MSA sites. For a given GFT, we distribute the SPR moves we intend to apply among the cores assigned to the reconciliation of the GFT and apply

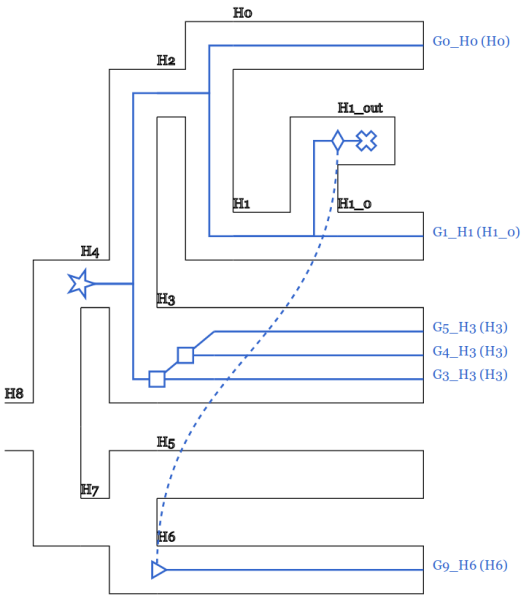


Figure 6.3: Reconciled GFT and species tree. Users can easily visualize reconciliations inferred with GeneRax using the online tool RecPhyloXML-visu [38]. This example illustrates one HGT and two gene duplication events.

Dataset	Database	Species	Families	Avg. sites	Avg. genes
Primates	ENSEMBL	13	1523	84	45
Cyanobacteria	HOGENOM	36	1099	239	37

Table 6.1: Description of the empirical datasets. We extracted the Primates dataset from the release 96 of the Ensembl Compara database [164]. The Cyanobacteria dataset was originally used in a previous study [143] and was extracted from the HOGENOM database [113].

them simultaneously. We adopted this parallelization approach for two reasons: (1) unlike the phylogenetic likelihood, the time for computing a reconciliation likelihood does not depend on the number of sites (i.e., a parallelization will not scale with the number of sites in contrast to the phylogenetic likelihood), and (2) per-MSA gene sequences are typically not long enough to efficiently parallelize the phylogenetic likelihood calculations over the sites.

6.3 Experiments

We compared GeneRax to competing GFT inference methods on both, simulated, and empirical datasets.

Software	Method type	Input trees	STA	HGT	Ref.
RAxML-NG	ML	Random	No	No	[76]
NOTUNG	Parsi	ML+S	Yes	No	[25]
Treerecs	Parsi + ML	ML+S	Yes	No	[27]
PHYLDOG	ML	ML	Yes	No	[18]
ecceTERA	Parsi	MCMC or ML+S	Yes	Yes	[132]
ALE	ML	MCMC	Yes	Yes	[143]
GeneRax	ML	Random or ML	Yes	Yes	(this paper)

Table 6.2: Software used in our benchmark. Method type indicates the type of method: ML (ML), parsimony (Parsi) or both. Input trees indicates the nature of the input GFTs: random tree (Random), ML tree (ML), ML tree with bootstrap support values (ML+S) or MCMC sample of trees (MCMC)). STA indicates whether the method is STA, and HGT whether the method accounts for HGT.

6.3.1 Tested software

This subsection describes the settings we used for executing the competing tools (summarized in Table 6.2) in all of our experiments.

We used ParGenes [103] to run RAxML-NG with 10 random and 10 parsimony starting trees and 100 bootstrap trees. For methods requiring starting GFTs, we selected the tree with the best likelihood found by RAxML-NG. We used 100 bootstrap trees to compute GFTs with branch support values as required for NOTUNG and Treerecs. As NOTUNG does not provide any explicit recommendation for setting the bootstrap support threshold, we used the default value (90%). We executed Treerecs with its automatic threshold selection from seven threshold values (seven is the default value). We executed PHYLDOG with a fixed species tree using a maximum SPR radius of 5, as in GeneRax, since PHYLDOG does not have a recommended setting. To execute ALE, we first generated posterior tree samples with MrBayes [127], using two independent runs, four chains, $1,000,000$ generations, a sampling frequency of $1,000$ and a burn-in of 100 trees. We used the UndatedDTL model to produce 100 tree samples per gene family. We used the same MrBayes tree samples to execute ecceTERA with the amalgamate option, without transfer from the dead, and with the dated species tree option.

Note that, Treerecs, NOTUNG, MrBayes, ecceTERA, and ALE do not provide a parallelization over gene families for typical distributed memory compute cluster systems. To execute them on large datasets, we scheduled them with a dedicated MPI program, by dynamically assigning jobs (with one job per gene family) to the available MPI ranks, starting from the most expensive jobs with the largest gene family MSAs. Henceforth, we refer to *sequential runtime* as the sum of the time required by each program, and to *parallel runtime* as the elapsed time spent for the entire MPI run. For a given number of cores, the *parallel efficiency* is the sequential runtime divided by the product of the parallel runtime and the number of cores.

We executed GeneRax with default parameters and with both, random (GeneRax-random), and RAxML-NG (GeneRax-raxml) starting trees. When not stated otherwise, we present GeneRax results for random starting trees.

When working on simulated datasets that were not expected to contain HGT, we executed both, ALE, and GeneRax with a HGT rate set to zero, and denote these runs as ALE-DL and GeneRax-DL. When accounting for HGT, we denote them as ALE-DTL and GeneRax-DTL.

6.3.2 Simulated datasets

We executed all tools listed in Table 6.2 on the dataset originally used to benchmark ALE [143]. Szöllosi *et al.* initially inferred GFTs for 1099 Cyanobacteria gene families using ALE. Then, they simulated new sequences under the LG+Γ+I model along these trees, retaining both, the MSA sizes, and branch lengths. In our experiments, we inferred GFTs once under LG+Γ+I (true substitution model) and once under WAG without rate heterogeneity (misspecified substitution model).

In addition, we generated additional simulated datasets to investigate the influence of various parameters on the methods and their respective accuracy. The parameters we studied are the number of sites, the average gene branch lengths, the species tree size, and the DTL intensity parameters. We also used putative species trees that were increasingly different from the true species tree to quantify the robustness of the methods with respect to topological errors in the species tree. We simulated the species tree and GFTs using GenPhyloData [135], and the sequences using Seq-Gen [118], which simulates a continuous time birth and death process along a time-like species tree.

Finally, we executed simulations using SimPhy [92] with increasing population sizes to assess the impact of ILS (the higher the population size, the more ILS occures). We define the *ILS discordance* of a simulated dataset as being the average relative Robinson-Foulds (RF) distance [123] between the true species tree and the true GFTs obtained when running the same simulations without D, T, or L events.

6.3.3 Empirical datasets

We executed all methods in Table 6.2 on the empirical datasets listed in Table 6.1. We measured both, sequential, and parallel runtimes. We also used GeneRax to evaluate the joint likelihood of the trees inferred with each method, to assess the quality of our tree search algorithm whose goal is to maximize this likelihood.

6.4 Results

In the following, we present the results of our experiments. For all methods, we report GFT quality (measured by RF distance to the true trees on simulated datasets, and joint likelihood on empirical datasets) and computational efficiency (measured by sequential runtime and parallel efficiency). All data and all inferred trees are available at `https://cme.h-its.org/exelixis/material/generax_data.tar.gz`.

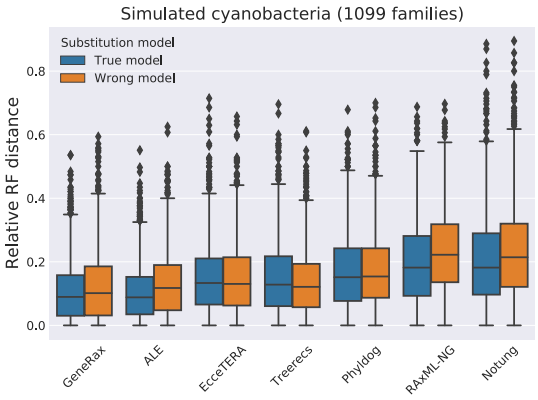


Figure 6.4: Accuracy results on the simulated cyanobacteria dataset. We represent the relative RF distances to true trees, by inferring GFTs with the true substitution model (LG+Γ+I) and a misspecified substitution model (WAG).

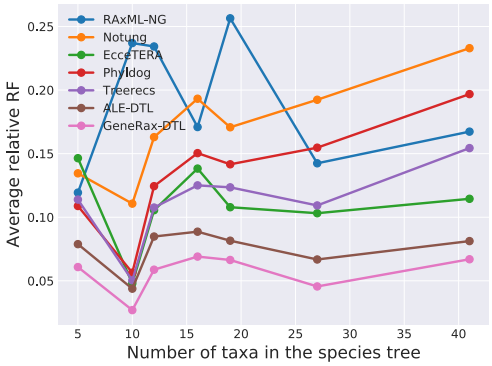
6.4.1 RF distances to true trees

We show the relative RF distances between the 1099 simulated Cyanobacteria true GTRs and the respective inferred GTRs in Figure 6.4. For methods that yield more than one potential GFT per gene family (ALE and RAxML-NG), we average the distance over all inferred trees.

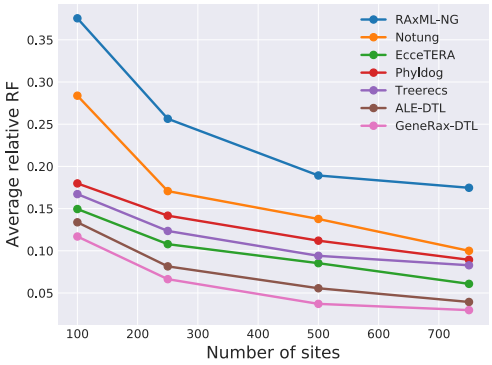
GeneRax and ALE perform better than all other methods, except in the case of the misspecified substitution model where Treerecs performs equally well. Under the true model, STA methods that do not account for HGT but use a joint likelihood score (PHYLDOG and Treerecs) perform better than the purely sequence-based method (RAxML-NG), but worse than methods accounting for HGT. Although ecceTERA accounts for transfers, it only performs as good as Treerecs, presumably because the ecceTERA algorithm only uses parsimony. We hypothesize that NOTUNG performs worse than all the other methods because it rearranges trees based on a parsimony score and an arbitrary support value threshold.

We summarize the results of the GenPhyloData simulations where we vary parameters (DTL intensity parameters, etc.) in presence of HGT in Figure 6.5, and the results of the simulations in absence of HGT in Figure 6.6. GeneRax finds the best trees in 90% of our simulation scenarios, but ALE finds trees that are almost as good in most simulations. Treerecs and PHYLDOG perform almost as well as GeneRax and ALE in the absence of HGT, but worse under HGT. NOTUNG almost always performs worse than all other STA methods.

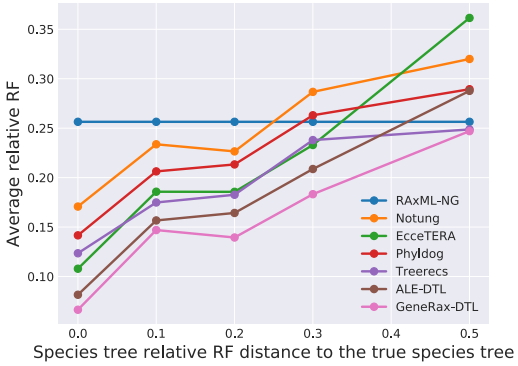
All STA methods show an analogous accuracy pattern when we vary parameters: they perform better with increasing gene sequence signal strength (Figure 6.5 (b)



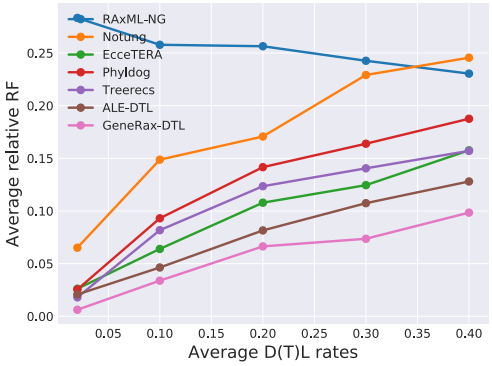
(a) Species taxa number



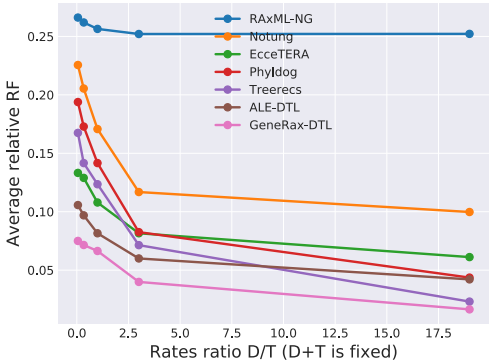
(b) Sites number



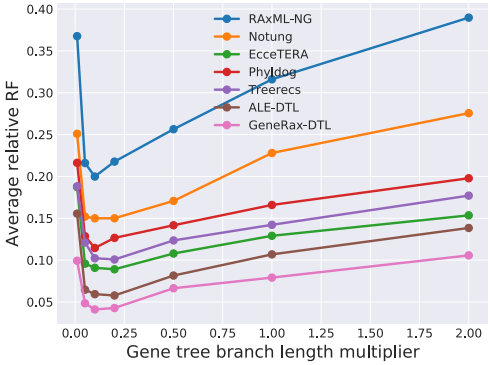
(c) Increasingly wrong species tree



(d) Average DTL rates

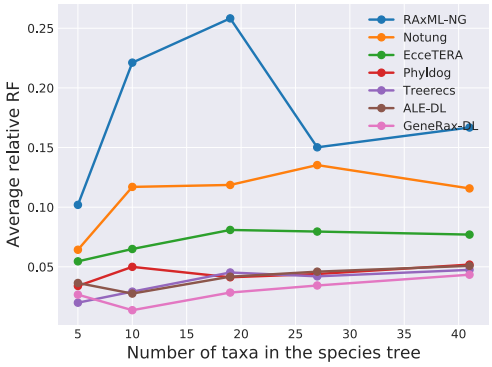


(e) Ratio between duplication and transfers rates

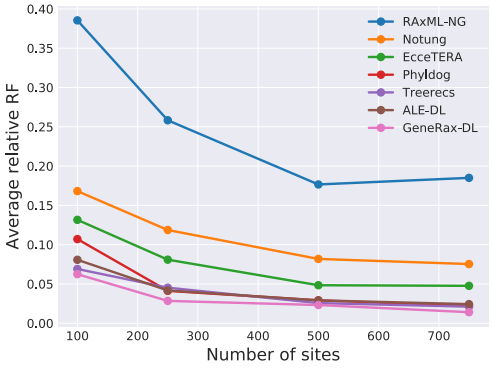


(f) GFT branch lengths

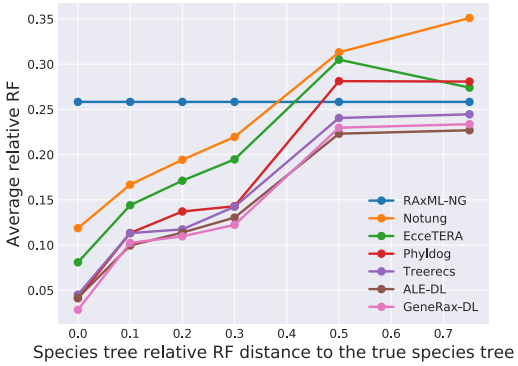
Figure 6.5: Accuracy results on simulations with HGT. Comparison of different GTF correction tools on simulated datasets, in *presence* of HGTs.



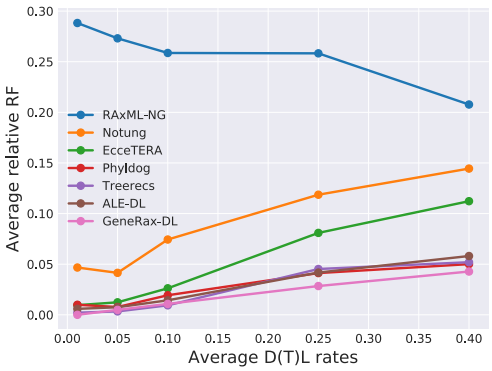
(a) Species taxa number



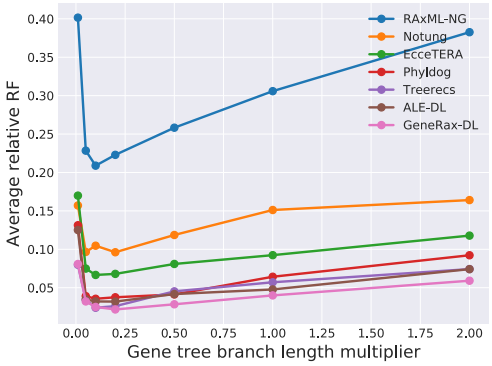
(b) Sites number



(c) Increasingly wrong species tree



(d) Average DTL rates



(e) GFT branch lengths

Figure 6.6: Accuracy results on simulations without HGT. Comparison of different GTF correction tools on simulated datasets, in *absence* of HGTs.

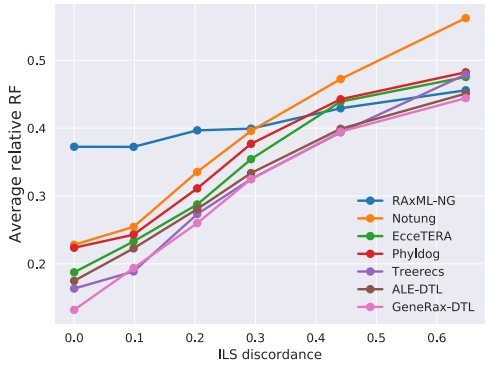


Figure 6.7: Accuracy results on simulations with ILS. RF distance to true trees on simulated datasets with increasing discordance due to ILS.

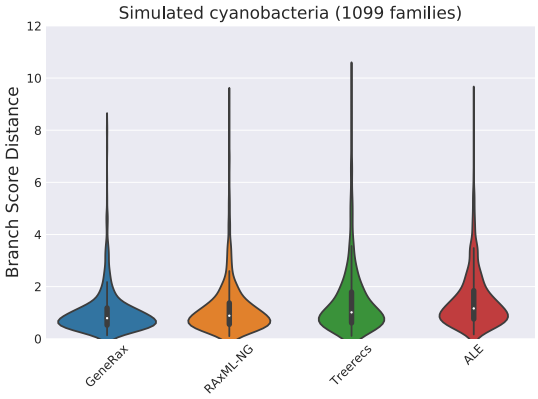


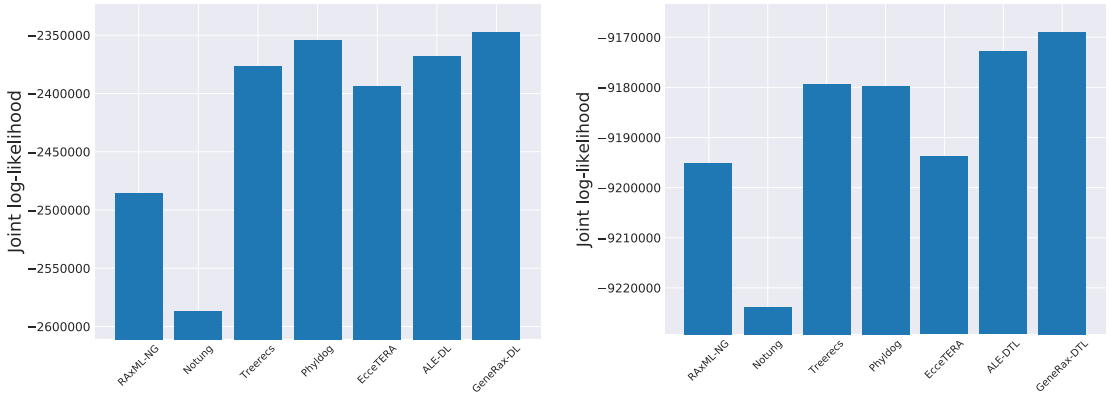
Figure 6.8: Branch score distance to true trees.. We excluded from the plot methods that do not infer the branch lengths.

and (f)), and perform worse with increasing discordance between the species tree and the GFTs (Figure 6.5 (c), (d) and (e)).

We show the results of the SimPhy simulations over varying ILS discordance scores in Figure 6.7. GeneRax outperforms all other STA tools. It finds better GFTs than the only non-STA method (RAxML-NG) up to an ILS discordance score of 0.6. Our findings suggest that GeneRax can be deployed for analyzing datasets that exhibit a moderate degree of ILS.

6.4.2 Branch score distances to true trees

To compare the quality of the gene branch lengths in terms of expected number of substitutions per site, we measured the average branch score distance [78] between the inferred trees and the true trees (Fig. (6.8) with the phangorn R library [129]. GeneRax performs better than all competing tools. In particular, GeneRax shows a better average branch score distance (1.02) than ALE (1.48). A possible explanation for this is that ALE does not infer the branch lengths by optimizing the



(a) Primates **(b)** Cyanobacteria

Figure 6.9: Log-likelihoods evaluated with GeneRax. When evaluating the joint likelihood for Primates, we set the HGT rate to 0.

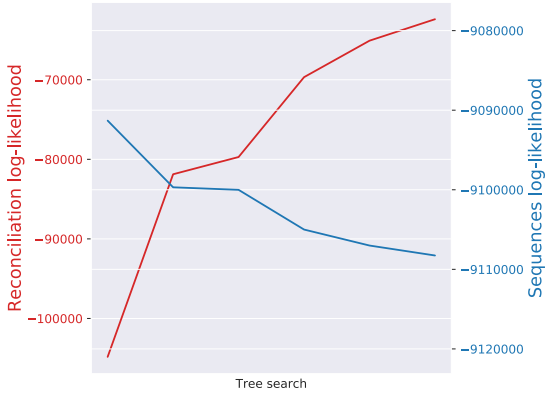


Figure 6.10: Reconciliation and sequence log-likelihoods during GeneRax tree search on the Cyanobacteria dataset.. The sequence likelihood decreases while the reconciliation likelihood increases.

phylogenetic likelihood score, as opposed to GeneRax, Treerecs, and RAxML-NG. When using ALE, NOTUNG, PHYLDOG, or eccetera, users interested in branch length accuracy would need to include an additional tool into their pipeline (e.g., RAxML-NG).

6.4.3 Joint likelihood

We report the joint maximum likelihood scores of the GFTs obtained with the different tools in Figure 6.9. As the true tree is generally not know for empirical

data, and given that we are willing to accept the maximum likelihood criterion, we must assume that the tree yielding the best joint maximum likelihood is also the one that best explains the data. This approach of benchmarking ML tools on empirical datasets has been used repeatedly for assessing standard tree inference tools [76, 110]. The rationale for this is that standard tree searches based on the phylogenetic likelihood are inherently more difficult on empirical than on smooth and perfect simulated data. That is, differences between tree search algorithms might sometimes only be observable on empirical data. As expected, GeneRax finds the highest joint likelihood score. ALE is close to GeneRax, because it strives to approximate the same model. As the remaining tools implement distinct models, our comparison might appear as being unfair. However, we mainly regard this as a means of verifying that GeneRax properly maximizes the likelihood under its specific reconciliation model. Treerecs, PHYLDOG are also very close to GeneRax in absence of transfers, because they deploy a similar joint likelihood model. ALE performs better than Treerecs and PHYLDOG in presence of HGT, because Treerecs and PHYLDOG only account for gene duplication and loss. RAxML-NG, eccetera, and NOTUNG do not implement a joint reconciliation likelihood model, which explains their low scores.

In addition, when running GeneRax on the empirical Cyanobacteria dataset, we recorded both, the reconciliation likelihood and the phylogenetic likelihood during the tree search (Figure 6.10). We observe that the joint likelihood optimization occurs through an increase of the reconciliation likelihood in conjunction with a decrease of the phylogenetic likelihood. We observed this consistently on all simulated and empirical datasets we experimented with. In general, we observed that our joint likelihood tree search heuristic is not efficient in improving the phylogenetic likelihood score, and thus needs to start from trees with a high phylogenetic likelihood. For this reason, when the user does not provide a starting tree, we initially only optimize the phylogenetic likelihood, and only subsequently start the joint likelihood optimization.

6.4.4 Sequential runtimes

We measured the sequential runtimes of all tools on the empirical Cyanobacteria dataset. Comparing runtimes is not straightforward: some tools are very fast, but require an external pre-processing step, as described in Table 6.2. For instance, NOTUNG is the fastest tool, but it requires GFTs with support values as input, and obtaining those can be extremely time-consuming. For a fair comparison, we plot both the time spent in the GFT inference tools alone, and the time spent in their respective pre-processing steps (Figure 6.11).

When only considering the stand-alone runtimes of the tools, GeneRax is the slowest method. However, when including the pre-processing cost, GeneRax becomes the fastest STA approach. In addition, using only a single tool for the entire inference process substantially improves usability and reproducibility of the analyses.

6.4.5 Parallel efficiency

We measured the parallel runtimes of GeneRax for different numbers of cores. For this experiment, we executed GeneRax on the empirical Cyanobacteria dataset

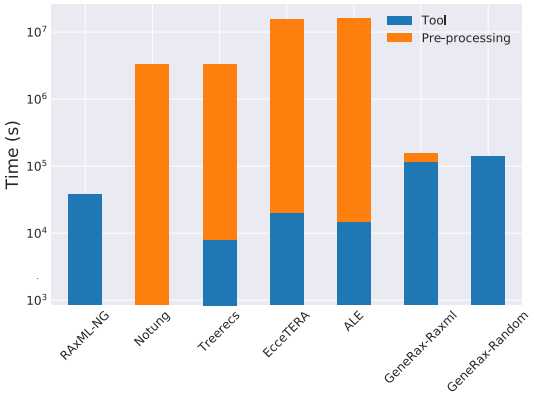


Figure 6.11: Sequential runtimes and additional overhead from precomputation steps. The precomputation steps include the bootstrap trees with RAxML-NG for NOTUNG and TREERECS, the MCMC samples with MRBAYES for ALE and ECCETERA, and the RAxML-NG starting trees for GeneRax-raxml). The RAxML-NG column corresponds to the time spent in one single tree search. We represent times with a logarithmic scale.

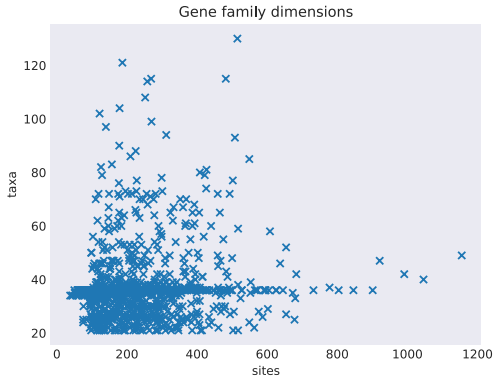


Figure 6.12: Gene family dimension in the cyanobacteria dataset. Each point is a gene family. The number of sites is the number of unique sites (we do not count duplicates).

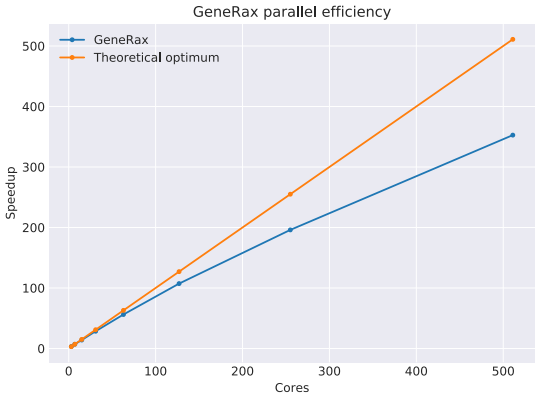


Figure 6.13: Parallel speedup of GeneRax. Parallel speedup of GeneRax on the empirical Cyanobacteria dataset (1099 families), using from 4 to 512 cores.

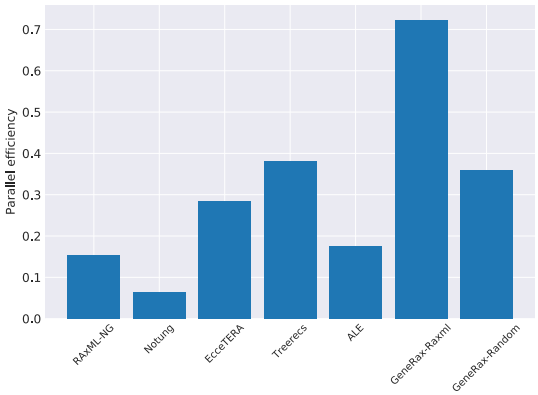


Figure 6.14: Parallel efficiency of the different methods. The parallel efficiency was computed using the cyanobacteria empirical dataset and 512 CPU cores. We do not include pre-processing steps.

(1099 families), starting from RAxML-NG trees. We used 4 up to 512 cores. Despite the highly heterogeneous gene family MSA sizes (in terms of both number of sites and number of taxa, see Figure 6.12), GeneRax achieves a high parallel efficiency of 70% on 512 cores. We plot the speedup as a function of the number of cores in Figure 6.13.

We also measured the parallel efficiency of running the competing methods as described in the Experiments section (Figure 6.14). GeneRax is the only tool that achieves good efficiency (70%) because it parallelizes both, over, *and* within gene families, thereby achieving a 'good' load balance. Despite a similar two-level parallelization scheme, the parallel efficiency of RAxML-NG (scheduled with ParGenes, with one starting tree per family) is below 20%. The reason for this is that ParGenes parallelizes individual tree searches over the sites whereas GeneRax parallelizes them over the SPR moves. Gene MSAs are often short, and there is typically not a sufficient number of sites to allocate several cores per tree search with RAxML-NG. Other competing tools also fail to attain good parallel efficiency (40%), because they do not parallelize individual GFT inferences, and are thus limited by the longest individual per-tree inference time. The parallel efficiency of GeneRax decreases when starting from random trees, because the initial phylogenetic likelihood optimization step is based on RAxML-NG code, which does not implement our aforementioned two-level parallelization scheme yet.

6.5 Discussion

6.5.1 An accurate, robust and fast approach

We present GeneRax, an open source STA GFT inference software. GeneRax can simultaneously account for substitution and DTL events. It performs a tree search to optimize a joint likelihood, that is, the product of the phylogenetic likelihood and the reconciliation likelihood. It can handle multiple gene families in parallel. To the best of our knowledge, GeneRax is the first STA tool that does not require any pre-processing of the MSAs. Also, it does not require any arbitrary threshold settings or parsimony weights, and it can account for HGT.

On simulated datasets, we demonstrate that GeneRax and ALE find trees that are closer to the true trees than those inferred by competing tools. We show that GeneRax can provide more accurate gene family trees even when the species tree is inaccurate and the substitution model is misspecified. Using two empirical datasets (Cyanobacteria and Primates), we confirm that GeneRax finds the best-scoring maximum likelihood trees under its specific model among the tested tools, both, with, and without HGT. Finally, we show that GeneRax is not only faster than the tested competing methods (when accounting for the computational cost of the pre-processing steps), but also has a substantially higher parallel efficiency, making it suitable for seamless large-scale analyses.

GeneRax is a production-level code. We released it on BioConda [55] to facilitate installation, and we kept its interface as simple as possible. While most competing

STA methods require input GFTs, sometimes, including additional information (e.g., support values), GeneRax can directly infer the GFTs from a set of given MSAs. This simplified analysis process reduces the number of *ad hoc* choices that users have to make: GeneRax does not require bootstrap-support thresholds, parsimony weights, MCMC convergence criteria, chain settings, proposal tuning, or priors. Reducing the number of arbitrary choices does not only yield the tool easier to run, but also substantially improves the reproducibility of the results. One could contest the parameters we used in our experiments for the pre-processing steps: Treerecs and NOTUNG *might* not need 100 bootstrap trees to obtain reliable support values. ALE and eccetera *might* not need as many MrBayes runs, chains, or generations to correctly approximate the phylogenetic likelihood. In general, it is possible to run the pre-processing steps faster than in our experiments. When running the competing methods, we tried to use the parameters that favor result quality/confidence over short runtimes, as we would have done in a real analysis.

Finally, our software GeneRax also implements SpeciesRax, our species tree inference method introduced in Section 5. Therefore, users can infer the species tree topology or root an unrooted input species tree from the GFTs that were inferred from the per-gene MSAs. It is thus possible to infer the rooted species tree as well as the reconciled GFTs from the per-gene MSAs in one single execution.

6.5.2 Limitations of GeneRax

GeneRax relies on two important assumptions: first, that the rooted species tree is known, and second, that the observed discordance between the GFTs and the species tree is mainly due to gene duplication, gene loss, and HGT events. Our experiments suggest that, when those assumptions are violated, GeneRax can only improve the quality of the GFTs up to a certain degree. In particular, users should be cautious when using GeneRax in the presence of ILS. Furthermore, GeneRax is not suitable for improving GFT topologies in the presence of hybridization. Nonetheless, GeneRax might be deployed for detecting potential hybridization events, by identifying species pairs exhibiting an "abnormally high" number of HGT events.

7. Conclusion and future work

7.1 Conclusion

In this thesis, I made several contributions to the field of computational phylogenetics. In particular, I developed several software tools that will help biologists to infer species trees and GFTs, as well as to disentangle the relationships between species and gene evolutionary histories. I have also parallelized my tools, and they can scale up to at least several thousands of gene families, thereby making large-scale phylogenetic analyses feasible on supercomputers.

First, I developed ParGenes, which allows users to infer unrooted GFTs from thousands of gene families via a *single* parallel run. ParGenes provides standard features found in phylogenetic data analysis pipelines, such as model testing, ML GFT inference from multiple starting trees, and bootstrap support value computation. It implements a novel parallelization strategy that yields high parallel efficiency, even when the per-gene MSAs dimensions vary substantially. The ParGenes output serves as a starting point for phylogenetic studies involving GFTs, for instance for inferring a species tree, or for performing GFT and species tree reconciliation.

Then, with SpeciesRax, I developed the first ML tool that can infer a rooted species tree from multiple unrooted GFTs and that explicitly models gene duplication, gene loss, and HGT events. On both, simulated, and empirical datasets, SpeciesRax is on par with, or more accurate than, its competitors. SpeciesRax infers a *rooted* topology, which represents a substantial advantage in comparison with its most accurate competitors (Astral-Pro and FastMulRFS), that can only infer *unrooted* species trees. In addition, SpeciesRax outputs quartet-based support values for each internal branch of the species tree, and estimates the species tree branch lengths in units of expected number of substitutions per site. Finally, we adapted the reconciliation likelihood function to partially account for potential missing data. This yields the inference process more robust with respect to errors in the gene family clustering as well as to incomplete gene family sampling.

Finally, GeneRax is the first ML tool for STA GFT correction that explicitly models gene duplication, gene loss, and HGT events. GeneRax exploits information from both, the input MSAs, and the input species tree, using a joint likelihood score that combines both, a model of sequence evolution, and a model of gene evolution. Under simulations, GeneRax is as accurate as ALE, and more accurate than its remaining competitors. In addition, GeneRax is the only STA GFT correction tool that does not need to pre-process the input MSAs. This will contribute to substantially simplifying and accelerating phylogenetic data analysis pipelines.

Surprisingly, although ML methods are often expected to be more computationally expensive than non-parametric methods, I have shown in Chapter 5 and Chapter 6 that both, SpeciesRax, and GeneRax, which are both based on ML tree search heuristics, are substantially faster than most of their competitors. In addition, ParGenes, GeneRax, and SpeciesRax have been parallelized using MPI, and can thus be easily deployed on distributed memory systems such as clusters (this is not the case for most of their respective competitors). Furthermore, ParGenes, SpeciesRax, and GeneRax each implement dedicated parallelization schemes that allow them to achieve good load balance.

Finally, ParGenes, SpeciesRax, and GeneRax are production-level tools, developed to facilitate phylogenetic analyses. In order to conduct a study, biologists often have to install, understand, and run many different, and complex tools. To simplify their installation, I also made those three tools available on BioConda [55], an open source package management system that has become increasingly popular in the biology community. Furthermore, ParGenes and GeneRax each replace several (e.g., strictly more than one) tools, thereby reducing the overall number of software components required to perform a study. Besides, competing tools for GFT inference, correction, and reconciliation can typically only process one gene family at a time. When dealing with thousands of gene families, biologists have to write their own custom scripts to run those tools and to parallelize them. In contrast to this, ParGenes, SpeciesRax, and GeneRax can simultaneously process multiple gene families in parallel, without any additional effort by the users.

As shown in Figure 1.1, ParGenes, SpeciesRax, and GeneRax can be used one after an other: ParGenes can infer unrooted GFTs using their respective input MSAs, SpeciesRax can infer a rooted species tree using the unrooted GFTs, and GeneRax can correct, root, and reconcile the GFTs using the MSAs and the species tree. The three tools also form a consistent pipeline in the sense that it relies on the same two models of (sequence and gene) evolution.

7.2 Future work

Despite the aforementioned advances, there are many challenges that still remain to be addressed. In the following, I outline some potential future directions of research.

First, both GeneRax and SpeciesRax could benefit from more complex models of gene evolution, as the UndatedDTL model that we introduced in this thesis

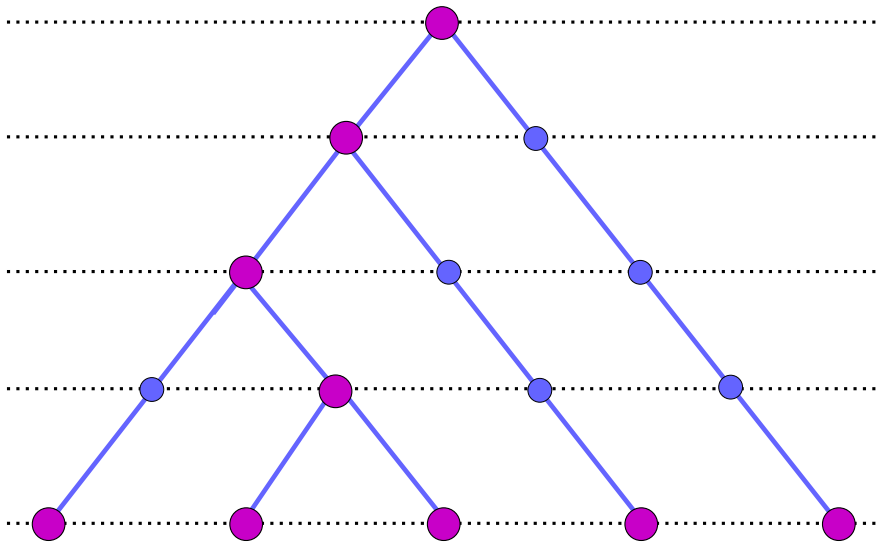


Figure 7.1: Illustration of a sliced species tree. Pink circles represent the nodes of the initial ultrametric binary species tree. For each internal node, a *slice* (horizontal dashed lines) generates one new virtual node (small blue circles) when it intersects a branch. Two branches of the resulting *sliced species tree* which are surrounded by the same two slices are said to be *contemporary*. In the *DatedDTL* model [142], only HGTs between contemporary branches are allowed.

has several limitations. First, it does not account for ILS, which is one additional major source of conflict between species and gene evolutionary histories [120]. Some promising work [23, 29, 85] has been conducted to account for both, DTL events, *and* ILS in a single model. Secondly, the UndatedDTL model does not take into account the branch length, neither in the species tree, nor in the GFTs. This leads to information loss, and furthermore allows for HGTs between non-contemporary species, which is impossible in reality. Thus, further adapting and extending the model of gene evolution might improve the accuracy of the results. For instance, one could exploit an ultrametric, dated species tree and use speciation events to slice the species tree (see Figure 7.1), as done in [142]. However, slicing the species tree increases the number of inner species nodes quadratically, and thus incurs a substantial increase in computational cost.

Secondly, SpeciesRax can currently not take into account GFT reconstruction error/uncertainty. This issue will become more prevalent with increasing taxon numbers and the associated increase in reconstruction uncertainty. Therefore, one can explore several ideas to address this limitation. A first idea consists in contracting the low-support branches of the GFTs and in adapting our reconciliation model to multifurcating GFTs. Alternatively, one can explore if co-estimating the species tree and the GFTs is feasible, as conducted by PHYLDOG [18], for instance. Finally, one could take as input a distribution of GFTs for each gene family (instead of just one maximum likelihood GFT) and integrate over this distribution of per gene family GFTs to compute the likelihood score. Such a GFTs distribution could be obtained

via Bayesian inference tools [127], from bootstrap trees [46], or from a set of plausible GFTs [105].

Finally, GeneRax does not provide any confidence measure for the inferred GFTs. Phylogenetic ML methods traditionally compute *Bootstrap support values* [46], via MSA subsampling (see Section 3.2.3). The main difficulty lies in adapting the subsampling step to GeneRax, whose input data consists of both, the species tree, and the MSA. First, because it is not clear how to subsample the species tree. One possible strategy is to randomly remove a subset of taxa from the species tree, a technique called *taxon jackknifing* [140]. GeneRax would then remove the gene sequences associated to these taxa, and apply its search algorithm using the pruned species tree and the pruned MSA, in order to compute a bootstrap GFT. A second question is, how to combine both, MSA, and species tree subsampling. One could, for instance, jackknife a given percentage p_1 of species, and subsample another percentage p_2 of MSA columns, in order to infer the bootstrap GFTs. One would then have to decide how to determine p_1 and p_2. Alternatively, instead of subsampling the input data, one could assign component weights to both, the phylogenetic, and reconciliation likelihoods, in order to control the respective contributions of the MSA and the species tree in the inference process. Support values could then be computed by inferring GFT samples for varying component weights. Via this method, users could also assess which gene bipartitions are more strongly supported by the species tree (resp. the MSA), by only taking into account those GFTs with a high phylogenetic (resp. reconciliation) likelihood component weight. This likelihood component weighting method could also be combined with the MSA and species tree subsampling method mentioned above.

Bibliography

[1] A. J. Aberer, K. Kobert, and A. Stamatakis. Exabayes: massively parallel bayesian tree inference for the whole-genome era. *Molecular Biology and Evolution*, 31(10):2553–2556, 2014.

[2] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.

[3] Ö. Åkerborg, B. Sennblad, L. Arvestad, and J. Lagergren. Simultaneous bayesian gene tree reconstruction and reconciliation analysis. *Proceedings of the National Academy of Sciences*, 106(14):5714–5719, 2009.

[4] R. Albalat and C. Cañestro. Evolution by gene loss. *Nature Reviews Genetics*, 17(7):379–391, 2016.

[5] V. A. Albert, W. B. Barbazuk, C. W. Depamphilis, J. P. Der, J. Leebens-Mack, H. Ma, J. D. Palmer, S. Rounsley, D. Sankoff, S. C. Schuster, et al. The amborella genome and the evolution of flowering plants. *Science*, 342(6165): 1241089, 2013.

[6] A. M. Altenhoff, N. M. Glover, and C. Dessimoz. Inferring orthology and paralogy. In *Methods in Molecular Biology*, pages 149–175. Springer New York, 2019.

[7] Y. Anselmetti, W. Duchemin, E. Tannier, C. Chauve, and S. Bérard. Phylogenetic signal from rearrangements in 18 Anopheles species by joint scaffolding extant and ancestral genomes. *BMC Genomics*, 19(Suppl 2), 2018.

[8] L. Arvestad et al. Bayesian gene/species tree reconciliation and orthology analysis using MCMC. *Bioinformatics*, 19(SUPPL. 1):1–10, 2003.

[9] I. Baar, L. Hübner, P. Oettig, A. Zapletal, S. Schlag, A. Stamatakis, and B. Morel. Data distribution for phylogenetic inference with site repeats via judicious hypergraph partitioning. In *2019 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, pages 175–184, 2019.

[10] M. S. Bansal, J. G. Burleigh, O. Eulenstein, and D. Fernández-Baca. Robinson-foulds supertrees. *Algorithms for Molecular Biology*, 5(1):18, 2010.

[11] P. Barbera, A. M. Kozlov, L. Czech, B. Morel, D. Darriba, T. Flouri, and A. Stamatakis. EPA-ng: Massively Parallel Evolutionary Placement of Genetic Sequences. *Systematic Biology*, 68(2):365–369, 2018.

[12] M. S. Bayzid, S. Mirarab, and T. Warnow. Inferring optimal species trees under gene duplication and loss. In *Biocomputing 2013*, pages 250–261. World Scientific, 2013.

[13] J. Bergsten. A review of long-branch attraction. *Cladistics*, 21(2):163–193, 2005.

[14] R. Betancur-R, R. E. Broughton, E. O. Wiley, K. Carpenter, J. A. López, C. Li, N. I. Holcroft, D. Arcila, M. Sanciangco, J. C. Cureton Ii, et al. The tree of life and a new classification of bony fishes. *PLoS currents*, 5, 2013.

[15] B. Bollobás and A. D. Scott. Judicious partitions of hypergraphs. *journal of combinatorial theory, Series A*, 78(1):15–31, 1997.

[16] R. Bouckaert, J. Heled, D. Kühnert, T. Vaughan, C.-H. Wu, D. Xie, M. A. Suchard, A. Rambaut, and A. J. Drummond. Beast 2: A software platform for bayesian evolutionary analysis. *PLOS Computational Biology*, 10(4):1–6, 2014.

[17] B. Boussau and C. Scornavacca. Reconciling gene trees with species trees. *Phylogenetics in the Genomic Era*, pages 3–2, 2020.

[18] B. Boussau et al. Genome-scale coestimation of species and gene trees. *Life Sciences*, pages 1–27, 2012.

[19] B. Buchfink, C. Xie, and D. H. Huson. Fast and sensitive protein alignment using DIAMOND. *Nature Methods*, 12(1):59–60, 2014.

[20] F. Burki, A. J. Roger, M. W. Brown, and A. G. Simpson. The new tree of eukaryotes. *Trends in Ecology & Evolution*, 35(1):43–55, 2020.

[21] J. G. Burleigh, M. S. Bansal, O. Eulenstein, S. Hartmann, A. Wehe, and T. J. Vision. Genome-Scale Phylogenetics: Inferring the Plant Tree of Life from 18,896 Gene Trees. *Systematic Biology*, 60(2):117–125, 2010.

[22] G. Butler, M. Rasmussen, M. Lin, S. Sakthikumar, C. Munro, E. Rheinbay, M. Grabherr, A. Forche, J. Reedy, I. Agrafioti, M. Arnaud, S. Bates, A. Brown, S. Brunke, M. Costanzo, D. Fitzpatrick, P. Groot, D. Harris, and C. Cuomo. Evolution of pathogenicity and sexual reproduction in eight candida genomes. *Nature*, 459:657–62, 2009.

[23] Y. Chan, V. Ranwez, and C. Scornavacca. Inferring incomplete lineage sorting, duplications, transfers and losses with reconciliations. *Journal of Theoretical Biology*, 432:1 – 13, 2017.

[24] H. J. Chatterjee, S. Y. Ho, I. Barnes, and C. Groves. Estimating the phylogeny and divergence times of primates using a supermatrix approach. *BMC Evolutionary Biology*, 9(1):259, 2009.

[25] K. Chen, D. Durand, and M. Farach-Colton. Notung: A program for dating gene duplications and optimizing gene family trees. *Journal of computational biology : a journal of computational molecular cell biology*, 7:429–47, 2000.

[26] S. Cheng, M. Melkonian, S. A. Smith, S. Brockington, J. M. Archibald, P.-M. Delaux, F.-W. Li, B. Melkonian, E. V. Mavrodiev, W. Sun, Y. Fu, H. Yang, D. E. Soltis, S. W. Graham, P. S. Soltis, X. Liu, X. Xu, and G. K.-S. Wong. 10KP: A phylodiverse genome sequencing plan. *GigaScience*, 7(3), 2018. giy013.

[27] N. Comte, B. Morel, D. Hasić, L. Guéguen, B. Boussau, V. Daubin, S. Penel, C. Scornavacca, M. Gouy, A. Stamatakis, E. Tannier, and D. P. Parsons. Treerecs: an integrated phylogenetic tool, from sequences to reconciliations. *Bioinformatics*, 36(18):4822–4824, 2020.

[28] A. Criscuolo and S. Gribaldo. BMGE (block mapping and gathering with entropy): a new software for selection of phylogenetic informative regions from multiple sequence alignments. *BMC Evolutionary Biology*, 10(1):210, 2010.

[29] M. D Rasmussen and M. Kellis. Unified modeling of gene duplication, loss, and coalescence using a locus tree. *Genome research*, 22:755–65, 2012.

[30] D. Darriba, T. Flouri, A. Kozlov, B. Morel, and A. Stamatakis. Pll-modules, 2019.

[31] D. Darriba, D. Posada, A. M. Kozlov, A. Stamatakis, B. Morel, and T. Flouri. ModelTest-NG: A New and Scalable Tool for the Selection of DNA and Protein Evolutionary Models. *Molecular Biology and Evolution*, 37(1):291–294, 2019.

[32] L. de Oliveira Martins and D. Posada. *Species Tree Estimation from Genome-Wide Data with guenomu*, pages 461–478. Springer New York, New York, NY, 2017.

[33] J. E. Decker, J. C. Pires, G. C. Conant, S. D. McKay, M. P. Heaton, K. Chen, A. Cooper, J. Vilkki, C. M. Seabury, A. R. Caetano, G. S. Johnson, R. A. Brenneman, O. Hanotte, L. S. Eggert, P. Wiener, J.-J. Kim, K. S. Kim, T. S. Sonstegard, C. P. Van Tassell, H. L. Neibergs, J. C. McEwan, R. Brauning, L. L. Coutinho, M. E. Babar, G. A. Wilson, M. C. McClure, M. M. Rolf, J. Kim, R. D. Schnabel, and J. F. Taylor. Resolving the evolution of extant and extinct ruminants with high-throughput phylogenomics. *Proceedings of the National Academy of Sciences*, 106(44):18644–18649, 2009.

[34] J. H. Degnan and N. A. Rosenberg. Discordance of species trees with their most likely gene trees. *PLOS Genetics*, 2(5):1–7, 2006.

[35] P. Dehal and J. L. Boore. Two rounds of whole genome duplication in the ancestral vertebrate. *PLoS Biology*, 3(10):e314, 2005.

[36] N. Dombrowski, T. A. Williams, J. Sun, B. J. Woodcroft, J.-H. Lee, B. Q. Minh, C. Rinke, and A. Spang. Undinarchaeota illuminate DPANN phylogeny and the impact of gene transfer on archaeal evolution. *Nature Communications*, 11(1), 2020.

[37] J.-P. Doyon, V. Ranwez, V. Daubin, and V. Berry. Models, algorithms and programs for phylogeny reconciliation. *Briefings in Bioinformatics*, 12(5): 392–400, 2011.

[38] W. Duchemin, G. Gence, A.-M. Arigon Chifolleau, L. Arvestad, M. S. Bansal, V. Berry, B. Boussau, F. Chevenet, N. Comte, A. A. Davín, C. Dessimoz, D. Dylus, D. Hasic, D. Mallo, R. Planel, D. Posada, C. Scornavacca, G. Szöllősi, L. Zhang, E. Tannier, and V. Daubin. RecPhyloXML: a format for reconciled gene trees. *Bioinformatics*, 34(21):3646–3652, 2018.

[39] N. El-Mabrouk and E. Noutahi. *Gene Family Evolution—An Algorithmic Framework*, pages 87–119. Springer International Publishing, 2019.

[40] D. Emms and S. Kelly. Stag: Species tree inference from all genes. *bioRxiv*, 2018.

[41] A. J. Enright. An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Research*, 30(7):1575–1584, 2002.

[42] R. I. Eytan, B. R. Evans, A. Dornburg, A. R. Lemmon, E. M. Lemmon, P. C. Wainwright, and T. J. Near. Are 100 enough? inferring acanthomorph teleost phylogeny using anchored hybrid enrichment. *BMC Evolutionary Biology*, 15 (1), 2015.

[43] P.-H. Fabre, A. Rodrigues, and E. Douzery. Patterns of macroevolution among primates inferred from a supermatrix of mitochondrial and nuclear dna. *Molecular Phylogenetics and Evolution*, 53(3):808 – 825, 2009.

[44] S. Federhen. The NCBI Taxonomy database. *Nucleic Acids Research*, 40(D1): 136–143, 2012.

[45] J. Felsenstein. Evolutionary trees from dna sequences: A maximum likelihood approach. *Journal of Molecular Evolution*, 17(6):368–376, 1981.

[46] J. Felsenstein. Confidence limits on phylogenies: an approach using the bootstrap. *Evolution*, 39(4):783–791, 1985.

[47] J. Felsenstein. *Inferring phylogenies*. Sinauer Associates, 2003.

[48] S. Feng, J. Stiller, Y. Deng, J. Armstrong, Q. Fang, A. H. Reeve, D. Xie, G. Chen, C. Guo, B. C. Faircloth, B. Petersen, Z. Wang, Q. Zhou, M. Diekhans, W. Chen, S. Andreu-Sánchez, A. Margaryan, J. T. Howard, C. Parent,

G. Pacheco, M. H. S. Sinding, L. Puetz, E. Cavill, Â. M. Ribeiro, L. Eckhart, J. Fjeldså, P. A. Hosner, R. T. Brumfield, L. Christidis, M. F. Bertelsen, T. Sicheritz-Ponten, D. T. Tietze, B. C. Robertson, G. Song, G. Borgia, S. Claramunt, I. J. Lovette, S. J. Cowen, P. Njoroge, J. P. Dumbacher, O. A. Ryder, J. Fuchs, M. Bunce, D. W. Burt, J. Cracraft, G. Meng, S. J. Hackett, P. G. Ryan, K. A. Jønsson, I. G. Jamieson, R. R. da Fonseca, E. L. Braun, P. Houde, S. Mirarab, A. Suh, B. Hansson, S. Ponnikas, H. Sigeman, M. Stervander, P. B. Frandsen, H. van der Zwan, R. van der Sluis, C. Visser, C. N. Balakrishnan, A. G. Clark, J. W. Fitzpatrick, R. Bowman, N. Chen, A. Cloutier, T. B. Sackton, S. V. Edwards, D. J. Foote, S. B. Shakya, F. H. Sheldon, A. Vignal, A. E. Soares, B. Shapiro, J. González-Solís, J. Ferrer-Obiol, J. Rozas, M. Riutort, A. Tigano, V. Friesen, L. Dalén, A. O. Urrutia, T. Székely, Y. Liu, M. G. Campana, A. Corvelo, R. C. Fleischer, K. M. Rutherford, N. J. Gemmell, N. Dussex, H. Mouritsen, N. Thiele, K. Delmore, M. Liedvogel, A. Franke, M. P. Hoeppner, O. Krone, A. M. Fudickar, B. Milá, E. D. Ketterson, A. E. Fidler, G. Friis, Á. M. Parody-Merino, P. F. Battley, M. P. Cox, N. C. B. Lima, F. Prosdocimi, T. L. Parchman, B. A. Schlinger, B. A. Loiselle, J. G. Blake, H. C. Lim, L. B. Day, M. J. Fuxjager, M. W. Baldwin, M. J. Braun, M. Wirthlin, R. B. Dikow, T. B. Ryder, G. Camenisch, L. F. Keller, J. M. DaCosta, M. E. Hauber, M. I. Louder, C. C. Witt, J. A. McGuire, J. Mudge, L. C. Megna, M. D. Carling, B. Wang, S. A. Taylor, G. Del-Rio, A. Aleixo, A. T. R. Vasconcelos, C. V. Mello, J. T. Weir, D. Haussler, Q. Li, H. Yang, J. Wang, F. Lei, C. Rahbek, M. T. P. Gilbert, G. R. Graves, E. D. Jarvis, B. Paten, and G. Zhang. Dense sampling of bird diversity increases power of comparative genomics. *Nature*, 587(7833):252–257, 2020.

[49] J. Garcia-Mas, A. Benjak, W. Sanseverino, M. Bourgeois, G. Mir, V. M. Gonzalez, E. Henaff, F. Camara, L. Cozzuto, E. Lowy, T. Alioto, S. Capella-Gutierrez, J. Blanca, J. Canizares, P. Ziarsolo, D. Gonzalez-Ibeas, L. Rodriguez-Moreno, M. Droege, L. Du, M. Alvarez-Tejado, B. Lorente-Galdos, M. Mele, L. Yang, Y. Weng, A. Navarro, T. Marques-Bonet, M. A. Aranda, F. Nuez, B. Pico, T. Gabaldon, G. Roma, R. Guigo, J. M. Casacuberta, P. Arus, and P. Puigdomenech. The genome of melon (cucumis melo l.). *Proceedings of the National Academy of Sciences*, 109(29):11872–11877, 2012.

[50] M. B. Gerstein, C. Bruce, J. S. Rozowsky, D. Zheng, J. Du, J. O. Korbel, O. Emanuelsson, Z. D. Zhang, S. Weissman, and M. Snyder. What is a gene, post-ENCODE? history and updated definition. *Genome Research*, 17(6): 669–681, 2007.

[51] G. I. Giraldo-Calderón, S. Emrich, R. M. MacCallum, G. Maslen, E. Dialynas, P. Topalis, N. Ho, S. Gesing, G. Madey, F. Collins, and D. Lawson. Vectorbase: an updated bioinformatics resource for invertebrate vectors and other organisms related with human diseases. *Nucleic Acids Research*, 43:D707 – D713, 2015.

[52] S. M. K. Glasauer and S. C. F. Neuhauss. Whole-genome duplication in teleost fishes and its evolutionary consequences. *Molecular Genetics and Genomics*, 289(6):1045–1060, 2014.

[53] S. G. Gornik, B. G. Bergheim, B. Morel, A. Stamatakis, N. S. Foulkes, and A. Guse. Photoreceptor Diversification Accompanies the Evolution of Anthozoa. *Molecular Biology and Evolution*, 2020. msaa304.

[54] R. D. Gray, Q. D. Atkinson, and S. J. Greenhill. Language evolution and human history. In *Culture Evolves*, pages 269–288. Oxford University Press, 2011.

[55] B. Grüning, R. Dale, A. Sjödin, B. Chapman, J. Rowe, C. Tomkins-Tinch, R. Valieris, J. Köster, K. Blin, M. Haudgaard, A. Kratz, A. Junge, and M. Knudsen. Bioconda: sustainable and comprehensive software distribution for the life sciences. *Nature Methods*, 15:475–476, 2018.

[56] M. W. Hahn. Bias in phylogenetic tree reconciliation methods: implications for vertebrate genome evolution. *Genome Biology*, 8(7):R141, 2007.

[57] M. W. Hahn, J. P. Demuth, and S.-G. Han. Accelerated Rate of Gene Gain and Loss in Primates. *Genetics*, 177(3):1941–1949, 2007.

[58] V. Hampl, L. Hug, J. W. Leigh, J. B. Dacks, B. F. Lang, A. G. B. Simpson, and A. J. Roger. Phylogenomic analyses support the monophyly of excavata and resolve relationships among eukaryotic "supergroups". *Proceedings of the National Academy of Sciences*, 106(10):3859–3864, 2009.

[59] B. J. Harris, C. J. Harrison, A. M. Hetherington, and T. A. Williams. Phylogenomic evidence for the monophyly of bryophytes and the reductive evolution of stomata. *Current Biology*, 30(11):2001–2012.e2, 2020.

[60] J. Helsen, K. Voordeckers, L. Vanderwaeren, T. Santermans, M. Tsontaki, K. J. Verstrepen, and R. Jelier. Gene Loss Predictably Drives Evolutionary Adaptation. *Molecular Biology and Evolution*, 37(10):2989–3002, 2020.

[61] E. B. Hodcroft, N. D. Maio, R. Lanfear, D. R. MacCannell, B. Q. Minh, H. A. Schmidt, A. Stamatakis, N. Goldman, and C. Dessimoz. Want to track pandemic variants faster? fix the bioinformatics bottleneck. *Nature*, 591(7848): 30–33, 2021.

[62] B. R. Holland, D. Penny, and M. D. Hendy. Outgroup Misplacement and Phylogenetic Inaccuracy Under a Molecular Clock—A Simulation Study. *Systematic Biology*, 52(2):229–238, 2003.

[63] K. Howe, A. Bateman, and R. Durbin. QuickTree: building huge Neighbour-Joining trees of protein sequences. *Bioinformatics*, 18(11):1546–1547, 2002.

[64] J. Huerta-Cepas, S. Capella-Gutiérrez, L. Pryszcz, M. Marcet-Houben, and T. Gabaldón. Phylomedb v4: zooming into the plurality of evolutionary histories of a genome. *Nucleic Acids Research*, 42:D897 – D902, 2014.

[65] J. Huerta-Cepas, F. Serra, and P. Bork. ETE 3: Reconstruction, Analysis, and Visualization of Phylogenomic Data. *Molecular Biology and Evolution*, 33(6): 1635–1638, 2016.

[66] L. C. Hughes, G. Ortí, Y. Huang, Y. Sun, C. C. Baldwin, A. W. Thompson, D. Arcila, R. Betancur-R., C. Li, L. Becker, N. Bellora, X. Zhao, X. Li, M. Wang, C. Fang, B. Xie, Z. Zhou, H. Huang, S. Chen, B. Venkatesh, and Q. Shi. Comprehensive phylogeny of ray-finned fishes (actinopterygii) based on transcriptomic and genomic data. *Proceedings of the National Academy of Sciences*, 115(24):6249–6254, 2018.

[67] O. T. P. T. Initiative. One thousand plant transcriptomes and the phylogenomics of green plants. *Nature*, 574(7780):679–685, 2019.

[68] A. Jain and D. Kihara. Phylo-PFP: improved automated protein function prediction using phylogenetic distance of distantly related sequences. *Bioinformatics*, 35(5):753–759, 2018.

[69] T. Jiang, P. Kearney, and M. Li. A polynomial time approximation scheme for inferring evolutionary trees from quartet topologies and its application. *SIAM J. Comput.*, 30(6):1942–1961, 2001.

[70] T. H. Jukes, C. R. Cantor, et al. Evolution of protein molecules. *Mammalian protein metabolism*, 3:21–132, 1969.

[71] S. Kalyaanamoorthy, B. Q. Minh, T. K. F. Wong, A. von Haeseler, and L. S. Jermiin. ModelFinder: fast model selection for accurate phylogenetic estimates. *Nature Methods*, 14(6):587–589, 2017.

[72] K. Katoh and D. M. Standley. MAFFT Multiple Sequence Alignment Software Version 7: Improvements in Performance and Usability. *Molecular Biology and Evolution*, 30(4):772–780, 2013.

[73] S. Kellner, A. Spang, P. Offre, G. J. Szöllosi, C. Petitjean, and T. A. Williams. Genome size evolution in the archaea. *Emerging Topics in Life Sciences*, 2(4): 595–605, 2018.

[74] K. Kobert, A. Stamatakis, and T. Flouri. Efficient Detection of Repeating Sites to Accelerate Phylogenetic Likelihood Calculations. *Systematic Biology*, 66(2):205–217, 2016.

[75] A. M. Kozlov, A. J. Aberer, and A. Stamatakis. Examl version 3: a tool for phylogenomic analyses on supercomputers. *Bioinformatics*, 31(15):2577–2579, 2015.

[76] A. M. Kozlov, D. Darriba, T. Flouri, B. Morel, and A. Stamatakis. RAxML-NG: a fast, scalable and user-friendly tool for maximum likelihood phylogenetic inference. *Bioinformatics*, 35(21):4453–4455, 2019.

[77] L. S. Kubatko and J. H. Degnan. Inconsistency of Phylogenetic Estimates from Concatenated Data under Coalescence. *Systematic Biology*, 56(1):17–24, 2007.

[78] M. K. Kuhner and J. Felsenstein. A simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates. *Molecular Biology and Evolution*, 11(3):459–468, 1994.

[79] P. Kumar, D. Velayutham, S. p k, B. Ps, A. Zachariah, A. Zachariah, C. Bathrachalam, S. S. Sajeevkumar, G. P., D. Bangarusamy, S. Iype, R. Gupta, S. Santhosh, and G. Thomas. Complete mitogenome reveals genetic divergence and phylogenetic relationships among indian cattle (bos indicus) breeds. *Animal Biotechnology*, 30:1–14, 2018.

[80] C. Lanave, G. Preparata, C. Sacone, and G. Serio. A new method for calculating evolutionary substitution rates. *Journal of molecular evolution*, 20(1):86–93, 1984.

[81] B. Larget. The estimation of tree posterior probabilities using conditional clade probability distributions. *Systematic biology*, 62, 2013.

[82] S. Q. Le and O. Gascuel. An Improved General Amino Acid Replacement Matrix. *Molecular Biology and Evolution*, 25(7):1307–1320, 2008.

[83] J. H. Leebens-Mack, M. S. Barker, E. J. Carpenter, M. K. Deyholos, M. A. Gitzendanner, S. W. Graham, I. Grosse, Z. Li, M. Melkonian, S. Mirarab, et al. One thousand plant transcriptomes and the phylogenomics of green plants. *Nature*, 574(7780):679–685, 2019.

[84] F. Leliaert, D. R. Smith, H. Moreau, M. D. Herron, H. Verbruggen, C. F. Delwiche, and O. D. Clerck. Phylogeny and molecular evolution of the green algae. *Critical Reviews in Plant Sciences*, 31(1):1–46, 2012.

[85] Q. Li, C. Scornavacca, N. Galtier, and Y.-B. Chan. The Multilocus Multispecies Coalescent: A Flexible New Model of Gene Family Evolution. *Systematic Biology*, 2020. syaa084.

[86] S. Linz et al. A likelihood framework to measure horizontal gene transfer. *Molecular Biology and Evolution*, 24(6):1312–1319, 2007.

[87] L. Liu and L. Yu. Estimating Species Trees from Unrooted Gene Trees. *Systematic Biology*, 60(5):661–667, 2011.

[88] F. Lutzoni, F. Kauff, C. Cox, D. McLaughlin, G. Celio, B. Dentinger, M. Padamsee, D. Hibbett, T. James, E. Baloch, M. Grube, V. Reeb, H. Valerie, C. Schoch, A. Arnold, J. Miadlikowska, J. Spatafora, D. Johnson, S. Hambleton,

and R. Vilgalys. Assembling the fungal tree of life: Progress, classification, and evolution of subcellular traits. *American journal of botany*, 91:1446–80, 2004.

[89] W. P. Maddison. Gene Trees in Species Trees. *Systematic Biology*, 46(3): 523–536, 1997.

[90] S. Magadum, U. Banerjee, P. Murugan, D. Gangapur, and R. Ravikesavan. Gene duplication as a major force in evolution. *Journal of Genetics*, 92(1): 155–161, 2013.

[91] J. Mallet. A species definition for the modern synthesis. *Trends in Ecology & Evolution*, 10(7):294–299, 1995.

[92] D. Mallo, L. De Oliveira Martins, and D. Posada. SimPhy : Phylogenomic Simulation of Gene, Locus, and Species Trees . *Systematic Biology*, 65(2): 334–344, 2015.

[93] M. Marcet-Houben and T. Gabaldón. The Tree versus the forest: The fungal tree of life and the topological diversity within the yeast phylome. *PLoS ONE*, 4(2), 2009.

[94] F. K. Mendes and M. W. Hahn. Why Concatenation Fails Near the Anomaly Zone. *Systematic Biology*, 67(1):158–169, 2017.

[95] K. Meusemann, M. Trautwein, F. Friedrich, R. G. Beutel, B. M. Wiegmann, A. Donath, L. Podsiadlowski, M. Petersen, O. Niehuis, C. Mayer, K. M. Bayless, S. Shin, S. Liu, O. Hlinka, B. Q. Minh, A. Kozlov, B. Morel, R. S. Peters, D. Bartel, S. Grove, X. Zhou, B. Misof, and D. K. Yeates. Are fleas highly modified mecoptera? phylogenomic resolution of antliophora (insecta: Holometabola). *bioRxiv*, 2020.

[96] A. Meyer and R. Zardoya. Recent advances in the (molecular) phylogeny of vertebrates. *Annual Review of Ecology and Systematics 34 (2003), pp. 311-338*, 34, 2003.

[97] B. Q. Minh, H. A. Schmidt, O. Chernomor, D. Schrempf, M. D. Woodhams, A. von Haeseler, and R. Lanfear. IQ-TREE 2: New Models and Efficient Methods for Phylogenetic Inference in the Genomic Era. *Molecular Biology and Evolution*, 37(5):1530–1534, 2020.

[98] S. Mirarab and T. Warnow. Astral-ii: coalescent-based species tree estimation with many hundreds of taxa and thousands of genes. *Bioinformatics*, 31(12): i44–i52, 2015.

[99] S. Mirarab, R. Reaz, M. Bayzid, T. Zimmermann, M. S Swenson, and T. Warnow. Astral: Genome-scale coalescent-based species tree estimation. *Bioinformatics (Oxford, England)*, 30:i541–i548, 2014.

[100] T. Miyashita, M. I. Coates, R. Farrar, P. Larson, P. L. Manning, R. A. Wogelius, N. P. Edwards, J. Anné, U. Bergmann, A. R. Palmer, and P. J. Currie. Hagfish from the cretaceous tethys sea and a reconciliation of the morphological–molecular conflict in early vertebrate phylogeny. *Proceedings of the National Academy of Sciences*, 116(6):2146–2151, 2019.

[101] E. K. Molloy and T. Warnow. FastMulRFS: fast and accurate species tree estimation under generic gene duplication and loss models. *Bioinformatics*, 36 (Supplement 1):i57–i65, 2020.

[102] B. Morel, T. Flouri, and A. Stamatakis. A novel heuristic for data distribution in massively parallel phylogenetic inference using site repeats. In *2017 IEEE 19th International Conference on High Performance Computing and Communications; IEEE 15th International Conference on Smart City; IEEE 3rd International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pages 81–88, 2017.

[103] B. Morel, A. M. Kozlov, and A. Stamatakis. ParGenes: a tool for massively parallel model selection and phylogenetic tree inference on thousands of genes. *Bioinformatics*, 2018.

[104] B. Morel, P. Barbera, L. Czech, B. Bettisworth, L. Hübner, S. Lutteropp, D. Serdari, E.-G. Kostaki, I. Mamais, A. M. Kozlov, P. Pavlidis, D. Paraskevis, and A. Stamatakis. Phylogenetic Analysis of SARS-CoV-2 Data Is Difficult. *Molecular Biology and Evolution*, 38(5):1777–1791, 2020.

[105] B. Morel, A. M. Kozlov, A. Stamatakis, and G. J. Szöllősi. GeneRax: A Tool for Species-Tree-Aware Maximum Likelihood-Based Gene Family Tree Inference under Gene Duplication, Transfer, and Loss. *Molecular Biology and Evolution*, 37(9):2763–2774, 2020.

[106] B. Morel, P. Schade, S. Lutteropp, T. A. Williams, G. J. Szöllősi, and A. Stamatakis. Speciesrax: A tool for maximum likelihood species tree inference from gene family trees under duplication, transfer, and loss. *bioRxiv*, 2021.

[107] Z. Musilova, F. Cortesi, M. Matschiner, W. I. L. Davies, J. S. Patel, S. M. Stieb, F. de Busserolles, M. Malmstrøm, O. K. Tørresen, C. J. Brown, J. K. Mountford, R. Hanel, D. L. Stenkamp, K. S. Jakobsen, K. L. Carleton, S. Jentoft, J. Marshall, and W. Salzburger. Vision using multiple distinct rod opsins in deep-sea fishes. *Science*, 364(6440):588–592, 2019.

[108] L. G. Nagy and G. Szöllősi. Chapter two - fungal phylogeny in the age of genomics: Insights into phylogenetic inference from genome-scale datasets. In J. P. Townsend and Z. Wang, editors, *Fungal Phylogenetics and Phylogenomics*, volume 100 of *Advances in Genetics*, pages 49–72. Academic Press, 2017.

[109] T. J. Near, A. Dornburg, R. I. Eytan, B. P. Keck, W. L. Smith, K. L. Kuhn, J. A. Moore, S. A. Price, F. T. Burbrink, M. Friedman, and P. C. Wainwright.

Phylogeny and tempo of diversification in the superradiation of spiny-rayed fishes. *Proceedings of the National Academy of Sciences*, 110(31):12738–12743, 2013.

[110] L.-T. Nguyen et al. Iq-tree: A fast and effective stochastic algorithm for estimating maximum-likelihood phylogenies. *Molecular Biology and Evolution*, 32(1):268–274, 2015.

[111] E. Noutahi, M. Semeria, M. Lafond, J. Seguin, B. Boussau, L. Guéguen, N. El-Mabrouk, and E. Tannier. Efficient gene tree correction guided by genome evolution. *PLOS ONE*, 11, 2016.

[112] N. D. Pattengale, M. Alipour, O. R. Bininda-Emonds, B. M. Moret, and A. Stamatakis. How many bootstrap replicates are necessary? *Journal of Computational Biology*, 17(3):337–354, 2010.

[113] S. Penel, A.-M. Arigon, J.-F. Dufayard, A.-S. Sertier, V. Daubin, L. Duret, M. Gouy, and G. Perrière. Databases of homologous gene families for comparative genomics. *BMC Bioinformatics*, 10(6):S3, 2009.

[114] P. Perelman, W. E. Johnson, C. Roos, H. N. Seuánez, J. E. Horvath, M. A. M. Moreira, B. Kessing, J. Pontius, M. Roelke, Y. Rumpler, M. P. C. Schneider, A. Silva, S. J. O'Brien, and J. Pecon-Slattery. A molecular phylogeny of living primates. *PLOS Genetics*, 7(3):1–17, 2011.

[115] P. Portin and A. Wilkins. The Evolving Definition of the Term "Gene". *Genetics*, 205(4):1353–1364, 2017.

[116] D. Posada and K. A. Crandall. MODELTEST: Testing the model of DNA substitution. *Bioinformatics*, 14(9):817–818, 1998.

[117] M. N. Puttick, J. L. Morris, T. A. Williams, C. J. Cox, D. Edwards, P. Kenrick, S. Pressel, C. H. Wellman, H. Schneider, D. Pisani, and P. C. Donoghue. The interrelationships of land plants and the nature of the ancestral embryophyte. *Current Biology*, 28(5):733–745.e2, 2018.

[118] A. Rambaut and N. C. Grass. Seq-Gen: an application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. *Bioinformatics*, 13(3):235–238, 1997.

[119] J. A. G. Ranea, C. Yeats, A. Grant, and C. A. Orengo. Predicting protein function with hierarchical phylogenetic profiles: The gene3d phylo-tuner method applied to eukaryotic genomes. *PLOS Computational Biology*, 3(11):1–13, 2007.

[120] B. Rannala and Z. Yang. Bayes estimation of species divergence times and ancestral population sizes using dna sequences from multiple loci. *Genetics*, 164(4):1645–1656, 2003.

[121] M. Ravenhall, N. Škunca, F. Lassalle, and C. Dessimoz. Inferring horizontal gene transfer. *PLOS Computational Biology*, 11(5):1–16, 2015.

[122] K. Raymann, C. Brochier-Armanet, and S. Gribaldo. The two-domain tree of life is linked to a new root for the archaea. *Proceedings of the National Academy of Sciences*, 112(21):6670–6675, 2015.

[123] D. Robinson and L. Foulds. Comparison of phylogenetic trees. *Mathematical Biosciences*, 53(1):131 – 147, 1981.

[124] 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.

[125] F. Rodriguez, J. L. Oliver, A. Marín, and J. R. Medina. The general stochastic model of nucleotide substitution. *Journal of theoretical biology*, 142(4):485–501, 1990.

[126] A. J. Roger, S. A. Muñoz-Gómez, and R. Kamikawa. The origin and diversification of mitochondria. *Current Biology*, 27(21):R1177–R1192, 2017.

[127] F. Ronquist, M. Teslenko, P. van der Mark, D. L. Ayres, A. Darling, S. Höhna, B. Larget, L. Liu, M. A. Suchard, and J. P. Huelsenbeck. MrBayes 3.2: Efficient Bayesian Phylogenetic Inference and Model Choice Across a Large Model Space. *Systematic Biology*, 61(3):539–542, 2012.

[128] N. Saitou and M. Nei. The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular biology and evolution*, 4(4):406–425, 1987.

[129] K. P. Schliep. phangorn: phylogenetic analysis in R. *Bioinformatics*, 27(4): 592–593, 2010.

[130] E. E. Schwager, P. P. Sharma, T. Clarke, D. J. Leite, T. Wierschin, M. Pechmann, Y. Akiyama-Oda, L. Esposito, J. Bechsgaard, T. Bilde, A. D. Buffry, H. Chao, H. Dinh, H. Doddapaneni, S. Dugan, C. Eibner, C. G. Extavour, P. Funch, J. Garb, L. B. Gonzalez, V. L. Gonzalez, S. Griffiths-Jones, Y. Han, C. Hayashi, M. Hilbrant, D. S. T. Hughes, R. Janssen, S. L. Lee, I. Maeso, S. C. Murali, D. M. Muzny, R. N. da Fonseca, C. L. B. Paese, J. Qu, M. Ronshaugen, C. Schomburg, A. Schönauer, A. Stollewerk, M. Torres-Oliva, N. Turetzek, B. Vanthournout, J. H. Werren, C. Wolff, K. C. Worley, G. Bucher, R. A. Gibbs, J. Coddington, H. Oda, M. Stanke, N. A. Ayoub, N.-M. Prpic, J.-F. Flot, N. Posnien, S. Richards, and A. P. McGregor. The house spider genome reveals an ancient whole-genome duplication during arachnid evolution. *BMC Biology*, 15(1), 2017.

[131] G. Schwarz. Estimating the Dimension of a Model. *The Annals of Statistics*, 6 (2):461 – 464, 1978.

[132] C. Scornavacca, E. Jacox, and G. J. Szöllősi. Joint amalgamation of most parsimonious reconciled gene trees. *Bioinformatics*, 31(6):841–848, 2014.

[133] B. Sennblad and J. Lagergren. Probabilistic Orthology Analysis. *Systematic Biology*, 58(4):411–424, 2009.

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

[135] J. Sjöstrand, L. Arvestad, J. Lagergren, and B. Sennblad. Genphylodata: realistic simulation of gene family evolution. *BMC Bioinformatics*, 14(1):209, 2013.

[136] A. Spang, J. H. Saw, S. L. Jørgensen, K. Zaremba-Niedzwiedzka, J. Martijn, A. E. Lind, R. van Eijk, C. Schleper, L. Guy, and T. J. G. Ettema. Complex archaea that bridge the gap between prokaryotes and eukaryotes. *Nature*, 521 (7551):173–179, 2015.

[137] M. S. Springer, R. W. Meredith, J. Gatesy, C. A. Emerling, J. Park, D. L. Rabosky, T. Stadler, C. Steiner, O. A. Ryder, J. E. Janečka, C. A. Fisher, and W. J. Murphy. Macroevolutionary dynamics and historical biogeography of primate diversification inferred from a species supermatrix. *PLOS ONE*, 7(11): 1–23, 2012.

[138] A. Stamatakis. Raxml version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioiberger2010accuracynformatics*, 30(9):1312–1313, 2014.

[139] A. Stamatakis. Using RAxML to Infer Phylogenies. *Current protocols in bioinformatics*, 51:6.14.1–6.14.14, 2015.

[140] A. Stamatakis, M. Göker, and G. W. Grimm. Maximum likelihood analyses of 3, 490 rbcL sequences: Scalability of comprehensive inference versus group-specific taxon sampling. *Evolutionary Bioinformatics*, 6:EBO.S4528, 2010.

[141] Z. D. Stephens, S. Y. Lee, F. Faghri, R. H. Campbell, C. Zhai, M. J. Efron, R. Iyer, M. C. Schatz, S. Sinha, and G. E. Robinson. Big data: Astronomical or genomical? *PLOS Biology*, 13(7):e1002195, 2015.

[142] G. J. Szöllősi, B. Boussau, S. S. Abby, E. Tannier, and V. Daubin. Phylogenetic modeling of lateral gene transfer reconstructs the pattern and relative timing of speciations. *Proceedings of the National Academy of Sciences*, 109(43): 17513–17518, 2012.

[143] G. J. Szöllősi, W. Rosikiewicz, B. Boussau, E. Tannier, and V. Daubin. Efficient exploration of the space of reconciled gene trees. *Systematic Biology*, 62(6): 901–912, 2013.

[144] G. J. Szöllősi, E. Tannier, N. Lartillot, and V. Daubin. Lateral Gene Transfer from the Dead. *Systematic Biology*, 62(3):386–397, 2013.

[145] G. J. Szöllősi, E. Tannier, V. Daubin, and B. Boussau. The Inference of Gene Trees with Species Trees. *Systematic Biology*, 64(1):e42–e62, 2014.

[146] N. Takezaki, F. Figueroa, Z. Zaleska-Rutczynska, and J. Klein. Molecular Phylogeny of Early Vertebrates: Monophyly of the Agnathans as Revealed by Sequences of 35 Genes. *Molecular Biology and Evolution*, 20(2):287–292, 2003.

[147] S. Tavaré et al. Some probabilistic and statistical problems in the analysis of dna sequences. *Lectures on mathematics in the life sciences*, 17(2):57–86, 1986.

[148] A. R. Templeton. The meaning of species and speciation: a genetic perspective. *The units of evolution: Essays on the nature of species*, 1992:159–183, 1989.

[149] M. Touchon, C. Hoede, O. Tenaillon, V. Barbe, S. Baeriswyl, P. Bidet, E. Bingen, S. Bonacorsi, C. Bouchier, O. Bouvet, A. Calteau, H. Chiapello, O. Clermont, S. Cruveiller, A. Danchin, D. Médéric, C. Dossat, M. El Karoui, E. Frapy, and E. Denamur. Organised genome dynamics in the escherichia coli species results in highly diverse adaptive paths. *PLoS genetics*, 5:e1000344, 2009.

[150] B. Venkatesh, A. P. Lee, V. Ravi, A. K. Maurya, M. M. Lian, J. B. Swann, Y. Ohta, M. F. Flajnik, Y. Sutoh, M. Kasahara, S. Hoon, V. Gangu, S. W. Roy, M. Irimia, V. Korzh, I. Kondrychyn, Z. W. Lim, B. H. Tay, S. Tohari, K. W. Kong, S. Ho, B. Lorente-Galdos, J. Quilez, T. Marques-Bonet, B. J. Raney, P. W. Ingham, A. Tay, L. W. Hillier, P. Minx, T. Boehm, R. K. Wilson, S. Brenner, and W. C. Warren. Elephant shark genome provides unique insights into gnathostome evolution. *Nature*, 505(7482):174–179, 2014.

[151] A. Wehe, M. S. Bansal, J. G. Burleigh, and O. Eulenstein. DupTree: a program for large-scale phylogenetic analyses using gene tree parsimony. *Bioinformatics*, 24(13):1540–1541, 2008.

[152] D. Wen, Y. Yu, J. Zhu, and L. Nakhleh. Inferring Phylogenetic Networks Using PhyloNet. *Systematic Biology*, 67(4):735–740, 2018.

[153] T. J. Wheeler. Large-scale neighbor-joining with ninja. In S. L. Salzberg and T. Warnow, editors, *Algorithms in Bioinformatics*, pages 375–389, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[154] S. Whelan and N. Goldman. A General Empirical Model of Protein Evolution Derived from Multiple Protein Families Using a Maximum-Likelihood Approach. *Molecular Biology and Evolution*, 18(5):691–699, 2001.

[155] M. Wilkinson, J. O. McInerney, R. P. Hirt, P. G. Foster, and T. M. Embley. Of clades and clans: terms for phylogenetic relationships in unrooted trees. *Trends in Ecology & Evolution*, 22(3):114–115, 2007.

[156] T. A. Williams and T. M. Embley. Archaeal "Dark Matter" and the Origin of Eukaryotes. *Genome Biology and Evolution*, 6(3):474–481, 2014.

[157] T. A. Williams, G. J. Szöllősi, A. Spang, P. G. Foster, S. E. Heaps, B. Boussau, T. J. G. Ettema, and T. M. Embley. Integrative modeling of gene and genome evolution roots the archaeal tree of life. *Proceedings of the National Academy of Sciences*, 114(23):E4602–E4611, 2017.

[158] T. A. Williams, C. J. Cox, P. G. Foster, G. J. Szöllősi, and T. M. Embley. Phylogenomics provides robust support for a two-domains tree of life. *Nature Ecology & Evolution*, 4(1):138–147, 2020.

[159] Z. Xi, L. Liu, and C. C. Davis. The Impact of Missing Data on Species Tree Estimation. *Molecular Biology and Evolution*, 33(3):838–860, 2015.

[160] Z. Yang. Maximum-likelihood estimation of phylogeny from dna sequences when substitution rates differ over sites. *Molecular biology and evolution*, 10 (6):1396–1401, 1993.

[161] Z. Yang. Statistical properties of the maximum likelihood method of phylogenetic estimation and comparison with distance matrix methods. *Systematic Biology*, 43(3):329–342, 1994.

[162] Z. Yang. A space-time process model for the evolution of dna sequences. *Genetics*, 139(2):993–1005, 1995.

[163] K. Zaremba-Niedzwiedzka, E. F. Caceres, J. H. Saw, D. Bäckström, L. Juzokaite, E. Vancaester, K. W. Seitz, K. Anantharaman, P. Starnawski, K. U. Kjeldsen, M. B. Stott, T. Nunoura, J. F. Banfield, A. Schramm, B. J. Baker, A. Spang, and T. J. G. Ettema. Asgard archaea illuminate the origin of eukaryotic cellular complexity. *Nature*, 541(7637):353–358, 2017.

[164] D. R. Zerbino et al. Ensembl 2018. *NAR*, 46(D1):D754–D761, 2018.

[165] C. Zhang, M. Rabiee, E. Sayyari, and S. Mirarab. ASTRAL-III: polynomial time species tree reconstruction from partially resolved gene trees. *BMC Bioinformatics*, 19(S6), 2018.

[166] C. Zhang, C. Scornavacca, E. K. Molloy, and S. Mirarab. Astral-pro: quartet-based species tree inference despite paralogy. *bioRxiv*, 2019.

[167] J. Zhang. Evolution by gene duplication: an update. *Trends in Ecology & Evolution*, 18(6):292–298, 2003.

[168] Y. Zheng and L. Zhang. Effect of incomplete lineage sorting on tree-reconciliation-based inference of gene duplication. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 11(3):477–485, 2014.

[169] X. Zhou, S. Lutteropp, L. Czech, A. Stamatakis, M. V. Looz, and A. Rokas. Quartet-Based Computations of Internode Certainty Provide Robust Measures of Phylogenetic Incongruence. *Systematic Biology*, 69(2):308–324, 2019.