Université de Montréal

Edit Distance Metrics for Measuring Dissimilarity between Labeled Gene Trees

par

Samuel Briand

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Mémoire présenté en vue de l'obtention du grade de
Maître ès sciences (M.Sc.)
en Informatique

Orientation Biologie computationelle

August 7, 2020

Université de Montréal

Faculté des arts et des sciences

Ce mémoire intitulé

Edit Distance Metrics for Measuring Dissimilarity between Labeled Gene Trees

présenté par

Samuel Briand

a été évalué par un jury composé des personnes suivantes :

Sylvie Hamel
(président-rapporteur)

Nadia El-Mabrouk
(directeur de recherche)

Esma Aïmeur
(membre du jury)

Résumé

Les arbres phylogénétiques sont des instruments de biologie évolutive offrant de formidables moyens d'étude pour la génomique comparative. Ils fournissent des moyens de représenter des mécanismes permettant de modéliser les relations de parenté entre les espèces ou les membres de familles de gènes en fonction de la diversité taxonomique, ainsi que des observations et des renseignements sur l'histoire évolutive, la structure et la variation des processus biologiques. Cependant, les méthodes traditionnelles d'inférence phylogénétique ont la réputation d'être sensibles aux erreurs. Il est donc indispensable de comparer les arbres phylogénétiques et de les analyser pour obtenir la meilleure interprétation des données biologiques qu'ils peuvent fournir. Nous commençons par aborder les travaux connexes existants pour déduire, comparer et analyser les arbres phylogénétiques, en évaluant leurs bonnes caractéristiques ainsi que leurs défauts, et discuter des pistes d'améliorations futures. La deuxième partie de cette thèse se concentre sur le développement de mesures efficaces et précises pour analyser et comparer des paires d'arbres génétiques avec des nœuds internes étiquetés. Nous montrons que notre extension de la métrique bien connue de Robinson-Foulds donne lieu à une bonne métrique pour la comparaison d'arbres génétiques étiquetés sous divers modèles évolutifs, et qui peuvent impliquer divers événements évolutifs.

Mots clés : évolution ; distance d'édition ; arbre génétique ; arbre étiqueté ; Robinson-Foulds ; métrique d'arbre; histoire de l'évolution

Abstract

Phylogenetic trees are instruments of evolutionary biology offering great insight for comparative genomics. They provide mechanisms to model the kinship relations between species or members of gene families as a function of taxonomic diversity. They also provide evidence and insights into the evolutionary history, structure, and variation of biological processes. However, traditional phylogenetic inference methods have the reputation to be prone to errors. Therefore, comparing and analysing phylogenetic trees is indispensable for obtaining the best interpretation of the biological information they can provide. We start by assessing existing related work to infer, compare, and analyse phylogenetic trees, evaluating their advantageous traits and flaws, and discussing avenues for future improvements. The second part of this thesis focuses on the development of efficient and accurate metrics to analyse and compare pairs of gene trees with labeled internal nodes. We show that our attempt in extending the popular Robinson-Foulds metric is useful for the preliminary analysis and comparison of labeled gene trees under various evolutionary models that may involve various evolutionary events.

Keywords: evolution; edit distance; gene tree; labeled tree; Robinson-Foulds; tree metric; evolutionary history

Contents

List of tables

List of figures

Liste des sigles et des abréviations

CM	Cluster mapping
DL	Duplication Loss
DNA	Deoxyribonucleic Acid
DTL	Duplication Transfer Loss
ELRF	Edge-based Labeled Robinson-Foulds
HGT	Horizontal Gene Transfer
HP	Hierarchy-preserving
LCA	Lowest Common Ancestor
LRF	Labeled Robinson-Foulds
MCMC	Markov Chain Monte Carlo
ML	Maximum Likelihood

NNI	Nearest Neighbor Interchange
RF	Robinson-Foulds
RNA	Ribonucleic Acid
SPR	Subtree Pruning ad Regrafting
TBR	Tree Bisection Reconnection
TED	Tree Edit Distance
UPGMA	Unweighted Pair Group Method with Arithmetic Mean

Remerciements

Cette thèse n'aurait pas vu le jour sans la participation et le soutien de nombreuses personnes, et je tiens à leur exprimer ma gratitude.

Je tiens à remercier avant tout ma directrice de recherche, Nadia El-Mabrouk, pour ses conseils, sa disponibilité tout au long de mon séjour au LBIT, son soutien moral et financier, ainsi que son dévouement pour m'impliquer dans la communauté de la recherche. Ce fut un réel plaisir de travailler sous votre supervision.

Je tiens également à remercier mes parents du fond du cœur, pour leur soutien, leurs conseils et leur présence tout au long de mes études, mais surtout pour toutes les opportunités qu'ils m'ont données et qui m'ont permis de devenir ce que je suis aujourd'hui.

Je tiens également à remercier Christophe Dessimoz pour m'avoir donné l'occasion de collaborer sur mes premiers articles scientifiques.

Merci à Anna, Robin, Mathieu et Miguel sans qui, mon séjour au LBIT n'aurait pas été une expérience aussi enrichissante.

Je voudrais également remercier Houari Sahraoui et Céline Bégin, qui m'ont aidé à trouver mon chemin et à prendre les bonnes décisions pendant le déroulement de cette maîtrise.

Enfin, j'aimerais remercier mes amis (surtout les Pals) sans qui je serais probablement tombé du wagon de ces montagnes russes émotionnelles.

Chapter 1

Introduction

In the study of biological entities (molecular sequences, genomes, or species), phylogenetics focuses on the inquiry of the evolutionary history and relationships among those entities. Phylogenetic trees are commonly used, in the context of the study of species, to provide a mechanism to model the kinship relations between species or taxa as a function of taxonomic diversity. Regarding the study of genes, they provide evidence or insight on the evolutionary history of gene families or groups of genes. A tree enables a representation of the degree of morphological or genetic divergence by branching from points of connection (internal nodes). The growing application of phylogenetic trees led to an increasing need to devise techniques to compare them. Most particularly, the ability to analyze the differences, similarities, and distance between phylogenetic trees is key in the field of computational phylogenetics to be able to perform certain tasks. These include, for example, the study of tree space, the evaluation of phylogenetic reconstruction, and the appraisal of consistency from tree topologies inferred by reconciliation, a method for inferring the evolutionary scenario for a gene family by embedding an inferred gene tree into a known species tree.

The need for rigorous comparisons entails the ability to measure phylogenetic trees and thus the development of tools relying on metrics. Mathematically speaking, a metric is a function that defines a distance between a pair of elements in a set. A metric gives structure and shape to a set of objects [**42**] and enables various analyses such as automated clustering. A measure of distance is said to be a metric if it satisfies the non-negative, symmetric, identity, and triangular inequality conditions.

A distance, when addressing evolutionary trees is, to a certain extent, a particular representation of the differences and similarities between compared trees. There exist many distance measures to compare phylogenetic trees but not all of them satisfy the conditions of a metric. In other words, not all of them induce a metric on the phylogenetic tree space. Further, all existing metrics have weaknesses and limitations, such as high computational complexities or poor statistical distributions (e.g., high skewness, low variance), which limit

their applicability. It is why, in practice, using an appropriate metric for a given task is a key decision.

The research community's endeavor to develop new metrics, in their attempt to increase the scope and applicability of comparisons between phylogenetic trees, has been growing over the last 30 years. This thesis is part of the wider initiative to broaden the applicability of phylogenetic tree comparisons.

The work that we are presenting lies in the general area of phylogenetic tree dissimilarity measurement. We have developed appropriate methods and metrics to measure the distance between labeled phylogenetic trees, which in this thesis specifically refers to trees with internal nodes labeled with qualitative information.

Though this work is not the first initiative to develop a distance to compare labeled phylogenetic trees (e.g., Tree Edit Distance [**80, 78**]), it is an attempt at broadening the applicability of the type of labeling that can be used for comparing phylogenetic trees.

Our objective is to develop an appropriate distance metric to compare labeled trees with labeling that is applicable in the context of genetic data comparisons. This is motivated by the importance of being able to compare gene trees, and in particular those labeled through reconciliation, a classical method allowing to infer the evolutionary event at the source of the gene tree branch bifurcation by embedding the gene tree in the known species tree.

We provide two new distance metrics for node-labeled phylogenetic trees, along with their computation algorithms and their analysis. Both are extensions of the well-known Robinson-Foulds distance (RF) [**58**].

The first metric is an edit distance that counts the total number of edit operations to transform one tree into the other. It is a costly algorithm in terms of computation time as it requires to actually modify the compared trees to obtain a result. This algorithm is not exact but guarantees an approximation ratio of 2. A python package was developed to implement the algorithm and perform experiments. Experimental results have shown that the proposed metric better reflects differences among labeled trees than RF. However, it shares the same skewed and low-variance distributions as the RF distance.

The second metric is an attempt at addressing some of the weaknesses of the first one. It is also an edit distance that computes the total number of edit operations to transform one tree into another. However, it is an exact, more efficient algorithm in terms of computational complexity. In fact, this distance between two trees can be computed in linear time, without requiring the actual transformations to change one tree into the other. This algorithm was also implemented in Python. Experimental results show that this metric is much faster to compute than the first one. Still, it shares the same skewed and low-variance distributions.

The general structure of the memoir is depicted hereafter. Chapter 2 contextualises the subject at hand. Chapter 3 reviews related work and provides motivations for our work.

Chapters 4 and 5 present our contributions, in the form of articles. Chapter 6 concludes the thesis.

Chapter 2 introduces the basic concepts in biology required for a proper understanding of the thesis, with motivations for the application of phylogenetic trees. In this chapter, we also present the basic notions related to phylogenetic trees, and introduce the concept of tree reconstruction and some of its well-known issues. Finally, we discuss the role of reconciliation, and its relation to our motivation to develop new metrics.

In chapter 3, we first present representative tree metrics and how they are used on different types of phylogenetic trees. We then review their strengths and weaknesses, and compare them to determine the situations where they are most applicable.

Chapter 4 contains the article titled *"A Generalized Robinson-Foulds Distance for Labeled Trees"* accepted for publication in *BMC Genomics*. We introduce a new metric which accounts for the labels of internal nodes in compared phylogenetic trees when computing their dissimilarity. This new metric is an extension of an edit distance, the well-known RF distance. This extension adds an edit operation to the set of operations of the RF distance that alters internal node labels, which indirectly enables us to account for them in the distance computation.

Chapter 5 introduces another attempt at computing distances for labeled phylogenetic trees, in the article titled *"A Linear Time Solution to the Labeled Robinson-Foulds Distance Problem"*. The goal is to improve efficiency and accuracy of distance measurement over the solution presented in Chapter 4. This new metric is a less constrained version of the previous extension of the RF distance as its edit operations have fewer applicability requirements, and as it allows for an arbitrary number of potential labels.

The last chapter will summarize the strengths and limitations of the newly introduced metrics and will then outline new directions for future work.

Chapter 2

Phylogenetic Trees: Background and Applications

The study of evolution has, since its outset as a branch of biology, led to the development of a wide array of tools and methods to analyze and understand the mechanisms of evolution, that have induced the diversity of life on our planet. In this chapter, we present the concepts needed to understand the following chapters of this thesis.

In section 2.1 we introduce the concepts related to phylogenetic trees and briefly discuss common types of trees used in phylogenetics.

Section 2.3 introduces the concept of a gene tree, describes the events leading to the creation of gene families over time, and explains the importance of inferring relationships between genes.

Section 2.4 presents the notion of phylogenetic tree inference and goes over popular methods of tree reconstruction. Additionally, it discusses the challenges due to errors and inconsistencies between the evolutionary trees inferred by different methods.

Finally, section 2.5 focuses on reconciliation between species trees and gene trees. First, we explain the incentives behind reconciliation and how reconciliation can be used to infer evolutionary events from gene families' and species trees. Moreover, we clarify how tree building errors in species trees or gene trees affect reconciled trees.

2.1. Phylogenetic Trees

While the first concepts of phylogenetic trees date back to the early 19^{th} century with the notion of *tree of life*, phylogenetic trees were popularized with the notion of *evolutionary tree* by Darwin [**18**]. Evolutionary trees are now tools that have been used to illustrate kinship relations between biological entities for more than a century.

Formerly, due to technological limitations, only a limited set of observed heritable traits could be used to depict phylogeny, such as morphological traits, as illustrated by a simple

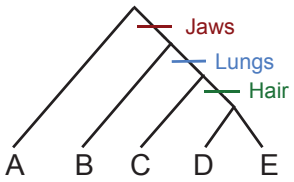


Fig. 2.1. An example of a simple species tree depicting the phylogeny of the taxa A, B, C, D, and E based on the presence/absence of three morphological traits: jaws, lungs, and hair.

example in figure 2.1. Even though the resulting species trees using these methods were a step in the right direction, since a morphological trait may be a defining aspect of a taxon, i.e., an evolutionary grade, they could also lead to erroneous conclusions, as similar looking morphological traits may have different origins in evolutionary history, e.g., "wings" in birds and bats, called homoplasies.

One of the great breakthroughs in phylogenetics is the discovery of the Deoxyribonucleic acid (DNA) during the 20^{th} century. This led to a significant change in phylogenetics, as genes became much more than discrete inheritable units. DNA sequences became observable heritable traits for phylogenetic inference. Today we can compute a species tree from genetic data. Moreover, we can compute gene trees, which infer the phylogeny of genes, with data from alleles of one or a few genes. Gene trees can help infer scenarios of species evolution and are commonly used to infer species trees. As the focus of this thesis is on gene trees, we further discuss the topic in section 2.3.

2.2. Notations and Concepts

A *tree* T is defined as an *undirected acyclic connected graph*. We define as $V(T)$ the set of nodes of T, $E(T)$ as the set of edges of T, and $L(T) \subset V(T)$ as the set of leaves of T. For an arbitrary node $x \in V(T)$, we define the *degree* of x as the number of edges incident to x. The leaves of T are all of degree one as each leaf is only incident to a single edge. The set of leaves $L(T)$ represents a set of biological entities $\mathcal{L}$, i.e., $L(T) = \mathcal{L}$. The size of a tree T is defined by the size of its set of nodes $V(T)$, i.e., $|V(T)|$. Moreover we define as *internal* nodes the set of nodes of T excluding leaves, i.e., $I(T) = V(T) \backslash L(T)$.

We define an edge of T which connects two nodes x and y by $e = (x,y)$. If $x,y \in I(T)$ then e is an internal edge, otherwise it is a terminal edge. We say that a tree T is *rooted* if T contains a specific node $r(T)$ called the *root*. The root of a tree represents the ancestral lineage of the biological entities represented by the tree and, further, indicates the direction of evolution. When a tree T is rooted, given two nodes x and y in $V(T)$, if x is on the unique path connecting $r(T)$ and y, then we say that y is a *descendant* of x, and that x is an *ancestor* of y. Additionally, if x and y further form an edge $e = (x,y)$, then we say that y is a *child* of x, written $y \in Ch(x)$, and that x is the *parent* of y, written $p(y)$. When x

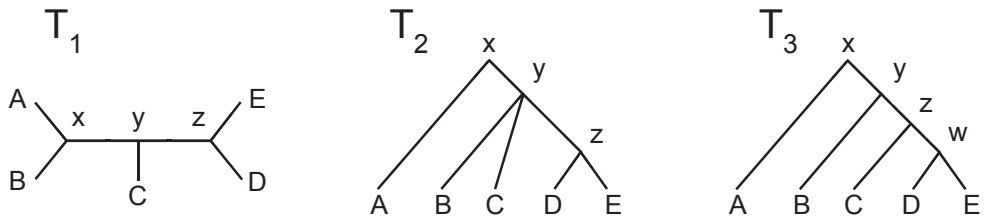


Fig. 2.2. The trees T_1, T_2, and T_3 all have the same set of leaves $\mathcal{L} = \{A, B, C, D, E\}$. T_1 is an unrooted binary tree. T_2 is a rooted non-binary tree. T_3 is a rooted binary tree. It can be observed that T_1 and T_3 both satisfy the requirements of binary trees as their internal nodes, terminal nodes, and root ($x \in T_3$) conform to the expected degrees 3, 1, and 2, respectively. Additionally, T_2 and T_3 may look similar but they indicate significant evolutionary differences, e.g., suppose $\mathcal{N} = \{C, E\}$, then $lca_{T_2}(\mathcal{N}) \neq lca_{T_3}(\mathcal{N})$.

is an ancestor of y, we denote it $x \geq y$, and conversely $x \leq y$ when x is a descendant of y. In the case where x is neither the ancestor nor the descendant of y, we say that they are incomparable.

For a tree T, we define by T_x the *subtree* of T rooted at x such that $V(T_x) \subseteq V(T)$, $E(T_x) \subseteq E(T)$, and $L(T_x) \subseteq L(T)$. T_x contains all descendants of x. In a rooted tree T, the *lowest common ancestor (LCA)* of a set of nodes $\mathcal{N}$ in $V(T)$ is the ancestor of all the nodes in $\mathcal{N}$ furthest from the root $r(T)$. We denote the LCA of a set of nodes $\mathcal{N}$ by $lca_T(\mathcal{N})$. We describe a tree T as *binary* if each internal node is of degree 3 (except $r(T)$, if T is rooted, which is of degree 2). Figure 2.2 depicts observable differences between binary and non-binary trees and illustrates some of the introduced concepts.

2.3. Gene Trees

Gene trees are the result of the inference of the evolution of one or more genes. They are an important tool for the inference of species trees. Though the evolution of species and the evolution of their genes are not identical, gene evolution is one of the important factors that determines the structure and variation of biological processes within organisms, that are being passed on from one generation to the next.

2.3.1. Gene Evolution

When a gene is being passed on by an organism to its direct descendent, the gene is replicated. During that time, replication errors, i.e., mutations, can occur and modify its DNA sequence. There are multiple kinds of mutations, most of which can be classified into two categories: point mutations and chromosomal rearrangements.

Point mutations are mutations that affect one or multiple nucleotides of the sequence of a gene. This kind of mutation can modify a sequence in different ways. They include *insertions*, which add a new nucleotide within the sequence, or inversely *deletions*, which

remove a nucleotide from the sequence, as well as *substitutions* which replace a nucleotide with another. Point mutations can have different effects on the amino acid sequence being produced. They can be *synonymous* and not affect the protein being expressed or they can be *non-synonymous* and affect protein production and function, resulting in biological change in a organism.

Chromosomal rearrangements involve changes in the structure of an entire chromosome, its content and the location of genes. There exist multiple types of events that cause these rearrangements. Such events may be *gene duplication*, *gene loss*, or *horizontal gene transfer*. Gene duplication consist in the creation of a new locus, and gene loss in the degeneration of a locus [**22**]. Horizontal gene transfer (HGT) is the movement of genetic material opposite to vertical genetic data transfer, where genetic information is inherited from parents to offspring through reproduction. More specifically HGT is the transmission of genetic material from a source organism to a target organism that is not its offspring [**22**].

Another important factor in the evolution of many organisms is *speciation*. A speciation is an event that leads to the creation of two new species. This may occur when the population of a species is the target of a large number of mutations over numerous generations. The accumulation of mutations can lead a species' population to form two groups that differ too much to reproduce with one another. This is usually due to a geographical barrier that has split the population. A speciation event can also affect a gene at a molecular level. When speciation event leads an ancestral specie to form two new species it also leads an ancestral gene to form two new genes, that will from that point on undergo distinct molecular changes as they are passed on from one generation to the next in different populations.

2.3.2. Gene Family

A gene family is a set of genes that all descend from one common ancestor. Two genes belonging to the same family are described as *homologous* genes. Moreover, genes within a family may differ in terms of their relationship with one another. The relationship between two homologous genes is determined by their LCA:

- *Orthologs*: A gene pair that diverged due to a speciation.
- *Paralogs*: A gene pair that diverged due to a duplication.
- *Xenologs*: A gene pair that diverged due to horizontal gene transfer.

Gene trees can provide evidence for gene evolutionary events, such as duplication events and speciation events, leading to illustrating the relationships of the members of a gene family. They represent the transmission history of a gene from an ancestral specie to existing ones. Figure 2.3 depicts the concept of a gene tree in a simple example.

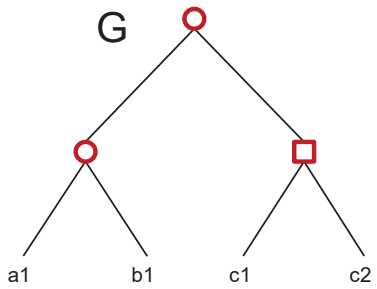


Fig. 2.3. The depicted gene tree G is constructed from one gene family. Its leaves represent genes within three species, and its internal nodes are labelled with evolutionary events. Each leaf is labelled with a gene copy in the corresponding species (e.g., gene a_1 is in species a). Speciation and duplication events are depicted with red circles and squares, respectively.

Inferring the relationships between members of a gene family is only one of the building blocks for the important endeavour to better understand the evolution of biological functions.

2.4. Phylogenetic Tree Inference and Errors

The objective of phylogenetic tree inference is to reconstruct a tree based on a hypothesis about the evolutionary relationships of a set of biological entities. However, building a phylogenetic tree that fully and perfectly represents the historical relationships between biological entities is unlikely. Tree inference may rely on morphological data but, for the purpose of this thesis, we will focus on the branch of computational phylogenetics that relies on sequence alignment.

2.4.1. Phylogenetic Tree Inference Methods

When reconstructing a phylogenetic tree to depict the evolution of a set of taxa, methods of inference may rely on the alignment of obtained DNA, RNA, or protein sequences to classify correlating sections that may be resulting from evolutionary relationships between the sequences. Molecular sequence analyses have been shown to be especially convenient to infer the relationships between species with a high degree of morphological similarity [**47**]. Additionally, sequences obtained from an organism are usually not as affected by the environment in which it lived as its morphological traits. Nonetheless, sequence analyses are not perfect, for example when dealing with recently diverged taxa, that have accumulated fewer substitutions since divergence, or with very old divergences, where the phylogenetic signal is obscured by homoplasy (gained or lost independently in separate lineages).

There exists a diverse array of methods for phylogenetic tree reconstruction or inference from molecular data. Popular methods of tree inference are usually grouped into two main classes: character-based methods and distance methods [**20**]. Each of these classes have their defining traits and properties as do the methods themselves. As the purpose of

this thesis is not tree inference, we will not enter into too many details and just briefly discuss the above-mentioned two inference classes and a few popular methods in each of them.

Character-based methods

Character-based methods are named after the fact that they use discrete phylogenetic characters directly, such as a DNA sequence, during tree reconstruction. The input is usually a multiple sequence alignment, and the output is a tree best representing this input.

- **Maximum likelihood method:**
  Maximum likelihood (ML) is a statistical method with the objective of estimating the parameters of a statistical model. This method was first developed by *Fisher* [**2**] and became popular in the early 1900s leading its application in multiple fields of biology, such as population genetics. However, the maximum likelihood method was only popularized in the field of phylogenetics with its use in the work of *Felsenstein* [**24, 26**]. The likelihood is the probability of obtaining the data at hand S with a model, given the parameters of the model, denoted $P(S|\theta)$. Maximal likelihood is, as its name points out, the scenario where $P(S|\theta)$ is maximized by finding optimal model parameters θ. The principle, when applying this method for phylogenetics, is using aligned molecular sequence data as input to estimate a phylogenetic tree (e.g., topology, branch lengths) and a substitution model's parameters (or model of sequence evolution), i.e., nucleotide substitution rates and nucleotide frequencies. However, finding the parameters that maximize the likelihood for phylogenetic tree inference is an NP-hard problem [**14**]. For this reason, the model of sequence evolution is often chosen, e.g., the Jukes and Cantor model or the Felsenstein model [**26, 40**], and heuristics have been developed that reduce the tree search space [**30, 62**].

- **Maximum Parsimony method:**
  The objective of this method is to find the phylogeny that requires the fewest necessary changes to explain the differences among the observed sequences. The maximum parsimony method consists in calculating a score for each tree on the set of leaves determined by the input aligned molecular sequence data, and then selecting the tree with minimal score. There are multiple ways to compute the score representing the evolutionary variation of a tree, such as the well-know Fitch and Sankoff algorithms [**28, 61**]. However, attempting to infer the most parsimonious tree can result in multiple equally parsimonious trees [**20**]. A way to deal with such cases is to make a combined tree that includes all the equally parsimonious

trees, called a *consensus tree*. There exist different types of consensus trees, and one popular type is called a *strict-consensus tree* [**20, 52**]. To construct it, for all equally parsimonious trees for a given aligned sequence data set, all the trees with any inconsistent branching patterns for a set of leaves are resolved by forming a multifurcating branching pattern [**20, 52**]. Nevertheless, the maximum parsimony method, unlike the maximum likelihood method, has the troublesome trait of being statistically inconsistent, which entails that the probability of converging towards a correct tree does not tend towards 1 with infinitely increasing input data [**25**]. Moreover, the tree search space increases drastically as the number of aligned sequences in the input data increases, making search over the entire space unpractical. Hence, branch and bound methods or heuristics are used [**34, 70**].

- **Bayesian inference method:**
  The Bayesian inference method is based on maximum likelihood methods but incorporates prior probability. More precisely, it is a probabilistic method based on Bayes' theorem which defines $P(\theta|S) = \frac{P(S|\theta)P(\theta)}{P(S)}$. In our case, $P(\theta)$ refers to the prior probability of the parameters of our model, such as the tree topology and branch length, $P(S|\theta)$ refers to the likelihood of the input data if our hypothesized parameters are true, and $P(S)$ is the probability of our input data (normalization constant). $P(\theta|S)$ corresponds to the probability that the chosen model, with its parameters, generated the input data. In other words, the posterior probability of a tree will indicate the probability of the tree to be correct. However, the normalization constant $P(S)$ is too expensive to compute. As a result, the Bayesian inference method produces a posterior probability distribution on trees, meaning it infers a set of trees and not a single tree. Moreover, inferring such a distribution is possible by using *Markov Chain Monte Carlo* methods (MCMC), e.g., Metropolis-Hasting [**32**]. Those methods generate a sample from the posterior $P(\theta|S)$, which can be used to estimate the posterior distribution. Finally, to infer a single tree, a consensus tree can be built from the obtained set of trees.

Distance-based methods

Distance-based methods take a matrix of distances of taxa as input during tree inference. To do so, the molecular sequence data is transformed into pairwise distances, which is achieved by calculating the genetic distances between each pair of sequences [**71**]. Distance-based methods usually output a weighted tree that realizes the distances between the taxa.

- **UPGMA method:**
  UPGMA stands for *Unweighted Pair Group Method with Arithmetic Mean* [**66, 67**]. This method is an agglomerative hierarchical clustering method, which is a type of cluster analysis method to build a hierarchy of clusters by merging them in a stepwise manner. UPGMA starts by grouping pairs of sequences (leaves) with minimal distance in the input distance matrix to form a subtree (at least once) and then goes on by iteratively grouping pairs of subtrees (or a subtree with a leaf) with minimal mean distance between their leaves. The UPGMA method takes $O(n^3)$ to construct a tree.
- **Neighbor-Joining method:**
  The Neighbor-Joining method is an agglomerative hierarchical clustering method proposed by *Naruya Saitou* and *Masatoshi Nei* [**60**]. Similarly to the UPGMA method, the Neighbor-Joining method iteratively groups subtrees (and starts by grouping leaves) until a tree with fully resolved topology is obtained. However, their selection criteria differ, the Neighbor-Joining method pairs subtrees (or leaves) that are closest to one another and that are the furthest from the rest of the subtrees (and leaves). The Neighbor-Joining method takes $O(n^3)$ to construct a tree.

It is important to realize that every method has strengths and weaknesses and that no method is perfect. For example, the distance-based methods introduced above are much more time-efficient compared to the other tree inference methods previously presented, but they are less accurate and tend to lack the qualitative information that may be obtained from within input alignments, due to the conversion to a pairwise-distance matrix [**27, 53, 68**].

2.4.2. Tree Inference Errors

Phylogenetic tree inference methods, whether they are distance-based or character-based, are not without flaws. A known issue with the maximum parsimony method is called *long branch attraction*, where related taxa are inaccurately inferred to be closely related, i.e., taxa with long branches may be grouped because of their branch length rather than because they are related by ancestry [**5, 25**]. There exist many different sources of error that may negatively affect the inference of a phylogeny [**9**]. As mentioned, some sources of error originate from the methods for tree inference, but errors may also be due to observed datasets, whether they are molecular datasets or morphological datasets, e.g., affected by homoplasy. Nonetheless, there has been a tremendous effort from the research community to deal with the causes of erroneous tree reconstruction. However, the mechanisms of evolution are so intricate that there will always be a lack of robust correlation between models and biological processes. For the purpose of this thesis, and understanding our motivation to develop

methods to analyse and compare gene trees, we outline some well-know sources of error affecting the inference of gene trees.

- **Erroneous molecular data:** The use of aligned sequences as input to phylogenetic inference means that, no matter the inference method, an inferred tree may suffer from errors introduced by the equipment or methods (e.g., sequencing methods, or alignment methods) used to produce it [**55**].
- **Lacking data:** The lack of data, due to genomes being partially sequenced for example, has been shown to drastically impact inference methods [**59, 75, 76**]. Moreover, even if there is a significant amount of data available, the lack of homologous genes in input data is still possible and may produce similar errors to those due to lack of data [**59, 75, 76**]. Additionally, the lack of sites for a single gene entails limited data which may result in poor inference quality [**56**].
- **Models of evolution:** Phylogenetic tree inference methods function under a certain set of assumptions which, if they are not met, may make these methods unfit for use and yield erroneous results. A parameter with great impact on inference is the chosen model of evolution. Variations in models of evolution have shown to affect tree reconstruction outputs [**37, 43**]. Unfortunately, a model may be chosen arbitrarily when a suitable model is not established. Therefore, it is important, based on a thorough analysis of the parameters, the inference method employed, and the quality of the molecular data, to select adequate models of evolution if the setting of the study allows it.
- **Heuristics:** Phylogenetic inference methods, as mentioned in section 2.4.1, cannot always search the entire tree space containing the optimal tree they seek to infer (they often correspond to NP-complete problems). Those methods often rely on heuristics to restrict the tree search space and, by doing so, the reconstructed trees can only be guaranteed to be optimal within the restricted space. Therefore, an optimal solution is not guaranteed.

2.5. Reconciliation

Reconciliation is a method for inferring the evolution of a gene family. However, it is well known that a species tree does not necessarily concur with a gene tree inferred from DNA sequences for a gene locus involved with said species. For example, a common reason for this difference is genetic polymorphism in ancestral species. In other words, reconciliation is also an approach for analyzing the inconsistencies between the evolutionary histories of genes, and the species through which they have evolved [**29**]. More specifically, reconciliation takes as input a species tree and gene tree(s), and then *reconciles* these trees to infer the minimum number of evolutionary events (e.g. gene duplication, gene loss, and speciation).

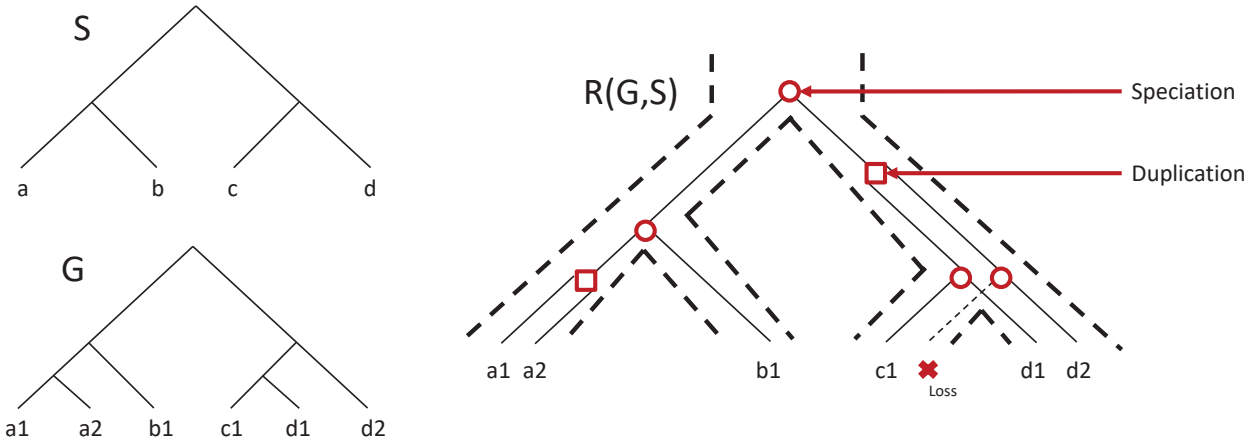


Fig. 2.4. *Top left:* A species tree $S = ((a,b),(c,d))$; *Bottom left:* A gene tree G for the gene family $F = \{a_1,a_2,b_1,c_1,d_1,d_2\}$, where the leaves are genes belonging to the species in S; *Bottom:* A reconciled tree $R(S,G)$, resulting from the embedding of G into S, with depicted speciation events, duplication events, and loss.

To reconcile a gene tree with a species tree means embedding the gene tree into the species tree to produce a new tree. Figure 2.4 illustrates this concept with a simple example.

There are different reconciliation models, such as the Duplication(D), Duplication-Loss(DL), and Duplication-Transfer-Loss(DTL) models, which have different attributes, depending on the array of evolutionary events they cover. We do not go over specific models of reconciliation but we recommend the work of El-Mabrouk and Noutahi [**22, 74**] as a good starting source for more information on the subject.

The information that can be obtained from phylogenetic reconciliation is only as good as its building blocks and, therefore, the quality of inference for the species tree and gene tree(s) is critical. Furthermore, it has been shown that reconciliation methods are biased when the inferred gene tree is not correct [**31**]. This raises the need for methods to analyse and compare phylogenetic trees, especially gene trees, since we know that sources of error in tree inference are common. There is a wide array of tools used to analyse and compare phylogenetic trees, but for the purpose of this thesis we focus on the development of distance metrics in the next chapter.

Chapter 3

Distance metrics: Background and related work

Over the last three decades, many new metrics have been developed and have extended the scope of applications of phylogenetic tree comparisons. Though the definition of these metrics was driven by different motivations and goals, they nevertheless share common principles.

A distance or a metric is a function defined on pairs of elements A and B of a space that satisfies the four following conditions: 1) The non-negative condition, as its name indicates, stating that a distance is always positive or equal to zero; 2) The symmetric condition stating that a distance from an element A to an element B is equal to the distance from B to A ($d(A,B) = d(B,A)$); 3) The identity condition stating that the distance is zero if and only if the elements are the same ($d(A,B) = 0 \Leftrightarrow A = B$); 4) The triangle inequality condition stating that the distance from an element A to an element B is lower or equal to the sum of the distances from A to C and C to B, for any C ($d(A,B) \leq d(A,C) + d(C,B)$).

Metrics designed for tree comparison are specific to some types of trees: some target binary trees, while others can handle polytomies, and certain metrics assume unrooted trees while others compare rooted trees; some metrics account for branch length or other tree properties. The challenge is to account for these specificities while preserving the conditions of a metric.

Another important aspect of a metric is its time complexity. In this field of study, the computing time of a metric is crucial because we may have to deal with very large trees. Metrics are also characterized by their theoretical maximum distance between two trees (diameter). This property relates to the accuracy of comparisons, as a larger diameter tends to enable more refined comparisons [**49**]. In addition, a common objective when developing a metric is to have a bell shaped, symmetric frequency distribution with significant variance. This property renders a metric more robust to errors in trees and increases its capacity to distinguish trees with small differences.

In this chapter we will introduce four different tree metric in the first four sections, where we report on theses metrics' properties, mechanisms, strengths, and weaknesses. In the fifth section we compare their properties, and the last section is the conclusion of this chapter.

The four metrics we are presenting are representative metrics covering two categories. The first category consists of *cluster-similarity* metrics. These metrics are denoted as such because they rely on information provided by tree clusters (or clades) and their attributes, such as their topology or leaves, to compute distances between compared trees. We refer to metrics belonging to the other category as *edge-based* metrics. As the name suggests, these metrics use the information provided by tree edges and their attributes to compute distances between compared trees.

The *cluster matching* distance discussed in section 3.2 and the *hierarchy-preserving* distance discussed in section 3.3 belong to the category of cluster-similarity metrics, while the *euclidean* distance discussed in section 3.4 belongs to edge-based metrics. These metrics were selected because they have interesting properties discussed in subsection 3.5.

We also describe the Robinson-Foulds distance which covers both categories in the following section (the version for unrooted trees is an edge-based metric while the one for rooted trees is an cluster-similarity metric). It was retained not because of its properties but because we extend it in the next chapters.

3.1. Robinson-Foulds Distance

3.1.1. Motivation

The RF distance is a popular distance measure, which has been used for a couple of decades now. Recently developed metrics are still based on the RF distance. Its original objective [**58**] was to devise a comparison method suitable for trees with internal nodes having an arbitrary number of edges (degree) i.e., binary and non-binary trees.

3.1.2. Detailed Description

The RF distance metric d_{RF} is a measure used to compare phylogenetic trees on the same set of leaves $\mathcal{L}$ of size n. Note, however, that d_{RF} is defined on a tree space where trees are unweighted and unlabeled. The RF distance remains widely used since it can be computed in linear time on rooted or unrooted trees. This makes the RF distance very practical, though it has limitations discussed below.

There are two methods to calculate $d_{RF}(T_1,T_2)$ between two trees T_1 and T_2. The first method is based on the number of tree edit operations between two trees. It quantifies the distance between two trees by calculating the minimum number of internal branch "contractions" ($Cont$) and "extensions" (Ext) needed to transform one tree into the other [**58**].

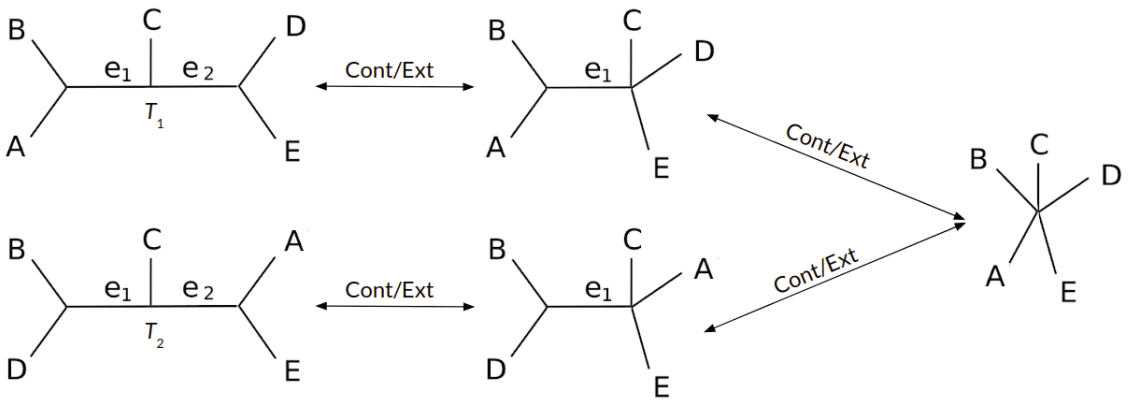


Fig. 3.1. Transforming T_1 into T_2, or T_2 into T_1 with a minimal number of edit operations, resulting in $d_{RF}(T_1,T_2) = 4$.

An edge contraction is an operation that takes in an edge $e = (x,y)$ and a tree T as input, then proceeds in transforming the tree T into the tree T' obtained from T by removing the edge e of T and identifying x and y i.e., T' is obtained by adding the edge (x, z) for each $z \in Ch(y) \setminus \{x\}$, and then removing y and its incident edges (including (x,y)).

An edge extension is and operation that takes in an internal non-binary node x, $X = \{y_1, \cdots, y_t\} \subsetneq Ch(x)$ a subset of $Ch(x)$ such that $|X| \geq 2$, and a tree T as input, the proceeds in transforming the tree T into the tree T' obtained from T by removing the edges (x,y_i), for $1 \leq i \leq t$, creating a node y and a new edge $e = (x,y)$ adjacent to x, and creating new edges (y, y_i), for $1 \leq i \leq t$. We provide an example for unrooted trees in Figure 3.1.

The second method is referred to as the *symmetric difference*. In contrast to the previous method it does not transform trees, but assesses elements forming them, more specifically, tree *clades* or *bipartitions*. A clade is a group of lineages that includes a common ancestor and all the descendants of that ancestor i.e., the leaf set of a subtree rooted at a node x: $L(T_x)$. The bipartition of a tree T corresponding to an internal edge $e = (x,y)$ is the unordered pair of clades $L(T_x)$ and $L(T_y)$ where T_x and T_y are the two subtrees rooted respectively at x and y obtained by removing e from T. A bipartition is *non-trivial* if it corresponds to an internal edge of T, and trivial otherwise. We denote by $\mathcal{B}(T)$ the set of non-trivial bipartitions of T.

For unrooted trees, it consists in counting the number of different bipartitions resulting from the internal branches of both trees. More precisely, for an unrooted tree T with the non-trivial bipartition set $\mathcal{B}(T)$, we define:

$$d_{RF}(T_1,T_2) = |(\mathcal{B}(T_1) \setminus \mathcal{B}(T_2)) \cup (\mathcal{B}(T_2) \setminus \mathcal{B}(T_1))|$$

As for rooted trees, the symmetric difference refers to the number of different clades in both trees. More precisely, for a rooted tree T^R with the non-trivial clade set $\mathcal{C}(T^R)$, we define:

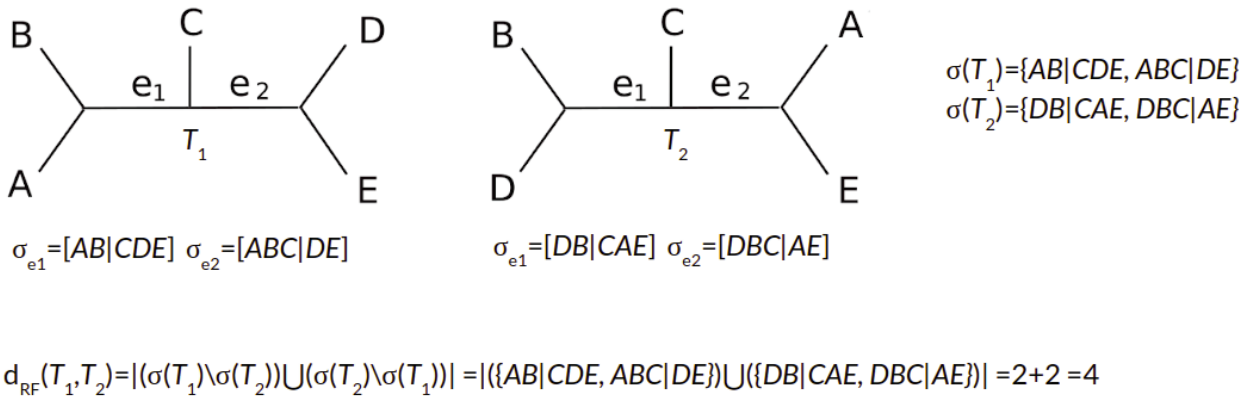


Fig. 3.2. $d_{RF}(T_1,T_2)$ computed as the symmetric difference between the trees' bipartitions.

$$d_{RF}(T_1^R,T_2^R) = |(\mathcal{C}(T_1^R) \setminus \mathcal{C}(T_2^R)) \cup (\mathcal{C}(T_2^R) \setminus \mathcal{C}(T_1^R))|$$

Since a binary unrooted tree with n leaves has $n-3$ internal branches, which corresponds to the number of non-trivial bipartitions of the tree, the maximum RF distance between two binary trees is $2(n-3)$. This maximum is reached when the trees being compared have an entirely different set of non-trivial bipartitions.

Similarly, since a binary rooted tree with n leaves has $n-2$ internal branches, which corresponds to the number of non-trivial clades of the tree, the maximum RF distance between two binary trees is $2(n-2)$. This maximum is reached when the trees being compared have an entirely different set of non-trivial clades.

We provide an example for unrooted trees in Figure 3.2. The two methods above used for computing the RF distance are mathematically equivalent since the number of bipartitions and the number of internal branches in a tree are the same.

3.1.3. Discussion

Distances between random pairs of binary trees tend to be near the maximum. This limits our ability to meaningfully distinguish pairs of arbitrary binary trees. Even small differences quickly lead to near maximum distances. As a result, the RF distance shows low robustness (high sensitivity) to errors in trees, as even a single error can maximize the distance between the compared trees [**58**]. For a number of reasons, such as limited data on past biological entities, the use of various evolutionary models and reconstruction algorithms, errors are common in the construction of phylogenetic trees. On account of that, we want to ensure that distances are not significantly distorted by such errors. This limitation is addressed by the distance measure presented next, the *cluster matching* distance. Nonetheless, the RF distance remains a widely used and intuitive method with efficient time complexity $\mathcal{O}(n)$, that has led to multiple extensions such as the bipartition matching distance [**46**].

3.2. Cluster Matching Distance

3.2.1. Motivation

The cluster matching (CM) distance is a recent distance metric on rooted trees that can be seen as a weighted version of the rooted RF distance. The objective was to address the main weaknesses of the RF distance, as discussed in the previous section, while focusing on rooted trees.

3.2.2. Detailed Description

The CM distance metric d_{CM} is a measure that compares binary phylogenetic trees on the same set of leaves $\mathcal{L}$ of size n. The cluster of an internal node x is defined as $\mathcal{C}_T(x) = L(T(x))$, where $T(x)$ is the subtree rooted in x. The set of all clusters of T is defined as $\mathcal{H}(T) = \bigcup_{y \in V(T)} \mathcal{C}_T(y)$. Recall that d_{CM} is defined on a tree space where trees are rooted and unlabeled.

d_{CM} is computed based on a bipartite graph. A bipartite graph is a means to illustrate a set of graph vertices separated into two disjoint sets, such that no two graph vertices within the same set are directly connected. In this context, we use a specific kind of bipartite graph, a complete weighted bipartite graph, where all the vertices of one set are connected to the other set and edges are weighted.

The CM distance between two trees, $d_{CM}(T_1,T_2)$, is defined as the resulting weight of the minimum weight perfect matching in the bipartite graph between those trees [**49**]. A bipartite graph is formed by defining two sets, each set containing the internal vertices of each of the compared trees T_1 and T_2. Edges are then drawn between each element of a set to all the elements of the other set. The weight of an edge (v_1,v_2) in the bipartite graph $B(T_1,T_2)$, is the cardinality of the symmetric difference ($\ominus$) between the elements of $C_T(v_1)$ and $C_T(v_2)$. Given that for a tree T, $r(T)$ denotes its root, $V_{int}(T)$ are internal vertices, and E is the set of edges between the internal vertices of T_1 and those of T_2 in the bipartite graph, the complete weighted bipartite graph is formalized as follows:

$$B(T_1,T_2) := ((V_{int}(T_1) \setminus \{r(T_1)\}) \cup (V_{int}(T_2) \setminus \{r(T_2)\}), E)$$

Where the weight of each edge $\{u,v\}$ is:

$$W(u,v) := |C_{T_1}(v_1) \ominus C_{T_2}(v)| = |(C_{T_1}(u) \setminus C_{T_2}(v)) \cup ((C_{T_2}(v) \setminus C_{T_1}(u))|$$

We provide an example for this metric in Figure 3.3.

3.2.3. Discussion

The CM distance, as mentioned in the previous section, has multiple advantageous properties compared to the rooted RF distance. The CM distance yields much better distribution

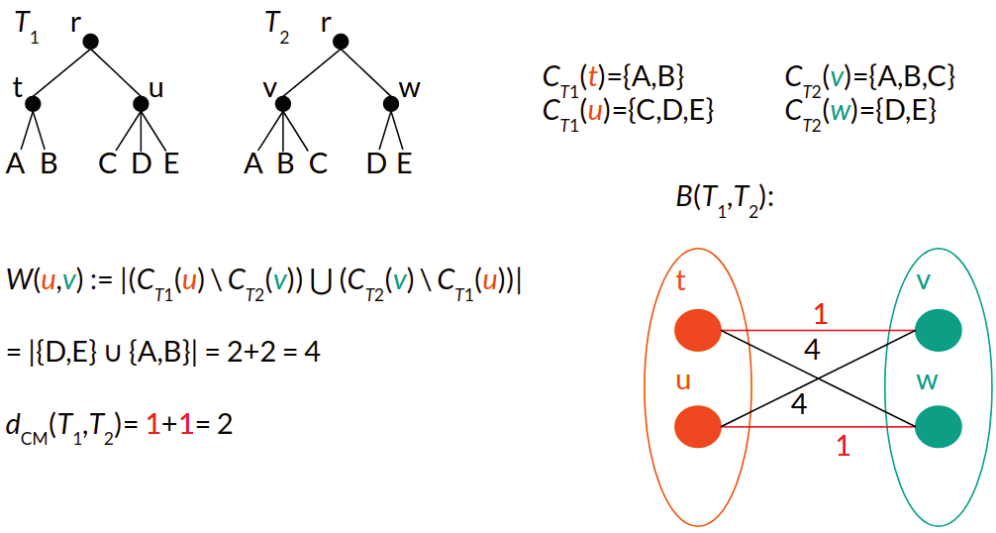


Fig. 3.3. Example of computation for the CM distance between two trees ($d_{CM}(T_1,T_2)$).

properties in terms of symmetry and variance. The CM distance is much more robust to errors and less sensitive to large differences between trees. The CM distance also has a larger diameter ($\theta(n^2)$) than the rooted RF distance, enabling more subtle comparisons that can distinguish differences between trees with a higher resolution [**49**]. Computing the CM distance has, however, a higher time complexity ($\mathcal{O}(n^{2.5}\log n)$) compared to the rooted RF distance, though it is still considered to be sufficiently efficient to compute in many practical situations [**49**].

3.3. Hierarchy-Preserving Distance

3.3.1. Motivation

This distance is another cluster-similarity based metric, like the CM distance. This kind of metric has shown to not be prone to skewed distribution of distances between most pairs of binary trees (e.g. [**6**]), a weaknesses of the RF distance where the majority of distances between a random pair of trees are comparatively very large. This is an attractive trait for the authors who's objective is precise and accurate discrimination between sets of trees. More specifically this distance is based on the concept of a hierarchy-preserving map, which relates trees that have similar hierarchies [**33**]. Another objective, stated by the authors, was the increase of accuracy of phylogenetic reconstruction using Markov Chain Monte Carlo methods [**33**].

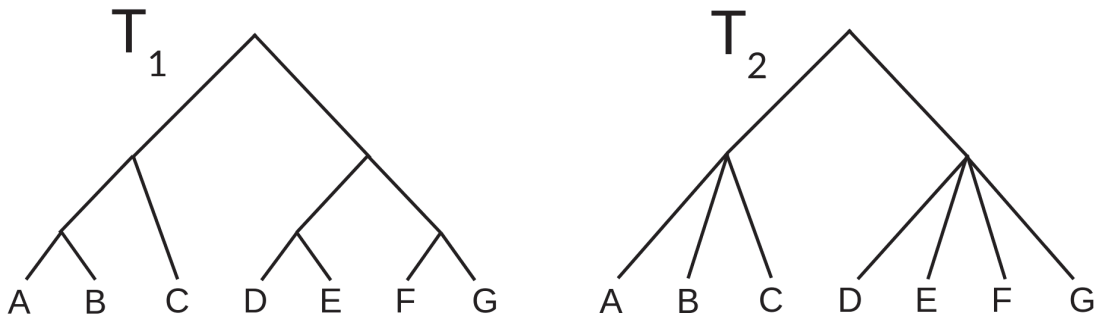


Fig. 3.4. Two trees T_1 and T_2 with a hierarchy-preserving map from $H(T_1)$ to $H(T_2)$ that maps AB to ABC, maps ABC to $ABCDEFG$, maps DE and FG to $DEFG$, and maps $DEFG$ to $ABCDEFG$.

3.3.2. Detailed Description

The hierarchy-preserving (HP) distance metric d_{HP} is a measure that compares rooted phylogenetic trees on the same set of taxa $\mathcal{L}$ of size n. A *hierarchy* H on a set $\mathcal{L}$ is the collection of all the subsets of $\mathcal{L}$, which contains both $\mathcal{L}$ and all singleton sets $\{l\}$ for $l \in \mathcal{L}$ [**33**]. Additionally, assuming H_1,$H_2 \in H$, then $H_1 \bigcap H_2 = \varnothing$, $H_1 \subseteq H_2$, or $H_2 \subseteq H_1$ [**33**]. We will referred to the elements of a hierarchy H hereafter as *sub-hierarchies*.

A *hierarchy-preserving map* is a depiction indicating the relationships of the elements of the hierarchies of the compared trees. Let T_1 and T_2 be two trees on $\mathcal{L}$ with hierarchies $H(T_1)$ and $H(T_2)$. Then the hierarchy-preserving map that maps $H(T_1)$ to $H(T_2)$ indicates, for each sub-hierarchy of $H(T_1)$, which sub-hierarchy of $H(T_2)$ they are a subset of. Figure 3.4 depicts such scenario. The authors specify two main properties of a hierarchy-preserving map [**33**]. For all A,$B \in H(T_1)$ and δ being the identity on singletons for the map $\delta : H(T_1) \to H(T_2)$:

(1) **Enveloping:** $A \subseteq \delta(A)$

(2) **Subset-Preserving:** $A \subset B$ implies $\delta(A) \subset \delta(B)$

The authors show that there is a partial order $\leq_{HP}$ on the set of trees on $\mathcal{L}$, denoted as $\mathcal{F}$. They do so by associating it to the concept of hierarchy-preserving maps, and convey that if there is a hierarchy-preserving map from $H(T_1)$ to $H(T_2)$ then we declare $T_1 \leq_{HP} T_2$.

Let $\mathcal{H}(\mathcal{L})$ denote the Hasse diagram of the set of trees $\mathcal{F}$ under $\leq_{HP}$. [**33**]. This Hasse diagram is the graphical representation of the partially ordered set $\mathcal{F}$. Figure 3.5 depicts a relatively simple example of $\mathcal{H}(\mathcal{L})$ with a small number of leaves n.

The distance denoted $d_{HP}(T_1,T_2)$ is defined to be the geodesic distance from T_1 to T_2 in $\mathcal{H}(\mathcal{L})$. The geodesic distance corresponds to the number of edges forming the shortest path between two vertices in a graph, in this case $\mathcal{H}(\mathcal{L})$. To calculate d_{HP} the authors decided to focus on the movement around $\mathcal{H}(\mathcal{L})$. To do so they defined two "movement" operations, one called an *Up-move*, and the other *Down-move*. An up-move corresponds the transition from a tree T_1 to a T_2 in $\mathcal{H}(\mathcal{L})$ if $T_1 \leq_{HP} T_2$, but if $T_2 \leq_{HP} T_1$ then the operation performed

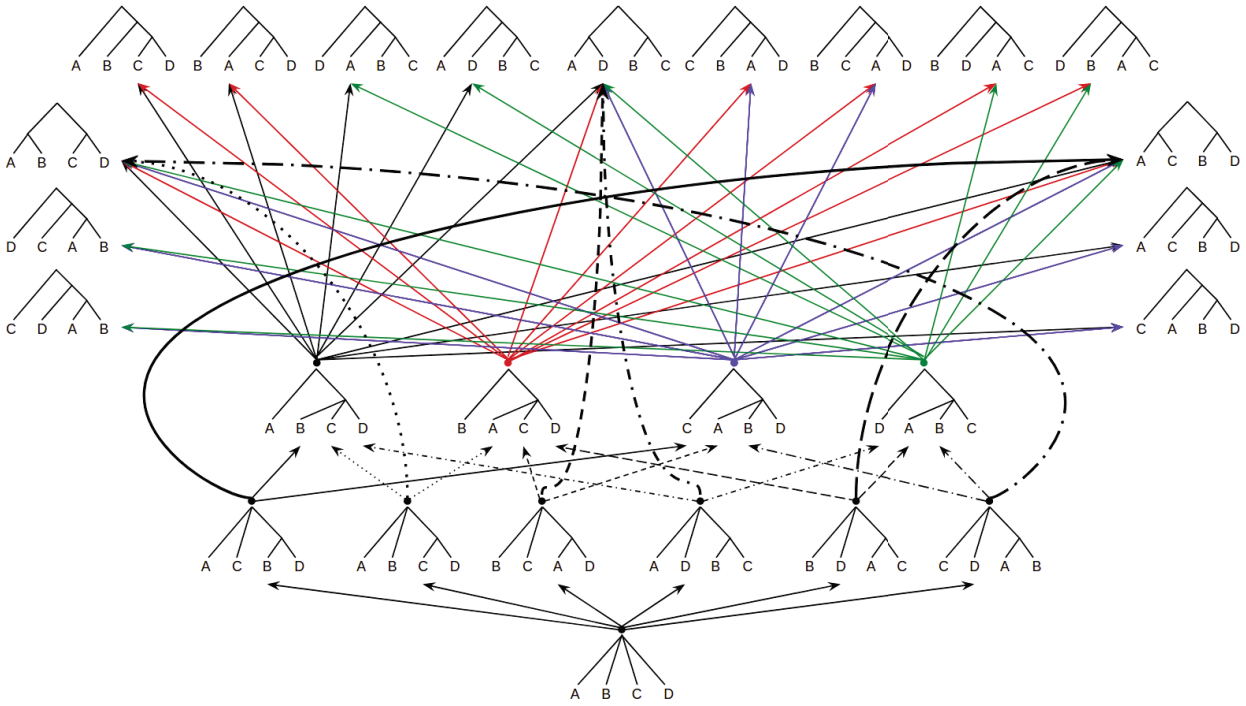


Fig. 3.5. The Hasse diagram of the set $\mathcal{F}$ under $\leq_{HP}$ with n=4. Note that the $\leq_{HP}$-minimal element of $\mathcal{H}(\mathcal{L})$ is the star tree, and that the $\leq_{HP}$-maximal elements are the binary trees.

is a down-move. The authors describe the up-move as the deletion of some distinct pair of clusters X,$Y \in H(T)$ that are inclusion-maximal in a third cluster Z, with $X \cup Y \subsetneq Z$ and then the addition of $X \cup Y$. A down-move is describe as the reverse of this[**33**]. To be precise, an inclusion-maximal cluster that is an element of a parent cluster, is a cluster that is not a sub-cluster of some other cluster also part of said parent cluster. Figure 3.6 depicts these movement operations. For more examples, see Figure 3 and 4 in[**33**]. This means the distance $d_{HP}(T_1,T_2)$ therefore behaves so that if an up-move or down-move on one of the compared trees turns it into the other, then their distance would be 1.

To simplify the calculation of the distance between certain trees the authors introduce the notion of *rank* of T, $f(T)$. Let $P(T)$ be the set of proper clusters, which exclude singletons and $\mathcal{L}$:

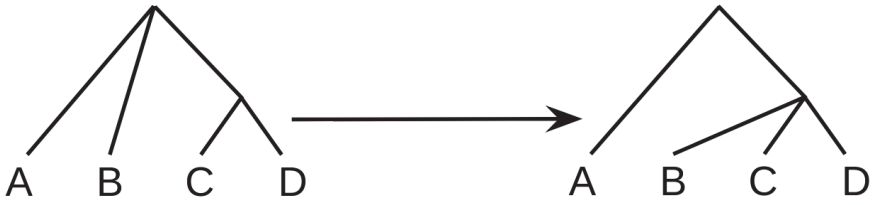
$$f(T) = \left(\sum_{A \in P(T)} |A| \right) - |P(T)| = \sum_{A \in P(T)} (|A| - 1)$$

Note the maximum rank of a tree with n leaves is $\frac{(n-1)(n-2)}{2}$.

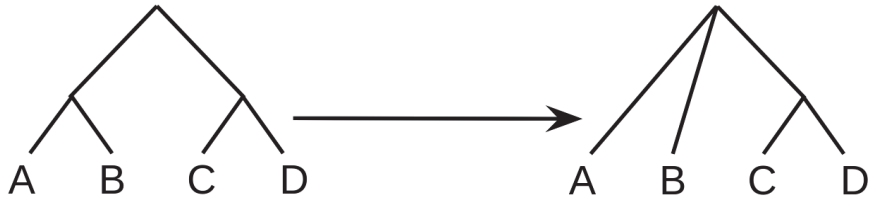
If one tree is above the other in $\mathcal{H}(\mathcal{L})$ when two trees are compared then we can use the authors' shortcut[**33**]:

Suppose $T_1 \leq_{HP} T_2$, then:

$$d_{HP}(T_1,T_2) = f(T_2) - f(T_1)$$



a) Up-move with a singleton and a cluster



b) Down-move with two singletons

Fig. 3.6. Examples of each operation used to traverse $\mathcal{H}(\mathcal{L})$, a) showing an up-move targeting cluster C,D and singleton B, and b) showing a down-move where both targeted clusters, A and B, are singletons.

If that is not the case then:

$$|f(T_1) - f(T_2)| \leq d_{HP}(T_1,T_2) \leq f(T_1) + f(T_2)$$

The authors state that exact calculations of the distance d_{HP} are computationally expensive because there is no algorithm enabling subexponential runtime. In response to that, they developed an upper bound e_{HP}, and show that their algorithm to determine it is polynomial. The upper bound is shown to often be equal to the true distance (i.e., over 80% on trees with $n = 9$ [**33**]). Note that the upper bound is not a metric as it does not satisfy the triangle inequality condition. The method to find the upper bound can be described in two main steps:

(1) Find $\leq_{HP}$-maximal trees that have a hierarchy-preserving map into both T_1 and T_2
(2) Then find a minimum path between T_1 and T_2 that goes through a $\leq_{HP}$-maximal tree.

In other words, let $max \leq_{HP} (T_1,T_2)$ be the set of trees T_i in $\mathcal{F}$ that are $\leq_{HP}$-maximal and satisfy $T_i \leq_{HP} T_1$ and $T_i \leq_{HP} T_2$, then $e_{HP} = min(f(T_1) + f(T_2) - 2f(T_i))$.

The algorithms needed to compute the upper bound e_{HP} and the metric d_{HP} were implemented and the authors have reported computational results on their distance. Those results suggest that the larger the upper bound, the less accurate the distances [**33**].They also point out that the true metric and the upper bound both share the same desirable statistical properties as other cluster-similarity metrics [**49, 46**]. The experimental results

show that they avoid the skewness that affects other easily computable metrics, like the RF distance [**33**].

3.3.3. Discussion

The HP distance features some compelling traits. It can meaningfully distinguish compared rooted trees thanks to its cluster-similarity properties. The authors also claim that these properties, paired with the operations used to move around the tree space, are expected to improve Markov Chain Monte Carlo searches of the tree-space around trees of similar hierarchies [**33**]. Nonetheless, the authors acknowledge significant weaknesses of their metric and upper bound, which is why their main objective remains to find a way to improve the runtime of d_{HP} and accuracy of e_{HP}.

3.4. Euclidean Distance

3.4.1. Motivation

Kendall and Colijn [**42**] defined an Euclidean distance metric to compare rooted phylogenetic trees. The authors' motivation was to create a method that would help cluster groups of trees, a requirement to address their objective of capturing and visualizing distinct patterns of evolution.

3.4.2. Detailed Description

As for previous metrics, the euclidean distance measure compares rooted trees defined on the same set of leaves. This metric can handle trees with internal vertices of any degree, and can optionally account for branch length in its comparisons. Branch length can be used to capture different types of evolutionary data such as the number of mutations or the time span between the species in two branch nodes. This new metric compares the positions of the most recent (i.e., closest in the tree) common ancestors of all pairs of leaves across trees [**42**]. This is a mechanism to evaluate the similarity of the shape of compared trees. Those shapes, or topologies, can then be analyzed to potentially determine patterns of evolution. To do that, the metric relies on combining and comparing two vectors, $m(T)$ and $M(T)$, of identical size. For a tree T, $M(T)$ captures the tree topology by including the distance $M_{i,j}$, in terms of path length, between the most recent common ancestors of every possible pair of leaves (i,j) in the leaf set of size n, appended with the length p_i of each pendant edge to every leaf i. The pendant edge of a leaf is the edge between the leaf and its direct ancestor. The vector $M(T)$ is represented as follows:

$$M(T) = (M_{1,2},M_{1,3},...,M_{n-1,n}, p_1,...,p_n)$$

Second, $m(T)$ has a similar structure to $M(T)$, except that the distance $m_{i,j}$ is measured in terms of number of edges, as opposed to path length, and all its pendant edge length values equal 1 (the immediate ancestor is always one edge away). In other words, $m(T)$ does not consider branch length but only tree topology. The vector $m(T)$ is represented as follows, with 1s for its last n components:

$$m(T) = (m_{1,2},m_{1,3},...,m_{n-1,n},1,...,1)$$

The next step in computing the Euclidean distance is to form a convex combination of these two vectors by parameterizing them with $\lambda \in [0,1]$. λ enables the analyst to determine to what extent the branch lengths of a tree, versus its topology, contribute to the tree distance [**42**]. The convex combination allows us to compute the relative mean of the Euclidean distance on the ["topology", "topology and branch length"] interval. We combine the two distances defined above as follows:

$$v_\lambda(T) = (1-\lambda)m(T) + \lambda M(T)$$

An appropriate value for λ depends on the targeted application, and the meaning of branch length. Finally, the Euclidean distance can be defined as follows:

$$d_\lambda(T_1,T_2) = ||v_\lambda(T_1) - v_\lambda(T_2)||$$

where $||\cdot||$ stands for the classic Euclidean distance between two vectors (L^2norm). We provide an example to illustrate this metric in Figure 3.7. We can estimate the impact of accounting for branch length since we are able to compare the results of distance calculations with different weights attributed to the tree topology only vector ($m(T)$) and the vector accounting for branch length ($M(T)$). This can help determine if the information provided by branch length is an advantage or a disturbance for the tree comparisons, and consequently to the establishment of patterns of evolution.

3.4.3. Discussion

The Euclidean distance is a metric that compares rooted phylogenetic trees and can account for branch length. It provides a mechanism, through a parameter ($\lambda \in [0,1]$), to control the extent to which branch length affects distance computations and therefore tree comparisons. Though the Euclidean distance metric was developed for rooted trees, the authors suggest that it could also be used for unrooted trees. Compared unrooted trees would first have to be re-rooted to their equivalent pendant edge of a given leaf i, and then the Euclidean distance metric would be applied as suggested for rooted trees. If the two unrooted trees have the same topology, then rooting them on the same pendant edge of leaf i should give them the same rooted topology.

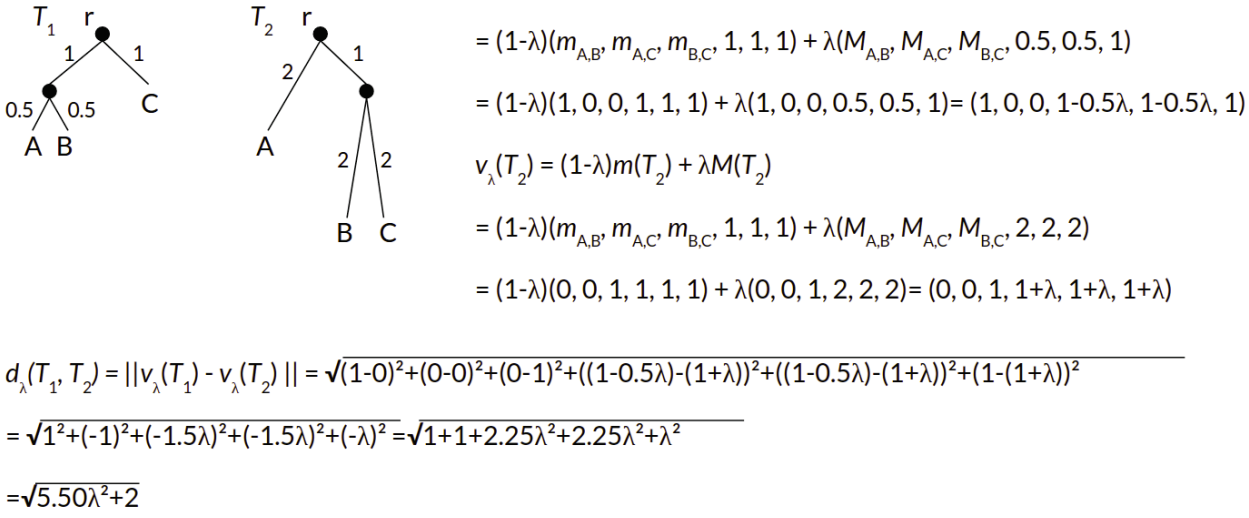


Fig. 3.7. The Euclidean distance ($d_\lambda(T_1,T_2)$) of the two trees T1 and T2.

Furthermore, we know that this metric accounts for all pairs of leaves. Since for n leaves, there are $n*(n-1)$ pairs, we can infer that the time complexity for computing the Euclidean distance is quadratic: $\mathcal{O}(n^2)$. Additionally, experimental results in [**42**] suggest that the Euclidean distance follows a symmetric distribution with significant variance. This results into a certain robustness to tree errors. High robustness entails limited effects of small changes in a tree, such as leaf regrafting. Indeed, a particular leaf is involved in a small percentage of computations for large trees. Nonetheless, the authors point out its sensitivity to deep branching structural differences (differences relatively close to the root) [**42**], meaning that errors leading to such differences would have great impact on the distance computation. However, this characteristic also renders this metric suitable to identify differences in deep tree structures within a set of trees.

3.5. Comparisons of Metrics

Table 3.1 provides a systematic comparison of the four metrics we covered in this section, according to the criteria that were discussed at the beginning of this chapter.

In general, what can be concluded from Table 3.1 is that the suitability of these metrics depends on their intended application. More specifically, the Euclidean distance seems to be the most versatile of the four metrics considered. Moreover, it can be noticed that it is the only metric that does not have a diameter, which is due to the metric allowing arbitrary branch lengths. However, its versatility does not entail that this metric is always the most suitable since it has a high time complexity. The RF distance, in contrast, has by far the lowest time complexity, and even though it is prone to distortions by tree errors and does not account for branch length, it is the most suitable when one requires a fast computations.

Table 3.1. Metrics Comparison Table

Properties / Metrics	**RF** (d_{RF})	**CM** (d_{CM})	**Euclidean** (d_λ)	**HP** (d_{HP})
Satisfies the conditions of a metric	+	+	+	+
Accepts nonbinary trees	+	+	+	+
Rooted / Unrooted trees	Both	Rooted	Both	Rooted
Diameter on trees with n leaves	$2(n-3)$	$\theta(n^2)$	N/A	$(n-1)(n-2)$
Use of branch length	−	−	+	−
Time complexity on trees with n leaves	$\mathcal{O}(n)$	$\mathcal{O}(n^{2.5}\log n)$	$\mathcal{O}(n^2)$	$\mathcal{O}(2^{poly(n)})$
Relatively robust to tree errors	−	++	+	++

The CM distance is the most robust to tree errors along with the HP distance but, when applicable, it is only suitable when one can afford to trade lower efficiency for gains in accuracy and resolution. The HP distance is the most computationally intensive of the four presented metrics and seems to be the least practical for general use.

3.6. Conclusion

As discussed in the previous sections, all presented metrics have weaknesses and applications contexts for which they are suited. However, it is important to note that, in this chapter, we selected a representative subset of metrics across the two presented metric categories, with interesting and diverse properties, and that there are many more metrics for phylogenetic trees that could be considered here. The RF distance has been identified as a fast, and relatively small tree distance calculator. This makes it ideal to compare a tree with other similar trees and determine their relative degree of similarity [**58**]. The CM distance, being highly precise even when comparing very different trees, is ideal when comparing a topologically diverse set of phylogenetic trees [**49**]. The Euclidean distance is relevant to many types of trees and comparisons [**42**], but experimental results show that its most useful application is in deep tree structure analysis, by emphasizing differences among trees that are located near the root rather than the ones near the leaves [**42**]. This is consistent with the authors' objective of detecting distinct patterns of evolution. Concerning the HP

distance, some of its properties are favorable are favorable for some target applications, such as phylogenetic reconstruction using Markov Chain Monte Carlo methods [**33**] though it seems to have room for improvement, both in terms of accuracy and time complexity.

There is no perfect metric and we do not know if there will ever be one. All existing metrics function under a certain set of assumptions which, if they are not met, make these metrics unfit for use. It is therefore important, based on a thorough analysis of the current needs in phylogenetic tree analysis, to devise new metrics that are suited for the various applications where current metrics are inadequate. Most particularly, there is no distance metric that can handle internally labeled trees, which are important in the context of the comparison of gene trees. This topic will be addressed in the next chapters.

Chapter 4

A Generalized Robinson-Foulds Distance for Labeled Trees

The article presented in this chapter was accepted for publication in the *BMC Genomics* journal. It describes a "labeled Robinson-Foulds" distance, a metric that accounts for internal node labels for the comparison of labeled gene trees. The supplementary material of this article is available in the appendix.

Comparing trees is an essential task for many purposes, and especially in phylogeny where different reconstruction tools may lead to different trees, likely representing contradictory evolutionary information. There is a large variety of pairwise measures of dissimilarity that have been developed for comparing trees with no information on internal nodes, such as the Robinson-Foulds distance. Unfortunately very few measures have been designed for node-labeled trees. For instance, this is required for the case of reconciled gene trees that may be labeled with evolutionary events such as speciation, duplication, or horizontal gene transfer. In this chapter, we present a simple and natural extension of the *RF* distance to node labeled trees. Our *RF* extension's characteristics make it useful for comparing gene trees under various evolutionary models that may involve speciation, duplication, loss, HGT, and other potential evolutionary events. We also implemented an accurate heuristic useful for preliminary analysis and comparison of labeled reconciled gene trees.

Contributions: Samuel Briand, Christophe Dessimoz, Nadia El-Mabrouk, and Manuel Lafond participated collectively in the development of the proofs and algorithms included in the article, and wrote the manuscript. Christophe Dessimoz and Gabriela Lobinska implemented the distance software presented in the manuscript. Christophe Dessimoz designed and performed the experiments discussed in the article. All authors have read and approved the final manuscript.

A Generalized Robinson-Foulds Distance for Labeled Trees

Samuel Briand[1], Christophe Dessimoz[2,4,5,6,7], Nadia El-Mabrouk[1], Manuel Lafond[3], and Gabriela Lobinska[4]

1. DIRO, Université de Montréal
2. Department of Computational Biology, University of Lausanne
3. Computer Science Department, Université de Sherbrooke
4. Department of Genetics Evolution and Environment, University College London
5. Center for Integrative Genomics, University of Lausanne
6. Swiss Institute of Bioinformatics
7. Department of Computer Science, University College London

4.1. Abstract

Background: The Robinson-Foulds (RF) distance is a well-established measure between phylogenetic trees. Despite a lack of biological justification, it has the advantages of being a proper metric and being computable in linear time. For phylogenetic applications involving genes, however, a crucial aspect of the trees ignored by the RF metric is the type of the branching event (e.g. speciation, duplication, transfer, etc).

Results: We extend RF to trees with labeled internal nodes by including a node *flip* operation, alongside edge contractions and extensions. We explore properties of this extended RF distance in the case of a binary labeling. In particular, we show that contrary to the unlabeled case, an optimal edit path may require contracting "good" edges, i.e. edges shared between the two trees.

Conclusions: We provide a 2-approximation algorithm which is shown to perform well empirically. Looking ahead, computing distances between labeled trees opens up a variety of new algorithmic directions.

Availibility and implementation: The software written in Python is available in the pylabeledrf repository at `https://github.com/DessimozLab/pylabeledrf`.

Keywords: edit distance, labeled trees, Robinson-Foulds, tree metric

4.2. Background

Phylogenic trees represent the evolutionary relationship between sets of genetic elements or taxa, where the elements of a set are in one-to-one relationship with the leaves of the

corresponding tree [**65**]. Different phylogenetic inference methods may lead to different trees, and each method, typically exploring a large space of trees, can also result in multiple equally likely solutions for the same dataset. It follows that comparing trees is an essential task for finding out how inferred trees are far from one another, or how an inferred tree is far from a simulated tree or from a gold standard tree for the same datasets.

Designing appropriate tree metrics is a widely explored branch of research. A variety of measures have been designed for different types of trees, rooted or unrooted, some restricted to comparing tree shapes [**15**], others considering multilabeled trees, i.e. trees with repeated leaf labels [**44**] and yet others considering information on edge length [**10**]. In particular, a large number of pairwise measures of similarity or dissimilarity have been developed for comparing two topologies on the same leafset. Among them are the methods based on counting the structural differences between the two trees in terms of path length, bipartitions or quartets for unrooted trees, clades or triplets for rooted trees [**12, 23, 16**], or those based on minimizing a number of rearrangements that disconnect and reconnect subpieces of a tree, such as nearest neighbour interchange (NNI), subtree-pruning-regrafting (SPR) or Tree-Bisection-Reconnection (TBR) moves [**39, 36, 3**]. While the latter methods are NP-hard [**46**], the former are typically computable in polynomial time. In particular, the Robinson-Foulds (RF) distance, defined in terms of bipartition dissimilarity for unrooted trees, and clade dissimilarity for rooted trees [**48**], can be computed in linear [**19**], and even sublinear time [**54**].

Despite several drawbacks such as lack of robustness (a small change in a tree may cause a disproportional change in the distance), skewed distribution [**69, 11, 13**], and a lack of biological rationale, RF remains the most widely used measure, not only in phylogenetics, but also in other fields such as in linguistics. To increase robustness, improved versions of the RF distance have also been developed [**46, 49**].

In addition of being efficiently computable, RF has the merit of being a true metric. It was originally defined on unrooted trees, in terms of edit operations on the tree edges: the minimum number of edge contraction and extension needed to transform one tree into the other [**58**]. Interestingly, the same metric, expressed in terms of node deletion and insertion, has been widely used in the context of data featuring hierarchical dependencies, modeled as trees with labeled nodes. In this case, the standard Tree Edit Distance (TED) is defined in terms of a minimum cost path of node deletion, node insertion and node relabeling (label substitution) transforming one tree to the other, for two trees sharing the same set of node labels (i.e. each label is present exactly once in each tree). While the less constrained version of the problem on unordered labeled trees is NP-complete [**80**], most variants are solvable in polynomial time [**78, 79, 64**].

Even though this kind of hierarchical node labeling has limited applicability for phylogenetic trees, other types of labeling can be used in the context of genetic data comparison.

In the case of gene trees, it is important to identify the evolutionary event (duplication, speciation, transfer, etc) that has led to a given bifurcation. For example, information on duplication and speciation node labeling is provided for the trees of the Ensembl Compara database [**73**] (reconciled with *TreeBest* [**63**]). Therefore, being able to compare labeled phylogenies is important in the context of gene tree reconstruction and analysis.

This paper is the first effort towards extending the RF distance to labeled trees involving, in addition to edge contraction and extension (operations that can alternatively be defined as node insertion and deletion), a node substitution or "relabeling" operation. Importantly, our extended RF remains a metric in the mathematical sense.

While the formulation of the RF distance in terms of edit operations is known, the bipartition and clade formulations are often those that are used in the literature. Though similar, the three formulations present some differences depending on whether the trees are rooted or unrooted. We begin by making these differences explicit. We then explore some properties of the extended RF distance in the case of two labels (e.g. speciation and duplication). In particular, we show that, in contrast to the RF distance for unlabeled trees, an optimal edit path for labeled trees may involve contracting good edges, i.e. edges representing common bipartitions of the two compared trees, which makes the extended RF much harder to compute than the basic RF. We then explore various avenues for computing the extended RF. We give an exact algorithm for contracting "mixed subtrees", i.e. subtrees with alternating labels, and a bounded heuristic for general trees that achieves a factor 2 approximation. In the following section, the heuristic is shown, on simulated datasets, to be efficient, by plotting the number of tree edits against the computed RF distance. Finally, we explore some avenues for improvement. All proofs are given in the appendix.

4.3. Notations and Concepts

Let T be a tree with a node set $V(T)$ and an edge set $E(T)$. Given a node x of T, the *degree of* x is the number of edges incident to x. We denote by $L(T) \subseteq V(T)$ the set of *leaves of* T, i.e. the set of nodes of T of degree one. A node of $V(T) \setminus L(T)$ is called an *internal node*. A tree with a single internal node is called a *star tree*. An edge connecting two internal nodes is called an *internal edge*; otherwise, it is a *terminal edge*. Moreover, a *rooted tree* admits a single internal node $r(T)$ considered as the root.

Let x and y be two nodes of a rooted tree T; y is an *ancestor* of x if y is on the path from x to the root (possibly y itself); y is a *descendant* of x if y is on the path from x to a leaf (possibly y itself) of T. For a rooted tree, we may write (x,y) for an edge between x and y where x is closer to the root. We say that y is a *child* of x. If T is unrooted, we call the set $\{y : \{x,y\} \in E(T)\}$ the set of children of x (this is an unusual definition, but defining a notion of children for both rooted and unrooted trees will be useful later). For a rooted or an unrooted tree T, we denote by $Ch(x)$ the set of children of an internal node x of T.

A tree T representing the evolution of a set $\mathcal{L}$ of entities (usually taxa or genes) is a tree with a one-to-one mapping between $L(T)$ and $\mathcal{L}$. We simply write $\mathcal{L} = L(T)$ and say that T is a *tree for* $\mathcal{L}$. An internal node represents an ancestral event (classically a speciation or a duplication) leading from one to many different entities. Moreover rooting a tree amounts to determining the common ancestor of all entities, i.e. determining the direction of evolution. Accordingly, internal nodes of an evolutionary tree (which are the trees considered in this paper) should be of degree at least 3, except the root which is of degree at least 2. An internal node $x \neq r(T)$ of a tree T is *binary* if and only if x is of degree 3 and $r(T)$ is *binary* if and only if $r(T)$ is of degree 2. A tree T is said *binary* if and only if all its internal nodes are binary.

A *subtree* S of T is a tree such that $V(S) \subseteq V(T)$, $E(S) \subseteq E(T)$ and any edge of $E(S)$ connects two nodes of $V(S)$. A *chain* of T is a subtree C with a node set $V(C) = \{x_1, \cdots, x_k\}$ and an edge set $E(C) = \{e_1, \cdots, e_{k-1}\}$ such that for each $1 \leq i \leq k$, e_i is incident to x_i and x_{i+1}.

If T is an unrooted tree, *rooting* T requires choosing an internal node as the root, or creating a new node $r(T)$ on an edge $e = \{x,y\}$ of T, namely removing e and adding two edges $\{r(T), x\}$ and $\{r(T),y\}$. If T is a rooted tree then the *unrooted version* of T is simply T (ignoring the description of $r(T)$ as the root) if $r(T)$ is non-binary; otherwise it is the tree obtained from T by removing $r(T)$ and its two incident edges going to its neighbors u and v, and adding an edge between u and v.

For a rooted tree T, we denote by T_x the subtree of T rooted at $x \in V(T)$, i.e. the subtree of T containing all the descendants of x. We call $L(T_x)$ the *clade of* x. A clade is *non-trivial* if it corresponds to an internal node of T. We denote by $\mathcal{C}(T)$ the set of non-trivial clades of T. It can be seen as a subset of the power set of $\mathcal{L}$.

The *bipartition* of an unrooted tree T corresponding to an internal edge $e = \{x,y\}$ is the unordered pair of clades $L(T_x)$ and $L(T_y)$ where T_x and T_y are the two subtrees rooted respectively at x and y obtained by removing e from T. A bipartition is *non-trivial* if it corresponds to an internal edge of T, and trivial otherwise. We denote by $\mathcal{B}(T)$ the set of non-trivial bipartitions of T. Note that bipartitions are sometimes called *splits* in the literature.

4.3.1. The Robinson-Foulds Distance

Definition 4.3.1 (edit operations)**.** *Two edit operations on the edges of a tree T (rooted or unrooted) are defined as follows:*

- *Let $e = \{x,y\}$ be an internal edge of $E(T)$. An* edge contraction *$Cont(T,e)$ is an operation transforming the tree T into the tree T' obtained from T by removing the edge e of T and identifying x and y; in other words, T' is obtained by adding the*

edge $\{x, z\}$ *for each* $z \in Ch(y) \setminus \{x\}$*, and then removing* y *and its incident edges (including* $\{x,y\}$*).*

- *Let* x *be a non-binary internal node of* $V(T)$ *and* $X = \{y_1, \cdots, y_t\} \subsetneq Ch(x)$ *be a subset of* $Ch(x)$ *such that* $|X| \geq 2$*. A* node extension $Ext(T,x,X)$ *is an operation transforming the tree* T *into the tree* T' *obtained from* T *by removing the edges* $\{x,y_i\}$*, for* $1 \leq i \leq t$*, creating a node* y *and a new edge* $e = \{x,y\}$ *adjacent to* x*, and creating new edges* $\{y, y_i\}$*, for* $1 \leq i \leq t$*.*

The function $\delta(T_1,T_2)$ assigning to each pair of rooted or each pair of unrooted trees the length of a minimum sequence of edit operations transforming T_1 into T_2 has been shown to be a metric, called the *Edit distance* or *Robinson-Foulds* distance between T_1 and T_2 [**58**].

For unrooted trees T_1 and T_2, this distance corresponds to the symmetric difference between the bipartitions of the two trees. More precisely, $\delta(T_1,T_2) = |\mathcal{B}(T_1) \setminus \mathcal{B}(T_2)| + |\mathcal{B}(T_2) \setminus \mathcal{B}(T_1)|$. In fact, to transform T_1 into T_2, edit operations are needed on *bad edges* representing bipartitions which are not shared by the two trees, i.e. edges of T_1 (respec. T_2) defining bipartitions in T_1 (respec. T_2) which are not in $\mathcal{B}(T_2)$ (respec. in $\mathcal{B}(T_1)$). An edge which is not bad is said to be *good*. Terminal edges are always good.

In the case of rooted trees T_1 and T_2, the Robinson-Foulds distance, that we denote in this case $\delta_R(T_1,T_2)$, is usually defined in the literature as the symmetric difference between the clades of the two trees. More precisely, for two rooted trees T_1 and T_2, $\delta_R(T_1,T_2) = |\mathcal{C}(T_1) \setminus \mathcal{C}(T_2)| + |\mathcal{C}(T_2) \setminus \mathcal{C}(T_1)|$.

The only thing that can make bipartitions and clades differ in number is rooting into a bad edge. In this case, the same bipartition, corresponding to the two edges adjacent to the root, would be counted twice. The link between this distance, defined in terms of clades (that we write δ_R) and the edit distance (that we write δ), has been established through the defined relation between the bipartition system (or split system) and the clade system (or cluster system) [**21**].

Although our extended distance is more likely useful for rooted trees, algorithmic analyses are simpler for unrooted trees, as in this case all internal nodes can be treated in the same way. Here, we make the link between the rooted and unrooted case, and then focus, for the rest of the paper, on unrooted trees.

Let T^r be a rooted version of an unrooted tree T, with a binary root. Denote by e_1, e_2 the two edges adjacent to $r(T^r)$. As e_1 and e_2 define the same bipartition of $\mathcal{B}(T)$, these edges are either both good or both bad. These notations are used in the following lemma.

Lemma 4.3.2 (Link between rooted and unrooted trees)**.** *Let* T_1 *and* T_2 *be two unrooted trees, and* T'_1*, respectively* T'_2*, be a rooting of* T_1*, respectively* T_2*.*

- *If* T'_1 *and* T'_2 *are both rooted into existing nodes of* T_1 *and* T_2 *or both rooted into good edges of* T_1 *and* T_2*, then* $\delta_R(T'_1,T'_2) = \delta(T_1,T_2)$*;*

- *If T_1' and T_2' are both rooted into bad edges of T_1 and T_2, then $\delta_R(T_1',T_2') = \delta(T_1,T_2)+2$;*
- *If exactly one among T_1' and T_2' is rooted into a bad edge of T_1 or T_2, then $\delta_R(T_1',T_2') = \delta(T_1,T_2) + 1$.*

The edit distance between two trees (rooted or unrooted) can be computed in linear time with the algorithm proposed by Day [**19**] in 1984. Our goal is to extend this distance to labeled trees.

4.3.2. Labeled Trees

Given a finite set of labels Λ, T *is labeled* if and only if each internal node x of T has a unique label $\lambda(x) \in \Lambda$.

Contraction and extension operations are generalized to labeled trees as follows: The node y created from an edge extension $Ext(T,x,X)$ is such that $\lambda(y) = \lambda(x)$; an edge contraction is only defined on edges $\{x, y\}$ for which $\lambda(x) = \lambda(y)$. It follows that a third edit operation should be introduced for labeled trees. Let x be a node of a labeled tree T with label $\lambda = \lambda(x)$. A *node flip* $Flip(x, \lambda')$ is an operation assigning a new label λ' to x, i.e. a label $\lambda' \in \Lambda$ such that $\lambda' \neq \lambda$. Those operations are depicted in Figure 4.1.

A node flip is required before contracting a *mixed edge*, i.e. an edge with its two extremities being differently labeled. A tree is said to be a *mixed tree* if all its edges are mixed edges.

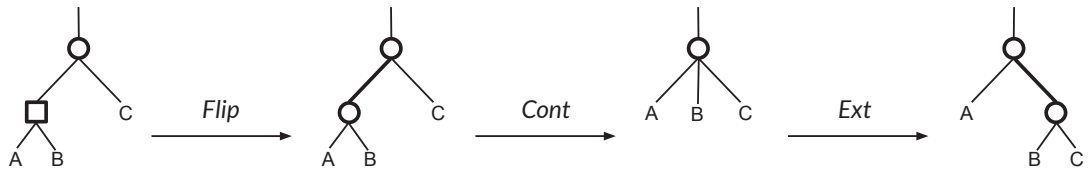


Fig. 4.1. The three edit operations defined for labeled trees. From left to right: Flip, Contraction and Extension.

Let $\mathcal{T}$ be the set of trees on $\mathcal{L}$, all trees being of the same type, i.e. all rooted or unrooted, all labeled or unlabeled. The following lemma (holding for all these cases) shows that introducing the flip operation does not prevent δ from being a distance.

Lemma 4.3.3 (Edit distance)**.** *The function $\delta(T_1, T_2)$ assigning to each pair $(T_1, T_2) \in \mathcal{T}^2$ the minimum length of a sequence of edit operations transforming T_1 into T_2 defines a distance on $\mathcal{T}$.*

In this paper, Λ is restricted to two labels. They are illustrated by a circle and a square in Figure 4.2. The two labels can, for example, represent speciation and duplication events. Notice however that labeling is not constrained to be consistent with a species tree [**35, 45**]. In other words, the intermediate trees in an optimal path transforming a tree to another are not required to be feasible according the the speciation/duplication labeling. Algorithmic

analyses are made independently of the nature of the two node labels. However, for notation purpose, we write $\Lambda = \{Spe, Dup\}$.

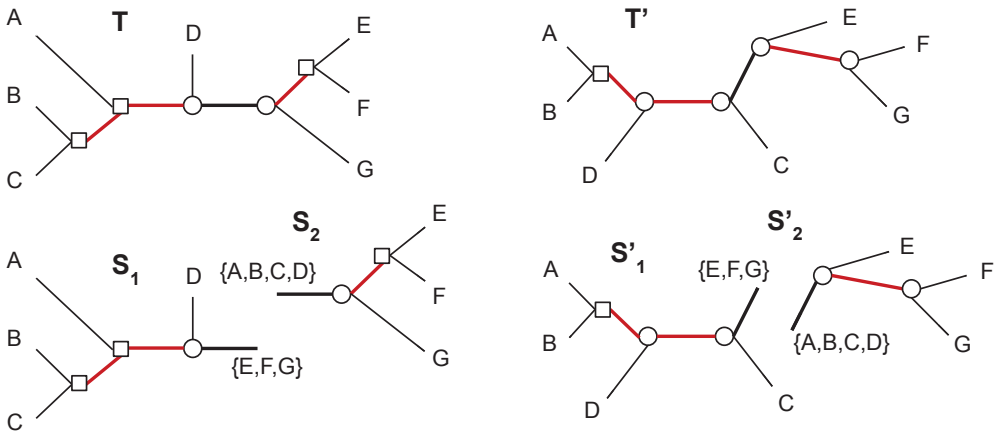


Fig. 4.2. Two unrooted and labeled trees T and T' on $\mathcal{L} = \{A, B, C, D, E, F, G\}$. The square and circle symbols represent the two possible labels for an internal node. Bad edges are red and good ones are black. $\{S_1, S_2\}$ are the maximal bad subtrees of T and $\{S'_1, S'_2\}$ the corresponding subtrees of T'.

4.4. Results on Labeled Trees

We focus now on unrooted trees. Using Lemma 4.3.2, our results can then be easily extrapolated to rooted trees. Consider $\mathcal{T}$ as the set of unrooted and labeled trees on $\mathcal{L}$. The goal is to compute the edit distance $\delta(T,T')$ for any pair T, T' of trees of $\mathcal{T}$, that is the number of operations in an *optimal sequence*, i.e a sequence of edit operations of minimum length transforming T into T'.

4.4.1. Reduction to Maximal Bad Subtrees

Let S be a subtree of T. Let $\{e_i = \{x_i,y_i\},$ for $1 \leq i \leq k\}$ be the set of terminal edges of S, with each y_i being a leaf of S, and $\{X_i, Y_i\}$ being the bipartition corresponding to e_i. Each leaf y_i of S is said to be *mapped* to Y_i. Notice that $\cup_{1\leq i\leq k} Y_i = \mathcal{L}$.

We say that S is a *bad subtree of* T if and only if S contains only bad edges, except the terminal edges of S which are all good edges of T. In other words, S is maximal in the sense that no more bad internal edges can be added into it. Intuitively, S can be obtained by taking a subtree with only bad edges, and adding edges adjacent to bad edges of S iteratively until the process stops. As a result, every terminal edge e_i of S will be good, i.e. there is an edge $e'_i = \{x'_i, y'_i\}$ in T' corresponding to $e_i = \{x_i,y_i\}$, that determine the same bipartition $\{X_i, Y_i\}$. Note that a maximal bad subtree may contain no bad edge at all (i.e. it is a star tree centered on good edges).

Lemma 4.4.1 (Pairs of maximal bad subtrees)**.** *Let S be a maximal bad subtree of T with the set $\{e_i\}_{1\leq i\leq k}$ of terminal edges, and let $\{e'_i\}_{1\leq i\leq k}$ be the corresponding set of edges in T'. Then the subtree S' of T', containing all e'_i edges as terminal edges, is unique. Moreover, it is a maximal bad subtree of T'.*

Let $\{S_1, S_2, \cdots, S_k\}$ be the set of maximal bad subtrees of T and $\{S'_1, S'_2, \cdots, S'_k\}$ be the corresponding subtrees of T' (see Figure 4.2 for an example). For $1 \leq i \leq m$, let $\mathcal{P}_i$ be an optimal sequence transforming S_i into S'_i. Then the sequence $\mathcal{P}$ obtained by performing consecutively $\mathcal{P}_1, \mathcal{P}_2, \cdots, \mathcal{P}_m$ transforms T into T'.

Although the traditional RF distance can be deduced from the above observation, in our case such a sequence is not necessarily optimal. In fact, in contrast with unlabeled trees, optimal sequences for labeled trees may involve contracting good edges, as illustrated in Figure 4.3.

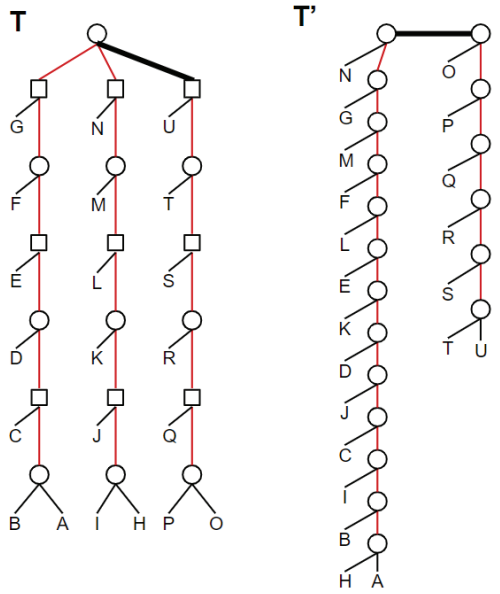


Fig. 4.3. Example where the minimal edit path requires contracting a good edge: if we contract the internal good edge of T (the bold one), then the 3 subtrees of T can be handled together, requiring 6 node flips and 18 edge contractions to reduce T into a star tree, and then 18 edge extensions to reach T', leading to 42 operations in total. By contrast, if we do not contract the good edge of T, then the two subtrees of T separated by this edge should be handled separately, requiring 9 flips, 17 edge contractions and 17 edge extensions to reach T', leading to 43 operations in total. The first scenario is the better one.

4.4.2. Reduction to Mixed Bad Subtrees

In the next section, we will describe an exact algorithm for optimally contracting a mixed tree. Before reaching this step, the question is how to obtain such a tree. The next lemma shows that non-mixed bad edges can be contracted first. The idea of the proof is that any

optimal solution must eventually contract a non-mixed bad edge $\{x,y\}$. We can thus contract $\{x,y\}$ first into a single node z, and "reproduce" all the events of the optimal solution by treating z as either x or y.

Lemma 4.4.2 (Contract non-mixed bad edges)**.** *Let e be any non-mixed bad edge of T, and let T_c be the tree obtained from T by contracting e. Then $\delta(T_c,T') = \delta(T,T') - 1$.*

According to this lemma, we can safely start by contracting all non-mixed bad edges of T and T' first, since there is always an optimal sequence of edit operations that also does this. The resulting trees T_c and T'_c can then be subdivided into pairs of maximal bad subtrees, all such bad subtrees being mixed subtrees.

4.5. Algorithms

We first consider a general framework which entails performing all required edge contractions first, and then all node extensions.

Methodology 1 *(T, T')*

Contract non-mixed bad edges of T and T', leading to T_c and T'_c;
for each pair S,S' of maximal bad subtrees of T_c,T'_c **do**
  Perform a sequence of flip and contraction operations leading from S to a star tree S_*;
  Perform a sequence of flip and extension operations leading from S_* to S';
end for

This general framework leads to the following upper bound for $\delta(T,T')$.

Lemma 4.5.1 (Upper bound δ)**.** *Let T and T' be two unrooted and labeled trees with n internal nodes each and let e (resp. e') be the number of internal bad edges of T (resp. T'). Then $\delta(T,T') \leq e + e' + n$.*

Notice that if both T and T' are binary, then $e = e'$. Moreover, in this this case $2e + n$ is a tight bound as it can be reached in some cases (see an example in Figure 4.4).

The first step of Methodology 1 leads to a star tree T_*. Instead of then extending nodes to reach T', a symmetric way would be to transform T' into a star tree T'_*. The difference between T_* and T'_* may be in the label of the single node of each of these trees, which would then need an additional flip operation to reconstruct a corresponding path from T to T'. This second methodology is given below, where *Contract-Tree*(T,T_*) takes as input a tree T and returns a sequence of operations *contracting a tree* T, i.e. transforming T into a star tree, and the star tree T_* resulting from this optimal contraction.

Methodology 2 is clearly simpler to handle and will be explored in the next section. The next lemma shows that it may overestimate an optimal sequence returned by Methodology 1 by at most one operation for each pair of maximal bad subtrees.

Lemma 4.5.2 (Compare Meth.1 and Meth.2)**.** *Let S and S' be a pair of maximal bad subtrees of T_c and T'_c, obtained similarly by Methodology 1 and Methodology 2. Let $M_1(S,S')$ (respec.*

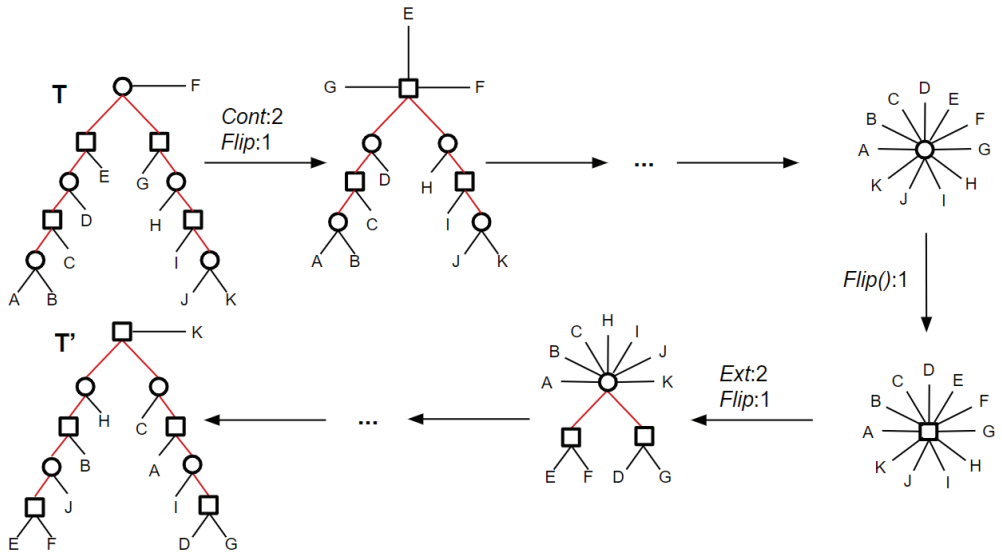


Fig. 4.4. A pair of unrooted mixed trees (T,T'), both with eight internal edges and nine internal nodes. All their internal edges are bad edges (red edges). Here $\delta(T,T') = 25 = 2 \cdot 8 + 9 = 2 \cdot e + n$.

Methodology 2 *(T, T')*

Contract non-mixed bad edges of T and T', leading to T_c and T'_c;
for each pair S,S' of maximal bad subtrees of T_c,T'_c **do**
  Contract-Tree(S, S_*);
  Contract-Tree(S', S'_*);
  Perform a final flip if required;
end for

$M_2(S,S')$) be the number of operations performed by the `for` *loop of Methodology 1 (respec. Methodology 2). Moreover, let S_* (respec. S'_*) be the star tree returned by* Contract-Tree *on S (respec. on S').*

(1) If $S_* = S'_*$ (same node label), then $M_2(S,S') = M_1(S,S')$;
(2) Otherwise, $M_1(S,S') \leq M_2(S,S') \leq M_1(S,S') + 1$

4.5.1. An Optimal Algorithm for Contracting a Tree

The remaining problem is the one of finding an optimal sequence of contraction and flip operations contracting a mixed tree T. For any such sequence, the number of contraction operations is just the number of internal edges of T. Therefore, the problem reduces to finding the minimum number of flip operations $\phi(T)$ in such an optimal sequence. Notice that the problem does not reduce to performing the minimum number of flips leading to the same label for all nodes, which would just be $min\{nb_{spe}, nb_{dup}\}$ with nb_{spe} (respec. nb_{dup})

being the number of *Spe* (respec. *Dup*) nodes of T. For example, for the tree T of Figure 4.3, $min\{nb_{spe}, nb_{dup}\} = 9$. However, proceeding by an alternating sequence of flip and contraction operations (the top node flipped to *Dup*, then the three top edges contracted, then the next top node flipped to a *Spe* node, then the three top edges contracted, etc.) leads to a total of 6 flips rather than 9.

We will proceed iteratively by starting a sequence of contraction operations from the center of a tree T, i.e. the midpoint of the longest mixed chain of T. The *diameter*, denoted $diam(T)$, of a tree T is the length of its longest chain (determined in terms of the number of edges). Note that any longest chain in a tree has two leaves at its extremities, as otherwise we could extend the chain. Assume that T has at least two terminal edges, so that $diam(T) \geq 2$. We show that $\phi(T)$ is equal to $\lceil diam(T)/2 \rceil - 1$. For a node v, let $ecc_T(v)$ denote the maximum distance from v to a leaf of T (this is known as the *eccentricity* of v) [1].

Lemma 4.5.3 (Optimal path contracting a mixed tree)**.** *The minimum number of flips in an optimal sequence of operations transforming a mixed tree T into a star tree is $\lceil diam(T)/2 \rceil - 1$.*

Algorithm Contract-Tree(T) (where T is a mixed tree)

```
Let P = (w_1, w_2, ..., w_k) be a longest chain of T;
Let w = w_⌈k/2⌉ be a midpoint of P; (w has minimum eccentricity)
while w has a non-leaf neighbor do
   Flip w;
   Contract the internal edges incident to w;
end while
```

Lemma 4.5.3 immediately lead to Algorithm *Contract-Tree*. The fact that the algorithm contracts T into a star tree using $\phi(T)$ flips follows from the proof of Lemma 4.5.3.

Theorem 4.5.4. *For T being a mixed tree, Algorithm Contract-Tree returns the length of an optimal sequence of operations contracting T.*

One should note that if T has even diameter, then there are two possible midpoints, i.e. two nodes with minimum eccentricity. This means that it is possible to choose the label of the internal node of the resulting star tree. This guarantees that when contracting a pair of bad subtrees T and T', we can always avoid a final flip by choosing the appropriate final label if either T or T' has even diameter. We cannot guarantee that this final flip is avoidable if both subtrees have odd diameter.

We now show that Methodology 2 has a guaranteed approximation ratio of 2 when using Algorithm *Contract-Tree* as a subroutine. The idea behind the approximation is to show that any optimal solution must contract all the bad edges and perform at least one flip or good edge contraction per bad subtree. Our algorithm only contracts bad edges, and we can

[1] The radius of T is a well-known graph parameter and is defined as the minimum eccentricity of a node of T. In a tree, the radius turns out to be $\lceil diam(T)/2 \rceil$.

show that the number of flips performed is at most the number of bad edges plus twice the number of bad subtrees.

Theorem 4.5.5 (Upper bound Meth.2)**.** *Let d be the number of operations performed by Methodology 2 when tree contractions are done by Algorithm* Contract-Tree*. Then $d \leq 2\delta(T, T')$.*

4.6. Experimental Results

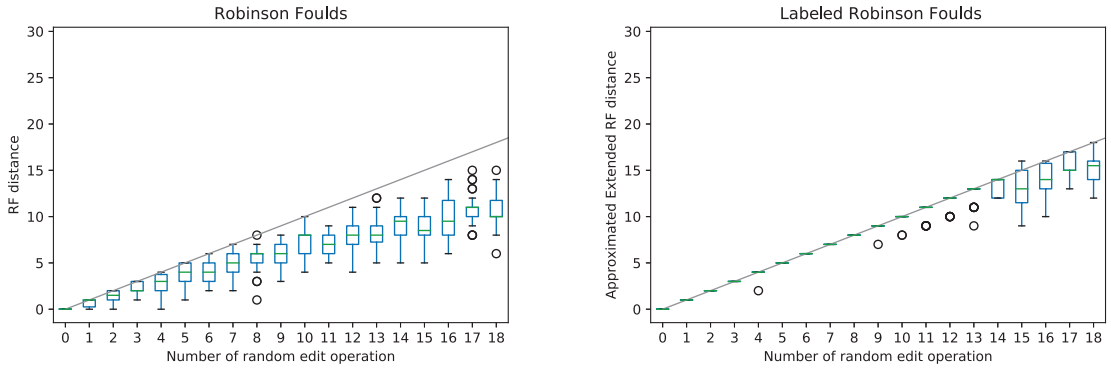


Fig. 4.5. Empirical comparison of the distance inferred for an increasing number of random edit operations (contraction, extensions, and flips), using the classical Robinson-Foulds distance (left) and *Contract-Tree* algorithm (right). Because the former ignores node labels, it grossly underestimates the actual number of edits. Our algorithm tracks more closely the actual number of edits.

We implemented a heuristic following Methodology 2, using the *Contract-Tree* algorithm. To test it on simulated data, we retrieved the TP53 gene family from Ensembl release 96 (542 genes), including the speciation and duplication labels, and introduced an increasing number of random edit operations, on 30 replicates. A random edit was introduced as follows: with probability 0.3, the label of one random internal node was flipped; the rest of the probability mass function was evenly distributed among all internal edges connecting nodes of the same type (which could be potentially contracted) and all nodes of degree > 3 (in which a new edge could potentially be expanded).

After each edit, we computed the classical *RF* distance and its extension to labeled trees using our heuristic (Fig. 4.5). Because it accounts for labels, the latter tracked more closely the true number of edits. At the same time, the estimated distances were never higher than the actual number of edits, which suggests that the heuristic can identify a minimum edit path when the total number of edit operations is relatively low. The implementation, including the function to mutate labeled trees, is available as an open source Python library (PyPI package `pylabeledrf`, also available at `https://github.com/DessimozLab/pylabeledrf`).

4.7. Discussion

In this paper, we have considered what we thought was the simplest and most natural extension of the Robinson-Foulds distance to labeled trees. Although its theoretical complexity is unknown and remains an open problem, this extension appears to be much harder to compute than the classical RF distance for unlabeled trees.

Despite the optimality of Algorithm *Contract-Tree* for contracting a mixed tree, neither *Methodology 1*, nor *Methodology 2* are guaranteed to lead to an optimal solution. This is due to two main reasons. The first one is that, as shown in Figure 4.3, an optimal path contracting a tree T may require contracting good edges, i.e. edges common to both trees, which is not the case for unlabeled trees. The second reason is that an optimal path from a tree T to a tree T' may not be one with all edge contraction events preceding all edge extension. An example, given in Figure 4.6, shows that it may be better to convert a given bad edge into a good edge rather than contracting all bad edges. It can be observed from this example that going from T to T' following the red path entails performing a nearest-neighbour interchange (NNI) operation on the edge e of T. A future direction for improving the algorithm will be to consider such "safe" edges, i.e. edges admitting an NNI leading to a bipartition of the target tree.

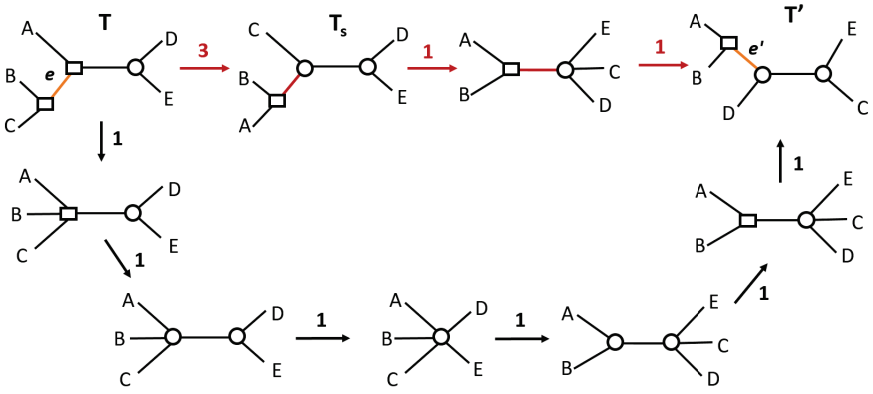


Fig. 4.6. An optimal path from T to T' following *Methodology 1* is depicted by black arrows and involves 6 operations. It is not optimal as another path, depicted by red arrows, involves only five operations. The path of length 3 from T to T_s acts on the safe edge, represented in orange. This path involves an edge contraction, an edge extension and a flip, leading to the good edge (red edge) in T_s.

Still, we have implemented a heuristic which constitutes a better baseline solution to quantifying differences between labeled tree topologies than the conventional RF measure, which is blind to labels. For instance, this implementation could be useful in the context of orthology benchmarking, to compare inferred labeled trees with reference curated ones [**4**].

Looking ahead, we envision several potential future directions. We see potential in identifying the good edges that should be contracted and characterizing classes of trees that may

be resolved optimally. In particular, it would be interesting to restrict the study to the class of labeled trees consistent with a species tree (which is not the case of the trees of Figure 4.3).

Another direction would be to consider an alternative extension of the RF distance. In this paper, edge contraction and edge extension, the two edit operations defining the classical RF, were re-defined in the context of labeled nodes, by constraining them to occur on edges with the same labels on their extremities. Another direction would be to consider edit operations on nodes, as for the Tree Edit Distance (TED) for hierarchical trees, i.e. node deletion, insertion and relabeling. In addition to the theoretical complexity and computational efficiency, it would be important to evaluate the robustness of these two RF extensions with respect to small changes in the topology or tree labeling. Although we do not expect robustness to be much better than the classical RF, knowing which extension is better can orient the study towards future improvements. Finally another direction would be to extend the study to an arbitrary set of possible labels.

More generally, we think that computing the distance between labeled trees conceals many new problems and opens a variety of new algorithmic directions.

Chapter 5

A Linear Time Solution to the Labeled Robinson-Foulds Distance Problem

The article presented in this chapter is in preparation for submission. This chapter is in the continuity of our work presented in the previous chapter. It describes the "Labeled Robinson-Foulds" distance, a new metric to account for internal node labels for the comparison of labeled gene trees.

We previously extended the Robinson-Foulds distance to trees with labeled internal nodes by extending the Robinson-Foulds edit operations on tree edges. The extension was an initiative to widen the applicability of pairwise measures of dissimilarity for comparing trees with information on internal nodes, as very few measures have been designed for such purpose. Unfortunately, the Robinson-Foulds' most attractive trait, which is to be computable in linear time, was lost in the process. In this chapter, we study a different approach based on edit operations on nodes, in an attempt to address some of the weaknesses of the algorithm presented in the previous chapter, more specifically, in terms of precision and time-efficiency. We also implemented an exact linear time algorithm useful for preliminary analysis and comparison of labeled reconciled gene trees.

Contributions: Samuel Briand, Christophe Dessimoz, and Nadia El-Mabrouk participated collectively in the development of the proofs and algorithms included in the article. Samuel Briand, Christophe Dessimoz, Nadia El-Mabrouk, and Yannis Nevers wrote the manuscript. Samuel Briand designed and implemented the first version of the distance software. Christophe Dessimoz and Yannis Nevers implemented the final version of the distance software, and performed the experiments presented in the manuscript. All authors have read and approved the final manuscript.

A Linear Time Solution to the Labeled Robinson-Foulds Distance Problem

Samuel Briand[1], Christophe Dessimoz[2,3,4,5,6], Nadia El-Mabrouk[1], and Yannis Nevers[2,3,6]

1. DIRO, Université de Montréal
2. Department of Computational Biology, University of Lausanne
3. Center for Integrative Genomics, University of Lausanne
4. Centre for Life's Origins and Evolution, Genetics Evolution and Environment, University College London
5. Department of Computer Science, University College London
6. SIB Swiss Institute of Bioinformatics

5.1. Abstract

Motivation: Comparing trees is a basic task for many purposes, and especially in phylogeny where different tree reconstruction tools may lead to different trees, likely representing contradictory evolutionary information. While a large variety of pairwise measures of similarity or dissimilarity have been developed for comparing trees with no information on internal nodes, very few measures have been designed for node labeled trees, which is for instance the case of reconciled gene trees. Recently, we proposed a formulation of the Labeled Robinson Foulds edit distance with edge extensions, edge contractions between identically labeled nodes, and node label flips. However, this distance proved difficult to compute, in particular because shortest edit paths can require contracting "good" edges, i.e. edges present in the two trees.

Results: Here, we report on a different formulation of the Labeled Robinson Foulds edit distance — based on node insertion, deletion and label substitution — which we show can be computed in linear time. The new formulation also maintains other desirable properties: being a metric, reducing to Robinson Foulds for unlabeled trees and maintaining an intuitive interpretation. The new distance is computable for an arbitrary number of label types, thus making it useful for applications involving not only speciations and duplications, but also horizontal gene transfers and further events associated with the internal nodes of the tree. To illustrate the utility of the new distance, we use it to study the impact of taxon sampling

on labeled gene tree inference, and conclude that denser taxon sampling yields better trees.

Availibility and implementation: The software written in Python is available in the pylabeledrf repository at `https://github.com/DessimozLab/pylabeledrf`.

5.2. Introduction

Gene trees are extensively used, not only for inferring phylogenetic relationships between corresponding taxa, but also for inferring the most plausible scenario of evolutionary events leading to the observed gene family from a single ancestral gene copy. This has important implications towards elucidating the functional relationship between gene copies. For this purpose, reconciliation methods (reviewed in Boussau and Scornavacca, 2020) [**7**] embed a given gene tree into a known species tree. This process results in the labeling of the internal nodes of the gene tree with the type of events which gave rise to them, typically speciations and duplications, but also horizontal gene transfers or possibly other events (whole genome duplication, gene convergence, etc). For example, information on duplication and speciation node labeling is provided for the trees of the Ensembl Compara database [**73**].

The existence of a variety of different phylogenetic inference methods, leading to different, potentially inconsistent trees for the same dataset, brings forward the need for appropriate tools for comparing them. Although comparing labeled gene trees remains a largely unexplored field, a large variety of pairwise measures of similarity or dissimilarity have been developed for comparing unlabeled evolutionary trees. Among them are the methods based on counting the structural differences between the two trees in terms of path length, bipartitions or quartets for unrooted trees, clades or triplets for rooted trees [**12, 23, 16**], or those based on minimizing a number of rearrangements that disconnect and reconnect subpieces of a tree, such as nearest neighbour interchange (NNI), subtree-pruning-regrafting (SPR) or Tree-Bisection-Reconnection (TBR) moves [**39, 36, 3**]. While the latter methods are NP-hard [**46**], the former are typically computable in polynomial time. In particular, the Robinson-Foulds (RF) distance, defined in terms of bipartition dissimilarity for unrooted trees, and clade dissimilarity for rooted trees [**48**], can be computed in linear [**19**], and even sublinear time [**54**].

On the other hand, metrics have also been developed for node labeled trees (rooted, and sometimes with an order on nodes) arising from many different applications in various fields (parsing, RNA structure comparison, computer vision, genealogical studies, etc), where node labels in a given tree are pairwise different. The standard Tree Edit Distance (TED), defined in terms of a minimum cost path of node deletion, node insertion and node relabeling (label substitution) transforming one tree to another, has been widely used in this context for

comparing two trees sharing the same set of node labels (i.e. each label present exactly once in each tree). While the less constrained version of the problem on unordered labeled trees is NP-complete [**80**], most variants are solvable in polynomial time [**78, 79, 64**].

The metric we developed in Briand *et al.*(2020) [**8**], referred to as $ELRF$, is the first effort towards comparing labeled gene trees, expressed in terms of trees with a binary node labeling (typically speciation and duplication). $ELRF$ is an extension of the RF distance, one of the most widely used tree distance, not only in phylogenetics, but also in other fields such as in linguistics, for its computational efficiency, intuitive interpretation and the fact that it is a true metric. Improved versions of the RF distance have also been developed [**46, 49**] to address the distance drawbacks, which are lack of robustness (a small change in a tree may cause a disproportional change in the distance) and skewed distribution. Classically defined in terms of bipartition or clade dissimilarity, the RF distance can similarly be defined in terms of edit operations on tree edges: the minimum number of edge contraction and extension needed to transform one tree into the other [**58**]. In Briand *et al.*(2020) [**8**], this definition of the RF distance was extended to node labeled trees by including a node *flip* operation, alongside edge contractions and extensions. While remaining a metric, $ELRF$ turned out to be much more challenging to compute, even for binary node labels. As a result, only a heuristic could be proposed to compute it.

In this paper, we explore a different extension of RF to node labeled trees, directly derived from TED, which is a reformulation of the RF distance in terms of edit operations on tree nodes rather than on tree edges. We show that this distance is computable in linear time for an arbitrary number of label types, thus making it useful for applications involving not only speciations and duplications, but also horizontal gene transfers and further events associated with the internal nodes of the tree. We show that the new distance compares favourably to RF and $ELRF$ by performing simulations on labeled gene trees of 182 leaves. Finally, we use our new distance in the purpose of measuring the impact of taxon sampling on labeled gene tree inference, and conclude that denser taxon sampling yields better predictions.

5.3. Notation and Concepts

Let T be a tree with node set $V(T)$ and edge set $E(T)$. Given a node x of T, the *degree of* x is the number of edges incident to x. We denote by $L(T) \subseteq V(T)$ the set of *leaves of* T, i.e. the set of nodes of T of degree one. In particular, given a set $\mathcal{L}$ (let us say taxa or genetic elements), a tree T on $\mathcal{L}$ is a tree with leafset $L(T) = \mathcal{L}$.

A node of $V(T) \setminus L(T)$ is called an *internal node*. A tree with a single internal node x is called a *star tree*, and x is called a *star node*. An edge connecting two internal nodes is called an *internal edge*; otherwise, it is a *terminal edge*. Moreover, a *rooted tree* admits a

single internal node $r(T)$ considered as the root. Now an internal node x is *binary* if x is of degree 3 and $r(T)$ is *binary* if $r(T)$ is of degree 2.

Let x and y be two nodes of a rooted tree T; y is a *descendant* of x if y is on the path from x to a leaf (possibly y itself) of T. If T is rooted, we say that y is a *child* of x if $e = \{x,y\}$ is an edge of $E(T)$ with y being a descendant of x. If T is unrooted, we call the set $\{y : \{x,y\} \in E(T)\}$ the set of children of x. For a rooted or an unrooted tree T, we denote by $Ch(x)$ the set of children of an internal node x of T.

A *subtree* S of T is a tree such that $V(S) \subseteq V(T)$, $E(S) \subseteq E(T)$ and any edge of $E(S)$ connects two nodes of $V(S)$. For a rooted tree T, we denote by T_x the subtree of T rooted at $x \in V(T)$, i.e. the subtree of T containing all the descendants of x. We call $L(T_x)$ the *clade of* x.

The *bipartition* of a tree T corresponding to an edge $e = \{x,y\}$ is the unordered pair of clades $L(T_x)$ and $L(T_y)$ where T_x and T_y are the two subtrees rooted respectively at x and y obtained by removing e from T. We denote by $\mathcal{B}(T)$ the set of non-trivial bipartitions of T, i.e. those corresponding to internal edges of T.

5.3.1. The Robinson-Foulds Distance

Given two unrooted trees T and T' on the leafset $\mathcal{L}$, the Robinson-Foulds (RF) distance between T and T' is the symmetric difference between the bipartitions of the two trees. More precisely,

$$RF(T,T') = |\mathcal{B}(T) \setminus \mathcal{B}(T')| + |\mathcal{B}(T') \setminus \mathcal{B}(T)|$$

In the case of rooted trees, the RF distance is defined as the symmetric difference between the clades of the two trees.

As recalled in Briand *et al.*(2020) [**8**], the RF distance is equivalently defined in terms of an edit distance on edges. However, as for labeled trees an additional substitution operation on node labels will be required, for the sake of standardization, we reformulate the edit operations to operate on nodes rather than on edges.

Definition 5.3.1 (node edit operations)**.** *Two edit operations on the nodes of a tree T (rooted or unrooted) are defined as follows:*

- ***Node deletion:*** *Let x be an internal node of T which is neither the root nor a star node, and let y be the parent of x if T is rooted, or y be a given child of x which is not a leaf if T is unrooted (such an y exists from the fact that x is not a star node). Deleting x means making the children of x become the children of y. More precisely, $Del(T,x,y)$ is an operation transforming the tree T into the tree T' obtained from T by removing the edge $\{x,z\}$ for each $z \in Ch(x)$, creating the edge $\{y,z\}$ for each $z \in Ch(x) \setminus \{y\}$, and then removing node x.*

- ***Node insertion:*** *Let y be a non-binary internal node of $V(T)$. Inserting x as a child of y entails making x the parent of a subset $Z \subsetneq Ch(y)$ such that $|Z| \geq 2$. More precisely, $Ins(T,x,y,Z)$ is an operation transforming the tree T into the tree T' obtained from T by removing the edges $\{y,z_i\}$, for all $z_i \in Z$, creating a node x and a new edge $e = \{x,y\}$, and creating new edges $\{x, z_i\}$, for all $z_i \in Z$.*

Notice the one-to-one correspondence between operations on nodes and operations on edges. In fact, deleting a node x by an operation $Del(T,x,y)$ results in deleting the edge $\{x,y\}$, while inserting a node x by an operation $Ins(T,x,y,Z)$ results in inserting the edge $\{x,y\}$. Here, we define the RF distance in terms of edit operations on nodes. This definition is equivalent to the more classical formulation in terms of edit operations on edges. Formally, let T and T' be two trees on the same leafset $\mathcal{L}$. The *Robinson-Foulds* or *Edit distance* [**58**] $RF(T,T')$ between T and T' is the length of a shortest path of node edit operations transforming T into T'. This distance measure, equivalently defined as the symmetrical difference between the bipartitions of the two trees in case of unrooted trees, or the symmetrical difference between the clades of the two trees in case of rooted trees, has been shown to be a metric.

Call a *bad edge* of T with respect to T' (or similarly of T' with respect to T; if there is no ambiguity, we will omit the "with respect to" precision) an edge representing bipartitions which are not shared by the two trees, i.e. an edge of T (respec. T') defining a bipartition of $\mathcal{B}(T)$ (respec. $\mathcal{B}(T')$) which is not in $\mathcal{B}(T')$ (respec. in $\mathcal{B}(T)$). An edge which is not bad is said to be *good*. Terminal edges are always good. The only thing that can make bipartitions and clades differ in number is rooting into a bad edge. In that case, the same bipartition, corresponding to the two edges adjacent to the root, would be counted twice. Given two rooted trees, their RF distance can then be deduced from the RF distance of the "unrooted version" of the two trees by applying Lemma 1 in Briand *et al.*(2020) [**8**].

In this paper, we focus on unrooted trees, thus avoiding the special case of the root. Therefore, for now on, all trees are considered unrooted.

5.4. Generalizing the Robinson-Foulds Distance to Labeled Trees

A tree T is *labeled* if and only if each internal node x of T has a label $\lambda(x) \in \Lambda$, Λ being a finite set of labels. For gene trees, labels usually represent the type of event leading to the bifurcation, typically duplications and speciations, although other events, such as horizontal gene transfers, may be considered. The metric defined in this paper works for an arbitrary number of labels. We generalize the RF distance to labeled trees by generalizing the edit operations defined above. This is simply done by introducing a third operation for node labels editing.

Definition 5.4.1 (Labeled node edit operations)**.** *Three edit operations on internal nodes of a labeled tree T are defined as follows:*

- ***Node deletion:*** *$Del(T,x,y)$ is an operation deleting an internal node x of T with respect to a child y of x which is not a leaf, defined as in Definition 5.3.1.*
- ***Node insertion:*** *$Ins(T,x,y,Z,\lambda)$ is an operation inserting an internal node x as a new child of a non-binary node y, and moving $Z \subsetneq Ch(y)$ such that $|Z| \geq 2$, to be the children of x, as defined in Definition 5.3.1. In addition, the inserted node x receives a label $\lambda \in \Lambda$.*
- ***Node label substitution:*** *$Sub(T,x,\lambda)$ is an operation substituting the label of the internal node x of T with $\lambda \in \Lambda$.*

These operations are illustrated in Figure 5.1.

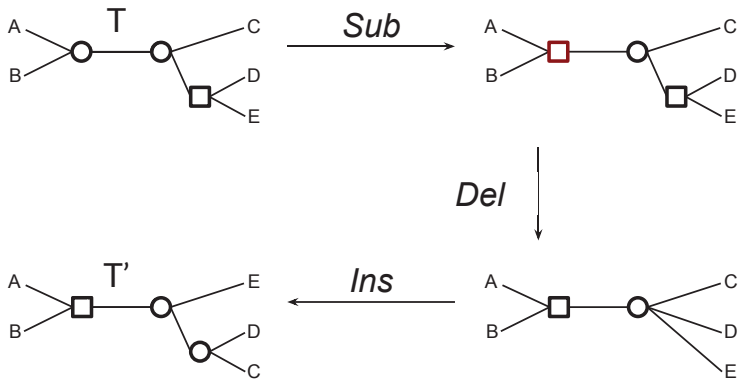


Fig. 5.1. The transformation of a tree T into a tree T' depicting the three edit operations on nodes. From top to bottom: node label substitution (leading to the red label), node deletion (the parent of D and E) and node insertion (the parent of D and C).

Let $\mathcal{T}_\mathcal{L}$ be the set of unrooted and labeled trees on the leafset $\mathcal{L}$. For two trees T, T' of $\mathcal{T}_\mathcal{L}$, we call the *Labeled Robinson Foulds* distance between T and T' and denote $LRF(T,T')$ the length of a shortest path of labeled node edit operations transforming T into T' (or vice versa). The two following lemma state that, similarly to RF, LRF is a true metric. Moreover, LRF is exactly RF for unlabeled trees (or similarly labeled with a single label).

In the following the *unlabeled version of* a tree $T \in \mathcal{T}_\mathcal{L}$ is simply T ignoring its node labels.

Lemma 5.4.2. *The function $LRF(T,T')$ assigning to each pair $(T,T') \in \mathcal{T}_\mathcal{L}^2$ the length of a shortest path of node edit operations transforming T into T' defines a distance on $\mathcal{T}_\mathcal{L}$.*

Proof. The non-negative and identity conditions are obvious. For the symmetric condition, notice that we can reverse every edit operation in a path from T to T' to obtain a path from T' to T with the same number of events, and vice versa (insertions and deletions are symmetrical operations, and any substitution can be reversed by a substitution). We thus have $LRF(T',T) \leq LRF(T,T')$ and $LRF(T,T') \leq LRF(T',T)$, and equality follows.

Finally, we prove the triangular inequality condition: for three trees T, T' and T'', to transform T into T', we may take any path of edit operations from T to T'', followed by any path of edit operations from T'' to T'. It follows that $LRF(T,T') \leq LRF(T,T'') + LRF(T'',T')$. □

Lemma 5.4.3. *If Λ is restricted to a single label, then for each pair $(T,T') \in \mathcal{T}_{\mathcal{L}}^2$, $LRF(T,T') = RF(T,T')$.*

Proof. Let l be the only label of Λ. Let $\mathcal{P}$ be a path of node edit operations transforming the unlabeled version of T into the unlabeled version of T', such that $|\mathcal{P}| = RF(T,T')$. Labeling by l each inserted node leads to a corresponding path of labeled node edit operations transforming T into T', and thus $LRF(T,T') \leq RF(T,T')$.

Conversely, Let $\mathcal{P}$ be a path labeled node edit operations transforming T into T', such that $|\mathcal{P}| = LRF(T,T')$. As a single label exists, node substitutions are not defined, and thus $\mathcal{P}$ is restricted to a set of node insertion and deletion transforming T into T', and thus *a fortiori* the unlabeled version of T into the unlabeled version of T'. Thus $RF(T,T') \leq LRF(T,T')$, which completes the proof. □

A previous extension of RF to labeled trees, based on edit operations on edges rather than on nodes, was introduced in Briand *et al.*(2020) [**8**]. This distance, which we call $ELRF$, was defined on three operations:

- Edge extension $Ext(T,x,X)$ creating an edge $\{x,y\}$ and defined as a node insertion $Ins(T,y,x,X,\lambda(x))$ inserting a node y as a child of x and assigning to y the label of x;
- Edge contraction $Cont(T,\{x,y\})$ similar to a node deletion $Del(T,y,x)$ deleting y, but only defined if $\lambda(x) = \lambda(y)$;
- Node flip $Flip(x,\lambda)$ assigning the label λ to x.

Given two labeled trees T and T' of $\mathcal{T}_{\mathcal{L}}$, $ELRF(T,T')$ is the length of the shortest path of edge extension, edge contraction and label flip required to transform T to T'.

The following lemma makes the link between LRF and $ELRF$.

Lemma 5.4.4. *For any pair $(T,T') \in \mathcal{T}_{\mathcal{L}}^2$,*

$$LRF(T,T') \leq ELRF(T,T')$$

Proof. Let $\mathcal{P}$ be a path of edge edit operations and label flip transforming T into T' such that $|\mathcal{P}| = ELRF(T,T')$. Then the sequence $\mathcal{P}'$ obtained from $\mathcal{P}$ by replacing each edge extension by the corresponding node insertion, each edge contraction by the corresponding node deletion and each node flip by the corresponding node substitution is clearly a path of node edit operations of length $|\mathcal{P}'| = |\mathcal{P}| = ELRF(T,T')$ transforming T into T'. And thus $LRF(T,T') \leq ELRF(T,T')$. □

The rest of this paper is dedicated to computing the edit distance $LRF(T,T')$ for any pair (T, T') of trees of $\mathcal{T}_{\mathcal{L}}$.

5.4.1. Reduction to Islands

In this section, we define a partition of the two trees into pairs of maximum subtrees that can be treated separately.

While a good edge e of T has a corresponding good edge e' in T' (the one defining the same bipartition), a bad edge in T has no corresponding edge in T'. However, these edges may be grouped into pairs of corresponding *islands* (called maximum bad subtrees in Briand *et al.*(2020) [**8**]), as defined bellow.

Definition 5.4.5 (Islands)**.** *An* island *of T is a maximum subtree (i.e. a subtree with a maximum number of edges) I of T such that I contains no internal edge which is a good edge of T, and all terminal edges of I are good edges of T. The* size *of I, denoted $\epsilon(I)$, is its number of internal edges.*

In other words, an island of T is a maximum subtree with all internal edges (if any) being bad edges of T, and all terminal edges being good edges of T. Notice that an island I of T may have no internal edge at all, i.e. it may be a start tree (if $\epsilon(I) = 0$). Moreover, a tree T is "partitioned" into its set $\{I_1, I_2, \cdots I_n\}$ of islands in the sense that $\{V(I_1),V(I_2), \cdots V(I_n)\}$ is a partition of $V(T)$. Notice also that each bad edge of T belongs to a single island, while each good edge belongs to exactly two islands of T if it is an internal edge of T, or to a single island if it is a terminal edge of T.

Finally, the following lemma from Briand *et al.*(2020) [**8**] shows that there is a one-to-one correspondence between the islands of T and those of T'.

Lemma 5.4.6. *Let I be an island of T with the set $\{e_i\}_{1\leq i\leq k}$ of terminal edges, and let $\{e'_i\}_{1\leq i\leq k}$ be the corresponding set of edges in T'. Then the subtree I' of T', containing all e'_i edges as terminal edges, is unique. Moreover, it is an island of T'.*

For any island I of T, let I' be the corresponding island of T'. We call (I,I') an *island pair* of (T,T'). See Figure 5.2 for an example.

Now, let $\mathcal{I}_{(T,T')} = \{(I_1,I'_1), (I_2,I'_2), \cdots, (I_n,I'_n)\}$ be the set of island pairs of (T,T'). For $1 \leq i \leq n$, let $\mathcal{P}_i$ be a shortest path of labeled node edit operations transforming I_i into I'_i. Then the path $\mathcal{P}$ obtained by performing consecutively $\mathcal{P}_1, \mathcal{P}_2, \cdots, \mathcal{P}_n$ (that we represent later as $\mathcal{P}_1.\mathcal{P}_2.\cdots.\mathcal{P}_n$) clearly transforms T into T'. Therefore we have

$$LRF(T,T') \leq \sum_{i=1}^{n} LRF(I_i, I'_i)$$

As described in Briand *et al.*(2020)[**8**], one major issue with $ELRF$ is that good edge contractions may not be avoided in a shortest path of edit operations transforming T into T',

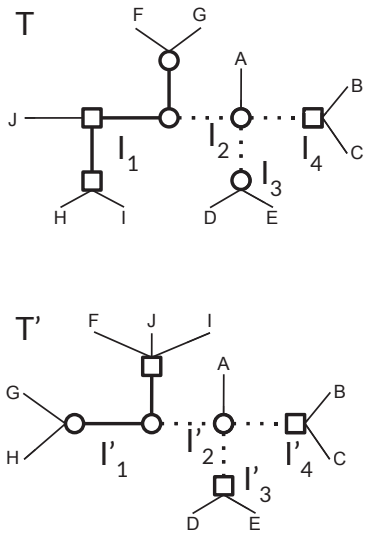


Fig. 5.2. Two trees T and T' on $\mathcal{T}_{\mathcal{L}}$ for $\mathcal{L} = \{A,B,C,D,E,F,I,J\}$, with a binary labeling of internal nodes (squares and circles). Dotted lines represent good internal edges, solid lines represent bad edges and thin lines represent terminal edges (which are good edges). This representation highlights the partition of the two trees into the island pairs $\mathcal{I}_{(T,T')} = \{(I_1,I'_1),(I_2,I'_2),(I_3,I'_3),(I_4,I'_4)\}$. Notice that each dotted line belongs to its two adjacent islands

resulting in island merging. In other words, treating island pairs separately may not result in an optimal scenario of edit operations under $ELRF$, preventing the above inequality from being an equality. Interestingly, the equality holds for the LRF distance, as we show in the next section.

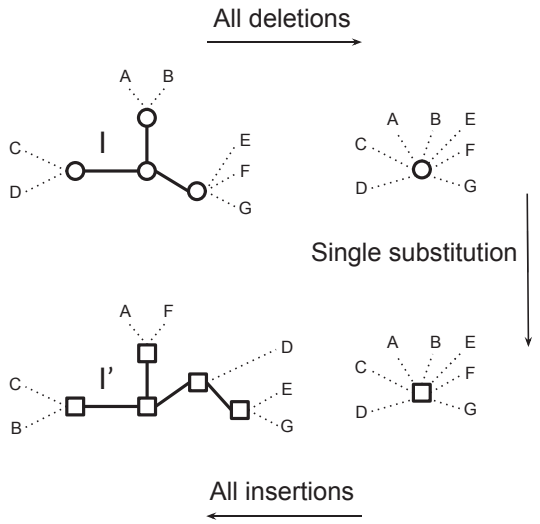


Fig. 5.3. An optimal sequence of edit operations for the island pair (I,I').

5.4.2. Computing the LRF Distance on Islands

We require an additional definition. Two trees I and I' of an island pair are said to *share a common label* $l \in \Lambda$ if there exist $x \in V(I)$ and $x' \in V(I')$ such that $\lambda(x) = \lambda(x') = l$. If I and I' do not share any common label, then (I,I') is called a *label disjoint* island pair. For example, the pair (I_3, I'_3) in Figure 5.2 or the pair (I, I') in Figure 5.3 are label disjoint.

Now let (I,I') be an island pair. Transforming I into I' can be done by reducing I into a star tree by performing a sequence of node deletions (if any, i.e. if I is not already a star tree), and then raising the star tree by inserting the required nodes to reach I'. Only the unique node not deleted during the first step might require a label substitution; for all inserted nodes, the label can be chosen to match that of I'. However, if I and I' share a common label l among their internal nodes, then the deletions can be done in a way such that the surviving node x of I is one with label $\lambda(x) = l$, thus avoiding the need for any substitution. The number of required operations is thus $\epsilon(I)$ deletions, followed by zero or one substitution, followed by $\epsilon(I')$ insertions. Alternatively, the problem can be seen as one of reducing the two trees into star trees by performing $\epsilon(I) + \epsilon(I')$ deletions, in a way reducing the two islands into two star trees sharing the same label, if possible. Figure 5.3 depicts an example of such tree editing for a label disjoint island pair.

The following lemma shows that the sequential way of doing described above is optimal.

Lemma 5.4.7. *Let (I,I') be an element of $\mathcal{I}_{(T,T')}$. Then:*

- *If I and I' share a common label, then $LRF(I,I') = \epsilon(I) + \epsilon(I')$.*
- *Otherwise $LRF(I,I') = \epsilon(I) + \epsilon(I') + 1$.*

Proof. The scenario depicted above for transforming I into I' clearly requires $\epsilon(I) + \epsilon(I')$ node insertions and deletions, and an additional node label substitution in case I ans I' are label-disjoint. We can conclude that $LRF(I,I') \leq \epsilon(I) + \epsilon(I')$ if I and I' share a common label and $LRF(I,I') \leq \epsilon(I) + \epsilon(I') + 1$, if I and I' are label-disjoint.

On the other hand, since an edit operation can remove or insert at most one edge, and the only operations removing an edge are node removal or node insertion, we clearly require at least $\epsilon(I) + \epsilon(I')$ node removals and insertions to transform the unlabeled form of the tree I into the unlabeled form of I'. Furthermore, as deletions do not affect star nodes, at least one node in I should survive (i.e. not be affected by a node deletion). Thus, if the two trees are label-disjoint, then at least one node label substitution is required. We can then conclude that $LRF(I,I') \geq \epsilon(I) + \epsilon(I')$ if I and I' share a common label and $LRF(I,I') \geq \epsilon(I) + \epsilon(I') + 1$, if I and I' are label-disjoint, which concludes the proof. □

The following lemma shows that good edge deletions can be avoided in a minimal edit path. Consequently island merging can also be avoided, which will then allow us considering each pair of islands separately.

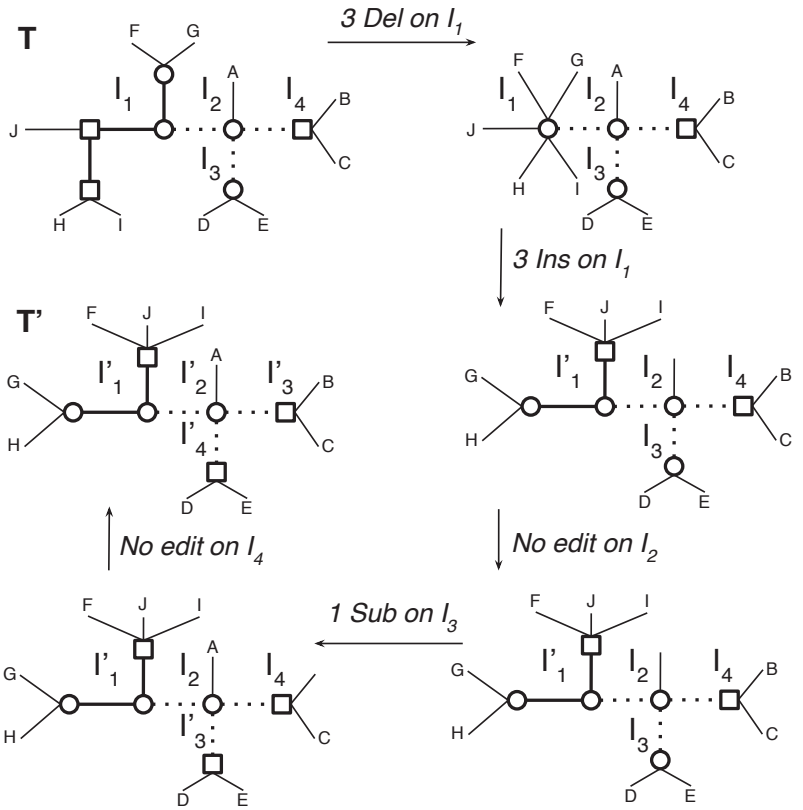


Fig. 5.4. A path $\mathcal{P}$ transforming T into T' of the form $\mathcal{P}_1.\mathcal{P}_2.\mathcal{P}_3.\mathcal{P}_4$, each $\mathcal{P}_i$ being a shortest path for the island pair (I_i, I'_i). Here $|\mathcal{P}_1| = 6$, $|\mathcal{P}_2| = 0$, $|\mathcal{P}_3| = 1$, and $|\mathcal{P}_4| = 0$.

Lemma 5.4.8. *Let T and T' be two trees of $\mathcal{T}_{\mathcal{L}}$. There exists a shortest path of edit operations transforming T into T' involving no deletion of a good edge of T.*

Proof. Let $\mathcal{P} = (o_1, o_2, \cdots o_p)$ be a path transforming T into T'. Let o_i be the leftmost operation of the form $o_i = Del(T,x,y)$ were $e = \{x,y\}$ is a good edge of T. We denote by $\{B_x, B_y\}$ with $B_1 = L(T_x)$ and $B_2 = L(T_y)$ the bipartition of $\mathcal{L}$ corresponding to e. As $\{B_1, B_2\}$ is also a bipartition in T', there should exist a smallest $j > i$ such that the operation o_j is a node insertion operation recreating this bipartition. Let T_{i-1} be the tree obtained after performing the sequence of operations $(o_1, \cdots, o_{i-1})$ on T, and T_j be the tree obtained from T_{i-1} after performing the sequence of operations $\mathcal{P}[i,j] = (o_i, o_{i+1}, \cdots, o_{j-1}, o_j)$. Now let $\mathcal{P}'[i,j] = (o'_{i+1}, \cdots, o'_{j-1})$ be the sequence of operations obtained from $\mathcal{P}[i,j]$ as follows: (1) Remove the two operations o_i and o_j; (2) For each k, $i+1 \leq k \leq j-1$, if o_k does not affect node y or if it is a node substitution, o'_k is simply o_k; (3) if $o_k = Del(T,z,y)$, then replace it by the operation $o'_k = Del(T,z,x)$ if $z \in B_1$, or by the operation $o'_k = Del(T,z,y)$ if $z \in B_2$; (4) if $o_k = Del(T,y,z)$, then replace it by the operation $o'_k = Del(T,x,z)$ if $z \in B_1$ and rename z as x, or replace it by the operation $o'_k = Del(T,y,z)$ if $z \in B_2$ and rename z as y. This sequence of operations then leads to the tree T'_j, which is the same as T_j except possibly the two labels of x and y, which can be corrected by at most two additional substitutions.

Therefore, we can substitute the subpath $\mathcal{P}[i,j]$ by a subpath of at most the same number of operations that do not involve deleting the good edge e.

It suffices then to proceed in the same way with the next leftmost good edge deletion of $\mathcal{P}$, and so on, until no good edge deletion remains. □

We are now ready to prove the equality leading to the efficient computation of the LRF distance of two trees (see Figure 5.4 for an example).

Theorem 5.4.9. *Let $\mathcal{I}_{(T,T')} = \{(I_1,I'_1),(I_2,I'_2),\cdots,(I_n,I'_n)\}$ be the island pairs of T and T'. Then*

$$LRF(T,T') = \sum_{i=1}^{n} LRF(I_i, I'_i)$$

Proof. Let $\mathcal{P}$ a shortest path transforming T into T' verifying the condition of Lemma 5.4.8, i.e. not involving any deletion of good edges. As islands can only share good edges, and good edges are never deleted by any operation of $\mathcal{P}$, islands are never merged during the process of transforming T into T', and thus $\mathcal{P}$ can be reordered in the form $\mathcal{P}_1.\mathcal{P}_2.\cdots.\mathcal{P}_n$ where each $\mathcal{P}_i$, $1 \leq i \leq n$, is a path of edit operations transforming I_i into I'_i. Each $\mathcal{P}_i$ should be a shortest path from I_i to I'_i as otherwise it can be replaced by a shortest path, contradicting the fact that $\mathcal{P}$ is a shortest path. □

The next result directly follows from Lemma 5.4.7 and Theorem 5.4.9.

Corollary 5.4.10. *Let $\mathcal{I}_{(T,T')} = \{(I_1,I'_1),(I_2,I'_2),\cdots,(I_n,I'_n)\}$ be the island pairs of T and T' and δ be the number of label-disjoint pairs. Then*

$$LRF(T,T') = \sum_{i=1}^{n}(\epsilon(I_i) + \epsilon(I'_i)) \; + \; \delta$$

5.5. Algorithm

We present our algorithm for computing the LRF distance at a logical level (Algorithm 1). The input is a pair of trees T_1, T_2 of $\mathcal{T}_{\mathcal{L}}$. We show that $LRF(T_1,T_2)$ can be computed in time $\mathcal{O}(n)$, where $n = |\mathcal{L}|$.

We start with the identification of good edges. Lines 1 and 2 of Algorithm 1 retrieve the non-trivial bipartitions for each input tree and Line 3 intersects the obtained bipartitions of T_1 and T_2 to generate the set of good edges shared by the two input trees. This can be done in time $\mathcal{O}(n)$ [**19**].

Next the algorithm identifies and characterises the islands of T_1 and T_2 (lines 4 and 5). This is performed by a traversal of each tree in pre-order and in doing so identifying the islands, which are separated by good edges, keeping track of the number of internal nodes, the labels of the internal nodes of the islands, and the nodes associated with each island. Each tree traversal is done in time $\mathcal{O}(n)$.

The next stem require pairing islands of T_1 and T_2 by iterating over the good edges ($\mathcal{O}(n)$). Line 8 first retrieves, for both input trees, the islands delimited by the current good edge, then it proceeds by pairing one island from T_1 to its matching island from T_2, and then by pairing the two remaining islands from each tree. Using the node-to-island map computed earlier, the retrieval of the two island pairs associated with a good edge can be done in constant time.

For each of the matching island pairs, at lines 9 and 14, the algorithm checks whether each island pair has already been visited in a previous iteration of the loop (the same island can be visited from multiple good edges). If not, the current distance is implemented by adding $\epsilon(I_1) + \epsilon(I_2)$.

Algorithm 1 $LRF(T_1,T_2)$

```
1: bipartitions_1 = getBiparitions(T_1);
2: bipartitions_2 = getBiparitions(T_2);
3: goodEdges = bipartitions_1 ∩ bipartitions_2;
4: islands_1 = getIslands(T_1, goodEdges);
5: islands_2 = getIslands(T_2, goodEdges);
6: distance = 0;
7: for i ∈ goodEdges:
8:       ((x_1,y_1),(x_2,y_2)) = islandPair(i, islands_1, islands_2);
9:       if x_1.visited == False:
10:          distance += x_1.ϵ + y_1.ϵ;
11:          if x_1.labels ∩ y_1.labels == ∅:
12:              distance += 1;
13:          x_1.visited = True
14:      if x_2.visited == False:
15:          distance += x_2.ϵ + y_2.ϵ;
16:          if x_2.labels ∩ y_2.labels == ∅:
17:              distance += 1;
18:          x_2.visited = True
19: if goodEdges == ∅ :
20:      distance += islands_1[0].ϵ + islands_2[0].ϵ
21:      if islands_1[0].labels ∩ islands_2[0].labels == ∅:
22:          distance += 1;
23: return distance;
```

The for-loop ends with lines 11-12 and 16-17 account for a potentially required single substitution between corresponding islands, in case they have no label in common (i.e. they form a label-disjoint island pair). These operations can also be performed in constant time, giving an overall $\mathcal{O}(n)$ runtime for the for-loop.

Finally, lines 19-22 are needed to handle the special case where there is no good edge between T_1 and T_2, for instance if T_1 or T_2 is a star. In such a case, there is only one island per tree, which is matching.

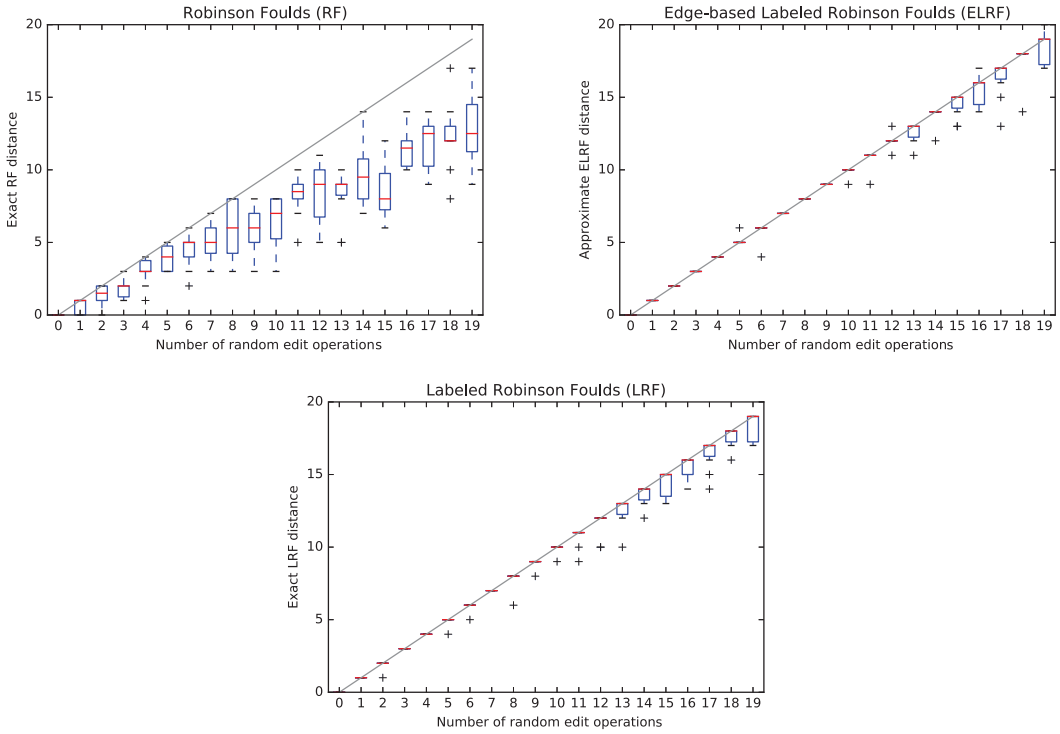


Fig. 5.5. Empirical comparisons of the distance inferred for an increasing number of random edit operations (node insertion, deletion, substitution) on the NOX4 gene tree (182 leaves), using the classical RF distance (top), the $ELRF$ approximation ([**8**]; middle), and the LRF exact distance (bottom).

We provide an open source implementation of LRF in Python as part of the pyLabeledRF package (https://github.com/DessimozLab/pylabeledrf).

5.6. Experimental Results

To illustrate the usefulness of LRF, we performed two experiments. First, we compared LRF with RF and $ELRF$ on a labeled gene tree with random edits. Second, we used LRF to tackle an open question in orthology inference: does labeled gene tree inference benefits from denser taxon sampling?

5.6.1. Empirical Comparison of LRF with RF and $ELRF$

We retrieved the labeled tree associated with human gene NOX4 from Ensembl release 99 [**77**], containing 182 genes, including speciation and duplication nodes. Next, we introduced a varying number of random edits, with 10 replicates, as follows: with probability 0.3, the label of one random internal node was substituted (from a speciation label into a duplication one or vice versa); the rest of the probability mass function was evenly distributed among all internal edges (each implying a potential node deletion) and all nodes of degree > 3 (each providing the opportunity of a potential node insertion). For $ELRF$, consistent with

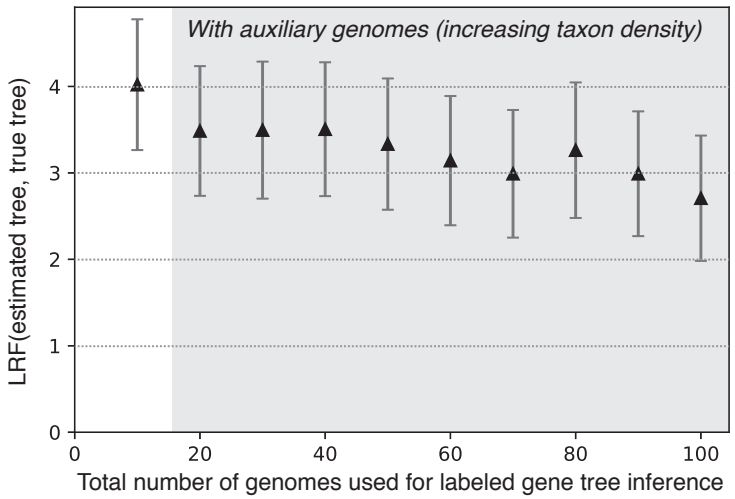


Fig. 5.6. Denser taxon sampling decreases labeled tree estimation error: labeled gene trees reconstructed with an increasing number of auxiliary genomes (i.e. obtained by including the additional genomes during tree inference and labeling, followed by pruning) have a smaller LRF distance to the true trees. Error bars depict 95% confidence intervals around the mean.

its underlying model, we added the requirement that edge deletion only affect edges with adjacent nodes with the same label.

For each of RF, LRF and $ELRF$, we provide the distance as a function of the number of random edits (Fig. 5.5). As expected, the conventional RF distance returns the smallest values because it ignores labels. The two labeled RF alternatives performed similarly, but the heuristic for $ELRF$ occasionally exceeded the true number of edit operations — a shortcoming that we do not have with LRF, as we have an exact algorithm for this distance. Both labeled RF variants tracked better the actual number of changes, until around 13 edits for LRF or $ELRF$, after which the minimum edit path starts to be often shorter than the actual sequence of random edits.

5.6.2. The Effect of Denser Taxon Sampling on Labeled Gene Tree Inference

We used LRF to assess the effect of species sampling for the purpose of labeled gene tree reconstruction. Consider the problem of reconstructing a labeled tree corresponding to homologous genes from 10 species. Our question is: is it better to infer and label the tree using these 10 species alone, or is it better to use more species to infer and label the tree, and then prune the resulting tree to only contain the leaves corresponding to the original 10 species? While denser taxon sampling is known to improve unlabeled phylogenetic inference [**51**], we are not aware of any previous study on labeled gene tree inference.

First, using ALF [**17**], we simulated the evolution of the genomes of 100 extant species from a common ancestor genome containing 100 genes (*Parameters*: root genome with 100

genes of 432 nucleic acids each; species tree sampled from a birth-death model with default parameters; sequences evolved using the WAG model, with Zipfian gap distribution; duplication and loss events rate of 0.001). In the simulation, genes can mutate, be duplicated or lost. All the genes in the extant species can thus be traced back to one of these 100 ancestral genes and be assigned to the corresponding gene family. The 100 true gene trees, including speciation and duplication labels, are known from the simulation. However, in our run, one tree ended up containing only two genes (due to losses on early branches) and was thus excluded from the rest of the analysis.

To evaluate the inference process, among the 100 species, we randomly selected nested groups of 10, 20, 30, 40, 50, 60, 70, 80 and 90 species. We considered the 10 species in the first group as the species of interest. All other species were used to potentially improve the reconstruction of the gene trees for the first 10 genomes. Then, for each group, we aligned protein sequences translated from homologous genes using MAFFT L-INS-i [**41**], inferred phylogenetic trees from the alignments using FastTree [**57**], and annotated their nodes using the species overlap algorithm [**72**] as implemented in the ETE3 python library [**38**]. Finally, we pruned both the inferred gene trees and the true trees to include only proteins corresponding to the 10 species of interest.

We used LRF to assess the distance between the estimated and true labeled trees, for the various number of auxiliary genomes considered. For each scenario, we computed the mean LRF distance over all gene trees (Fig. 5.6). The mean error (expressed in LRF distance) decreases as the number of auxiliary species increases. This simple simulation study suggests that denser species sampling improves labeled gene tree inference.

5.7. Discussion and Conclusion

The LRF distance introduced here overcomes the major drawback of $ELRF$ namely the lack of an exact polynomial algorithm for the latter. Indeed, with $ELRF$, minimal edit paths can require contracting "good" edges, i.e., edges present in the two trees [**8**].

By contrast, with LRF, we demonstrated that there is always a minimal path which does not contracts good edges. Better yet, we proved that LRF can be computed exactly in linear time. The new formulation also maintains other desirable properties: being a metric and reducing to the conventional Robinson Foulds distance in the presence of trees with only one type of label. Finally, we showed that the new distance is computable for an arbitrary number of label types.

Our experimental results illustrate the utility of computing tree distances taking labels into account, as the conventional RF distance is blind to label changes. At first sight, it may seem surprising that in a tree of 182 leaves, the minimum edit path under LRF or $ELRF$ already starts underestimating the actual number of random edit operations after around

13 operations. However, this can be explained by the "birthday paradox" [**1**]: to be able to reconstruct the actual edit path, no two random edits should affect the same node. Yet the odds of having, among 13 random edits, at least two edits affecting the same internal node (among 179) is in fact substantial — approximately 36% in our case — just like the odds of having two people with the same birthday in a given group is higher than what most people intuit.

Like RF and $ELRF$, the main limitation of LRF is the lack of biological realism. For one thing, there is no justification to assign equal weight to the three kinds of edits in all circumstances. For instance, it is typically highly implausible to introduce a speciation node at the root of a subtree containing multiple copies of a gene in the same species.

However, LRF complement analyses performed using more realistic models are either unavailable or too onerous to compute. In particular, the ability of LRF to support an arbitrary number of labels makes it applicable to gene trees containing more than just speciations and duplications, such as horizontal gene transfers or gene conversion events.

Finally, LRF constitutes a clear improvement over RF in the context of gene tree benchmarking, where trees inferred by various reconciliation models are compared using a distance measure [**4, 50**]. Such an application was illustrated in the simulation study of the previous section, in which we observed that denser taxon sampling improved labeled tree inference computed using the widely used species overlap method. More work will be needed to assess the generality of this result.

Chapter 6

Conclusion

In the context of the study of biological entities, phylogenetics focuses on the investigation of the evolutionary history and relationships among those entities. Inferring phylogenies does indeed require a proper understanding of genetic variation, heredity in organisms, and evolutionary processes. The importance of grasping these concepts for us stems from how they can be used to infer gene families. A gene family represents the relationships between genes that descend from one common ancestor. Additionally, gene trees are an important tools for the inference of species trees. Overall, phylogenetic trees are important instruments that help determine the structure and variation of biological processes within organisms. Nonetheless, the biological information that can be obtained from phylogenetic tree reconstruction should not be completely relied on as species trees or gene trees that fully and perfectly represents the historical relationships between biological entities are unlikely because tree inference methods are not without flaws. This is why being able to analyse them and compare them can lead to a better understanding of biological mechanisms and thus benefit a number of important applications that depend on it, such as the design of appropriate gene therapies [**22**].

Comparing trees is therefore an essential task for many purposes, and especially in phylogeny where different reconstruction tools may lead to different trees, likely representing contradictory evolutionary information. The research community has made many attempts to increase the scope and applicability of comparisons between phylogenetic trees. To do so, a number of articles have proposed several metrics across two general categories that we referred to as cluster-similarity metrics and edge-based metrics. There exists a variety of types of phylogenetic trees with a wide range of features. This diversity makes it impossible to have a perfect metric that suits all needs and accounts for all potential features. This is why the continuous development of diverse metrics is important to the progress of the field of phylogenetics. While a large variety of pairwise measures of dissimilarity have been developed for comparing trees with no information on internal nodes, very few measures have

been designed for node-labeled trees, which is for instance the case of reconciled gene trees that may be labeled with evolutionary events such as speciation, duplication, or horizontal gene transfer. The inability to perform such comparisons with an adequate metric was the motivation for this thesis.

The core contributions of this thesis is the proposal two natural extensions of the RF distance to node labeled trees. We first defined and evaluated $ELRF$, which appears to have worse computational efficiency than the RF distance, but which can be applied to labeled trees with a limit of two node label types. We then proposed and evaluated LRF, which features the same efficiency as the classical RF distance, and which can be applied with an arbitrary number of node label types. These characteristics are making it very useful for comparing gene trees under various evolutionary models that may involve speciation, duplication, loss, HGT, and other potential evolutionary events. To illustrate the usefulness of the presented extensions of RF, we performed experiments where we compared them with RF on labeled gene trees. We observed that the conventional RF distance did not perform as well because it could not account for internal node labels. The two labeled RF alternatives performed similarly, but the heuristic for $ELRF$ occasionally lacked precision at estimating the true number of edit operations, unlike LRF, as we developed an exact algorithm for this distance. Overall both labeled RF variants were better at estimating the actual number of changes than their predecessor. This thesis thus contributes useful solutions and tools for the preliminary analysis and comparison of labeled gene trees under various evolutionary models that may involve various evolutionary events.

Nonetheless, it is important to remember that our extensions still inherited some of RF disadvantages, such as a lack of biological realism as there no rationale to assign equal weight to all types of edit operations in all cases, as well as a limited ability to meaningfully distinguish pairs of arbitrary trees as a result of low robustness (high sensitivity) to errors in trees. However, the strengths of our extensions are traits that more realistic models lack, and this is why we claim that our extensions complement them by enabling preliminary analysis and comparisons of phylogenetic trees. Finally, as other extensions of the original RF distance have successfully addressed some of its weaknesses for non-labeled trees, we are hopeful that future research could lead to further improved metrics for labeled gene trees.

References

[1] Morton Abramson et WOJ Moser : More birthday surprises. *The American Mathematical Monthly*, 77(8):856–858, 1970.

[2] John Aldrich *et al.* : Ra fisher and the making of maximum likelihood 1912-1922. *Statistical science*, 12(3):162–176, 1997.

[3] Benjamin L Allen et Mike Steel : Subtree transfer operations and their induced metrics on evolutionary trees. *Annals of combinatorics*, 5(1):1–15, 2001.

[4] Adrian M Altenhoff, Brigitte Boeckmann, Salvador Capella-Gutierrez, Daniel A Dalquen, Todd DeLuca, Kristoffer Forslund, Jaime Huerta-Cepas, Benjamin Linard, Cécile Pereira, Leszek P Pryszcz *et al.* : Standardized benchmarking in the quest for orthologs. *Nature methods*, 13(5):425–430, 2016.

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

[6] Damian Bogdanowicz et Krzysztof Giaro : On a matching distance between rooted phylogenetic trees. *International Journal of Applied Mathematics and Computer Science*, 23(3):669–684, 2013.

[7] Bastien Boussau et Celine Scornavacca : Reconciling gene trees with species trees. *Phylogenetics in the Genomic Era*, page 3.2:1–3.2:23, 2020.

[8] Samuel Briand, Christophe Dessimoz, Nadia El-Mabrouk, Manuel Lafond et Gabriala Lobinska : A generalized robinson-foulds distance for labeled trees. *Asia Pacific Bioinformatics Conference (APBC)*, 2020.

[9] Luciano Brocchieri : Phylogenetic inferences from molecular sequences: review and critique. *Theoretical population biology*, 59(1):27–40, 2001.

[10] David Bryant et Celine Scornavacca : An $o(n \log n)$ time algorithm for computing the path-length distance between trees. *Algorithmica*, 81(9):3692–3706, 2019.

[11] David Bryant et Mike Steel : Computing the distribution of a tree metric. *IEEE/ACM transactions on computational biology and bioinformatics*, 6(3):420–426, 2009.

[12] Gabriel Cardona, Mercè Llabrés, Francesc Rosselló et Gabriel Valiente : Nodal distances for rooted phylogenetic trees. *Journal of mathematical biology*, 61(2):253–276, 2010.

[13] Ruchi Chaudhary, J Gordon Burleigh et David Fernandez-Baca : Fast local search for unrooted robinson-foulds supertrees. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 9(4):1004–1013, 2012.

[14] Benny Chor et Tamir Tuller : Finding a maximum likelihood tree is hard. *Journal of the ACM (JACM)*, 53(5):722–744, 2006.

[15] C. Colijn et G. Plazzotta : A metric on phylogenetic tree shapes. *Syst. Biol.*, 67(1):113–126, 2018.

[16] Douglas E Critchlow, Dennis K Pearl et Chunlin Qian : The triples distance for rooted bifurcating phylogenetic trees. *Systematic Biology*, 45(3):323–334, 1996.

[17] Daniel A Dalquen, Maria Anisimova, Gaston H Gonnet et Christophe Dessimoz : Alf—a simulation framework for genome evolution. *Molecular biology and evolution*, 29(4):1115–1123, 2012.

[18] Charles Darwin : *On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life*. John Murray, 1859.

[19] William HE Day : Optimal algorithms for comparing trees with labeled leaves. *Journal of classification*, 2(1):7–28, 1985.

[20] Alexandre De Bruyn, Darren P Martin et Pierre Lefeuvre : Phylogenetic reconstruction methods: an overview. *In Molecular Plant Taxonomy*, pages 257–277. Springer, 2014.

[21] A Dress : Towards a theory of holistic clustering. *DIMACS Ser. Discrete Math. Theoret. Comput. Sci*, 37:271–289, 1997.

[22] Nadia El-Mabrouk et Emmanuel Noutahi : Gene family evolution—an algorithmic framework. *In Bioinformatics and Phylogenetics*, pages 87–119. Springer, 2019.

[23] George F Estabrook, FR McMorris et Christopher A Meacham : Comparison of undirected phylogenetic trees based on subtrees of four evolutionary units. *Systematic Zoology*, 34(2):193–200, 1985.

[24] Joseph Felsenstein : Maximum-likelihood estimation of evolutionary trees from continuous characters. *American journal of human genetics*, 25(5):471, 1973.

[25] Joseph Felsenstein : Cases in which parsimony or compatibility methods will be positively misleading. *Systematic zoology*, 27(4):401–410, 1978.

[26] Joseph Felsenstein : Evolutionary trees from dna sequences: a maximum likelihood approach. *Journal of molecular evolution*, 17(6):368–376, 1981.

[27] Joseph Felsenstein et Joseph Felenstein : *Inferring phylogenies*, volume 2. Sinauer associates Sunderland, MA, 2004.

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

[29] Paweł Górecki, Alexey Markin, Agnieszka Mykowiecka, Jarosław Paszek et Oliver Eulenstein : Phylogenetic tree reconciliation: Mean values for fixed gene trees. *In International Symposium on Bioinformatics Research and Applications*, pages 234–245. Springer, 2017.

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

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

[32] W Keith Hastings : Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, April 1970.

[33] Michael Hendriksen et Andrew Francis : A partial order and cluster-similarity metric on rooted phylogenetic trees. *arXiv preprint arXiv:1906.02411*, 2019.

[34] Michael D Hendy et David Penny : Branch and bound algorithms to determine minimal evolutionary trees. *Mathematical Biosciences*, 59(2):277–290, 1982.

[35] Maribel Hernandez-Rosales, Marc Hellmuth, Nicolas Wieseke, Katharina T Huber, Vincent Moulton et Peter F Stadler : From event-labeled gene trees to species trees. *In BMC bioinformatics*, volume 13, page S6. BioMed Central, 2012.

[36] Glenn Hickey, Frank Dehne, Andrew Rau-Chaplin et Christian Blouin : Spr distance computation for unrooted trees. *Evolutionary Bioinformatics*, 4:EBO–S419, 2008.

[37] John P Huelsenbeck : Performance of phylogenetic methods in simulation. *Systematic biology*, 44(1): 17–48, 1995.

[38] Jaime Huerta-Cepas, François Serra et Peer Bork : Ete 3: reconstruction, analysis, and visualization of phylogenomic data. *Molecular biology and evolution*, 33(6):1635–1638, 2016.

[39] Bhaskar DasGupta Xin He Tao Jiang, Ming Li, John Tromp et Louxin Zhang : On computing the nearest neighbor interchange distance. *In Discrete Mathematical Problems with Medical Applications: DIMACS Workshop Discrete Mathematical Problems with Medical Applications, December 8-10, 1999, DIMACS Center*, volume 55, page 125. American Mathematical Soc., 2000.

[40] Thomas H Jukes, Charles R Cantor *et al.* : Evolution of protein molecules. *Mammalian protein metabolism*, 3(21):132, 1969.

[41] Kazutaka Katoh et Daron M Standley : Mafft multiple sequence alignment software version 7: improvements in performance and usability. *Molecular biology and evolution*, 30(4):772–780, 2013.

[42] Michelle Kendall et Caroline Colijn : Mapping phylogenetic trees to reveal distinct patterns of evolution. *Molecular biology and evolution*, 33(10):2735–2743, 2016.

[43] Bryan Kolaczkowski et Joseph W Thornton : Performance of maximum parsimony and likelihood phylogenetics when evolution is heterogeneous. *Nature*, 431(7011):980–984, 2004.

[44] M. Lafond, N. El-Mabrouk, K.T. Huber et V. Moulton : The complexity of comparing multiply-labelled trees by extending phylogenetic-tree metric. *Theoretical Computer Science*, 760:15–34, 2019.

[45] Manuel Lafond et Nadia El-Mabrouk : Orthology and paralogy constraints: satisfiability and consistency. *BMC genomics*, 15(6):S12, 2014.

[46] Yu Lin, Vaibhav Rajan et Bernard ME Moret : A metric for phylogenetic trees based on matching. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 9(4):1014–1022, 2012.

[47] Judith C Masters et Luca Pozzi : Phylogenetic inference. *The International Encyclopedia of Primatology*, pages 1–6, 2016.

[48] S. Mittal et G. Munjal : Tree mining and tree validation metrics: A review. *IOSR: Journal of Computer Engineering*, pages 31-36, 2015.

[49] Jucheol Moon et Oliver Eulenstein : Cluster matching distance for rooted phylogenetic trees. *In International Symposium on Bioinformatics Research and Applications*, pages 321–332. Springer, 2018.

[50] Benoit Morel, Alexey M Kozlov, Alexandros Stamatakis et Gergely J Szöllősi : Generax: A tool for species tree-aware maximum likelihood based gene tree inference under gene duplication, transfer, and loss. *BioRxiv*, page 779066, 2019.

[51] Ahmed Ragab Nabhan et Indra Neil Sarkar : The impact of taxon sampling on phylogenetic inference: a review of two decades of controversy. *Briefings in bioinformatics*, 13(1):122–134, 2012.

[52] Masatoshi Nei et Sudhir Kumar : *Molecular evolution and phylogenetics*. Oxford university press, 2000.

[53] T Heath Ogden et Michael S Rosenberg : Multiple sequence alignment accuracy and phylogenetic inference. *Systematic biology*, 55(2):314–328, 2006.

[54] Nicholas D Pattengale, Eric J Gottlieb et Bernard ME Moret : Efficiently computing the robinson-foulds metric. *Journal of Computational Biology*, 14(6):724–735, 2007.

[55] Hervé Philippe, Henner Brinkmann, Dennis V Lavrov, D Timothy J Littlewood, Michael Manuel, Gert Wörheide et Denis Baurain : Resolving difficult phylogenetic questions: why more sequences are not enough. *PLoS biology*, 9(3), 2011.

[56] Hervé Philippe, Frédéric Delsuc, Henner Brinkmann et Nicolas Lartillot : Phylogenomics. *Annu. Rev. Ecol. Evol. Syst.*, 36:541–562, 2005.

[57] Morgan N Price, Paramvir S Dehal et Adam P Arkin : Fasttree 2–approximately maximum-likelihood trees for large alignments. *PloS one*, 5(3):e9490, 2010.

[58] David F Robinson et Leslie R Foulds : Comparison of phylogenetic trees. *Mathematical biosciences*, 53(1-2):131–147, 1981.

[59] Béatrice Roure, Denis Baurain et Hervé Philippe : Impact of missing data on phylogenies inferred from empirical phylogenomic data sets. *Molecular biology and evolution*, 30(1):197–214, 2013.

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

[61] David Sankoff et Pascale Rousseau : Locating the vertices of a steiner tree in an arbitrary metric space. *Mathematical Programming*, 9(1):240–246, 1975.

[62] Heiko A Schmidt et Arndt von Haeseler : Phylogenetic inference using maximum likelihood methods. *The Phylogenetic Handbook: a Practical Approach to Phylogenetic Analysis and Hypothesis Testing 2nd Edition Cambridge University Press, Cambridge*, pages 181–209, 2009.

[63] F. Schreiber, M. Patricio, M. Muffato, M. Pignatelli et A. Bateman : Treefam v9: a new website, more species and orthology-on-the-fly. *Nucleic Acids Research*, 2013. doi: 10.1093/nar/gkt1055.

[64] Stefan Schwarz, Mateusz Pawlik et Nikolaus Augsten : A new perspective on the tree edit distance. *In International Conference on Similarity Search and Applications*, pages 156–170. Springer, 2017.

[65] Charles Semple, Mike Steel *et al.* : *Phylogenetics*, volume 24. Oxford University Press on Demand, 2003.

[66] Peter HA Sneath, Robert R Sokal *et al.* : *Numerical taxonomy. The principles and practice of numerical classification.* W.H. Freeman and Company, San Francisco, CA, 1973.

[67] Robert R Sokal : A statistical method for evaluating systematic relationships. *Univ. Kansas, Sci. Bull.*, 38:1409–1438, 1958.

[68] MA Steel, MD Hendy et D Penny : Loss of information in genetic distances. *Nature*, 336(6195):118–118, 1988.

[69] Mike A Steel et David Penny : Distributions of tree comparison metrics—some new results. *Systematic biology*, 42(2):126–141, 1993.

[70] Koichiro Tamura, Daniel Peterson, Nicholas Peterson, Glen Stecher, Masatoshi Nei et Sudhir Kumar : Mega5: molecular evolutionary genetics analysis using maximum likelihood, evolutionary distance, and maximum parsimony methods. *Molecular biology and evolution*, 28(10):2731–2739, 2011.

[71] Yves Van de Peer : Phylogeny inference based. *The phylogenetic handbook: a practical approach to DNA and protein phylogeny*, page 101, 2003.

[72] Rene TJM Van der Heijden, Berend Snel, Vera Van Noort et Martijn A Huynen : Orthology prediction at scalable resolution by phylogenetic tree analysis. *BMC bioinformatics*, 8(1):83, 2007.

[73] A.J. Vilella, J. Severin, A. Ureta-Vidal, L. Heng, R. Durbin et E. Birney : EnsemblCompara gene trees: Complete, duplication-aware phylogenetic trees in vertebrates. *Genome Research*, 19:327-335, 2009.

[74] Tandy Warnow : *Bioinformatics and Phylogenetics: Seminal Contributions of Bernard Moret*, volume 29:87-119. Springer, 2019.

[75] John J Wiens : Incomplete taxa, incomplete characters, and phylogenetic accuracy: is there a missing data problem? *Journal of Vertebrate Paleontology*, 23(2):297–310, 2003.

[76] John J Wiens *et al.* : Missing data and the accuracy of bayesian phylogenetics. *Journal of Systematics and Evolution*, 46(3):307–314, 2008.

[77] Andrew D Yates, Premanand Achuthan, Wasiu Akanni, James Allen, Jamie Allen, Jorge Alvarez-Jarreta, M Ridwan Amode, Irina M Armean, Andrey G Azov, Ruth Bennett *et al.* : Ensembl 2020. *Nucleic acids research*, 48(D1):D682–D688, 2020.

[78] Kaizhong ZHANG : A new editing based distance between unordered labeled trees. *In Annual Symposium on Combinatorial Pattern Matching*, pages 254–265. Springer, 1993.

[79] Kaizhong ZHANG : A constrained edit distance between unordered labeled trees. *Algorithmica*, 15(3): 205–222, 1996.

[80] Kaizhong ZHANG, Rick STATMAN et Dennis SHASHA : On the editing distance between unordered labeled trees. *Information processing letters*, 42(3):133–139, 1992.

Appendix

Proof of Lemma 4.3.2 (Link between Rooted and Unrooted Trees):

There is a one-to-one relationship between the set of non-trivial bipartitions and the set of internal edges of an unrooted tree T. Similarly, for a rooted tree T', there is a one-to-one relationship between the set of internal edges of T' and its set of non-trivial clades, excluding the clade $L(T')$. However, the number of edges may differ between a tree T and a rooting T' of T.

- If T_1 and T_2 are both rooted into existing nodes, then T_1 and T'_1 (respec. T_2 and T'_2) have exactly the same edge sets, and we conclude from what precedes that there is a one-to-one relationship between the set of non-trivial bipartitions of T_1 (respec. T_2) and the set of non-trivial clades excluding $L(T_1)$ (respec. $L(T_2)$) of T'_1 (respec. T'_2). As $L(T_1) = L(T_2)$, this clade does not contribute to the symmetric difference computation of $\delta_R(T'_1,T'_2)$, and thus $\delta_R(T'_1,T'_2) = \delta(T_1,T_2)$.
- If both T_1 and T_2 are rooted into good edges, then T'_1 (respec. T'_2) has one edge more than T_1 (respec. T_2). But these new edges are good edges and therefore do not contribute to the symmetric difference computation of the δ_R distance, and thus $\delta_R(T'_1,T'_2) = \delta(T_1,T_2)$.
- If both T_1 and T_2 are rooted into bad edges, then T'_1 (respec. T'_2) has one edge more than T_1 (respec. T_2). These two new edges are bad edges, and thus contribute to the symmetric difference computation of the δ_R distance by adding two clades, and thus $\delta_R(T'_1,T'_2) = \delta(T_1,T_2) + 2$.
- If exactly one among T_1 and T_2 is rooted into a bad edge, than only one new edge contributes to the symmetric difference computation of the δ_R distance by adding one clade, thus $\delta_R(T'_1,T'_2) = \delta(T_1,T_2) + 1$.

Proof of Lemma 4.3.3 (Edit Distance):

The non-negative and identity conditions are obvious. For the symmetric condition, notice that we can reverse every edit operation in an optimal sequence from T_1 to T_2 to

obtain a sequence from T_2 to T_1 with the same number of events, and vice-versa (extensions and contractions are inverses of each other, and any flip can be reversed by a flip). We thus have $\delta(T_2, T_1) \leq \delta(T_1, T_2)$ and $\delta(T_1, T_2) \leq \delta(T_2, T_1)$, and equality follows.

Finally, we prove the triangular inequality condition: for 3 trees T_1, T_2 and T_3, to transform T_1 into T_2, we may take any edit sequence from T_1 to T_3, followed by any edit sequence from T_3 to T_2. It follows that $\delta(T_1, T_2) \leq \delta(T_1, T_3) + \delta(T_3, T_2)$.

Proof of Lemma 4.4.1 (Pairs of Maximal Bad Subtrees):

As $\cup_i Y_i = \mathcal{L}$, $\{e'_i\}_{1 \leq i \leq k}$ are the only terminal edges of any subtree S' of T' containing the set $\{e'_i\}_{1 \leq i \leq k}$ as terminal edges. As T' is a tree, for any $1 \leq i \neq j \leq k$, there is only one possible path from x'_i to x'_j. Uniqueness follows.

Suppose that such a subtree S' is not a bad subtree. Then it contains an internal good edge $e' = (x', y')$. In other words, there is a non-trivial bipartition of $\{Y_i\}_{1 \leq i \leq k}$ which is also a bipartition in S. This contradicts the fact that S is a bad subtree of T. Finally, as all terminal edges of S' are good edges of T', it follows that S' is a maximal bad subtree of T'.

Proof of Lemma 4.4.2 (Contract Non-Mixed Bad Edges):

We first introduce a definition that will be of use later in the proof. For two rooted trees S_1 and S_2, define the *union* of S_1 and S_2 as the tree obtained by identifying their roots, i.e. by removing the root of S_2 and making all its children now children of the root of S_1.

Let $e = \{u, v\}$ be a non-mixed bad edge and assume, without loss of generality, that both u and v have the label Spe (recall that $\Lambda = \{Spe, Dup\}$). Notice that any sequence of operations turning T into T', at some point, must contract the $\{u, v\}$ edge, as otherwise, the (bad) bipartition corresponding to $\{u, v\}$ would remain in the transformed tree and we would not obtain T' (noting that extensions cannot remove bipartitions). We now prove the Lemma by induction over $\delta(T, T')$. As a base case, suppose that $\delta(T, T') = 1$. Then $\{u, v\}$ must be the only bad edge of T and the single operation is to contract it, proving the base case.

Now assume that for any tree $\tilde{T}$ satisfying $\delta(\tilde{T}, T') < \delta(T, T')$, contracting any non-mixed bad edge of $\tilde{T}$ reduces its distance to T' by 1. Let $Q = (q_1, \ldots, q_l)$ be an optimal sequence of operations transforming T into T' (here each q_i denotes either a contraction, extension or flip). Let q_j be the event that contracts $\{u, v\}$. If $q_1 = q_j$, then we are done, so assume otherwise. We make the assumption that whenever there is a contraction involving u prior to q_j, the contracted node is still called u. Furthermore, we assume that if an extension prior to q_j splits the neighbors of u, the node v is still a neighbor of u after the operation. All the same assumptions hold for v. This just changes the names we give to nodes and does not

alter the scenario, but observe that this means that $\{u,v\}$ is in every tree obtained before the first j operations.

For each $i \in \{1,\ldots,l\}$, let T_i be the tree obtained after applying $q_1,\ldots,q_i$ on T, and define $T_0 = T$. Furthermore, for $i \in \{0,1,\ldots,j-1\}$, denote by T_i^u and T_i^v the two trees obtained from T_i by removing the edge $\{u,v\}$, where u is in T_i^u and v is in T_i^v. Define $T^u = T_0^u$ and $T^v = T_0^v$. We will assign u and v as the respective roots of each T_i^u and T_i^v. Notice that for each $i \in \{1,\ldots,j-1\}$, q_i only modifies either the subtree T_{i-1}^u or T_{i-1}^v. Therefore, if events q_i and q_{i+1} modify T_{i-1}^u and T_i^v, respectively, we could apply q_{i+1} before q_i and T_{i+1} would still be the same tree. This lets us assume that we may reorder events such that all events affecting T^u (prior to q_j) occur before those affecting T^v. That is, there is some h such that $q_1,\ldots,q_h$ only affects the T^u subtree, $q_{h+1},\ldots,q_{j-1}$ only affects the T^v subtree, so that $T_h^u = T_{h+1}^u = \ldots = T_{j-1}^u$ and $T^v = T_1^v = \ldots = T_h^v$.

Suppose first that u is labeled *Spe* in T_h, and thus also in T_{j-1}. Then v is also labeled *Spe* in T_{j-1} (and also in T_h since v was untouched until q_{h+1}). Let $\hat{T}$ be the tree obtained after contracting $\{u,v\}$ in T, and let z be the resulting node. Observe that if we interpret z as u, then we may apply the events $q_1,\ldots,q_h$ on $\hat{T}$, since these events only affected the T^u subtrees. To be formal, we "reproduce" q_1 through q_h on $\hat{T}$ by applying the events $Q' = (q'_1,\ldots,q'_h)$ on $\hat{T}$, defining $\hat{T}_i$ as the tree obtained after the i-th event of Q', where each q'_i in Q' is defined as follows:

- if q_i contracts $\{x,y\}$ in T_{i-1}, then q'_i contracts $\{x,y\}$ in $\hat{T}_{i-1}$ if $x,y \neq u$, otherwise if, say, $x = u$, then q'_i contracts $\{z,y\}$ (and calls the resulting node z);
- if q_i flips x in T_{i-1}, then q'_i flips x in $\hat{T}_{i-1}$ if $x \neq u$, or flips z otherwise;
- if q_i is an extension and splits the neighborhood of x, then q'_i does the same if $x \neq u$ (replacing u by z if needed). If $x = u$, then let X be the set of neighbors of v in T_{i-1}, excluding u. If $Ch(u)$ is split into A and B by q_i, where $v \in B$, then q'_i splits the neighbors $A \cup (B \setminus \{v\}) \cup X$ of z into A and $(B \setminus \{v\}) \cup X$ (and z is the neighbor of $(B \setminus \{v\}) \cup X$ and the newly created node).

One can verify the following that the following invariant holds on each $\hat{T}_i$, $i \in \{1,\ldots,h\}$: if we take T_i and contract the edge $\{u,v\}$, ignoring the labels and keeping the label of u, then we obtain $\hat{T}_i$ (the invariant is also true for T and $\hat{T}$).

The resulting tree $\hat{T}_h$ obtained from applying $q'_1,\ldots,q'_h$ on $\hat{T}$ will therefore contain z as a *Spe* node, and will be the union of T_h^u and T_0^v. From this point, in a similar fashion, we may interpret z as v and apply $q_{h+1},\ldots,q_{j-1}$ on $\hat{T}_h$, resulting a tree that is the union of $T_h^u = T_{j-1}^u$ and T_{j-1}^v. The corresponding events are the same as above, we omit the formal details. Since T_j is obtained from T_{j-1} by contracting $\{u,v\}$, this means that $\hat{T}_{j-1} = T_j$, which we have attained with j events but contracting $\{u,v\}$ first, which proves this case.

Suppose instead that u is labeled *Dup* in T_h. Then v is a *Dup* node in T_{j-1}. We may further assume that v is a *Spe* node in $T_{h+1}, \ldots, T_{j-2}$, since whenever we flip v into a *Dup*, we may assume by induction that $\{u, v\}$ gets contracted. Therefore, q_{j-1} flips v from *Spe* to *Dup*, and for the first time. We may then do the following: first apply the events $q_{h+1}, \ldots, q_{j-2}$ on $\hat{T}$, interpreting z as v. The resulting tree $\hat{T}'$ contains z as a *Spe* node, and is the union of T^v_{j-2} and T^u_0. We may now apply $q_1, \ldots, q_h$ on $\hat{T}'$ by interpreting u as z, resulting in a tree $\hat{T}''$ that contains z as a *Dup* node and is the union of $T^u_h = T^u_{j-1}$ and T^v_{j-1}. We have thus attained T_j, but this time without the q_{j-1} flip on v, contradicting the optimality of Q. This concludes the proof.

Proof of Lemma 4.5.1 (Upper Bound δ):

Methodology 1 performs e contractions and e' extensions. As for the number of flips, we have to flip at most all the nodes belonging to the smallest label group, which means at most half the nodes in each tree, and thus at most n flips in total.

Proof of Lemma 4.5.2 (Compare Meth.1 and Meth.2):

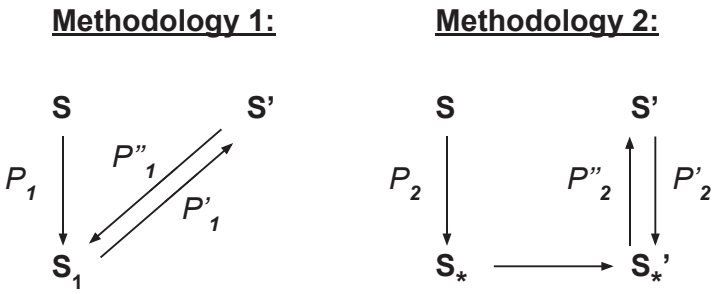


Fig. 6.1. Notations for the Proof of Lemma 4.5.2

We denote by $Cont(T)$ the minimum length of a sequence of operations contracting T, and by $l(\mathcal{P})$ the length of a sequence $\mathcal{P}$ of edit operations.

Let $\mathcal{P}_2$ be an optimal sequence contracting S to S_* and $\mathcal{P}'_2$ be an optimal sequence contracting S' to S'_*. As each operation is reversible, $\mathcal{P}'_2$ leads to a corresponding sequence $\mathcal{P}''_2$ of the same length between S'_* and S'. Thus, $\mathcal{P}_2$, concatenated with a possible flip operation transforming S_* to S'_*, concatenated with $\mathcal{P}''_2$ is a sequence from S to S' following Methodology 1, and thus $M_1(S,S') \leq M_2(S,S')$ (R1).

Conversely, let $\mathcal{P}$ be an optimal sequence following Methodology 1. Then this sequence can be subdivided into a sequence $\mathcal{P}_1$ from S to a star tree S_1, and $\mathcal{P}'_1$ from S_1 to S'. As each operation is reversible, $\mathcal{P}'_1$ leads to a corresponding sequence $\mathcal{P}''_1$ of the same length between S' and S_1. In other words, $M_1(S,S') = l(\mathcal{P}_1) + l(\mathcal{P}'_1) = l(\mathcal{P}_1) + l(\mathcal{P}''_1) \geq Cont(S) + Cont(S')$.

(1) If $S_* = S'_*$, then $M_2(S,S') = Cont(S) + Cont(S')$ and thus $M_1(S,S') \geq M_2(S,S')$, and the result follows from (R1).
(2) Otherwise, S_* and S'_* are different and $M_2(S,S') = Cont(S) + Cont(S') + 1$. Thus $M_1(S,S') \geq Cont(S) + Cont(S') = M_2(S,S') - 1$, and thus $M_2(S,S') \leq M_1(S,S') + 1$.

Proof of Lemma 4.5.3 (Optimal Path Contracting a Mixed Tree):

We first show that at least $\lceil diam(T)/2 \rceil - 1$ flips are needed, by induction over the diameter of T. When $diam(T) = 2$, T is a star tree and $0 = diam(T)/2 - 1$ flips are needed. For the induction step, we assume that any tree T' with $diam(T') < diam(T)$ requires at least $\lceil diam(T')/2 \rceil - 1$ flips. Take any optimal sequence of events S, and observe that in S, when we flip a node v of T, by Lemma 4.4.2 we may assume that S contracts all the incident edges to v until we obtain another mixed tree. Let $T_1, T_2, \ldots, T_k$ be the sequence of mixed trees encountered when applying S, i.e. each T_i is obtained after flipping a node and contracting its incident edges. Define $T_0 = T$. Let i be the smallest index such that $diam(T_i) < diam(T)$. Then in T_{i-1}, there was a longest chain $P = (u_1, \ldots, u_l)$ of length $diam(T)$. The flip-and-contract operations from T_{i-1} to T_i can reduce the length of P by at most 2 since we flip one node and only its incident edges, of which there are at most two on P. Hence $diam(T_i) \geq diam(T) - 2$. We deduce by induction that the number of required flips is at least $1 + \lceil (diam(T) - 2)/2 \rceil - 1 = \lceil diam(T)/2 \rceil - 1$.

We now turn to the converse bound $\phi(T) \leq \lceil diam(T)/2 \rceil - 1$. Fix any node v of T, and suppose that we run the following procedure: as long as T is not a star tree, flip v and contract its incident internal edges. Since each flip-and-contraction iteration reduces the length from v to any leaf by 1 (except its neighbors), $ecc_T(v)$ is reduced by 1 each round. We stop when $ecc_T(v) = 1$, in which case only terminal edges remain, and in the end, this means that $ecc_T(v) - 1$ flips are needed.

To see why this proves our bound, we show that there always exists a node with eccentricity $\lceil diam(T)/2 \rceil$. Consider a longest chain P of T with nodes $w_1, \ldots, w_k$. Observe that $diam(T) = k - 1$ (recall that distances are counted in terms of edges). Consider a midpoint node $w := w_{\lceil k/2 \rceil}$ on P. We claim that $ecc_T(w) = \lceil diam(T)/2 \rceil$. It is easy to check that w has distance at most $\lceil diam(T)/2 \rceil$ and at least $\lfloor diam(T)/2 \rfloor$ to the leaves w_1 and w_k on P. Assume for contradiction that w is at distance at least $\lceil diam(T)/2 \rceil + 1$ from some leaf l of T not in P. Then either we can form a chain from w_1 to w and then to l, or a chain from w_k to w and then to l. This chain has length at least $\lfloor diam(T)/2 \rfloor + \lceil diam(T)/2 \rceil + 1 > diam(T)$, a contradiction. This shows that $ecc_T(w) = \lceil diam(T)/2 \rceil$ and concludes the proof.

Proof of Theorem 4.5.5 (Upper Bound Meth.2):

Consider a given instance (T, T'). Take any leaf of T and assign it as the root, and do the same for T'. Although we have assumed roots of degree at least two so far, we use this rooting only for our analysis in order fix a parent-child relationship between nodes. Let Q be an optimal sequence of operations turning T into T'. We may assume that Q first contracts every non-mixed edge, and our algorithm does the same. Therefore, we suppose that T and T' contain no non-mixed edges. Assume for our purposes that whenever a contraction takes place in Q between a node u and a child v, the u node stays in the tree and v gets removed (here the notion of a child is in the rooted sense with respect to our rooting above). Also assume that when there is an extension splitting a node u, then the newly created node becomes a child of u and u retains the same parent. It is easily checked that this only alters the name of nodes and not the sequence itself.

Call an internal node v of T a *good child* if the edge between v and its parent is good. Note that v has a unique corresponding node in T' which we denote v' (i.e. v' is the root of the same clade as the subtree rooted at v). Further, call v a *bad-good* child if v is a good child, but either the label of v differs from that of v', or v is incident to at least one bad edge. Note that every maximal bad subtree of T has a (good) terminal edge with one endpoint being a bad-good child. Also note that a bad-good child v that is incident to only good edges is a particular case of a maximal bad subtree (i.e. v just has the wrong label).

We already know that $\delta(T, T')$ is at least the number of bad edges in T and T'. Let Q' be the set of operations of Q that are either flips, or contraction of good edges. We argue that $|Q'|$ is at least the number of bad-good children in T. To see this, let v be a bad-good child. Assume first that v is not incident to any bad edge. If we never flip v nor remove it by contracting its parent edge, then Q cannot transform T into T', as v and its underlying clade remain present in every tree from T to T', but with the wrong label (because a contraction not removing v cannot remove the v clade, and extensions can create clades but not remove them). So we may assume that v gets flipped or that its parent edge gets contracted. A flip must be in Q' and, observing that at any point the parent edge of v must be good, a contraction removing v must also be in Q'. Assume instead that v is incident to at least one bad edge $\{v, w\}$, with w a child of v. If v is never flipped nor removed owing to a contraction of its parent edge, then at some point w must be flipped so that the $\{v, w\}$ edge gets contracted. Otherwise, if v gets removed, then its parent edge was contracted, again implying the contraction of a good edge. Either case implies an operation in Q'. Importantly, observe that the operations in Q' identified above are all distinct, since each one implies a flip or a the removal of a node in a different bad subtree of T.

Now, let $T_1, \ldots, T_k$ be the maximal bad subtrees of T and T', and for each $i \in \{1, \ldots, k\}$, let t_i be the number of bad edges in T_i. Further denote $b = \sum_{i=1}^{k} t_i$. Since bad subtrees form

pairs, our arguments above imply that Q' has at least $k/2$ operations (because $|Q'|$ is at least the number of maximal bad trees in T, which is half the number of bad subtrees). The contraction of bad edges plus the operations of Q' show that Q has at least $\sum_{i=1}^{k} t_i + k/2 = b + k/2$ operations. Our algorithm contracts b edges in total. To count the number of flips, take any bad subtree T_i. Then $t_i \geq diam(T_i) - 2$ and the number of flips we perform is at most $\lceil diam(T_i)/2 \rceil - 1 = \lceil (diam(T_i) - 2)/2 \rceil \leq t_i/2 + 1$. Note that this also holds when T_i contains no bad edge. Therefore, the number of operations that we perform is at most $b + \sum_{i=1}^{k}(t_i/2 + 1) = 3b/2 + k$. Our approximation ratio is therefore $\frac{3b/2+k}{b+k/2} \leq \frac{2b+k}{b+k/2} = 2$.