`reconcILS`: A gene tree-species tree reconciliation algorithm that allows for incomplete lineage sorting

Sarthak R. Mishra[1]*, Megan L. Smith[2], Matthew W. Hahn[1,3]

[1]Department of Computer Science, Indiana University, Bloomington, Indiana 47405

[2]Department of Biological Sciences, Mississippi State University, Starkville MS 39762

[3]Department of Biology, Indiana University, Bloomington, Indiana 47405

*Corresponding author. E-mail: samishr@iu.edu

Abstract

Reconciliation algorithms infer the evolutionary history of individual gene trees given a species tree. Many reconciliation algorithms consider only duplication and loss events (and sometimes horizontal transfer), ignoring effects of the coalescent process, including incomplete lineage sorting (ILS). Here, we present a new heuristic algorithm for carrying out reconciliation that accurately accounts for ILS by treating it as a series of nearest neighbor interchange (NNI) events. For discordant branches of the gene tree identified by last common ancestor (LCA) mapping, our algorithm recursively chooses the optimal history by comparing the cost of duplication and loss to the cost of NNI and loss. We demonstrate the accuracy of our new method, which we call `reconcILS`, using a new simulation engine (`dupcoal`) that generates gene trees produced by the interaction of duplication, loss, and ILS. Despite being a heuristic method, `reconcILS` is much more accurate than models that ignore ILS, and at least as accurate or better than leading methods that can model ILS, while also able to handle much larger datasets. We demonstrate the use of `reconcILS` by applying it to a dataset of 23 primate genomes, highlighting its accuracy compared to standard methods in the presence of large amounts of ILS.

Introduction

The sequencing of a large number of genomes from many different species has highlighted the dynamic nature of genes: they are often duplicated, lost, or moved between species via introgression (no gain of an extra copy) or horizontal gene transfer (gain of an extra copy). One of the main goals of evolutionary and comparative genomics has been to quantify this dynamism, both in terms of the number of events of each type and their timing. A popular approach for quantifying both measures is gene tree-species tree reconciliation (reviewed in Boussau and Scornavacca 2020). Given a gene tree topology and a species tree topology, reconciliation algorithms attempt to infer the evolutionary events necessary to explain any discordance in the respective topologies. Many different reconciliation algorithms have been developed, most often considering only duplication and loss (so-called "DL" models; e.g. Goodman et al. 1979; Page 1994; Guigo et al. 1996; Page and Charleston 1997; Chen et al. 2000; Zmasek and Eddy 2001; Durand et al. 2006), but also sometimes considering duplication, transfer, and loss ("DTL" models; e.g. Hallett and Lagergren 2001; Hallett et al. 2004; Bansal et al. 2012; Stolzer et al. 2012; Szöllősi et al. 2013; Jacox et al. 2016; Morel et al. 2020).

Despite the success—and widespread use—of DL and DTL reconciliation algorithms, they do not consider gene tree discordance due to the coalescent process. The coalescent is a model that describes how sequences within a population are related (Kingman 1982; Hudson 1983; Tajima 1983), and is relevant even when considering sequences from different species because these must also coalesce in shared ancestral populations. Importantly, gene trees and species trees can be discordant solely due to coalescence, in a process often called incomplete lineage sorting (ILS) or deep coalescence (Maddison 1997). Standard DL and DTL algorithms will incorrectly infer one duplication and three losses for every discordance due only to ILS (Hahn 2007), resulting in misleading results when applying reconciliation methods to trees with ILS (e.g. Stull et al. 2021). Duplication, loss, and coalescence can also interact in complex and non-intuitive ways (Rasmussen and Kellis 2012; Mallo et al. 2014; Li et al. 2021), making it almost impossible to simply exclude gene trees discordant due to ILS from an analysis.

A few different approaches to reconciliation have been used to accommodate discordance due to coalescence. An early approach allowed short branches in the species tree—which can lead to ILS—to be collapsed (Vernot et al. 2008; Stolzer et al. 2012). The use of non-binary species trees means that such algorithms will not mistake ILS alone for duplication and loss; however, they do not account for all effects of coalescence on incorrect inferences of duplication and loss (see below). Chan et al. (2017) introduced a reconciliation algorithm that included duplication, loss, transfer, and ILS, but it assumed that new duplicates were completely linked to their parental locus (i.e. no recombination between them). Given that the majority of new gene duplicates are not completely linked (Schrider and Hahn 2010), such an assumption limits the number of genes that can be analyzed. Rasmussen and Kellis (2012) introduced a probabilistic model considering duplication, loss, and ILS and an associated reconciliation algorithm. While promising, DLCoalRecon is highly parameterized, requiring users to specify rates of duplication and loss, population sizes, and more prior to inference, and is computationally intensive. Wu et al. (2014) developed a parsimony-based algorithm (DLCpar) based on this same model that runs faster and without the need to specify duplication and loss rates ahead of time.

Here, we introduce a new heuristic method for finding the most parsimonious reconciliation of a gene tree and a species tree in the presence of duplication, loss, and ILS. Our goal is to improve upon existing methods by increasing speed and accuracy, as well as by incorporating more coalescent processes (i.e. ILS is not the only cause of discordance due to the coalescent—see section below on "Complex coalescence"). In addition, although DLCpar has been shown to perform well on smaller trees (Wu et al. 2014; Du et al. 2021), it is not straightforward to run it on larger datasets or include its algorithm within other tools that also have reconciliation steps (e.g. Thomas et al. 2017; Zhang et al. 2020). We hope that the algorithm introduced here will be easily adaptable to such scenarios. We demonstrate the use of this new algorithm by applying it to 23 whole genomes from primates, a dataset that has previously been shown to contain a large amount of discordance due to ILS (Vanderpool et al. 2020).

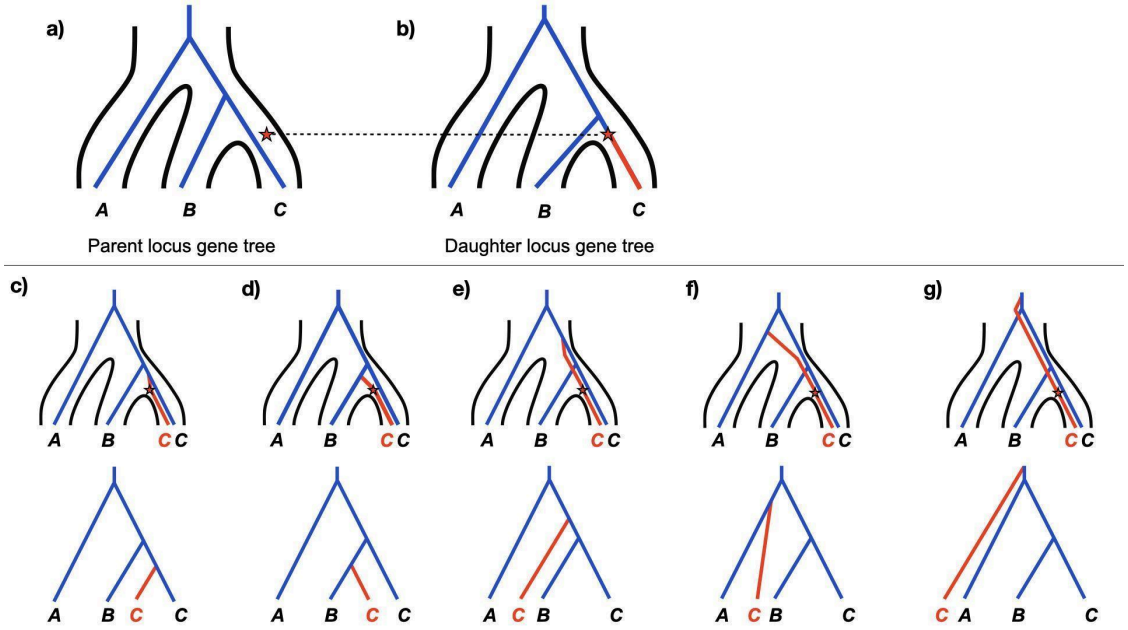

Figure 1. The multilocus multispecies coalescent model generates full gene trees. *a) A gene tree at the parent locus (blue) is drawn from the species tree (outlined in black) via the multispecies coalescent (MSC) model. A duplication event (red star) is drawn from a birth-death model, in this case occurring in the population ancestral to species B and C. b) A gene tree at the daughter locus is drawn from the MSC. The duplication is placed on this gene tree at the time at which it occurred in the species tree (shown as a dotted line), choosing one of the two possible branches at random. The inserted DNA now takes on the gene tree branch labeled in red. c)-g) Possible ways that the parent and daughter loci coalesce to form the full gene tree. In all panels the star and red branch below it come from the daughter gene tree, and the blue tree comes from the parent gene tree. The coalescent process joining the two trees is shown as a red branch above the star. The trees in the bottom row are simply unfolded views of the full gene trees.*

New Approaches

The multilocus multispecies coalescent model

We begin with a description of the multilocus multispecies coalescent (MLMSC) model. Conceptually, this model generates gene trees that are an approximation of the trees that would be produced by a Wright-Fisher model with duplication and loss (referred to as the "WFDL" model in Li et al. 2024). An algorithm for generating trees under the MLMSC was originally described in Li et al. (2021), with further updates in Li et al. (2024). Here, we describe the most important parts of the biological model, leaving algorithmic differences between simulators to the next section. We hope that this explanation makes clearer which common biological processes are modeled by the MLMSC, but not by DL and DTL methods.

The MLMSC model treats each individual locus (i.e. non-recombing position in the genome) as evolving via the multispecies coalescent (MSC) model, in which discordance can arise only via incomplete lineage sorting. The MLMSC further models relationships among these loci via duplication and coalescence, in addition to losses at any locus. In particular, the MLMSC specifies:

i. Every locus in the genome has a history (sometimes called a "locus tree") that follows the species tree.
ii. Each locus in the genome has a gene tree topology drawn from the species tree according to the MSC model. This gene tree (or "genealogy") exists at this locus whether or not a duplication or loss has occurred.

iii. Duplication events copy genetic material—i.e. DNA sequence—from one locus (the "parent" locus) and insert it at another locus (the "daughter" locus). At the parent locus, duplicated genetic material descends from a lineage that existed at the time of duplication, and so must coalesce above this point with other lineages at the parent locus according to the MSC (Figure 1a). At the daughter locus, this material inserts into a randomly chosen lineage at the time of duplication, following the gene tree of this lineage after this point (Figure 1b). Nucleotide mutations occur in the sequence that was duplicated, allowing us to reconstruct gene trees connecting one or more loci.

iv. Parent and daughter loci can be any recombination distance apart. If they are completely linked they share the same gene tree topology, while if they are unlinked each locus has an independent topology.

v. Duplications and losses are generated at rates μ and λ via a birth-death process, starting at the original locus. Each duplication event adds a sub-tree from a new daughter locus, consisting only of lineages descending from the duplication (see point iii). Losses remove descendant branches of gene trees below their occurrence (see point vii).

vi. Duplications and losses can occur on any gene tree lineage at any locus that contains genetic material descended from the original parent locus, regardless of whether they were the original parent or daughter locus.

vii. Loss events do not remove a locus, but rather do not allow lineages descended from a loss to extend to the present (Supplementary Figure S1). These lineages are therefore not present in the full gene tree.

viii. The gene tree at each daughter locus coalesces with the gene tree at its parent locus via the MSC (Figure 1c-g). Note that even if parent and daughter loci are completely linked, they still must coalesce together at the parent locus—linkage only determines whether the two marginal gene trees have the same topology or not.

ix. The "full" gene tree consists of the combined genealogies at all parent and daughter loci together, including all duplication and loss events.

Simulation using `dupcoal`

We created a new simulator—`dupcoal`—for generating gene trees under the MLMSC model. This simulator was originally needed because it was not possible to track events using the simulator of Li et al. (2021; the only one available at the time), and to evaluate the performance of `reconcILS` we require detailed output. Our algorithm works as follows (code can be found at https://github.com/meganlsmith/dupcoal):

1. Generate the original parent gene tree from the species tree according to the MSC model.
2. Simulate duplications along the branches of existing gene trees (only once for any tree) at a rate λ.
3. For each duplication, generate a daughter tree at a new locus according to the MSC model. Then, place the duplication on the daughter tree by randomly selecting a lineage available at the time of duplication on the appropriate branch of the species tree (Figure 1b). Discard all other lineages from the daughter locus, and truncate the sub-tree at the time of duplication.
4. Return to step 2 and recursively generate new duplications (and subsequent daughter trees) on each new daughter tree. Proceed to step 5 when no additional duplications have been generated on the last daughter tree.
5. Coalesce all trees together. Begin at the present and work backwards in time. First, attempt to coalesce a daughter tree with its parent, which could itself be a duplicated tree. If coalescence fails to occur before the tree originates, attempt to coalesce the daughter tree with the parent of its parent, and so on. Repeat until all trees have been joined together.
6. Generate losses on the final, full gene family tree at a rate μ.

In general, this algorithm is similar to the updated "MLMSC-II" simulator of Li et al. (2024) and is capable of generating copy-number hemiplasy, along with several other interactions between ILS, duplication, and loss not modeled by DL and DTL approaches (see next section). However, our algorithm differs in a few ways from those of Li et al. (2021;

2024). First, we generate duplications directly on the sampled parent and daughter trees (step 2 in the `dupcoal` algorithm), rather than using the "coalescent-rate" process introduced in Li et al. (2021). Second, in MLMSC-II, there is a rejection step after the generation of a daughter tree, before the mutation is placed (i.e. the "haplotype forest" rejection step in their algorithm). We do not include such a rejection step, as we always place a duplication on a branch of the daughter tree (step 3). Third, when we coalesce all marginal trees together (step 5), if a duplication fails to coalesce with its parent tree before the time of origination of the parent (which can happen when it itself is a duplicate), we coalesce it further in the past with the parent of its parent, rather than discarding the duplication as in MLMSC-II.

For each simulated gene family, `dupcoal` outputs the full simulated gene tree, including annotation of parent and daughter copies. Along with the tree, a table for each family reports: the number of duplications, the number of observable duplications (i.e. duplications that are not completely obscured by loss), the number of losses, the number of cases of different types of hemiplasy (see next section), and the number of ILS events. In contrast, MLMSC-II outputs a full, unannotated gene tree, and the only information output is the number of observed duplications, the number of genes, and the number of species. The runtimes for the two methods were similar, with `dupcoal` being slightly more computationally intensive than MLMSC-II, taking 12.017 cpu seconds to generate 1,000 gene families when $\lambda=\mu=2 \times 10^{-10}$, compared to 10.009 cpu seconds for MLMSC-II.

While our algorithm does not currently include the ability to simulate linked duplications, introgression, or horizontal gene transfers, it is straightforward to add these types of events in future iterations of our software. Linked duplications simply require that the topology of the daughter gene tree is correlated with the topology of the parent gene tree (proportionally to recombination distance), while both introgression and transfer can be accommodated by drawing gene trees from a different locus history (e.g. a history of introgression) in step 1 or step 3.

Complex coalescence: ILS, unsorted homologs, and copy-number hemiplasy

The process we are modeling—the interaction of duplication, loss, and coalescence—can be quite complex, leading to both conceptual and linguistic confusion. Here we briefly attempt to clarify some important terms that have been used in different ways by different researchers. While we try to avoid introducing new language, we do want to clearly define how terms are being used in this paper and their relation to previous uses.

Incomplete lineage sorting (ILS). ILS occurs when two or more lineages do not coalesce in the population in which they co-exist. When this happens, these lineages enter an ancestral population, at which time they may again coalesce, either with each other or with additional lineages that are present. If ILS results in lineages from the same original population coalescing first with a lineage from a different population, gene trees will be discordant with the species tree. This process can result in discordant gene trees at the parent and daughter loci considered here.

Importantly for the MLMSC model, the probability of ILS for two lineages is normally expressed as e^{-T}, where T represents the length of time during which the lineages can coalesce in the same population, and is usually taken as the length of the relevant branch in the species tree (see Rosenberg 2002; Bryant and Hahn 2020). However, when considering ILS between a pair of duplicates, the amount of time available for coalescence before ILS has occurred is limited to the time between the duplication event and the next speciation event (going backward in time). This time is always shorter than the length of the entire branch of the species tree on which the duplication occurred, which can lead to ILS even when all branches of the species tree are long. The importance of this distinction will hopefully become clearer in the next sub-section.

Unsorted homology. Gene duplication is fundamentally a process by which an existing haplotype in a population at one locus is inserted into another location. Because of this, the inserted DNA at the new daughter locus carries with it a history of coalescence with all the other haplotypes at the parent locus, and the full gene tree will reflect this history. In other words, all duplication events also involve coalescence events. Mallo et al. (2014) described a common phenomenon that can occur when duplicates coalesce together, which they called "unsorted homology." The general idea is that standard DL/DTL methods assume that duplication along a lineage also requires the joining of parent and daughter trees along that same lineage; if this does not occur, there is unsorted homology.

As an example, duplication of a gene along the *C* lineage (Figure 1b) is assumed to only lead to gene trees with the two copies in species *C* as sister to one another (Figure 1c). Any alternative joining of these trees will lead DL/DTL methods to incorrect reconciliations, which will in turn lead to incorrect homology assignments (i.e. orthology and paralogy). However, due to the coalescent process, many other trees can be formed (Figure 1d-g). Because of the mismatch between the lineage containing the duplicate and the lineage to which it coalesces in Figures 1d-g, all of these trees would be considered unsorted homologs. Note that, in the scenario shown here, the trees in Figures 1d and 1e are equiprobable with the topology in Figure 1c, making unsorted homology the most likely outcome.

It should be emphasized that all of the topologies presented in Figure 1 involve only a single duplication and no losses, and neither the parent tree nor the daughter tree is itself discordant due to ILS. In fact, the example in Figure 1d is not technically due to ILS either, since coalescence occurs in the same population of the species tree in which the duplication appeared (the ancestral population of species *B* and *C*).

This example highlights an additional complication of unsorted homology: it can occur with or without ILS, and even when the species tree only contains two species—and therefore neither parent nor daughter gene trees can themselves be discordant. Supplementary Figure S2 shows examples of unsorted homology in a two-species tree with ILS, while Supplementary Figure S3 shows an example without ILS. One can see that there

is very little time for coalescence (i.e. T) before ILS occurs in Supplementary Figure S2, as it is the timing of duplication relative to speciation, not the length of the branch in the species tree, that determines the probability of ILS. Some approaches to reduce the effects of ILS on reconciliation have required the presence of "apparent duplications" (Lafond et al. 2014) or "observable duplications" (Scornavacca et al. 2011)—essentially, duplication nodes that have more than one copy of the gene in a descendant species. While such approaches do prevent the inference of duplication when none has occurred, as shown by the examples of unsorted homology here (Figure 1d-f, Supplementary Figure S2, Supplementary Figure S3), they do not ensure the correct identification of duplication nodes, nor do they prevent excess losses from being inferred.

Hemiplasy. Gene trees at any locus in the genome can be discordant with the species tree. In such cases, the term "hemiplasy" (Avise and Robinson 2008) describes the incorrect inference of homoplasy that can arise when mutations occur on a discordant branch of a gene tree (i.e. an internal branch that does not exist in the species tree; see Figure 1 in Hahn and Nakhleh 2016). In the context of either duplication or loss, such mutations must occur on a discordant branch of a gene tree to lead to "copy-number hemiplasy" (CNH). Loss events can cause CNH when they occur either on a discordant parent tree (Supplementary Figure S4a) or a discordant daughter tree (Supplementary Figure S4b). In contrast, duplication events can only cause CNH on discordant daughter trees (Supplementary Figure S4c).

However, there is also a broader definition of copy-number hemiplasy in use, which includes any duplication or loss that does not fix in all descendant lineages, regardless of underlying gene tree topologies (Rasmussen and Kellis 2012; Wu et al. 2014). While this definition of CNH encompasses the stricter one given above, it also includes cases such as the one shown in Figure 1 (and Supplementary Figure S3), which the stricter definition does not include. Because the mutation in Figure 1 occurs in the common ancestral population of species B and C, but was only inherited by lineage C, this scenario would be considered to be hemiplasy by the broader definition. A similar event occurs in Supplementary Figure S3, even though there is no ILS, and no gene tree discordance is possible. Importantly, unlike the strict definition of CNH (and unlike unsorted homology),

there is not necessarily a genealogical signal left by events under this broader definition of CNH. For instance, if the duplication event in Figure 1a occurred a few generations later—after the split between species *B* and *C*, along a tip lineage—the scenario would not be considered hemiplasy by this broader definition, but would still produce the same range of gene trees (i.e. Figure 1c-g). (This would now involve ILS, however.)

Regardless of the exact definitions and terms used, many of these important processes are not modeled by common simulators (e.g. SimPhy does not simulate either type of hemiplasy) and are not considered by common reconciliation algorithms (e.g. DLCpar assumes that neither type of hemiplasy occurs). `Dupcoal` simulates all of these phenomena—as does MLMSC-II—but also reports the occurrence of each in generating the underlying gene trees.

Reconciliation using `reconcILS`

Overview of main algorithm

Our reconciliation approach uses nearest-neighbor interchange (NNI) rearrangements to model ILS and coalescence events, and to avoid invoking unnecessary duplications and losses. An NNI move on a rooted tree takes the three lineages defined by a single internal branch, for instance (*A*,(*B*,*C*)), and rearranges them into the two alternative trees, either (*B*,(*A*,*C*)) or (*C*,(*B*,*A*)). Any two bifurcating trees with the same set of labels at the tips can always be transformed one into the other by a series of NNI rearrangements (Robinson 1971).

NNI has been used previously in reconciliation algorithms to correct gene trees that may have been inferred incorrectly (e.g. Chen et al. 2000; Chaudhary et al. 2011; Górecki and Eulenstein 2011; Nguyen et al. 2013). Here, we use NNI because the effects of ILS on gene trees can be modeled as a series of NNI rearrangements (Than and Rosenberg 2013), not because there is necessarily a biological link between NNI and ILS. For instance, for three species undergoing ILS, there are two possible discordant topologies, each of which is one NNI move away from the species tree topology (e.g. Figure 2). For four lineages undergoing ILS, there are 14 possible discordant topologies, each of which are between one

and three NNI moves away from the species tree topology. (Note that it is the number of lineages undergoing ILS at a specific point on the tree that determines the NNI distance between trees, not the total number of tips in a tree.) Our algorithm takes advantage of this connection between NNI and ILS in order to reconcile gene trees, but would of course also accommodate gene tree error using the same NNI events.

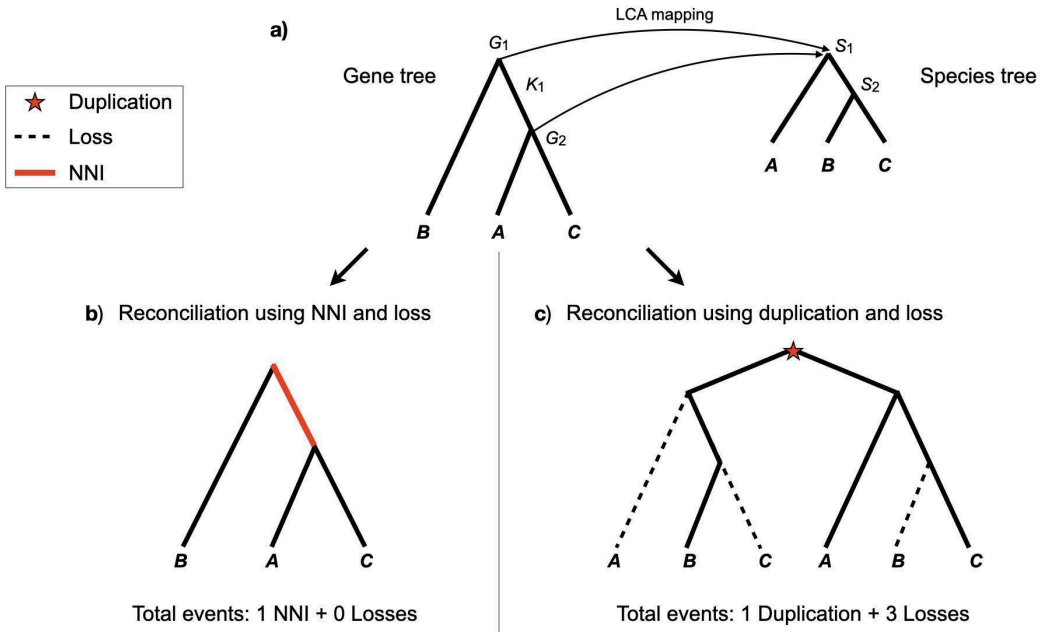

Figure 2. Summary of `reconcILS`. *a) The initial step generates an LCA mapping function between the gene tree and the species tree. Here, nodes G_1 and G_2 in the gene tree both map to node S_1 in the species tree; we therefore refer to S_1 as a multiply mapped node. Additionally, G_1 and G_2 together define a discordant branch, K_1, in the gene tree. For clarity, all mappings involving tip nodes have been excluded. b) The algorithm attempts to reconcile the discordant branch using NNI and loss. Here, reconciliation can be carried out with 1 NNI and 0 losses. c) The algorithm attempts to reconcile the discordant branch using duplication and loss. Here, reconciliation can be carried out with 1 duplication and 3 losses. `reconcILS` must decide between the reconciliations in b) and c) for every discordant branch.*

An overview of `reconcILS` is given in Algorithm 1 (see Supplementary Materials). Our approach combines multiple sub-approaches that are commonly used in DL reconciliation and other phylogenetic inferences. As a first step, we use the last common ancestor (LCA) algorithm (Zmasek and Eddy 2001) to create a mapping function between each node in a gene tree, G, and nodes in a species tree, S (Figure 2a). We refer to each of these pairwise relationships (arrows in Figure 2a) as "mappings" from here on.

Unlike in standard DL methods, the LCA mapping here does not represent a final reconciliation. Instead, this step identifies cases in which multiple nodes in the gene tree map to an identical node in the species tree. For instance, in Figure 2a nodes G_1 and G_2 in the gene tree both map to node S_1 in the species tree. We therefore call S_1 a "multiply mapped" node. Additionally, we refer to the branch in the gene tree bracketed by nodes G_1 and G_2 as a "discordant" branch (labeled K_1 in Figure 2a); such branches do not exist in the species tree. The focus of our method is in determining whether discordant branches should be resolved via NNI and loss (Figure 2b) or via duplication and loss (Figure 2c). Choosing among events (NNI, duplication, and loss) is carried out greedily for each branch, with no consideration of a globally optimal set of events among all discordant branches in the gene tree. Consequently, after choosing a set of events for one branch (given specified costs), the process repeats itself until there are no multiply mapped nodes and no discordant branches. Duplications implied by multiply mapped tip nodes in the species tree are inferred via standard DL methods; inferences of losses are explained in the Methods.

As with all parsimony-based reconciliations, `reconcILS` requires a cost for each event type: one for duplications (C_D), one for losses (C_L), and one for NNIs (C_I). Reconciliations can be very sensitive to costs, such that maximum parsimony solutions under one set of costs can be very different than under another set of costs. Although there are few ways to determine costs independently (but see Discussion), there are some approaches for finding the Pareto-optimal costs within a reconciliation context (e.g. Libeskind-Hadas et al. 2014; Mawhorter et al. 2019). Here, as in most commonly used

reconcilation algorithms, we allow the user to set costs. The default costs within `reconcILS` are set to $C_D = 1.1$, $C_L = 1$, and $C_I = 1$.

Labeling a reconciled gene tree

A full reconciliation of a gene tree, G, and species tree, S, will have reconciled all multiply mapped nodes, leaving no discordant branches. The total number of duplications, losses, and NNIs required for this reconciliation is the sum of each event type across all branches; these numbers are reported by `reconcILS`. We also aim to label the gene tree with these events—identifying where each event occurred in history—as this is an important output of reconciliation algorithms. As `reconcILS` traverses the species tree finding optimal local reconciliations for each branch, it also records the number and location of all events. Labeling therefore consists of placing these events on the gene tree. Traditional DL reconciliation algorithms label gene tree nodes as either D (for duplication) or S (for speciation); they may also label the location of losses, L. Here, we additionally aim to label the NNI events induced by ILS (I).

Importantly, there is no standard way to label ILS events on a tree, largely because there is no exact map between gene tree nodes and species tree nodes under ILS. A branch undergoes NNI (and ILS), not a node, so it is not clear which node flanking such a branch should be chosen to be labelled. This labeling problem therefore differs fundamentally from labeling under DL and DTL algorithms because of the lack of a one-to-one map between nodes in the two trees. Here, we have created our own labeling system for ILS, one that we think is both understandable and easy to work with.

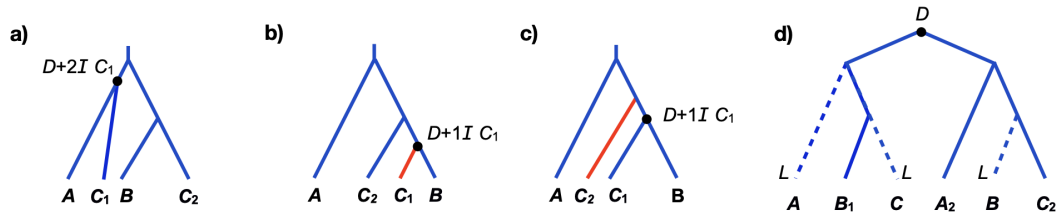

Figure 3. Labeling reconciled gene trees. *a) One possible reconciliation of the gene tree from Figure 1f labeled by* `reconcILS`*. The single ILS node (I) has been labeled with a branch that has been rearranged by NNI (C_1), the number of NNIs it has undergone (2), and the fact that it also represents a duplication (D). b) The gene tree from Figure 1d labeled by* `reconcILS`*. This tree is correctly labeled, but note that it is the same topology as in panel c (see Methods for further explanation). c) The gene tree from Figure 1e labeled by* `reconcILS`*. Note that the program has labeled the bottom-most node flanking the discordant branch with D and I, although that is not the true history. d) A gene tree with only duplications and losses (L) labeled by* `reconcILS`*. Internal branches on a tree can also be labeled with L, though none are shown here.*

Our solution for placing the *I* label on nodes of gene trees in `reconcILS` is to denote both a daughter branch affected and the number of NNI moves that were associated with this branch. A few examples are shown in Figure 3, which also helps to illustrate why these choices were made. Figure 3a labels the gene tree from Figure 1f—with more complete tip names—with output from `reconcILS`. As can be seen, this tree has one node (denoted with a small black dot for clarity) labeled with both duplication and ILS. Further, the labeling tells you which daughter branch was reconciled with NNI (C_1), and how many NNI moves were required to reconcile it (2). Without such information, simply labeling the node with *I* would not have distinguished histories involving coalescent events along the *A* branch from those along the C_1 branch. Further details of this labeling are provided in the Methods, and caveats are covered in the Discussion.

`reconcILS` also labels duplications and losses in the absence of ILS. Figure 3d shows how both individual duplications and losses are labelled: observe that neither requires a label with a specified number of events (like the *I* label), nor does either need to specify on which branch they act. `reconcILS` places the label *L* on any node in the implied gene tree that is not represented by a gene copy; for example, generic tip nodes *A*, *C*, and *B* in Figure 3d do not exist in the input gene tree itself, but are labelled with an *L*. `reconcILS` can do this because every duplication event is represented internally by generating a new copy of the species tree below the duplicated node—if losses occur on these implied trees, we can track their timing as well. Note that internal nodes can also be labelled with an *L*, so that the timing of loss events must be inferred by comparing the state of parent and daughter nodes (i.e. losses on internal branches will lead the descendant internal node and all of its daughter nodes to be labelled with an *L*). Finally, any node without a label can be inferred to be a speciation node.

Results

Accuracy and advantages of `dupcoal`

We assessed the accuracy of `dupcoal` by first simulating 1000 gene trees along a 40-tip species tree with no duplication and no loss (Supplementary Figure S5). These simulations therefore only model ILS as a cause of discordance. We compared the degree of discordance in these simulations to ones carried out using the same species tree with SimPhy (after appropriate branch-length scaling; see Methods). We found a very high correlation between concordance factors on each branch calculated from gene trees produced by the two methods (ρ=0.99, P=3.53 x 10^{-31}), indicating that we are accurately simulating ILS. We also carried out simulations in `dupcoal` with both duplication and loss, again comparing our results to SimPhy (after scaling duplication and loss rates appropriately; see Methods). Again, the average number of duplications and losses per simulated gene tree (n=1000) were highly similar and not significantly different between the two methods: 3.82 vs. 3.84 duplications and 3.08 vs. 3.12 losses for `dupcoal` and SimPhy, respectively (P=0.88 for duplications and P=0.64 for losses; t-tests), indicating that we are accurately simulating duplications and losses in `dupcoal`.

To highlight additional features contained within `dupcoal`, we also tracked the number of simulated gene trees in which copy-number hemiplasy occurred (this is a standard output reported by `dupcoal`). Out of 1000 gene trees simulated on the 40-tip tree with duplication, loss, and ILS, 511 contained at least one (strict) hemiplasy event. As a straightforward way to demonstrate the effect of such events, we separately inferred the number of duplications and losses on gene trees with and without copy-number hemiplasy using ETE3 (Huerta-Cepas et al. 2016). While ETE3 implements a DL reconciliation algorithm that does not model any type of ILS (see next section), this contrast shows the effects of hemiplasy above and beyond the effects of ILS. As expected, both the number of inferred duplications (2.61 excess duplications; P=6.44 x 10^{-11}, t-test) and the number of inferred losses (8.31 excess losses; P=1.59 x 10^{-12}, t-test) were higher among the trees

containing hemiplasy. These results emphasize the importance of having a simulation engine that can simulate hemiplasy.

Accuracy and advantages of `reconcILS`

As a simple test of DL reconciliation algorithms—which do not allow for ILS—we used the DL method implemented in ETE3 (Huerta-Cepas et al. 2016). We initially ran simulations in `dupcoal` without duplication or loss (i.e. ILS only); this allows us to see incorrectly inferred duplications and losses more easily. As expected, on a species tree with only 3 tips, ETE3 always inferred 1 extra duplication and 3 extra losses on gene trees with discordance due to ILS (not shown). On a larger species tree with 40 tips, ETE3 inferred a large excess of duplications and losses when only ILS is acting (9.5 duplications and 34.4 losses per gene tree, on average). On datasets simulated on a 40-tip tree with duplication, loss, and ILS, the accuracy of ETE3 is low for losses (ρ=0.23) but is still fairly good for duplications (ρ=0.79) (Supplementary Figure S6).

In contrast, `reconcILS` accurately infers the numbers of duplications and losses (Figure 4a and 4b), as well as ILS events (Figure 4c) on the 40-tip tree. The correlation between inferred and simulated duplications and losses were high (ρ=0.95 and 0.63, respectively). We also observed a high correlation between the inferred and simulated number of NNI events needed to reconcile a gene tree (ρ=0.67). ILS immediately followed by loss will not leave a signal in the resulting gene trees, leading to a slight undercounting of NNI events caused by ILS. It should also be noted that all inferences here are carried out with the true gene trees, as we have not included any (error-prone) steps to infer the trees from sequence data.

We also examined results from `reconcILS` when using different costs (the numbers in the previous paragraph all used the default costs: $C_D = 1.1$, $C_L = 1$, and $C_I = 1$). When increasing the cost of duplication and loss ($C_D = 4$, $C_L = 3$, and $C_I = 1$), the results remained largely the same, with high correlations between inferred and true duplication, loss, and NNI: ρ=0.92, 0.61, and 0.69, respectively). However, when increasing the cost of

ILS ($C_D = 1$, $C_L = 1$, and $C_I = 5$), the results are much worse, closely resembling those from ETE3: ρ=0.79, 0.23, and 0.25, respectively. This match between `reconcILS` and ETE3 should not be surprising, as having a high cost of ILS is similar to having no ILS possible at all.

As we did with ETE3 in the previous section, we also contrasted the number of duplications and losses inferred by `reconcILS` on simulated gene trees with and without copy-number hemiplasy (using default costs). Unfortunately, `reconcILS` also does not deal perfectly with strict hemiplasy, though in a slightly different way than ETE3. Rather than overestimating both the number of duplications and losses, `reconcILS` appears to generally infer a true ILS event as a loss event instead. This occurs because (for instance) a duplication on a discordant daughter gene tree branch will duplicate a non-sister pair of lineages (e.g. *A* and *B*), which `reconcILS` interprets as a duplication and a loss (of *C*, in this case) rather than as a duplication and an NNI event (Supplementary Figure S4). Consistent with this, we observed no excess duplications in the simulations with hemiplasy (0 extra duplications, on average), but significantly more losses (1.36; P=2.2 x 10^{-25}, *t*-test) and significantly fewer NNI events (3.63; P=4.1 x 10^{-13}, *t*-test) compared to simulations that did not have hemiplasy.

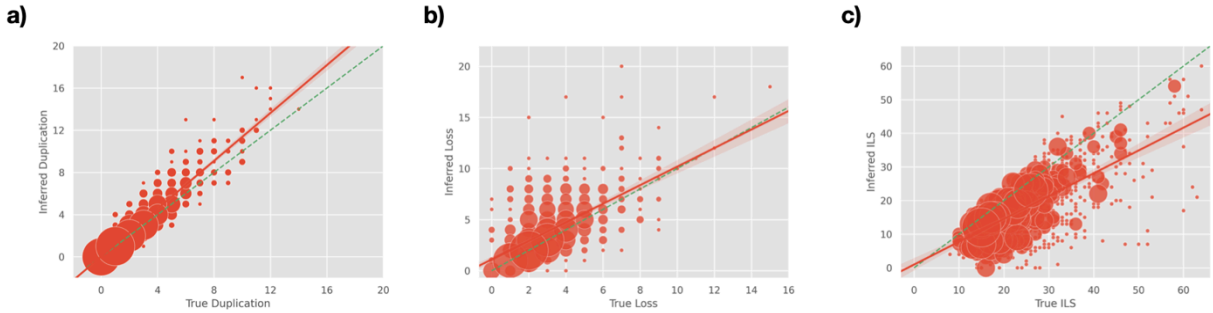

Figure 4. Accuracy of `reconcILS` on data simulated from the 40-tip tree. *a) Number of inferred versus true duplication events. b) Number of inferred versus true loss events. c) Number of inferred versus true ILS events. Circle size is proportional to the number of reconciliations with each value, with jitter applied for clarity. Orange line represent best-fit*

regression (as calculated in Seaborn; Waskom 2021), plus confidence intervals. The dashed line represents y=x. See Supplementary Figure S6 for results from ETE3 on the same data.

We also compared the results from `reconcILS` with another reconciliation method that allows for ILS, DLCpar (Wu et al. 2014). DLCpar was written in Python 2, and—possibly because we had to run it in a virtual environment—could not run on the larger gene trees produced by the 40-tip tree. Therefore, we compared the performance of `reconcILS` and DLCpar on 1000 gene trees simulated on a 3-tip species tree. Overall, the accuracy of both methods was high for duplication (`reconcILS`, ρ=0.97; DLCpar, ρ=0.97; Supplementary Figure S7a). The correlation between simulated and inferred ILS events was also high for both (`reconcILS`, ρ=0. 85; DLCpar, ρ=0. 76; Supplementary Figure S7b). It is important to note that DLCpar infers the number of ILS events, rather than the number of NNI rearrangements that occur within such an event (which are what is inferred by `reconcILS`). To ensure that a fair comparison can be made, results reported for DLCpar are simulated vs. inferred ILS events, rather than NNI events. In both cases the methods are doing well in their respective inferences, but with slightly higher accuracy for `reconcILS`.

There is quite a difference between `reconcILS` and DLCpar in the inference of loss (correlations of ρ=0. 84 and ρ=0. 48, respectively; Supplementary Figure S7c). Upon closer inspection, the lower correlation in DLCpar is due to many loss events of one particular type not being counted. Specifically, DLCpar does not count losses that occur above the root of the gene tree being considered. This means that, for instance, in gene trees with the topology (B,C), DLCpar does not count the loss that occurred along the A branch (assuming the species tree in Figure 1). We confirmed this behavior on slightly larger trees in DLCpar, as well as in all trees in ETE3—no losses above the root of a gene tree are considered. The difference in counting of losses among algorithms arises because `reconcILS` explicitly traverses the species tree, while the other methods do not. This enables our method to track losses that occur above the root of the input gene tree, but below the root of the input species tree. However, there may be some cases in which the behavior of DLCpar and ETE3 is preferred. For instance, since gene families must originate on some branch of the species

tree (e.g. Richter et al. 2018), forcing reconciliation algorithms to infer losses on branches that pre-date this appearance would be incorrect. It is not clear which approach to counting losses one should prefer when analyzing real data.

Finally, we assessed the runtime and memory requirements of `reconcILS` compared to ETE3 on a single node of the Indiana University Research Desktop, running an Intel Xeon processor with 100 GB of RAM (DLCpar can only be run in Python 2, which makes comparisons highly unfair). On a 3-species tree, both programs are extremely fast (average time of 0.0086 seconds per gene tree for reconcILS and 0.013 seconds per gene tree for ETE3; Supplementary Figure S8a) and both use very little memory (average peak memory per gene tree of 0.075 MB for `reconcILS` and 0.054 MB for ETE3; Supplementary Figure S8b). Both programs remain very fast on data produced from a 40-species tree, though the faster speed of ETE3 is also evident (4.9 seconds per gene tree for `reconcILS` and 1.3 seconds for ETE3; Supplementary Figure S8a). Regardless, this means that the entire 1000-gene-tree simulated dataset on the 40-species tree took less than 84 minutes to analyze using `reconcILS` on a moderately powerful computer. Similarly, the memory requirements of `reconcILS` increase on the 40-tip tree (18.79 MB per gene tree for `reconcILS` and 3.19 MB for ETE3; Supplementary Figure S8a), but are not onerous for modern computers. As the 40-species tree can contain many more events (duplication, loss, or ILS), we also plotted the average time and memory required for trees containing different numbers of total events (Supplementary Figure S8c,d). Both programs show approximately linear increases in time and memory with the number of events, which is promising for analyzing larger datasets with larger species trees.

Analysis of primate genomes

To further demonstrate the power and accuracy of `reconcILS`, we applied it to a dataset of gene trees inferred from 23 primate genomes (Vanderpool et al. 2020; Smith et al. 2022). These genomes all come from the suborder Haplorhini, which includes all primates except the lemurs and lorises (suborder Strepsirhini). Among these genomes, we analyzed two sets of gene trees that were previously inferred. First, we considered 1,820

single-copy orthologs. This dataset is not expected to contain any duplicates, though it does contain losses. However, because there are high levels of ILS among the primates sampled here (Vanderpool et al. 2020), the dataset serves as an ideal comparator for reconciliation approaches with and without ILS. Second, 11,555 trees containing both orthologs and paralogs were used to infer duplications, losses, and ILS among the primates. We used both `reconcILS` and ETE3 to analyze the two datasets, assigning inferred events to each branch of the primate tree.

Single-copy ortholog dataset

Results obtained from analysis of only the single-copy orthologs are shown graphically in Figure 5a (see Supplementary Figure S9a for numerical values on each branch). The pattern that stands out from this graph is that ETE3 is vastly overestimating the number of duplications and, especially, the number of losses. Because ETE3 does not model ILS, all such events must be reconciled by one duplication (placed above the discordant branch) and three losses (placed below the discordant branch; Hahn 2007). We can see this effect in the data by looking at the correlation between the gene concordance factor (gCF) estimated for a branch with the number of duplicates inferred on its parent branch. Gene concordance factors quantify the fraction of loci that contain the same branch as the species tree; when single-copy genes are analyzed, low gCFs can indicate discordance due to ILS or other factors, like gene tree estimation error (Baum 2007; Lanfear and Hahn 2024). The correlation between gCFs for these data (estimated in Smith et al. 2022) and duplications inferred by ETE3 on parent branches is ρ=-0.56. Lower gCFs lead to higher numbers of duplicates, as expected when ILS is not accounted for—this problem is not unique to ETE3, but should be expected for all standard DL and DTL approaches. Such errors can lead to incorrect inferences of bursts of duplication and loss.

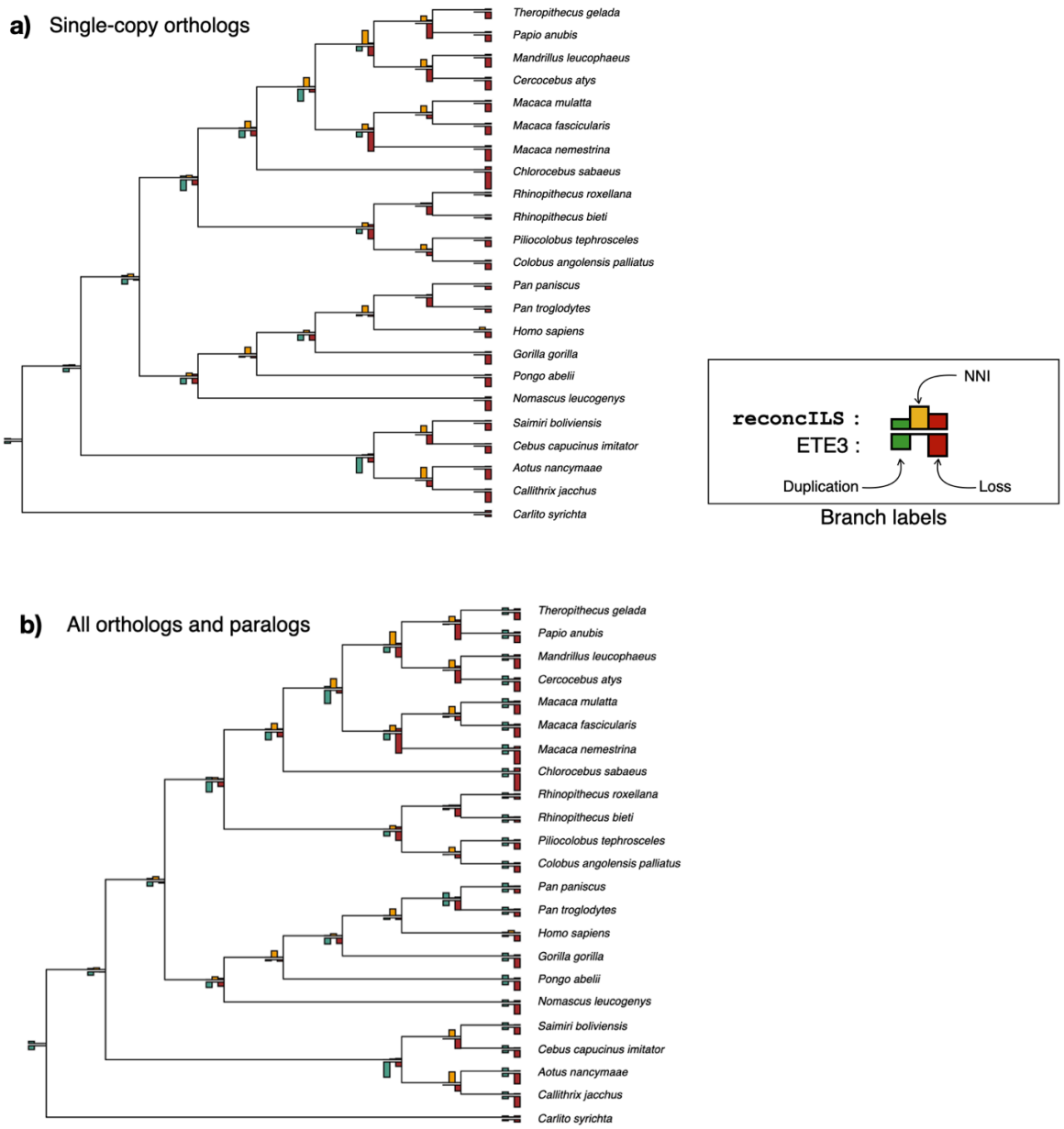

Figure 5. Results from primates. *a) Analysis of 1,820 single-copy orthologs. Both* `reconcILS` *and ETE3 were used to analyze the data, with duplication, NNI, and loss events mapped to each branch of the species tree. On each branch, the bar plots show results from* `reconcILS` *on top, with results from ETE3 mirrored below. ETE3 cannot infer NNI events, so this bar is always set to 0. b) Analysis of 11,555 gene trees containing both orthologs and paralogs. Results are presented in the same manner as in panel a. However, note that the*

height of all bars is normalized within a panel, so that heights of bars within a tree can be compared among branches, but the heights of bars between the trees cannot be compared. Supplementary Figure S9 shows the absolute counts of all events from both panels and Supplementary Figure S10 shows species tree with branch lengths in substitution units rather than as a cladogram, as shown here.

Using the same reasoning, we can assess the accuracy of `reconcILS` by examining the correlation between NNI events inferred by our algorithm and gCF values for single-copy genes. High NNI counts indicate more ILS, and therefore this measure should be negatively correlated with gCF values for the same branch. Indeed, the correlation between NNI counts and gCF values is ρ=-0.83, demonstrating the high accuracy of `reconcILS` in quantifying ILS (or in correcting for gene tree error). However, `reconcILS` does sometimes infer a small number of duplications (Figure 5a; Supplementary Figure S9a). While it is possible that some of the single-copy genes we analyzed represent so-called pseudoorthologs—and therefore do contain actual duplication events—these are expected to be rare (Smith and Hahn 2022). Instead, these false duplicates are likely due to ILS among four or more lineages. Recall that, although our algorithm has been developed to account for rearrangements due to the coalescent process, ILS among four lineages can result in gene trees that are three NNI moves away from the species tree; ILS among more lineages will produce trees that are even further away. Therefore, multiple NNI events are sometimes simply more costly than duplication and loss, even when ILS is the true cause of discordance.

Full gene tree dataset

We also examined patterns of ILS, duplication, and loss for the entire primate dataset. The results from the full set of gene trees (Figure 5b; Supplementary Figure S9b) largely mirror the results from the single-copy orthologs (which are a subset of the entire dataset). Part of the apparent similarity is simply due to normalizing the two datasets separately, but several non-graphical factors also contribute to the pattern. First, although there are many true duplicates in the full dataset, the vast majority of gene trees only have

events on tip branches: 7,693 out of 11,555 gene trees have duplicates specific to one species (Smith et al. 2022). Another ~1,000 trees have duplicates specific to two species. Combined with the 1,820 single-copy genes, it becomes clear that there are in fact relatively few trees with duplicates on internal branches even in the full dataset.

Second, errors in inferring duplications and losses due to the coalescent process still dominate the results from standard DL reconciliation (i.e. ETE3), possibly even more so than in the single-copy dataset. Because ILS (and introgression) can affect all gene trees, trees with duplicates can experience twice as much discordance when there are twice as many branches. One example of this can be seen among the Papionini (macaques, baboons, geladas, and mandrills): ETE3 has to infer >6,000 losses on each of the tip-branches in this clade, and more than 10,000 losses on most of the internal branches (Figure 5b; Supplementary Figure S9b). There are also many thousands of duplicates inferred on internal branches of this subclade, though these species have the same number of total protein-coding genes as all the others analyzed here (Vanderpool et al. 2020). Although the inferred duplications and losses in this group were also relatively higher in the single-copy dataset, it is clear that even a small number of true duplicates can have an outsized effect on methods that cannot account for ILS.

The analysis using `reconcILS` is also highly similar between the two datasets, again largely because most duplicates are confined to tip branches. `reconcILS` is able to deal with discordance due to coalescence, even in the presence of duplication: the correlation between NNI events inferred on each branch of the species tree from the single-copy orthologs and the full dataset is $\rho = 0.99$. Although the numbers of events are of course larger with the larger dataset (Supplementary Figure S9b), the effect of ILS on any given branch of the species tree is experienced by all gene trees. Given its ability to deal with ILS, duplication events across the primate tree are more reliably inferred with `reconcILS`.

Discussion

We have introduced a new reconciliation algorithm and associated software package (`reconcILS`) that considers discordance due to duplication, loss, and coalescence. To better demonstrate the accuracy of our method, we have also developed a new simulation program (`dupcoal`) that generates gene trees while allowing these processes to interact. Our results demonstrate that `reconcILS` is a fast and accurate approach to gene tree reconciliation, one that offers multiple advantages over alternative approaches, even those that also consider ILS (e.g. DLCpar; Wu et al. 2014). Nevertheless, there are still several caveats to consider, as well as future directions to explore.

Despite our demonstration of good performance in practice, our method is heuristic: we have no theoretical proof that our method finds most parsimonious reconciliations. While such proofs often require strict assumptions—e.g. the proofs for DL algorithms assume there is no ILS—they are very helpful in providing confidence under the specified conditions. However, given the high accuracy of `reconcILS` demonstrated with both simulations and empirical data from single-copy genes, we hope the heuristic nature of our algorithm is not a barrier to its utility. Future theoretical work on this or similar approaches using NNIs (e.g. Górecki and Eulenstein 2011) would of course be of benefit to the field.

As mentioned earlier, there is often ambiguity in how to label nodes in gene trees affected by ILS and duplication (e.g. Figures 3b and 3c). This is due in part to the nature of ILS—it is not clear which node to label because there is not a one-to-one map between gene tree nodes and species tree nodes—and in part to the intrinsic lack of information contained only within tree topologies when coalescence and duplication interact. The difficulty in labeling is not confined to this work: DLCpar does not label nodes, and RecPhyloXML, a format for representing reconciled trees, does not include a way to label ILS events at all (Duchemin et al. 2018).

Nevertheless, labeling gene trees serves multiple important functions, and is often interpreted as biological truth. For example, multiple methods use inferred gene tree labels to identify orthologous genes (genes related by speciation events; Fitch 1970). The general

approach taken is to label nodes using reconciliation or similar approaches, treating any node not labeled with a *D* as a speciation node; genes whose common ancestor is a speciation node are subsequently defined as orthologs. Both standard DL reconciliation and so-called "overlap" methods (e.g. Huerta-Cepas et al. 2007; Emms and Kelly 2019) will consistently label the top-most node flanking a discordant branch with a duplication event. This means that, in cases like the one shown in Figure 3b, these methods will label the wrong node with a *D*, and subsequently identify the wrong pair of genes as orthologs. (Figure 3c would be labeled correctly; recall that this outcome has an equal probability to the one in Figure 3b in the scenario considered here.) Cases such as these have been referred to as "unsorted homologs" (Mallo et al. 2014), and will occur any time duplication and coalescence interact, either before or after a speciation event. They can also occur even if only two species are analyzed (e.g. Supplementary Figures S2 and S3), which means that approaches that simply collapse short branches to avoid ILS will also be affected. Although `reconcILS` currently only labels one node with a *D*, in the future we could allow for multiple possible labelings in order to emphasize the inherent ambiguity of some labels.

The algorithm used by `reconcILS` deals, in part, with an additional interaction between duplication, loss, and coalescence: hemiplasy. Hemiplasy is the phenomenon by which a single mutational event on a discordant gene tree will be interpreted as multiple events ("homoplasy") when reconciled to the species tree (Avise and Robinson 2008; Hahn and Nakhleh 2016). In analyses of nucleotide substitutions, generic binary traits, and even continuous traits, hemiplasy requires that mutations have occurred on a discordant branch of a gene tree (Mendes and Hahn 2016; Guerrero and Hahn 2018; Hibbins et al. 2023). In the context of either duplication or loss, such mutations would have to occur on a discordant branch of a gene tree to lead to copy-number hemiplasy. The simulation engine introduced here, `dupcoal`, simulates hemiplasy events and reports any gene trees affected. Our results show that these events lead to increased error in reconciliation, and are therefore an important process to include. Interestingly, such events have little effect on species tree inference when paralogs are used for this purpose (Li et al. 2024).

Unfortunately, strict copy-number hemiplasy events as defined above are still not accurately inferred by `reconcILS`: we have found that our algorithm consistently turns NNIs into losses when hemiplasy occurs. While the total number of events does not seem to increase, it is not ideal that NNIs are mistaken for losses. However, `reconcILS` does deal accurately with copy-number hemiplasy under the broader definition in use (see section on "Complex coalescence…"), and `dupcoal` also simulates and records such events. In contrast, DLCpar assumes that this type of hemiplasy does not occur, and SimPhy does not simulate this type of hemiplasy.

The algorithm used by `reconcILS` deals with ILS differently than does DLCpar. While DLCpar treats ILS as an event, `reconcILS` considers NNIs as events induced by ILS. It is not clear that either approach should necessarily be preferred on first principles. In simple cases involving three species and a single discordant branch (e.g. Figure 2), ILS and NNI are more or less equivalent, though it should be noted that the process of ILS can also lead to concordant trees and is not synonymous with discordance. However, in cases with more than one discordant branch there is no simple relationship between the two. For instance, in a 4-tip tree undergoing ILS, lineages may undergo ILS across two branches, but result in between 0 and 3 NNI events (Than and Rosenberg 2013). We think that directly modeling NNIs—as is done in `reconcILS`—is a natural way to express the events needed to reconcile a gene tree with a species tree, including discordance due to gene tree error. In fact, NNI and other moves have been used previously in reconciliation algorithms to correct gene trees that may have been inferred incorrectly (e.g. Chen et al. 2000; Chaudhary et al. 2011; Górecki and Eulenstein 2011; Nguyen et al. 2013). Though these algorithms minimize DL reconciliation costs using NNIs, they could also have been used within `reconcILS` for each local reconciliation step. Regardless of the exact algorithm used, the utility of such methods also provides a motivation for using `reconcILS` to obtain more accurate inferences of duplications and losses in the presence of estimation error, not just ILS.

One relatively straightforward improvement that could be made to `reconcILS` is in the cost of different events. Costs must be specified in `reconcILS` ahead of time, but it is difficult to determine biologically realistic costs for duplication, loss, and ILS. One approach to determining duplication and loss costs would be to infer the number of such events using an alternative, non-reconciliation approach (e.g. Mendes et al. 2020), setting the cost in `reconcILS` to be inversely related to the inferred number of events of each type. Varying rates of duplication and loss can also be inferred along different branches of the species tree, and so the costs of these events could vary commensurately along the tree. Similarly, we can envision a simple way to scale the costs of NNI with the length of species tree branches: this is because the probability of discordance due to ILS is inversely proportional to internal branch lengths on a species tree (Hudson 1983). We could therefore make NNI events on longer branches of the species tree more costly, which would properly weight the relative probability of such events locally on the tree.

There are also several extensions that could be made to `dupcoal`, our new simulation engine. Currently, `dupcoal` draws a daughter gene tree from the multispecies coalescent (MSC) at random: this implicitly assumes that the parent and daughter trees are completely unlinked. It is straightforward, however, to draw the daughter tree at a given recombination distance from the parent tree, ranging from fully linked to fully unlinked. In addition, though the software currently draws gene trees from the MSC, we could also include introgression as a source of discordance by drawing trees from the multispecies network coalescent (Meng and Kubatko 2009; Yu et al. 2011). In this case, users would simply input a species network to simulate from, rather than a species tree. We should also note the differences between the current version of `dupcoal` and alternative simulators. As mentioned previously, SimPhy (Mallo et al. 2016) does not simulate hemiplasy of any kind, as it uses the same three-tree model of Rasmussen and Kellis (2012) and Wu et al. (2014). However, this method does include gene transfers, which are not simulated by `dupcoal`. The MLMSC-II simulator of Li et al. (2024) simulates hemiplasy under the same model used here; `dupcoal` is therefore most similar to this method, with a few minor differences. Specifically, `dupcoal` omits two rejection steps that lead to discarded

duplication events in MLMSC-II and simulates duplications directly on parent and daughter gene trees, rather than using a coalescent-rate process. Despite these minor differences, both methods are capable of generating a wide variety of gene trees impacted by duplication, loss, coalescence, and the interactions of these processes. Unlike MLMSC-II, `dupcoal` records the exact counts of several different event types, a necessity when evaluating the performance of reconciliation approaches.

Finally, we envision multiple ways in which the `reconcILS` algorithm could be used in conjunction with other methods. For instance, ASTRAL-Pro (Zhang et al. 2020) infers a species tree from gene trees containing paralogs by first labeling nodes in each gene tree as either duplication or speciation. The authors of this software rightfully recognize the problems with labeling discussed here, but nevertheless the labeling step might benefit from methods that include ILS. Likewise, GRAMPA (Thomas et al. 2017) allows for reconciliation of polyploidy events, but does not include coalescence as a source of discordance; its algorithm could easily accommodate this extension. There are also multiple approaches that can be used to correct gene trees via reconciliation (e.g. El-Mabrouk and Ouangraoua 2017; Christensen et al. 2020). These and many other tasks might benefit from approaches that model duplication, loss, and coalescence in a single framework.

Methods

Building and testing `dupcoal`

The simulator `dupcoal` is written in Python 3 and uses functions from DendroPy (Sukumaran and Holder 2010) to generate trees under the MSC and to manipulate parent and daughter trees. It also uses functions from NumPy (Harris et al. 2020) to draw coalescent times, duplication times, and loss times from exponential distributions.

To test the accuracy of `dupcoal` we simulated gene trees along a species tree with 40 tips (Supplementary Figure S5). The species tree topology was drawn at random using the program ape (Paradis and Schliep 2018), with the "Grafen" method for drawing branch

lengths (scaled up by 10); this tree was used as input to both `dupcoal` and SimPhy. The amount of ILS in this tree, as measured by gene concordance factors (Minh et al. 2020a), ranged from 29.1 to 99.6; in comparison, the values estimated from the primate tree in Vanderpool et al. (2020) ranged from 27 to 96, so the two are quite similar. Because `dupcoal` specifies branch lengths in coalescent units, while SimPhy uses time, branch lengths were converted by assuming a population size of N=10 and a generation time of 1 year. Simulations were run on this tree with only ILS (setting duplication and loss rates to $\lambda = 0$ and $\mu = 0$, respectively). When using positive duplication and loss rates (see next paragraph), they were scaled to the appropriate branch lengths to make them comparable in `dupcoal` and SimPhy. Comparisons between `dupcoal` and SimPhy with no duplication and loss were carried out by calculating gene concordance factors (gCFs) on the species tree using IQ-TREE2 (Minh et al. 2020a).

To test the accuracy of inferences using `reconcILS` and ETE3, the same 40-taxon tree was used for simulations in `dupcoal` with ILS, duplication, and loss ($\lambda = 0.03$ and $\mu = 0.03$). These parameter values were chosen so that the simulations produced a large number of duplications and losses to assess accuracy with, not because they are biologically motivated. DLCpar is written in Python 2 and could not be run in reasonable time on the 40-taxon tree. We therefore also simulated data on a 3-taxon tree with topology (A:1.5,(B:0.5,C:0.5):1), where branch lengths are measured in coalescent units and duplication and loss rates were the same as on the larger tree. The default costs ($C_D = 1.1$, $C_L = 1$, $C_I = 1$) were used for all analyses of simulated data for all methods, except where noted. Note that it is the relative value of costs that matters for analysis, and the absolute values do not correspond to any biological value.

Implementation of `reconcILS`

Inputs and outputs

Given a rooted, bifurcating species tree, S, and a rooted, bifurcating gene tree, G, as input, `reconcILS` outputs the number of duplications, losses, and NNIs required to

reconcile *G* and *S* given event costs. Inferred events are all also assigned to nodes in the gene tree.

Traversing trees

After generating an initial LCA mapping between a gene tree and the species tree, `reconcILS` performs a preorder traversal of the species tree. For each species tree node, the algorithm checks how many gene tree nodes are mapped to it. If there are fewer than 2 mappings, the algorithm asks whether speciation or loss can be inferred. As is standard in reconciliation, species tree nodes mapped to one gene tree node imply a speciation event at that gene tree node—these do not incur a cost. Nodes mapped to zero gene tree nodes imply at least one loss in the gene tree. Our program asks whether the parent node of a node with 0 mappings itself has 0 mappings. If not, a loss is inferred; if so, no event is recorded, because the loss must have occurred further up the tree. If there are 2 or more mappings the species tree node is multiply mapped, and a function to carry out "local reconciliation" is called. Local reconciliation is described in more detail in the next section, but here can simply be described as determining the events that have generated the multiple mappings to a single species tree node. It should also be noted that if a duplication node is inferred for a gene tree, both sub-trees induced by the duplication are further locally reconciled (Algorithm 1); this means that `reconcILS` can detect cases in which multiple independent losses have occurred along the same lineage (e.g. Supplementary Figure 1).

If there are only two adjacent nodes in the gene tree mapped to a node in the species tree (i.e. one discordant branch, as in Figure 2), then the method proceeds as described below. However, when there are more than two nodes mapped to an internal species tree node—and therefore more than one discordant branch in the gene tree—we must decide which branch to reconcile with the multiply mapped node first. We have found cases where the order in which the branches are resolved matters, with only one of the choices leading to an optimal reconciliation. Algorithm 2 (see Supplementary Materials) describes an algorithm for choosing which discordant branch to locally reconcile first. This algorithm calculates the "bipartition cost" of each possible branch to reconcile. We calculate the

bipartition cost of a branch by examining the two trees induced by an NNI move applied to the branch. For each of these new topologies, we calculate costs using two measures, both of which are applied independently to the two sub-trees extending from the root of these topologies (denoted by the red vertical line in Supplementary Figure 11b,c). First, we count the number of bipartitions present in these sub-trees but not present in the species tree ("bipartition" cost in Supplementary Figure 11); second, we count the number of lineages missing relative to the node in the species tree the root of the topology is mapped to ("lost" cost in Supplementary Figure 11). The bipartition cost of a single branch therefore requires eight values to be calculated, four for each of the trees created by NNI (e.g. Supplementary Figure 11b). Such calculations are simple and can be carried out quickly, and by choosing the branch with the lowest bipartition cost we can more accurately proceed with reconciliation.

Once reconciliation for a single multiply mapped node in the species tree is carried out (i.e. it is no longer multiply mapped), our algorithm begins again: a new round. In each round we generate an LCA mapping function and continue traversing the species tree, counting gene tree nodes mapped to each species tree node.

Choosing among events

In order for our algorithm to more efficiently and optimally choose among different evolutionary events (e.g. Figure 2b vs 2c), we introduce the concept of a "local reconciliation" for a multiply mapped node. Here, a species tree node is locally reconciled if the number of gene tree nodes mapped to it, m, is reduced. For internal nodes of the species tree, this, by definition, also reduces the number of discordant branches in the gene tree.

The use of local reconciliation has several advantages. Most importantly, it provides a criterion for moving forward: we can determine if a reconciliation has been achieved within a single round of the algorithm. It also makes it clear that there are some scenarios that can only be reconciled using duplication and not NNI (e.g. a multiply mapped tip node), in which case choosing events is straightforward. In a single round of our algorithm, we attempt to locally reconcile a single multiply mapped node using both NNI+loss and duplication+loss (Figure 2).

Note that, given this definition, a species tree node can be locally reconciled by NNI while still remaining multiply mapped; e.g., if $m = 3$ is reduced to $m = 2$, the node still requires more reconciliation. In this case our algorithm will carry out up to m NNI moves within a single round, as long as each one results in a local reconciliation. For example, starting with $m = 3$, the first NNI move will locally reconcile a node if one of the two NNI topologies reduces the number of mappings to the species tree (we have never observed a case where both NNI topologies reduce this number). In this case, $m = 2$, and we now carry out another NNI move on the updated gene tree. Only one move needs to be examined at this point, however, since the back-NNI rearrangement would return us to the starting state. The cost of all of these NNI moves is then compared to the cost of duplication and loss at the end of the round. Duplication+loss naturally allows for more than one of each type of event for a local reconciliation, and is therefore not handled differently when $m > 2$.

Finally, at the end of a round, `reconcILS` will sum the cost of all events in order to choose the optimal local reconciliation. If there are equally optimal reconciliations, `reconcILS` will report the duplication+loss solution.

Labeling events

Our approach for labeling ILS on reconciled gene trees is described above in the "New Approaches" section. Here, we consider two important caveats to this approach, one perhaps more obvious than the other. First, a specific labeling is associated with a specific reconciliation. That is, if we had found a different optimal reconciliation for the tree in Figure 3a (for instance, one duplication and two losses), we would have had different labeled events. Therefore, a labeling is always reported in the context of a particular reconciliation.

Second, as mentioned, for some ILS events it is not always clear which of the two nodes flanking a discordant branch to label with I: this is why there is no exact map between gene tree node labels and species tree nodes in the presence of ILS. For a discordant branch, we have arbitrarily set `reconcILS` to always choose the node closest to the tips as I. In some cases, this choice should not matter very much. Consider the toy

reconciliation in Figure 2b—should we have placed the NNI event on node G_1 or on node G_2? In this case the choice does not seem to imply any difference in evolutionary histories. In contrast, consider the reconciliations shown in Figure 3b and 3c—these represent the output of `reconcILS` applied to the gene trees in Figure 1d and 1e, respectively. While we have left the daughter tree branch in red for clarity, it is important to note that these trees are identical topologically, which is why `reconcILS` has labelled them in exactly the same way. The algorithm has no way to "see" the red branches, and has consequently simply picked the bottom-most node in both cases as the one to label as having undergone ILS and duplication, even though this is not precisely correct in Figure 3c. At the moment, this is a limitation of `reconcILS` (and all methods attempting to label gene trees with ILS events), but we mention other possible solutions in the Discussion.

Primate data

We used two sets of gene trees that were previously constructed in Smith et al. (2022) using IQ-TREE2 (Minh et al. 2020b), based on data from Vanderpool et al. (2020). The first set consists of 1,820 single-copy orthologs; this set was denoted as "Single-copy clusters" in Smith et al. (2022). The second set consists of 11,555 trees containing both orthologs and paralogs; this set was denoted "All paralogs" in Smith et al. (2022). We randomly resolved all polytomies. When using `reconcILS`, costs were set to $C_D = 2$, $C_L = 2$, and $C_I = 1$.

Acknowledgements

We thank Yao-ban Chan for many helpful conversations about the MLMSC model, Jessica Wu for answering questions about DLCpar, and Ben Fulton for tips on best programming practices. Reviewers of the manuscript also provided many constructive comments. This work was supported by NSF grant DEB-1936187 and DBI-2146866 to MWH.

Data availability

`reconcILS` is available at https://github.com/smishra677/reconcILS. `dupcoal` is available at https://github.com/meganlsmith/dupcoal.

References

Avise, J. C., and T. J. Robinson. 2008. Hemiplasy: A new term in the lexicon of phylogenetics. Systematic Biology **57**:503-507.

Bansal, M. S., E. J. Alm, and M. Kellis. 2012. Efficient algorithms for the reconciliation problem with gene duplication, horizontal transfer and loss. Bioinformatics **28**:i283-i291.

Baum, D. A. 2007. Concordance trees, concordance factors, and the exploration of reticulate genealogy. Taxon **56**:417-426.

Boussau, B., and C. Scornavacca. 2020. Reconciling gene trees with species trees. Pp. 3.2:1-3.2:23 *in* C. Scornavacca, F. Delsuc, and N. Galtier, eds. Phylogenetics in the Genomic Era. No commercial publisher | Authors open access book.

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

Chaudhary, R., J. G. Burleigh, and O. Eulenstein. 2012. Efficient error correction algorithms for gene tree reconciliation based on duplication, duplication and loss, and deep coalescence. BMC Bioinformatics **13**:S11.

Chen, K., D. Durand, and M. Farach-Colton. 2000. NOTUNG: A program for dating gene duplications and optimizing gene family trees. Journal of Computational Biology **7**:429-447.

Christensen, S., E. K. Molloy, P. Vachaspati, A. Yammanuru, and T. Warnow. 2020. Non-parametric correction of estimated gene trees using TRACTION. Algorithms for Molecular Biology **15**:1.

Du, H., Y. S. Ong, M. Knittel, R. Mawhorter, N. Liu, G. Gross, R. Tojo, R. Libeskind-Hadas, and Y.-C. Wu. 2021. Multiple optimal reconciliations under the duplication-loss-coalescence model. IEEE/ACM Transactions on Computational Biology and Bioinformatics **18**:2144-2156.

Duchemin, W., G. Gence, A.-M. Arigon Chifolleau et al. 2018. RecPhyloXML: a format for reconciled gene trees. Bioinformatics **34**:3646-3652.

Durand, D., B. V. Halldórsson, and B. Vernot. 2006. A hybrid micro-macroevolutionary approach to gene tree reconstruction. Journal of Computational Biology **13**:320-335.

El-Mabrouk, N., and A. Ouangraoua. 2017. A general framework for gene tree correction based on duplication-loss reconciliation. Pp. 8:1-8:14. WABI.

Emms, D. M., and S. Kelly. 2019. OrthoFinder: phylogenetic orthology inference for comparative genomics. Genome Biology **20**:238.

Fitch, W. M. 1970. Distinguishing homologous from analogous proteins. Systematic Zoology **19**:99-113.

Gillespie, J. H., and C. H. Langley. 1979. Are evolutionary rates really variable? Journal of Molecular Evolution **13**:27-34.

Górecki P, Eulenstein O. 2011. A linear time algorithm for error-corrected reconciliation of unrooted gene trees. ISBRA 2011: Springer Berlin Heidelberg. p. 148-159.

Goodman, M., J. Czelusniak, G. W. Moore, A. E. Romero-Herrera, and G. Matsuda. 1979. Fitting the gene lineage into its species lineage, a parsimony strategy illustrated by cladograms constructed from globin sequences. Systematic Zoology **28**:132-163.

Guerrero, R. F., and M. W. Hahn. 2018. Quantifying the risk of hemiplasy in phylogenetic inference. Proceedings of the National Academy of Sciences **115**:12787-12792.

Guigo, R., I. Muchnik, and T. F. Smith. 1996. Reconstruction of ancient molecular phylogeny. Molecular Phylogenetics and Evolution **6**:189-213.

Hahn, M. W. 2007. Bias in phylogenetic tree reconciliation methods: implications for vertebrate genome evolution. Genome Biology **8**:R141.

Hahn, M. W., and L. Nakhleh. 2016. Irrational exuberance for resolved species trees. Evolution **70**:7-17.

Hallett, M., J. Lagergren, and A. Tofigh. 2004. Simultaneous identification of duplications and lateral transfers. Pp. 347-356. Proceedings of the eighth annual international conference on Research in computational molecular biology (RECOMB '04).

Hallett, M. T., and J. Lagergren. 2001. Efficient algorithms for lateral gene transfer problems. Pp. 149-156. Proceedings of the fifth annual international conference on Computational biology (RECOMB '01).

Harris, C. R., K. J. Millman, S. J. Van Der Walt et al. 2020. Array programming with NumPy. Nature **585**:357-362.

Hibbins, M. S., L. C. Breithaupt, and M. W. Hahn. 2023. Phylogenomic comparative methods: Accurate evolutionary inferences in the presence of gene tree discordance. Proceedings of the National Academy of Sciences **120**:e2220389120.

Hudson, R. R. 1983. Testing the constant-rate neutral allele model with protein sequence data. Evolution **37**:203-217.

Huerta-Cepas, J., H. Dopazo, J. Dopazo, and T. Gabaldón. 2007. The human phylome. Genome Biology **8**:R109.

Huerta-Cepas, J., F. Serra, and P. Bork. 2016. ETE 3: Reconstruction, analysis, and visualization of phylogenomic data. Molecular Biology and Evolution **33**:1635-1638.

Jacox, E., C. Chauve, G. J. Szöllősi, Y. Ponty, and C. Scornavacca. 2016. ecceTERA: comprehensive gene tree-species tree reconciliation using parsimony. Bioinformatics **32**:2056-2058.

Kingman, J. F. C. 1982. The coalescent. Stochastic Processes and their Applications **13**:235-248.

Lafond M, Chauve C, Dondi R, El-Mabrouk N. 2014. Polytomy refinement for the correction of dubious duplications in gene trees. Bioinformatics **30**:i519-i526.

Lanfear R, Hahn MW. 2024. The meaning and measure of concordance factors. Molecular Biology and Evolution **41**:msae214.

Li, Q., Y.-b. Chan, N. Galtier, and C. Scornavacca. 2024. The effect of copy number hemiplasy on gene family evolution. Systematic Biology **73**:355-374.

Li, Q., C. Scornavacca, N. Galtier, and Y.-B. Chan. 2021. The multilocus multispecies coalescent: A flexible new model of gene family evolution. Systematic Biology **70**:822-837.

Libeskind-Hadas, R., Y.-C. Wu, M. S. Bansal, and M. Kellis. 2014. Pareto-optimal phylogenetic tree reconciliation. Bioinformatics **30**:i87-i95.

Maddison, W. P. 1997. Gene trees in species trees. Systematic Biology **46**:523-536.

Mallo, D., L. De Oliveira Martins, and D. Posada. 2014. Unsorted homology within locus and species trees. Systematic Biology **63**:988-992.

Mallo, D., L. De Oliveira Martins, and D. Posada. 2016. *SimPhy*: Phylogenomic simulation of gene, locus, and species trees. Systematic Biology **65**:334-344.

Mawhorter, R., and R. Libeskind-Hadas. 2019. Hierarchical clustering of maximum parsimony reconciliations. BMC Bioinformatics **20**:612.

Mendes, F. K., and M. W. Hahn. 2016. Gene tree discordance causes apparent substitution rate variation. Systematic Biology **65**:711-721.

Mendes, F. K., D. Vanderpool, B. Fulton, and M. W. Hahn. 2020. CAFE 5 models variation in evolutionary rates among gene families. Bioinformatics **36**:5516-5518.

Meng, C., and L. S. Kubatko. 2009. Detecting hybrid speciation in the presence of incomplete lineage sorting using gene tree incongruence: A model. Theoretical Population Biology **75**:35-45.

Minh, B. Q., M. W. Hahn, and R. Lanfear. 2020a. New methods to calculate concordance factors for phylogenomic datasets. Molecular Biology and Evolution **37**:2727-2733.

Minh BQ, Schmidt HA, Chernomor O, Schrempf D, Woodhams MD, Von Haeseler A, Lanfear R. 2020b. IQ-TREE 2: New models and efficient methods for phylogenetic inference in the genomic era. Molecular Biology and Evolution **37**:1530-1534.

Morel, B., A. M. Kozlov, A. Stamatakis, and G. J. Szöllősi. 2020. GeneRax: A tool for species-tree-aware maximum likelihood-based gene family tree inference under gene duplication, transfer, and loss. Molecular Biology and Evolution **37**:2763-2774.

Nguyen, T. H., V. Ranwez, S. Pointet, A.-M. A. Chifolleau, J.-P. Doyon, and V. Berry. 2013. Reconciliation and local gene tree rearrangement can be of mutual profit. Algorithms for Molecular Biology **8**:12.

Page, R. D. M. 1994. Maps between trees and cladistic analysis of historical associations among genes, organisms, and areas. Systematic Biology **43**:58-77.

Page, R. D. M., and M. A. Charleston. 1997. From gene to organismal phylogeny: Reconciled trees and the gene tree/species tree problem. Molecular Phylogenetics and Evolution **7**:231-240.

Paradis, E., and K. Schliep. 2018. ape 5.0: an environment for modern phylogenetics and evolutionary analyses in R. Bioinformatics **35**:526-528.

Rasmussen, M. D., and M. Kellis. 2012. Unified modeling of gene duplication, loss, and coalescence using a locus tree. Genome Research **22**:755-765.

Richter, D. J., P. Fozouni, M. B. Eisen, and N. King. 2018. Gene family innovation, conservation and loss on the animal stem lineage. eLife **7**:e34226.

Robinson DF. 1971. Comparison of labeled trees with valency three. Journal of Combinatorial Theory, Series B **11**:105-119.

Schrider, D. R., and M. W. Hahn. 2010. Lower linkage disequilibrium at CNVs is due to both recurrent mutation and transposing duplications. Molecular Biology and Evolution **27**:103-111.

Scornavacca C, Berry V, Ranwez V. 2011. Building species trees from larger parts of phylogenomic databases. Information and Computation **209**:590-605.

Smith, M. L., and M. W. Hahn. 2022. The frequency and topology of pseudoorthologs. Systematic Biology **71**:649-659.

Smith, M. L., D. Vanderpool, and M. W. Hahn. 2022. Using all gene families vastly expands data available for phylogenomic inference. Molecular Biology and Evolution **39**:msac112.

Stolzer, M., H. Lai, M. Xu, D. Sathaye, B. Vernot, and D. Durand. 2012. Inferring duplications, losses, transfers and incomplete lineage sorting with nonbinary species trees. Bioinformatics **28**:i409-i415.

Stull, G. W., X.-J. Qu, C. Parins-Fukuchi et al. 2021. Gene duplications and phylogenomic conflict underlie major pulses of phenotypic evolution in gymnosperms. Nature Plants **7**:1015-1025.

Sukumaran, J., and M. T. Holder. 2010. DendroPy: a Python library for phylogenetic computing. Bioinformatics **26**:1569-1571.

Szöllősi, G. J., W. Rosikiewicz, B. Boussau, E. Tannier, and V. Daubin. 2013. Efficient exploration of the space of reconciled gene trees. Systematic Biology **62**:901-912.

Tajima, F. 1983. Evolutionary relationships of DNA sequences in finite populations. Genetics **105**:437-460.

Than, C. V., and N. A. Rosenberg. 2013. Mathematical properties of the deep coalescence cost. IEEE/ACM Transactions on Computational Biology and Bioinformatics **10**:61-72.

Thomas, G. W. C., S. H. Ather, and M. W. Hahn. 2017. Gene-tree reconciliation with MUL-trees to resolve polyploidy events. Systematic Biology **66**:1007-1018.

Vanderpool, D., B. Q. Minh, R. Lanfear et al. 2020. Primate phylogenomics uncovers multiple rapid radiations and ancient interspecific introgression. PLoS Biology **18**:e3000954.

Vernot, B., M. Stolzer, A. Goldman, and D. Durand. 2008. Reconciliation with non-binary species trees. Journal of Computational Biology **15**:981-1006.

Waskom, M. 2021. Seaborn: Statistical data visualization. Journal of Open Source Software **6**:3021.

Wu, Y.-C., M. D. Rasmussen, M. S. Bansal, and M. Kellis. 2014. Most parsimonious reconciliation in the presence of gene duplication, loss, and deep coalescence using labeled coalescent trees. Genome Research **24**:475-486.

Yu, Y., C. Than, J. H. Degnan, and L. Nakhleh. 2011. Coalescent histories on phylogenetic networks and detection of hybridization despite incomplete lineage sorting. Systematic Biology **60**:138-149.

Zhang, C., C. Scornavacca, E. K. Molloy, and S. Mirarab. 2020. ASTRAL-Pro: Quartet-based species-tree inference despite paralogy. Molecular Biology and Evolution **37**:3292-3307.

Zmasek, C. M., and S. R. Eddy. 2001. A simple algorithm to infer gene duplication and speciation events on a gene tree. Bioinformatics **17**:821-828.