Comparison of Annotating Duplication, Tree Mapping, and Copying as Methods to Compare Gene Trees with Species Trees

Oliver Eulenstein, Boris Mirkin, and Martin Vingron

Abstract. This paper reviews the authors' work on the idea of employing the mechanism of gene duplication in explaining the differences between a gene and species trees. Three existing approaches are presented as based on: (a) tree-mapping, (b) annotating duplication, and (c) copying duplication modeling. Correspondences between (a) and (b), and (b) and (c) are mathematically explored. It is proven, in particular, that approaches (b) and (c) lead to equivalent duplication histories. Moreover, all the three approaches equivalently count the numbers of duplications and losses needed to explain all the differences between trees.

1. Introduction

It is today generally accepted that any two forms of life on earth have evolved from a common ancestor (J.M. Smith [**18**], Li and Graur [**13**]). One aim of evolutionary biology is the reconstruction of the evolutionary history of current species. Based on the assumption of common ancestors this history can be depicted as a tree, generally called a phylogenetic tree. Its nodes correspond to ancestral species and its edges are lines of descent.

The identities of species and the states of their various characters changed along the branches of the same evolutionary tree. Studying the history of a character is the main source of information for the reconstruction of evolutionary relationships among species. However, it is of prime importance to study characters that are based on evolutionarily comparable structures, called homologous. A fly and a bird both have wings and yet the bird is not more closely related to insects than to other vertebrates. The wings of birds and flies are believed to be incomparable structures. They seem unlikely to have evolved from the same structure in their most recent common ancestor. Estimating history from comparisons of structures non-comparable in this sense may lead to errors. With the rise of molecular biology the DNA sequences of genes have become available. These sequences provide DNA base pairs that can be treated as characters from which to estimate phylogenetic trees (see e.g. Fitch and Margoliash [**9**], Nei [**15**], Felsenstein [**8**]). However, determining which genes actually are comparable may be problematic. There exist large families of related genes that have evolved through the processes of gene duplication. Once a gene has been duplicated, each copy can evolve distinct variations. Distinct copies of the same gene are called paralogous. Subsequently a single species may contain none, one, or several copies of what was a single gene in an ancestor. In order to derive a tree that correctly reflects the evolution of species of this particular family, one would like to know which copies of the gene are the comparable ones. Good estimates are generally only possible after careful study of the entire family. The tree derived from a selection of genes from a gene family and the tree describing the evolution of species will frequently have different topologies. We will call a tree describing the evolution of a set of genes the *gene tree* and the tree describing the evolution of species the *species tree.*

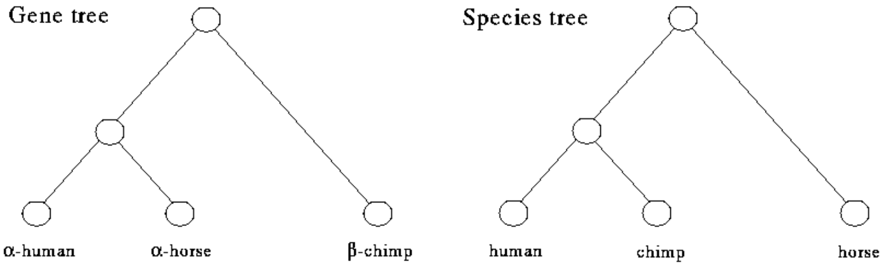

FIGURE 1. Incongruent gene and species trees.

Because they describe the evolution of different entities, a *gene tree* and *species tree* may be different in spite of the fact that their evolutionary representation is correct. Consider for example the *gene tree* and *species tree* for the hemoglobin family in Figure 1 (see, e.g. Li and Graur [**13**]). Goodman et al. [**10**] reasoned that this incongruence might result from mistaking paralogous genes for orthologous. The gene family of hemoglobin genes in vertebrates contains, among others, two types of genes: α-hemoglobin and β-hemoglobin. Both types evolved from an ancestral hemoglobin that existed prior to the vertebrates. This ancestral gene then was duplicated and the two new paralogous genes (copies) gave rise to vertebrate α- and β-hemoglobins, respectively. A researcher studying the α-hemoglobins from man, chimpanzee, and horse will find that man and chimpanzee have a common ancestor which in turn has a common ancestor with the horse. If the researcher studied β-hemoglobins from the same set of species he would find the same result. Were this family not as well-studied as it is today, the researcher might, however, have chosen a β-gene from chimp and α-genes from man and horse as the basis of his analysis. Consequently he would have found that man and horse group together with chimp of older evolutionary origin. This is believed to be correct for this particular selection of genes, but incorrect for the evolution of these species.

Figure 2 reflects the complete *gene tree* for the α-genes and the β-genes of man, chimp and horse. Note that the *gene tree* of Figure 1 is a subtree of the complete *gene tree* and the complete *gene tree* is a duplication of the *species tree* in Figure 1. Assume we would be aware of the duplication events in the *species tree.* Then we would be able to outline the topology of the *species tree* and to embed our *gene tree* into it. We would obtain a reconciled gene tree which represents in our case the complete *gene tree.*

Thus, possible discrepancies between a *gene tree* and a *species tree* can be explained by postulating duplication events that gave rise to different copies of a gene. From

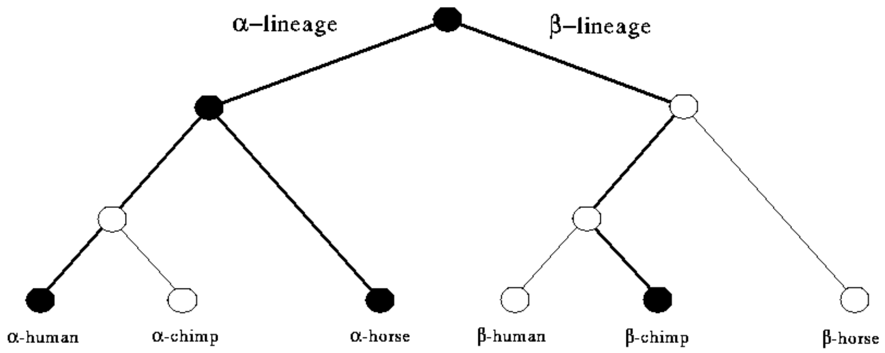

FIGURE 2. Complete gene tree.

the theoretical point of view there is an unlimited number of sets of duplication events that result in a possible reconciled tree. Goodman et al. [**10**] outlined a strategy by postulating the duplications required for a reconciled tree. More recently this method has been elaborated in Page [**17**], Mirkin et al. [**14**] and Guigó et al. [**12**].

To introduce the basic logic we need to introduce gene and species trees in more detail. The basic assumption will be that exactly one gene from each contemporary species is present in the gene tree. Such an over-simplification is made for the sake of simplicity and, moreover, possibility of the resulting mathematical analysis. Besides, it is more a requirement to the representation of data rather than a restriction to the evolutionary processes covered. In the example considered, one should have two gene trees, one for hemoglobin α-lineage, the other for β-lineage, to compare each with a species tree. We may say that the term "gene" is used here in the meaning of molecular biology, as any distinct sequence variant, not a member of a prespecified paralogous family. In general, different genes should be treated within different gene families to yield gene trees satisfying the basic assumption (as it was unintentionally done by Guigó, Muchnik and Smith [**12**]). On the other hand, the assumption can be relaxed, which is however a subject for separate treatment.

Based on the assumption above, we may use the convention to denote a contemporary species and a contemporary gene from that species by the same symbol, which much simplifies the subsequent mathematical analysis. We will use integers for this purpose. Note that an ancestral gene is uniquely specified by the set of contemporary genes (leaves of the gene tree) descending from it. Likewise, an ancient species is uniquely specified by the contemporary species descending from it.

Duplication events are postulated based on a function, called the *tree mapping function*. This function maps each contemporary or ancestral gene of the gene tree onto a species in the species tree. This species, too, may either be contemporary, i.e. correspond to a leaf of the species tree, or ancestral, i.e. correspond to an inner node of the species tree. The tree mapping function maps a gene onto the most recent species that is presumed to have contained that gene. How to find out whether a species possessed a certain gene is easily seen from an example. Assume that an ancestral gene has as contemporary descendant genes, 1, 2, and 3, and so call the ancestral gene $\{1,2,3\}$. Any species having 1, 2, and 3 among descendant species possessed a gene ancestral to genes 1, 2, and 3 and thus ancestral also to gene $\{1,2,3\}$. The most recent of these ancient species (called sometimes the *least common ancestor*) is the mapping image of $\{1,2,3\}$.

The tree mapping needs not be injective: it may map a parent gene, say a node a of the gene tree, onto the same species as one of a's immediate descendants called children, ca (we denote a fixed but arbitrarily chosen child of a node a with ca). This means that the most recent species which possessed gene a is also the most recent species in which one finds its child gene ca. In this case the bifurcation of a in the gene tree is not consistent with the bifurcation of its image in the species tree. The bifurcation of a takes place in the image and makes gene and species tree inconsistent with each other. In our model the bifurcation of a suggests that the species that is its mapping image possessed two copies of gene a, say $a+$ and $a-$. The existence of two copies is postulated to be due to a prior duplication of a predecessor gene, at what we call a *duplication node* in the species tree. Since we do not have knowledge of a species that possessed ca and not a we can identify ca with one copy, say $a+$. The sibling, denoted as $\bar{c}a$, of ca is the other copy $a-$ if $\bar{c}a$ maps onto the same species node as ca. If $\bar{c}a$ maps onto a species descendant to the duplication node, it is a descendant gene of $a-$. This distinction is the basis for distinguishing between two-side and one-side duplications below.

The number of duplications and other relevant events needed to explain a *gene tree* from a given *species tree* has been used as an asymmetric distance measure between the two trees by Goodman *et al* [**10**]. Among those events, the only one visible in terms of evolutionary trees is the loss of a certain set of genes (see, e.g., Nelson and Platnick [**16**], Page [**17**]). In our example those are the β-hemoglobin genes from man and horse and the α-hemoglobin gene from chimp. The lost genes might not constitute leaves but entire subtrees. Of course, the number of lost genes will grow with the number of duplications.

To determine the placement of gene duplication along the species tree we need to further elaborate on the inconsistencies between a gene and species trees caused by duplications and losses. We discuss three approaches to this: (a) tree-mapping, (b) annotating duplication, and (c) copying duplication.

Tree-mapping (a) has been considered by Guigò, Muchnik and Smith [**12**], whose basic idea is to measure the inconsistency, or mapping cost, of any duplication hypothesis by the sum over all genes of numbers of intermediate species nodes between the mapping image of a gene node and the image of its parent. No biologically meanigful motivation is given to this cost evaluation of the mapping which is claimed by the authors to count for all the loss events related to underlying duplications. This cost measure is minimized (with a local search algorithm) in the reconstructed evolutionary tree found in Guigò, Muchnik and Smith [**12**].

Annotating duplication (b) developed in Mirkin *et al* [**14**] (see also Eulenstein and Vingron [**7**] and L. Zhang [**19**]) involves a mathematical model of the duplication history corresponding to a duplication gene node g. All the leaves in the species tree whose corresponding genes belong to cg are annotated by $+$, and to $\bar{c}g$ by -. The pattern of the sign labels then naturally ascends in the species tree to the image of g. The maximum species nodes having all their content annotated by the same sign label correspond to the loss events.

Copying duplication (c), though never developed mathematically, is expressed quite clearly in the concept of reconciled tree (see Nelson and Platnick [**16**] and Page [**17**]). In this concept, the species subtree having a duplication node at its root is doubled so that one copy of the subtree keeps one copy of the duplicate gene g while the other subtree, another copy of the gene (as shown in Figure 2). In each of the tree copies, the gene is not observed in some of the species that are claimed to be "extinct". Being a two-dimensional graphical construction, the copying duplication becomes less clear when multiple and nested duplications of genes are hypothesized to explain the differences between a species and gene tree.

The purpose of this paper is to formally define all three approaches and to describe correspondences found among them. In particular, we prove that the number of loss events accounted by each of the approaches is the same. The paper integrates the earlier work of the authors exploring interconnections between approaches (a) and (b) (Eulenstein and Vingron [**7**], Eulenstein et al. [**5**]) and (b) and (c) (Eulenstein et al. [**6**]); proofs of some statements from those papers are omitted.

In Section 2, we formally define the notions just introduced and illustrate them with an example. Comparing approaches (a) and (b) is done in Section 3. In Section 4, approaches (b) and (c) are compared. Section 5 is a short conclusion.

2. Three Approaches to Comparing Gene and Species Trees

2.1. Basic Definitions. The model for evolutionary history we deal with is a rooted binary tree with the leaf set labeled by indices from an N-element set I. The indices label biological taxa under consideration and, simultaneously, genes one-to-one corresponding to the taxa. The basic assumption "one species - one gene" is the ground for such a twofold function of the indices, which simplifies mathematical formulas and derivations. If, for instance, a subset $g \subset I$ refers to genes and $s \subset I$ refers to species, expressions $s \subset g$, $s \cap g$, and $s \cup g$ are meaningful since both g and s are index subsets. Relation $s \subset g$ can be interpreted as "the genes corresponding to species in s all belong to g" or "the set of species corresponding to genes in g includes set s". The other set-theoretic expressions are interpreted similarly.

A *rooted tree*, T, is considered as a nested set of clusters, $T \subset 2^I$, which includes the singletons (leaves) $\{i\}$, $i \in I$, and I itself (the root). This implies that the terms "node of a tree" and "cluster in a tree" are considered synonymous in this paper. By $T(t)$ we denote the subtree of T rooted at $t \in T$; that is, $T(t) = \{t' \in T : t' \subseteq t\}$. An important property of a nested tree is that any two of its nodes are either nonoverlapping or nested (Estabrook and McMorris [**3**]).

A node $t \in T$ is internal if it is neither a singleton nor the root. The two children of an internal node $t \in T$ will be denoted by ct and $\bar{c}t$ (assigning c and $\bar{c}$ arbitrarily). The parent of a node $t \in T$ will be denoted by pt. For every subset $J \subset I$, the least common ancestor of J in T is minimum of the nodes $t \in T$ such that $J \subseteq t$. The least common ancestor of J will be denoted by $a_T(J)$.

A species tree will be denoted by S, with its clusters (nodes) $s \in S$, and a gene tree by G, with its clusters (nodes) $g \in G$. As explained above, both types of trees are subsets of 2^I.

Fig. 3 shows a species tree and gene tree. The letter c in the gene tree marks gene cluster $\{1, 2, 3\}$ and the letter F, in the species tree, marks species cluster $\{4, 5, 7\}$. Clusters G and 7 are children of F. Subtree $S(F)$ contains 4, 5, 7,

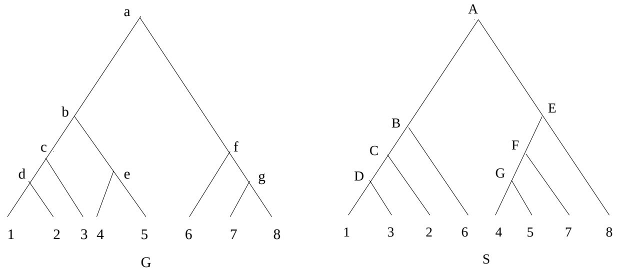

FIGURE 3. A gene tree, G, and a species tree, S.

$G = \{4, 5\}$ and F. For a subset $J = \{1, 2, 3, 4\}$ its least common ancestor is b in G and A in S.

2.2. Tree Mapping Modeling. The *tree mapping* function is just the least common ancestor mapping a_S in tree S applied to gene clusters $g \in G$: $a_S(g)$ is the minimum species cluster containing all genes from g.

A pair $(g, s) \in G \times S$ is called a *one-side duplication* if either $a_S(g) = a_S(cg)$ or $a_S(g) = a_S(\bar{c}g)$ (but not both). A pair $(g, s) \in G \times S$ is called a *two-side duplication* if both of the equations hold. A pair (g, s) is called a *duplication* if it is either a one-side or two-side duplication. The number of one-side duplications will be denoted $O(G, S)$. Sometimes, when there is no ambiguity, we refer to either s or g in a duplication pair (g, s) as a duplication, too.

Let us say that s is between s' and s'', or $s \in [s', s'']$ if $s' \subset s \subset s''$ $(s, s', s'' \in S)$. A node $s \in S$ will be called a g-intermediate if it is between $a_S(g)$ and $a_S(pg)$. The set of all g-intermediate nodes will be denoted I_g. Cardinalities of these sets can be employed as (local) measures of difference between trees G and S. In particular, the sets are empty if $G = S$. Let us refer to the total number of intermediate nodes in mapping G into S as to the *cost mapping* index, $M(G, S)$:

$$M(G, S) = \sum_{g \in G} |I_g| \tag{1}$$

This g-intermediate node concept is implicitly exploited in Guigó, Muchnik and Smith [**12**] where the following cost $C(g)$ (in comparing G and S) is assigned to every node $g \in G$: $C(g)$ equals $|I_{cg}| + |I_{\bar{c}g}|$, the total number of cg- and $\bar{c}g$-intermediate nodes in S (plus 1 when g is a one-side duplication). Their cost function $C(G, S)$ is defined as the sum of all costs $C(g)$, $g \in G$. (Actually, Guigó, Muchnik, and Smith [**12**] add to that the number of all the duplications; we drop this latter term since it is irrelevant in the present context.) Evidently, $C(G, S)$ equals the total number of the intermediate nodes in S plus the number of one-side duplications, as also observed by Eulenstein and Vingron [**7**] in their streamlining of the measures:

$$C(G, S) = M(G, S) + O(G, S) \tag{2}$$

To illustrate the tree-mapping concepts, let us consider the example of gene and species trees presented on Fig. 3. The mapping is shown in Fig. 4. It is readily seen that there are two duplications, at gene nodes a (two-side) and c (one-side), needed to explain all the differences between G and S. The images of all gene

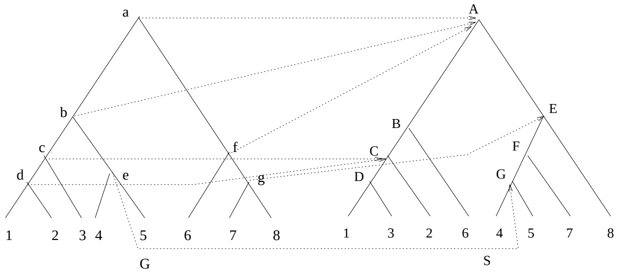

FIGURE 4. The least common ancestor mapping tree G into tree S from Fig. 3.

child-parent pairs are listed in Table 1. We can see that there are 7 intermediate nodes in total, which makes the cost function equal to 8.

TABLE 1. Mapping of the gene node-parent pairs into the species nodes.

G-node	Parent	Node S-image	Parent S-image	Number of Intermediates
1	d	1	C	1
2	d	2	C	0
3	c	3	C	1
4	e	4	G	0
5	e	5	G	0
6	f	6	A	1
7	g	7	E	1
8	g	8	E	0
g	f	E	A	0
f	a	A	A	0
e	b	G	A	2
d	c	C	C	0
c	b	C	A	1
b	a	A	A	0

2.3. Annotating Duplication Modeling. Any duplication node marks a difference in patterns of divergence between the species tree and the gene tree. The difference is readily seen in comparing the children of a duplication pair (g, s). It is characterized by the following "inconsistency" condition (Mirkin et al. [**14**]):

$$g \subseteq s \text{ and } cg \cap cs \neq \emptyset \text{ and } cg \cap \bar{c}s \neq \emptyset \tag{3}$$

for an appropriate denotation of children of g. Such an inconsistency requires a duplication of the ancestral gene g postulated in the ancestral species s so that all species containing genes descendant from gene cg carry one copy of the duplicate gene and all species containing genes descendant from gene $\bar{c}g$ contain the other copy (Duplication/Speciation Principle in Mirkin, Muchnik and Smith [**14**]).

A mapping $\delta : S(s) \rightarrow \{+/-, +, -, \emptyset\}$ is referred to as an *annotating duplication* if and only if for any $s' \in S(s)$,

$\delta(s') = \emptyset$ if both of the two following conditions hold: (a) $s' \cap cg = \emptyset$ and (b) $s' \cap \bar{c}g = \emptyset$;

$\delta(s')$ is $+$ or $-$ if either of (a) or (b) holds, and

$\delta(s') = +/-$ if none of (a) and (b) holds.

The mapping δ indicates whether one, the other, both or neither duplicate gene is present at each node of the species tree (see Fig. 5).

The evolutionary history of the duplication generated by an inconsistent pair (g, s), where $s = s_S(g)$, can be considered in the framework of the basic partition $\{cg, \bar{c}g\}$ of g put into the context of species subtree $S(a_S(g))$. The elements of each of the classes, cg or $\bar{c}g$, are interpreted as the currently living species bearing only one of the copies of the duplicated gene g. Due to the definition of annotating δ above any node $s \in S(a_S(g))$ can be qualified as *one-copy* ($+$ or $-$ only) if $g \cap s$ is included in one of the classes only or *mixed* (and labeled by $+/-$ mark) if s overlaps both of them. A particular evolutionary meaning is assigned to maximal one-copy nodes $s \in S(a_S(g))$: each of them corresponds to the event of *loss* of a duplicate copy.

Explicitly, the concept of loss can be formulated as follows. A node $s \in S$ will be referred to as a g-loss if and only if any of the two equivalent statements is true:

(i) $s \cap g \subseteq cg$ or $s \cap g \subseteq \bar{c}g$, but this is not true for its parent, ps;

(ii) $s \cap cg = \emptyset$ or $s \cap \bar{c}g = \emptyset$, but this is not true for its parent, ps.

In the case when both of the inclusions in (i), or equations in (ii), are satisfied, that is, if $s \cap g = \emptyset$, there is no information to assign a copy of the duplication to s; this corresponds to what is called *gap* in Mirkin, Muchnik and Smith [**14**]. It should be noted however that, in the latter paper, the concept of gap is considered somewhat ambigously so that the current meaning corresponds to that based on the Duplication/speciation principle (p. 500) while that introduced on p. 498 may refer not only to the losses, but some smaller nodes, too.

A g-loss s that is not gapped (so that $s \cap g \neq \emptyset$) will be referred to as a charged loss.

As the gene and species trees presented in Fig. 4 give two duplication pairs, (a, A) and (c, C), corresponding annotating duplications are shown in Fig. 5. The losses are shown with boxes in the copies of the species tree. Species from different children of a duplicate genes are labeled by different sign labels, $+$ or $-$, according to annotating duplication. We can see that there are 5 losses for (a, A) and 3 losses for (c, C), which gives 8 losses in total, which was the same counted by the cost function above.

Guigó, Muchnik and Smith [**12**] defined that $C(G, S)$ in (2) was always exactly the total number of losses in duplication-based comparing G and S, which was conjectured by Mirkin, Muchnik and Smith [**14**] with regard to the losses defined in terms of the latter's concept of (annotating) duplication. The conjecture has been differently proved by Eulenstein and Vingron [**7**], Zhang [**19**], and Eulenstein, Mirkin and Vingron [**5**] (see Section 3 below).

2.4. Copying/Reconciling Duplication Modeling. In the previous account, the duplications have been considered as virtual ones: the species evolutionary tree was not affected, but just annotated with particular gene histories. However, one might also think of a duplication $s \in S$ as of really occurred in the evolution, as that presented in Fig.2, and thus requiring corresponding change of the species tree. Such a change assumes that the subtree $S(s)$ is doubled in S so that one copy of the subtree keeps one copy of the duplicate gene g while the other subtree, another copy of the gene. Such a transformation of S into a "reconciled"

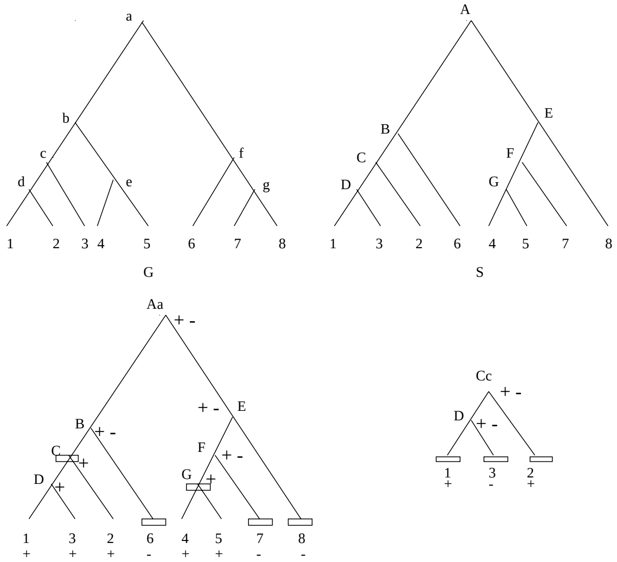

FIGURE 5. The losses for two duplications, Aa and Cc, presented by boxes in sign labeled copies of species subtrees.

tree has been developed by natural historians (see Page [**17**] for references) to reconcile the information in tree S with that in tree G.

The definition of reconciled tree presented by Page in [**17**], p. 63-64, is based on the concept of multiset tree containing any species node as many times as needed. Evidently, care must be taken to label the duplicates accurately, which has not always been done in [**17**] and earlier works.

To give an appropriate mathematical definition to the concept of copying duplication, suppose a duplication of gene G occurred during the evolution of $s \in S$, just before the ancestral species s appeared. The duplication is conceived as two copies of subtree $S(s)$, each labeled by the corresponding copy, c_1 or c_2, of the gene. This means that every node $s' \in S(s)$ is recoded as $s'c_1$ in one copy of $S(s)$ and $s'c_2$ in the other copy. The copies are denoted as $S(s)c_1$ and $S(s)c_2$. In each of the trees, $S(s)c_1$ and $S(s)c_2$, some of the leaves correspond to species in I and some do not. The first will be called active, the second non-active (extinct, in terminology of Page [**17**]). Let us denote the set of active species $i \in s$ in $S(s)c_k$ by $c_k(s)$ ($k = 1, 2$). In principle, $c_1(s) \cap c_2(s) \neq \emptyset$ since both of the gene copies can be present in currently living species. However, in the context of present paper, we analyze the case when only one of the diverged gene copies is present in the material under consideration. This requirement is a prerequisite to defining the concept of annotating duplication above. Therefore, we postulate here that the active species sets are not overlapping: $c_1(s) \cap c_2(s) = \emptyset$. At the same time, there is no need that every species in s is active; that is, there can be $s - (c_1(s) \cup c_2(s)) \neq \emptyset$ so that no information on the copy c_1 or c_2 in $i \in s - (c_1(s) \cup c_2(s))$ is available. Finally, let us denote by $S(s)\{c_1, c_2\}$ the labeled rooted tree consisting of two subtrees, $S(s)c_1$ and $S(s)c_2$, whose roots, sc_1 and sc_2 are children of the formal root labeled by s, not necessarily a subset of I. This time, it is the union of all species in s copied, that is, the cluster corresponding to s consists of new, relabeled leaves, ic_1 and ic_2, for all $i \in s$. The tree $S(s)\{c_1, c_2\}$ along with the active sets $c_1(s)$ and $c_2(s)$ is a graph-theoretic model of the concept of copying duplication. Note that all the nodes of $S(s)\{c_1, c_2\}$ are labeled, in contrast to the original gene and species trees that are considered leaf-labeled only.

The maximal subsets of non-active leaves in $S(s)\{c_1, c_2\}$ can be interpreted as losses of the corresponding genetic material (see Page [**17**]), which leads to the following definition. Let us consider any node $sc \in S(s)\{c_1, c_2\}$ as a cluster consisting of the leaves in the corresponding subtree. Then, a subset $t \in I$ will be referred to as a *species cluster* if there exists a node $sc \in S(s)\{c_1, c_2\}$ such that $i \in t$ if and only if $ic \in sc$. A maximal species cluster consisting of the non-active copies will be referred to as a *-loss.

In Fig. 6, a copying duplication in the species tree root, A, is present as corresponding to the duplication pair Aa. The labels of children of a in G, b and f, are exploited as the duplicate copy labels. The leaves in corresponding children b and f clusters are marked by a as active ones; the other, non-active, leaves by star; *-losses are shown by boxes.

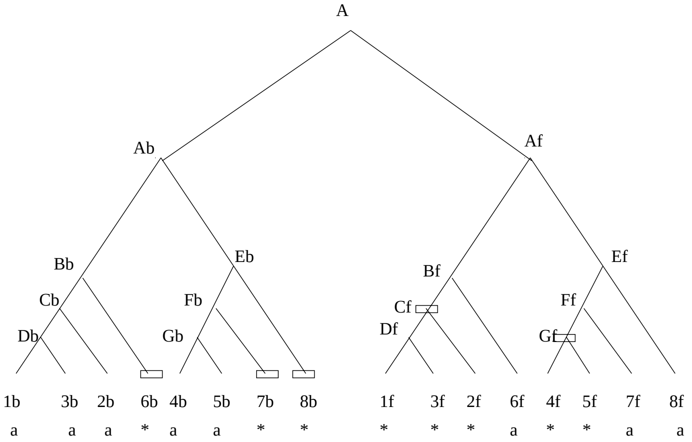

FIGURE 6. A copying duplication tree for duplication pair Aa in Fig. 5; b and f, the labels of children of a in G, are used as the duplicate labels.

In terms of the species tree S, implementation of a copying gene duplication in its node $s \in S$ is by substitution of the subtree $S(s)$ with the duplicated tree $S(s)\{c_1, c_2\}$. Other duplications in $S(s)$ can be implemented recursively in either subtree, $S(s)c_1$ or $S(s)c_2$, by adding new labels to those of previous duplications. The active species sets of these subsequent duplications must be subsets of the previous active sets. Let us formulate a "labeled" version of the reconciling algorithm from Page [**17**] implementing this construction.

Algorithm for Constructing Labeled Reconciled Tree

The algorithm recursively updates the tree R under processing by observing pairs $(g, r) \in G \times R$ in their natural partial order:

Initially, put $R = S$ and the natural partial order being just set-theoretic inclusion on pairs $(g, s) \in G \times S$.

Any time when (g, r) is a duplication, replace the subtree $R(r)$ with the copying duplication subtree, $R(r)\{cg, \bar{c}g\}$, consisting of two copies of $R(r)$ having r as their parent. To all the node labels in one of the copies is added the symbol cg; to the nodes in the other copy the symbol $\bar{c}g$ is added. In the copy labeled cg, all the leaves $i \notin cg$ are labeled non-active with "*"; in the other copy, label the leaves $i \notin \bar{c}g$ with "*".

Update the natural partial order to include those pairs (g, r) whose label suffixes contain a node above g in G.
Go to the next duplication pair (g, r); end, if there is no duplication pair left.

The final output is the labeled reconciled tree $R(G, S)$.

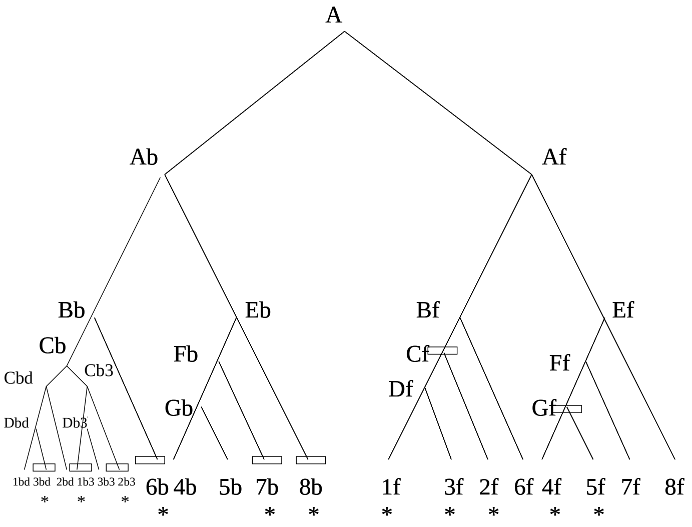

FIGURE 7. The reconciled labeled tree $R(G, S)$.

To illustrate the algorithm, let us apply it to G and S in Fig. 4. Since there are only two duplication pairs, (a, A) and (c, C), there are only two copying iterations needed. Tree R resulting from S after the first iteration is, actually, that shown in Fig. 6. The final reconciled tree is drawn in Fig. 7. We can see that the number of *-losses is again 8. Moreover, they obviously correspond to the losses in Fig. 5.

In Eulenstein, Mirkin and Vingron [**6**], a general concept of the joint duplication history tree is introduced. The questions of correctness of the algorithm above (impossibility of cycles) and correspondence between losses and *-losses are also addressed in that paper, which will be reviewed in Section 4.

3. Correspondence Between Losses and Intermediates

3.1. Some Structural Properties of the Losses. The set of all g-losses, for a $g \in G$ given, will be denoted by L_g; it has a fairly simple structure.

STATEMENT 1. *For any $g \in G$, L_g is a partition of $a_S(g)$ whose restriction to g fits strictly within the basic partition $\{cg, \bar{c}g\}$.*

Proof: Indeed, the losses are nodes in S and, thus, must not overlap each other (since they may not be nested by the requirement of their maximality). On the other hand, any leaf $i \in a_S(g)$ belongs to a g-loss: if not, $\{i\}$ is a g-loss on its own. The fact that $g \cap s$ fits within partition $\{cg, \bar{c}g\}$ for any g-loss s follows directly from the definition. Moreover, it cannot be only two g-losses because there is no inconsistency in such a case. □

It follows, from the statement, that the minimum number of g-losses, for any g, is 3. The only case when it is possible is that g is a one-side duplication and that one of the images $a_S(cg)$ and $a_S(\bar{c}g)$ which is strictly included in $a_S(g)$ has both of its children being g-losses.

Particular attention will be paid to the losses that are children of the corresponding duplication nodes.

STATEMENT 2. *A child, cs, of a duplication node $s \in S$ is a g-loss if and only if (g, s) is a one-side duplication; that is, $a_S(\bar{c}g) \subset s = a_S(g) = a_S(cg)$ (under appropriate c and $\bar{c}$ labeling), and $cs \cap a_S(\bar{c}g) = \emptyset$.*

Proof: Let us show, initially, that if cs overlaps $a_S(\bar{c}g)$ then it is not a g-loss since it overlaps both cg and $\bar{c}g$. Indeed, the fact of overlapping means that $a_S(\bar{c}g) \subseteq cs$, which implies $\bar{c}g \cap cs \neq \emptyset$. The other condition, $cg \cap cs \neq \emptyset$ follows from the fact that $s = a_S(cg)$, which means that cg must overlap both children of s.

If cs does not overlap $a_S(\bar{c}g)$, then it does not overlap $\bar{c}g$ either, which is not true for its parent s. □

It follows from statement 2 that the total number of the "child" losses (each being a child of a duplication) is equal to the number of one-side duplications, $O(G, S)$.

The property proven implies also that no child of a two-side duplication (g, s) can be a g-loss, thus both must be mixed nodes.

3.2. Principal Correspondence. Let us consider an arbitrary $s \in S$ and denote the set of duplications (or, those one-side duplications) $g \in G$ for which s is a g-loss (or, a g-loss being a child of $a_S(g)$) by D_s (or, by O_s). Obviously, $O_s \subseteq D_s$. Let us denote by P_s the set of all nodes g for which ps is a g-intermediate while s is its collateral child. The latter denotation means that $g \in P_s$ if and only if the parent of s, ps, belongs in the path connecting $a_S(g)$ and $a_S(pg)$ in S so that $ps \in [a_S(g), a_S(pg)]$ along with the other child, s', of ps (not s) also belonging to the path as presented in Fig. 8 where the trees, G and S, are drawn as just triangles. While ps must be between the images, $a_S(g)$ and $a_S(pg)$, coinciding with neither of them, the sibling of s, s', may coincide with $a_S(g)$.

We are going to prove that these sets are interelated so that

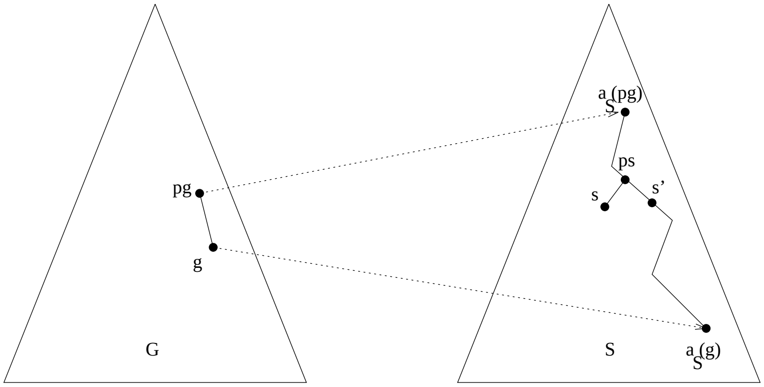

FIGURE 8. The set P_s consists of nodes $g \in G$ satisfying the configuration concerning path between $a_S(g)$ and $a_S(pg)$ in S presented.

$$|D_s| = |P_s| + |O_s| \tag{4}$$

Summing up equations in (4) by all $s \in S$, we will get the following result.

STATEMENT 3. *The total number of losses, $L(G,S)$, is equal to the total number of the mapping intermediate nodes plus the number of one-side duplications:*

$$L(G,S) = M(G,S) + O(G,S) \tag{5}$$

This statement proves that the number of losses $L(G,S)$ is exactly the cost function, $C(G,S)$, introduced by Guigó et al. in [**12**], which gives a correspondence between the tree-mapping and annotating approaches. However, this correspondence relates only to the numbers: the intermediate nodes are not, and cannot, be the losses. An interpretation of the intermediate nodes following from (4) is discussed below in section 3.4.

3.3. A Technical Proof. In Eulenstein and Vingron [**7**] equation (4) is considered in the form $|D_s| - |O_s| = |P_s|$. The equation is stated as a main result (Theorem 4.2), though in a different notation. An edge, (s, ps), is denoted by e while $L^*(e)$ denotes subset P_s and $C^*(e)$ is used to denote subset $D_s - O_s$.

Then, $L^*(e)$ is partitioned into two parts, $L^*_=(e)$, being defined as those elements of $L^*(e)$ also in $C^*(e)$, and $L^*_{\neq}(e)$, the rest. Similarly, $C^*(e)$ is partitioned into $C^*_=(e)$ and $C^*_{\neq}(e)$. The Venn diagram of these subsets in the set of all gene tree nodes is shown in Fig. 9.

After this, the essential statement becomes $|L^*_{\neq}(e)| = |C^*_{\neq}(e)|$ which is proven in Eulenstein and Vingron [**7**] by, first, observing that $L^*_{\neq}(e)$ is an antichain (none of its elements is a part of another one) while every node in $C^*_{\neq}(e)$ contains a node from $L^*_{\neq}(e)$, and, second, analyzing the properties of the tree obtained from G by cutting off all the nodes descending from nodes in $L^*_{\neq}(e)$.

A problem with the proof outlined above is that it is rather technical and gives no biologically interpreted correspondence between sets $L^*(e) = D_s - O_s$ and

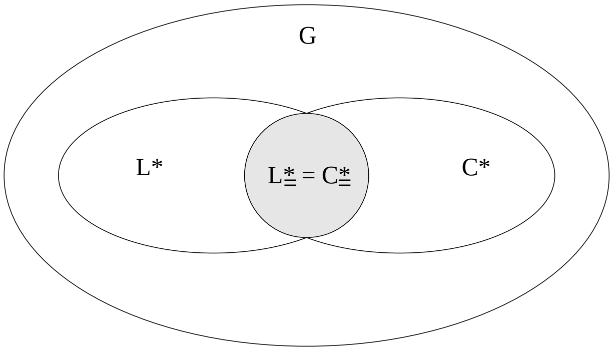

FIGURE 9. Venn diagram of sets $L^*(e)$ and $C^*(e)$.

$C^*(e) = P_s$ where $e = (s, ps)$. In the paper by Eulenstein et al. [**5**], such an interpretational proof has been provided, as follows.

3.4. An Interpretative Proof. Let us fix an interior $s \in S$ and consider all those duplication pairs $(g, a_S(g))$ that satisfy the following conditions: (i) s is not a child of $a_S(g)$, and (ii) s is a g-loss. A typical pattern is shown in Fig. 10. Now, let us fix an inclusion-maximal g from the set of duplications satisfying (i) and (ii) to deal with it in the rest of this section. It can happen that, for an s given, there is no such duplications at all. The following relates to the case when duplications satisfying (i) and (ii) exist.

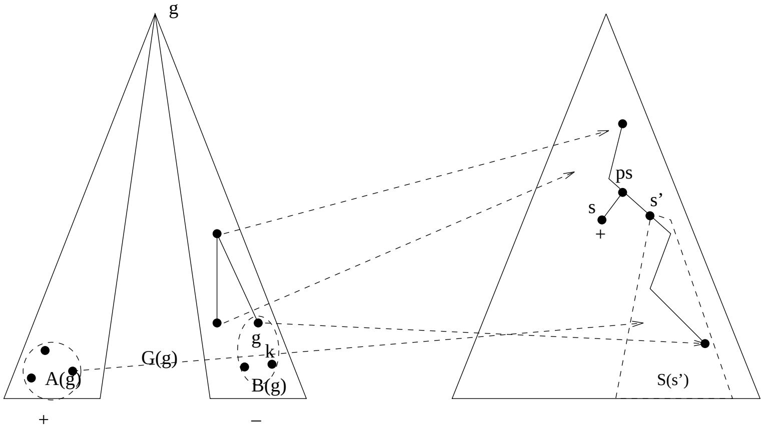

FIGURE 10. The structure of $Q(g)$ with respect to s and its sibling s'.

It can be proven that, since g under consideration is maximal, s is a charged (not gapped) g-loss so that, say $g \cap s \subseteq cg$ [**5**]; in Fig. 10 this loss is labeled by plus. Let us consider subtree $G(g)$ and define nodes $g_1, ..., g_q \in G(g)$ that are maximal parts of some nodes in $S(s')$ (s' is the collateral child of ps, see Fig. 10) while their siblings are not parts of s. Let us denote the set of these nodes by $Q(g) = \{g_1, ..., g_q\}$. Obviously, none of the nodes $g_k \in Q(g)$ overlaps s since they are parts of its sibling.

Statement 4. *For any $g_k \in Q(g)$, ps is a g_k-intermediate while s is the collateral child. Moreover, no other node $\bar{g} \in G(g)$ exists so that ps is a $\bar{g}$-intermediate and s is the collateral child.*

Proof: Indeed, $a_S(g_k) \subset ps$ because, as was defined above, $g_k \subseteq s'$ where s' is the sibling of s. On the other hand, $ps \subset a_S(pg_k)$ since pg_k is not included in $ps = s \cup s'$ (by definition of g_k) though overlaps it. Node s is the collateral child since $g_k \subseteq s'$.

Then, for ps to be an intermediate along with s the collateral child, for a pair $(\bar{g}, p\bar{g})$, $p\bar{g} \in G(g)$, it must be $\bar{g} \subseteq s'$ where s' is the other sibling of s, while $p\bar{g} \not\subseteq ps$. That implies that $\bar{g}$ must be a maximal node to be a part of a node in $S(s')$, but its sibling is not a part of s. Thus, $\bar{g}$ has been counted in $Q(g)$. □

Thus, we have proven that

$$Q(g) = P_s \cap G(g) \tag{6}$$

Now, we are going to explore whether different duplications in $(D_s - O_s) \cap G(g)$ can be assigned to different elements in $Q(g)$. It is expected that the answer is yes, which can be exploited in proving equation (4).

A one-to-one assignment of the duplications to elements of $Q(g)$ can be done with a procedure involving the following concepts. Let us partition the nodes in $Q(g)$ into two classes by charge: $A(g)$ - those charged as s; and $B(g)$ - those charged alternately. An important thing is that $B(g)$ cannot be empty since ps must be mixed. Thus, $q = |A(g)| + |B(g)|$ and $|B(g)| > 0$. It is not difficult to establish a one-to-one duplication assignment for the nodes in $B(g)$ and then to move on considering the other part of $Q(g)$, which can be exploited in a recursive manner: at each step, a non-empty subset of the remaining part of $Q(g)$ is separated to make an easy duplication assignment and then move on to the rest, until no nonassigned elements in $Q(g)$ remain. The procedure can be described as follows.

One-to-One Assigning Duplications from $(D_s - O_s) \cap G(g)$ to Elements of $Q(g)$

0. Initial setting: $Q \leftarrow Q(g)$.

1. Find $|B(g)|$ different duplications for the nodes in $B(g)$ and extract $B(g)$ from Q: $Q \leftarrow Q - B(g)$ along with $q \leftarrow |Q|$. This is done via considering the least common ancestors, $a_S(g_1, g_2)$, for every pair $g_1, g_2 \in B(g)$. All of them are right duplications as stated in the Statement 5 below. By the properties of binary trees, exactly $|B(g)| - 1$ of them are different, which, together with the top duplication g, gives $|B(g)|$ different duplications.

2. Check whether $Q = \emptyset$. If yes, end. If not, go to 3.

3. Find a $g_k \in Q$ and a corresponding duplication $\bar{g}_k$ along with the corresponding set $Q(\bar{g}_k) \subseteq Q$ such that $B(\bar{g}_k) \neq \emptyset$ in the corresponding partition of $Q(\bar{g}_k)$ by charge. [For any g_k, duplication $\bar{g}_k$ is defined as the minimal ancestor of g_k (in G) to overlap s, which is justified by Statement 6 below. By its very definition, $\bar{g}_k$ belongs to the part of $G(g)$ charged as s and, thus, is different from the duplications assigned at step 1.] Since all the $\bar{g}_k$s are partially ordered by inclusion, pick a maximal one; its $B(\bar{g}_k) \neq \emptyset$, obviously. Go to step 1 with $g = \bar{g}_k$.

The procedure obviously converges since Q is decreased at every step. Correctness of step 1 in the procedure is proved by the following.

STATEMENT 5. *For any pair of nodes in $B(g)$, their minimal common ancestor in G is a duplication, in $G(\bar{c}g)$, such that s is its non-child loss.*

Proof: Let $\bar{g} = a_G(g_1 \cup g_2)$, then, obviously, pg_1 and pg_2 are among the nodes in $G(\bar{g})$. This implies that the images of $\bar{g}$ and each of its children, $c\bar{g}$ and $\bar{c}\bar{g}$, include ps and, thus, are nested. Let $a_S(c\bar{g}) \subseteq a_S(\bar{c}\bar{g})$, then both, $c\bar{g}$ and $\bar{c}\bar{g}$, are parts of $a_S(\bar{c}\bar{g})$, which means that $a_S(\bar{g}) = a_S(\bar{c}\bar{g})$ and, thus, $\bar{g}$ is a duplication. Since both g_1 and g_2 are from $\bar{c}g$, so is $\bar{g}$, which proves that both $\bar{g} \in G(\bar{c}g)$ and $s \cap \bar{g} = \emptyset$. However ps includes both g_1 and g_2 thus overlapping both children of $\bar{g}$, which means that s is a $\bar{g}$-loss (gapped). The fact that s is not a child of $a_S(\bar{g})$ follows from inclusion $ps \subset a_S(\bar{g})$. □

Correctness of step 3 in the procedure is proved by the following.

STATEMENT 6. *For any $g_k \in A(g)$, its minimal ancestor $\bar{g}_k$ overlapping s is a duplication, in $G(cg)$, for which s is a non-child loss.*

Proof: The fact that such a $\bar{g}_k \in G(cg)$ exists follows from $g_k \in G(cg)$ and the assumption that $s \cap cg \neq \emptyset$. There are two cases possible, $pg_k = \bar{g}_k$ and $pg_k \subset \bar{g}_k$, which will be considered in turn.

First, let $pg_k = \bar{g}_k$ so that $pg_k \cap s \neq \emptyset$ and, thus, the other sibling, g'_k, of g_k overlaps s, too, since g_k and s are not overlapping. Let us show that $a_S(pg_k) = a_S(g'_k)$, that is, $\bar{g}_k = pg_k$ is a one-side duplication. Indeed, if $a_S(g'_k) \subset a_S(pg_k)$, then $a_S(g_k) \cap a_S(g'_k) = \emptyset$ and thus g'_k may not overlap anything in the path between $a_S(g_k)$ and $a_S(pg_k)$, ps included, which is not true. The fact that s is a non-child pg_k-loss follows from that ps overlaps both children of $a_S(pg_k)$ but ps does not coincide with $a_S(pg_k)$.

Second, let $pg_k \subset \bar{g}_k$ so that pg_k is a part of a child of $\bar{g}_k$, say, $pg_k \subseteq c\bar{g}_k$. Then $g_k \subset c\bar{g}_k$ so that $s \cap \bar{c}\bar{g}_k \neq \emptyset$ since, otherwise, $s \cap c\bar{g}_k \neq \emptyset$ which contradicts the minimality of $\bar{g}_k$. Thus, s and $a_S(\bar{c}\bar{g}_k)$ are overlapping and so do $a_S(\bar{c}\bar{g}_k)$ and $a_S(c\bar{g}_k)$ because $s \subset ps \subset a_S(pg_k) \subseteq a_S(c\bar{g}_k)$. This means that the latter two sets are nested so that, for instance, $a_S(\bar{c}\bar{g}_k) \subseteq a_S(c\bar{g}_k)$. Thus, both $\bar{c}\bar{g}_k$ and $c\bar{g}_k$ are included in $a_S(c\bar{g}_k)$, which implies $\bar{g}_k = \bar{c}\bar{g}_k \cup c\bar{g}_k$ is included in $a_S(c\bar{g}_k)$ and, therefore, $a_S(\bar{g}_k) = a_S(c\bar{g}_k)$, that is, $\bar{g}_k$ is a duplication indeed. Node s is $\bar{g}_k$ one-copy since $s \cap c\bar{g}_k = \emptyset$. However, ps includes g_k and thus overlaps $c\bar{g}_k$, which proves that s is a $\bar{g}_k$-loss (not being a child of $\bar{g}_k$ since $ps \subset a_S(\bar{g}_k)$). □

STATEMENT 7. *The cardinalities of $P_s \cap G(g)$ and $(D_s - O_s) \cap G(g)$ coincide:*

$$|P_s \cap G(g)| = |(D_s - O_s) \cap G(g)| \tag{7}$$

Proof: The fact that the left part is not greater then the right follows from correctness of the assignment procedure because equation (6) holds. On the other hand, $g' \in (D_s - O_s) \cap G(g)$ implies that s is a g'-loss and its parent ps is between the images of g' and a child, say cg', $ps \in [a_S(cg'), a_S(g')]$, which means that $cg' \in P_s \cap G(g)$. This proves that the right part in (7) is not greater than the left. □

STATEMENT 8. *For any node $s \in S$,*

$$|P_s| = |D_s| - |O_s| \tag{8}$$

Proof: Since maximal duplications g are nonoverlapping, so are corresponding subtrees, $G(g)$. Thus, summing up all the equations (7) gives (8). □

This completes the proof of statement 3 since (8) is equivalent to (5).

4. Equivalence Between Annotating and Copying Duplications

4.1. The Structure of Duplication Nodes in G.

Let us consider the set of all duplication nodes in S. These nodes along with edges representing set theoretic inclusion as in a Hasse diagram (for any node, only the nodes being its "immediate" subsets are its children) form a forest. Since there is no overlap between the species in different connected components of the forest, each of the components can be considered independently. Let S^* be such a component and G^* be its corresponding counterpart in G; that is, $g \in G^*$ if and only if (g, s) is a duplication at some $s \in S^*$.

STATEMENT 9. *The set G^* is a connected graph (by set-theoretic inclusion), and thus a rooted tree.*

Proof: Let (g_1, s_1) and (g_2, s_2) be duplications such that s_1 and s_2 are nodes in S^* connected by a path so that one of them is a part of the other, for instance, $s_1 \subseteq s_2$. If $g_1 \subseteq g_2$, then g_1 and g_2 are obviously connected in G^*. If $g_1 \cap g_2 = \emptyset$, then let us consider their least common ancestor in G, g, and prove that $g \in G^*$ which implies g_1 and g_2 are connected in this case, too. If both g_1, g_2 were parts of cg (or $\bar{c}g$), then cg (or $\bar{c}g$), not g, would have been the least common ancestor, which shows that $g_1 \subseteq cg$ and $g_2 \subseteq \bar{c}g$. So $g_1 \subseteq a_S(cg)$ and $g_2 \subseteq a_S(\bar{c}g)$, which implies $s_1 \subseteq a_S(cg)$ and $s_2 \subseteq a_S(\bar{c}g)$. Thus, $a_S(cg) \cap a_S(\bar{c}g) \neq \emptyset$, i.e. one of the sets is a part of the other, say $a_S(cg) \subseteq a_S(\bar{c}g)$. Thus $a_S(g) = a_S(\bar{c}g)$, and so $(g, a_S(g))$ is a duplication with $a_S(g)$ obviously belonging to S^*.

It remains to prove now that g_1 and g_2 are also connected for nonoverlapping $s_1, s_2 \in S^*$. The immediate predecessor, s, of s_1 and s_2 in S^* corresponds to a vertex $g \in G^*$ so that pairs $\{g_1, g\}$ and $\{g_2, g\}$ are connected in G^* which proves that g_1 and g_2 are connected. □

In the remainder, we restrict ourselves to the case when S^* is a connected component, because the results can be easily extended to the general case when S^* is a forest.

The set of the nodes of tree G^* can be partitioned into classes of nodes mapped into the same image $s \in S^*$. Obviously, these classes are "convex" parts of G^*: if g_1 and g_2 belong to such a class and there exists a $g_3 \in G^*$ such that $g_1 \subset g_3 \subset g_2$, then g_3 belongs to the same class. Indeed, $a_S(g_1) = a_S(g_2) = s$ implies $a_S(g) = s$ for any gene tree node between g_1 and g_2. This property allows us to partition the tree G^* into rooted subtrees (within the mapping classes), $G^*(g, s)$, where s is the common image of all the nodes in $G^*(g, s)$ and g is its root. The subtrees $G^*(g, s)$

are supposed to be maximal, that is, any leaf of a $G^*(g,s)$ has its children in G^* mapped into node(s) different from s. This implies that any leaf of a $G^*(g,s)$ has at least one of its G-children out of G^*. Indeed, by the very definition of a duplication node, one or both of its children must have the same mapping image and, thus, belong to $G^*(g,s)$ if this one or both belong to G^*.

Let us denote by G^{**} the set-inclusion tree obtained from G^* by adding to it all G-children of its nodes (compare G^* with G^{**} in Fig. 11). Each of the subtrees, $G^*(g,s)$, of G^* also will be extended in G^{**} by adding all G-children of its nodes. The extended $G^*(g,s)$ will be denoted by $G^{**}(g,s)$. Obviously, any $G^{**}(g,s)$ is a subtree in G^{**}. According to this construction, among the leaves, $g_1,...,g_m$, of a subtree $G^{**}(g,s)$ there may occur both nonduplication and duplication nodes. Any leaf, g_k, which is a duplication node, is the root of another subtree $G^{**}(g_k,s_k)$. Any leaf that is not a duplication node is either a leaf of G^{**} or the parent of another subtree $G^{**}(g',s')$. Obviously, the leaves, $g_1,...,g_m$, of $G^{**}(g,s)$ considered as clusters in I form a partition of the cluster g. The number of the leaves, m, is the number of duplication nodes in $G^{**}(g,s)$ plus 1.

In Fig. 11, the two subtrees $G^{**}(g,s)$ in G^{**} are separated by the dashed boxes.

4.2. Plait Frame and Plait Tree. Let us define a concept of *plait frame*, $P(G^*,S^*)$, which is a labeled rooted binary tree following, in general, the pattern of tree G^{**}. However, insertions from S are made between different subtrees $G^{**}(g,s)$.

To define $P(G^*,S^*)$, take G^{**} and consider all situations when a subtree, $G^{**}(g,s)$, is incident (from below) to another subtree, $G^{**}(g',s')$; that is, $s \subset s'$, and g is either a leaf of $G^{**}(g',s')$ or a child of a leaf, $\bar{g}$, of $G^{**}(g',s')$. In the former case, the parent of g in G^{**} is a node $g'' \in G^*(g',s')$. In the latter, the grandparent of g (parent of $\bar{g}$) is a node $g'' \in G^*(g',s')$. Let us denote by $S(s,s')$ a subtree of S consisting of the path between s and s' in S (s' included, s excluded) along with the collateral children of the vertices in the path added. This subtree must be inserted between g and the g'' just defined. When a leaf, $\bar{g}$, of $G^{**}(g',s')$ is g's parent, the edge $(g,\bar{g})$ is substituted by the subtree $S(s,s')$ so that the parent of s in $S(s,s')$ becomes the parent of g. When there is no node between g and g'' (the former case), a node $\bar{g}$ is inserted in the edge (g,g') artificially just to be substituted by the subtree $S(s,s')$ as above. After all insertions are made, the structure of the plait tree, $P(G^*,S^*)$, is defined. It remains to label all nodes in $P(G^*,S^*)$.

In Fig. 11 node b is a leaf parent node (the latter case) replaced by the path AB from S along with collateral children, E of A and 6 of B, in $P(G^*,S^*)$.

The nodes of $P(G^*,S^*)$ are labeled by words of form sw where $s \in S^*$ and suffix $w \in W$; W stands for the set of finite words over alphabet G^{**} (that is, its letters are nodes of G^{**}) containing not more than one occurrence of each of the letters; moreover, the length of $w \in W$ is not greater than the depth of tree G^*. The labeling is defined differently for the two types of nodes in $P(G^*,S^*)$: those inherited from G^{**} and those inserted from S. There are thus two rules:

(a) Each node $g' \in G^{**}(g,s)$ is labeled by swg' where $w \in W$ is the suffix of the label assigned to its parent in $P(G^*,S^*)$ (so that w is empty at the root of G^{**}).

(b) Each node $s'' \in S(s,s')$ carries the label $s''w$ where suffix w is the same as that in the label, $s'w$, of the corresponding child of g'', which would have been assigned to it according to (a).

We can see that the suffix is constant within inserted parts and the prefix is constant within subtrees $G^{**}(g,s)$. Evidently, the suffix shows the path in G^{**}

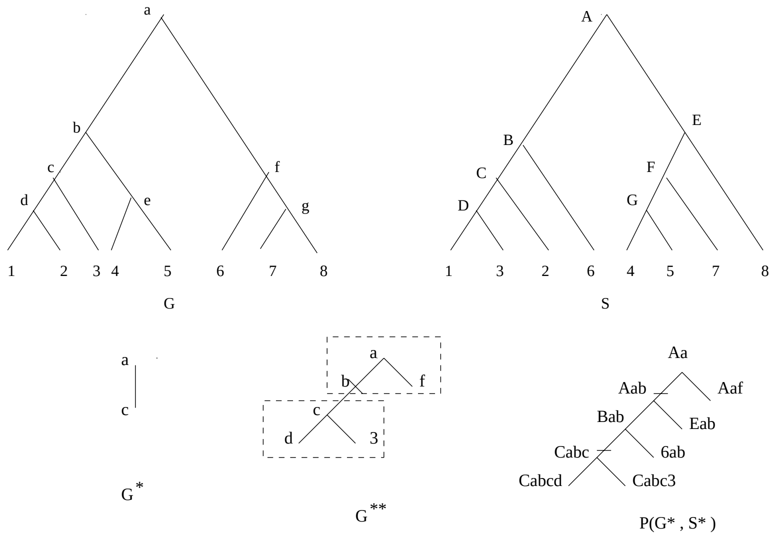

FIGURE 11. The plait frame, $P(G^*, S^*)$, for a gene tree, G, and species tree, S as an extended and relabeled structure of gene subtrees, G^* and G^{**}.

leading from the root to the duplicate copy of the gene at the correspondingly labeled species node: any time when a duplication occurs (at a gene node g), the duplicate copies are distinguished by adding to them "marking signs" corresponding to its children, cg and $\bar{c}g$. Obviously, any suffix assigned in the plait frame, $w = g_1...g_p$, satisfies inclusions $g_p \subset ... \subset g_1$ along the corresponding path in G^{**} where $g_1, ..., g_p$ are considered as clusters in G.

In the gene and species trees shown in Fig. 11, tree G^*, tree G^{**} and the labeled plait frame $P(G^*, S^*)$ are present. The inserted part (between A and C) in S is shown in tree $P(G^*, S^*)$ with cuts.

The plait frame sketches the duplication history of gene G according to species tree S. To show a complete history involving the list of all current species, I, under observation, the *plait tree*, $H(G, S)$, can be defined as obtained from S by substituting subtree $S(s^*)$ (where s^* is the root of S^*) by the plait frame with each of its leaves sw replaced by the subtree $S(s)$ along with all its nodes s' relabeled as $s'w$ by adding the suffix w of the plait tree leaf (see Fig. 12). An important feature in this representation is the labels of the leaves of $H(G, S)$ presenting joint duplication history of gene G in the species set I under investigation. Any leaf labeled by $ig_1...g_p$ relates to a copy of gene G in species i. If $i \notin g_p$, the leaf copy is extinct or just missing in the data. If $i \in g_p$, the leaf $ig_1...g_p$ corresponds to the observed occurrence of the gene in species $i \in I$. This is why we mark leaves $ig_1...g_p$ with $i \in g_p$ as active while the others are marked as non-active. The non-active copies in the plait tree in Fig. 12 are marked by the asterisk, "*".

It can be easily proven that the plait tree, in fact, coincides with the reconciled tree as does the tree in Fig. 12 with the tree in Fig. 7 (up to minor labeling differences). To do that, we need to translate the format of encoding plait tree nodes, $sw \in S \times W$, into the format of labeling nodes of the reconciled tree. In the latter case, the suffix must be a subset of only copy labels. Such a subset, $c(w)$, is obtained, for any $sw \in H(G, S)$, by picking from w those of its symbols that relate only to children of the duplications occurring along the path leading to sw. For example, for the node $Cabcd$ in $H(G, S)$ in Fig. 12, the copy set is $\{b, d\}$, because a and c are themselves duplications.

STATEMENT 10. *The plait tree $H(G, S)$, with its node label suffix words, w, relabeled as subsets $c(w)$, is the labeled reconciled tree $R(G, S)$.*

Proof: The rule for updating the natural partial order in the algorithm for constructing the labeled reconciled tree follows the structure of plait frame $P(G^*, S^*)$. The suffixes are added only from the children cg and $\bar{c}g$, not from the duplicate node g itself. These additions correspond to gene copies. □

4.3. Losses and *-Losses. To analyze the structure of the set of *-losses, we use the subtrees $G^{**}(g, s)$ of the subtree G^{**} defined in section 4.1. For any leaf, g_k, $k = 1, ..., m$, of $G^{**}(g, s)$, the set of non-active copies in the correspondingly relabeled subtree $S(s)$ is $s - g_k$. This is obvious if g_k is a leaf of G^{**}. However, this also holds if g_k is not a leaf of G^{**} because it is substituted by that piece of S which contains $s - g_k$, because further duplications (and copying process) are within g_k as its set-inclusion descent. Thus corresponding *-losses are maximal nodes $s' \in S$ satisfying $s' \subseteq s - g_k$.

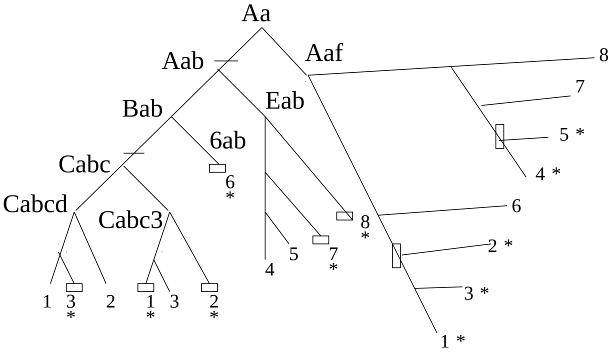

FIGURE 12. The plait tree $H(G, S)$, for the gene and species trees in Fig.1. The label suffixes are omitted below the plait frame nodes, for notational simplicity.

In the example of Fig. 12, there are two subtrees corresponding to children b and f of duplication a with their roots Aaf and Aab, respectively. Since $A = I$ in this case, corresponding sets of non-active copies are $A - f = \{1, 2, 3, 4, 5\}$ and $A - b = \{6, 7, 8\}$, respectively. The *-losses are shown by boxes as the maximal species clusters of non-active copies.

The following two statements show that the concepts of loss and *-loss are equivalent.

STATEMENT 11. *The set of losses corresponding to all the duplication nodes of any subtree $G^{**}(g, s)$ of S^{**}, coincides with the set of *-losses in the corresponding part of $H(G, S)$ (with the number of occurrences of the same loss taken into account).*

Proof: Let us recall that a g'-loss, $s' \in S$, is defined by the condition that ($s' \cap g' \subseteq cg'$ or $s' \cap g' \subseteq \bar{c}g'$) and $s' \cup s''$ overlaps both cg' and $\bar{c}g'$, where s'' is the sibling of s' in S. Say $s' \cap g' \subseteq cg'$ and $s'' \cap \bar{c}g' \neq \emptyset$; thus there exists $i_0 \in s'' \cap \bar{c}g'$. By construction of G^{**}, there exists a leaf $g'' \in G^{**}$, $g'' \subset g'$, such that $i_0 \in g''$. Therefore, there exists a leaf $i_0wg'' \in H(G, S)$ such that $i_0 \in g''$ and w contains cg' as its letter. This implies that in the corresponding subtree of S carrying label suffix wg'' in $H(G, S)$, the species cluster s' contains only non-active copies but $s' \cup s''$ does not, which means that s' is a *-cluster.

To prove that there are no other *-clusters, let us make the following calculation. Let $g_1, ..., g_m$ be the leaves of $G^{**}(g, s)$. As we have seen above, the *-losses are maximal species clusters in $s - g_1$,..., $s - g_m$ which implies that they cover $m-1$ times every $i \in s$. On the other hand, according to Statement 1, the losses corresponding to each of the $m-1$ duplication nodes in $G^{**}(g, s)$ form a partition of s so that these partitions cover s also $m-1$ times, which implies that there cannot exist any *-loss not being a loss. □

Summing up all the losses (and *-losses) in all $G^{**}(g, s)$, we have the following corollary proven.

Statement 12. *The sets and numbers of all losses and *-losses coincide.*

5. Conclusion

Two concepts of duplication, apparently different though based on similar ideas, have been analyzed here. It is proven that the patterns of duplications and their histories emerging in two approaches to modeling duplications are equivalent in the special case that each species contains exactly one gene of a duplicated gene family. One of the approaches deals with copying duplications presented in the reconciled tree, the other with annotating duplications.

In the reconciled duplication history tree all duplications are inferred in such a way that the tree shows how the evolutionary process might have occurred. In particular, the nesting of duplications is easily seen. In contrast, the annotating duplications may be inferred in any order, which makes Mirkin-Muchnik-Smith's annotating construction not so graphical. Moreover, as noted by Page [**17**] the reconciling techniques can be easily extended to the case when several gene copies can be present for the same species (see also section 2.4 in this paper). The problem of extension of the annotating techniques to the case of multiple gene copies has, to the authors' knowledge, never been explored.

However, some good features can be found in the latter approach, too: (i) it allows separating those non-active leaves that definitely are based on a lack of data rather than on real loss - this is the essence of the "gapped loss" concept (though some of the other non-actives also can be due to a lack of data); (ii) several gene trees can be mapped with the annotating duplications onto the same species tree, thus admitting multifaceted evolutionary interpretation, which hardly can be done with the reconciling representation.

Most amazingly, these two biologically meaningful constructions appear to be highly connected with a purely combinatorial approach based on the least-common-ancestor mapping. It is proven that the combinatorial cost function gives exactly the number (though not location) of losses/*-losses. Finally, we have shown that counting the intermediate nodes is equivalent to counting the duplications for which the collateral children are losses.

The least-common-ancestor tree mapping and counting the intermediate nodes (simultaneously, as Eulenstein [**4**] does, or subsequently, as suggested in Zhang [**19**]) can be done in a linear time (over the size of the trees), which makes tree-mapping a welcome instrument in comparing and reconciling gene/species evolutionary trees. The question of how computationally expensive it is to enumerate all the losses (with regard to corresponding duplications) remains open: the annotating and reconciling duplication constructions here requires cubic time (over the size of the trees).

The concepts considered here can be further investigated. For instance, our concept of the labeled reconciled tree involves given species and gene trees. However, characteristics of the reconciled or plait tree should be found in general terms, with no particular species/gene tree given. The problems of revealing corresponding gene and species structures from an abstract reconciled tree have yet to be posed and solved.

Also, it is interesting to investigate what kind of interrelation exists between the duplication/loss measures analyzed in this paper and those tree-difference measures developed in the literature earlier (see, for example, [**1**], [**2**] and [**11**]).

6. Acknowledgements

The authors are indebted to I. Muchnik, R. Page, T. Smith and L. Zhang for presenting their results while unpublished. The authors thank G. Estabrook for his numerous revising suggestions.

References

[1] H. Bobisud and L. Bobisud, A metric for classifications, Taxon. 21 (1972) 607 – 613.

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

[3] G. Estabrook and F. McMorris, When is one estimate of evolutionary relationships a refinement of another?, J. Math. Biology. 10 (1980) 367 – 373.

[4] O. Eulenstein, A linear time algorithm for tree mapping. "Arbeitspapiere der GMD" (1997) No. 1046, Germany.

[5] O. Eulenstein, B. Mirkin and M. Vingron, Duplication-based measures of difference between gene and species trees. (1997) submitted.

[6] O. Eulenstein, B. Mirkin and M. Vingron, Modeling joint history of duplications in evolutionary trees. (1997) submitted.

[7] O. Eulenstein and M. Vingron, On the equivalence of two tree mapping measures, "Arbeitspapiere der GMD" (1995) No. 936, Germany.

[8] J. Felsenstein, Phylogenies from molecular sequences: Inference and reliability, Annu. Rev. Genet. . 22 (1988) 521 – 565.

[9] W. Fitch and E. Margoliash, Construction of phylogenetic trees, Science. 155 (1967) 279 – 284.

[10] M. Goodman, J. Czelusniak, G. Moore, A. Romero-Herrera, and G. Matsuda, Fitting the gene lineage into its species lineage: A parsimony strategy illustrated by cladograms constructed from globin sequences, Syst.Zool. 28 (1979) 132 – 168.

[11] A. Gordon, Hierarchical classification, in P. Arabie, L. Hubert, and G. De Soete (Eds.) Classification and Clustering (World Scientific, Singapore, 1996).

[12] R. Guigó, I. Muchnik, and T. F. Smith, Reconstruction of ancient molecular phylogeny. Molecular Phylogenetics and Evolution. 6 (1996) 189 – 213.

[13] W. Li and D. Graur. Fundamentals of Molecular Evolution (Sinauer Associates, Inc., Sunderland, Massachusetts, 1991).

[14] B. Mirkin, I. Muchnik, and T. F. Smith, A Biologically Consistent Model for Comparing Molecular Phylogenies, Journal of Computational Biology. 2 (1995) 493 – 507.

[15] M. Nei. Molecular Evolution Genetics (Columbia University Press, New York, 1987).

[16] G. Nelson and N. Platnick. Systematics and Biogeography: Cladistics and Vicariance (Columbia University Press, New York, 1981).

[17] R. D. Page, Maps between trees and cladistic analysis of historical associations among genes, organisms, and areas, Systematic Biology. 43 (1994) 58 – 77.

[18] J. Maynard Smith. The Theory of Evolution (Penguin Books Ltd, Harmondsworth, Middleessex, England, 1958).

[19] L. Zhang, A Proof of a Mirkin-Muchnik-Smith Conjecture for Comparing Molecular Phylogenies, Journal of Comp. Biology. (1997) to appear.

University of Bonn, Dept. of Computer Science, Research Group of Prof. Lengauer, Römerstr. 164, D-53117 Bonn, Germany.
E-mail address: `Oliver.Eulenstein@gmd.de`

DIMACS, Rutgers University, P.O.Box 1179, Piscataway NJ 08855, USA and DKFZ, Abteilung Theoretische Bioinformatik, Heidelberg, Germany.
E-mail address: `mirkin@dimacs.rutgers.edu`

DKFZ, Abteilung Theoretische Bioinformatik, INF 280, D-69120 Heidelberg, Germany.
E-mail address: `m.vingron@dkfz-heidelberg.de`