Notung: A Program for Dating Gene Duplications and Optimizing Gene Family Trees

Kevin Chen

Department of Computer Science, Princeton University, Princeton, NJ 08544, USA

(*kcchen@princeton.edu*)

Dannie Durand*

Department of Molecular Biology, Princeton University, Princeton, NJ 08544, USA

(*durand@cs.princeton.edu, http://www.cs.princeton.edu/∼durand*)

Tel: 1-609-258-6874, Fax: 1-609-258-4575

Martin Farach-Colton

Department of Computer Science, Rutgers University, Piscataway, NJ 08855, USA

(*farach@cs.rutgers.edu, http://www.cs.rutgers.edu/∼farach*).

June 6, 2000

Running head: Dating Gene Duplications

Keywords: Gene duplication, gene family, duplication/loss models, phylogeny, evolutionary tree

Abstract

Large scale gene duplication is a major force driving the evolution of genetic functional innovation. Whole genome duplications are widely believed to have played an important role in the evolution of the maize, yeast and vertebrate genomes. The use of evolutionary trees to analyze the history of gene duplication and estimate duplication times provides a powerful tool for studying this process. Many studies in the molecular evolution literature have used this approach on small data sets, using analyses performed by hand. The rapid growth of genetic sequence data will soon allow similar studies on a genomic scale, but such studies will be limited unless the analysis can be automated. Even existing data sets admit alternative hypotheses that would be too tedious to consider without automation.

In this paper, we describe a program called NOTUNG that facilitates large scale analysis, using both rooted and unrooted trees. When tested on trees analyzed in the literature, NOTUNG consistently yielded results that agree with the assessments in the original publications. Thus, NOTUNG provides a basic building block for inferring duplication dates from gene trees automatically and can also be used as an exploratory analysis tool for evaluating alternative hypotheses.

1 Introduction

Yeast is a single cell organism with 6000 genes (Goffeau et al. 1996), while mice have an estimated 50,000 - 100,000 genes (Silver 1995). How did this order-of-magnitude increase in gene number, with its concomitant increase in functional complexity, arise? Gene duplication followed by mutation leads to new function and is considered a principal force driving developmental innovation in vertebrates (Ohno 1970).

The availability of sequence data has catalyzed the study of the impact of duplication, especially whole genome duplication, on the evolution of genomic structure (see (Skrabanek and Wolfe 1998) for a survey), as well as the specialization of function through the evolution of gene families. An important tool in the study of both questions is the construction and analysis of trees based on the sequences of duplicated genes, so called gene family trees.

Until recently, such studies involved a small number of gene families, each represented by ten or twenty sequences, and the analysis could be carried out by visual inspection of the trees (Endo et al. 1997; Hughes 1998; Hughes 1999; Kasahara 1997; Martin 1999; Pebusque et al. 1998; Ruvinsky and Silver 1997). However, as genomic sequence data grows, the number of gene families to be considered in a single genome will grow, and so will the number of trees to be analyzed. For example, in their analysis of duplications in the yeast genome, (Wolfe and Shields 1997) identified 446 duplicated genes. This data set is an order of magnitude larger than the gene duplication studies currently being carried out by hand.

In this paper, we formalize the analytic methods described anecdotally in the molecular evolution studies and cast them into a unified framework. Using this framework, we develop computational methods for analyzing duplication histories and determining duplication dates in rooted trees, as well as exploring two kinds of alternative hypotheses: local rearrangements when the evidence supporting an edge is weak and alternate rootings for unrooted trees. These methods were implemented in a set of tools called NOTUNG that

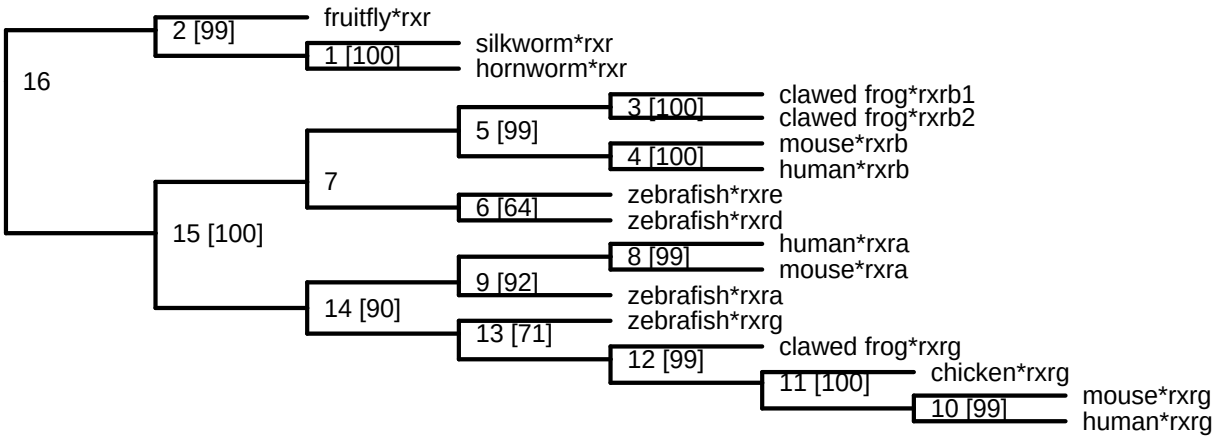

Figure 1: A rooted Neighbor Joining tree for the RXR family reproduced from (Hughes 1998). Interior nodes are labeled numerically. Labels in square brackets represent the percentage of bootstrap samples supporting that branch leading from the label to the root. Values $\leq 50\%$ are not shown.

can be used for exploring alternative hypotheses about duplication events and is a step towards the automated analysis of duplications in large genomic data sets.

An Example of Duplication Analysis. A gene family is "a set of genes descended by duplication and variation from some ancestral gene" (King and Stansfield 1990), typically exhibiting related sequence and function. A *gene family tree (GFT)* is a phylogeny constructed from the sequences of family members, including representatives of the same gene in different species (orthologs) and duplicate genes in the same species (paralogs). A GFT differs from a species tree in that a species may appear more than once.

We begin by considering a typical analysis of gene duplication using a gene family tree. (Hughes 1998) analyzed the evolution of the RXR family, using the rooted tree reproduced in Figure 1, which was constructed using the Neighbor Joining heuristic. Confidence in clustering patterns was assessed using bootstrapping, a statistical resampling method (Efron and Gong 1983).

Summarizing the history of the RXR family that can be inferred from the tree, Hughes states "RXR genes from three insects fell outside of all the vertebrate RXRA, RXRB and RXRG genes. The phylogeny suggests that RXRB diverged first followed by RXRA and RXRG. . . . Zebrafish genes were found to cluster with

mammalian RXRB, RXRA and RXRG, but bootstrap support for these clustering patterns was not strong. Frog RXRB and RXRG genes cluster with their mammalian counterparts and, in each of these cases, there is strong (99%) bootstrap support. The tree thus suggests that RXRA, RXRB and RXRG diverged before the divergence of amphibians and amniotes and probably before the divergence of tetrapods and bony fishes."

Hughes' description makes the following technical points:

- Every node in the tree represents either a speciation or a duplication event. It is possible to find the set of duplication nodes by comparing the gene family tree to a species tree such as the cartoon of the Tree of Life shown in Figure 2. Hughes identified two duplication nodes (14 and 15). There are two more duplication nodes in the RXRB clade (3 and 6) that he does not mention.

- Bounds on the time of duplication, given in terms of major speciation events, can be inferred for each duplication node from the relative positions of speciation and duplication nodes in the tree. According to the topology shown in Figure 1, duplications 14 and 15 are both bounded above by the divergence of vertebrates and insects, and bounded below by the divergence of tetrapods and bony fishes. The upper bound can be inferred from the clustering of insect genes outside the gene family clades and the lower bound from the presence of a fish gene in each subfamily clade.

- When a duplication hypothesis depends on a node with weak support in the sequence data, alternative hypotheses should be considered. Because the bootstrap values associated with the zebrafish branches in Figure 1 are low, topologies in which zebrafish genes do not cluster within the subfamilies should also be considered. For this reason, the divergence of the amphibian lineage may be a more reliable lower bound for duplications 14 and 15.

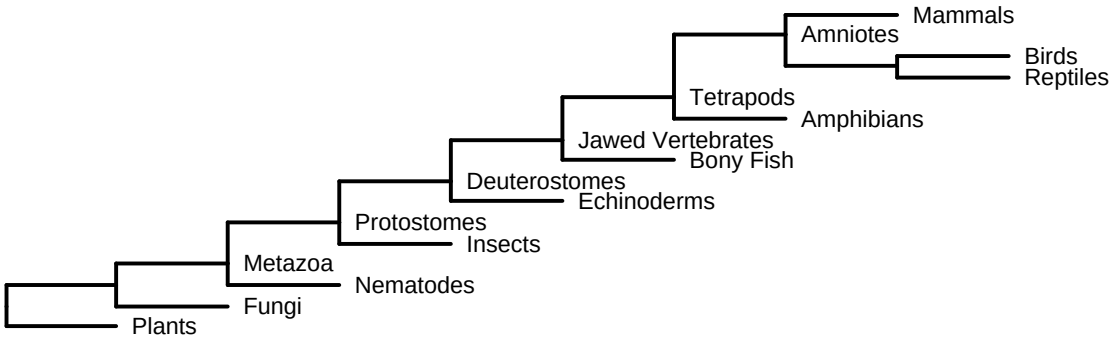

Figure 2: A species tree showing major speciation events in the eukaryote lineage.

Our Results. Hughes' analysis is typical of many studies in the biology literature of gene duplication using ad hoc analysis of gene family trees (Endo et al. 1997; Hughes 1998; Hughes 1999; Kasahara 1997; Pebusque et al. 1998; Ruvinsky and Silver 1997). These analyses are based on the assumption that a rooted GFT is a hypothesis concerning the evolution of a gene family. The duplication history of the family can be inferred from a GFT, where we define the duplication history to be a list of the duplications that occurred and a time range for each. If the topology of the tree is unambiguous and the tree is rooted, then the inferred history is unique.

However, as Hughes' discussion of weak bootstrap values illustrates, in many instances it is not possible to infer the tree topology unambiguously from the sequences and, in this case, alternate histories must be considered. Alternate histories must also be considered if the tree is unrooted. While a rooted GFT is a hypothesis concerning the evolutionary history of a gene family, an unrooted gene family tree represents a set of such hypotheses, one for each possible rooting. Unlike the RXR example, many of the trees reported in the literature are unrooted because it is frequently not possible to find a sequence from the gene family in a suitable outgroup species.

These considerations suggest two computational problems. The solution to the first problem is used as a subroutine in the solution of the second.

1. When the correct, rooted gene family tree is known, the problem is to infer the duplication history from the tree.

2. When the correct rooted GFT is not known, the problem is to find the rooted gene family tree that best represents the evolutionary history of the gene family.

The most general approach to the second question is to consider all rooted binary trees with respect to an optimization criterion based on a model of gene family evolution. That is, given G, a set of orthologous and paralogous gene sequences from a gene family and T_S, a binary species tree containing exactly the species in G, find the rooted binary tree, T_G, whose leaf set is G, that optimizes a given optimization criterion. The forces that govern the evolution of gene families involve gene duplication and loss as well as the evolution of the gene sequences themselves, as we discuss at greater length in Section 4. The optimization criterion should therefore take all three processes into account. However, it is not obvious how to determine the relative importance of macroscopic mutations like gene duplication and loss with microscopic mutations like point mutations in a single optimization criterion. Instead, our approach is to start with a gene family tree inferred from sequence alone. When the sequence data is not sufficient to reconstruct an unambiguous, rooted GFT, we use optimization criteria based on gene duplication and loss to consider alternate hypotheses.

We review previous work involving the relationship between gene family trees and species trees in Section 2. In Section 3, we present a linear time algorithm for inferring the duplication history of a gene family from a rooted GFT. We address the problem of finding the optimal alternate hypothesis, when the correct rooted GFT is not known, in Section 4. After introducing two optimization criteria for GFTs based on duplication and loss in Section 4.1, we discuss how to use them to select the optimal rooting of an unrooted tree in Section 4.2. In Section 4.3, we present an algorithm for finding the optimal gene duplication history given a rooted GFT with weak branches. Although in the general case, the number alternate hypotheses is

superexponential in the number of taxa, we show that with respect to the criteria introduced in Section 4.1 it is possible to vastly reduce the search space if the weak branches are sparse, by identifying sets of weak branches that can be evaluated independently.

We implemented these algorithms and tested them on gene family trees published in the molecular evolution literature (Hughes 1998; Pebusque et al. 1998; Ruvinsky and Silver 1997). As summarized in Section 5, the gene family histories generated by our program are consistent with the assessments presented in the original papers. For unrooted trees and nodes with low bootstrap values, our program generates and scores alternate hypotheses, providing an exploratory analysis tool. In addition, an explicit statement of all hypotheses helps mitigate any biased expectations of the data the user might have.

2 Related Work

The problem of disagreement between gene trees and species trees was first raised by (Goodman et al. 1979) in the context of inferring a species tree from a gene tree that may contain paralogies. They introduced the notion of a map between a gene tree and a species tree and suggested a cost function for evaluating a species tree with respect to a gene tree based on edit distance, gene duplication and gene loss.

These concepts were further developed and formalized in (Guigo et al. 1996; Hallett and Lagergren 2000; Ma et al. 1998; Ma et al. 2000; Mirkin et al. 1995; Page 1994; Page and Charleston 1996; Stege 1999; Zhang 1997). Formally, given a set of rooted gene trees, $\{T_G\}$, the problem is to find the species tree, T_S, that optimizes an evaluation criterion. Several optimality criteria have been proposed (see (Eulenstein et al. 1996; Eulenstein et al. 1998) for a comparative survey), all of which attempt to capture the notion that gene duplication and subsequent loss are rare events. These criteria involve constructing a mapping, $M : T_G \mapsto T_S$, between a gene tree and a species tree, that is used to compute the cost function. Several authors have pointed out that it is difficult to distinguish true gene loss from genes that have not yet been sequenced and discuss approaches to distinguishing true losses from apparent losses in the cost function (Goodman et al. 1979; Mirkin et al. 1995; Page and Charleston 1996).

When inferring a species tree from a gene tree, the gene tree is assumed to be correct and the true species tree is unknown. We, on the other hand, assume that the true species tree is known and use it to infer the duplication history from a gene tree. While we share some mathematical structure with (Guigo et al. 1996; Hallett and Lagergren 2000; Ma et al. 1998; Mirkin et al. 1995; Page 1994; Stege 1999), most notably the mapping $M(\cdot)$, we consider the problem of dating duplication events and generating and evaluating alternate hypotheses. The problem of finding an optimal species tree is NP-hard (Ma et al. 1998; Ma et al. 2000) for the optimality criteria considered so far. In contrast, dating duplication events in rooted and unrooted trees

is a computationally tractable problem, which is crucial if we hope to apply this to large data sets.

The methods for inferring species trees from gene trees surveyed here do implicitly generate duplication histories in rooted trees although the time of duplication is generally not considered. In addition, most optimality criteria surveyed here are subject to the constraint that each species may only be represented once in G and hence would not be suitable for our application. A notable exception is the work of Page and Charleston (Page and Charleston 1997), who have developed two software packages, COMPONENT and GENETREE, that, as well as inferring species trees, will compute and display duplication histories for rooted gene trees. This provides an interactive, exploratory analysis tool, but cannot be used to automate the analysis of large data sets.

The first analyses of alternate rootings of unrooted gene trees have appeared very recently. (Hallett and Lagergren 2000) presented an algorithm to infer the species tree from a set of rooted gene trees in polynomial time under the restriction that the species tree can be reconciled with the gene tree using at most k simultaneous copies of the gene along its branches. In this context, they also developed a polynomial time algorithm that selects a root for each unrooted gene tree such that the total number of duplications required to explain the data is minimized. Their algorithm considers all possible rootings of all trees using a brute force approach. In work developed independently and presented simultaneously, we (Chen et al. 2000) presented an algorithm that, given an unrooted gene tree and a fixed species tree, ranks all possible rootings of the gene tree according to an optimization criterion that estimates the plausibility of the resulting duplication histories. By using a data structure that stores intermediate results, we avoid the brute force approach, ranking all rootings in time linear in the size of the gene tree. In the current paper, we apply this approach to a wider range of optimization criteria based on duplication and loss. None of the work surveyed here addresses alternate hypotheses due to weak edges. In (Chen et al. 2000), we introduced this problem

and proposed a heuristic solution. In the current paper, we develop an exact approach.

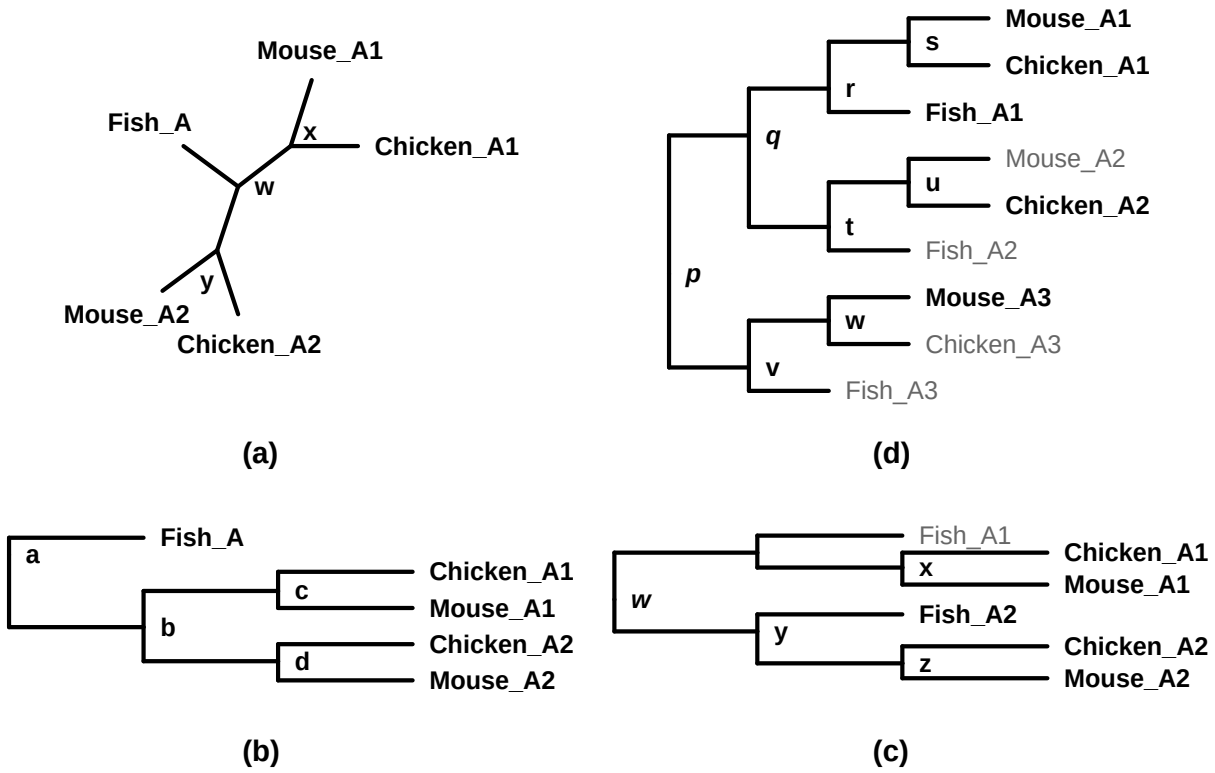

Figure 3: Gene family trees for the hypothetical gene family, A, with five known gene sequences (two in mouse, two in chicken and one in fish.) (a) An unrooted GFT for A. (b) – (d) Three alternate rootings of the GFT in (a). Duplication nodes are shown in italics and missing genes are shown in grey.

3 Inferring Gene Family Histories from Rooted Trees

The process of inferring a duplication history from a gene family tree is illustrated in Figure 3 which shows an unrooted GFT for a hypothetical gene family, A, and three possible rootings of that tree. Each rooted tree corresponds to a single duplication history. For example, the rooted tree in Figure 3(b) suggests that gene A was present in the common ancestor of fish and tetrapods and was duplicated after the divergence of fish and before the separation between birds and mammals. In contrast, Figure 3(c) implies that the duplication took place before the divergence of fish and tetrapods. Although there is only one fish sequence, it clusters with the genes in the A_2 family, suggesting either that the fish A_1 gene has been lost due to mutation or that

it has not yet been sequenced. Because these trees are rooted, we can observe evidence of the duplication through clustering, even when one of the two paralogs is missing.

The problem of reconstructing a duplication history from a rooted gene family tree can be stated formally as follows:

> **Problem 1:** Given a rooted GFT, T_G, and T_S, a binary species tree containing only the species in T_G, identify all duplication nodes and determine lower and upper bounds on the time of each duplication in terms of speciation events.

Both the identification of duplication nodes and the calculation of duplication dates require constructing a mapping, $M(\cdot)$, from every node in T_G to a target node in T_S. We construct this map as follows. Let v be a node in T_G and let $l(v)$ and $r(v)$ be its left and right children, respectively. $M(\cdot)$ maps each leaf node in T_G to the node in T_S representing the species from which the sequence was obtained. (Leaf nodes in T_G represent sequences, whereas leaf nodes in T_S represent species.) Each internal node in T_G is mapped to the least common ancestor (lca) in T_S of the target nodes of its children; that is, $M(v) = lca(M(l(v)), M(r(v)))$. For example, in Figure 3(b), the leaf nodes are mapped to *fish*, *chicken*, *mouse*, *chicken*, *mouse*, from top to bottom. $M(c) =$ *amniote*, since the lca of *mouse* and *chicken* is *amniote* in the Tree of Life (Figure 2). Nodes b and d both map to *amniote*, while $M(a) =$ *jawed_vertebrate*.

An algorithm for constructing the mapping, $M(\cdot)$, and identifying duplication nodes has been developed independently in the context of using multiple gene trees to generate a species tree. By using fast lca queries (Bender and Farach-Colton 2000; JáJá 1991)[1], $M(\cdot)$ can be computed in $O(|T_G|)$ time. While our goals are different, we share a key algorithmic component with this work. We refer the reader to (Mirkin

[1] Several early papers on lca computation were too complicated to implement, even papers which claimed to be "simplifications", and had large hidden constants. Thus, it is a "folk theorem" that any algorithm which uses lca precomputation is impractical. However, the state of the art of lca computation has progressed since those early papers, and there now exist lca algorithms which are very simple and very practical.

et al. 1995) for a complete description and proofs.

Observe that under the mapping, a node v in T_G is a speciation node if its children are mapped to independent lineages in T_S. In Figure 3(c), x is a speciation node since mammals and birds are separate lineages. If the children of $M(v)$ share a lineage, then v is a duplication node. When this occurs, either both children have the same label or one child's target in T_S is an ancestor of the other's and v will be mapped to the same label as the ancestral child. For example, node w is a duplication node in Figure 3(c) because $M(y) =$ *jawed_vertebrate* is an ancestor of $M(x) =$ *amniote*.

Observation 1 *Node v is a duplication node if and only if $M(v) = M(l(v))$ or $M(v) = M(r(v))$ or both.*

The mapping, $M(\cdot)$, can also be used to compute lower and upper bounds on the time of duplication. Let v be a duplication node in T_G. Since copies of the duplicated gene are observed in descendents of both $l(v)$ and $r(v)$, the duplication must have been present in their least common ancestor, yielding the lower bound $L(v) = M(v)$. By a similar argument, the upper bound can be shown to be the target of the nearest ancestor, a_v, of v that is a speciation node. Since copies of the duplicated gene are present in only one of the subtrees rooted at children of a_v, the duplication must have occurred in a more recent species. If v has an ancestor that is a speciation node, we set the upper bound $U(v) = M(a_v)$. Otherwise, $U(v)$ is the origin of life. For example, in Figure 3(c), the bounds on the duplication node, w, are $L(w) =$ *jawed_vertebrate* and $U(w) = \infty$, since w is the root node of T_G. In Figure 3(b), b is a duplication node with label *amniote*. Its parent, a is a speciation node with label *jawed_vertebrate*. Thus, $L(b) =$ *amniote* and $U(b) =$ *jawed_vertebrate* .

Observation 2 *The duplication associated with a node, v, in T_G, occurred after the speciation event $U(v)$ and before the speciation event $L(v) = M(v)$.*

Notice that we can compute U in $O(|T_G|)$ time.

4 Evaluating Alternate Gene Family Histories

As the example in Figure 3 shows, alternate rootings of an unrooted GFT give alternate duplication histories. Alternate hypotheses such as these can be evaluated in terms of the number of gene duplications and losses implied by the associated rooted trees. For example, Figure 3(a) requires a single gene duplication to explain, while Figure 3(c) involves one gene duplication and one gene loss. In contrast, the tree shown in Figure 3(d) has three subfamilies resulting from two duplications (nodes p and q) and four losses. A duplication and loss model can also be used to evaluate rearrangements of weak branches.

We formalize the intuition provided by this example in the remainder of this section. First, we develop two optimization criteria based on duplication and loss for evaluating alternate hypotheses in Section 4.1. We discuss how to apply these to unrooted trees in Section 4.2. Algorithms for evaluating rearrangements of weak edges with respect to these criteria are presented in Section 4.3.

4.1 Optimization Criteria for Duplication Histories

We consider two criteria for evaluating alternate gene duplication histories. The first scoring function, $C_{dl}(\cdot)$, calculates gene duplication and loss explicitly and can be stated as

$$C_{dl}(T_G) = c_\lambda \cdot \lambda + c_\delta \cdot \delta,$$

where δ represents the number of duplication nodes in T_G, λ represents the number of gene losses and c_λ and c_δ are constants that determine the relative importance of duplications and losses in the scoring function. The best choice of c_λ and c_δ is an open problem. In this paper, we use $c_\lambda = c_\delta = 1$ consistent with the work cited in Section 2.

We define λ to be the most parsimonious number of gene losses required to explain the duplication

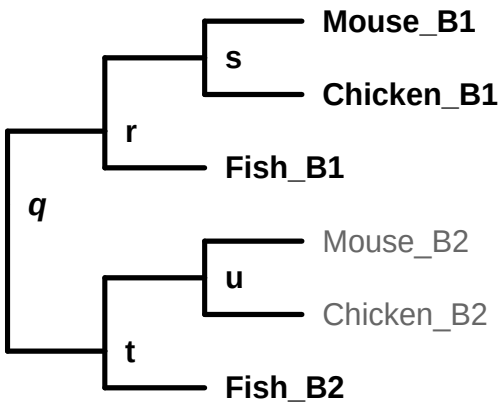

Figure 4: A rooted tree for the fictional B gene family showing a gene duplication at node q and a gene loss loss in the common ancestor of mouse and chicken.

history. That is, if an entire subtree of T_S is missing from T_G, we assume that the loss occurred at the root of the subtree and count it as a single event. For example, in Figure 4, we assume that B_2 was lost in the common ancestor of mouse and chicken rather than two independent gene losses in the leaf species, resulting in $\delta = 1$ and $\lambda = 1$.

The number of duplications, δ, is a byproduct of the algorithm for inferring duplication histories described in Section 3. The loss, can be computed by summing the gene losses over all edges in T_G:

$$\lambda = \sum_{e \in T_G} \textsc{EdgeLoss}\ (e).$$

The loss associated with the edge $e = (\textit{parent}, \textit{child})$ is given by

$$\textsc{EdgeLoss}\ = (\textsc{Depth}\ (M(\textit{child})) - \textsc{Depth}\ (M(\textit{parent})) - 1) + \textsc{IsDup}\ (\textit{parent}),$$

where $\textsc{Depth}\ (M(v))$ returns the depth of the target of node v in T_S, (the species tree for the species in T_G only) and $\textsc{IsDup}\ (v) = 1$ if v is a duplication node and zero otherwise. The first term in this equation

counts gene losses associated with speciation nodes. The labels of a speciation node and its child should differ by one if no loss has occurred and will increase by one with each gene loss. In contrast, if no gene loss has occurred on the edge immediately below a duplication node, the labels of a duplication node and its child should be identical. Thus, if *parent* is a duplication node, the first term will be off by one. The second term corrects for this case.

We define a second cost function, $C_{ub}(\cdot)$, that is an upper bound on the maximum number of missing leaf nodes plus the number of duplication nodes. Let the cost, $k(t)$, of a node t in T_S, be the number of leaf nodes under t. That is, $k(t)$ is the maximum number of gene losses that could occur following a duplication in the ancestral species t. If t is a leaf, then $k(t) = 1$. The cost of the gene tree, T_G, is

$$C_{ub}(T_G) = \sum_{d \in \mathcal{D}} (k(M(d)) + 1),$$

where $\mathcal{D}$ be the set of duplication nodes in T_G. Under $C_{ub}(\cdot)$, the cost of the tree in Figure 4 is four, since $k(M(q)) = k(\textit{jawed\_vertebrate}) = 3$. In comparison, under $C_{dl}(\cdot)$, the cost of the tree is two, when $c_\lambda = c_\delta = 1$.

These criteria represent two extremes on the spectrum of gene-loss models. If a gene is missing in species which share a common ancestor, is this due to a single loss in the common ancestor or did several independent losses occur? $C_{dl}(\cdot)$ represents the minimum number of gene losses required to explain the data, while $C_{ub}(\cdot)$ is an upper bound on the maximum number of possible gene losses. We believe that gene loss is a rare event and that in most cases, the more parsimonious model is likely to be the correct one. However, by testing our methods on two diametrically opposed models, we can gain insight into how sensitive our methods are to the choice of model.

4.2 Unrooted Trees

Unlike a rooted tree, which encodes a single evolutionary hypothesis, an unrooted tree with $n = |E|$ edges represents up to n different hypotheses, one for each possible rooting. We propose to use the two cost functions presented above to rank these n hypotheses, making it possible to consider the most biologically plausible hypotheses first. We can state the unrooted tree problem formally as follows:

> **Problem 2.1:** Given an unrooted GFT, $T_G = (V, E)$, and an optimization criterion, $C(\cdot)$, determine the duplication history of the tree rooted at e, for every edge, $e \in E$. Rank the histories according to the criterion, $C(\cdot)$.

To do this, we must label each node in T_G as either a duplication or speciation node under every possible rooting. However, the mapping, $M(\cdot)$ changes with the position of the root. A simple quadratic time algorithm would be to apply the rooted tree algorithm to every possible rooting. However, we can derive a linear time algorithm to compute all labelings as follows. Notice that, with respect to a node v, we can partition all possible rootings of the tree into three groups: the root must be in one of three directions, depending on which of the edges incident on v is on the path from v to the root. Let e_1, e_2 and e_3 be the edges incident on v. The status of v as either a duplication or speciation only depends on which edge points towards the root. This is because if we fix which edge is up, the subtree rooted at v is fixed, and so is the bottom-up lca computation. Now, we need only compute $M_{e_1}(v)$, $M_{e_2}(v)$ and $M_{e_3}(v)$, one $M(\cdot)$ value for each possible "up" edge, from which we can compute the labeling under any desired rooting in linear time.

To compute the three values we simply do the recursive computation at each node in any order. That is, suppose we want to compute $M_{e_2}(v)$, for some v. This determines which two nodes are down. Call them u and w. Then $M_{e_2}(v) = lca(M_{\{v,u\}}(u), M_{\{v,w\}}(w))$. We recursive compute $M_{\{v,u\}}(u)$ and $M_{\{v,w\}}(w)$. In order to keep from recomputing the same value over and over, we simply store all values in a table as we

compute them. Thus, after we have computed $M_{\{v,u\}}(u)$ once recursively, we can look it up in constant time without need for recomputation in the future. Thus, all $3n$ values can be computed in $O(n)$ time. Similarly, any cost function that depends on a function of $M(\cdot)$, including $C_{dl}(\cdot)$ and $C_{ub}(\cdot)$, can be computed in $O(n)$ time. Thus, linear time is sufficient to rank all n rootings and print a constant number of top ranking duplication histories. Printing all duplication histories takes longer.

4.3 Rooted Tree Rearrangements

As in the case of unrooted trees, alternate hypotheses for weak edges can be selected with respect to an optimization criterion. While the following analysis can be applied to any arbitrary set of edges deemed suspect by the user, typically bootstrapping (Efron and Gong 1983) is used to associate a measure of confidence with every edge in a phylogeny. We use a rearrangement strategy to identify appropriate alternate hypotheses for weak edges as follows.

The removal of any edge, e, in a tree bipartitions the set of leaf nodes. If the bootstrap value of e is low, it suggests that the evidence in the data for that bipartition is weak. It does not reflect on the certainty of the structure of any other part of the tree. In reconstructing the duplication history of a rooted GFT, we consider alternate hypotheses associated with a weak edge, e, by generating *Nearest Neighbor Interchanges (NNI's)* around e. Examples of NNI rearrangements are shown in Figure 5 (see, for example, (Swofford et al. 1996), for a more detailed description.) The NNI, around an edge $e = (\textit{parent}, \textit{child})$ shown in bold, is effected by swapping the subtree rooted at the sibling of *child* with either its left or its right subtree. This rearrangement generates alternate bipartitions for e while leaving all other bipartitions associated with the tree unchanged.

An NNI in T_G will change the mapping, $M : T_G \mapsto T_S$, resulting in a new mapping and in some cases, will also change cost of the tree and the duplication history. For example, Figure 5(a) shows a scenario where, presumably, the frog and fish genes were incorrectly placed with respect to each other due to weak

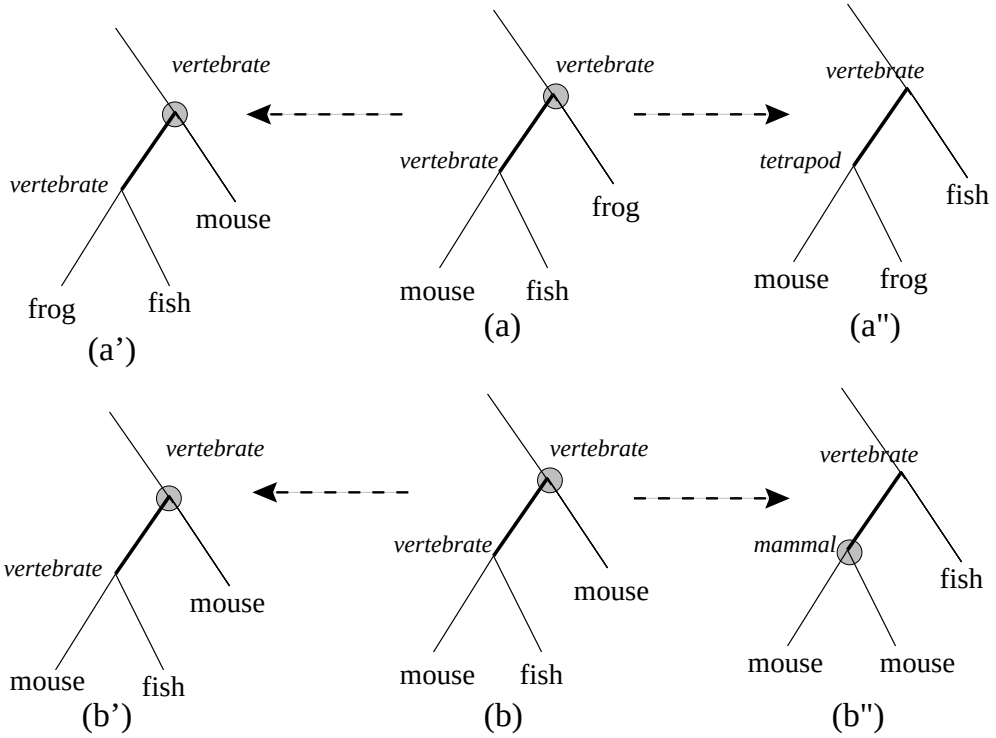

Figure 5: Two tree fragments, each with the three possible Nearest Neighbor Interchanges around the edge shown in bold. Duplication nodes are shown as grey circles. Each node, v, is labeled with its target, $M(v)$, in the species tree.

signal in the sequence data. The two internal nodes in this tree fragment are both labeled *vertebrate*. NNI a'' changes the label of the deeper internal node, thereby eliminating a false duplication. The example in Figure 5(b) is more ambiguous. The middle tree represents a duplication before the divergence of fish followed by a loss in the fish lineage. The rearrangement b'' changes $M(\cdot)$ and moves the duplication to the deeper node, resulting in a duplication after the fish-tetrapod split and eliminating the gene loss. Cost functions based on gene loss tend to favor histories with more recent duplications (e.g., b'' rather than b). This is a disadvantage to the duplication/loss model since both b'' and b are plausible and which scenario is more likely requires specialized knowledge of gene family.

We now state the problem of optimizing a tree with weak edges formally:

> **Problem 2.2:** Let $T_G = (V, E)$ be a rooted GFT and let $W \subseteq E$ be a set of weak edges. Define $\mathcal{T}_{G,W}$ to be the set of trees that can be derived from T_G by NNI operations across edges in W. Find the tree $T_G^* \in \mathcal{T}_{G,W}$ such that T_G^* optimizes a given criterion, $C(\cdot)$.

Note that when $W = E$, the problem reduces to that of finding the optimal gene family tree counting only duplications and losses and ignoring sequence data.

It is not known if an efficient algorithm for finding T_G^* exists. No doubt the existence of such an algorithm depends on the choice of $C(\cdot)$. In (Chen et al. 2000), we considered the following simple heuristic for finding optimal trees: Enumerate the weak edges bottom-up and perform any NNI that reduces the label of the *child* node of the weak edge under consideration. Once that choice has been made, the weak edge is considered fixed and rearrangements of weak edges higher in the tree are considered. While we found that this greedy heuristic works well on the real data sets we applied it to, it is possible to produce biologically plausible examples where the heuristic performs badly.

In the current paper, we therefore consider algorithms that find the actual optimum. The naïve algorithm

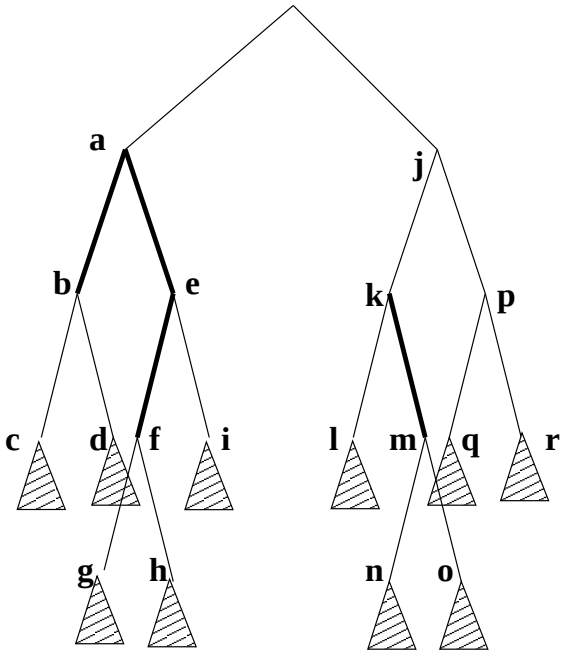

Figure 6: A GFT with four weak edges, shown in bold. The triangles represent subtrees whose structure is not made explicit.

for such an optimization is to enumerate all possible trees and to evaluate $C(\cdot)$ on each one. This is clearly not feasible in general, since the number of trees is superexponential in the number of taxa. However, in many practical cases, the weak edges are sparse and typically appear in small connected components. We will show that for some choices of $C(\cdot)$, in particular $C_{dl}(\cdot)$ and $C_{ub}(\cdot)$, it is sufficient to optimize these connected components independently rather than exhaustively enumerating all trees in $\mathcal{T}_{G,W}$, reducing the cases to be considered to a feasible number.

Consider the tree in Figure 6, which contains four weak edges, shown in bold. These edges form two connected components: ((a,b),(a,e),(e,f)) and ((k,m)). Given a connected component of weak edges, W_i, let T_i be the rooted binary tree formed by adding all edges that are immediate descendents of a weak edge to W_i. T_i has $k_i = |W_i| + 2$ leaves. For example, for the connected component ((a,b),(a,e),(e,f)), T_i is the tree with root a and leaves c, d, g, h and i. (Note that the leaves of T_i may be subtrees of T_G.)

Thus, each connected component induces a rooted, binary tree embedded in the GFT. For each such T_i, the set of trees that can be generated by applying an arbitrary sequence of NNIs to edges in W_i is equivalent to the set of all rooted binary trees with the same root and leaf set as T_i. We call this set $\mathcal{T}_i$. The number of trees in $\mathcal{T}_i$ is $(2k_i-3)!/[2^{k_i-2}(k_i-2)!]$. Let T_i^* be the tree from $\mathcal{T}_i$ that optimizes $C(\cdot)$ and let $\hat{T}_G$ be the tree obtained by replacing each T_i with T_i^*. The tree, $\hat{T}_G$, can be generated through independent optimization of the $\{T_i\}$ by the following algorithm:

Algorithm A:

Compute $M(x)$ for every node in T_G.

Foreach T_i,

Compute the optimal rearrangement tree, T_i^*, by exhaustive enumeration.

Replace T_i with T_i^* in T_G.

Recompute $M(x)$ for every node in T_G.

Note that there may be more than one optimal rearrangement T_i^* of T_i. If T_i is optimal, we report this as T_i^*. Otherwise, we pick an arbitrary optimal solution to be T_i^*.

Below we show that, when $C(\cdot)$ is $C_{dl}(\cdot)$ or $C_{ub}(\cdot)$, optimization of each T_i independently will yield the globally optimal tree, T_G^*. Thus, we do not need to consider the product of all possible rearrangements of all T_i and this approach will substantially speed up the computation in cases where no T_i is too large. Our exhaustive algorithm will, in fact, run in time linear in the number of components and exponential in the size of the largest component.

Theorem 1 *Let $C(\cdot)$ be any cost function such that the cost of the subtree of $\hat{T}_G$ rooted at any node v is of*

the form

$$C(T_v) = C(T_{l(v)}) + C(T_{r(v)}) + f(M(v), M(r(v)), M(l(v)));\tag{1}$$

that is, the cost of a tree rooted at v depends on the cost of its subtrees and a function, $f(\cdot)$, that depends only on the labels of v and its children. Then $\hat{T}_G = T^*_G$.

Proof: Let $\hat{T}_G$ be the tree resulting from running Algorithm **A** on $T_G = (V, E)$. We show that $C(T_v)$ is optimal for any vertex, $v \in V$, if v is the root of some T_i or if v is not in $\bigcup\{T_i\}$.

Case 1: v is a root of some weak component, T_i. In this case, $C(T_v)$ is optimal by brute force optimization.

Case 2: v is outside the T_i. $C(T_v)$ is optimal if each of the terms of Equation 1 is optimal. $C(T_{l(v)})$ and $C(T_{r(v)})$ are optimal by induction. Thus, it is sufficient to show that there is no rearrangement that can reduce the value of $f(\cdot)$, to show that $C(T_v)$ is optimal. To see that this is true, note that the label of any node depends only on the set of leaves below it. Although rearrangement can change the structure of the $\{T_i\}$ and hence the labels of their interior nodes, the labels of the roots of the $\{T_i\}$ and nodes outside them are unchanged.

Since the root, r, of T_G is either outside of $\bigcup\{T_i\}$ or a root of some T_i, $C(T_r)$ is optimal. Thus, $\hat{T}_G = T^*_G$. ■

Corollary 1 *If $C(\cdot) = C_{dl}(\cdot)$ or $C(\cdot) = C_{ub}(\cdot)$, then $\hat{T}_G = T^*_G$.*

Proof: Both $C_{dl}(\cdot)$ and $C_{ub}(\cdot)$ satisfy Equation 1.

■

The time to optimize a single connected component, W_i, is the product of the number of possible rearrangements of T_i and the time to score each rearrangement. Thus, the time to compute optimal rearrange-

ment tree using Algorithm **A** is

$$\sum_{i=1}^{N}(\frac{(2k_i-3)!}{2^{k_i-2}(k_i-2)!}\cdot O(k_i)),$$

where N is the number of components. The term corresponding to the number of rearrangements of the largest component will dominate and is exponential in the number of edges in the largest component. However, the running time is linear in the number of components. If weak edges are sparse, and in particular, individual components are small, this run-time will not be onerous in practise. In the trees culled from the literature for this study, no connected component had more than three edges. NOTUNG 's running time on these trees never exceeded one second of cpu-time per tree on an UltraSparc 10.

5 Experimental Results

The algorithms described in the previous sections have been implemented in Java program called NOTUNG. NOTUNG takes a gene family tree, T_G, a species tree, T_s and a bootstrap threshold, τ, as input. The user may select either of the cost functions described in Section 4.1 to evaluate alternate hypotheses. Input trees are represented in Newick format (http://evolution.genetics.washington.edu/phylip/newicktree.html). For rooted trees, NOTUNG generates a gene duplication history as output; that is, a list of duplication nodes, with bounds on the time of duplication for each one. If the rearrangement option is turned on, NOTUNG will also compute the optimal rearrangement tree according to Algorithm **A**. Currently, we use the tie–breaking rule described in Section 4.3. In this case, NOTUNG will present the original duplication history and the history for the optimized tree, giving scores for both. In the next release, we plan to report all possible optima to the user. For unrooted trees, NOTUNG considers all possible rootings and computes a duplication history for each. These histories are ranked according to the cost function selected. Note that our algorithms, as designed, work with binary and higher degree trees, but NOTUNG currently only handles binary trees. Similarly, there is no algorithmic reason why the rearrangement option cannot be applied to every every possible rooting of an unrooted tree automatically, but this approach has not implemented it. Since most rootings of an unrooted tree are biologically implausible, we leave it to the user to select the rootings of interest and apply the rearrangement algorithm to them explicitly.

One goal of the experimental work presented here is to determine whether these cost functions rank alternative hypotheses effectively. Below we describe NOTUNG'S performance on rooted trees, unrooted trees and trees with low bootstrap values. As test data, we used all "non-pathological" trees from three recent articles on large scale duplication (Hughes 1998; Pebusque et al. 1998; Ruvinsky and Silver 1997). We eliminated non-binary trees and trees based on genes with complicated internal structure such as mosaic

Gene Family	Source	
ANK	(Pebusque et al. 1998)	unrooted
C	(Hughes 1998)	rooted
CRYB	(Ruvinsky and Silver 1997)	unrooted
EGR	(Pebusque et al. 1998)	unrooted
HSP70	(Hughes 1998)	rooted
LHX	(Ruvinsky and Silver 1997)	unrooted
NOTCH[2]	(Hughes 1998)	rooted
PBX	(Hughes 1998)	rooted
PSMB	(Hughes 1998)	unrooted
RXR	(Hughes 1998)	rooted
TCF	(Ruvinsky and Silver 1997)	unrooted
TEN	(Hughes 1998)	rooted
VMAT	(Pebusque et al. 1998)	unrooted

Table 1: Gene family trees used in experiments.

genes or genes with repeated domains, and trees that show evidence of horizontal gene transfer. We analyzed the remaining thirteen trees (summarized in Table 1) using NOTUNG and compared the automatically generated results with the verbal analysis presented in the source paper.

NOTUNG compares the input GFT with a species tree to infer the duplication history. Since there are many competing hypotheses concerning the topology of the Tree of Life, our program allows the user to supply a species tree as input. In the experiments described below, we tried, to the extent that it was possible to determine from the text, to use the same species tree as the authors who originally analyzed the tree. Most authors used a tree consistent with that shown in Figure 2. Pebusque *et al.* (Pebusque et al. 1998) used a variant in which nematodes are included in the protostome clade. The exact tree used is shown in Figure 7. This tree is constructed from information in the University of Arizona Tree of Life project (http://phylogeny.arizona.edu/tree/phylogeny.html) edited by D. R. Maddison and W. P. Maddison, the NCBI Taxonomy database (http://www.ncbi.nlm.nih.gov/Taxonomy/tax.html) and (Graur et al. 1996), for the classification of rabbits.

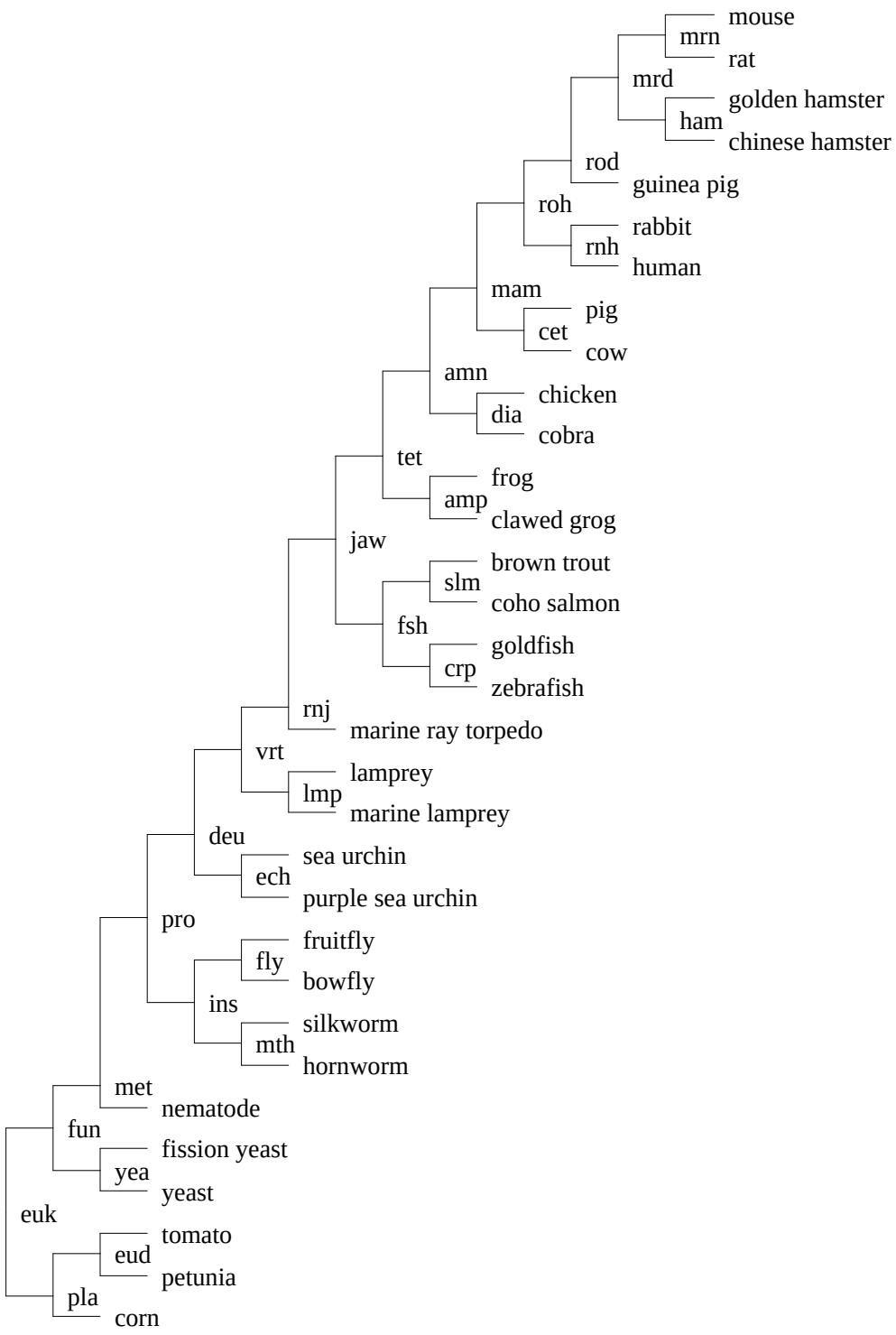

Figure 7: Tree of Life

Gene Family	Original Tree			Optimal Tree		
	δ	λ	C_{ub}	δ	λ	C_{ub}
C	4	14	30	3	11	22
HSP70	9	18	50	6	7	25
NOTCH	5	13	28	4	10	22
PBX	3	3	11	3	3	11
RXR	4	3	16	4	3	16
TEN	2	8	12	2	8	12

Table 2: Scores of rooted trees with respect to $C_{dl}(\cdot)$, given in terms of the number of duplications and losses, and $C_{ub}(\cdot)$. Scores are given for the original tree and the optimal rearrangement tree. The PBX, RXR and TEN trees were unchanged by rearrangement.

5.1 Rooted Trees

NOTUNG was applied to the rooted trees in Table 1. The scores of the resulting duplication histories are shown in Table 2. The histories constructed by the program were consistent with the analysis of (Hughes 1998) in all cases. Generally, NOTUNG finds a superset of the duplications discussed by Hughes, since he only mentions those duplications that are relevant to the biological question he is addressing. This was true of all the trees reported here; the authors of the original studies did not attempt to describe the entire duplication history. They simply reported the aspects they considered relevant to their research. In contrast, NOTUNG reports the entire history, including variants, and allows the user to triage the information.

As an example, we show the duplication history generated by NOTUNG for the RXR tree shown in Figure 1:

```
Duplication at 15  Lower bound:  jaw          Upper bound:  pro
Duplication at 14  Lower bound:  jaw          Upper bound:  pro
Duplication at 6   Lower bound:  zebrafish    Upper bound:  jaw
Duplication at 3   Lower bound:  clawed frog  Upper bound:  tet
```

Here “jaw” refers to *jawed_vertebrate*, “pro” to *protostomes* and “tet” to *tetrapods*. Both duplications

14 and 15 occurred after the divergence of protostomes (insects and molluscs) from deuterostomes (fish and tetrapods), which is consistent with Hughes' analysis. It also finds more recent duplications (3 and 6) not discussed by Hughes.

5.2 Alternate Hypotheses for Weak Branches

Alternate hypotheses were evaluated with respect to both cost functions for every branch with a bootstrap value less than 90% in the six rooted trees in Table 1. In all cases, $C_{ub}(\cdot)$ and $C_{dl}(\cdot)$ selected the same optimal rearrangement tree, although they gave different numerical scores. For three input trees, the original tree was unchanged. The PBX tree had no weak edges at the 90% level and the RXR and TEN trees could not be improved through rearrangement.

The other three trees were modified by the rearrangement process. The improvements in score, with respect to both cost functions, are summarized in Table 2. To illustrate NOTUNG'S performance on weak edges, we discuss the rearrangement of one GFT in the data set, HSP70, in detail below. Similar behavior was observed in the remaining two trees, the C family tree and the NOTCH tree, which are simpler than HSP70.

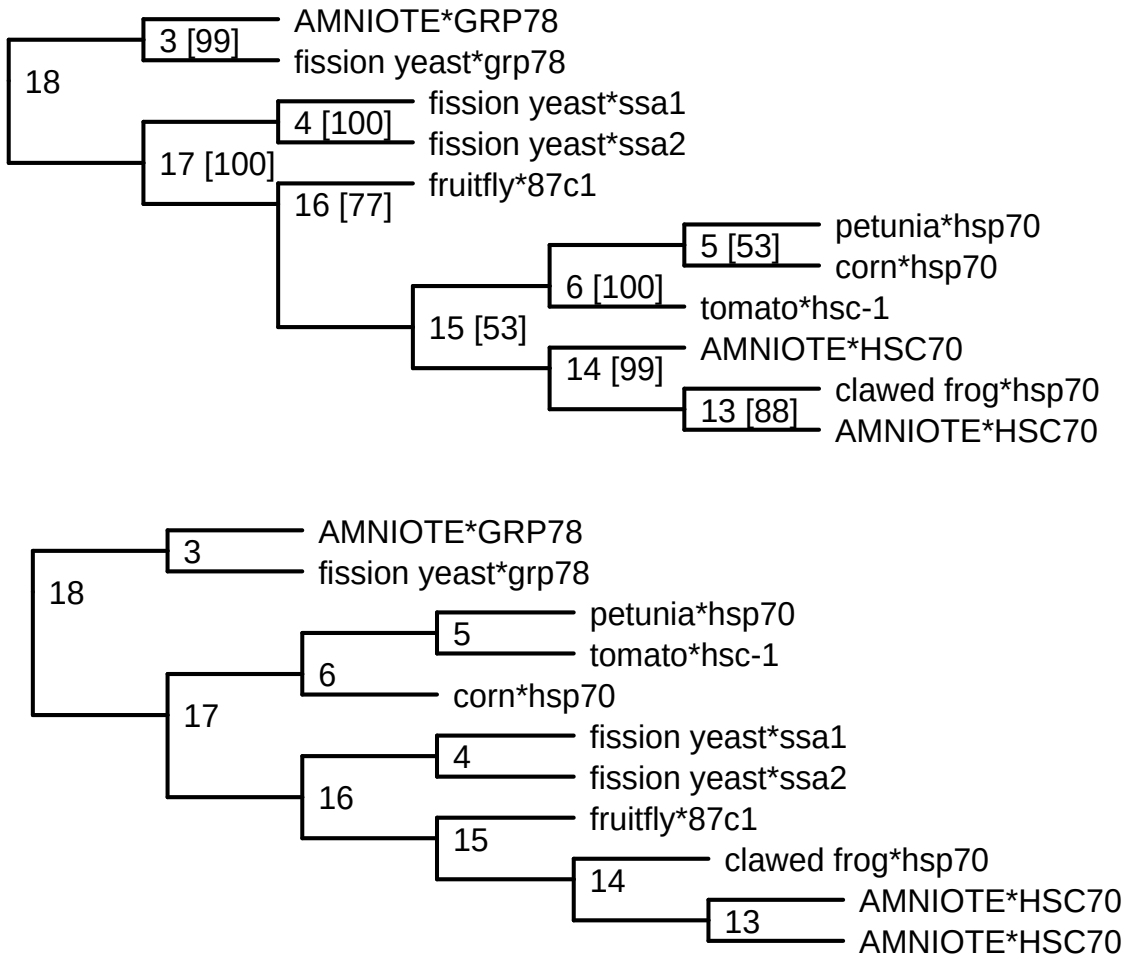

Figure 8: The HSP70 tree before and after NNI rearrangements. The trees have been simplified by compressing clades containing only mammals and birds (AMNIOTE*GRP78m, AMNIOTE*HSP70, AMNIOTE*HSC70). No rearrangements were accepted in these clades. Internal nodes are given numerical labels. In the upper tree, the bootstrap values of edges with bootstrap support above 50 are labeled with square brackets.

The examples in Figure 5 show that some rearrangements correct phylogenetic mistakes, eliminating duplications and resulting in a GFT that is more consistent with the species tree, while others move duplications down in the tree, reducing loss while maintaining the same number of duplications. Both types of rearrangement appear in the HSP70 trees shown in Figure 8. These trees have been simplified for the purposes of exposition. Subtrees containing only birds and mammals have been compressed and are shown in capital letters (e.g., AMNIOTE*HSP70). The upper tree shows the original topology before rearrangements were considered. This tree contains five branches with bootstrap values below the threshold. Two of them are adjacent. Initially, our program inferred a duplication history with nine duplications and eighteen losses:

```
Duplication at 18  Lower bound:  euk
Duplication at 17  Lower bound:  euk
Duplication at 16  Lower bound:  euk
Duplication at 14  Lower bound:  tet           Upper bound:  euk
Duplication at 12  Lower bound:  roh           Upper bound:  tet
Duplication at 10  Lower bound:  human         Upper bound:  roh
Duplication at 8   Lower bound:  amn           Upper bound:  euk
Duplication at 6   Lower bound:  pla           Upper bound:  euk
Duplication at 4   Lower bound:  fission yeast
```

The structure of this tree, (fungi (insects (plants, vertebrates))), is at odds with the structure of the Tree of Life, (plants (fungi, (insects, vertebrates))). The structure within the plant clade also disagrees with the Tree of Life since petunias and tomatoes are more closely related to each other than either is to corn.

In contrast, the topology after rearrangement had three fewer duplication nodes and seven losses,

```
Duplication at 18  Lower bound:  euk
```

```
Duplication at 13  Lower bound:  amn           Upper bound:  tet
Duplication at 12  Lower bound:  roh           Upper bound:  tet
Duplication at 10  Lower bound:  human         Upper bound:  roh
Duplication at 8   Lower bound:  amn           Upper bound:  tet
Duplication at 4   Lower bound:  fission yeast Upper bound:  euk
```

as shown in the lower tree. The removal of duplications from nodes 6, 16 and 17 can be interpreted as correcting errors in the original topology. The original topology implies that an ancestral HSP gene was duplicated twice early in the eukaryote lineage; subsequently each of the four resulting copies survived in only one lineage (fungi, insects, plants and vertebrates, respectively) and was lost in the other three. In view of the low bootstrap support, it seems more plausible that the yeast and fly sequences are placed incorrectly. In the rearranged tree, the branching of plants, yeast and insects is compatible with the Tree of Life. This second hypothesis is more compelling than the original hypothesis of two early duplications followed by massive gene loss. The exchange of the corn and tomato genes to remove the duplication at node 6 also appears to correct an error in the reconstruction of the tree topology. The rearrangement of the frog sequence that led to the replacement of the duplication at node 14 with one at node 13 results from the tendency of the duplication/loss model to favor more recent duplications and is more controversial. It is open to interpretation whether a duplication in the amniote lineage is more or less likely than a duplication before the divergence of amphibians followed by loss of one copy.

This example shows that rearrangement can result in substantially different hypotheses. The number of duplication nodes in the rearranged HSP tree decreased from nine to six. As this illustrates, although it is possible to pick out individual rearrangements of low confidence branches by eye, when a tree contains many weak branches it is helpful to have a tool to integrate all the alternate hypotheses automatically.

5.3 Unrooted Trees

We tested NOTUNG on the seven unrooted trees in Table 1. For each tree, NOTUNG computed the duplication history for every root and ranked them according to both cost functions. We compared these rankings with the rootings favored by the authors. Although possible rootings are rarely, if ever, mentioned explicitly by the author's whose trees we tested, they frequently imply that only a subset of the possible rootings lead to plausible hypotheses. Consider, for example, the TCF family tree, shown in Figure 9. In their analysis, Ruvinsky and Silver (Ruvinsky and Silver 1997) state that "it is difficult to conclude whether the split between the TCF1 and TCF2 subfamilies occurred before or after the separation between fish and tetrapods," but "in any case, divergence between the two sub families has taken place prior to the amniote-amphibian separation." These conclusions are consistent with a rooting on the bold edges in Figure 9 and no others.

For each tree, we partitioned the set of edges into plausible and implausible rootings from the analysis presented by the original authors. We executed NOTUNG on all trees and compared the rankings induced by each cost function with the author's rankings. The two cost functions always agreed on the ranking of the trees with the best scores, ranging from an identical ranking of the top nine rootings for the EGR family to identical rankings of all rootings for ANK, LHX and VMAT. When NOTUNG 's rankings were compared with those of the original authors, Noting ranked all plausible rootings ranked above all implausible rootings for five out of the seven trees. For the remaining two trees, the costs of all implausible rootings were greater than or equal to the costs of all plausible rootings. For one of these, the PSMB tree, the set of highest ranked edges is a superset of the rootings deemed plausible by Hughes. One of these edges has weak bootstrap support. A rearrangement around this edge improves the score of the tree with respect to both $C_{dl}(\cdot)$ and $C_{ub}(\cdot)$. When the alternate rootings of this rearranged tree were rescored, the set of lowest cost rootings exactly agreed with Hughes' analysis. In the other case, the CRYB tree, there were eight top-ranked

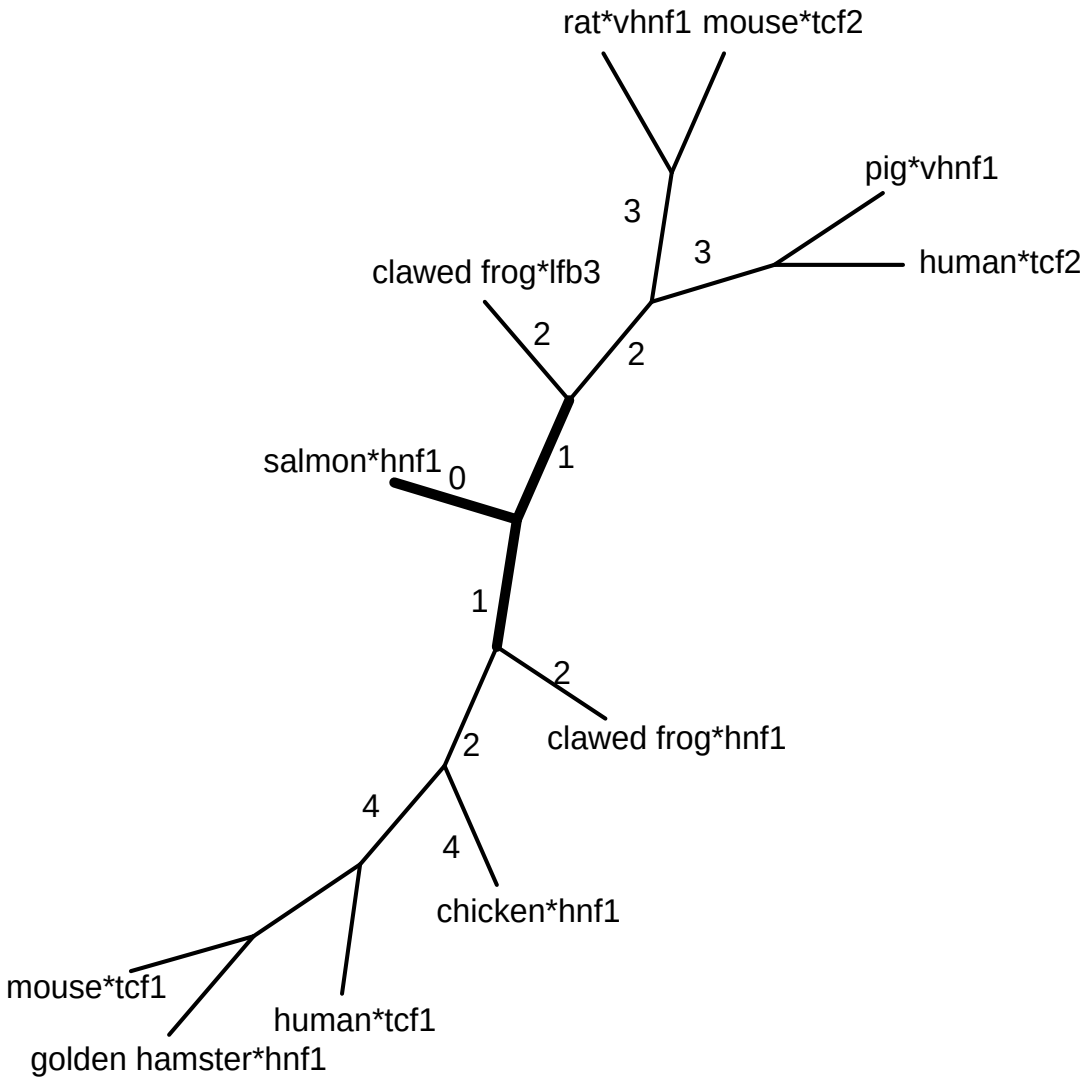

Figure 9: An unrooted tree for the TCF family (Ruvinsky and Silver 1997). Each edge, e, is labeled with the ranking, with respect to both $C_{dl}(\cdot)$ and $C_{ub}(\cdot)$, of the tree rooted at e. Edges in bold correspond to rootings supported by the analysis in (Ruvinsky and Silver 1997).

rootings of equal cost, while the authors' analysis implied that only one rooting is possible. The duplication histories (i.e., the set of duplication nodes with time ranges) were identical for the eight edges. Only the ordering of the duplication nodes differed. This suggests either that the authors did not consider all alternate scenarios, possibly missing something of interest, or that they had additional information about the gene family, such as the biochemical properties or functional roles of the proteins, that allowed them to rule out other rootings.

Within the set of plausible rootings, the ordering of scores does not always agree with the biologists' assessments. In contrast to the analysis of Ruvinsky and Silver, who ranked the three best edges equally, the edge adjacent to the fish sequence in Figure 9 is ranked higher than the other two because this rooting requires one fewer gene loss.

6 Discussion

In this study, we analyzed every non-pathological tree in three recent gene duplication studies (Hughes 1998; Pebusque et al. 1998; Ruvinsky and Silver 1997). The duplication histories generated for rooted trees were consistent with the analyses of the authors of the original papers for all trees considered. For unrooted trees, alternate histories were computed for each rooting and ranked according to two different cost functions based on duplication and loss. One of these is based on a model of parsimonious gene loss while the other is an upper bound on the maximum number of gene losses that occurred. Although the rankings for the two cost functions were different, for top ranking rootings the cost functions agreed for every tree in the test set. In particular, they agreed on the hypotheses most likely to be of interest to the user. Both rankings correctly identified unlikely hypotheses, providing the user with a way to control the quantity of output to be reviewed. When used to select alternate hypotheses for trees with weak edges, both cost functions were effective in correcting errors in the duplication history stemming from errors in the original tree topology.

The fact that the cost functions agree on both unrooted trees and rearrangements, suggest that the problem is robust with respect to the duplication/loss model. Since the cost functions represent models that are opposite extremes of gene loss, our results imply that inferring duplication histories is not highly sensitive to the choice of model and we can have confidence in an exploratory analysis tool based on such scoring functions. This approach does present some limitations. Because both cost functions count gene losses, they favor hypotheses with more recent duplications. As a result, in addition to correcting phylogenetic errors when evaluating alternatives for weak branches, in some cases they select more controversial alternatives. By combining duplication and loss with sequence information, this approach offers a richer source of information for reconstructing gene duplication histories than sequence information alone. Nevertheless, there are other sources biological information (e.g., biochemical properties and intron/exon structure) that can

guide analysis that are not incorporated in this approach. Another limitation of this approach is that it is difficult to distinguish true gene loss from genes that have not yet been sequenced.

This paper suggests several open problems for future research. Currently, there is no known polynomial algorithm for finding the optimal rearrangement tree with respect to either cost function. Nor has this optimization problem been shown to be NP-hard. The sensitivity of the cost function, $C_{dl}(\cdot)$, to the relative importance of gene duplications and gene losses (i.e., the values of c_λ and c_δ) has not been investigated. Finally, the more general problem of phylogeny reconstruction using a cost function that combines sequence comparison, gene duplication and gene loss is an outstanding challenge.

Currently, there is a great deal of interest in using gene duplications to study the role of whole genome duplications in genome evolution (Skrabanek and Wolfe 1998; Wolfe and Shields 1997). This will require dating all paralogs in a genome. In its current form, NOTUNG can be used to estimate duplication dates of rooted GFT's automatically. With more reliable evaluation methods, NOTUNG can also be adapted to the automatic analysis of unrooted trees and rearrangements of trees with weak edges.

Acknowledgements

The authors wish to thank Jim Brown, Ilya Ruvinsky, Lee Silver and Mona Singh for helpful discussions. Particular thanks go to an anonymous reviewer for detailed comments and for suggesting a simpler formulation of the $C_{dl}(\cdot)$ cost function. Species trees in Newick format were downloaded from the Ribosomal Database Project (Maidak et al. 1999). Several figures in this paper were drawn with Treetool, an interactive tree-plotting program written by Mike Maciukenas, for the Ribosomal Database Project (Maidak et al. 1999).

References

Bender, M. and Farach-Colton, M. 2000. Least common ancestors revisited. *Latin '00* .

Chen, K., Durand, D., and Farach-Colton, M. 2000. Notung: Dating gene duplications using gene family trees. *In* RECOMB2000, Fourth Annual International Conference on Computational Molecular Biology.

Efron, B. and Gong, G. 1983. A leisurely look at the bootstrap, jackknife, and cross-validation. *The American Statistician* 37:36–48.

Endo, T., Imanishi, T., Gojobori, T., and Inoko, H. 1997. Evolutionary significance of intra-genome duplications on human chromosomes. *Gene* 205:19–27.

Eulenstein, O., Mirkin, B., and Vingron, M. 1996. Comparison of a annotating duplication, tree mapping, and copying as methods to compare gene trees with species trees. *Mathematical Hierarchies and Biology, DIMACS Series in Discrete Mathematics and Theoretical Computer Science* 37:71–93.

Eulenstein, O., Mirkin, B., and Vingron, M. 1998. Duplication-based measures of difference between gene and species trees. *Journal of Computational Biology* 5:135–148.

Goffeau, A., Barrell, B. G., Bussey, H., Davis, R. W., Dujon, B., Feldmann, H., Galibert, F., Hoheisel, J. D., Jacq, C., Johnston, M., Louis, E. J., Mewes, H. W., Murakami, Y., Philippsen, P., Tettelin, H., and Oliver, S. G. 1996. Life with 6000 genes. *Science* 274:563–7.

Goodman, M., Czelusniak, J., Moore, G., Romero-Herrera, A., and Matsuda, G. 1979. Fitting the gene lineage into its species lineage, a parsimony strategy illustrated by cladograms constructed from globin sequences. *Syst Zool* .

Graur, D., Duret, L., and Gouy, M. 1996. Phylogenetic position of the order lagomorpha (rabbits, hares and allies). *Nature* .

Guigo, R., Muchnik, I., and Smith, T. 1996. Reconstruction of ancient phylogenies. *Molecular Phylogenet-*

ics and Evolution 6:189–213.

Hallett, M. T. and Lagergren, J. 2000. New algorithms for the duplication-loss model. *In* RECOMB2000, Fourth Annual International Conference on Computational Molecular Biology.

Hughes, A. L. 1998. Phylogenetic tests of the hypothesis of block duplication of homologous genes on human chromosomes 6, 9, and 1. *MBE* 15:854–70.

Hughes, A. L. 1999. Phylogenies of developmentally important proteins do not support the hypothesis of two rounds of genome duplication early in vertebrate history. *JME* 48:565–76.

JáJá, J. 1991. Introduction to Parallel Algorithms. Addison-Wesley, Reading, MA.

Kasahara, M. 1997. New insights into the genomic organization and origin of the major histocompatibility complex: role of chromosomal (genome) duplication in the emergence of the adaptive immune system. *Hereditas* 127:59–65.

King, R. C. and Stansfield, W. D. 1990. A Dictionary of Genetics. Oxford University Press.

Ma, B., Li, M., and Zhang, L. 1998. From gene trees to species trees. *In* RECOMB'98, Second Annual International Conference on Computational Molecular Biology, pp. 182–191.

Ma, B., Li, M., and Zhang, L. 2000. From gene trees to species trees. *SIAM J. on Comput.* .

Maidak, B. L., Cole, J. R., Parker, C. T., Garrity, G. M., Larsen, N., Li, B., Lilburn, T. G., McCaughey, M. J., Olsen, G. J., Overbeek, R., Pramanik, S., Schmidt, T. M., Tiedje, J. M., and Woese, C. R. 1999. A new version of the rdp (Ribosomal Database Project). *Nucleic Acids Res* 29:171–3.

Martin, A. P. 1999. Increasing genomic complexity by gene duplication and the origin of vertebrates. *American Naturalist* .

Mirkin, B., Muchnik, I., and Smith, T. 1995. A biologically consistent model for comparing molecular phylogenies. *Journal of Computational Biology* 2:493–507.

Ohno, S. 1970. Evolution by Gene Duplication. Springer-Verlag.

Page, R. 1994. Maps between trees and cladistic analysis of historical associations among genes, organisms and areas. *Syst Zool* .

Page, R. and Charleston, M. 1996. Reconciled trees and incongruent gene and species trees. *Mathematical Heirarchies and Biology, DIMACS Series in Discrete Mathematics and Theoretical Computer Science* 37:57–70.

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

Pebusque, M.-J., Coulier, F., Birnbaum, D., and Pontarotti, P. 1998. Ancient large-scale genome duplications: phylogenetic and linkage analyses shed light on chordate genome evolution. *MBE* 15:1145–59.

Ruvinsky, I. and Silver, L. M. 1997. Newly indentified paralogous groups on mouse chromosomes 5 and 11 reveal the age of a t-box cluster duplication. *Genomics* 40:262–266.

Silver, L. M. 1995. Mouse Genetics. Oxford University Press.

Skrabanek, L. and Wolfe, K. 1998. Eukaryote genome duplication - where's the evidence? *Curr Opin Genet Dev* 8:559–565.

Stege, U. 1999. Gene trees and species trees: The gene-duplication problem is fixed-parameter tractable. *In* Proceedings of the 6th International Workshop on Algorithms and Data Structures (WADS'99).

Swofford, D. L., Olsen, G. J., Waddell, P. J., and Hillis, D. M. 1996. Phylogeny inference, pp. 407–514. *In* D. M. Hillis, C. Moritz, and B. K. Mable (eds.), Molecular Systematics. Sinauer Associates Inc., Sunderland, MA.

Wolfe, K. H. and Shields, D. C. 1997. Molecular evidence for an ancient duplication of the entire yeast genome. *Nature* 387:708–713.

Zhang, L. 1997. On a Mirkin-Muchnik-Smith conjecture for comparing molecular phylogenies. *Journal of Computational Biology* 4:177–188.