



## A simple algorithm to infer gene duplication and speciation events on a gene tree

Christian M. Zmasek and Sean R. Eddy

Howard Hughes Medical Institute, Department of Genetics, Washington University  
School of Medicine, St Louis, MO 63110, USA

Received on January 24, 2001; revised on April 5, 2001; accepted on April 6, 2001

### ABSTRACT

**Motivation:** When analyzing protein sequences using sequence similarity searches, orthologous sequences (that diverged by speciation) are more reliable predictors of a new protein's function than paralogous sequences (that diverged by gene duplication), because duplication enables functional diversification. The utility of phylogenetic information in high-throughput genome annotation ('phylogenomics') is widely recognized, but existing approaches are either manual or indirect (e.g. not based on phylogenetic trees). Our goal is to automate phylogenomics using explicit phylogenetic inference. A necessary component is an algorithm to infer speciation and duplication events in a given gene tree.

**Results:** We give an algorithm to infer speciation and duplication events on a gene tree by comparison to a trusted species tree. This algorithm has a worst-case running time of  $O(n^2)$  which is inferior to two previous algorithms that are  $\sim O(n)$  for a gene tree of  $n$  sequences. However, our algorithm is extremely simple, and its asymptotic worst case behavior is only realized on pathological data sets. We show empirically, using 1750 gene trees constructed from the Pfam protein family database, that it appears to be a practical (and often superior) algorithm for analyzing real gene trees.

**Availability:** <http://www.genetics.wustl.edu/eddy/forester>

**Contact:** [zmasek@genetics.wustl.edu](mailto:zmasek@genetics.wustl.edu);  
[eddy@genetics.wustl.edu](mailto:eddy@genetics.wustl.edu)

### INTRODUCTION

Automated sequence function prediction becomes a necessity due to the enormous amount of sequence data currently produced by the various genome projects. The fact that many proteins belong to large superfamilies that consist of subfamilies with different biological functions complicates such efforts.

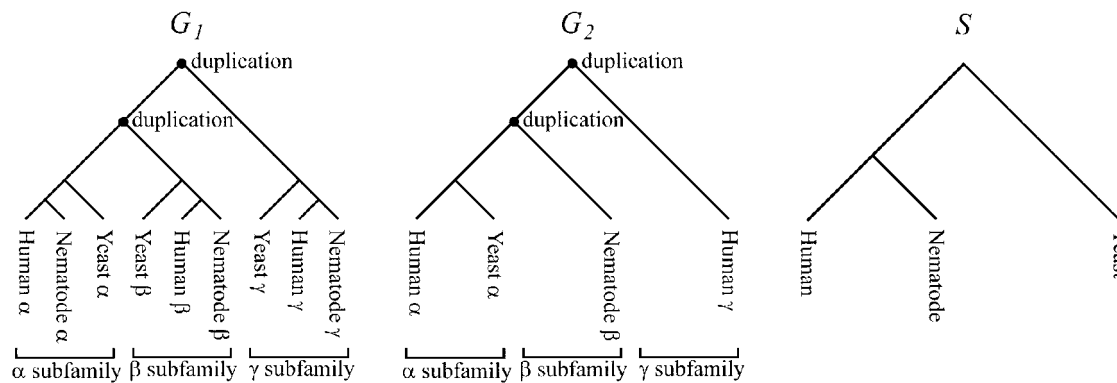
Usually, automated sequence function prediction is done using methods based on pairwise sequence similarity, such as BLAST (Altschul *et al.*, 1990). Annotating a new sequence by transferring annotation from its best BLAST

hits tends to classify novel sequences too aggressively. Without careful human intervention, it is impossible to detect when a new sequence is not as similar to known homologs as it should be, and it in fact represents the first member of a novel functional subfamily in a larger superfamily—often an extremely interesting result.

In contrast, analyses using profile search algorithms such as HMMER (Eddy, 2000) and protein family databases such as Pfam (Bateman *et al.*, 2000) and InterPro (Apweiler *et al.*, 2000), classify sequences too conservatively. They recognize that a new sequence belongs to a certain family, but do not subclassify the sequence.

Profile algorithms can be used to align the novel sequence to a curated alignment of the known family members. A human annotator can use this multiple alignment as input for a phylogenetic tree analysis, and from the placement of the new sequence in the tree of known sequences can infer a more specific function. This approach was called 'phylogenomics' by Eisen (1998). This procedure is different from schemes such as the COG database (Tatusov *et al.*, 2001) in that it directly uses phylogenetic trees, whereas COG clusters sequences based on evolutionary relationships indirectly inferred from sequence similarities.

It is impossible to automate this process fully, because it is impossible to precisely define what 'protein function' means. However, a principle of phylogenomics is that orthologous sequences (that diverged by speciation) are more likely to conserve protein function than paralogous sequences (that diverged by gene duplication). Orthology and paralogy are precisely defined and can be inferred from gene and species trees. One simple example of a phylogenomics approach that is reasonable and automatable could thus be stated as follows. If a novel sequence has orthologs, functional annotation can be transferred from them (as in best BLAST analysis); if there are no orthologs, the sequence is classified as just a family member (as in Pfam/InterPro analysis) and flagged as possibly the first representative of a novel subfamily. Other, more sophisticated analyses could be devised.



**Fig. 1.** Gene trees and species trees.  $G_1$  and  $G_2$  are gene trees,  $S$  is a species tree. Internal tree nodes representing gene duplications are labeled as such, other internal nodes represent speciations. The sequence family in tree  $G_1$  is comprised of three functional subfamilies:  $\alpha$ ,  $\beta$  and  $\gamma$ . The two duplications in  $G_1$  can be inferred directly from the redundancy of species names.  $G_2$  is a tree of the same family as  $G_1$ . In contrast to  $G_1$ , some sequences are not present in  $G_2$ , either due to gene loss or incomplete sampling. The second duplication in  $G_2$  can only be inferred by comparing it to the species tree  $S$  and recognizing the anomaly of placing the human gene closer to yeast than to nematodes.

At the core of such approaches stands therefore the distinction between orthologs and paralogs, and hence the ability to discriminate between duplication and speciation events on a gene tree.

Algorithms to distinguish between duplications and speciations have been employed previously in calculating the dissimilarity between gene trees and species trees, and in inferring parsimonious species trees from gene trees by minimizing the number of duplications and gene losses that must be invoked to reconcile a given gene sequence tree with the inferred species tree (Eulenstein and Vingron, 1995; Goodman *et al.*, 1979; Guigo *et al.*, 1996; Mirkin *et al.*, 1995; Page and Charleston, 1997; Zhang, 1997). Brute force algorithms to solve this problem can have unfavorable  $O(n^3)$  running times. Two known algorithms solve the problem efficiently with excellent worst-case running times of  $\sim O(n)$  for a gene tree of  $n$  sequences (Zhang, 1997; Eulenstein, 1998; the Eulenstein algorithm is implemented in the program 'GeneTree', Page, 1998) but both algorithms are somewhat complex. We describe here a very simple algorithm that appears to solve the problem even more efficiently on realistic data sets, though it has an asymptotic worst-case running time that is less favorable.

## ALGORITHM

A gene tree  $G$  and the species tree  $S$  of the species harboring the genes of  $G$  do not necessarily have to exhibit the same topology (Page and Holmes, 1998). Gene duplication, gene loss, and horizontal genetic transfer are some of the forces causing inconsistencies. Gene duplication can be trivially inferred when a species contains two or more homologs belonging to the same gene family (tree  $G_1$  in

Figure 1). However, due to gene loss or incomplete sampling of genes in partially sequenced genomes, not all duplications are detectable by simple redundancy in a gene tree (tree  $G_2$  in Figure 1). Reliable assignment of nodes in the gene tree as either duplication events or speciation events requires comparison to the phylogenetic tree of the species (tree  $S$  in Figure 1).

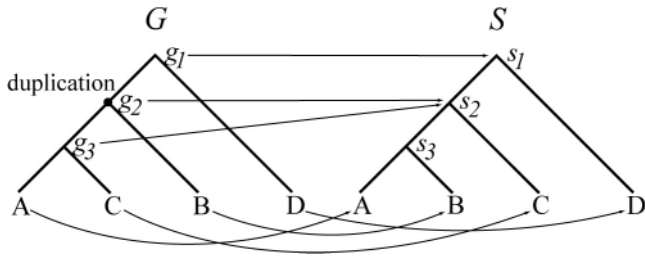
First let us define how we recognize that a node in a gene tree  $G$  should be assigned as a duplication, given species tree  $S$ . We use a mapping function  $M$  which was first introduced by Goodman *et al.* (1979) and used elsewhere (Chen *et al.*, 2000; Eulenstein *et al.*, 1998; Eulenstein and Vingron, 1995; Guigo *et al.*, 1996; Mirkin *et al.*, 1995; Page, 1994; Page and Charleston, 1997; Zhang, 1997):

**DEFINITION 1.** Let  $G$  be the set of nodes in a rooted binary gene tree and  $S$  the set of nodes in a rooted binary species tree. For any node  $g \in G$ , let  $\gamma(g)$  be the set of species in which occur the extant genes descendant from  $g$ . For any node  $s \in S$ , let  $\sigma(s)$  be the set of species in the external nodes descendant from  $s$ . For any  $g \in G$ , let  $M(g) \in S$  be the smallest (lowest) node in  $S$  satisfying  $\gamma(g) \subseteq \sigma(M(g))$ . That is,  $M(g)$  points to the ancestral species in  $S$  that (we infer) harbored ancestral gene  $g$ .

Duplications are then defined using  $M(g)$  in Goodman *et al.* (1979) and formally in Guigo *et al.* (1996) and Page and Charleston (1997) as follows:

**DEFINITION 2.** Let  $g_1$  and  $g_2$  be the two child nodes of an internal node  $g$  of a rooted binary gene tree  $G$ . Node  $g$  is a duplication if and only if  $M(g) = M(g_1)$  or  $M(g) = M(g_2)$ .

An example is shown in Figure 2. This approach makes a parsimony assumption. It postulates the minimal number



**Fig. 2.** The mapping function  $M$  and the definition of a duplication.  $M$  is symbolized by arrows originating at nodes of the gene tree  $G$  and pointing to nodes in the species tree  $S$ . Letters A to D represent species names. As an example, the mapping for  $g_3$  is computed as follows. According to Definition 1,  $\gamma(g_3) = \{A, C\}$ , hence  $M(g_3) = s_2$  since the smallest node  $s \in S$  satisfying  $\gamma(g) \subseteq \sigma(s)$  is  $s_2$  for which  $\sigma(s_2) = \{A, B, C\}$ . Each external node of  $G$  maps to the external node in  $S$  that is associated with the same species name.  $g_2$  is a duplication according to Definition 2, since it and its child  $g_3$  maps to the same node  $s_2$ .

of duplications necessary to reconcile the gene tree with the species tree, and it places those duplications as close to the external nodes as possible. It minimizes the number of unobserved genes—due to gene loss or incomplete sampling—that need to be invoked.

Given the mapping function  $M(g)$ , using Definition 2 to assign duplications requires only a linear time,  $O(n)$  traversal of a gene tree  $G$  for  $n$  genes. What about calculating  $M(g)$ ? To our knowledge, Page was the first to implement an algorithm for this problem (Page, 1994), but the description given is a brute force approach (for each node  $g$  in  $G$ , visit each node  $s$  in  $S$ , compile the sets  $\gamma(g)$  and  $\sigma(s)$ , and compare them). This algorithm has a running time of  $O(n^3)$ , if the number of species in  $S$  is  $O(n)$ . To speed this up, observe that  $M(g)$  cannot be lower than  $M(g_1)$  or  $M(g_2)$  in  $S$ . Furthermore, observe that  $M(g)$  must in fact be the Last Common Ancestor (LCA) of  $M(g_1)$  and  $M(g_2)$ . Therefore if we are careful to traverse  $G$  in the right direction, we can assign  $M(g)$  recursively without ever having to explicitly compile or compare the lists  $\gamma(g)$  and  $\sigma(s)$ , and without having to traverse all of  $S$  for each node  $g$ . This recursive algorithm goes as follows:

**Input:** Rooted binary gene tree  $G$ , rooted binary species tree  $S$  of all species in  $G$ .

**Output:**  $G$  with ‘duplication’ or ‘speciation’ assigned to each of its internal nodes.

#### Initialization

Number nodes of  $S$  in preorder traversal (root = 1, child nodes always larger than parent node);

For each external node  $g$  of  $G$ , set  $M(g)$  to the

number of the external node in  $S$  with the matching species name;

#### Recursion

Visit each internal node  $g$  of  $G$  in postorder

traversal (from external nodes upwards to root):

set  $a = M(g_1)$ ; [ $g_1$  is child 1 of the current node  $g$ ]

set  $b = M(g_2)$ ; [ $g_2$  is child 2 of the current node  $g$ ]

while ( $a \neq b$ ):

if  $a > b$ :

set  $a = \text{parent of node } a$ ;

else:

set  $b = \text{parent of node } b$ ;

set  $M(g) = a$ ;

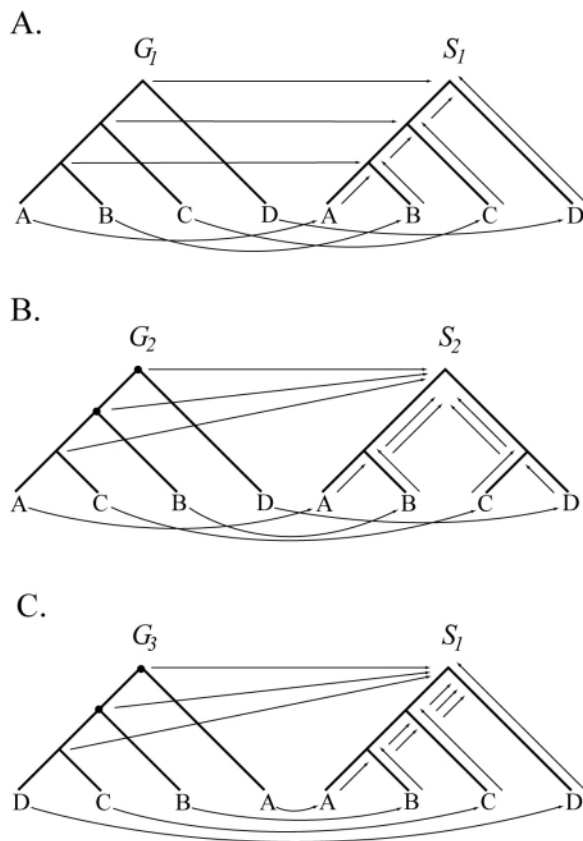
if ( $M(g) == M(g_1)$ ) or ( $M(g) == M(g_2)$ ):

$g$  is a duplication;

else:

$g$  is a speciation.

A sketch of the running time analysis of this algorithm is as follows. Initializing  $M(g)$  for the external nodes of  $G$  is  $O(n)$  if species names are looked up in a hash table (Cormen *et al.*, 1990). Initializing the numbering of  $S$  is  $O(n)$  (again assuming that the number of nodes in  $S$  scales linearly with the number of nodes in  $G$ ;  $S$  can be smaller than  $G$  but not larger). Thus initialization is  $O(n)$  and will not be the rate determining step. In the recursion, we visit each of the  $n - 1$  internal nodes in  $G$  individually, and at each node we find the LCA of  $M(g_1)$  and  $M(g_2)$  simply by brute force, by climbing the tree from both points until we meet. The computational cost of finding LCAs in this manner depends on the topology of  $G$  and  $S$ . In the best case,  $G$  has no duplications and the topology of  $G$  and  $S$  are the same; each LCA determination costs  $O(1)$ , no node in  $S$  will be reached more than twice in the whole algorithm, and the overall running time is therefore  $O(n)$  (Figure 3A). In a pathological bad case, if  $M(g)$  for all internal nodes in  $G$  pointed to the root of the species tree (itself a special case of the unusual situation in which all parent nodes of all internal nodes are gene duplication events), and nonetheless no more than one gene in  $G$  is found in each species, each LCA determination would require climbing the entire height of tree  $S$ , which for a balanced binary tree would be  $\log n$ , giving an overall running time of  $O(n \log n)$  (Figure 3B). Finally, in the pathological worst case, not only would each LCA require climbing all of the height of  $S$ , but  $S$  could also be a maximally unbalanced tree (a tree in which each internal node has at least one external child, also called a ‘pectinate’ tree) with a height of  $n$ , giving an overall running time of  $O(n^2)$  (Figure 3C). The space complexity of the algorithm is  $O(n)$ , since only the two trees and a constant number of auxiliary variables need to be stored.



**Fig. 3.** The number of duplications and the topology of the species tree influence the time complexity of our algorithm.  $G_1$  to  $G_3$  are gene trees,  $S_1$  and  $S_2$  are species trees.  $M$  is symbolized by arrows originating at nodes of the gene tree and pointing to nodes in the species tree. Letters A to D represent species names. Circled nodes are duplications. Arrows inside the species trees symbolize the movement of variables *a* and *b* (see text).

Algorithms with more efficient asymptotic bounds on running time have been published. The closest to ours are those of Zhang (1997) and Chen *et al.* (2000). Both observe that LCA calculations can be done in  $O(1)$  time, for instance using the LCA algorithms described by Schieber and Vishkin (1988) or by Jájá (1991). The trick is that the LCA of any two nodes on a complete binary tree can be calculated by direct arithmetic. The tree  $S$  (which in general is not a complete binary tree) is therefore preprocessed in such a way that the nodes of  $S$  are associated with nodes in a complete binary tree; this preprocessing takes  $O(n)$  time. A quite different algorithm, developed by Eulenstein (1998), calculates  $M$  in  $O(n\alpha(n, n))$  time, where  $\alpha(n, n)$  is the very slowly growing inverse of Ackermann's function (Cormen *et al.*, 1990). This algorithm visits each node of the species tree  $S$  and in the process calculates  $M$  for each internal node of

the gene tree, using a data structure similar to a disjoint-set forest (Cormen *et al.*, 1990).

Both kinds of algorithm, though asymptotically more efficient than ours, require relatively complex preprocessing. We reasoned that since our algorithm has so few steps, we were likely to have a better constant factor than both. Furthermore, our intuition was that the worst case bounds of our algorithm were pathological and would never be realized on realistic data sets. Eulenstein comments that his algorithm has a lower constant factor than Zhang's. We decided to implement both our algorithm and Eulenstein's, and compare their performance on real data.

## IMPLEMENTATION

Both algorithms were implemented in Java. The Java classes are named SDI for 'Speciation vs Duplication Inference' and are part of our FORESTER classes for working with phylogenetic trees. FORESTER including SDI is freely available at <http://www.genetics.wustl.edu/eddy/forester/>. It should run on every platform with a Java 1.2 JDK.

A preprocessing step deletes external nodes in  $S$  that have no genes in  $G$ , allowing a single trusted species tree to be used for all gene trees.

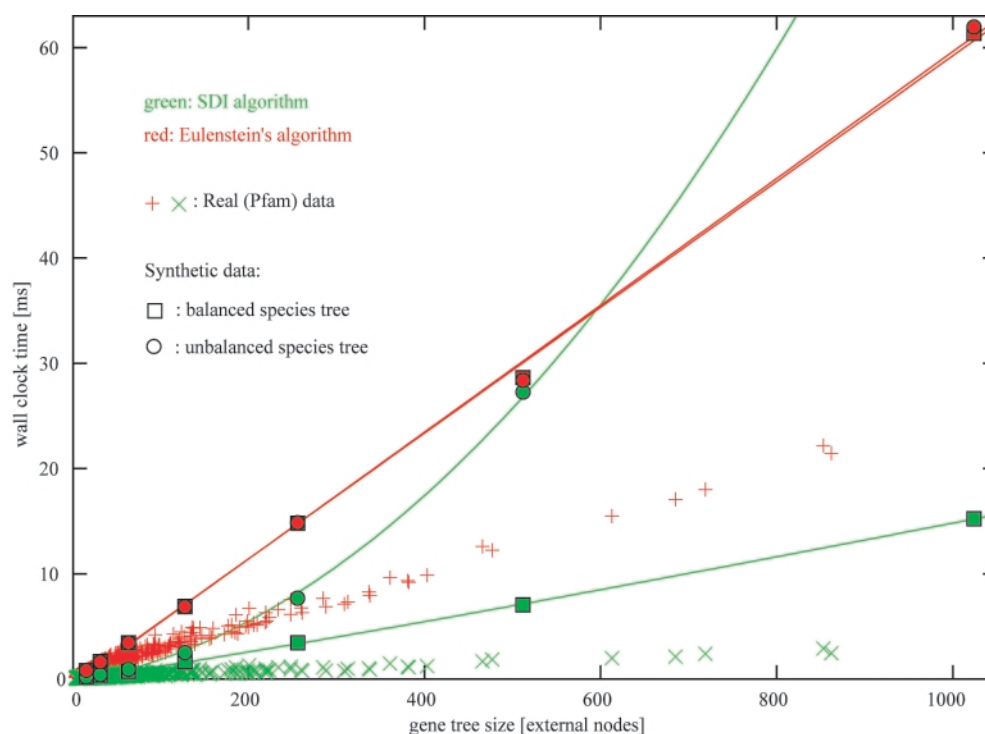
All timings reported are the average of three runs on a single processor 500 MHz Pentium III system running Red Hat Linux 6.0 and Sun Microsystems' Java 1.2 SDK for Linux.

## RESULTS

We first timed the two implementations on synthetic data sets that would exercise the worst-case behavior of our algorithm. We synthesized gene trees with  $n$  genes, for a range of values of  $n$ , where  $M(g)$  for every internal node would map to the root of the corresponding species tree with  $n$  species (e.g. the situations in Figure 3B and C). Plots of wall clock time versus  $n$  are shown in Figure 4. For a balanced species tree, both algorithms have running times that scale nearly linearly in tree size (our  $O(n \log n)$  is not appreciably different from linear at first glance), and our algorithm exhibits a lower constant than our implementation of the Eulenstein algorithm. For a maximally unbalanced species tree, we confirm our algorithm worst case  $O(n^2)$  behavior, but because of our lower overhead, SDI is still more efficient for smaller trees. Over about  $n = 550$  genes and species, our implementation of Eulenstein's algorithm outperforms SDI. If only the actual calculation of  $M(g)$  is compared (excluding all preprocessing and initialization steps), Eulenstein's algorithm outperforms SDI for  $n$  larger than about 200 taxa (data not shown).

We then tested both implementations on real data to empirically determine their average-case behavior. We





**Fig. 4.** Timing benchmarks on real trees to determine average-case behavior, and synthetic trees that exercise our algorithm's worst case behavior. For the synthetic trees, every internal node of the gene tree maps to the root of the corresponding species tree and each gene tree has the same size as the corresponding species tree. Each synthetic data point is the average of three measurements. Curves were fitted using GNUPLOT's nonlinear least squares fitting mechanism (Marquardt–Levenberg algorithm). Real trees are from Pfam alignments and were created as described in the text. In the case of real trees, the species trees usually have fewer taxa than gene trees (each species may contain more than one paralogous gene)—hence the smaller times relative to synthetic data tests. Each Pfam data point is the average of 100 measurements.

obtained 2478 multiple sequence alignments from the 'full' alignments (as opposed to the smaller 'seed' alignments) in the protein family database Pfam (release 5.5; Bateman *et al.*, 2000).

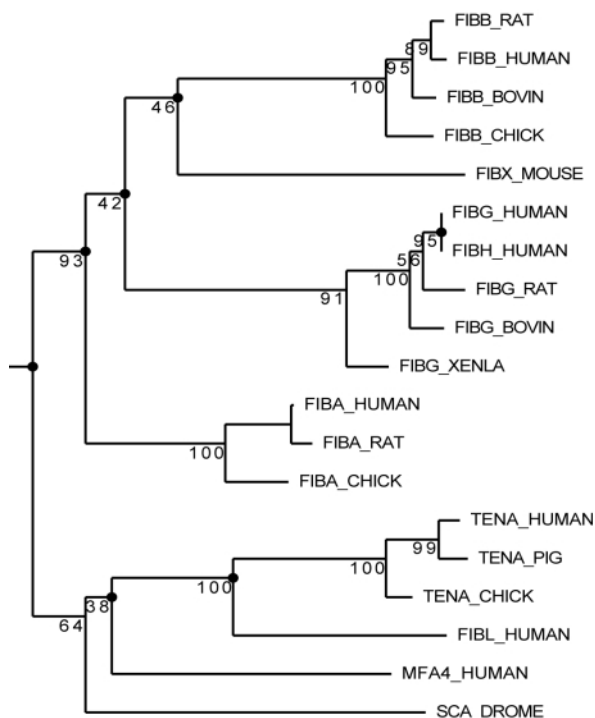
Gene trees were constructed from these alignments as follows. All sequences not originating from the curated SWISS-PROT database (Bairoch and Apweiler, 2000) and not from species in our species tree (see below) were removed from the alignments. Alignments with fewer than four or more than 1000 sequences were discarded, leaving 1750 alignments. Columns containing one or more gap symbols were removed from the alignment if the resulting alignment after this filtering was at least 100 amino acids in length. Pairwise distances were calculated based on the Dayhoff PAM matrix (Dayhoff *et al.*, 1978) using the program PROTDIST from Felsenstein's (1993) PHYLIP package. A neighbor-joining tree (Saitou and Nei, 1987) was constructed using the program NEIGHBOR from the PHYLIP package. Roots were placed by the midpoint rooting method (Swofford *et al.*, 1996).

A single master species tree was compiled manually, containing 200 of the most commonly encountered species

in Pfam. The topology of this species tree is based on the taxonomy database at NCBI (<http://www.ncbi.nlm.nih.gov/Taxonomy/tax.html/>), the Tree of Life project (Maddison and Maddison, at <http://phylogeny.arizona.edu/tree/phylogeny.html>), Barns *et al.* (1996), and Aguinaldo *et al.* (1997). This tree is available at <http://www.genetics.wustl.edu/eddy/forester/>.

The individual running times of the SDI algorithm and of the Eulenstein algorithm for each of these 1750 trees are shown in Figure 4. These data imply that the average case behavior of our algorithm on real data sets is approximately  $O(n)$ , and its worst case behavior is not realized.

As an example of the results from such an analysis, and how they might be useful in sequence annotation, the gene tree for the fibrinogen beta and gamma chain Pfam family (Pfam accession number: PF00147) is presented in Figure 5. The fibrinogen sequence family contains fibrinogen alpha, beta and gamma chains (sequences with FIBA, FIBB, FIBG prefixes) which together form the fibrinogen hexamer (Stryer, 1995). Each chain type appears on the tree as a paralogous subtree. A special case



**Fig. 5.** A gene tree for the fibrinogen beta and gamma chain Pfam family. Circled internal nodes represent gene duplication events inferred by SDI. The suffix of each SWISS-PROT sequence name indicates the species (BOVIN, *Bos taurus*; CHICK, *Gallus gallus*; DROME, *Drosophila melanogaster*; HUMAN, *Homo sapiens*; PIG, *Sus scrofa*; RAT, *Rattus norvegicus*; XENLA, *Xenopus laevis*). Bootstrap values were calculated from 100 replicates and are shown as numbers below the corresponding branch. The tree was rooted by the midpoint rooting method. The figure was produced with our tree display tool ATV (Zmasek and Eddy, 2001, available at <http://www.genetics.wustl.edu/eddy/atv/>).

is FIBH\_HUMAN (fibrinogen gamma-B chain) which appears to be the result of alternative splicing of the human gamma chain gene (Fornace *et al.*, 1984). In addition, the fibrinogen family also contains various proteins probably involved in adhesion, which share the fibrinogen-like domain with the fibrinogen sequences (e.g. Jones *et al.*, 1988; Baker *et al.*, 1990), such as tenascins (sequences with TENA prefixes). Interestingly—FIBX\_MOUSE (also named FGL2\_MOUSE), a mouse enzyme with prothrombinase activity (conversion of prothrombin into thrombin) is similar to fibrinogen beta and gamma chains (Parr *et al.*, 1995). Thrombin is an enzyme responsible for cleaving fibrinogen into monomers which in turn polymerize into fibrin (Stryer, 1995). The node connecting FIBX\_MOUSE to the rest of the tree is inferred to be a duplication event, since the placement of FIBX\_MOUSE contradicts the species tree and hence FIBX\_MOUSE is inferred to be paralogous to the fibrinogen beta chain subfamily

(FIBB). In contrast, a naive best BLAST analysis of the FIBX\_MOUSE sequence could easily have misannotated it as the mouse fibrinogen beta chain.

## DISCUSSION

In this paper we have presented a simple algorithm to infer gene duplication events on a gene tree by comparing it to a species tree.

Computer science textbooks often warn that comparison of asymptotic worst-case running times may be misleading. Our algorithm is  $O(n^2)$ , yet empirically outperforms at least one more complex algorithm with a superior asymptotic bound close to  $O(n)$  (Eulenstein, 1998), at least in our implementation of the two algorithms. Partly this is because our algorithm has very few steps, so it has a low constant. Also, the worst case behavior of our algorithm is only realized in a pathological case: a gene tree where  $M(g)$  for every internal node points to the root of the species tree, and there are no two genes from the same species (e.g. the number of species in  $S$  is  $O(n)$ ), and the species tree is maximally unbalanced. Figure 4 argues that we do not see such cases in real data. In real data our algorithm is nearly linear time. The Zhang (1997)  $O(n)$  algorithm has not been analyzed in this work, but we expect that there too, the improved asymptotic bound will not be worth the cost of the extra complexity nor the extra computational overhead. We conclude from our results that we will use SDI for future work.

Our goal is to use SDI as part of a system for automating phylogenomics (Eisen, 1998). SDI gives us a clean, simple computational engine that can become part of that larger goal, but there are additional difficulties that must be faced before we put it to practical use. Most importantly, the algorithm assumes at its peril that the gene tree and species tree are both properly rooted and biologically correct.

Phylogenetic inference algorithms produce unrooted gene trees that will have to be rooted before duplication inference can be performed. Usually trees are rooted using either a molecular clock assumption or by defining an outgroup. A molecular clock assumption is generally dubious, and will be especially dubious in a sequence family with different paralogous clades with different functions that are under differing selective pressures. Defining an outgroup in a complicated family of paralogous sequences depends on recognizing the paralogies in the first place, so cannot be done independently of duplication inference. Ironically, one plausible approach to root the gene trees might be to minimize the dissimilarity between the gene tree and a species tree as described in Eulenstein and Vingron (1995), Goodman *et al.* (1979), Guigo *et al.* (1996), Mirkin *et al.* (1995), and Zhang (1997), using a duplication inference algorithm.

Phylogenetic inference algorithms also rarely produce completely reliable gene trees. Even a consensus species

tree based on all available evidence (from paleontological to molecular) will always have ambiguities. Errors in either tree will produce spurious inferred duplications. This is obviously problematic if duplications are to be used as indicators of potential functional changes. One way to portray uncertainty in phylogenetic trees is lack of resolution (i.e. multifurcations). However, the current algorithms are limited to completely resolved (i.e. completely binary) gene and species trees. In addition, the concept of orthology and paralogy is applicable only to completely resolved gene trees. Instead, we think we can approach this issue using sampling methods, such as bootstrap (Mueller and Ayala, 1982; Felsenstein, 1985) or Markov chain Monte Carlo (Mau *et al.*, 1996), to integrate orthology assignments over tree space. This would allow us to calculate a probability, or at least a bootstrap confidence value, for a particular assertion that a known sequence is orthologous to the new sequence being analyzed, and to rank the inferred orthologs by this confidence. Sampling methods can also help us with dealing with ambiguities in rooting the trees. Having a fast algorithm for duplication inference ought to help in any sampling procedure that explores large numbers of tree topologies. However, we recognize that the rate limiting step is more likely to be the tree sampling procedure itself, rather than the duplication inference procedure.

## ACKNOWLEDGEMENTS

This work was supported primarily by a grant from Monsanto Company, and also by the Howard Hughes Medical Institute and grant HG01363 from the NIH National Human Genome Research Institute.

## REFERENCES

- Aguinaldo,A.M.A., Turbeville,J.M., Linford,L.S., Rivera,M.C., Garey,J.R., Raff,R.A. and Lake,J.A. (1997) Evidence for a clade of nematodes, arthropods and other moulting animals. *Nature*, **387**, 489–493.
- Altschul,S.F., Gish,W., Miller,W., Myers,E.W. and Lipman,D.J. (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.
- Apweiler,R., Attwood,T.K., Bairoch,A., Bateman,A., Birney,E., Bucher,P., Codani,J.-J., Corpet,F., Croning,M.D.R., Durbin,R., Etzold,T., Fleischmann,W., Gouzy,J., Hermjakob,H., Jonassen,I., Kahn,D., Kanapin,A., Schneider,R., Servant,F. and Zdobnov,E. (2000) InterPro—an integrated documentation resource for protein families, domains and functional sites. *CCPII Newsletter*, **10**.
- Bairoch,A. and Apweiler,R. (2000) The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000. *Nucleic Acids Res.*, **28**, 45–48.
- Baker,N.E., Mlodzik,M. and Rubin,G.M. (1990) Spacing differentiation in the developing *Drosophila* eye: a fibrinogen-related lateral inhibitor encoded by scabrous. *Science*, **250**, 1370–1377.
- Barns,S.M., Delwiche,C.F., Palmer,J.F. and Pace,N.R. (1996) Perspectives on archaeal diversity, thermophily and monophyly from environmental rRNA sequences. *Proc. Natl Acad. Sci. USA*, **93**, 9188–9193.
- Bateman,A., Birney,E., Durbin,R., Eddy,S.R., Howe,K.L. and Sonnhammer,E.L. (2000) The Pfam protein families database. *Nucleic Acids Res.*, **28**, 263–266.
- Chen,K., Durand,D. and Farach-Colton,M. (2000) Notung: dating gene duplications using gene family trees. In *Proceedings of the Fourth Annual International Conference on Computational Molecular Biology on RECOMB 2000*, pp. 96–106.
- Cormen,T.H., Leiserson,C.E. and Rivest,R.L. (1990) *Introduction to Algorithms*. MIT Press, MA.
- Dayhoff,M.O., Schwartz,R.M. and Orcutt,B.C. (1978) A model of evolutionary change in proteins. In *Atlas of Protein Sequence and Structure*, vol. 5, (Suppl. 3), National Biomedical Research Foundation, Silver Springs, MD, pp. 345–352.
- Eddy,S.R. (2000) HMMER: profile hidden Markov models for biological sequence analysis. Washington University School of Medicine, St Louis, MO (<http://hmmerr.wustl.edu/>).
- Eisen,J.A. (1998) Phylogenomics: improving functional predictions for uncharacterized genes by evolutionary analysis. *Genome Res.*, **8**, 163–167.
- Eulenstein,O. (1998) Vorhersage von Genduplikationen und deren Entwicklung in der Evolution. In *GMD Research Series*, vol. 20, Sankt Augustin, Germany.
- Eulenstein,O. and Vingron,M. (1995) On the equivalence of two tree mapping measures. In *Arbeitspapiere der GMD*, vol. 936, Sankt Augustin, Germany.
- Eulenstein,O., Mirkin,B. and Vingron,M. (1998) Duplication-based measures of difference between gene and species trees. *J. Comput. Biol.*, **5**, 135–148.
- Felsenstein,J. (1985) Confidence limits on phylogenies: an approach using the bootstrap. *Evolution*, **39**, 783–791.
- Felsenstein,J. (1993) PHYLIP: Phylogeny Inference Package, Version 3.5. University of Washington, Seattle, WA.
- Fornace,A.J., Cummings,D.E., Comeau,C.M., Kant,J.A. and Crabtree,G.R. (1984) Structure of the human gamma-fibrinogen gene. Alternate mRNA splicing near the 3' end of the gene produces gamma A and gamma B forms of gamma-fibrinogen. *J. Biol. Chem.*, **259**, 12 826–12 830.
- Goodman,M., Czelusniak,J., Moore,G.W., Romero-Herrera,A.E. and Matsuda,G. (1979) Fitting the gene lineage into its species lineage, a parsimony strategy illustrated by cladograms constructed from globin sequences. *Syst. Zool.*, **28**, 132–168.
- Guigo,R., Muchnik,I. and Smith,T.F. (1996) Reconstruction of ancient phylogenies. *Mol. Phylogenet. Evol.*, **6**, 189–213.
- JáJá,J. (1991) *An Introduction to Parallel Algorithms*. Addison-Wesley, Reading, MA, pp. 128–136.
- Jones,F.S., Burgoon,M.P., Hoffman,S., Crossin,K.L., Cunningham,B.A. and Edelman,G.M. (1988) A cDNA clone for cyto-tactin contains sequences similar to epidermal growth factor-like repeats and segments of fibronectin and fibrinogen. *Proc. Natl Acad. Sci. USA*, **85**, 2186–2190.
- Mau,B., Newton,M.A. and Larget,B. (1996) Bayesian phylogenetic inference via Markov chain Monte Carlo methods. *Technical Report 961*, Statistics Department, University of Wisconsin-Madison.
- Mirkin,B., Muchnik,I. and Smith,T.F. (1995) A biologically con-

- sistent model for comparing molecular phylogenies. *J. Comput. Biol.*, **2**, 493–507.
- Mueller,L.D. and Ayala,F.J. (1982) Estimation and interpretation of genetic distance in empirical studies. *Genet. Res.*, **40**, 127–137.
- Page,R.D.M. (1994) Maps between trees and cladistic analysis of historical associations among genes, organisms, and areas. *Syst. Biol.*, **43**, 58–77.
- Page,R.D.M. (1998) GeneTree: comparing gene and species phylogenies using reconciled trees. *Bioinformatics*, **14**, 819–820.
- Page,R.D.M. and Charleston,M.A. (1997) Reconciled trees and incongruent gene and species trees. In Mirkin,B., McMorris,F.R., Roberts,F.S. and Rzhetsky,A. (eds), *Mathematical Hierarchies in Biology*, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, vol. 37, American Mathematical Society, Providence, RI, pp. 57–70.
- Page,R.D.M. and Holmes,E.C. (1998) *Molecular Evolution: a Phylogenetic Approach*. Blackwell Science, Oxford, pp. 30–31.
- Parr,R.L., Fung,L., Reneker,J., Myers-Mason,N., Leibowitz,J.L. and Levy,G. (1995) Association of mouse fibrinogen-like protein with murine hepatitis virus-induced prothrombinase activity. *J. Virol.*, **69**, 5033–5038.
- Saitou,N. and Nei,M. (1987) The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol. Biol. Evol.*, **4**, 406–425.
- Schieber,B. and Vishkin,U. (1988) On finding lowest common ancestors: simplification and parallelization. *SIAM. J. Comput.*, **17**, 1253–1262.
- Stryer,L. (1995) *Biochemistry*. Freeman, New York.
- Swofford,D.L., Olsen,G.J., Waddell,P.J. and Hillis,D.M. (1996) Phylogenetic inference. In Hillis,D.M., Moritz,C. and Mable,B.K. (eds), *Molecular Systematics*. Sinauer, Sunderland, MA, pp. 488.
- Tatusov,R.L., Natale,D.A., Garkavtsev,I.V., Tatusova,T.A., Shankavaram,U.T., Rao,B.S., Kiryutin,B., Galperin,M.Y., Fedorova,N.D. and Koonin,E.V. (2001) The COG database: new developments in phylogenetic classification of proteins from complete genomes. *Nucleic Acids Res.*, **29**, 22–28.
- Zhang,L. (1997) On a Mirkin–Muchnik–Smith conjecture for comparing molecular phylogenies. *J. Comput. Biol.*, **4**, 177–187.
- Zmasek,C.M. and Eddy,S.R. (2001) ATV: display and manipulation of annotated phylogenetic trees. *Bioinformatics*, **17**, 383–384.