

WASHINGTON UNIVERSITY
Division of Biology and Biomedical Sciences
Program in Molecular Genetics

Dissertation Examination Committee:

Sean R. Eddy, Chair

Susan K. Dutcher

Warren R. Gish

Jingdong Liu

Katherine P. Ponder

Tim Schedl

Gabriel Waksman

FUNCTIONAL ANALYSES OF PROTEOMES BY PHYLOGENETIC METHODS

by

Christian Matthias Zmasek

A dissertation presented to the Graduate School of Arts and Sciences of
Washington University in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

May 2002

Saint Louis, Missouri

Acknowledgements

I would like to thank my thesis advisor, Dr. Sean Eddy, for making this dissertation possible. Sean has provided me with guidance, support, advice and an unwavering interest in my project. Perhaps most importantly he has taught me how to think critically and independently about my research.

I would also like to thank the members of my committee, Dr. Susan Dutcher, Dr. Warren Gish, Dr. Jingdong Liu, Dr. Katherine Ponder, Dr. Tim Schedl, Dr. Alan Templeton and Dr. Gabriel Waksman, for many insights, suggestions and encouragement throughout my years of research as a doctoral student.

Thanks to the past and current members of the Eddy laboratory including Zhirong Bao, Goran Ceric, Robin Dowell, Tom Jones, Ajay Khanna, Robbie Klein, HaiNing Kong, Dr. Todd Lowe, Mary Pichler and Dr. Elena Rivas for their help and patience over the years. Finally, thanks to Carolyn Lawrence for porting ATV to the mac and thanks to Xiaoqing You for useful advice.

This work is dedicated to my parents, Monica and Richard Zmasek.

Table of Contents

Acknowledgements	ii
Table of Contents	iii
List of Figures	vi
List of Tables	viii
Abbreviations Used	ix
Abstract of Dissertation	x
1 Introduction	2
1.1 Scope of Thesis	3
1.2 Sequence Function Analysis	4
1.2.1 Most functional predictions for uncharacterized genes are based on sequence similarity	5
1.2.2 Functional predictions based on sequence similarity fail in certain cases	5
1.2.3 Phylogenomic methods might allow accurate functional predictions in cases where similarity based methods fail	8
1.3 Molecular Evolution	11
1.3.1 Historical Background	11
1.3.2 Mechanisms of Protein Evolution	13
1.3.2.1 Gene Duplication	13
1.3.2.2 Domain Shuffling	19
1.4 Phylogenetic Inference Based on Molecular Sequences	21
1.4.1 Phylogenetic Trees	21
1.4.1.1 Gene Trees and Species Trees	25
1.4.2 Methods for Phylogenetic Inference	28
1.4.2.1 Multiple Sequence Alignment	31
1.4.2.2 Pairwise Protein Distance Calculation	32
1.4.2.2.1 A maximum Likelihood Approach to Distance Calculation	35
1.4.2.3 Optimality Criteria Based on Pairwise Distances	39
1.4.2.3.1 Least Squares	42
1.4.2.3.2 Minimum Evolution	43
1.4.2.4 Optimality Criteria Based on Character Data	43

1.4.2.4.1 Maximum Parsimony	43
1.4.2.4.2 Maximum Likelihood	44
1.4.2.5 Algorithmic Methods Based on Pairwise Distances.....	46
1.4.2.5.1 UPGMA	46
1.4.2.5.2 Neighbor Joining.....	47
1.4.2.6 Bootstrapping	50
1.5 Overview	52
2 ATV: Display and Manipulation of Annotated Phylogenetic Trees	55
2.1 Abstract	56
2.2 Introduction	57
2.3 Features	59
2.4 Implementation	61
2.5 Acknowledgements	62
2.6 The New Hampshire X Format (NHX)	63
3 SDI: A Simple Algorithm to Infer Gene Duplication and Speciation Events on a Gene Tree.....	67
3.1 Abstract	68
3.2 Introduction	69
3.3 Algorithm.....	72
3.4 Implementation	80
3.5 Results	81
3.6 Discussion	86
3.7 Acknowledgements	89
4 RIO: Analyzing proteomes by automated phylogenomics using resampled inference of orthologs.....	90
4.1 Abstract	91
4.2 Introduction	92
4.3 Algorithm	98
4.3.1 Definitions	99
4.3.2 The RIO procedure	105
4.3.3 Precalculation of pairwise distances for increased time efficiency.....	109
4.3.4 Rooting of gene trees	110
4.3.5 Master species tree.....	112
4.4 Implementation	114

4.5 Results and Discussion	114
4.5.1 Precalculation of pairwise distances.....	115
4.5.2 Phylogenomic analyses of the <i>A. thaliana</i> and <i>C. elegans</i> and proteomes	117
4.5.2.1 Domain structure analysis	117
4.5.2.2 RIO analysis	118
4.5.2.2.1 How many sequences can be analyzed with RIO?	119
4.5.2.2.2 RIO analysis of lactate/malate dehydrogenase family members.....	121
4.5.2.2.3 Sequences with no orthologs in the current databases	126
4.5.2.2.4 Inconsistency between orthology bootstrap values and sequence similarity.....	132
4.6 Conclusions	138
4.7 Acknowledgements	142
4.8 Appendix A: Precalculation of pairwise distances.....	143
4.9 Appendix B: Speciation Duplication Inference combined with rooting	145
5 Conclusions and future directions.....	148
6 Bibliography	151

List of Figures

Figure 1.1. The evolutionary history of human globin genes.....	18
Figure 1.2. Examples of orthologs and paralogs.	18
Figure 1.3. Tissue plasminogen activator is a mosaic protein.	20
Figure 1.4. A phylogenetic tree proposed by Haeckel (1866).	24
Figure 1.5. A gene tree of orthologs based on Interleukin-3 protein sequences.	26
Figure 1.6. A gene tree of orthologs and paralogs based on Bcl-2 family protein sequences.	27
Figure 1.7. Additive and ultrametric trees.	41
Figure 1.8. Tree of N external nodes in which i and j are neighbors.....	50
Figure 1.9. An example of the bootstrap resampling procedure.	51
Figure 2.1. ATV displaying a phylogenetic tree of biotin-requiring enzymes.....	60
Figure 2.2. A sample tree to illustrate the NHX format.	66
Figure. 3.1. Gene trees and species trees.	73
Figure. 3.2. The mapping function M and the definition of a duplication.	75
Figure. 3.3. The number of duplications and the topology of the species tree influence the time complexity of our algorithm.	78
Figure. 3.4. Timing benchmarks on real trees to determine average-case behavior, and synthetic trees that exercise our algorithm's worst case behavior.	83
Figure. 3.5. A gene tree for the fibrinogen beta and gamma chain Pfam family.....	85
Figure 4.1. Over annotation due to database bias or gene loss under equal rates of evolution.....	95
Figure 4.2. Over annotation due to unequal rates of evolution.	96
Figure 4.3. The purpose of super-orthologs.	101
Figure 4.4. An example of ultra-paralogous sequences.....	103
Figure 4.5. An illustration of subtree-neighbors.....	104
Figure 4.6. A simple example of the RIO procedure.....	108
Figure 4.7. RIO output for the <i>A. thaliana</i> protein F12M16_14 analyzed against the Pfam ldh domain alignment (PF00056).	123

Figure 4.8. A phylogenetic tree for zinc-binding dehydrogenases produced by RIO.	130
Figure 4.9. RIO output for the <i>A. thaliana</i> protein F28P22_13 analyzed against the Pfam adh_zinc domain alignment (PF00107).....	132
Figure 4.10. A phylogenetic tree for O-methyltransferases produced by RIO.	136
Figure 4.11. RIO output for the <i>A. thaliana</i> protein F16P17_38 analyzed against the Pfam Methyltransf_2 domain alignment (PF00891).....	137

List of Tables

Table 1.1. PAM ₁ , a transition probability matrix for the evolutionary distance of 1 PAM.....	38
Table 2.1. Tags used in the NHX format.....	64
Table 4.1. Number of domains which can be analyzed with RIO.....	120
Table 4.2. RIO analysis of <i>A. thaliana</i> lactate/malate dehydrogenase family members.....	124
Table 4.3. RIO analysis of <i>C. elegans</i> lactate/malate dehydrogenase family members.	125
Table 4.4. Top orthology bootstrap values of RIO analyses.	127
Table 4.5. The numbers of sequences for which the orthology bootstrap values do not correspond to sequence similarity.	133

Abbreviations Used

HMM: hidden Markov model

LCA: last (least) common ancestor

LDH: lactate dehydrogenase

MDH: malate dehydrogenase

ML: maximum likelihood

NJ: neighbor joining

OTU: operational taxonomic unit

UPGMA: unweighted pair group method using arithmetic averages

Abstract of Dissertation

In this work, computational methods for the purpose of sequence function prediction based on molecular evolution were developed and tested.

When analyzing molecular sequences using sequence similarity searches, orthologous sequences (diverged by speciation) are more reliable predictors of biological function than paralogous sequences (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 (not based on phylogenetic trees). Therefore, a procedure for automated phylogenomics using explicit phylogenetic inference was produced.

At the center of a phylogenomic approach stands the inference of gene duplications by comparing the gene tree containing the sequence to be analyzed to a trusted species tree. An algorithm for this purpose was developed. This algorithm exhibits an inferior worst case behavior compared to previously published ones but appears to be superior in most practical cases, partially due to its simplicity.

A major caveat of all phylogenetic analyses is the unreliability of the resulting trees. Therefore, inference of gene duplications is performed over bootstrap-resampled phylogenetic trees to estimate the reliability of the orthology assignments. Additionally, supplementary measures extending the concepts of orthology and paralogy were introduced and assessed for their effectiveness in functional prediction.

The phylogenomic approach developed in this work was tested on the proteomes of the flowering plant *Arabidopsis thaliana* and the nematode *Caenorhabditis elegans*. It appears that this approach is particularly useful for the automated detection of first representatives of novel protein subfamilies.

“Nothing in biology makes sense except in the light of evolution.”

Theodosius Dobzhansky (1900-1975)

1 Introduction

1.1 Scope of Thesis

The scope of this thesis is the development and evaluation of computational methods for analyzing the biological roles of protein sequences using concepts from the field of molecular evolution.

The following three main components were developed in the course of this work:

ATV: A program for the display of annotated phylogenetic trees as a tool for studying large gene trees. The resulting publication makes up chapter 2 of this dissertation.

SDI: An algorithm to determine which nodes on a phylogenetic tree represent gene duplications. The resulting publication makes up chapter 3.

RIO: Resampled Inference of Orthologs: This is a package of programs for the automated analysis of protein sequences by phylogenetic methods. The resulting publication makes up chapter 4.

At the center of the methods described in this thesis stands the inference of gene duplications by comparing the gene tree containing the sequence to be analyzed to a trusted species tree. An algorithm for this purpose (“SDI”) has been developed and analyzed.

A major caveat of all phylogenetic analyses is the unreliability of the resulting trees. Therefore, inference of gene duplications is performed over bootstrap resampled phylogenetic trees to estimate the reliability of the orthology assignments. “RIO” is a package which bundles programs for these purposes.

1.2 Sequence Function Analysis

In 1977, the genome of the bacteriophage Φ X174 was determined by the emerging technology of DNA sequencing (Sanger et al., 1977). After two decades of technology development, the genome of *Haemophilus influenza* was reported in 1995 (Fleischmann et al., 1995) as the first complete bacterial genome. Now, the genomes of three metazoans, human (Lander et al., 2001; Venter et al., 2001), the nematode *Caenorhabditis elegans* (C.elegans-Sequencing-Consortium, 1998), and the fruit fly *Drosophila melanogaster* (Adams et al., 2000), the flowering plant *Arabidopsis thaliana* (Arabidopsis-Initiative, 2000), the yeast *Saccharomyces cerevisiae* (Goffeau et al., 1996), as well as those of many Bacteria and Archaea are known (Doolittle, 1998). For a constantly updated list of published microbial genomes and microbial genomes in progress see [<http://www.tigr.org/tdb/mdb/mdbcomplete.html>].

This information will lead toward a basic understanding of the fundamental problems in life sciences, as well as stimulation of practical applications in medical, pharmaceutical, and agricultural sciences. However, the sequence data obtained by genome sequencing projects do not by themselves provide direct answers to such fundamental problems or practical applications. The sequencing of a genome is an easier task than the understanding of functional implications of when, where, and most importantly, how genes and molecules function and interact in organisms.

1.2.1 Most functional predictions for uncharacterized genes are based on sequence similarity

Functional predictions based on sequence similarity are widely used. The most simple method is based on the highest scoring hit (the “top one hit”). The uncharacterized sequence is assigned the function of the sequence that is identified as having the highest degree of similarity by a similarity search program like BLAST (Altschul et al., 1990). For example, the *Helicobacter pylori* genomic sequence has been analyzed in such a way (Tomb et al., 1997). Another method is based on examining a certain number of top hits. Depending on the degree of consensus of the genes identified as having the highest degree of similarity, the query sequence is assigned a specific function, a general activity, or an unknown function. The *Escherichia coli* genomic sequence has been analyzed this way (Blattner et al., 1997). The predicted coding regions with putative identifications are typically assigned biological roles with the classification system adapted from Riley (1993).

1.2.2 Functional predictions based on sequence similarity fail in certain cases

Sequences can be similar due to convergence or homology. Homologs are sequences which share a common ancestor, whereas convergent sequences lack a

common evolutionary history. Homologs can be divided into orthologs and paralogs. Orthologs are homologous sequences that diverged from each other by speciation. Paralogs are homologous sequences that diverged from each other by gene duplication (for more detailed information see section 1.3.2.1). It follows that there are at least three difficulties with functional predictions based on sequence similarity. First, sequence similarities can be due to convergence and therefore not necessarily indicate functional similarity. This problem can be partially overcome by just considering similarities that are too high to be due to convergence (although a threshold for such an inference is not well established). Second, sequences can have a high degree of sequence similarity without fulfilling the same biological role; this is particularly likely for paralogous sequences. An example of this is L-lactate dehydrogenase (EC 1.1.1.27) and malate dehydrogenase (EC 1.1.1.37). These two enzymes are thought to have evolved from a common ancestor (Golding and Dean, 1998). They share a high degree of sequence similarity (BLASTP E-value is in the range of 5.0×10^{-7}) but have different substrate specificities and therefore different biological roles. (See Figures 4.1 and 4.2 later in this work for illustrations of how paralogy combined with gene loss or database bias and/or unequal rates of evolution can lead to erroneous predictions if only sequence similarity is considered.) Third, homologous sequences can diverge so much that sequence similarities are difficult to detect.

In addition, we need to consider that no clear definition for sequence “function” exists. Depending on the sequences studied and the methods used, predictions with different levels of resolution will result. Functional prediction

can be understood as a rough classification of proteins according to the “class” of reaction catalyzed, such as hydrogenases, kinases, etc. A more detailed classification also includes information about substrates, products, and cofactors. An example of such a classification is the EC-number system. In the EC (Enzyme Commission) nomenclature, enzymes are principally classified and named according to the reaction they catalyze. The enzymes are divided into hierarchical groups where group membership is encoded into a code of four numbers. The first number shows to which of the six main divisions the enzyme belongs (EC 1. - oxidoreductases, EC 2. - transferases, EC 3. - hydrolases, EC 4. - lyases, EC 5. - isomerases, EC 6. - ligases). The second number indicates the subclass (e.g. EC 1.1.: oxidoreductases acting on the CH-OH group of donors). The third number gives the sub-subclass (e.g. EC 1.1.1.: oxidoreductases acting on the CH-OH group of donors with NAD⁺ or NADP⁺ as acceptor). The fourth number is the serial number of the enzyme in its sub-subclass (e.g. EC 1.1.1.145: 3 β -hydroxy- Δ^5 -steroid dehydrogenase, EC 1.1.1.146: 11 β -hydroxysteroid dehydrogenase) (Webb, 1992). A database of EC-numbers is available at [<http://www.chem.qmw.ac.uk/iubmb/enzyme/>]. An even more detailed functional description might include explicit information about the temporal and spatial expression (during development and/or in response to external stimuli), sub-cellular localization, regulatory properties (inhibitors and activators), biochemical properties such as K_m , V_{max} , temperature and pH optimum, etc. A very promising approach is gene ontologies which are controlled vocabularies to describe sequences (somewhat similar to the EC-number system, but not limited

to enzymes and biochemistry) (Gene-Ontology-Consortium, 2001). For more information, see [<http://www.geneontology.org/>].

1.2.3 Phylogenomic methods might allow accurate functional predictions in cases where similarity based methods fail

Although sequence similarity based methods for functional prediction are very fast, readily automated, and usually sufficiently accurate, *per se* they make use of phylogenetic information no more than indirectly – as an array of numerical values instead of a tree-topology. Ignoring the tree-topology can lead to inaccurate predictions in certain situations (for example in sequence families where paralogs with different functions are present combined with gene loss or incomplete databases).

On the other hand, methods based on sequence or motif family profiles are very robust but, oftentimes, the resulting annotations are too broad (e.g. a new sequence might be annotated just as “kinase”). An example of this approach is using the HMMER software (Eddy, 2000) to search the protein domain database Pfam (Bateman et al., 2000).

Another approach for improved functional prediction are methods based on catalytic key residues (or sequence patterns). Identification of the amino acids responsible for the reaction catalyzed for each type of reaction would allow to make inferences about the catalytic activity of unknown sequences by pattern

matching. Unfortunately, such methods require intimate knowledge about each catalytic mechanism. In addition, by just concentrating on the key residues, all the information buried in the rest of the sequence is not utilized, and therefore the resolution of methods based on key residues is expected to be rather limited. An example of a database containing patterns (and profiles) is the PROSITE database (Bairoch et al., 1997) [<http://ca.expasy.org/prosite/>]. But this database does not necessarily concentrate on the catalytically important residues, it is a collection of any type of pattern or profile which can be used for the classification of sequences.

Realizing these shortcomings, Tatusov et al. (1997; 2001) developed the Clusters of Orthologous Groups (COGs) method [<http://www.ncbi.nlm.nih.gov/COG/>]. This method is based on the assumption that orthologs are more similar to each other than they are to paralogs. The procedure to construct COGs starts with building groups of three sequences from three different species, whose members are reciprocal best hits to each other, and therefore *assumed* to be orthologs. This is done for all possible combinations of three species. Then, groups which share two members are merged into larger COGs until no more of such mergers are possible. The expectation is that each COG consists of individual orthologs or orthologous groups of paralogs from within the same species (i.e. no speciations after duplications). Each COG is assumed to have evolved from an individual ancestral gene through a series of speciation and duplication events. The COG method is probably superior to simpler sequence similarity based methods but it still does not use the power of phylogenetic analysis since clustering is a way of classifying levels of similarity

and is not an accurate method of inferring evolutionary relationships (Swofford et al., 1996).

In order to obtain more reliable functional predictions, one might incorporate explicit evolutionary relationships into sequence function prediction methods. One way to accomplish this goal is by creating a phylogenetic tree of all homologs. The topology of the tree will allow the distinction between orthologs and paralogs by comparing with the species tree. The likely function of the sequences of interest can then be inferred by overlying the known function onto the tree. This approach has been termed **phylogenomics** by Eisen (Eisen, 1998a; Eisen, 1998b; Eisen, 2001; Eisen and Hanawalt, 1999; Eisen et al., 1997; Eisen et al., 1995).

It is the goal of this work to extend and automate phylogenomics. The rest of chapter 1 discusses the background of molecular evolution and methods for phylogenetic tree inference.

1.3 Molecular Evolution

Molecular evolution is the study of the evolution of macromolecules. This section is a review of some key concepts from this large field of study. A brief overview of the history of molecular evolution and its most important concepts and controversies is followed by a discussion of the mechanisms of protein evolution (for reviews see Avise, 1994; Li, 1997; Nei, 1987; Page and Holmes, 1998). In particular, the concept of gene duplication and its significance for protein function evolution are introduced here. The next section (1.4) discusses methods for the reconstruction of evolutionary histories of macromolecules.

1.3.1 Historical Background

The study of molecular evolution began at the turn of the twentieth century. Studies in immunohistochemistry showed that serological cross-reactions were stronger for more closely related organisms than for less related ones. Nuttall (1904) used immunohistochemistry to infer that, for example, man's closest relatives were the apes. Yet, intense research in molecular evolution started only in the 1950s, due to the introduction of new techniques such as protein sequencing, tryptic fragment pattern analysis, starch-gel electrophoresis, and improvements in immunohistochemistry (e.g. Brown et al., 1955; Zuckerkandl et al., 1960). In particular, Zuckerkandl and Pauling (1962; 1965b) showed that the characters in molecular sequences can contain a large amount of information ("molecules as documents of evolutionary history"). Many studies in

the 1960s were centered around the molecular evolution of humans, apes, and primates in general (e.g. Goodman, 1962). By then, the amino acid sequences of a variety of proteins such as hemoglobins and cytochromes c had been determined (e.g. Margoliash et al., 1968). Comparative studies of these sequences revealed that the rate of amino acid substitution in each of these sequences was approximately the same among different lineages. This led to the proposal of a **molecular clock** (Zuckerkandl and Pauling, 1962; Zuckerkandl and Pauling, 1965a), a theory which is controversial to this day (Ayala, 1999; Tajima, 1993; Zuckerkandl, 1987).

An unexpectedly high rate of evolution in terms of nucleotide substitutions led Kimura (1968a; 1968b) to propose the **neutral theory** of evolution. King and Jukes (1969) published a similar idea, although from a more biochemical perspective. The neutral theory claims that molecular evolution is dominated by genetic drift of neutral mutations which have no selective cost. In other words, the neutralists model states that majority of mutations are deleterious and quickly removed by negative selection. The majority of fixed mutations are neutral, and only a small percentage is advantageous (summarized in Kimura, 1983; Kimura, 1991). The opposing argument is that the natural selection of advantageous mutations is the more important force in molecular evolution (King, 1972). In this model, the majority of the fixed mutations are advantageous. The **nearly neutral theory** has been proposed by Ohta (1973) to explain the fact that the level of heterozygosity observed is oftentimes not as high as expected under the neutral theory. The nearly neutral theory claims that the majority of fixed mutations are either neutral or slightly deleterious (or slightly

advantageous) (reviewed in Ohta, 1992a; Ohta, 1992b; Ohta, 1996). The selection-neutrality debate has not been settled, even though the neutral (or nearly neutral) theory is oftentimes considered the null-model which has to be rejected before other, more specialized, models can be entertained (Moritz and Hillis, 1996; Ohta, 1996).

1.3.2 Mechanisms of Protein Evolution

Gene duplication and domain shuffling are important mechanisms for generating novel biochemical and regulatory functionality (Lynch and Conery, 2000; Ohno, 1970). In particular, gene duplication might have been the primary mechanism for the evolution of complexity in higher organisms (Miklos and Rubin, 1996; Ohta, 1991). This section reviews these two related mechanisms and introduces some important definitions.

1.3.2.1 Gene Duplication

It is generally supposed that new genes evolve if mutations accumulate while selective constraints are relaxed by gene duplication (Kimura, 1983; Ohno, 1970). The importance of gene duplication for evolution has probably first been recognized by Haldane (1932) (“... it [mutation pressure] will favour polyploids, and particularly allopolyploids, which possess several pairs of sets of genes, so that one gene may be altered without disadvantage...”, p. 194) and Muller (1935; 1936). Cytological studies of the fruit fly *Drosophila melanogaster* showed that

certain banding patterns appear duplicated, illustrating “the manner of origination of extra genes in evolution”. Serebrowsky (1938) was the first to formulate a hypothesis of the possible steps involved (although his interpretation only attempted to explain a specialization in function and did not include the acquisition of new functions). Stephens (1951) concluded that “theoretically, duplication of loci would appear to offer a means of gaining a new function without losing the old one”, yet he was unable to find a convincing example in the available data. In the early 1960s the amino acid sequences for human hemoglobins became available (e.g. Konigsberg et al., 1961). Comparing the amino acid sequences of human myoglobin, and hemoglobins α , β , γ , and δ led Ingram (1961) to propose a model where myoglobin and the hemoglobins form a family of homologous proteins, and are related to each other by gene duplication events, similar to the illustration in Figure 1. [**Homologs** are defined as sequences which share a common ancestor (Fitch, 1966). This definition becomes unclear if mosaic proteins, which are composed of structural units originating from different genes (section 1.3.2.2), are considered.] These studies led Ohno (1970) to conclude that gene duplication is the only means for the creation of new genes. Even though it is now known that there are other means for creating new genes or new functionality, gene duplication is still considered the most important one (Doolittle, 1995; Miyata et al., 1994; Ohta, 1989a). [Other means for creating new functionality include: alternative splicing (Smith et al., 1989), RNA editing (Chan, 1993), overlapping genes such as tRNA genes on complementary strands of a DNA sequence (Anderson et al., 1981), and genes with more than one function or “gene sharing” (Piatigorsky et al., 1988) or “gene

recruitment” – a fascinating example for this are the crystallins which are responsible for the transparency of the eye lens but which also act as enzymes, catalyzing a variety of biochemical reactions. Crocodile ξ crystallin for example also serves as lactate dehydrogenase, bird δ crystallin also has argininosuccinate lyase functionality; for a complete list see Wistow (1993).] More recent studies on gene duplication concentrate on the simulation of duplications and the corresponding population genetic models (e.g. Gu, 1999; Gu, 2001; Ohta, 1987; Ohta, 1988a; Ohta, 1988b; Ohta, 1989b). Recently, the significance of duplications has also been studied using artificial life simulations (Calabretta et al., 2000). The result of these studies appear to confirm the ideas presented in this section.

Duplications may affect a part of a gene (partial or internal gene duplication or possibly domain duplication), a complete gene (complete gene duplication), parts of a chromosome, a complete chromosome, or a whole genome (Lander et al., 2001; Sankoff, 2001; Venter et al., 2001). For example, a current (controversial) theory suggests that vertebrates underwent two rounds of whole genome duplication (e.g. Friedman and Hughes, 2001; Meyer and Schartl, 1999).

Possible mechanisms for gene duplication include (Fitch et al., 1991; Lander et al., 2001; Ohta, 1989a; Venter et al., 2001): unequal crossing-over (recombination between nonallelic genes caused by misalignment of chromosomes during meiosis which leads to one chromosome with a duplication and to one with a deletion), gene conversion [exchange of strands between DNA molecules (originally proposed in Holliday, 1964; reviewed in Szostak et al.,

1983)], errors during recombination and repair, and retrotransposition (resulting in intronless gene copies).

Internal gene duplications result in **gene elongation** (Eck and Dayhoff, 1966), which is an import mechanism for evolving complex genes from simple ones (similar to domain shuffling, see below). An example of a gene arisen by internal duplications is the $\alpha 2$ type I collagen gene. 42 of its 52 exons contain multiples of the 9 basepairs coding for the triplet Gly-X-Y. It is likely that these 42 exons arose from one exon by multiple internal duplications (Li, 1983; Yamada et al., 1980). For a slightly curious example of a two domain hemoglobin in a water flea caused by internal duplication, see Kato et al (2001). Internal gene duplications can also lead to specific integrated assemblies such as β -propellers and β -trefoils (Andrade et al., 2001).

All other types of duplications (other than internal gene duplications) result in two identical copies of each duplicated gene. As indicated above, one copy may acquire mutations or become subject to domain shuffling (section 1.3.2.2) and eventually assume a different biological role (or become silenced by deleterious mutations) (Lynch and Conery, 2000). Several examples are known in which amino acid substitution in duplicated genes is accelerated relative to synonymous substitutions (Ohta, 1991; Ohta, 1993; Ohta, 1994). It is also possible that the duplicated copies are simply used to increase the amount of gene product (rRNA genes, for example).

Multiple gene duplications lead to **gene families** (Dayhoff, 1976). For examples of gene families see Figures 1.1 and 1.6. Multiple gene duplications combined with exon shuffling lead to **super families**. For the purpose of this

work, gene families are defined as families of genes which have one common ancestor (and are therefore homologs) and which exhibit the same domain organization (“end to end similarity”). The term family is also used for individual domains with a common ancestor. Super families are defined as groups of genes containing at least one structural unit of common evolutionary origin (Go, 1981).

Homologous sequences can be divided into orthologs, paralogs and xenologs (for examples see Figure 1.2). **Orthologs** are defined as two sequences which diverged by a speciation event (their last common ancestor on a phylogenetic tree corresponds to a speciation event). **Paralogs** are defined as two sequences which diverged by a duplication event (their last common ancestor corresponds to a duplication) (Fitch, 1970). **Xenologs** are defined as two sequences which are related to each other by horizontal gene transfer (via retroviruses, for example) (Gray and Fitch, 1983). [Some common misconceptions surround the concepts of orthology and paralogy. For example, a common mistake is the assumption that in order for two sequences to be paralogous to each other, they have to occur in the same species. For a review of these issues see Jensen (2001).]

1.3.2.2 Domain Shuffling

Domain shuffling is another mechanism in addition to point mutations which can lead to modification of protein function. Domain shuffling can be divided into two types: domain duplication and domain insertion (also called domain recruitment) (Li, 1997). Domain duplication describes the duplication of one or more domains and is a type of internal duplication discussed above. Domain insertion leads to **mosaic proteins**, proteins composed of domains (or structural subunits) originating from different proteins (Doolittle, 1985; Doolittle, 1995; Patthy, 1987; Patthy, 1991). An example for a mosaic protein is tissue plasminogen activator (TPA) (see Figure 1.3). TPA converts plasminogen into plasmin, a serin protease which in turn lyses fibrin in blood clots (van Zonneveld et al., 1986). TPA is composed of four structural domains: one finger module originating from fibronectin (function: binding of fibrin to activate TPA), one growth factor module from epidermal growth factor (function: stimulation of cell proliferation), and two kringle modules from plasminogen (function: binding of clot proteins) (Patthy, 1985). For more examples see Doolittle (1995). Interestingly, it has been proposed to use domain shuffling for the rational design of novel protein functions (Ostermeier and Benkovic, 2000).

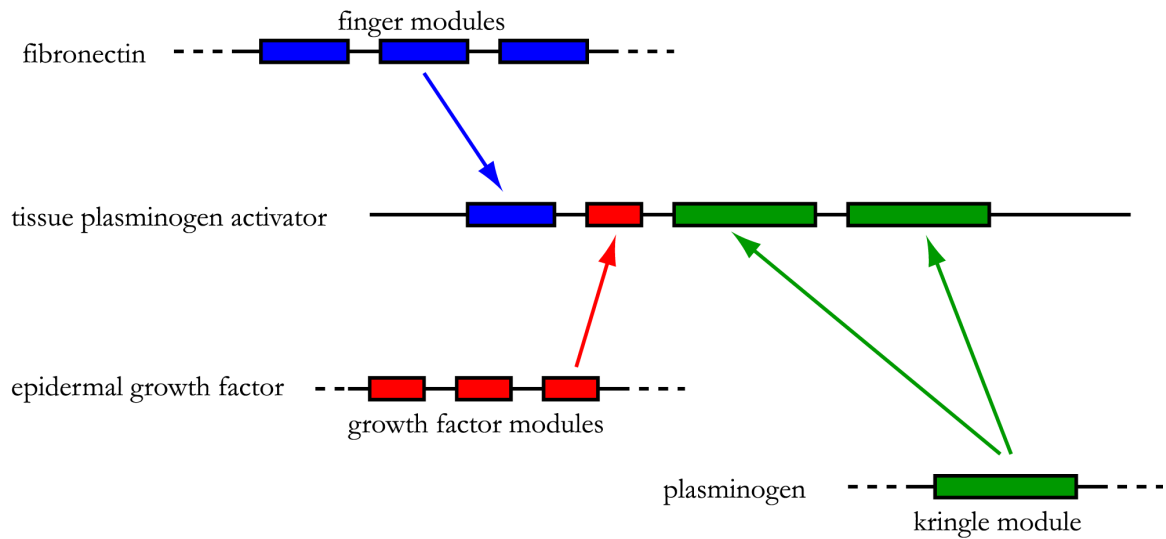


Figure 1.3. Tissue plasminogen activator is a mosaic protein.

TPA is the result of domain shuffling and is composed of four structural domains: one finger module from fibronectin, one growth factor module from epidermal growth factor, and two kringle modules from plasminogen (Patthy, 1985).

1.4 Phylogenetic Inference Based on Molecular Sequences

A phylogeny is the evolutionary history of a species or a group of species. Lately, the term is also being applied to the evolutionary history of individual DNA or amino acid sequences. This section discusses some of the methods and models used for the reconstruction of phylogenies based on sequence data (for the most, part amino acid sequences). In particular, the background for the tree building algorithms used in chapter 4 are introduced here. For reviews see Durbin et al. (1998), Felsenstein (1982; 1988; 1996), Nei (1996), Page and Holmes (1998), Saitou (1996), and Swofford et al. (1996).

1.4.1 Phylogenetic Trees

The evolutionary history of organisms or sequences can be illustrated using a tree-like diagram – a phylogenetic tree. For an example, see Figure 1.4, showing a phylogenetic tree proposed in 1866 by Haeckel (1866).

A phylogenetic tree is a representation of the evolutionary relationships among a set of sequences, species or populations. The tree is a kind of graph and is composed of branches (edges) and nodes (vertices). Nodes are divided into internal and external ones. The external nodes are also called operational taxonomic units (OTUs), leaves, or tips. Typically, the external nodes correspond to contemporary sequences, species, or populations.

Branch lengths might be according to time or evolutionary distance, or the tree simply represents the evolutionary relationships and branch lengths are arbitrary (as in the trees shown in Figures 1.5 and 1.6).

Phylogenetic trees can either be completely binary, which means that each node has two child nodes (bifurcation or dichotomy), or they can contain multifurcations or polytomies (more than two children per node). Multifurcations are used to express radiations and/or uncertainties about the tree topology (Hennig, 1966).

A tree can be either rooted if the direction of time is known or unrooted if the direction of time is unknown. A rooted tree has a special internal node, called the root which is defined as the position of the common ancestor.

A unrooted completely binary tree with N external nodes has $2N-3$ branches, and $N-2$ internal nodes. A rooted completely binary tree with N external nodes has $2N-2$ branches, and $N-1$ internal nodes.

The number of different tree topologies increases rapidly with an increase in number of external nodes. The general equation for the possible number of topologies for unrooted completely binary trees (T) with $N (>2)$ external nodes is (Cavalli-Sforza and Edwards, 1967):

$$Tp = \frac{(2N-5)!}{2^{N-3}(N-3)!} \quad (1-1)$$

For example, there are 221,643,095,476,699,771,875 different unrooted tree topologies with 20 external nodes.

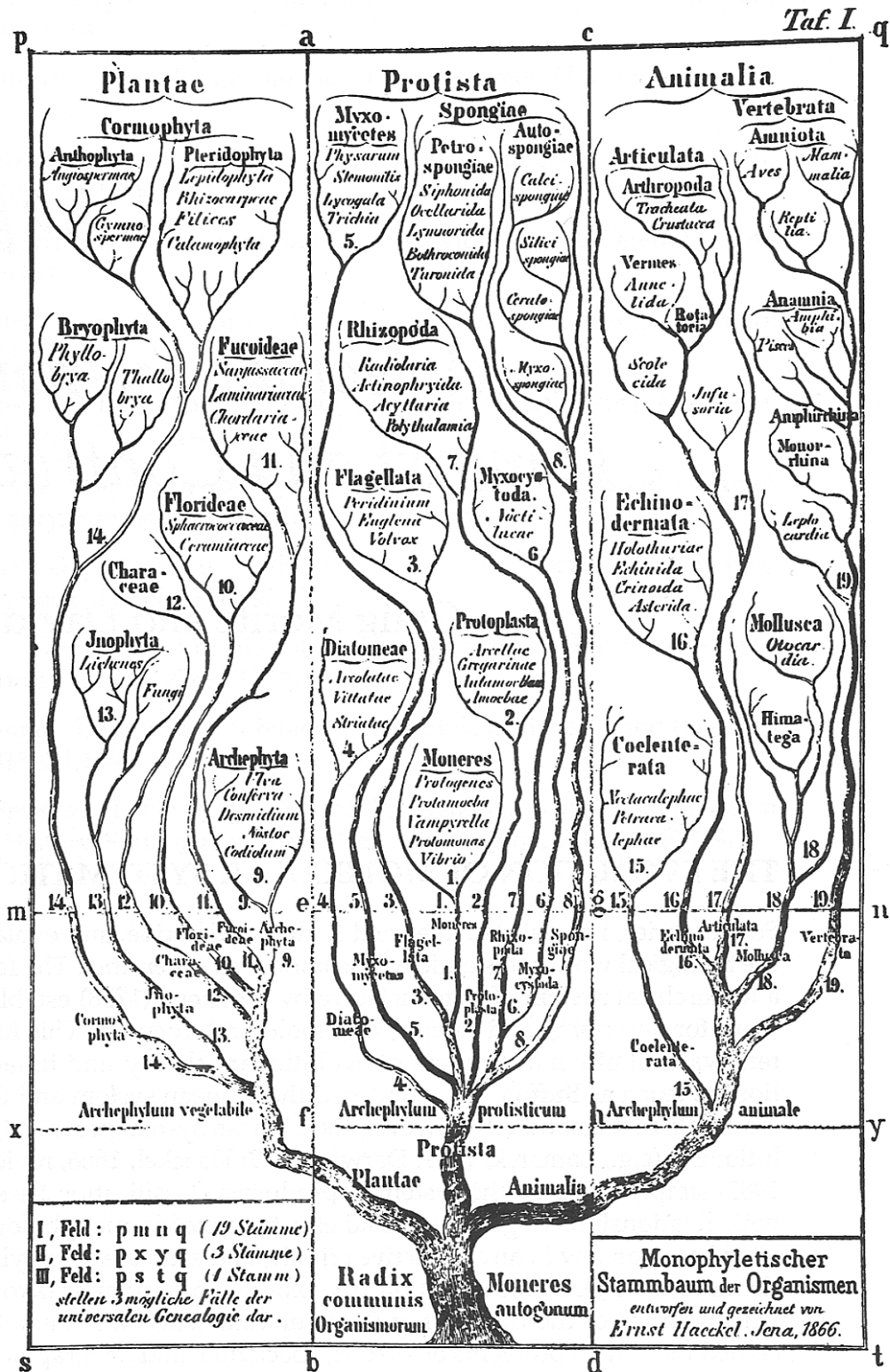


Figure 1.4. A phylogenetic tree proposed by Haeckel (1866).

1.4.1.1 Gene Trees and Species Trees

Initially, phylogenetic trees were built based on the morphology of organisms. Around 1960 molecular sequences were recognized as containing phylogenetic information and hence as valuable for tree building (section 1.3.1) (Zuckerkandl and Pauling, 1962; Zuckerkandl and Pauling, 1965b). A tree built based on sequence data is called a **gene tree** since it is a representation of the evolutionary history of genes, as opposed to organisms. Figures 1.1, 1.5, and 1.6 are illustrations of gene trees. A tree illustrating the evolutionary history of organisms is called a **species tree** (the tree in Figure 1.4 is a species tree based on morphology). In general, a gene tree does not reflect the evolutionary history of all the host species associated with the genes in the tree, as in Figure 1.6. This is due to the presence of gene duplications. Only in the complete absence of duplications can a gene tree correspond to a species tree, as shown in Figure 1.5.

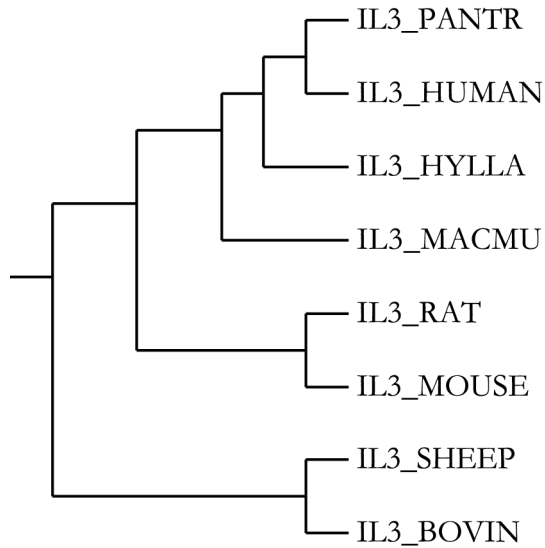


Figure 1.5. A gene tree of orthologs based on Interleukin-3 protein sequences.

Sequences are named with their SWISS-PROT identifiers. PANTR stands for *Pan troglodytes* (chimpanzee), HYLLA for *Hylobates lar* (common gibbon), MACMU for *Macaca mulatta* (rhesus macaque), BOVIN for *Bos taurus*. The tree is based on the Pfam (Bateman et al., 2000) alignment for Interleukin-3 (Accession number: PFO2059) (Burger et al., 1994). The tree was constructed by neighbor joining (section 1.4.2.5.2) from Felsenstein's PHYLIP package (Felsenstein, 2001). The distance used for neighbor joining were PAM-based maximum likelihood distances (section 1.4.2.2), calculated by PROTDIST from PHYLIP. The tree diagram was produced by ATV (chapter 2) (Zmasek and Eddy, 2001a).

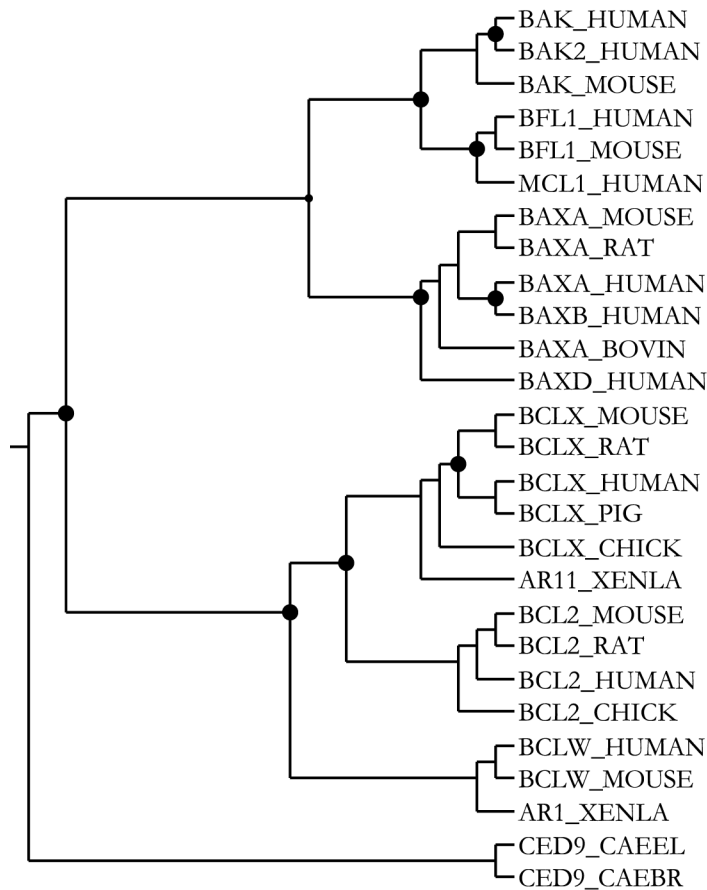


Figure 1.6. A gene tree of orthologs and paralogs based on Bcl-2 family protein sequences.

Circled nodes indicate gene duplication events. BOVIN stands for *Bos taurus*, XENLA for *Xenopus laevis*, CAEEL for *Caenorhabditis elegans*, CAEER for *Caenorhabditis briggsae*. The tree is based on the Pfam (Bateman et al., 2000) alignment for the apoptosis regulator proteins of the Bcl-2 family (Accession number: PF00452) (reviewed in Chao and Korsmeyer, 1998). The tree was constructed in the same manner as described for Figure 1.5. The speciation duplication inference algorithm SDI (chapter 3) (Zmasek and Eddy, 2001b) was used to determine the positions of the gene duplications. The tree diagram was produced by ATV (chapter 2) (Zmasek and Eddy, 2001a).

1.4.2 Methods for Phylogenetic Inference

Phylogenetic methods differ both in the type of input data and in the actual tree building method. The input data falls into one of two categories: discrete characters states, and similarities or distances. Discrete characters states include molecular sequence data, restriction endonuclease data, gene order data, or morphological character states. Similarities or distances are either measured directly (with hybridization experiments, for example), or discrete characters states are transformed into distances (section 1.4.2.2). In this work, the focus is on molecular sequence data (protein sequences). Trees are either built by an algorithmic method, which usually will yield one tree, or optimality criteria are used to evaluate the “likelihood” of a given tree or to select the “most likely” tree out of a set of given trees.

Before individual tree building methods are discussed, a historical overview is presented.

Initially, phylogenetic trees were built intuitively based on the morphology of organisms (as in Figure 1.4). In 1950, Hennig (1950; 1965; 1966) stated a systematic procedure (“**Hennig’s method**”) for inferring phylogenies from a set of morphological characters (“the rules for evaluating morphological characters as indicators of degree of phylogenetic relationship”). In this approach, hierarchical monophyletic groups of species (i.e. trees) are constructed based on knowledge of which states are ancestral (“plesiomorphous”) and which ones are derived (“apomorphous”). A weakness of Hennig’s method is that it cannot deal

with conflicting data. Although Hennig is well known for stating this approach, it had been applied previously (e.g. Mitchell, 1901).

Sokal and Sneath, were among the first to develop objective schemes for measuring the pairwise distance (similarity) between organisms (and to use computers to do so) (Sneath, 1957a; Sneath, 1957b; Sneath and Sokal, 1962; Sneath and Sokal, 1973; Sokal, 1956; Sokal, 1961; Sokal and Sneath, 1963). For example, to calculate a mean squared difference between two species, Sokal (1961) proposed the following formula:

$$\delta_{1,2}^2 = \frac{1}{n} \sum_{i=1}^n \left({}^1X_i - {}^2X_i \right)^2 \quad (1-2)$$

where 1X_i is the state code for species 1 for character i , and n is the number of characters. Similar formulas have been proposed earlier, in particular to measure the resemblance among anthropological material (mostly skulls) (e.g. Pearson, 1926). Especially, Rao (1952) used an intuitive approach to cluster analysis (see [below](#)) to produce tree-like diagrams of Indian tribes (based on anthropometrical characters).

Using pairwise distances between taxa as input, the **UPGMA** algorithm (section [1.4.2.5.1](#)) presented by Sokal and Michener (1958) clusters according to average similarity. The resulting clusterings correspond to a phylogenetic tree under the assumption of a molecular clock (see [above](#)).

Parsimony methods (section [1.4.2.4.1](#)) on continuous data were introduced by Edwards and Cavalli-Sforza (1963; 1964) in an effort to use gene

frequencies to make a phylogeny of human populations. Edwards and Cavalli-Sforza (1964) also introduced the **maximum likelihood** method (section 1.4.2.4.2), with gene frequencies as data.

Doolittle and Blomback (1964) used the number of amino acid changes in the sequences of fibrinopeptides from various artiodactyls to manually construct a phylogenetic tree.

Edwards and Cavalli-Sforza (1965) introduced a systematic method for **cluster analysis** which minimizes the within-cluster sum of squares of distances. It starts with all elements as members of the same cluster and proceeds to subdivide that cluster into successively smaller ones until each element is the only member of its own cluster. As an example, they applied their method to the same morphological data of Indian tribes as Rao (1952) to produce tree diagrams.

The first **discrete character parsimony** (section 1.4.2.4.2) method was introduced by Camin and Sokal (1965). Eck and Dayhoff (1966) were the first to devise a parsimony method for protein sequence data, which they employed to construct trees for cytochrome c and globins.

In 1967, both Fitch and Margoliash (1967), and Cavalli-Sforza and Edwards (1967) presented **least squares methods** (section 1.4.2.3.1) which find the optimal tree by minimizing the differences between the observed distances and the distances on the tree.

Neyman (1971) was the first to apply maximum likelihood to molecular sequences.

In the last 30 years, many more methods for phylogenetic inference based on molecular sequences have been published. Many of them are reviewed in

Durbin et al. (1998), Felsenstein (1982; 1988; 1996), Nei (1996), Page and Holmes (1998), Saitou (1996), and Swofford et al. (1996). Some of the major methods will be discussed later in the chapter.

1.4.2.1 Multiple Sequence Alignment

Before a tree building or evaluation approach can be applied on molecular sequences, the sequences in question have to be aligned. The quality of the multiple alignment will determine the quality of the tree. In other words, the alignment should reflect biology (positional homology) (Swofford et al., 1996).

Up until 1987, it was standard practice to construct multiple alignments manually. This is obviously very tedious and error prone. Early computer programs for multiple alignments were either too slow (such as standard dynamic programming approaches (Durbin et al., 1998) if more than three or four sequences were to be aligned) or not widely used [such as methods based on trying to find an alignment block or establishing a consensus sequences in a iterative manner (e.g. Bains, 1986; Johnson and Doolittle, 1986; Sobel and Martinez, 1986)]. More practical methods are based on a idea by Sankoff (1975) (progressive alignment). Progressive alignment starts with making an initial guess (guide tree) about the phylogenetic relationship of the sequences to be aligned. Then, it uses the branching order of this initial phylogenetic tree to align the sequences, starting with the most closely related pairs, and then gradually aligns these groups together. There are many variations of this approach, most of them using various heuristics to improve the basic progressive alignment (e.g.

Corpet, 1988; Feng and Doolittle, 1987; Gribskov et al., 1987; Hein, 1989). Currently, the most widely used programs for multiple alignment are CLUSTAL W (Thompson et al., 1997; Thompson et al., 1994) and PILEUP (Wisconsin Package; Genetics Computer Group, Madison, WI). CLUSTAL W uses neighbor joining (section 1.4.2.5.2) to build the guide tree, whereas PILEUP uses UPGMA (section 1.4.2.5.1). Various multiple sequence alignment programs are compared in Thompson (1999).

1.4.2.2 Pairwise Protein Distance Calculation

All possible pairwise distances have to be calculated from a multiple sequence alignment (section 1.4.2.1) prior to any tree building method or optimality criterion based on pairwise distances. Most textbooks do an excellent job at describing this and the corresponding models for DNA sequences (e.g. Swofford et al., 1996). Thus, and because this work concentrates on proteins, only amino acids sequences are considered in this section. For a review of some of the ideas presented here, see Lio and Goldman (1998).

The simplest method to measure the distance between two amino acid sequences is by their fractional dissimilarity p , defined as follows:

$$p = \frac{n_d}{n_d + n_s} \quad (1-3)$$

where n_d is the number of aligned sequence positions containing non-identical amino acids and n_s is the number of aligned sequence positions containing

identical amino acids. Unfortunately, this is unrealistic. For instance, it does not take into account superimposed changes (multiple mutations at the same sequence location) and the different chemical properties of amino acids (for example, changing leucine into isoleucine is more likely and should be weighted less than changing leucine into proline). To take into account superimposed changes, we can model amino acid substitution as a Poisson process (see section 1.4.2.4.2) (Nei, 1987), and calculate the distance between two amino acid sequences as follows:

$$d = -\ln(1 - p) \quad (1-4)$$

To better approximate distances calculated by Dayhoff et al. (1978), Kimura (1983) proposed the following correction:

$$d = -\ln(1 - p - 0.2p^2) \quad (1-5)$$

But even with this correction, realistic distances cannot be expected, in particular if p is larger than 0.7.

Karlin and Ghandour (1985) proposed a method of weights based on chemical, functional, charge and structural properties of the amino acids. Similarly, Feng et al. (1985) proposed weights based on the structural similarities and the ease of genetic interchange. The problem with models that attempt to incorporate “real” amino acid similarities is that they are based on groupings which are still artificial and do not reflection evolutionary processes (Jones et al., 1992).

A more realistic approach for estimating evolutionary distances is to apply maximum likelihood (section 1.4.2.2.1) to empirical amino acid replacement models, such as the well known PAM transition probability matrices (Dayhoff et al., 1978), or the BLOSUM matrices (Henikoff and Henikoff, 1992).

A **PAM transition probability matrix** is composed of 20×20 elements which correspond to the probabilities for each possible amino acid transition in one evolutionary time unit (see Table 1.1). The time unit used in the matrix is the time during which, on the average, one amino acid substitution per 100 residues takes place. This time unit is also called one PAM, PAM standing for “accepted point mutation” (“accepted” by natural selection). The PAM1 matrix (the PAM matrix for the evolutionary time unit of one PAM) has been constructed by Dayhoff et al. (1978) from empirical data for 71 groups of closely related proteins. First, phylogenetic trees for each of these groups were constructed by parsimony (section 1.4.2.4.2). Based on these trees, relative frequencies for substitutions among various amino acids were inferred. These frequencies were then normalized into values that represented the probability that 1 amino acid in 100 would undergo change, resulting in the PAM1 matrix shown in Table 1.1. Other probability matrices for proteins that had undergone x amino acid substitutions per 100 residues were then derived by multiplying PAM1 by itself x times (see, for example, Mirsky, 1982), resulting in matrices such as PAM50 or PAM250.

A different approach was used by Henikoff and Henikoff (1992) for the construction of the **BLOSUM** matrices. These matrices were derived from local, ungapped alignments of distantly related protein sequences. Matrices in this

series are also identified by a number (e.g. BLOSUM250). But in contrast to the PAM matrices, these numbers refer to the minimum percentage identity of the blocks of aligned amino acids used for matrix construction.

Many more rate matrices have been developed. Some examples of more recently developed ones are: the JTT matrix which was built in the same way as the PAM1 matrix but on larger data sets (Jones et al., 1992), the mtREV matrix which was built specifically for proteins encoded by mitochondrial DNA (Adachi and Hasegawa, 1996), the VT matrix (Mueller and Vingron, 2000), and the WAG matrix (Whelan and Goldman, 2001).

1.4.2.2.1 A maximum Likelihood Approach to Distance Calculation

The problem of finding the evolutionary distance between two sequences using rate matrices can be described as follows. Given an instantaneous rate matrix \mathbf{M} (such as PAM1) and an alignment A of two sequences a and b , we would like to determine the “most likely” evolutionary distance or time between a and b . This is a maximum likelihood approach. The likelihood L_H of a hypothesis H (an evolutionary distance, for example) given same data D (an alignment, for example) is the probability of D given H :

$$L_H = P(D | H) \tag{1-6}$$

Maximum likelihood approaches estimate hypotheses (or parameters) by maximizing L_H for a given D .

In order to apply the maximum likelihood approach to distance calculations we need to be able to calculate a transition probability matrix $\mathbf{P}(t)$ for a finite time interval (or evolutionary distance) t , given a transition probability matrix \mathbf{M} for a unit of time (such as PAM1); $\mathbf{P}(1) = \mathbf{M}$.

According to Kishino et al. (1990) $\mathbf{P}(t)$ is well approximated by:

$$\mathbf{P}(t) = \mathbf{M}^t = e^{t\mathbf{R}} \quad (1-7)$$

where \mathbf{R} is a function of the eigenvalues and eigenvectors of \mathbf{M} :

$$\mathbf{R} = \mathbf{U} \begin{bmatrix} \lambda_1 & & & \\ & \ddots & & \\ & & \ddots & \\ & & & \lambda_{20} \end{bmatrix} \mathbf{U}^{-1} \quad (1-8)$$

$$\lambda_i = \frac{0.01}{1 - \sum_{i=1}^{20} f_i M_{ii}} \log \rho_i \quad (1-9)$$

$1 - \sum_{i=1}^{20} f_i M_{ii}$ corresponds to the number of substitutions in a unit time; f_i ($i=1, \dots, 20$)

is the normalized frequency of amino acid i (e.g. Table 22 in Dayhoff et al., 1978);

and ρ_i ($i=1, \dots, 20$) is an eigenvalue of \mathbf{M} .

\mathbf{U} is a matrix whose columns are the eigenvectors \mathbf{u}_i ($i=1,\dots,20$) of \mathbf{M} :

$$\mathbf{U} = (\mathbf{u}_1, \dots, \mathbf{u}_{20}) \quad (1-10)$$

Components of $\mathbf{P}(t)$ can be written as:

$$P_{ij}(t) = \sum_{k=1}^{20} c_{ijk} e^{t\lambda_k} \quad (1-11)$$

where (Adachi and Hasegawa, 1996):

$$c_{ijk} = U_{ik} U_{kj}^{-1} \quad (1-12)$$

Using (1-6), we can obtain the maximum likelihood estimate for t through the Newton-Raphson method or the bisection method (see Press et al., 1992), for which we need the following derivatives:

$$\frac{d}{dt} P_{ij}(t) = \sum_{k=1}^{20} c_{ijk} \lambda_k e^{t\lambda_k} \quad (1-13)$$

$$\frac{d^2}{dt^2} P_{ij}(t) = \sum_{k=1}^{20} c_{ijk} \lambda_k^2 e^{t\lambda_k} \quad (1-14)$$

These calculation can be made more time efficient if an initial guess for t is provided, possibly by an equation similar to (1-5).

The resulting distance computation is quite slow, but appears to be adequate (Felsenstein, 1996). Yet, one unrealistic assumption is still being made: all amino acid positions are assumed to change at the same rate. A more realistic model allows for a gamma distribution of evolutionary rates among sites as described in Jin and Nei (1990) or Nei et al. (1976). Unfortunately this added realism comes at a huge loss in time efficiency.

	ORIGINAL AMINO ACID (i)																			
	Ala	Arg	Asn	Asp	Cys	Gln	Glu	Gly	His	Ile	Leu	Lys	Met	Phe	Pro	Ser	Thr	Trp	Tyr	Val
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
REPLACEMENT AMINO ACID (j)	A	9867	2	9	10	3	8	17	21	2	6	4	2	6	2	22	35	32	0	18
	R	1	9913	1	0	1	10	0	0	10	3	1	19	4	1	4	6	1	8	0
	N	4	1	9822	36	0	4	6	6	21	3	1	13	0	1	2	20	9	1	4
	D	6	0	42	9859	0	6	53	6	4	1	0	3	0	0	1	5	3	0	0
	C	1	1	0	0	9973	0	0	0	1	1	0	0	0	1	5	1	0	3	2
	Q	3	9	4	5	0	9876	27	1	23	1	3	6	4	0	6	2	2	0	0
	E	10	0	7	56	0	35	9865	4	2	3	1	4	1	0	3	4	2	0	1
	G	21	1	12	11	1	3	7	9935	1	0	1	2	1	1	3	21	3	0	5
	H	1	8	18	3	1	20	1	0	9912	0	1	1	0	2	3	1	1	1	4
	I	2	2	3	1	2	1	2	0	0	9872	9	2	12	7	0	1	7	0	33
	L	3	1	3	0	0	6	1	1	4	22	9947	2	45	13	3	1	3	4	15
	K	2	37	25	6	0	12	7	2	4	1	9926	20	0	3	8	11	0	1	1
	M	1	1	0	0	0	2	0	0	0	5	8	4	9874	1	0	1	2	0	4
	F	1	1	1	0	0	0	0	1	2	8	6	0	4	9946	0	2	1	3	28
	P	13	5	2	1	1	8	3	2	5	1	2	2	1	1	9926	12	4	0	2
	S	28	11	34	7	11	4	6	16	2	2	1	7	4	3	17	9840	38	5	2
	T	22	2	13	4	1	3	2	2	1	11	2	8	6	1	5	32	9871	0	9
	W	0	2	0	0	0	0	0	0	0	0	0	0	1	0	1	0	9976	1	0
	Y	1	0	3	0	3	0	1	0	4	1	1	0	0	21	0	1	2	9945	1
	V	13	2	1	1	3	2	2	3	3	57	11	1	17	1	3	2	10	0	9901

Table 1.1. PAM1, a transition probability matrix for the evolutionary distance of 1 PAM.

An element of this matrix, M_{ij} , gives the probability that the amino acid in column i will be replaced by the amino acid in row j after a given evolutionary interval, in this case 1 PAM. Thus, there is a 0.56% probability that Asp will be replaced by Glu. To simplify the appearance, the elements are shown multiplied by 10,000. Adapted from Dayhoff et al (1978).

1.4.2.3 Optimality Criteria Based on Pairwise Distances

Optimality criteria are used to assign a score to a given tree. For tree reconstruction, all possible trees (or an “appropriate” subset thereof) have to be proposed by a proposal mechanism and then searched for the tree with the best score. Many different (heuristic) algorithms for proposing and searching trees exist. These algorithms are not discussed here. For a review see Swofford et al. (1996). Since usually a large number of trees have to be evaluated (see equation 1-1), optimality-criteria based methods tend to be time consuming.

For both optimality criteria as well as for algorithmic methods based on pairwise distances, it is crucial to establish whether the pairwise distance data is additive or ultrametric as well – which methods are applicable depends on this distinction. In the following, the terms additive and ultrametric are defined.

If we could determine the true evolutionary distances from a given alignment, these distances would have the property of **additivity**, as illustrated in Figure 1.7. In this case, a tree exists for which the sum of branch lengths between each pair of external nodes (e.g. $a+e+b$) precisely equals the evolutionary distance between them (e.g. d_{AB}). Additive distances satisfy the four-point condition (Buneman, 1971): for any four external nodes A, B, C, and D:

$$d_{AC} + d_{BD} \leq \max(d_{AB} + d_{CD}, d_{AD} + d_{BC}) \quad (1-15)$$

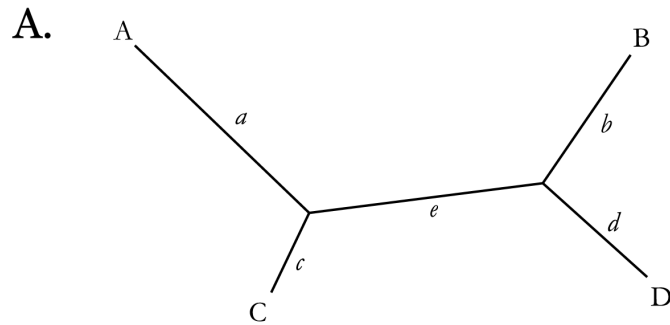
Unfortunately, due to the finite amount of data available, stochastic errors will prevent the estimated distances from fitting exactly onto a tree. Least squares

optimality criteria (section 1.4.2.3.1) measure the fit of the estimated evolutionary distances to an additive tree. Minimum evolution optimality criteria (section 1.4.2.3.2) and the neighbor joining algorithm (section 1.4.2.5.2) assume additivity in their input data.

Ultrametric distances (Figure 1.7.B) are a subset of additive distances. They adhere to the three-point condition: for any three external nodes A, B, and C:

$$d_{AC} \leq \max(d_{AB}, d_{BC}) \quad (1-16)$$

In terms of phylogenetic trees, an ultrametric tree is an additive tree under the additional constraint of a (constant) molecular clock (see above). The UPGMA algorithm (section 1.4.2.5.1) constructs an ultrametric tree from ultrametric distances.



Additive properties:

$$d_{AB}=a+e+b$$

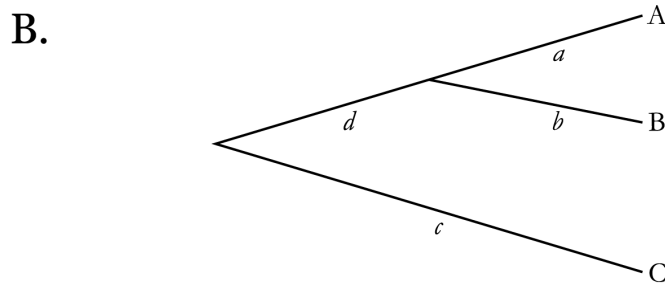
$$d_{AC}=a+c$$

$$d_{AD}=a+e+d$$

$$d_{BC}=b+e+c$$

$$d_{BD}=b+d$$

$$d_{CD}=c+e+d$$



Additive properties:

$$d_{AB}=a+b$$

$$d_{AC}=a+d+c$$

$$d_{BC}=b+d+c$$

Ultrametric properties:

$$a=b$$

$$c=d+a$$

$$c=d+b$$

Figure 1.7. Additive and ultrametric trees.

An additive tree is shown in A. The tree in B also exhibits ultrametric properties. Adapted from Swofford et al. (1996).

1.4.2.3.1 Least Squares

An optimal tree according to least squares (and related) criteria is selected by minimizing the disagreement E between the tree and the estimated pairwise distances (estimated from a multiple alignment):

$$E = \sum_{i=1}^{N-1} \sum_{j=i+1}^N w_{ij} |d_{ij} - p_{ij}|^{\alpha} \quad (1-17)$$

where N is the number of external nodes, d_{ij} is the distance estimate between sequences i and j and p_{ij} the length of the path connecting external nodes i and j in the given tree. Setting $\alpha=2$ represents a least squares criterion. For $\alpha=1$, the absolute differences will be minimized. Setting the weighting $w_{ij}=1$ assumes that all distance estimates are subject to the same magnitude of error and corresponds to a unweighted least squares criterion (Cavalli-Sforza and Edwards, 1967).

Setting $w_{ij} = \frac{1}{d_{ij}^2}$ assumes that all estimates are uncertain by the same percentage (Fitch and Margoliash, 1967).

1.4.2.3.2 Minimum Evolution

By means of the minimum evolution method, branch lengths are fitted to a tree according to a unweighted least squares criterion, but the optimality criterion to evaluate and compare trees is to minimize the sum of all branch lengths L (Kidd and Sgaramella-Zonta, 1971; Rzhetsky and Nei, 1992; Rzhetsky and Nei, 1993):

$$L = \sum_{k=1}^{2N-3} b_k \quad (1-18)$$

where N is the number of external nodes, and b_k is a branch length. The minimum evolution tree is the one which minimizes L .

1.4.2.4 Optimality Criteria Based on Character Data

Character data based methods work directly on molecular sequences and thus do not require the calculation of pairwise distances.

1.4.2.4.1 Maximum Parsimony

Maximum parsimony criteria are based on the principle of Occam's razor, which states "One should not increase, beyond what is necessary, the number of entities required to explain anything". The central idea is that the preferred evolutionary tree requires the smallest number of evolutionary changes to explain the differences observed among the sequences under study. Hypothetical

sequences are assigned to the ancestral (internal) nodes in such a way as to minimize the total number of substitutions. In traditional parsimony (Fitch, 1971) any substitution is assigned a cost of 1, whereas identical residues have a cost of 0. Weighted parsimony (Sankoff and Kruskal, 1983) assigns different costs to different types of substitutions. Maximum parsimony is in fact an approximation to maximum likelihood, as pointed out by Felsenstein (1981b) and reviewed in Durbin et al. (1998).

1.4.2.4.2 Maximum Likelihood

Probabilistic methods can be used to assign a likelihood to a given tree and therefore allow the selection of the tree which is most likely given the observed sequences (Edwards and Cavalli-Sforza, 1964; Felsenstein, 1981a; Kashyap and Subas, 1974; Neyman, 1971).

The probability for one residue a to change to b in time t along a branch of a tree is given by $P(b|a,t)$. Its actual calculation is dependent on what model for sequence evolution is used. The simplest model is a Poisson process, which assumes that all changes between amino acids occur at the same rate and that the equilibrium frequencies of all amino acids are equal. The probabilities for this model are:

$$\begin{aligned}
P(b|a,t) &= \frac{1}{20} + \frac{19}{20}e^{-\mu t} \quad \text{for } a = b \\
P(b|a,t) &= \frac{1}{20} + \frac{1}{20}e^{-\mu t} \quad \text{for } a \neq b
\end{aligned} \tag{1-19}$$

where μ is the substitution rate. In practice, more sophisticated models, such as the PAM matrices (see [above](#)), are usually used (Adachi et al., 1993; Adachi and Hasegawa, 1992; Strimmer and von Haeseler, 1996).

The likelihood of a tree T with branch lengths t_\bullet , given the observed sequences x_\bullet , can be written as $P(x_\bullet|T, t_\bullet)$ (see Durbin et al., 1998). For one site (one position in the multiple alignment of x_\bullet), the likelihood can be calculated as follows:

$$P(x_1, \dots, x_n | T, t_\bullet) = \sum_{a_{n+1}, \dots, a_{2n-1}} q_{a_{2n-1}} \prod_{i=n+1}^{2n-2} P(a_i | a_{\alpha(i)}, t_i) \prod_{i=1}^n P(x_i | a_{\alpha(i)}, t_i) \tag{1-20}$$

x_1, \dots, x_n are the amino acid residues at the n external nodes of T . $\alpha(i)$ denotes the parent node of i . $P(a_i|a_{\alpha(i)}, t_i)$ is the probability of observing residue a_i at internal node i , given $a_{\alpha(i)}$ at its parent node and branch length t_i . These probabilities are multiplied over all internal nodes (labeled from $n+1$ to $2n-1$). $P(x_i|a_{\alpha(i)}, t_i)$ is probability of observing x_i at an external node i , given $a_{\alpha(i)}$ at its parent node and branch length t_i . These probabilities are multiplied over all external nodes (labeled from 1 to n). Since we generally do not know the residues at the internal

nodes, we have to sum over all possible assignments of residue a_k to internal nodes k . q_a is the equilibrium frequency of a .

The probabilities at each internal node can be calculated based solely on the probabilities at its direct child nodes. Therefore, the complete probability can be computed by working up the tree from the external nodes in post order traversal, as described by Felsenstein (1981a).

To calculate the likelihood for the complete alignment, the likelihood values for each site are multiplied with each other.

As for ML methods for distance calculation, a limitation of the methods described above is that they assume the same rate of evolution for all positions. This limitation has been removed from nucleotide sequence ML methods, using gamma distributed rates or hidden Markov model approaches (Felsenstein and Churchill, 1996; Yang, 1993; Yang, 1994; Yang, 1995).

1.4.2.5 Algorithmic Methods Based on Pairwise Distances

As mentioned above, these algorithmic approaches produce one tree, taking pairwise distances as input.

1.4.2.5.1 UPGMA

UPGMA (Sokal and Michener, 1958) stands for unweighted pair group method using arithmetic averages. This clustering algorithm produces ultrametric, rooted trees based on ultrametric distances (see [above](#)). If the input

distances are not ultrametric (no molecular clock), then UPGMA might reconstruct an incorrect tree. UPGMA first places each sequence in its own cluster and then iteratively clusters together the two most similar clusters, assigning the new cluster the weighted average distances of its members. The main advantage of this method is its speed [the overall time complexity of UPGMA is $O(N^2)$].

1.4.2.5.2 Neighbor Joining

As opposed to UPGMA, neighbor joining (NJ) is not misled by the absence of a molecular clock. It recreates the correct additive tree as long as the input distances are additive (Studier and Keppler, 1988). NJ is effective even if additivity is only approximated (Atteson, 1997). Trees produced by NJ are unrooted. The NJ method was introduced by Saitou and Nei (1987).

The NJ algorithm is as follows (Studier and Keppler, 1988; Swofford et al., 1996):

Input: $N \times N$ matrix of estimated pairwise distances.

Output: One unrooted, fully resolved binary tree.

1. For each pair i, j of N OTUs, compute (for all $j > i$):

$$S_{ij} = (N - 2)D_{ij} - R_i - R_j \quad (1-21)$$

where D_{ij} is the estimated distance between i and j , and:

$$R_i = \sum_{k=1}^N D_{ik} \quad (1-22)$$

2. Pick a pair i, j for which S_{ij} is minimal. Create a new node u whose three branches join nodes i, j , and the rest of the tree (see Figure 1.8). The branch lengths from u to i and j are:

$$b_{iu} = \frac{D_{ij}}{2} + \frac{R_i - R_j}{2(N - 2)} \quad (1-23)$$

$$b_{ju} = D_{ij} - b_{iu} \quad (1-24)$$

3. Compute the pairwise distances from u to each other OTU (for all $k \neq i, j$):

$$D_{ku} = \frac{1}{2}(D_{ik} + D_{jk} - D_{ij}) \quad (1-25)$$

4. Remove distances to nodes i and j and decrease N by 1.

5. If more than two nodes remain, go back to step A. Otherwise join the two remaining nodes with a branch of length D_{ij} .

The overall time complexity of NJ is $O(N^3)$ (Studier and Keppler, 1988).

Step 2. can be explained as follows (Saitou, 1996). Starting with a star-like tree (no clustering) of N OTUs we would like to choose the one pair of OTUs that results in the smallest sum of branch lengths if they were to be joined as neighbors (two nodes are called neighbors if they are connected through a single internal node). For the tree in Figure 1.8, OTUs i and j were chosen to be joined as neighbors. Minimizing equation (1-21) allows us to find the two OTUs joining of which results in the smallest sum of branch lengths. In this lies the crucial difference to UPGMA: in UPGMA, the pair for which D_{ij} minimal is picked in each iteration cycle. In NJ, the pair for which – informally speaking – D_{ij} is minimized and the distance to all other OTUs is maximized is selected in each cycle.

NJ is very fast, suitable for large data sets, and reasonably accurate as long as enough sequence data is available for analysis and the internal branches are not small compared to the length of the branches leading to the leaves (Hillis et al. 1994). As mentioned above, the crucial advantage of NJ over UPGMA is that it does not have the precondition of a molecular clock.

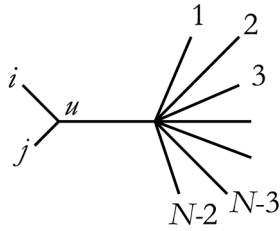


Figure 1.8. Tree of N external nodes in which i and j are neighbors.

Adapted from Saitou (1996).

1.4.2.6 Bootstrapping

Bootstrapping is a resampling with replacement. It provides us with numbers that indicate how much we should trust a particular feature of a phylogenetic tree (e.g. man and gorilla forming a clade which does not include the platypus) (Felsenstein, 1985; Mueller and Ayala, 1982). It works as follows (see Figure 1.9 for a simple example). A resampled multiple alignment is created by randomly picking columns from a given multiple alignment (bootstrap resample 1 in Figure 1.9 is created by picking the columns from the original alignment in the following order: 2, 2, 6, 5, 5, 1). Since the resampling is with replacement, a column from the original alignment can appear multiple times in the resampled alignment (and this is the point, since simply changing the order of the columns would not make a difference for tree inference). In practice, the original alignment is resampled many times (100 to 1000 times). For each resampled alignment, a phylogenetic tree is then inferred. The frequency with which a particular feature appears is taken as a measure of the confidence we can have in it. Oftentimes, each node of a phylogenetic tree is associated with a

bootstrap value. These numbers indicate the frequency with which that particular node appears (the presence of a binary node is determined by the fact that a binary node separates all OTUs of a tree in precisely two groups: those which are children of this node and those which are not).

Original sequence alignment:

Sequence 1: ARNDCQ

Sequence 2: VRNDCQ

123456

Bootstrap resample 1:

Sequence 1: RRQCCA

Sequence 2: RRQCCV

226551

Bootstrap resample 2:

Sequence 1: AQCDCQ

Sequence 2: VQCDCQ

165456

Figure 1.9. An example of the bootstrap resampling procedure.

Two bootstrap resamples of the original amino acid multiple alignment are shown. It is indicated which columns of the original alignment were picked to create the two resamples.

1.5 Overview

The objective of this work is the development and evaluation of computational methods for sequence function prediction based on molecular evolution (phylogenomics). A phylogenomic analysis of a sequence with unknown function can be divided into three steps: First, the domain composition of the query is determined, for instance by using the HMMER software (Eddy, 2000) and the Pfam domain database (Bateman et al., 2000). Second, a gene tree is inferred for each domain of the query sequence based on the appropriate Pfam alignments. For this step, the tree building methods discussed in section 1.4.2 are employed. Third, various inferences about the query are made based on the topology (and possibly branch lengths) of the gene tree(s). It is this third step which is the focus of this work.

Visual inspection of gene trees annotated with functional information and “duplication” or “speciation” on internal nodes can be an easy and intuitive approach to make phylogenomic inferences. ATV (A Tree Viewer), a computer program for this purpose was developed during the course of this work and is presented in chapter 2. While ATV can be used as a general purpose tree display tool [it can display any tree described in the commonly used “New Hampshire” format (Felsenstein, 2001)], the primary design goal was to create a means for manual phylogenomics. ATV allows the display of trees annotated with concepts related to sequence function (EC-numbers and natural language descriptions). Internal nodes can be shown as either speciation or gene duplication events.

Duplications and speciations can be inferred with the SDI algorithm presented in chapter 3 (in fact, newer versions of ATV include SDI). ATV is also part of RIO, the automated system for phylogenomics presented in chapter 4 (for example, ATV can display, and color according to the orthology bootstrap values calculated by RIO).

As stated in section 1.3.2.1, gene duplication is thought to oftentimes lead to the generation of new sequence functionality. Hence, knowing which nodes of a gene tree represent duplication events is important for any functional analysis based on phylogeny. The only general means by which duplications and speciations on a gene tree can be inferred is by comparing it to a trusted species tree. SDI (Speciation Duplication Inference), a simple but fast algorithm for this purpose, was developed and evaluated during the course of this work. SDI is presented in chapter 3.

Once duplications and speciations on a gene tree are known, sequences can be divided into orthologs and paralogs relative to a query sequence whose function is to be inferred. Ideally, functional annotation is then transferred from the orthologs to the query. Unfortunately, gene trees can be unreliable. Therefore, it is advantageous to make phylogenomic inferences over a set of bootstrap resampled trees (see section 1.4.2.6). The frequency with which a particular sequence appears orthologous to the query sequence is taken as a measure of the confidence we can have in that particular orthology. This is implemented in the RIO (Resampled Inference of Orthologs) procedure described and evaluated in chapter 4. Besides orthology, RIO implements further measures based on the topology of gene trees. These measures – “super-

orthology”, “ultra-paralogy”, and “subtree-neighbors” – are introduced and justified in chapter 4 as well.

2 ATV: Display and Manipulation of Annotated Phylogenetic Trees

Christian M. Zmasek and Sean R. Eddy

Howard Hughes Medical Institute

Department of Genetics

Washington University School of Medicine

St. Louis, MO 63110, USA

Published as:

Zmasek, C. M. and Eddy, S. R. (2001) “ATV: display and manipulation of annotated phylogenetic trees”. *Bioinformatics*, Vol. 17, no. 4, pages 383-384.

On-line version is available at:

[<http://bioinformatics.oupjournals.org/cgi/content/abstract/17/4/383>]

2.1 Abstract

Summary: A Tree Viewer (ATV) is a Java tool for the display and manipulation of annotated phylogenetic trees. It can be utilized both as a standalone application and as an applet in a web browser.

Availability: ATV is available via WWW at [\[http://www.genetics.wustl.edu/eddy/atv/\]](http://www.genetics.wustl.edu/eddy/atv/) and via FTP at [\[ftp://ftp.genetics.wustl.edu/pub/eddy/software/forester.tar.Z\]](ftp://ftp.genetics.wustl.edu/pub/eddy/software/forester.tar.Z)

Contact: zmasek@genetics.wustl.edu

2.2 Introduction

Many proteins belong to large families consisting of subfamilies with different biological functions. This complicates efforts to infer the function of new proteins by computational sequence analysis. Neither of the two main sequence analysis methods handle large protein families satisfactorily in high-throughput automated annotation. Pairwise sequence similarity searches, exemplified by BLAST (Altschul et al., 1990), lead to overly specific annotations. A new sequence in a protein family is always “most similar” to something, so it is difficult to recognize when the new sequence is the pioneer member of a novel functional subfamily. Profile search methods, exemplified by HMMER (Eddy, 2000), lead to overly general annotations. They recognize that a new sequence fits a general profile of a family, but do not attempt to subclassify the sequence at all.

Phylogenetic inference is a sensible approach to sub-classifying sequences, by grouping them hierarchically into evolutionary clades. The use of phylogenetic inference to improve genome sequence annotation has been termed “phylogenomics” by Eisen (1998b). A key idea of phylogenomics is to distinguish sequences that have diverged by speciation (orthologs) from sequences that have diverged by duplication (paralogs). Although orthology does not equate with functional conservation, as is sometimes assumed, orthologs often do conserve more aspects of a protein’s function than paralogs do. We are working on automating a phylogenomic approach to improve Pfam-based annotations.

During phylogenomic analysis, gene trees are annotated with various data. Nodes are annotated as either a gene duplication or a speciation, and subtrees are

annotated according to sequence function (as description and/or EC number). In addition, information about species (as name and/or taxonomy ID) and sequence names, branch lengths, and bootstrap values are likely to be present. We needed a tool for visualizing heavily annotated phylogenetic trees. Although a variety of excellent tree browsers exist, including DRAWTREE from the PHYLIP package (Felsenstein, 2001), TREEVIEW (Page, 1996), NIFAS [<http://www.cgr.ki.se/Pfam/nifas.html>], NJPLOT (Perriere and Gouy, 1996), and Phylodendron [<http://iubio.bio.indiana.edu/soft/molbio/java/apps/trees/>] none of them exactly suited our annotation needs. Hence, we developed our own design.

2.3 Features

ATV is mouse and menu driven. The user can choose which data elements to display on the tree. All the data fields associated with nodes can be edited. The tree can be rerooted on any branch. ATV allows visualization of very large trees (>500 sequences): the user can display any subtree of the tree, zoom in or out, or collapse any subtree into a single node. The applet hyperlinks to SWISS-PROT entries for sequences with a SWISS-PROT name. Branches can be colored according to likelihood values associated with them. The Swing version (see below) of the application allows printing trees in color. Depending on the user's environment, it also allows tree images to be exported as PostScript or PDF files (which in turn gives the user the opportunity to employ graphics software to manipulate tree images beyond the capabilities of ATV). An example of ATV displaying an annotated tree is shown in Figure 2.1.

Trees can be read and saved in the standard “New Hampshire” format (Felsenstein, 2001), but this format is not suitable for storing annotated trees. Currently we use a simple extension of the format that we call “New Hampshire eXtended” format (NHX). In NHX, additional tag/value pairs are used to associate annotation with nodes (e.g. “:E=” is a tag for a EC number, “:S=” is a tag for a species name). In the long term, we envision replacing NHX with a structured markup language, such as the XML document type definition for the description of taxonomic relationships described in Gilmour (2000).

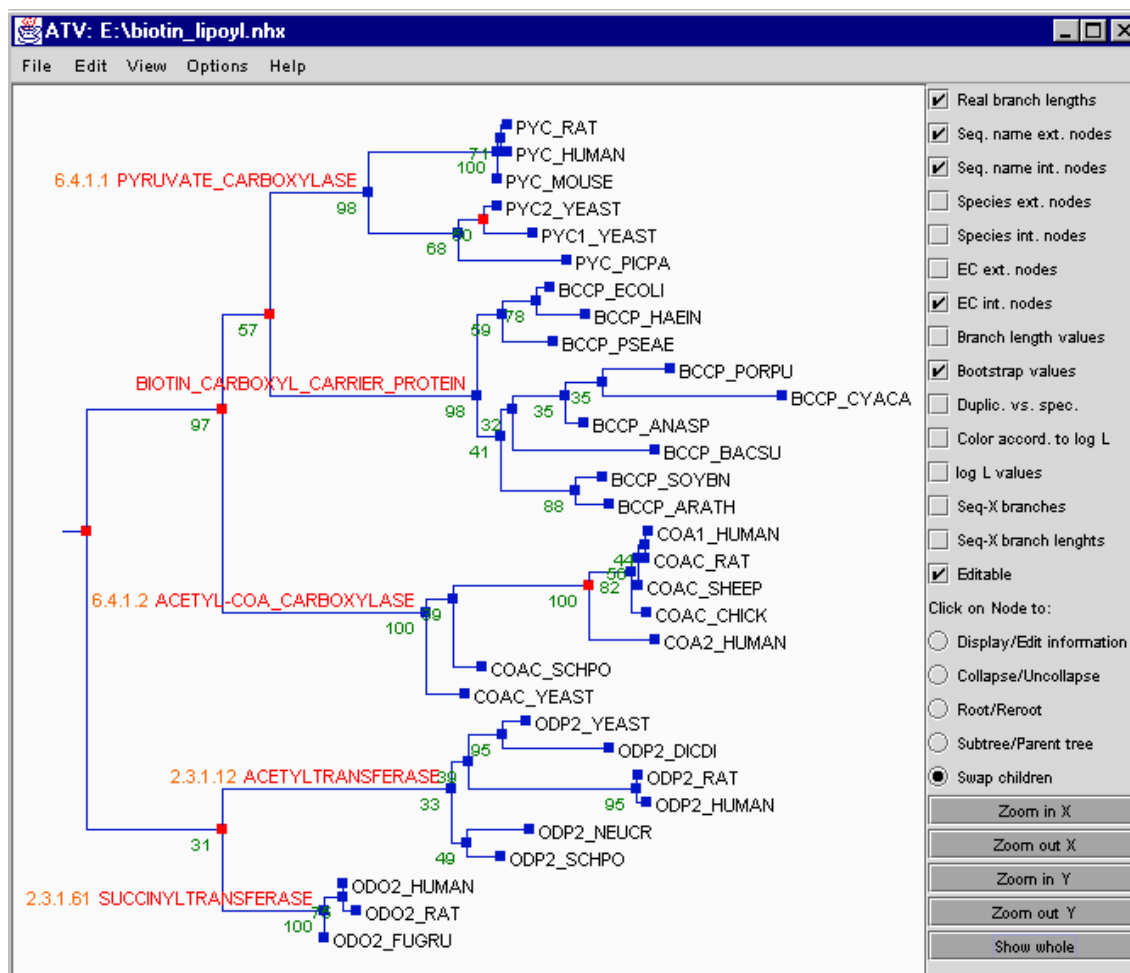


Figure 2.1. ATV displaying a phylogenetic tree of biotin-requiring enzymes.

Red nodes indicate duplications, green numbers represent bootstrap values, orange numbers are EC numbers, and the functional description of subtrees is in red. The check boxes in the right side panel are used to choose which information is displayed, whereas the radio buttons are used to determine the behavior for node clicking.

2.4 Implementation

ATV is coded in Java, for portability reasons. ATV can be used either as an applet in a web browser or as a standalone application. ATV should run on any platform for which a Java 1.1.x runtime environment is available. It has been tested on Red Hat Linux 6.1, SGI IRIX 6.5, Sun Solaris 5.6, and Microsoft Windows 95B and Windows NT Workstation 4.0 using various Java runtime environments from Sun Microsystems and Silicon Graphics. Two versions of ATV exist. One version uses Swing graphics classes, and is less portable but more aesthetically pleasing. The other version uses basic AWT (Advanced Windowing Toolkit) and is more portable. It is straightforward to incorporate ATV and forester into other Java applications.

ATV is freely available under a BSD open source license. The ATV distribution includes all source code files, as well as extensive documentation (including a definition of the NHX format).

2.5 Acknowledgements

We would like to thank Peter Ernst for useful additions. This work was supported primarily by a grant from Pharmacia Corporation, and also by the Howard Hughes Medical Institute and the NIH National Human Genome Research Institute.

2.6 The New Hampshire X Format (NHX)

(This section is not present in the original publication text.)

NHX is based on the New Hampshire (NH) standard (also called "Newick tree format") (Felsenstein, 2001). It has the following extensions (compared to NH used in the PHYLIP package):

- it introduces tags to associate various data fields with a node of a phylogenetic tree
- both internal and external nodes can be tagged
- number of children per node is at least two (allows polytomous trees)
- the tree is assumed to be rooted if the deepest node is a bifurcation
- the order of the tags does not matter, with the exception that the sequence name must be first (if assigned)
- the length of all character string based data is unlimited (name, species, EC number)
- Comments between '[' and ']' are removed (unless the opening bracket is followed by "&&NHX")

In order to remain compatible with the NEXUS format (Maddison et al., 1997), all fields except sequence name and branch length (in other words, all fields eXtending NH) must be wrapped by "[&&NHX" and "]". E.g. "ADH1:0.11[&&NHX:S=human:E=1.1.1.1]". In contrast to its name, NHX also has restrictions compared to Felsenstein's definition of the NH format: "Empty"

nodes are not allowed (e.g. "(,(),)" is not acceptable). The following characters cannot be part of names: '(' ')' '[' ']' ',' ':' as well as white spaces. The tags are listed in Table 2.1.

TAG	VALUE	MEANING
no tag	String	sequence name of this node (MUST BE FIRST, IF ASSIGNED)
:	double	branch length to parent node (MUST BE SECOND, IF ASSIGNED)
:B=	integer	bootstrap value at this node (does not apply to external nodes)
:S=	String	species name of the species/phylum at this node
:T=	integer	NCBI taxonomy ID of the species/phylum at this node
:E=	String	EC number at this node
:D=	'Y' or 'N'	'Y' if this node represents a duplication event – 'N' if this node represents a speciation event (does not apply to ext nodes)
:O=	integer	orthologous to this external node
:SO=	integer	"super orthologous" (no duplications on paths) to this external node
:L=	float	log likelihood value on parent branch
:Sw=	'Y' or 'N'	placing a subtree on the parent branch of this node makes the tree significantly worse according to Kishino/Hasegawa test (or similar)
:Co=	'Y' or 'N'	Collapse this node when drawing the tree (default is not to collapse)

Table 2.1. Tags used in the NHX format.

The following is the NHX description of the tree shown in Figure 2.2:

```
(( (ADH2:0.1 [&&NHX:S=human:E=1.1.1.1], ADH1:0.11 [&&NHX:S=human:E=1.1.1.1]) :0.05 [&&NHX:S=Primates:E=1.1.1.1:D=Y:B=100], ADHY:0.1 [&&NHX:S=nematode:E=1.1.1.1], ADHX:0.12 [&&NHX:S=insect:E=1.1.1.1]) :0.1 [&&NHX:S=Metazoa:E=1.1.1.1:D=N], (ADH4:0.09 [&&NHX:S=yeast:E=1.1.1.1], ADH3:0.13 [&&NHX:S=yeast:E=1.1.1.1], ADH2:0.12 [&&NHX:S=yeast:E=1.1.1.1], ADH1:0.11 [&&NHX:S=yeast:E=1.1.1.1]) :0.1 [&&NHX:S=Fungi]) [&&NHX:E=1.1.1.1:D=N];
```

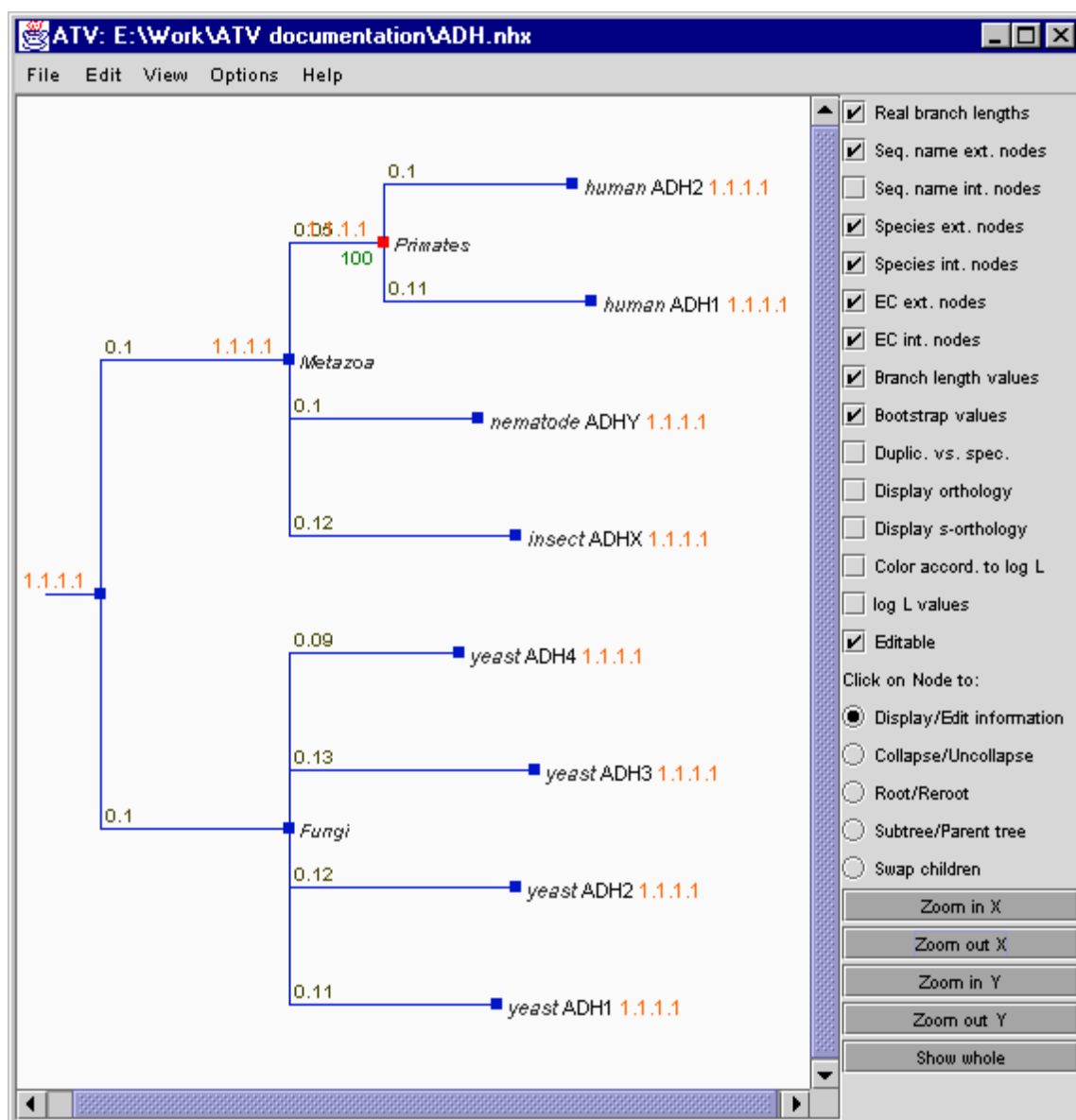


Figure 2.2. A sample tree to illustrate the NHX format.

3 SDI: 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

Published as:

Zmasek, C. M. and Eddy, S. R. (2001) “A simple algorithm to infer gene duplication and speciation events on a gene tree”. *Bioinformatics*, Vol. 17, no. 9, pages 821-828.

On-line version is available at:

[<http://bioinformatics.oupjournals.org/cgi/content/abstract/17/9/821>]

3.1 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; eddy@genetics.wustl.edu;

3.2 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 homologues 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 (1998b).

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 as 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. 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 (Eulenstein, 1998; Page, 1998; Zhang, 1997) 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.

3.3 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 homologues belonging to the same gene family (tree G_1 in Figure 3.1). 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 3.1). 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 3.1).

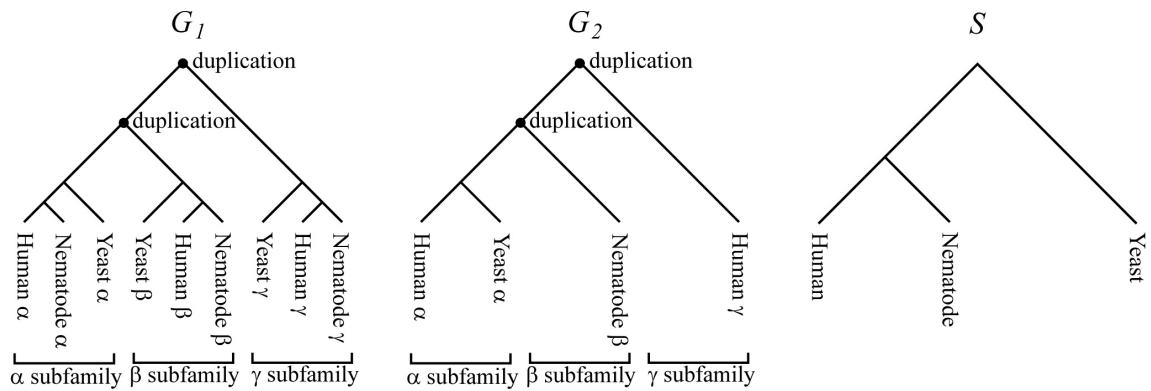


Figure. 3.1. Gene trees and species trees.

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: α , β and γ . 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.

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 3.1. 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 3.2. 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 3.2. This approach makes a parsimony assumption. It postulates the minimal number 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.

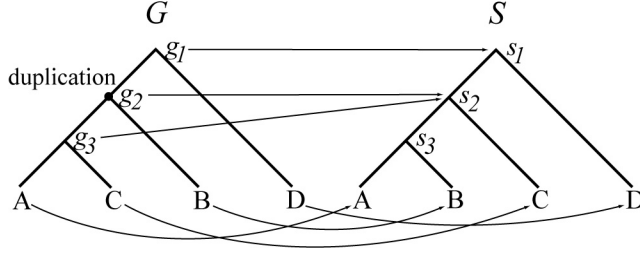


Figure. 3.2. The mapping function M and the definition of a duplication.

M is symbolized by arrows originating at nodes of the gene tree G and pointing to nodes in the species tree S . Letter A to D represent species names. As an example, the mapping for g_3 is computed as follows. According to definition 3.1, $\gamma(g_3) = \{A, C\}$, 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 3.2, since it and its child g_3 maps to the same node s_2 .

Given the mapping function $M(g)$, using definition 3.2 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 refer to 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 3.3 A). 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 3.3 B). 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 a least one external child, also called a “pectinate” tree) with a height of n , giving an overall running time of $O(n^2)$ (Figure 3.3 C). 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.

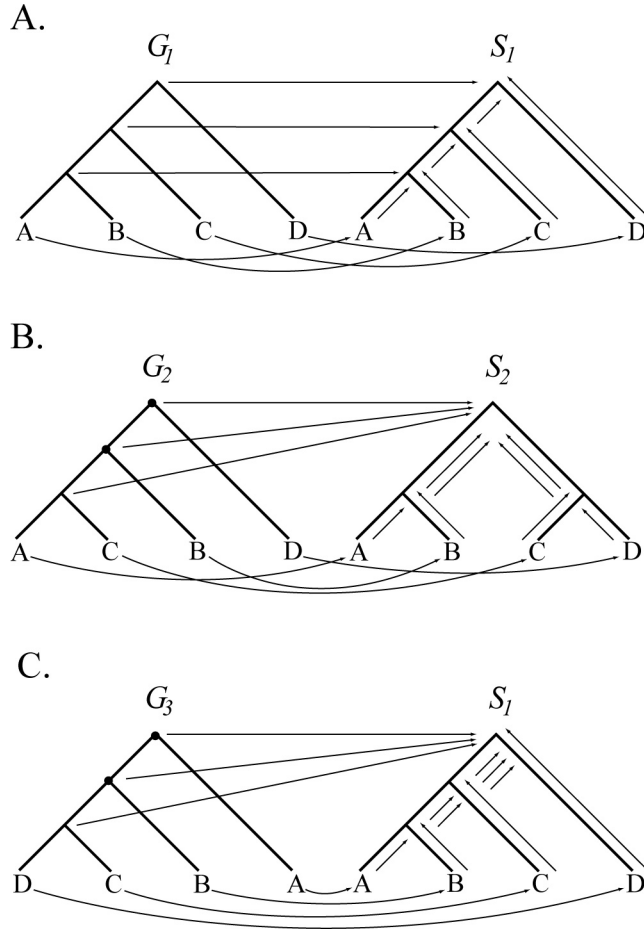


Figure. 3.3. The number of duplications and the topology of the species tree influence the time complexity of our algorithm.

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. Letter 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.

3.4 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.

3.5 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 3.3B and 3.3C). Plots of wall clock time versus n are shown in Figure 3.4. 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’s 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 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 PHYLIP package (Felsenstein, 1993). 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 3.4. 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.

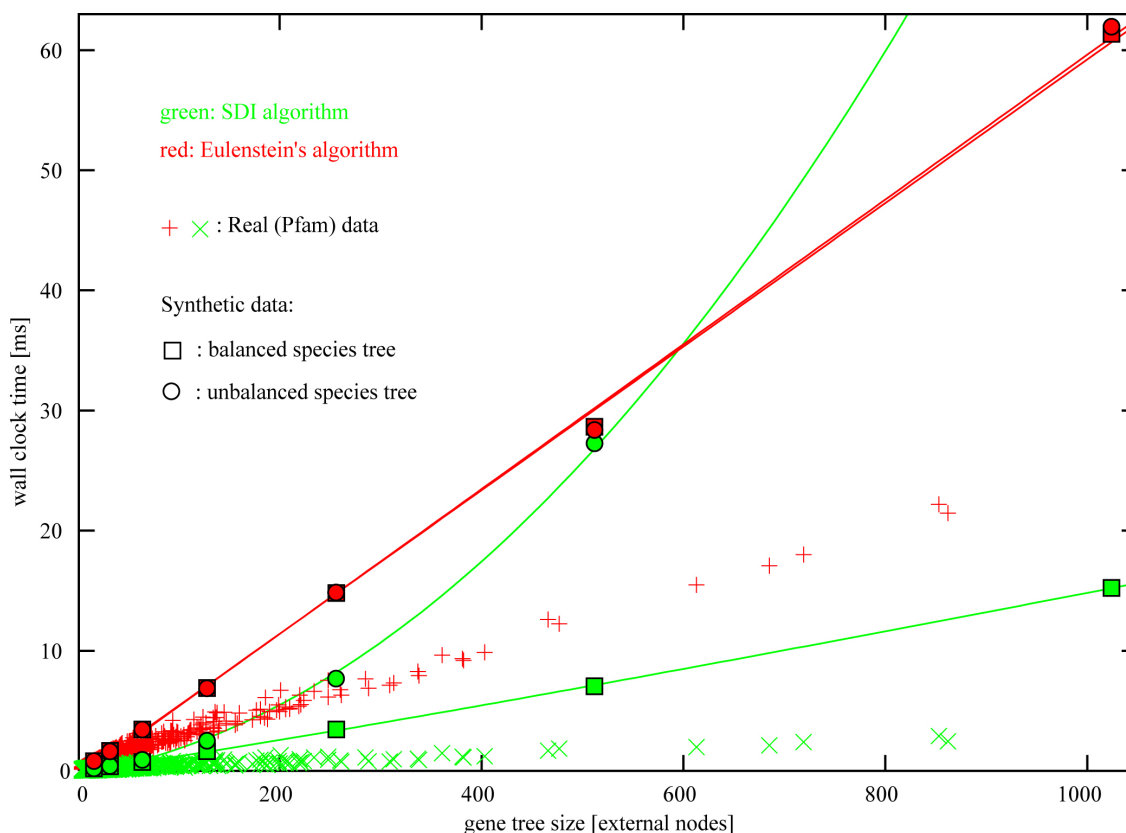


Figure. 3.4. Timing benchmarks on real trees to determine average-case behavior, and synthetic trees that exercise our algorithm's worst case behavior.

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.

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 3.5. 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 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 (Baker et al., 1990; Jones et al., 1988) 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 naïve best BLAST analysis of the FIBX_MOUSE sequence could easily have misannotated it as the mouse fibrinogen beta chain.

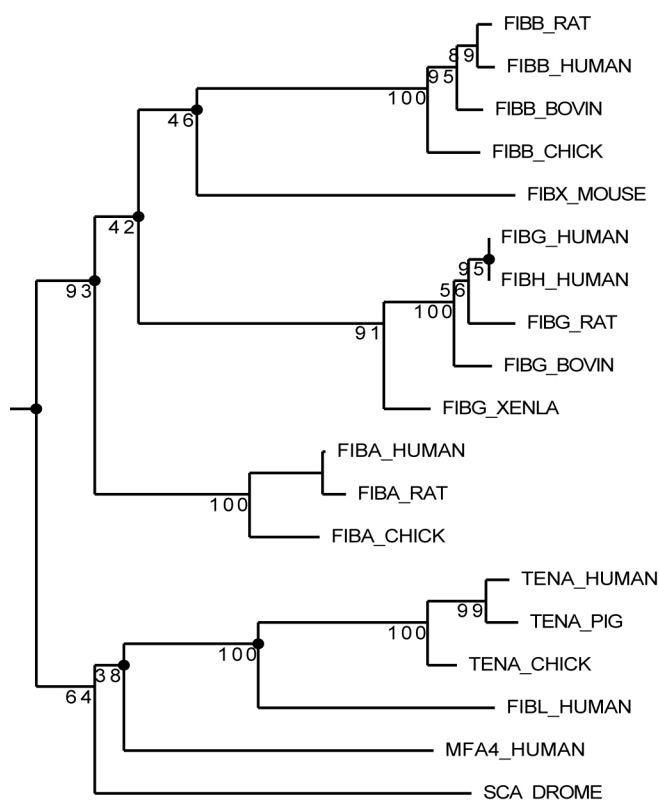


Figure. 3.5. A gene tree for the fibrinogen beta and gamma chain Pfam family.

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, 2001a).

3.6 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 3.4 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, 1998b). 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 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 (Felsenstein, 1985; Mueller and Ayala, 1982) 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.

3.7 Acknowledgements

This work was supported primarily by a grant from Monsanto Company, and also by the Howard Hughes Medical Institute and grant HGo1363 from the NIH National Human Genome Research Institute.

4 RIO: Analyzing proteomes by automated phylogenomics using resampled inference of orthologs

Christian M. Zmasek and Sean R. Eddy

To be submitted for publication in *BMC Bioinformatics*.

4.1 Abstract

Background: When analyzing protein sequences using sequence similarity searches, orthologous sequences (diverged by speciation) are more reliable predictors of a new protein's function than paralogous sequences (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).

Results: Here we present RIO (Resampled Inference of Orthologs), a procedure for automated phylogenomics using explicit phylogenetic inference. A major caveat of all phylogenetic analyses is the unreliability of the resulting trees. Therefore, all RIO analyzes are performed over bootstrap resampled phylogenetic trees to estimate the reliability of the assignments. We also introduce supplementary concepts which might be helpful for functional inference. RIO has been implemented as Perl pipeline of a variety of C and Java computer programs. It is available at [<http://www.genetics.wustl.edu/eddy/forester/>]. A web server allowing to perform RIO analyzes has been set up at [<http://www.rio.wustl.edu/>]. RIO was tested on the *Arabidopsis thaliana* and *Caenorhabditis elegans* proteomes.

Conclusion: The RIO procedure is particularly useful for the automated detection of first representatives of novel protein subfamilies. We also describe how certain types of orthologs might be misleading for functional inference.

4.2 Introduction

Accurate computational protein function analysis is an important means to extract value from the growing amount of primary sequence data. Due to the large amount of data, automated systems seem unavoidable (at least for initial, prioritizing steps). Such efforts are complicated, for a variety of reasons. The focus of this work is problems stemming from the fact that many proteins belong to large families, as suggested by Dayhoff (1976). Such families are oftentimes composed of subfamilies related to each other by gene duplication events. For example, it was shown by Ingram (1961) that human α , β , and γ chains of hemoglobins are related to each other by gene duplications. Gene duplication allows one copy to assume a new biological role through mutation, while the other copy prevents the loss of the original functionality (Haldane, 1932; Ohno, 1970). Hence, subfamilies oftentimes differ in their biological functionality yet still exhibit a high degree of sequence similarity amongst each other (for the human α , β , and γ hemoglobin chains the sequence similarity at the amino acid level is between 41 and 73 percent).

Other complications in functional analysis include: ignoring the multi-domain organization of proteins; error propagation caused by transfer of information from previously erroneously annotated sequences; insufficient masking of low complexity regions; and alternative splicing [for a detailed discussion see (Galperin and Koonin, 1998)].

Typically, automated sequence function analysis is accomplished using methods based on pairwise sequence similarity [such as BLAST (Altschul et al., 1990) or FASTA (Pearson, 1990)]. Annotating a sequence by transferring annotation from its most similar sequence(s) tends to classify too aggressively (overly detailed annotation).

In contrast, analyses using profile search algorithms such as HMMER (Eddy, 2000) together with a protein family database such as Pfam (Bateman et al., 2000), classify sequences too conservatively (under annotation). They recognize that a query sequence belongs to a certain family (or, to be more precise, indicate which domain(s) the query is likely to contain), but do not subclassify the sequence. Such methods are effective at dealing the multi-domain organization of proteins.

Erroneous predictions caused by protein families consisting of subfamilies with different biological roles can often be avoided by taking into account the evolutionary history of sequences, as illustrated in Figures 4.1 and 4.2. Profile search algorithms can be used to align the query 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 gene tree of known sequences can infer a more specific function. This approach was called “phylogenomics” by Eisen (Eisen, 1998b). 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.

In particular, the following two scenarios can cause misleading predictions when using sequence similarity alone for annotation: (i) gene loss and/or incomplete sequence databases to run the similarity search against (incomplete sequence databases: not containing at least one representative for each subfamily) (Figure 4.1), and (ii) unequal rates of evolution (Figure 4.2). When dealing with a first (or only) representative of a novel subfamily we always have a situation where the database is incomplete (since by definition it does not contain other examples of the novel subfamily). Thus, similarity based methods alone cannot tell whether a sequence is a first (or only) representative of a novel subfamily and therefore does not belong into any currently known subfamily (e.g. “orphan” G-protein coupled receptors) since every sequence is most similar to some other sequence. In contrast, when constructing a phylogenetic tree, this fact is easy to observe (as illustrated in Figure 4.1).

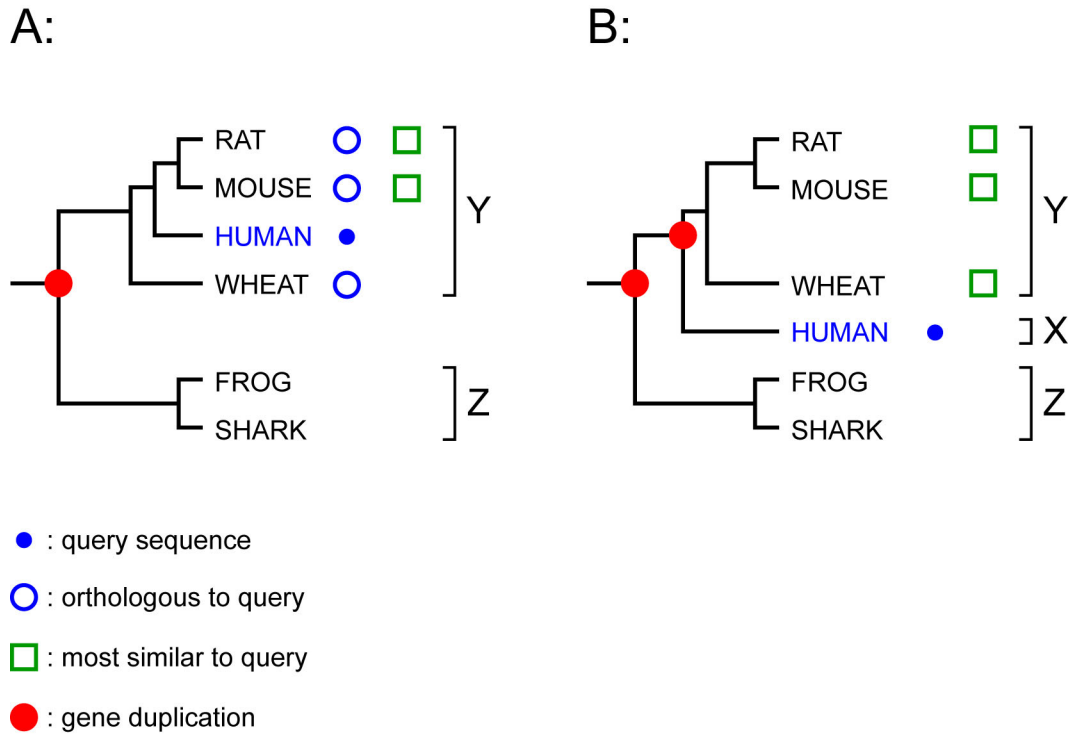
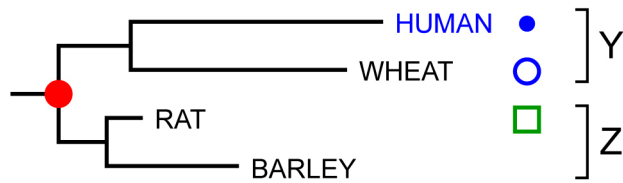


Figure 4.1. Over annotation due to database bias or gene loss under equal rates of evolution.

Species harboring the sequences are indicated. Two cases are depicted. In A, the query sequence belongs to the “Y” subfamily which can be correctly inferred by both sequence similarity and phylogenetic tree based methods (in situation A, the query is most similar to “Y” of rat and mouse). In short, in situation A, orthology and “most similar” do (partially) overlap. In B, a situation is depicted where the query is actually a member of a third subfamily “X” but this can only be inferred by considering the evolutionary history of this sequence family. Sequence similarity based methods would misleadingly indicate that this query belongs to “Y” since it is most similar to “Y” in rat, mouse and wheat. In short, in situation B, orthology and “most similar” do not correspond. Observe that if there would have been already members of “X” in the database (no gene loss and complete sampling) the query in B could have been correctly determined to belong to a “X” subfamily (under equal rates of evolution).



- : query sequence
- : orthologous to query
- : most similar to query
- : gene duplication

Figure 4.2. Over annotation due to unequal rates of evolution.

Sequence similarity based methods would indicate that the query is a member of the “Z” subfamily. Phylogenetic tree based methods correctly identify it as a member of subfamily “Y”.

It is infeasible to completely automate functional analysis, 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 well defined and can be inferred from gene and species trees. One simple example of a phylogenomics approach that is 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. 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. Various efficient algorithms to infer gene duplications on a gene tree by comparing it to a species have been described [for example: by Eulenstein (Eulenstein, 1998), and by Zhang (Zhang, 1997)]. We developed a simple algorithm (named SDI for Speciation Duplication Inference) that appears to solve this problem even more efficiently on realistic data sets, though it has an asymptotic worst-case running time that is less favorable (Zmasek and Eddy, 2001b).

In practice, most gene trees tend to be unreliable. Errors in trees will produce spurious inferred duplications. This is obviously problematic if duplications are to be used as indicators of potential functional changes. Therefore, instead of determining the orthologs of a query sequence on just one gene tree, inference might be performed over bootstrap resampled gene trees (Felsenstein, 1985; Mueller and Ayala, 1982). This gives a bootstrap estimate of the reliability of the assignments. Here we describe and test a procedure – RIO (for Resampled Inference of Orthologs) – which allows to perform such analyses in an automated fashion. [A similar procedure named “orthostrapper” has been proposed by Storm and Sonnhammer (personal communication). In contrast to the RIO approach, “orthostrapper” does not employ a species tree for duplication inference. It works by pairwise comparison of two species or two groups of species. Therefore it is suitable for finding orthologs in a given species or group of species but it cannot be used to detect orthologs from any species.]

The design goals for the RIO system were as follows: (i) Given the input of a query sequence and a sequence alignment, the output should consist of a list of

orthologs, ordered according to a confidence value and useable in a bioinformatics pipeline. (ii) The response time should be fast enough, so that RIO can be used as a web server, and allow the analysis of whole genomes in a reasonable time.

In addition, we present results from analyzing a plant [*A. thaliana* (Arabidopsis-Initiative, 2000)] and a animal [the nematode *C. elegans* (C.elegans-Sequencing-Consortium, 1998)] proteome.

4.3 Algorithm

4.3.1 Definitions

Orthologs are defined as two molecular sequences which diverged by a speciation event (their last common ancestor on a phylogenetic tree corresponds to a speciation event). Paralogs are defined as two sequences which diverged by a duplication event (their last common ancestor corresponds to a duplication) (Fitch, 1970). In addition to orthology, other concepts derived from gene trees can be used as means for functional prediction. In the following we introduce and justify three such concepts (“super-orthologs”, “ultra-paralogs”, and “subtree-neighbors”):

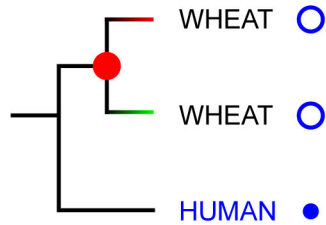
Even though orthologs are theorized to be good sources to transfer functional annotation from, their indiscriminate use for this purpose can leave to incorrect annotations as well. In particular, situations like the one described for the *A. thaliana* O-methyltransferase F16P17_38 later in this work pose potential pitfalls. In the simple example shown in Figure 4.3A, the human query sequence has two orthologous sequences in wheat. These two wheat sequences are related to each other by a gene duplication and one (or, less likely, both) of them might have undergone functional modification after their divergence. Such situations might be revealed by the only partial (or complete absence of) consensus among the annotations of the two orthologs (assuming we are given a list of orthologs as opposed to the gene tree). If *one* ortholog *has* to be chosen to transfer annotation

from, the best guess is to choose the one with the smallest evolutionary distance to the query. The situation in Figure 4.3B is trickier, since in this case only one ortholog is present and the condition is not be exposed by only partial consensus among orthologs. While we do not attempt to solve this problem (a possible solution is careful manual analysis of the gene tree combined with all other possible sources of information) we intend to at least give the user a warning that this situation might be present. For this purpose we introduce the concept of “super-orthologs”:

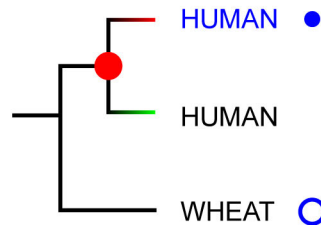
Definition 4.1. Given a completely binary and rooted gene tree with duplication or speciation assigned to each of its internal nodes, two sequences are defined super-orthologous toward each other if and only if each internal node on their connecting path represents a speciation event.

Hence, the query sequences in Figure 4.3 have no super-orthologs. In contrast, the rat, mouse, and wheat sequences in Figure 4.1A are super-orthologous towards the human query sequence. By definition, the super-orthologs of a given sequence are a subset of its orthologs.

A:



B:



● : query sequence

○ : orthologous to query

● : gene duplication

— : potential functional modification after duplication

Figure 4.3. The purpose of super-orthologs.

Examples of how inferring the biological role of a query sequence by simply transferring functional annotation from a orthologous sequence might lead to inaccuracies. These potential pitfalls lead us to introduce the concept of super-orthologs (Definition 4.1).

Certain sequences underwent multiple duplications relatively recently resulting in large and species specific sequence families. Examples for such families are the *C. elegans* seven-transmembrane proteins acting as odorant and chemosensory receptors (Mombaerts, 1999; Troemel, 1999). For query sequences belonging to such sequence families, orthologs (if present) are less effective for predicting specific information. In these cases, paralogs of the same (sub) family might be more informative for functional prediction (as long as the duplications indeed happened “late” in evolutionary times). To formalize this, we introduce the concept of “ultra-paralogs”:

Definition 4.2. Given a completely binary and rooted gene tree with duplication or speciation assigned to each of its internal nodes, two sequences are defined ultra-paralogous towards each other if and only if the smallest subtree containing them both contains only internal nodes representing duplications.

Figure 4.4 illustrates the concept of ultra-paralogs. It follows from definition 4.2 that two sequences which are ultra-paralogous towards each other must occur in the same species and are connected by a path consisting solely of duplication events (being connected by a path of only duplication and being in the same species are necessary conditions for ultra-paralogy, but not sufficient ones).

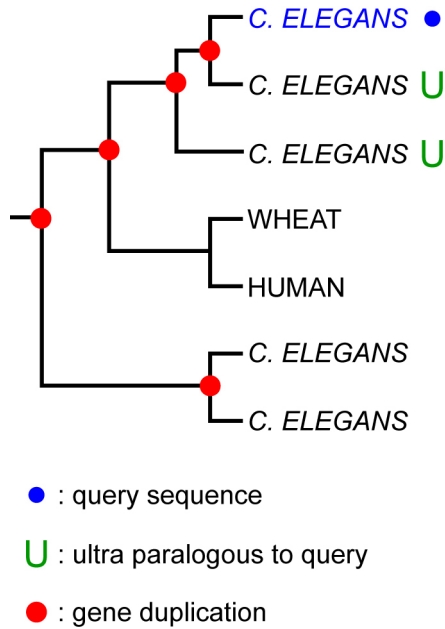


Figure 4.4. An example of ultra-paralogous sequences.

Oftentimes, researchers construct a gene tree and then rather informally use “subtrees” to make inferences about sequences (without regard to duplications and speciations). We introduce this concept into our procedure as well, formalized as “subtree-neighbors” (illustrated in Figure 4.5):

Definition 4.3. Given a completely binary and rooted gene tree, the k -subtree-neighbors of a sequence q are defined as all sequences derived from the k -level parent node of q , except q itself (the level of q itself is 0, q ’s parent is 1, and so forth). The default value of k is 2.

In general, subtree-neighbors are a less strict criterion than orthologs. They can be useful if there is (partial) consensus among them (for example: if the

4.3.2 The RIO procedure

This part portrays the basic RIO procedure. For a simple example with only four bootstrap resamples, see Figure 4.6.

The method described here utilizes the Pfam protein family database (Bateman et al., 2000) as a source of high quality curated sequence alignments and profile HMMs [Hidden Markov Models, see (Eddy, 1996) for a review], as well as programs from the HMMER package (Eddy, 2000). The procedure can easily be adapted to work with different sources of alignments and different alignment programs. For tree reconstruction, the neighbor joining (NJ) algorithm (Saitou and Nei, 1987) is used, since it is reasonably fast and does not assume a molecular clock. It recreates the correct additive tree as long as the input distances are additive (Studier and Keppler, 1988), and is effective even if additivity is only approximated (Atteson, 1997). This is essential, since we try to avoid erroneous annotations caused by the absence of a molecular clock (see Figure 4.2).

Input: A query protein sequence Q with unknown function.
 A curated multiple alignment A from the Pfam database for the protein family to which Q belongs to (as determined by hmmpfam from the HMMER package).
 A profile HMM H for the protein family to which Q belongs to.

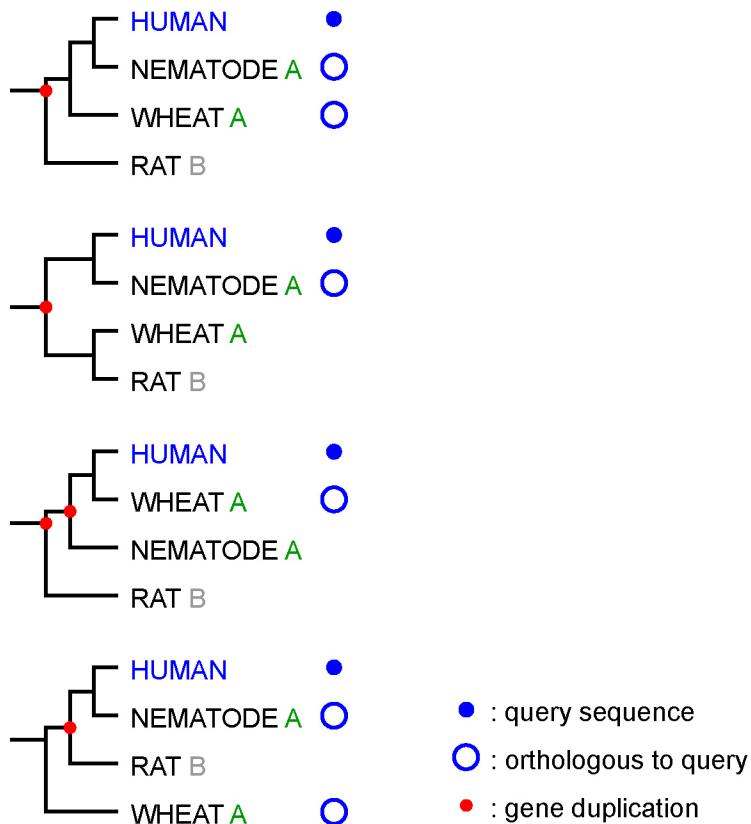
Output: A list (as in Figure 4.7) of proteins orthologous to Q , sorted according to a bootstrap confidence value (based on orthology, super-orthology, or subtree-neighborings).

Optional: A gene tree based on the multiple alignment A and the query Q annotated with orthology bootstrap confidence values for the query Q .

Procedure:

1. Query sequence Q is aligned to the existing alignment A (using `hmmalign` from the HMMER package and the Pfam profile HMM H).
2. The alignment is bootstrap resampled x times (usually, $x = 100$).
3. Pairwise distances are calculated for each of the x alignments using a model of amino acid substitution [for example, BLOSUM (Henikoff and Henikoff, 1992) or Dayhoff PAM (Dayhoff et al., 1978)].
4. A phylogenetic tree is inferred for each of the x sequence alignments [by Neighbor joining (Saitou and Nei, 1987)]. This results in x gene trees.
5. For each of the x gene trees: For each node it is inferred whether it represents a duplication or a speciation event by comparing the gene tree to a trusted species tree. Note: Neighbor joining produces unrooted trees, yet speciations and duplications are only meaningful on a rooted tree. Therefore, a modified version of our SDI algorithm (Zmasek and Eddy, 2001b) is employed. This algorithm infers gene duplications and at the same time roots the tree by minimizing the sum of duplications. For a more detailed description, see below.

6. For each sequence s in the gene tree (except Q): Count the number of gene trees where s is orthologous to Q (see Figure 4.6 for an illustration of steps 5. and 6.).
7. Additionally, unusually long or short branch lengths on the gene tree (either a consensus tree with maximum likelihood branch lengths or a tree based on the original alignment including Q) are used as an indicator of highly unequal rates of evolution which might warrant special consideration. This gene tree is also a optional part of the output.



Result:

Orthologous to HUMAN query in n/4 times:

NEMATODE A: $3/4 = 75\%$

WHEAT A: $3/4 = 75\%$

RAT B: $0/4 = 0\%$

Figure 4.6. A simple example of the RIO procedure.

Four bootstrap resampled gene trees are shown. Letters represent sequence names/"functions". "A" (nematode and wheat) are true orthologs of the human query sequence, whereas "B" (rat) is a true paralog of the query (i.e. the first tree happens to be the real one). In 3 out of 4 trees nematode "A" appears orthologous to the query, in 3 out of 4 trees wheat "A" appears orthologous to the query. Rat "B" never appears to be orthologous. For an example of actual RIO output see Figure 4.7.

Even though the RIO algorithm as described above only calculates values based on orthologies, values for super-orthologies, ultra-paralogies and subtree-neighbors can be calculated in exactly the same manner (it only requires to replace “orthologous” in step 6. with “super-orthologous”, “super-paralogous”, or “subtree-neighboring”).

4.3.3 Precalculation of pairwise distances for increased time efficiency

The most time consuming step in the procedure described above is the calculation of the pairwise distances. [The time complexity is $O(N^2)$, N being the number of sequences. On an average Intel processor the wall clock time for 100 bootstrapped datasets is in the range of hours for N in the range of hundreds.]

Since the query sequence is aligned to stable Pfam alignments it is possible to precalculate the pairwise distances for each alignment and store the results. Then, when RIO is being used to analyze a query sequence, only the distances of the query to each sequence in the Pfam alignment have to be calculated. This step becomes thus $O(N)$ instead of $O(N^2)$ (and what was hours before is reduced to minutes).

The crucial part is that the query sequence has to be bootstrap resampled in exactly the same way as has been used for precalculating the pairwise distances of the Pfam alignment. For this purpose, the bootstrap positions are saved to a file while precalculating the pairwise distances. With this file it is possible to

bootstrap both the Pfam alignment and the query sequence in precisely the same manner.

A technical note: The HMMER program `hmmalign` (used in the RIO procedure) does not necessarily keep the non-match columns of a input alignment unchanged. Yet, RIO utilizing precalculated distances is critically dependent on completely fixed alignments. Therefore, the precalculation of pairwise distances also includes the creation of specific profile HMMs which together with the appropriate steps in the RIO procedure itself (“`--mapali`” option for `hmmalign`, removal of non-match columns after the alignment of the query sequence) result in completely fixed alignments. For a description of this precalculations in the form of an algorithm see Appendix A.

4.3.4 Rooting of gene trees

The concept of speciation and duplication is only meaningful on rooted gene trees. Yet neighbor joining produces unrooted trees. For the purpose of this work we decided based both on empirical grounds as well as on theoretical ones, that the following parsimony criterion for rooting is probably adequate: Gene trees are rooted on each branch, resulting in $2N-3$ differently rooted trees for a gene tree of N sequences. For each of these trees the sum of duplications is determined. From the trees with a minimal number of duplications (if there is more than one) the tree with the shortest total height is chosen as the “correctly rooted one”. Empirical studies on gene trees based on 1750 Pfam alignments show that about 60% of trees rooted in such a way have their root in the same

position as direct midpoint rooting (Swofford et al., 1996) would place it (results not shown).

Even though some algorithms used for duplication inference run in (approximately) linear time (Eulenstein, 1998; Zhang, 1997; Zmasek and Eddy, 2001b), naively performing a full duplication/speciation analysis on each of $2N-3$ differently rooted trees results in a overall time complexity of approximately $O(N^2)$. Fortunately, this can be avoided.

For the purpose of the following discussion it is assumed that SDI, our algorithm for speciation duplication inference, is employed. But it applies to all algorithms which calculate a mapping function M . M has been defined as follows (Goodman et al., 1979):

Definition 4.4. 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 $\gamma(g) \subseteq \sigma(M(g))$.

Duplications are then defined using $M(g)$ as follows:

Definition 4.5. 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 $M(g) = M(g_2)$.

The main task of most algorithms for duplication inference is the calculation of M . After M has been calculated for a randomly rooted gene tree G it is possible to explore different root placements without having to recalculate M for each node of G . As long as the root is moved one node at the time, M has to be recalculated only for two nodes: the one node which was child 1 (if the new root is placed on a branch originating from child 1 of the previous root) or child 2 (otherwise) of the previous root, as well as for the new root itself. Hence, two postorder traversal steps (child 1 or 2 of the old root, then the new root) in the SDI algorithm are all that is needed. The new sum of duplications is simply determined by keeping track of the change in duplication/speciation status in the two recalculated nodes as well as in the previous root.

Performing this over the whole gene tree (some nodes will be visited twice) it is possible to explore all possible root placements and calculate the resulting duplications in practically linear time. See Appendix B for a description of this in the form of an algorithm.

4.3.5 Master species tree

Duplication inference on a gene tree requires a species tree to compare the gene tree to. For this purpose, a single completely binary master species tree was

compiled manually, containing 249 of the most commonly encountered species in Pfam (spanning Archaea, Bacteria, and Eukaryotes). This tree is based mainly on information from Maddison's "Tree of Life" project [<http://phylogeny.arizona.edu/tree/phylogeny.html>], NCBI's taxonomy database [<http://www.ncbi.nlm.nih.gov/Taxonomy/taxonomyhome.html/>], the "Deep Green" project [<http://ucjeps.berkeley.edu/bryolab/greenplantpage.html>], and (Aguinaldo et al., 1997; Barns et al., 1996; de Rosa et al., 1999; Morris, 1998). This master tree groups nematodes and arthropods into a clade of ecdysozoans (molting animals) as first proposed by Aguinaldo (Aguinaldo et al., 1997), a classification which is still controversial. The tree is available in NHX format (Zmasek and Eddy, 2001a) at [http://www.genetics.wustl.edu/eddy/forester/tree_of_life_bin_1-4.nhx]

4.4 Implementation

RIO is implemented in the form of the following perl pipeline: Alignment of the query sequence is accomplished by programs from the HMMER package (Eddy, 2000). Bootstrapping is performed by a specifically designed C program. Pairwise distances are calculated by a modified version of TREE-PUZZLE (Strimmer and von Haeseler, 1996). Neighbor joining trees are calculated by a modified version of NEIGHBOR from the PHYLIP package (Felsenstein, 2001). Rooting and duplication inference are accomplished by “SDIunrooted” – a Java implementation of our SDI algorithm which incorporates various methods for rooting (see Appendix A). The actual counting of orthologs is performed by methods of the Java class “RIO”.

These programs, with the exception of HMMER, are part of the FORESTER package and are available under the GNU GPL license at [<http://www.genetics.wustl.edu/eddy/forester/>].

In order to run RIO locally, the following packages and databases need to be present: HMMER (Eddy, 2000), the Pfam database (Bateman et al., 2000), the SWISS-PROT and TrEMBL databases (Bairoch and Apweiler, 2000).

RIO is also available as a webserver at [<http://www.rio.wustl.edu/>]. For increased time efficiency, the pairwise distance and tree calculations are parallelized in this version.

4.5 Results and Discussion

4.5.1 Precalculation of pairwise distances

Pairwise distances to be used in RIO analyses were calculated using the “full” alignments (as opposed to the smaller “seed” alignments) from Pfam 6.6 (August 2001, 3071 families, (Bateman et al., 2000)). The maximum likelihood distances were calculated based on the BLOSUM (Henikoff and Henikoff, 1992) distance matrices using the TREE-PUZZLE (Strimmer and von Haeseler, 1996) software (non-match columns of the alignments were discarded prior to distance calculation, as described above). For each family, pairwise distances for 100 bootstrap samples were prepared. Pfam alignments which were either too short or did not include enough sequences were ignored since analyses based on such alignments would probably be meaningless. The detailed rules and justifications for this selection are as follows: Alignments of an average length of less than 30 amino acids were ignored, since they are unlikely to contain enough phylogenetic signal. For zinc-finger domains this minimal average length was set to 40 amino acids (as empirical results have shown, the signal in these is particularly poor). Sequences from species not present in the master species tree (see above) were removed from the alignments (which results in the rejection of all families containing solely viral sequences, since our master species tree does not include viruses). Resulting alignments containing less than six sequences were ignored. The reason for this is: The addition of sequences to a gene tree can turn

sequences which were orthologous to each other into paralogs. (For example, imagine an internal node leading to a mouse and a human sequence. Adding a yeast sequence to the branch connecting this internal node with the human sequence, changes to mouse and human sequences into paralogs.) Thus, the smaller an alignment is, the more likely it will result in an incomplete tree in which sequences appear orthologous to each other simply due to the absence of certain sequences. The threshold of six was chosen arbitrarily.

Alignments containing more than 600 sequences (after removal of sequences from species not present in the master species tree) were dealt with in following manner: Sequences not originating from SWISS-PROT were discarded. In addition, sequences from certain mammals were excluded (all primates except human, mouse, rabbit, hamsters, and goat), since mammals are likely to be over represented in most Pfam families (primates and rodents in particular). For extremely large families [immunoglobulin domain (PF00047), protein kinase domain (PF00069), collagen triple helix repeat (PF01391), and rhodopsin-type 7 transmembrane receptor (PF00001)], all mammalian sequences except those from human and rat were excluded.

Following the above rules, pairwise distances (and other the files described in Appendix B) were precalculated for 2384 alignments from a total of 3071 in Pfam 6.6 (75 alignments were too short and 612 alignments did contain less than six sequences from species in our master species tree).

4.5.2 Phylogenomic analyses of the *A. thaliana* and *C. elegans* and proteomes

We used the RIO procedure to analyze the *A. thaliana* (Arabidopsis-Initiative, 2000) and *C. elegans* (C.elegans-Sequencing-Consortium, 1998) proteomes in order to get an estimate of the effectiveness of this implementation of automated phylogenomics.

4.5.2.1 Domain structure analysis

The input for RIO consists of a query protein sequence together with a Pfam alignment for the/a protein family to which the query belongs to. Before RIO could be applied we therefore had to determine the matching domains for each protein in the *A. thaliana* and *C. elegans* proteomes. For proteins composed of different domains, a RIO analysis has to be performed for each domain individually.

The source for protein sequences were: ATH1.pep.03202001, a flatfile database of 25,579 *A. thaliana* amino acid sequences (hypothetical, predicted and experimentally verified) that have been identified as part of the Arabidopsis Genome Initiative (AGI) [<http://www.arabidopsis.org/info/agi.html>], and wormpep 43, a flatfile database of 19,730 *C. elegans* amino acid sequences [http://www.sanger.ac.uk/Projects/C_elegans/wormpep/].

The program hmmpfam (version 2.2g) from the HMMER package was used to search each protein sequence in ATH1.pep.03202001 and wormpep 43

against Pfam 6.6. Only domains with a score above the so-called gathering cutoff were reported (“cut_ga” option) in order to prevent too many erroneous domain assignments (which would make subsequent RIO analyses harder to interpret).

The sum of domains assigned to the 25,579 *A. thaliana* protein sequences was 17,847 (counting multiple copies of the same domain in one protein as one). 12,431 sequences matched one domain (containing possibly multiple copies of this one domain). 1,982 sequences matched two different domains (containing possibly multiple copies of both). 453 sequences matched three or more different domains (containing possibly multiple copies of each). Therefore, a total of 14,866 (58%) sequences from ATH1.pep.03202001 could be assigned to one or more Pfam families.

Similarly, a sum of 12,314 domains was assigned to the 19,769 *C. elegans* protein sequences. 7,698 sequences matched one domain, 1,632 matched two different domains, and 388 matched three or more different domains. Thus, 9,718 (49%) sequences from wormpep 43 could be assigned to one or more Pfam families.

4.5.2.2 RIO analysis

After it has been determined which domains the proteins in the *A. thaliana* and *C. elegans* proteomes were likely to contain, RIO was used to analyze each protein sequence matching one or more Pfam families. Since the precalculated distances described above were used, all the results are based on maximum likelihood distances calculated on the BLOSUM matrices and the

number of bootstrap resamples is 100. The results from these analyses can be found at [http://www.genetics.wustl.edu/eddy/forester/rio_analyses/]. The approximate time requirement was between two and three weeks, performed on eight Pentium III 800Mhz processors.

4.5.2.2.1 How many sequences can be analyzed with RIO?

The first question we tried to answer was how many sequences can be analyzed with RIO. For an overview, see Table 4.1. From the 17,847 *A. thaliana* domain sequences matching a Pfam family, 14,905 (84%) could be analyzed with RIO using the precalculated distances. 2859 (16%) domain sequences were not analyzed because the corresponding Pfam alignments were either too short or did not contain enough sequences (as described above). 83 (0.5%) domain sequences were not analyzed because the E-value for the match to their profile HMM was below the threshold of 0.01. This represents a second filtering step for preventing analyzing false domain assignments (besides only analyzing domain sequences which score above the gathering cutoff in the domain analysis). (RIO performs a preprocessing step before aligning the query sequence to a Pfam alignment, in which the program `hmmsearch` is used to trim the query sequence by searching it with the appropriate profile HMM. If the resulting E-value was below 0.01 no analysis was performed.) Multiple copies of the same domain in certain sequences result in a sum of individual analyses larger than the number of analyzed domain sequences. In case of *A. thaliana* this number was 17,940.

Correspondingly, from the 12,314 *C. elegans* domain sequences matching a Pfam family, 11,287 (92%) could be analyzed with RIO using the precalculated distances. 901 (7%) domain sequences were not analyzed because the corresponding Pfam alignments were either too short or did not contain enough sequences. 53 (0.4%) domain sequences were not analyzed because the E-value for the match to their profile HMM was below the threshold of 0.01. In addition, we did not analyze the 73 *C. elegans* sequences matching the immunoglobulin family (PF00047). It turned out that the phylogenetic signal in this alignment is questionable. Furthermore, most of the 73 sequences contain multiple copies of the immunoglobulin domain (for example, CE08028 contains 48 immunoglobulin domains) and we therefore worried that the results from this family might skew our overall results. The sum of RIO analyses was 14,740.

In summary, while RIO itself (using precalculated distances prepared as described above) could analyze most of its query sequences, a high number of proteins did not match any Pfam family and were therefore precluded from being analyzed with RIO.

	Protein sequences in proteome	Sum of domains assigned to proteome	Domain sequences analyzed with RIO	Sum of individual RIO analyses
<i>A. thaliana</i>	25,579	17,847	14,905	17,940
<i>C. elegans</i>	19,769	12,314	11,287	14,740

Table 4.1. Number of domains which can be analyzed with RIO.

4.5.2.2.2 RIO analysis of lactate/malate dehydrogenase family members

Second, in order to test that RIO performs well on an “easy” case, RIO was used to analyze lactate/malate dehydrogenase family members both in *A. thaliana* and *C. elegans*. L-Lactate and malate dehydrogenases are members of the same protein family (represented in Pfam as ldh for the NAD-binding domain and ldh_C for the alpha/beta C-terminal domain), yet they catalyze different reactions. L-lactate dehydrogenase (EC 1.1.1.27) catalyzes the following reaction: (S)-lactate + NAD⁺ = pyruvate + NADH (Dennis and Kaplan, 1960). Malate dehydrogenase (NAD) (EC 1.1.1.37) catalyzes: (S)-malate + NAD⁺ = oxaloacetate + NADH (Banaszak and Bradshaw, 1975). NADP-dependent malate dehydrogenase (EC 1.1.1.82) utilizes NADP⁺ as cofactor instead of NAD⁺ (Johnson, 1971; Webb, 1992). According to the Pfam domain analysis described above, the *A. thaliana* proteome contains ten lactate/malate dehydrogenase family members, whereas the *C. elegans* proteome contains three. (In addition, *C. elegans* also contains two putative members of a second lactate/malate dehydrogenase family (Jendrossek et al., 1993), ldh_2, which are not discussed here.) The RIO output for the *A. thaliana* protein F12M16_14 analyzed against the ldh domain alignment is shown as an example in Figure 4.7. The results are summarized in Tables 4.2 and 4.3. Complete RIO output files (as well as NHX (Zmasek and Eddy, 2001a) tree files) are available at [http://www.genetics.wustl.edu/eddy/forester/rio_analyses/RIO_paper/AT_LD_H_MDH/] for *A. thaliana* and at [http://www.genetics.wustl.edu/eddy/forester/rio_analyses/RIO_paper/CE_LD

H_MDH/]. In all cases, distinction between malate dehydrogenase (NAD) and lactate dehydrogenase is unquestionable and in accordance with existing annotations and BLAST results (data not shown) irrespective which domain (ldh or ldh_C) was used for the RIO analysis (which implies that no domain swapping occurred over long evolutionary times). Furthermore, the same results are achieved whether only the top 1 sequence (the one with the highest orthology value, shown in Tables 4.2 and 4.3) or the top 10 sequences are used to transfer annotation from. The only likely NADP-dependent malate dehydrogenase is the *A. thaliana* sequence MCK7_20. For some query sequences, the top orthology values are low. Yet, all subtree-neighborings above 50% exhibit consensus at distinguishing between malate and lactate dehydrogenase. In contrast, a finer distinction (e.g. between mitochondrial and cytoplasmic malate dehydrogenase) proves more problematic. While there is no case of actual conflict between the existing annotation and the RIO results, in many cases there is no compelling evidence in the RIO results to confirm the finer distinctions in the existing annotations. Obviously, the resolution power of RIO is limited by the given annotations and by the number (or even presence) of sequences for each sub(sub)family.

Sequence	Description	o[%]	n[%]	s[%]	distance
MDHM_BRANA/27-173	MALATE DEHYDROGENASE, MITOCHONDRIAL PRECURSOR (EC 1.1.1.37).	89	100	89	0.028000
Q9SPB8_SOYBN/31-177	MALATE DEHYDROGENASE.	87	100	42	0.109080
MDH_ECOLI/1-145	MALATE DEHYDROGENASE (EC 1.1.1.37).	53	0	0	0.458890
MDH_SALTY/1-145	MALATE DEHYDROGENASE (EC 1.1.1.37).	53	0	0	0.468930
...					
MDHM_CHLRE/60-205	MALATE DEHYDROGENASE, MITOCHONDRIAL PRECURSOR (EC 1.1.1.37).	32	2	4	0.358410
MDHM_RAT/22-168	MALATE DEHYDROGENASE, MITOCHONDRIAL PRECURSOR (EC 1.1.1.37).	18	2	0	0.470390
MDHM_PIG/22-168	MALATE DEHYDROGENASE, MITOCHONDRIAL PRECURSOR (EC 1.1.1.37).	18	2	0	0.471480
MDHM_HUMAN/22-168	MALATE DEHYDROGENASE, MITOCHONDRIAL PRECURSOR (EC 1.1.1.37).	18	2	0	0.491850
MDHM_MOUSE/22-168	MALATE DEHYDROGENASE, MITOCHONDRIAL PRECURSOR (EC 1.1.1.37).	18	2	0	0.491910
O15769_TRYBB/6-151	MALATE DEHYDROGENASE.	14	3	0	0.492340
Q9VU29_DROME/25-171	MALATE DEHYDROGENASE.	6	3	0	0.718600
Q9Y7R8_SCHPO/26-173	MALATE DEHYDROGENASE, MITOCHONDRIAL PRECURSOR.	4	2	0	0.557380
Q9VEB1_DROME/22-168	CG7998 PROTEIN.	3	0	0	0.455680
O76731_TRYBB/1-154	GLYCOSOMAL MALATE DEHYDROGENASE.	2	1	0	0.726530
Q9U140_LEIMA/1-153	MALATE DEHYDROGENASE.	2	1	0	0.832380
MDHC_YEAST/10-176	MALATE DEHYDROGENASE, CYTOPLASMIC (EC 1.1.1.37).	2	0	0	0.845440
MDHM_YEAST/15-163	MALATE DEHYDROGENASE, MITOCHONDRIAL PRECURSOR (EC 1.1.1.37).	1	1	0	0.605030
MDHF_YEAST/1-143	MALATE DEHYDROGENASE, PEROXISOMAL (EC 1.1.1.37).	1	0	0	0.580820
MDHG_ORYSA/42-188	MALATE DEHYDROGENASE, GLYOXYSOMAL PRECURSOR (EC 1.1.1.37).	0	12	0	0.338480
MDHG_SOYBN/39-185	MALATE DEHYDROGENASE, GLYOXYSOMAL PRECURSOR (EC 1.1.1.37).	0	12	0	0.350720
MDHG_CUCCA/42-188	MALATE DEHYDROGENASE, GLYOXYSOMAL PRECURSOR (EC 1.1.1.37).	0	12	0	0.368460
MDHG_BRANA/39-185	MALATE DEHYDROGENASE, GLYOXYSOMAL PRECURSOR (EC 1.1.1.37).	0	12	0	0.424130
O81609_PEA/77-223	NODULE-ENHANCED MALATE DEHYDROGENASE.	0	1	0	0.399520
O81844_ARATH/80-226	MALATE DEHYDROGENASE PRECURSOR.	0	1	0	0.428890
Q9SN86_ARATH/80-226	MALATE DEHYDROGENASE.	0	1	0	0.428890
Q9XQP4_TOBAC/91-237	MALATE DEHYDROGENASE PRECURSOR.	0	1	0	0.442160
O81278_SOYBN/92-238	MALATE DEHYDROGENASE.	0	1	0	0.446470
Q9U8L4_LEIMA/1-71	MALATE DEHYDROGENASE (FRAGMENT).	0	1	0	0.468950
P93106_CHLRE/34-180	NAD-DEPENDENT MALATE DEHYDROGENASE (EC 1.1.1.37) (MALIC DEHYDROGENASE).	0	0	0	0.462200
MDHM_CAELI/26-172	PROBABLE MALATE DEHYDROGENASE, MITOCHONDRIAL PRECURSOR (EC 1.1.1.37).	0	0	0	0.483690
Q9VU28_DROME/20-166	MALATE DEHYDROGENASE.	0	0	0	0.907050
O59312_PRRHO/1-23	HYPOTHEICAL 40.1 KDA PROTEIN PH1688.	0	0	0	1.000670
MDH_SULAC/1-37	MALATE DEHYDROGENASE (EC 1.1.1.37) (FRAGMENT).	0	0	0	1.270070
MDH_RICPR/2-145	MALATE DEHYDROGENASE (EC 1.1.1.37).	0	0	0	1.369000
Q29385_PIG/18-42	LACTATE DEHYDROGENASE-A (FRAGMENT).	0	0	0	1.384020
Q55383_SYNY3/11-154	2-KETOACID DEHYDROGENASE (MALATE DEHYDROGENASE, LACTATE DEHYDROGENASE).	0	0	0	1.468610
MDH_BACSU/2-147	MALATE DEHYDROGENASE (EC 1.1.1.37) (VEGETATIVE PROTEIN 69) (VEG69).	0	0	0	1.482390
MDH_CHLVI/1-142	MALATE DEHYDROGENASE (EC 1.1.1.37).	0	0	0	1.509210
MDH_ARCFU/1-142	MALATE DEHYDROGENASE (EC 1.1.1.37).	0	0	0	1.523550
MDH_AERPE/7-145	MALATE DEHYDROGENASE (EC 1.1.1.37).	0	0	0	1.531830
LDH_THEMA/1-140	L-LACTATE DEHYDROGENASE (EC 1.1.1.27).	0	0	0	1.545580
LDH_THEAQ/1-140	L-LACTATE DEHYDROGENASE (EC 1.1.1.27).	0	0	0	1.603000
O67581_AQUAE/11-161	MALATE DEHYDROGENASE.	0	0	0	1.617760
LDHA_HORVU/41-183	L-LACTATE DEHYDROGENASE A (EC 1.1.1.27) (LDH-A).	0	0	0	1.618550
LDHH_RABIT/2-45	L-LACTATE DEHYDROGENASE H CHAIN (EC 1.1.1.27) (LDH-B) (FRAGMENT).	0	0	0	1.618900
...					

Figure 4.7. RIO output for the *A. thaliana* protein F12M16_14 analyzed against the Pfam ldh domain alignment (PF00056).

The “Sequence” column identifies sequences in the Pfam alignment either by their SWISS-PROT “ID” or their TrEMBL “AC” (Bairoch and Apweiler, 2000) with added species information (the numbers after the dash are the Pfam domain boundaries added by HMMER). “Description” is the “DE” information either from SWISS-PROT or TrEMBL. The number of observed orthologies (“o”), subtree-neighborings (“n”), and super-orthologies (“s”) to the query in the 100 bootstrapped trees are indicated (in %) for the sequences in the Pfam alignment. Furthermore the evolutionary distances (average number of amino acid replacements per residue calculated by maximum likelihood based on the BLOSUM 62 matrix) between the query and the sequences in the Pfam alignment are shown. For space reasons some lines of the output are not shown (“...”) (the complete output is available at [\[http://www.genetics.wustl.edu/eddy/forester/rio_analyses/RIO_paper/AT_LDH_MDH/J\]](http://www.genetics.wustl.edu/eddy/forester/rio_analyses/RIO_paper/AT_LDH_MDH/J)).

The output is sorted by orthology values. According to this RIO analysis the query sequence is likely to be orthologous and a subtree-neighbor to the plant sequences MDHM_BRANA and Q9SPB8_SOYBN. In addition, the query is likely to be super-orthologous to MDHM_BRANA. The bacterial sequences MDH_ECOLI and MDH_SALTY are also possibly orthologs but no subtree-neighbors. Hence, F12M16_14 is very likely to be a malate dehydrogenase and possibly mitochondrial.

Sequence ID	Annotation	RIO top 1 hit (highest orthology value)	
		Domain used for analysis:	
		ldh (PF00056)	Ldh_C (PF02866)
dl4665w	LDH (LDH1)	L-LDH (o=91%, n=3%)	L-LDH (o=94%, n=12%)
F19P19_13	putative MDH	MDH (o=2%, n=98%)	cytoplasmic MDH (o=40%, n=78%)
F12M16_14	mitochondrial NAD-dependent MDH	mitochondrial MDH (o=89%, n=100%)	mitochondrial MDH (o=94%, n=66%)
T30L20.4	putative glyoxysomal MDH precursor	MDH (o=55%, n=0%)	glyoxysomal MDH (o=95%, n=37%)
K15M2_16	mitochondrial NAD-dependent MDH, putative	MDH (o=89%, n=100%)	mitochondrial MDH (o=84%, n=80%)
F1P2_70	Chloroplast NAD-dependent MDH	MDH (o=87%, n=21%)	MDH (o=85%, n=6%)
F17I14_150	microbody NAD-dependent MDH	glyoxysomal MDH (o=100%, n=100%)	glyoxysomal MDH (o=80%, n=97%)
MWF20_2	cytoplasmic MDH	MDH (o=2%, n=100%)	MDH (o=38%, n=75%)
MIK19_17	cytoplasmic MDH	cytoplasmic MDH (o=5%, n=99%)	MDH (o=31%, n=84%)
MCK7_20	NADP-dependent MDH	MDH (o=60%, n=30%)	chloroplast NADP-MDH (EC 1.1.1.82) (o=68%, n=82%)

Table 4.2. RIO analysis of *A. thaliana* lactate/malate dehydrogenase family members.

Annotations are from ATH1.pep.03202001 (Arabidopsis Genome Initiative [<http://www.arabidopsis.org/info/agi.html>]). “o=” and “n=” are orthology and subtree-neighboring values for the sequence in the Pfam alignment (ldh or ldh_C) with the highest orthology value towards the respective query sequence. LDH stands for L-lactate dehydrogenase. MDH stands for malate dehydrogenase.

Sequence ID	Annotation	RIO top 1 hit (highest orthology value)	
		Domain used for analysis:	
		Idh (PF00056)	Idh_C (PF02866)
F13D12.2 (CE02181)	LDH (predicted)	L-LDH (o=75%, n=61%)	L-LDH (B chain) (o=66%, n=23%)
F20H11.3 (CE09512)	Member of the MDH protein family (predicted)	MDH (o=42%, n=16%)	MDH (o=53%, n=34%)
F46E10.10 (CE20820)	Putative MDH, possible ortholog of H. sapiens Hs.75375 gene product (cytoplasmic MDH) (predicted)	cytoplasmic MDH (o=13%, n=95%)	MDH (o=76%, n=52%)

Table 4.3. RIO analysis of *C. elegans* lactate/malate dehydrogenase family members.

Annotations are from WormPDTM (Costanzo et al., 2001) (12/31/2001) [<http://www.proteome.com/databases/index.html>]. For more explanations see Table 4.2.

4.5.2.2.3 Sequences with no orthologs in the current databases

Third, we determined the distribution of the top orthology bootstrap values. The sequence with the top orthology bootstrap value is the one which is most likely to be the true ortholog of the query. If the top orthology bootstrap value is low, then the query sequence is likely to have no ortholog in the Pfam alignment. These results are summarized in Table 4.4. For example, for 2252 *A. thaliana* query sequences at least one sequence was orthologous in at least 95 out of 100 resampled trees. In contrast, for 930 *A. thaliana* query sequences, no sequence was orthologous in more than five out of 100 bootstrapped trees. For query sequences with more than one copy of the same domain, each copy had to meet the conditions individually in order for the whole query sequence being counted to be below or above the threshold.

It is beyond the scope of this work to attempt to determine threshold values for “true orthologs” or “absence of orthologs”. Such thresholds are very likely to be different for different Pfam families since families vary in the phylogenetic signal their alignment contains. The only conclusion we would like to make here is that some sequences which are very likely to be true orthologs to the query, exhibit somewhat low orthology bootstrap values (in the range of 70% or even lower).

Top orthology bootstrap values [%]	<i>A. thaliana</i> (total: 14,905)	<i>C. elegans</i> (total: 11,287)
≥ 95	2252	922
≥ 90	2982	1224
≥ 80	4185	1858
≥ 70	5198	2393
≥ 50	7493	3459
≤ 20	2680	4751
≤ 10	1360	3171
≤ 5	930	2452

Table 4.4. Top orthology bootstrap values of RIO analyses.

Query sequences with no orthologs in the current databases are candidates for wrong functional predictions if such predictions are made solely on sequence similarity (as illustrated in Figure 4.1). An example for this is the *A. thaliana* sequence F28P22_13. (Files related to this analysis are available at [http://www.genetics.wustl.edu/eddy/forester/rio_analyses/RIO_paper/F28P22_13/].) This sequence is a zinc-binding dehydrogenase (Pfam: adh_zinc, PF00107). F28P22_13 has been annotated in ATH1.pep.03202001 “as putative cinnamyl-alcohol dehydrogenase”, based on sequence similarity (its top 10 BLAST matches are all cinnamyl-alcohol dehydrogenases with E-values in the range of 10^{-94} if analyzed against all non-redundant GenBank CDS translations+PDB+SwissProt+PIR+PRF on Jan 2, 2002). Cinnamyl-alcohol

dehydrogenase (EC 1.1.1.195) catalyzes the following reaction: cinnamyl alcohol + NADP⁺ = cinnamaldehyde + NADPH (but it can also act on coniferyl alcohol, sinapyl alcohol and 4-coumaryl alcohol) in the flavonoid, stilbene and lignin biosynthesis pathways (Webb, 1992; Wyrambik and Grisebach, 1979). According to the RIO analysis, F28P22_13 has no orthologs (see Figure 4.8 for the corresponding tree and Figure 4.9 for the RIO output). Furthermore its subtree-neighbors above 90%, cinnamyl-alcohol dehydrogenases and NADP-dependent alcohol dehydrogenases (EC 1.1.1.2), exhibit only partial consensus (namely that of some type of NADP-dependent alcohol dehydrogenase, but not EC 1.1.1.2 or EC 1.1.1.195). Hence, F28P22_13 is likely to be a (possibly novel) type of NADP-dependent alcohol dehydrogenase (other than EC 1.1.1.2), possibly a novel type of cinnamyl-alcohol dehydrogenase.

One might expect that each query sequence which appears to have no orthologs is connected with scenario similar to the one described above for F28P22_13. Yet, this is clearly not the case, for the following reasons: (i) Gene duplications might not be followed by functional modification (many Pfam families are composed of sequences which have all the same function, at least at the resolution of the current annotation). (ii) Some Pfam families are composed solely of sequences originating from closely related (or the same) species (such as PFO2362, the B3 DNA binding domain of higher plants). For such families, query sequences from the same species group are expected to have low orthology values. In such cases the concept of subtree-neighbors and ultra-paralogs is more useful than orthologs. (iii) Erroneous RIO results caused by a insufficient phylogenetic signal (due to short sequences, for example) can lead to low

orthology values. For this reason, RIO also outputs the average bootstrap value for the consensus tree to give the user a hint about the amount of phylogenetic signal in the alignment used.

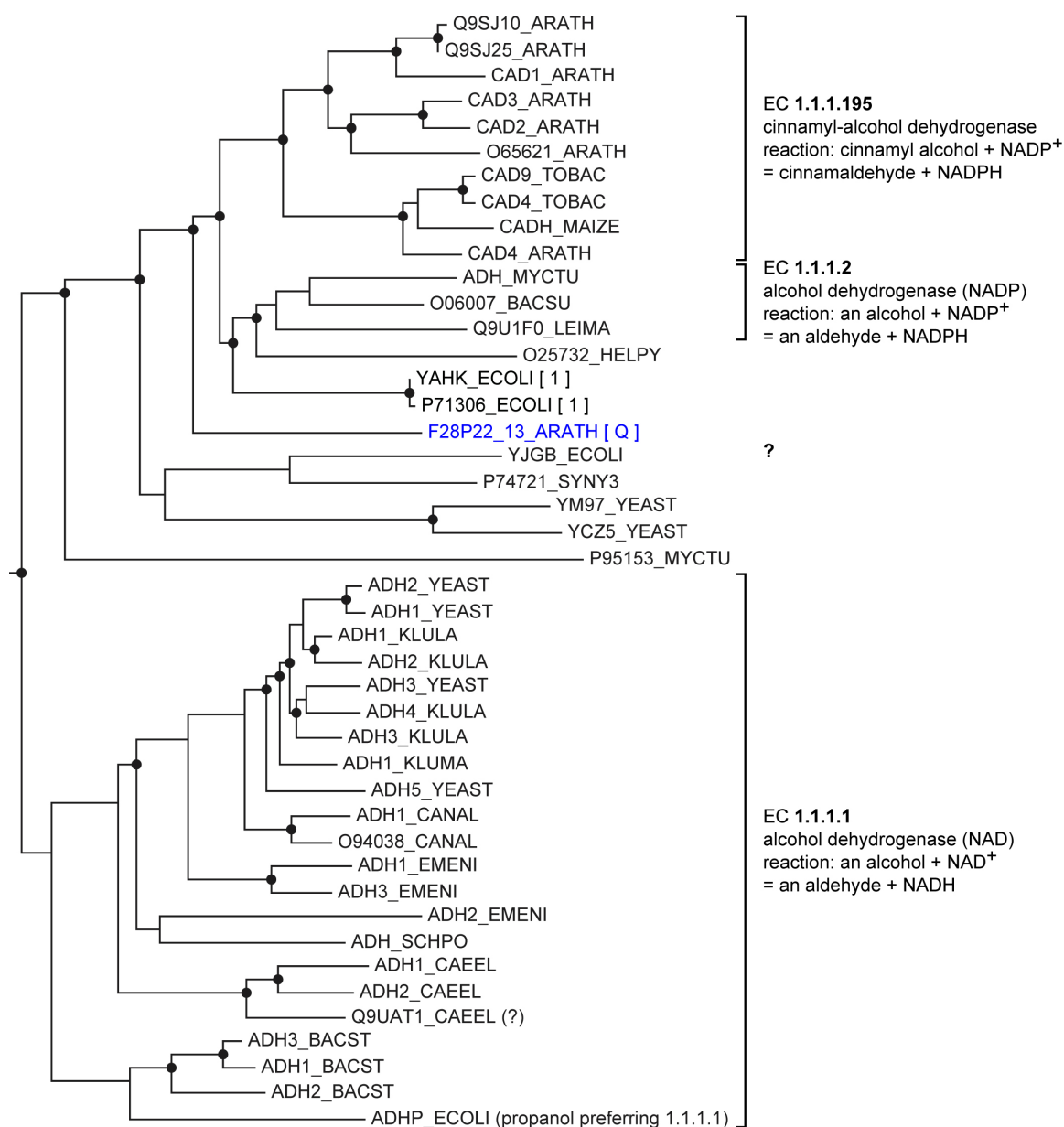


Figure 4.8. A phylogenetic tree for zinc-binding dehydrogenases produced by RIO.

This tree is based on the Pfam alignment adh_zinc (PF00107) and is a subtree of a larger tree. It has been calculated by the neighbor joining method (Felsenstein, 2001) using maximum likelihood pairwise distances (Strimmer and von Haeseler, 1996) based on the BLOSUM 62

matrix (Henikoff and Henikoff, 1992). Gene duplication are indicated by circles (inferred by our SDI algorithm (Zmasek and Eddy, 2001b)). The tree was rooted by minimizing the sum of duplications. The tree image was produced by ATV (Zmasek and Eddy, 2001a). Species are represented by their SWISS-PROT abbreviations (ARATH: *Arabidopsis thaliana*, TOBAC: *Nicotiana tabacum*, MAIZE: *Zea mays*, MYCTU: *Mycobacterium tuberculosis*, BACSU: *Bacillus subtilis*, LEIMA: *Leishmania major*, HELPY: *Helicobacter pylori*, SYNY3: *Synechocystis* sp. strain PCC 6803, YEAST: *Saccharomyces cerevisiae*, KLULA: *Kluyveromyces lactis*, KLUMA: *Kluyveromyces marxianus*, CANAL: *Candida albicans*, EMENI: *Emericella nidulans*, SCHPO: *Schizosaccharomyces pombe*, CAEEL: *Caenorhabditis elegans*, BACST: *Bacillus stearothermophilus*). The *A. thaliana* query sequence F28P22_13 is labeled with Q. The bootstrap orthology values for potential orthologs are indicated in brackets. According to this tree, F28P22_13 has no orthologs.

Sequence	Description	o[%]	n[%]	s[%]	distance
YAHK_ECOLI/14-343	HYPOTHETICAL ZINC-TYPE ALCOHOL DEHYDROGENASE-LIKE PROTEIN IN BETT-PRPR IN TERGENIC REGION.	1	98	0	0.923480
P71306_ECOLI/14-343	SIMILAR TO CINNAMYL-ALCOHOL DEHYDROGENASE OF P. CRISPUM.	1	98	0	0.923760
XYLB_PSEPU/14-365	ARYL-ALCOHOL DEHYDROGENASE (EC 1.1.1.90) (BENZYL ALCOHOL DEHYDROGENASE) (BADH).	1	1	1	1.768320
Q9SJ10_ARATH/18-348	PUTATIVE CINNAMYL-ALCOHOL DEHYDROGENASE.	0	99	0	0.788690
Q9SJ25_ARATH/18-349	PUTATIVE CINNAMYL-ALCOHOL DEHYDROGENASE.	0	99	0	0.801010
CAD1_ARATH/24-353	CINNAMYL-ALCOHOL DEHYDROGENASE 1 (EC 1.1.1.195) (CAD).	0	99	0	0.813150
CAD2_ARATH/20-349	CINNAMYL-ALCOHOL DEHYDROGENASE ELI3-1 (EC 1.1.1.195) (CAD).	0	99	0	0.888760
O65621_ARATH/25-354	CINNAMYL ALCOHOL DEHYDROGENASE-LIKE PROTEIN, SUBUNIT A (CINNAMYL ALCOHOL DEHYDROGENASE-LIKE PROTEIN, LCADA).	0	99	0	0.905050
CAD3_ARATH/20-349	CINNAMYL-ALCOHOL DEHYDROGENASE ELI3-2 (EC 1.1.1.195) (CAD).	0	99	0	0.911850
CAD4_TOBAC/21-350	CINNAMYL-ALCOHOL DEHYDROGENASE (EC 1.1.1.195) (CAD).	0	99	0	0.996520
CAD9_TOBAC/21-350	CINNAMYL-ALCOHOL DEHYDROGENASE (EC 1.1.1.195) (CAD).	0	99	0	0.998400
CADH_MAIZE/21-350	CINNAMYL-ALCOHOL DEHYDROGENASE (EC 1.1.1.195) (CAD) (BROWN-MIDRIB 1 PROTEIN).	0	99	0	1.036040
CAD4_ARATH/22-351	CINNAMYL-ALCOHOL DEHYDROGENASE 2 (EC 1.1.1.195) (CAD).	0	99	0	1.039940
ADH_MYCTU/15-343	NADP-DEPENDENT ALCOHOL DEHYDROGENASE (EC 1.1.1.2).	0	98	0	0.935120
O06007_BACSU/18-346	NADP-DEPENDENT ALCOHOL DEHYDROGENASE.	0	98	0	0.955200
Q9U1F0_LEIMA/16-346	NADP-DEPENDENT ALCOHOL HYDROGENASE.	0	98	0	0.968460
O25732_HELPY/16-343	CINNAMYL-ALCOHOL DEHYDROGENASE ELI3-2 (CAD).	0	97	0	1.123840
YM97_YEAST/20-353	HYPOTHETICAL ZINC-TYPE ALCOHOL DEHYDROGENASE-LIKE PROTEIN IN PRE5-FET4 IN TERGENIC REGION.	0	76	0	1.388040
YC25_YEAST/20-354	HYPOTHETICAL ZINC-TYPE ALCOHOL DEHYDROGENASE-LIKE PROTEIN YCR105W (EC 1.1.1.-).	0	76	0	1.439990
P74721_SYNY3/13-333	ZINC-CONTAINING ALCOHOL DEHYDROGENASE FAMILY.	0	60	0	1.354540
YJGB_ECOLI/15-337	HYPOTHETICAL ZINC-TYPE ALCOHOL DEHYDROGENASE-LIKE PROTEIN IN GNTV-LEUX IN TERGENIC REGION (ORF1).	0	60	0	1.368110
P95153_MYCTU/25-346	ADHA.	0	9	0	1.931400
ADH3_BACST/12-336	ALCOHOL DEHYDROGENASE (EC 1.1.1.1) (ADH-HT).	0	8	0	1.272530
...					

Figure 4.9. RIO output for the *A. thaliana* protein F28P22_13 analyzed against the Pfam adh_zinc domain alignment (PF00107).

For an explanation of the output see Figure 4.7. For space reasons some lines of the output are not shown (“...”) (the complete output is available at http://www.genetics.wustl.edu/eddy/forester/rio_analyses/RIO_paper/F28P22_13/). The output is sorted by orthology values. According to this RIO analysis the query sequence is likely to have no orthologs in this alignment. In contrast, the query probably has subtree-neighbors which are cinnamyl-alcohol dehydrogenases (EC 1.1.1.195), NADP-dependent alcohol dehydrogenases (EC 1.1.1.2), as well as other zinc-containing alcohol dehydrogenases.

4.5.2.2.4 Inconsistency between orthology bootstrap values and sequence similarity

Forth, we were interested in the number of sequences in the two proteomes for which the orthology bootstrap values do not correspond to sequence similarity (Table 4.5). Such disagreements can be caused by the situation illustrated in Figure 4.2. To determine these numbers, we used to following rules. Two thresholds for orthology bootstrap values were chosen: *O*, the minimum for being an ortholog (e.g. 90%) and *N*, the maximum for not being

an ortholog (e.g.10%). Furthermore, a maximal ratio R for the distance of the query to non-orthologs to the distance of the query to orthologs was chosen (e.g. 0.5). In order for being counted as exhibiting disagreement between the orthology bootstrap values and sequence similarity a query sequence had to fulfill the following two conditions: (i) it must have a least one ortholog with bootstrap orthology value above or equal to O , and (ii) *all* sequences in the alignment with bootstrap orthology values above N , must have distance ratios smaller or equal to R for at least one sequence with bootstrap orthology lower or equal to N . Sequences from the following species were ignored in this analysis (since they were the species of the query sequence or related to it): *A. thaliana* proteome: *Rosidae* (*A. thaliana*, *Pisum sativum*, *Glycine max*, *Cucurbita maxima*, *Cucumis sativus*, *Brassica campestris*, *Brassica napus*, *Citrus unshiu*, *Citrus sinensis*, *Theobroma cacao*, *Gossypium hirsutum*); *C. elegans* proteome: nematodes (*C. elegans*, *Caenorhabditis briggsae*, *Haemonchus contortus*, *Ascaris suum*).

Thresholds			Number of query sequences	
O	N	R	<i>A. thaliana</i>	<i>C. elegans</i>
90%	10%	0.5	128	19
90%	10%	0.8	328	102
80%	20%	0.5	254	45

Table 4.5. The numbers of sequences for which the orthology bootstrap values do not correspond to sequence similarity.

Manual inspection of the RIO output leads to the following, somewhat unexpected, conclusion. In many cases a discrepancy between orthology bootstrap values and sequence similarity is caused by orthologs in only phylogenetically distant (relatively to the query sequence) species. This can lead to errors if functional annotation is blindly transferred from these orthologs to the query. As an example of this, the results of analyzing the *A. thaliana* O-methyltransferase F16P17_38 are shown in Figures 4.10 and 4.11. (Complete files are [at](http://www.genetics.wustl.edu/eddy/forester/rio_analyses/RIO_paper/F16P17_38/) [\[http://www.genetics.wustl.edu/eddy/forester/rio_analyses/RIO_paper/F16P17_38/\]](http://www.genetics.wustl.edu/eddy/forester/rio_analyses/RIO_paper/F16P17_38/).) Even though the F16P17_38 sequence is orthologous to the bacterial hydroxyneurosporene methyltransferases (EC 2.1.1.-) (Armstrong et al., 1989) it would be dangerous to annotate it as such. A more reasonable annotation for this query would be to annotate it based on subtree-neighbors and hence call it a plant O-methyltransferase. An indication of this problem (besides a discrepancy between orthology bootstrap values and sequence similarity) is the meeting of the following three conditions: A query sequence has (i) likely orthologs and (ii) likely subtree-neighbors in other species than the query itself, yet (iii) there is no significant overlap between the orthologs and the subtree-neighbors.

We were unable to find convincing examples in the *C. elegans* and *A. thaliana* proteomes where wrong sequence similarity based annotations might be caused by unequal rates of evolution (as illustrated in Figure 4.2). This is not to say that such cases do not exist in those two proteomes but are likely to be quite rare. Similarly to the issues described in the previous section, the detection of such examples is complicated by the fact that for many cases in which a

discrepancy between orthology bootstrap values and sequence similarity exists, all sequences in the Pfam alignment appear to have to same function, the Pfam family is lineage specific, or the annotations are too poor/confusing to make any kind of inference.

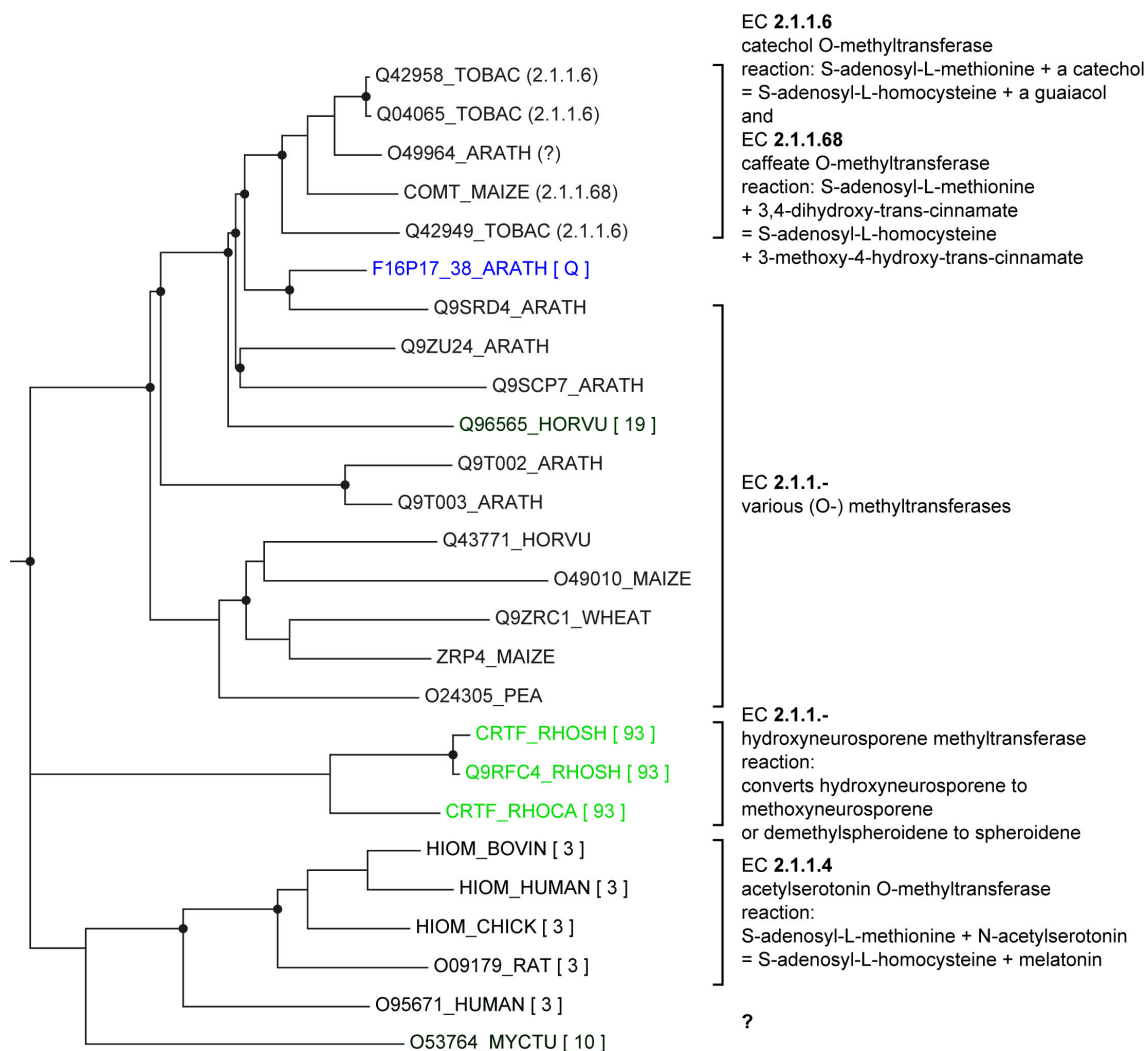


Figure 4.10. A phylogenetic tree for O-methyltransferases produced by RIO.

This tree is based on the Pfam alignment Methyltransf_2 (PF00891). It has been constructed in the same manner as the tree in Figure 4.8. (TOBAC: *Nicotiana tabacum*, ARATH: *Arabidopsis thaliana*, MAIZE: *Zea mays*, HORVU: *Hordeum vulgare*, WHEAT: *Triticum aestivum*, PEA: *Pisum sativum*, RHOSH: *Rhodobacter sphaeroides*, RHOCA: *Rhodobacter capsulatus*, BOVIN: *Bos taurus*, CHICK: *Gallus gallus*, RAT: *Rattus norvegicus*, MYCTU: *Mycobacterium tuberculosis*). The *A. thaliana* query sequence F16P17_38 is labeled with Q. The bootstrap orthology values for potential orthologs are indicated in brackets (the brightness of the green color is proportional to this value). The apparent trifurcation at the root is caused by a branch length of 0.0 (the bacterial hydroxyneurosporene methyltransferases subtree and the plant O-methyltransferases subtree are connected by a speciation event). Inferred gene duplication are indicated by circles. According to this tree, F16P17_38 has orthologs only in bacteria.

Sequence	Description	o[%]	n[%]	s[%]	distance
Q9RFC4_RHOSH/112-349	CRTF.	93	0	0	1.666990
CRTF_RHOCA/137-367	HYDROXYNEUROSPORENE METHYLTRANSFERASE (EC 2.1.1.1.-) (O-METHYLASE).	93	0	0	1.707230
CRTF_RHOSH/109-346	HYDROXYNEUROSPORENE METHYLTRANSFERASE (EC 2.1.1.1.-) (O-METHYLASE).	93	0	0	1.713780
Q96565_HORVU/110-352	CAFFEIC ACID O-METHYLTRANSFERASE (EC 2.1.1.6) (CATECHOL O- METHYLTRANSFERASE) (O-METHYLTRANSFERASE).	19	43	0	0.913640
O53764_MYCTU/71-316	PUTATIVE METHYLTRANSFERASE.	10	0	0	1.602520
O95671_HUMAN/349-595	ASMTL PROTEIN.	3	0	0	1.580280
O09179_RAT/80-322	HYDROXYINDOLE-O-METHYLTRANSFERASE (EC 2.1.1.4) (ACETYLSEROTONIN O- METHYLTRANSFERASE) (HYDROXYINDOLE O-METHYLTRANSFERASE).	3	0	0	1.674460
HIOM_HUMAN/79-322	HYDROXYINDOLE O-METHYLTRANSFERASE (EC 2.1.1.4) (HIOMT) (ACETYLSEROTONIN O-METHYLTRANSFERASE) (ASMT).	3	0	0	1.749550
HIOM_BOVIN/79-322	HYDROXYINDOLE O-METHYLTRANSFERASE (EC 2.1.1.4) (HIOMT) (ACETYLSEROTONIN O-METHYLTRANSFERASE) (ASMT).	3	0	0	1.764290
HIOM_CHICK/81-323	HYDROXYINDOLE O-METHYLTRANSFERASE (EC 2.1.1.4) (HIOMT) (ACETYLSEROTONIN O-METHYLTRANSFERASE) (ASMT).	3	0	0	1.787620
Q9SRD4_ARATH/100-342	PUTATIVE CATECHOL O-METHYLTRANSFERASE.	0	100	0	0.526350
O49964_ARATH/97-338	O-METHYLTRANSFERASE 1.	0	72	0	0.632160
Q42958_TOBAC/99-340	CATECHOL O-METHYLTRANSFERASE (EC 2.1.1.6).	0	72	0	0.639820
Q04065_TOBAC/99-340	CATECHOL O-METHYLTRANSFERASE.	0	72	0	0.649210
Q42949_TOBAC/100-342	CATECHOL O-METHYLTRANSFERASE (EC 2.1.1.6).	0	72	0	0.663620
COMT_MAIZE/100-341	CAFFEIC ACID 3-O-METHYLTRANSFERASE (EC 2.1.1.68) (S-ADENOSYSL-L- METHIONINE;CAFFEIC ACID 3-O-METHYLTRANSFERASE) (COMT).	0	72	0	0.721520
Q9SCP7_ARATH/93-336	CAFFEIC ACID O-METHYLTRANSFERASE-LIKE PROTEIN.	0	37	0	0.988010
Q9ZU24_ARATH/96-339	F5F19.5 PROTEIN.	0	36	0	0.701190
Q9T003_ARATH/103-358	O-METHYLTRANSFERASE-LIKE PROTEIN.	0	2	0	0.974450
Q9T002_ARATH/46-301	O-METHYLTRANSFERASE-LIKE PROTEIN.	0	2	0	1.100820
ZRP4_MAIZE/94-341	O-METHYLTRANSFERASE ZRP4 (EC 2.1.1.-) (OMT).	0	2	0	1.116310
O24305_PEA/93-337	6A-HYDROXYMAACKIAIN METHYLTRANSFERASE.	0	2	0	1.182120
Q43771_HORVU/117-367	CATECHOL O-METHYLTRANSFERASE (EC 2.1.1.6).	0	2	0	1.264630
Q9ZRC1_WHEAT/97-359	O-METHYLTRANSFERASE.	0	2	0	1.270800
O49010_MAIZE/90-340	HERBICIDE SAFENER BINDING PROTEIN.	0	2	0	1.530230

Figure 4.11. RIO output for the *A. thaliana* protein F16P17_38 analyzed against the Pfam Methyltransf_2 domain alignment (PF00891).

For an explanation of the output see Figure 4.7. The output is sorted by orthology values. According to this RIO analysis the orthologs of F16P17_38 are bacterial hydroxyneurosporene methyltransferases. These contrast with the subtree-neighbors of F16P17_38 which are all plant O-methyltransferases.

4.6 Conclusions

In this work we present RIO, a procedure for automated phylogenomics, in particular – automated orthology detection. A major caveat of all phylogenetic analyses is the unreliability of the resulting trees. Therefore, inference of gene duplications is performed over bootstrap resampled phylogenetic trees to estimate the reliability of the orthology assignments. In addition, we introduce supplementary concepts which may be useful for functional prediction: super-orthologs, ultra-paralogs and subtree-neighbors. Initial testing and evaluation of RIO was performed by analyzing the *A. thaliana* and *C. elegans* proteomes.

It appears that the RIO procedure is particularly useful for the detection of first representatives of novel protein subfamilies. Sequence similarity based methods can be misleading in these cases since every query is always “most similar to something”, whereas RIO can detect the absence of orthologs.

Super-orthology is a very stringent criterion. If a query sequence is likely to have super-orthologs, they represent an excellent source to transfer functional annotation from. In contrast, the absence of super-orthologs does not imply that a function for a query sequence cannot be inferred (in the two proteomes analyzed in this work, most sequences appear to have no super-orthologs in Pfam 6.6).

Ultra-paralogs are sequences in the same species as the query and are likely to be the result of recent duplications and therefore might not have yet undergone much functional divergence. Operationally, splice variants can also be

thought of as ultra-paralogs (at least as long as protein sequences are considered).

Subtree-neighbors have two uses: (i) The commonly used “subtree concept”: If the subtree-neighbors of the query sequence exhibit (partial) consensus in their functional annotations, the elements in which they agree might be used to infer a (partial) function for the query. This is useful for query sequences which appear to have no orthologs in the current databases. (ii) For query sequences which do have orthologs, absence of overlap between the sequences considered orthologous and those which appear to be subtree-neighbors should be treated as a red flag. It might indicate that the orthologs are in phylogenetically distant species relative to the query. Transferring annotation from such orthologs is risky. In this case, subtree-neighbors are a more reliable source to transfer annotation from.

RIO outputs warnings if the distance of the query sequence to other sequences is unusually short or long. The usefulness of this was not investigated in this work.

A RIO procedure based on Pfam alignments analyzes each protein domain individually since Pfam is a protein family database based on individual domains (Bateman et al., 2000). While this seems to be a disadvantage it also has a powerful advantage: Due to domain shuffling many proteins are mosaic proteins, proteins composed of domains with different evolutionary histories (Doolittle, 1985; Patthy, 1985). For such proteins it makes much sense to analyze each domain individually. Furthermore, mosaic proteins from sufficiently distant species might be impossible to be aligned over more than one domain at the time,

since they are unlikely to exhibit the same domain organization. The same is true for multiple copies of the same domain in protein: Each of them are analyzed individually (such proteins oftentimes differ in their number of domain copies and could therefore not be aligned from end to end for the whole family).

RIO's most serious drawback is its reliance on a reasonably strong phylogenetic signal in the alignment. Additionally, if the alignment does not contain enough sequences, the result might be meaningless. RIO is obviously also dependent on the quality of the species tree used (in particular for *C. elegans*: currently, it is not clear whether a clade including both nematodes and arthropods exists, the so called ecdysozoa; or whether a more classic view of animal evolution holds true).

In order to make RIO more time efficient it can use precalculated pairwise distances. This allows analyzing a complete proteome in a few weeks utilizing about ten average personal computers. One query sequence can be analyzed against an alignment of 221 sequences in about two minutes on one average PC (Pentium III, 800Mhz). In order for RIO to be used on-line we produced a parallelized version.

In general, the concept of "consensus" is very important in this work (for example consensus between subtree-neighbors, or between subtree-neighbors and orthologs). A useful future extension would be to incorporate automated consensus detection into RIO. This would include annotation of internal nodes of a gene tree with a "biological function". Automated consensus detection is trivial for a highly formalized notation system, such as EC numbers (the consensus of EC 1.1.1.3 and EC 1.1.1.23 is EC 1.1.1, a oxidoreductase acting on the CH-OH

group of donors with NAD⁺ or NADP⁺ as acceptor (Webb, 1992)). Obviously, it is much more difficult to analyze natural language annotations in the same manner, yet this could be accomplished by utilizing the set of structured vocabularies of the Gene Ontology (GO) project (Gene-Ontology-Consortium, 2001) [<http://www.geneontology.org/>].

4.7 Acknowledgements

This work was supported primarily by a grant from Monsanto Company, and also by the Howard Hughes Medical Institute and grant HGo1363 from the NIH National Human Genome Research Institute.

4.8 Appendix A: Precalculation of pairwise distances

Input: Pfam full alignment A .

Output: “aln” file containing modified full alignment

“hmm” file containing a profile HMM

“nbd” file containing pairwise distances

“bsp” file bootstrap positions file

“pwd” file containing pairwise distances for bootstrap resampled alignment

1. If necessary: remove certain sequences (species not in master species tree) from alignment A .
2. If A does not contain enough sequences (<6), abort.
3. Run `hmmbuild` on A , resulting in alignment A' (using the same options as were used to build the original HMM for A).
4. Keep A' as “aln” file.
5. Run `hmmbuild` with “--hand” option on A' , resulting in HMM H' (using the same options as were used to build the original HMM for A).
6. Calibrate H' with `hmmcalibrate` and keep as “hmm” file.
7. remove non-match columns from A' , resulting in alignment A'' .

- 8.** Calculate pairwise distances for A ", resulting in the "nbd" file (non-bootstrapped distances).
- 9.** Bootstrap resample A ", resulting in the "bsp" file (bootstrap positions file).
- 10.** Calculate pairwise distances for bootstrapped A ", resulting in the "pwd" file.

4.9 Appendix B: Speciation Duplication Inference combined with rooting

Input : binary gene tree G , rooted binary species tree S .

Output: G with "duplication" or "speciation" assigned to each internal node and rooted in such a way that the sum of duplications is minimized.

SDIunrooted(G, S)

root gene tree G at the midpoint of a branch of choice;

set $B = \text{getBranchesInOrder}(G);$

$\text{SDIse}(G, S)$ [see chapter 3 or (Zmasek and Eddy, 2001b)];

for each branch b in B :

 set $n_1 = \text{child 1 of root of } G$;

 set $n_2 = \text{child 2 of root of } G$;

 root G at the midpoint of branch b ;

$\text{updateM}(n_1, n_2);$

 if (sum of duplications in $G < d_{\min}$):

 set $d_{\min} = \text{sum of duplications in } G$;

 set $G_{d\min} = G$;

return $G_{d\min}$;

updateM(n_1 , n_2)

```
set r = root of G;  
if ( child 1 of r ==  $n_1$  || child 2 of r ==  $n_1$  ):  
    calculateMforNode(  $n_1$  );  
else:  
    calculateMforNode(  $n_2$  );  
calculateMforNode( r );
```

calculateMforNode(n)

```
if ( !n.isExternal() ):  
    set a = M( child 1 of n );  
    set b = M( child 2 of n );  
    while ( a != b ):  
        if ( a > b ):  
            set a = parent of a;  
        else:  
            set b = parent of b;  
    set M( n ) = a;  
    if ( M( n ) == M( child 1 of n ) || M( n ) == M( child 2 of n ) ):  
        n is duplication;  
    else:  
        n is speciation;
```

getBranchesInOrder(G)

```
set n = root of G;

set i = 0;

while !( n == root && indicator of n == 2 ):

    if ( n != external && indicator of n != 2 ):

        if ( indicator of n == 0 ):

            set indicator of n = 1;

            set n = child 1 of n;

        else:

            set indicator of n = 2;

            set n = child 2 of n;

        if ( parent of n != root ):

            set B[ i ] = branch connecting n and parent of n;

        else:

            set B[ i ] = branch connecting child 1 of root and child
            2 of root;

        set i = i + 1;

    else:

        if ( parent of n != root && n != external ):

            set B[ i ] = branch connecting n and parent of n;

            set i = i + 1;

        set n = parent of n;

return B;
```

5 Conclusions and future directions

In this work, RIO, a procedure for automated phylogenomics was developed and evaluated.

As pointed out in chapter 4, RIO is particularly useful for the automated detection of first representatives of previously unknown protein subfamilies. Additionally, RIO can be used to prioritize further (experimental) studies. RIO allows to automatically scan for proteins which show “unexpected” properties (such as unusual branch lengths, inconsistency between similarity and orthology, inconsistency between subtree-neighbors and orthologs). On the other hand, RIO can also be employed to scan for the opposite, namely sequences which do not show any unusual properties and for which a function can be inferred with confidence.

The resolution which is achievable with RIO is dependent primarily on three things: (i) the amount of phylogenetic signal in the alignment used for tree construction, (ii) the resolution with which the sequences in this alignment are annotated, and (iii) the number of sequences in the alignment.

Related to the question of resolution is the issue of the “transitive annotation catastrophe”. This is caused by automated annotation systems which are prone to propagate erroneous annotations from one sequence to another, leading to an exponentially growing number of misannotated sequences. As for

similarity based methods, RIO is expected to be less prone to already existing incorrect predictions if instead of “top 1” hits, only information based on consensus is used. Furthermore, one might expect that a system like RIO is less likely to spread wrong annotations since its results can easily be interpreted to determine whether added annotation (besides family membership) is reasonable or not (absence of orthologs and/or absence of consensus). This is of course only true as long as the number of misannotated sequences is small compared to the number of correctly annotated ones.

Besides incorporating automated consensus detection into RIO, as discussed in section 4.6, the following future developments might prove to be of some value.

Biochemical- and signaling-pathway analysis: Whereas functional prediction for individual sequences is an important and difficult task, detailed knowledge about sequence function is only one step towards the even more important goal of biochemical- and signaling-pathway analysis and reconstruction (and simulation). This, of course, is a precondition for rational pathway engineering and whole organism analysis and simulation. Combining RIO with a protein function database (containing information about the substrates and products for each known enzyme, targets and effectors for signaling proteins) could eventually lead to the automated reconstruction of pathways.

Association of sequence patterns with biological functions: Overlaying biological properties as well as amino acid sequences over a gene tree could be used to determine the sequence pattern(s) associated with a given biological

property associated with a subtree/subfamily. While this is obviously not tied to the RIO procedure as such, it is another example of how phylogenetic analysis could be used for sequence analysis.

Curated subtree definitions: This is a possible addition to a protein or domain alignment database (such as Pfam), allowing for subtree/subfamily-level classification. In this approach, subtrees are defined by two so-called "outposts". The "outposts" of a given subtree are sequences whose last common ancestor is thought to be the ancestral sequence of the subtree. For example, the BAX subfamily in Figure 1.6 could be defined by the following two "outposts": "BAXA_MOUSE" and "BAXD_HUMAN". Combining this approach with phylogenomics makes it straight forward to determine whether a query sequence is a first representative of a novel subfamily or whether it belongs to a defined subtree (as well as to determine to which subtree it belongs to).

6 Bibliography

- Adachi, J., Cao, Y. and Hasegawa, M. (1993) Tempo and mode of mitochondrial DNA evolution in vertebrates at the amino acid sequence level: rapid evolution in warm-blooded vertebrates. *J Mol Evol*, **36**, 270-281.
- Adachi, J. and Hasegawa, M. (1992) Amino acid substitution of proteins coded for in mitochondrial DNA during mammalian evolution. *Jpn J Genet*, **67**, 187-197.
- Adachi, J. and Hasegawa, M. (1996) Model of amino acid substitution in proteins encoded by mitochondrial DNA. *J Mol Evol*, **42**, 459-468.
- Adams, M.D., Celniker, S.E., Holt, R.A. and ... (2000) The genome sequence of *Drosophila melanogaster*. *Science*, **287**, 2185-2195.
- Aguinaldo, A.M., 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.
- Anderson, S., Bankier, A.T., Barrell, B.G., de Bruijn, M.H., Coulson, A.R., Drouin, J., Eperon, I.C., Nierlich, D.P., Roe, B.A., Sanger, F., Schreier, P.H., Smith, A.J., Staden, R. and Young, I.G. (1981) Sequence and organization of the human mitochondrial genome. *Nature*, **290**, 457-465.
- Andrade, M.A., Perez-Iratxeta, C. and Ponting, C.P. (2001) Protein repeats: structures, functions, and evolution. *J Struct Biol*, **134**, 117-131.
- Apweiler, R., Attwood, T.K., Bairoch, A., Bateman, A., Birney, E., Biswas, M., Bucher, P., Cerutti, L., Corpet, F., Croning, M.D., Durbin, R., Falquet, L., Fleischmann, W., Gouzy, J., Hermjakob, H., Hulo, N., Jonassen, I., Kahn, D., Kanapin, A., Karavidopoulou, Y., Lopez, R., Marx, B., Mulder, N.J., Oinn, T.M., Pagni, M., Servant, F., Sigrist, C.J. and Zdobnov, E.M. (2000) InterPro--an integrated documentation resource for protein families, domains and functional sites. *Bioinformatics*, **16**, 1145-1150.
- Arabidopsis-Initiative. (2000) Analysis of the genome sequence of the flowering plant *Arabidopsis thaliana*. *Nature*, **408**, 796-815.

- Armstrong, G.A., Alberti, M., Leach, F. and Hearst, J.E. (1989) Nucleotide sequence, organization, and nature of the protein products of the carotenoid biosynthesis gene cluster of *Rhodobacter capsulatus*. *Mol Gen Genet*, **216**, 254-268.
- Atteson, K. (1997) The performance of the neighbor-joining method of phylogeny reconstruction. In Rzhetsky, A. (ed.), *Mathematical Hierarchies and Biology*. American Mathematical Society, pp. 133-148.
- Avise, J.C. (1994) *Molecular markers, natural history and evolution*. Chapman & Hall, New York.
- Ayala, F.J. (1999) Molecular clock mirages. *Bioessays*, **21**, 71-75.
- Bains, W. (1986) MULTAN: a program to align multiple DNA sequences. *Nucleic Acids Res*, **14**, 159-177.
- Bairoch, A. and Apweiler, R. (2000) The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000. *Nucleic Acids Res*, **28**, 45-48.
- Bairoch, A., Bucher, P. and Hofmann, K. (1997) The PROSITE database, its status in 1997. *Nucleic Acids Res*, **25**, 217-221.
- 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.
- Banaszak, L.J. and Bradshaw, R.A. (1975) Malate dehydrogenase. In Boyer, P.D. (ed.), *The Enzymes*. Academic Press, New York, Vol. 11, pp. 369-396.
- Barns, S.M., Delwiche, C.F., Palmer, J.D. and Pace, N.R. (1996) Perspectives on archaeal diversity, thermophily and monophyly from environmental rRNA sequences. *Proc Natl Acad Sci U S A*, **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.
- Blattner, F.R., Plunkett, G., 3rd, Bloch, C.A., Perna, N.T., Burland, V., Riley, M., Collado-Vides, J., Glasner, J.D., Rode, C.K., Mayhew, G.F., Gregor, J., Davis, N.W., Kirkpatrick, H.A., Goeden, M.A., Rose, D.J., Mau, B. and Shao, Y. (1997) The complete genome sequence of *Escherichia coli* K-12. *Science*, **277**, 1453-1474.
- Brown, H., Sanger, F. and Kitai, R. (1955) The structure of pig and sheep insulins. *Biochem J*, **60**, 556-565.

- Buneman, P. (1971) The recovery of trees from measures of dissimilarity. In Tautu, P. (ed.), *Mathematics in the archeological and historical sciences*. Edinburgh University Press, Edinburgh, pp. 387-395.
- Burger, H., Wagemaker, G., Leunissen, J.A.M. and Dorssers, L.C.J. (1994) Molecular evolution of Interleukin-3. *Journal of Molecular Evolution*, **39**, 255-267.
- C.elegans-Sequencing-Consortium. (1998) Genome sequence of the nematode *C. elegans*: a platform for investigating biology. *Science*, **282**, 2012-2018.
- Calabretta, R., Nolfi, S., Parisi, D. and Wagner, G.P. (2000) Duplication of modules facilitates the evolution of functional specialization. *Artif Life*, **6**, 69-84.
- Camin, J.H. and Sokal, R.R. (1965) A method for deducing branching sequences in phylogeny. *Evolution*, **19**, 311-326.
- Cavalli-Sforza, L.L. and Edwards, A.W.F. (1967) Phylogenetic analysis. Models and estimation procedures. *Evolution*, **32**, 550-570.
- Chan, L. (1993) RNA editing: exploring one mode with apolipoprotein B mRNA. *Bioessays*, **15**, 33-41.
- Chao, D.T. and Korsmeyer, S.J. (1998) BCL-2 family: regulators of cell death. *Annu Rev Immunol*, **16**, 395-419.
- Chen, K., Durand, D. and Farach-Colton, M. (2000) Notung: dating gene duplications using gene family trees. *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, Cambridge, MA.
- Corpet, F. (1988) Multiple sequence alignment with hierarchical clustering. *Nucleic Acids Res*, **16**, 10881-10890.
- Costanzo, M.C., Crawford, M.E., Hirschman, J.E., Kranz, J.E., Olsen, P., Robertson, L.S., Skrzypek, M.S., Braun, B.R., Hopkins, K.L., Kondu, P., Lengieza, C., Lew-Smith, J.E., Tillberg, M. and Garrels, J.I. (2001) YPD, PombePD and WormPD: model organism volumes of the BioKnowledge library, an integrated resource for protein information. *Nucleic Acids Res*, **29**, 75-79.
- Dayhoff, M.O. (1976) The origin and evolution of protein superfamilies. *Fed Proc*, **35**, 2132-2138.

- 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*. Natl Biomed Res Found, Silver Springs, MD, Vol. 5, pp. 345-352.
- de Rosa, R., Grenier, J.K., Andreeva, T., Cook, C.E., Adoutte, A., Akam, M., Carroll, S.B. and Balavoine, G. (1999) Hox genes in brachiopods and priapulids and protostome evolution. *Nature*, **399**, 772-776.
- Dennis, D. and Kaplan, N.O. (1960) D and L-lactic acid dehydrogenase in *Lactobacillus plantarum*. *J. Biol. Chem.*, **235**, 810-818.
- Doolittle, R.F. (1985) The genealogy of some recently evolved vertebrate proteins. *Trends Biochem Sci*, **10**, 233-237.
- Doolittle, R.F. (1995) The multiplicity of domains in proteins. *Annu Rev Biochem*, **64**, 287-314.
- Doolittle, R.F. (1998) Microbial genomes opened up. *Nature*, **392**, 339-342.
- Doolittle, R.F. and Blomback, B. (1964) Amino-acid sequence investigations of fibrinopeptides from various mammals: evolutionary implications. *Nature*, **202**, 147-152.
- Durbin, R., Eddy, S.R., Krogh, A. and Mitchison, G. (1998) *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge University Press, Cambridge, UK.
- Eck, R.V. and Dayhoff, M.O. (1966) Inferences from protein sequence comparisons. In *Atlas of protein sequence and structure*. Natl Biomed Res Found, Silver Spring, MD, Vol. 3, pp. 161-202.
- Eddy, S.R. (1996) Hidden Markov models. *Curr Opin Struct Biol*, **6**, 361-365.
- Eddy, S.R. (2000) HMMER: Profile hidden Markov models for biological sequence analysis. Washington University, St. Louis, MO.
- Edwards, A.W.F. and Cavalli-Sforza, L.L. (1963) The reconstruction of evolution. *Ann Hum Genet*, **27**, 104-105.
- Edwards, A.W.F. and Cavalli-Sforza, L.L. (1964) Reconstruction of evolutionary trees. In McNeill, J. (ed.), *Phenetic and Phylgenetic Classification*. Systematics Association Publication, London, Vol. 6, pp. 67-66.
- Edwards, A.W.F. and Cavalli-Sforza, L.L. (1965) A method for cluster analysis. *Biometrics*, **21**, 362-375.

- Eisen, J.A. (1998a) A phylogenomic study of the MutS family of proteins. *Nucleic Acids Res*, **26**, 4291-4300.
- Eisen, J.A. (1998b) Phylogenomics: improving functional predictions for uncharacterized genes by evolutionary analysis. *Genome Res*, **8**, 163-167.
- Eisen, J.A. (2001) Gastrogenomics. *Nature*, **409**, 463, 465-466.
- Eisen, J.A. and Hanawalt, P.C. (1999) A phylogenomic study of DNA repair genes, proteins, and processes. *Mutat Res*, **435**, 171-213.
- Eisen, J.A., Kaiser, D. and Myers, R.M. (1997) Gastrogenomic delights: a movable feast. *Nat Med*, **3**, 1076-1078.
- Eisen, J.A., Sweder, K.S. and Hanawalt, P.C. (1995) Evolution of the SNF2 family of proteins: subfamilies with distinct sequences and functions. *Nucleic Acids Res*, **23**, 2715-2723.
- Eulenstein, O. (1998) Vorhersage von Genduplikationen und deren Entwicklung in der Evolution. *GMD Research Series*, **20**.
- Eulenstein, O., Mirkin, B. and Vingron, M. (1998) Duplication-based measures of difference between gene and species trees. *J Comput Biol*, **5**, 135-148.
- Eulenstein, O. and Vingron, M. (1995) On the equivalence of two tree mapping measures. *Arbeitspapiere der GMD*, **936**.
- Felsenstein, J. (1981a) Evolutionary trees from DNA sequences: a maximum likelihood approach. *J Mol Evol*, **17**, 368-376.
- Felsenstein, J. (1981b) A likelihood approach to character weighting and what it tells us about parsimony and compatibility. *Biol J Linnean Soc*, **16**, 183-196.
- Felsenstein, J. (1982) Numerical methods for inferring phylogenetic trees. *Quart Rev Biol*, **57**, 379-404.
- Felsenstein, J. (1985) Confidence limits on phylogenies: An approach using the bootstrap. *Evolution*, **39**, 783-791.
- Felsenstein, J. (1988) Phylogenies from molecular sequences: inference and reliability. *Annu Rev Genet*, **22**, 521-565.
- Felsenstein, J. (1993) PHYLIP: Phylogeny Inference Package, Version 3.5. University of Washington, Seattle, WA.

- Felsenstein, J. (1996) Inferring phylogenies from protein sequences by parsimony, distance, and likelihood methods. *Methods Enzymol*, **266**, 418-427.
- Felsenstein, J. (2001) PHYLIP: Phylogeny Inference Package, Version 3.6. University of Washington, Seattle, WA.
- Felsenstein, J. and Churchill, G.A. (1996) A Hidden Markov Model approach to variation among sites in rate of evolution. *Mol Biol Evol*, **13**, 93-104.
- Feng, D.F. and Doolittle, R.F. (1987) Progressive sequence alignment as a prerequisite to correct phylogenetic trees. *J Mol Evol*, **25**, 351-360.
- Feng, D.F., Johnson, M.S. and Doolittle, R.F. (1985) Aligning amino acid sequences: comparison of commonly used methods. *J Mol Evol*, **21**, 112-125.
- Fitch, D.H., Bailey, W.J., Tagle, D.A., Goodman, M., Sieu, L. and Slightom, J.L. (1991) Duplication of the gamma-globin gene mediated by L1 long interspersed repetitive elements in an early ancestor of simian primates. *Proc Natl Acad Sci U S A*, **88**, 7396-7400.
- Fitch, W., M. (1966) An improved method of testing for evolutionary homology. *J Mol Biol*, **16**, 9-16.
- Fitch, W., M. (1971) Toward defining the course of evolution: minimum change for a specified tree topology. *Syst Zool*, **20**, 406-416.
- Fitch, W.M. (1970) Distinguishing homologous from analogous proteins. *Syst Zool*, **19**, 99-113.
- Fitch, W.M. and Margoliash, E. (1967) Construction of phylogenetic trees. *Science*, **155**, 279-284.
- Fleischmann, R.D., Adams, M.D., White, O., Clayton, R.A., Kirkness, E.F., Kerlavage, A.R., Bult, C.J., Tomb, J.F., Dougherty, B.A., Merrick, J.M. and et al. (1995) Whole-genome random sequencing and assembly of *Haemophilus influenzae* Rd. *Science*, **269**, 496-512.
- Fornace, A.J., Jr., 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**, 12826-12830.
- Friedman, R. and Hughes, A.L. (2001) Gene duplication and the structure of eukaryotic genomes. *Genome Res*, **11**, 373-381.

- Galperin, M.Y. and Koonin, E.V. (1998) Sources of systematic error in functional annotation of genomes: domain rearrangement, non-orthologous gene displacement and operon disruption. *In Silico Biol*, **1**, 55-67.
- Gene-Ontology-Consortium. (2001) Creating the gene ontology resource: design and implementation. *Genome Res*, **11**, 1425-1433.
- Gilmour, R. (2000) Taxonomic markup language: applying XML to systematic data. *Bioinformatics*, **16**, 406-407.
- Go, M. (1981) Correlation of DNA exonic regions with protein structural units in haemoglobin. *Nature*, **291**, 90-92.
- 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**, 546, 563-547.
- Golding, G.B. and Dean, A.M. (1998) The structural basis of molecular adaptation. *Mol Biol Evol*, **15**, 355-369.
- Goodman, M. (1962) Immunochemistry of the primates and primate evolution. *Ann N.Y Acad Sci*, **102**, 219-234.
- 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.
- Gray, G.S. and Fitch, W.M. (1983) Evolution of antibiotic resistance genes: the DNA sequence of a kanamycin resistance gene from *Staphylococcus aureus*. *Mol Biol Evol*, **1**, 57-66.
- Gribskov, M., McLachlan, A.D. and Eisenberg, D. (1987) Profile analysis: detection of distantly related proteins. *Proc Natl Acad Sci U S A*, **84**, 4355-4358.
- Gu, X. (1999) Statistical methods for testing functional divergence after gene duplication. *Mol Biol Evol*, **16**, 1664-1674.
- Gu, X. (2001) Maximum-likelihood approach for gene family evolution under functional divergence. *Mol Biol Evol*, **18**, 453-464.
- Guigo, R., Muchnik, I. and Smith, T.F. (1996) Reconstruction of ancient phylogenies. *Mol Phylogenet Evol*, **6**, 189-213.

- Haeckel, E. (1866) *Generelle Morphologie der Organismen: Allgemeine Grundzuege der organischen Formen-Wissenschaft, mechanisch begründet durch die von Charles Darwin reformirte Descendenz-Theorie*. Georg Rieme, Berlin.
- Haldane, J.B.S. (1932) *The causes of evolution*. Harper & Brothers Publishers, New York and London.
- Hein, J. (1989) A new method that simultaneously aligns and reconstructs ancestral sequences for any number of homologous sequences, when the phylogeny is given. *Mol Biol Evol*, **6**, 649-668.
- Henikoff, S. and Henikoff, J.G. (1992) Amino acid substitution matrices from protein blocks. *Proc Natl Acad Sci U S A*, **89**, 10915-10919.
- Hennig, W. (1950) *Grundzuege einer Theorie der phylogenetischen Systematik*. Deutscher Centralverlag, Berlin.
- Hennig, W. (1965) Phylogenetic systematics. *Ann Rev Entomol*, **10**, 97-116.
- Hennig, W. (1966) *Phylogenetic systematics*. University of Illinois Press, Urbana, IL.
- Holliday, R. (1964) A mechanism for gene conversion in fungi. *Genet Res*, **5**, 282-304.
- Ingram, V.M. (1961) Gene evolution and the haemoglobins. *Nature*, **189**, 704-708.
- JáJá, J. (1991) *An Introduction to Parallel Algorithms*. Addison-Wesley, Reading, MA.
- Jendrossek, D., Kratzin, H.D. and Steinbuchel, A. (1993) The *Alcaligenes eutrophus* ldh structural gene encodes a novel type of lactate dehydrogenase. *FEMS Microbiol Lett*, **112**, 229-235.
- Jensen, R.A. (2001) Orthologs and paralogs - we need to get it right. *Genome Biol*, **2**, INTERACTIONS1002.
- Jin, L. and Nei, M. (1990) Limitations of the evolutionary parsimony method of phylogenetic analysis. *Mol Biol Evol*, **7**, 82-102.
- Johnson, H.S. (1971) NADP-malate dehydrogenase: photoactivation in leaves of plants with Calvin cycle photosynthesis. *Biochem. Biophys. Res. Commun.*, **43**, 703-709.
- Johnson, M.S. and Doolittle, R.F. (1986) A method for the simultaneous alignment of three or more amino acid sequences. *J Mol Evol*, **23**, 267-278.

- Jones, D.T., Taylor, W.R. and Thornton, J.M. (1992) The rapid generation of mutation data matrices from protein sequences. *Comput Appl Biosci*, **8**, 275-282.
- Jones, F.S., Burgoon, M.P., Hoffman, S., Crossin, K.L., Cunningham, B.A. and Edelman, G.M. (1988) A cDNA clone for cytactin contains sequences similar to epidermal growth factor-like repeats and segments of fibronectin and fibrinogen. *Proc Natl Acad Sci U S A*, **85**, 2186-2190.
- Karlin, S. and Ghandour, G. (1985) Multiple-alphabet amino acid sequence comparisons of the immunoglobulin kappa-chain constant domain. *Proc Natl Acad Sci U S A*, **82**, 8597-8601.
- Kashyap, R.L. and Subas, S. (1974) Statistical estimation of parameters in a phylogenetic tree using a dynamic model of the substitutional process. *J Theor Biol*, **47**, 75-101.
- Kato, K., Tokishita, S., Mandokoro, Y., Kimura, S., Ohta, T., Kobayashi, M. and Yamagata, H. (2001) Two-domain hemoglobin gene of the water flea *Moina macrocopa*: duplication in the ancestral Cladocera, diversification, and loss of a bridge intron. *Gene*, **273**, 41-50.
- Kidd, K.K. and Sgaramella-Zonta, L.A. (1971) Phylogenetic analysis: concepts and methods. *Am J Hum Genet*, **23**, 235-252.
- Kimura, M. (1968a) Evolutionary rate at the molecular level. *Nature*, **217**, 624-626.
- Kimura, M. (1968b) Genetic variability maintained in a finite population due to mutational production of neutral and nearly neutral isoalleles. *Genet Res*, **11**, 247-269.
- Kimura, M. (1983) *The neutral theory of molecular evolution*. Cambridge University Press, Cambridge.
- Kimura, M. (1991) Recent development of the neutral theory viewed from the Wrightian tradition of theoretical population genetics. *Proc Natl Acad Sci U S A*, **88**, 5969-5973.
- King, J.L. (1972) The role of mutation in evolution. In *Proc 6th Berkely Symp Math Stat and Prob*. University of California Press, Berkeley, pp. 69-100.
- King, J.L. and Jukes, T.H. (1969) Non-Darwinian evolution. *Science*, **164**, 788-798.
- Kishino, H., Miyata, T. and Hasegawa, M. (1990) Maximum likelihood inference of protein phylogeny and the origin of chloroplasts. *J Mol Evol*, **31**, 151-160.

Konigsberg, W., Guidotti, G. and Hill, R.J. (1961) The amino acid sequence of the alpha chain of human hemoglobin. *J Biol Chem*, **236**, PC55-56.

Lander, E.S., Linton, L.M., Birren, B., Nusbaum, C., Zody, M.C., Baldwin, J., Devon, K., Dewar, K., Doyle, M., FitzHugh, W., Funke, R., Gage, D., Harris, K., Heaford, A., Howland, J., Kann, L., Lehoczky, J., LeVine, R., McEwan, P., McKernan, K., Meldrim, J., Mesirov, J.P., Miranda, C., Morris, W., Naylor, J., Raymond, C., Rosetti, M., Santos, R., Sheridan, A., Sougnez, C., Stange-Thomann, N., Stojanovic, N., Subramanian, A., Wyman, D., Rogers, J., Sulston, J., Ainscough, R., Beck, S., Bentley, D., Burton, J., Clee, C., Carter, N., Coulson, A., Deadman, R., Deloukas, P., Dunham, A., Dunham, I., Durbin, R., French, L., Grafham, D., Gregory, S., Hubbard, T., Humphray, S., Hunt, A., Jones, M., Lloyd, C., McMurray, A., Matthews, L., Mercer, S., Milne, S., Mullikin, J.C., Mungall, A., Plumb, R., Ross, M., Shownkeen, R., Sims, S., Waterston, R.H., Wilson, R.K., Hillier, L.W., McPherson, J.D., Marra, M.A., Mardis, E.R., Fulton, L.A., Chinwalla, A.T., Pepin, K.H., Gish, W.R., Chissoe, S.L., Wendl, M.C., Delehaunty, K.D., Miner, T.L., Delehaunty, A., Kramer, J.B., Cook, L.L., Fulton, R.S., Johnson, D.L., Minx, P.J., Clifton, S.W., Hawkins, T., Branscomb, E., Predki, P., Richardson, P., Wenning, S., Slezak, T., Doggett, N., Cheng, J.F., Olsen, A., Lucas, S., Elkin, C., Uberbacher, E., Frazier, M., Gibbs, R.A., Muzny, D.M., Scherer, S.E., Bouck, J.B., Sodergren, E.J., Worley, K.C., Rives, C.M., Gorrell, J.H., Metzker, M.L., Naylor, S.L., Kucherlapati, R.S., Nelson, D.L., Weinstock, G.M., Sakaki, Y., Fujiyama, A., Hattori, M., Yada, T., Toyoda, A., Itoh, T., Kawagoe, C., Watanabe, H., Totoki, Y., Taylor, T., Weissenbach, J., Heilig, R., Saurin, W., Artiguenave, F., Brottier, P., Bruls, T., Pelletier, E., Robert, C., Wincker, P., Smith, D.R., Doucette-Stamm, L., Rubenfield, M., Weinstock, K., Lee, H.M., Dubois, J., Rosenthal, A., Platzer, M., Nyakatura, G., Taudien, S., Rump, A., Yang, H., Yu, J., Wang, J., Huang, G., Gu, J., Hood, L., Rowen, L., Madan, A., Qin, S., Davis, R.W., Federspiel, N.A., Abola, A.P., Proctor, M.J., Myers, R.M., Schmutz, J., Dickson, M., Grimwood, J., Cox, D.R., Olson, M.V., Kaul, R., Shimizu, N., Kawasaki, K., Minoshima, S., Evans, G.A., Athanasiou, M., Schultz, R., Roe, B.A., Chen, F., Pan, H., Ramser, J., Lehrach, H., Reinhardt, R., McCombie, W.R., de la Bastide, M., Dedhia, N., Blocker, H., Hornischer, K., Nordsiek, G., Agarwala, R., Aravind, L., Bailey, J.A., Bateman, A., Batzoglu, S., Birney, E., Bork, P., Brown, D.G., Burge, C.B., Cerutti, L., Chen, H.C., Church, D., Clamp, M., Copley, R.R., Doerks, T., Eddy, S.R., Eichler, E.E., Furey, T.S., Galagan, J., Gilbert, J.G., Harmon, C., Hayashizaki, Y., Haussler, D., Hermjakob, H., Hokamp, K., Jang, W., Johnson, L.S., Jones, T.A., Kasif, S., Kasprzyk, A., Kennedy, S., Kent, W.J., Kitts, P., Koonin, E.V., Korf, I., Kulp, D., Lancet, D., Lowe, T.M., McLysaght, A., Mikkelsen, T., Moran, J.V., Mulder, N., Pollara, V.J., Ponting, C.P., Schuler, G., Schultz, J., Slater, G., Smit, A.F., Stupka, E., Szustakowski, J., Thierry-Mieg, D., Thierry-Mieg, J., Wagner, L., Wallis, J., Wheeler, R., Williams, A., Wolf, Y.I., Wolfe, K.H., Yang, S.P., Yeh, R.F., Collins, F., Guyer, M.S., Peterson, J., Felsenfeld, A., Wetterstrand, K.A., Patrinos, A., Morgan, M.J. and Szustakowski, J. (2001) Initial sequencing and analysis of the human genome. *Nature*, **409**, 860-921.

Li, W.H. (1983) Evolution of duplicate genes and pseudogenes. In Koehn, R.K. (ed.), *Evolution of Genes and Proteins*. Sinauer Associates, Sunderland, MA.

- Li, W.H. (1997) *Molecular Evolution*. Sinauer Associates, Sunderland, MA.
- Lio, P. and Goldman, N. (1998) Models of molecular evolution and phylogeny. *Genome Res*, **8**, 1233-1244.
- Lynch, M. and Conery, J.S. (2000) The evolutionary fate and consequences of duplicate genes. *Science*, **290**, 1151-1155.
- Maddison, D.R., Swofford, D.L. and Maddison, W.P. (1997) Nexus: An extensible file format for systematic information. *Syst. Biol.*, **46**, 590-621.
- Margoliash, E., Fitch, W., M. and Dickerson, R.E. (1968) Molecular expression of evolutionary phenomena in the primary and tertiary structures of cytochrome c. *Brookhaven Symp Biol*, **21**, 259-305.
- Mau, B., Newton, M.A. and Larget, B. (1996) Bayesian phylogenetic inference via Markov chain Monte Carlo methods. *Technical Report, Statistics Department, University of Wisconsin-Madison*, **961**.
- Meyer, A. and Schartl, M. (1999) Gene and genome duplications in vertebrates: the one-to-four (-to-eight in fish) rule and the evolution of novel gene functions. *Curr Opin Cell Biol*, **11**, 699-704.
- Miklos, G.L. and Rubin, G.M. (1996) The role of the genome project in determining gene function: insights from model organisms. *Cell*, **86**, 521-529.
- Mirkin, B., Muchnik, I. and Smith, T.F. (1995) A biologically consistent model for comparing molecular phylogenies. *J Comput Biol*, **2**, 493-507.
- Mirsky, L. (1982) *An introduction to linear algebra*. Dover Publications, New York.
- Mitchell, P.C. (1901) On the intestinal tract of birds, with remarks on the valuation and nomenclature of zoological characters. *Linnean Soc London (Zool Ser 2)*, **8**, 173-275.
- Miyata, T., Kuma, K., Iwabe, N. and Nikoh, N. (1994) A possible link between molecular evolution and tissue evolution demonstrated by tissue specific genes. *Jpn J Genet*, **69**, 473-480.
- Mombaerts, P. (1999) Seven-transmembrane proteins as odorant and chemosensory receptors. *Science*, **286**, 707-711.
- Moritz, C. and Hillis, D.M. (1996) Molecular Systematics: Context and Controversies. In Mable, B.K. (ed.), *Molecular Systematics*. Sinauer Associates, Sunderland, MA.

- Morris, S.C. (1998) Metazoan phylogenies: falling into place or falling to pieces? A palaeontological perspective. *Curr Opin Genet Dev*, **8**, 662-667.
- Mueller, L.D. and Ayala, F.J. (1982) Estimation and interpretation of genetic distance in empirical studies. *Genet Res*, **40**, 127-137.
- Mueller, T. and Vingron, M. (2000) Modeling amino acid replacement. *J Comput Biol*, **7**, 761-776.
- Muller, H.J. (1935) The origination of chromatin deficiencies as minute deletions subject to insertion elsewhere. *Genetics*, **17**, 237-252.
- Muller, H.J. (1936) Bar duplication. *Science*, **83**, 528-530.
- Nei, M. (1987) *Molecular Evolutionary Genetics*. Columbia University Press, New York.
- Nei, M. (1996) Phylogenetic analysis in molecular evolutionary genetics. *Annu Rev Genet*, **30**, 371-403.
- Nei, M., Chakraborty, R. and Fuerst, P.A. (1976) Infinite allele model with varying mutation rate. *Proc Natl Acad Sci U S A*, **73**, 4164-4168.
- Neyman, J. (1971) Molecular studies of evolution: a source of novel statistical problems. In Yackel, J. (ed.), *Statistical Decision Theory and Related Topics*. Academic, New York, pp. 1-27.
- Nuttall, G.H.F. (1904) *Blood Immunity and Blood Relationship*. Cambridge University Press, Cambridge.
- Ohno, S. (1970) *Evolution by gene duplication*. Springer-Verlag, New York.
- Ohta, T. (1973) Slightly deleterious mutant substitutions in evolution. *Nature*, **246**, 96-98.
- Ohta, T. (1987) Simulating evolution by gene duplication. *Genetics*, **115**, 207-213.
- Ohta, T. (1988a) Evolution by gene duplication and compensatory advantageous mutations. *Genetics*, **120**, 841-847.
- Ohta, T. (1988b) Time for acquiring a new gene by duplication. *Proc Natl Acad Sci U S A*, **85**, 3509-3512.
- Ohta, T. (1989a) Role of gene duplication in evolution. *Genome*, **31**, 304-310.

- Ohta, T. (1989b) Time for spreading of compensatory mutations under gene duplication. *Genetics*, **123**, 579-584.
- Ohta, T. (1991) Multigene families and the evolution of complexity. *J Mol Evol*, **33**, 34-41.
- Ohta, T. (1992a) The meaning of natural selection revisited at the molecular level. *Trends Ecol Evol*, **7**, 311-312.
- Ohta, T. (1992b) The nearly neutral theory of molecular evolution. *Annu Rev Ecol Syst*, **23**, 263-286.
- Ohta, T. (1993) Pattern of nucleotide substitutions in growth hormone-prolactin gene family: a paradigm for evolution by gene duplication. *Genetics*, **134**, 1271-1276.
- Ohta, T. (1994) Further examples of evolution by gene duplication revealed through DNA sequence comparisons. *Genetics*, **138**, 1331-1337.
- Ohta, T. (1996) The current significance and standing of neutral and neutral theories. *Bioessays*, **18**, 673-677; discussion 683.
- Ostermeier, M. and Benkovic, S.J. (2000) Evolution of protein function by domain swapping. *Adv Protein Chem*, **55**, 29-77.
- 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. (1996) TreeView: an application to display phylogenetic trees on personal computers. *Comput Appl Biosci*, **12**, 357-358.
- 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 Rzhetsky, A. (ed.), *Mathematical hierarchies in Biology, DIMACS Series in Discrete Mathematics and Theoretical Computer Science*. American Mathematical Society, Providence, RI, Vol. 37, pp. 57-70.
- Page, R.D.M. and Holmes, E.C. (1998) *Molecular evolution : a phylogenetic approach*. Blackwell Science, Oxford ; Malden, MA.
- 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.

- Patthy, L. (1985) Evolution of the proteases of blood coagulation and fibrinolysis by assembly from modules. *Cell*, **41**, 657-663.
- Patthy, L. (1987) Intron-dependent evolution: preferred types of exons and introns. *FEBS Lett*, **214**, 1-7.
- Patthy, L. (1991) Exons--original building blocks of proteins? *Bioessays*, **13**, 187-192.
- Pearson, K. (1926) On the coefficient of racial likeness. *Biometrika*, **18**, 105-117.
- Pearson, W.R. (1990) Rapid and sensitive sequence comparison with FASTP and FASTA. *Methods Enzymol*, **183**, 63-98.
- Perriere, G. and Gouy, M. (1996) WWW-query: an on-line retrieval system for biological sequence banks. *Biochimie*, **78**, 364-369.
- Piatigorsky, J., O'Brien, W.E., Norman, B.L., Kalumuck, K., Wistow, G.J., Borrás, T., Nickerson, J.M. and Wawrousek, E.F. (1988) Gene sharing by delta-crystallin and argininosuccinate lyase. *Proc Natl Acad Sci U S A*, **85**, 3479-3483.
- Press, W.H., Vetterling, W.T., Teukolsky, S.A. and Flannery, B.P. (1992) *Numerical recipes in C: The art of scientific computing*. Cambridge University Press, Cambridge, MA.
- Rao, C.R. (1952) *Advanced statistical methods in biometric research*. John Wiley and Sons, New York.
- Riley, M. (1993) Functions of the gene products of Escherichia coli. *Microbiol Rev*, **57**, 862-952.
- Rzhetsky, A. and Nei, M. (1992) Statistical properties of the ordinary least-squares, generalized least-squares, and minimum-evolution methods of phylogenetic inference. *J Mol Evol*, **35**, 367-375.
- Rzhetsky, A. and Nei, M. (1993) Theoretical foundation of the minimum-evolution method of phylogenetic inference. *Mol Biol Evol*, **10**, 1073-1095.
- Saitou, N. (1996) Reconstruction of gene trees from sequence data. *Methods Enzymol*, **266**, 427-449.
- Saitou, N. and Nei, M. (1987) The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol Biol Evol*, **4**, 406-425.

- Sanger, F., Air, G.M., Barrell, B.G., Brown, N.L., Coulson, A.R., Fiddes, C.A., Hutchison, C.A., Slocombe, P.M. and Smith, M. (1977) Nucliotide sequence of bacteriophage phi X174 DNA. *Nature*, **265**, 687-695.
- Sankoff, D. (1975) Minimal mutation trees of sequences. *SIAM J Appl Math*, **28**, 35-42.
- Sankoff, D. (2001) Gene and genome duplication. *Curr Opin Genet Dev*, **11**, 681-684.
- Sankoff, D. and Kruskal, J.B. (1983) *Time warps, string edits, and macromolecules : the theory and practice of sequence comparison*. Addison-Wesley Pub. Co. Advanced Book Program, Reading, MA.
- Schieber, B. and Vishkin, U. (1988) On finding lowest common ancestors: simplification and parallelization. *SIAM J Comput*, **17**, 1253-1262.
- Serebrowsky, A.S. (1938) *Curr Rev Acad Sci USSR*, **19**, 77-81.
- Smith, C.W., Patton, J.G. and Nadal-Ginard, B. (1989) Alternative splicing in the control of gene expression. *Annu Rev Genet*, **23**, 527-577.
- Sneath, P.H.A. (1957a) The application of computers in taxonomy. *J Gen Microbiol*, **17**, 201-226.
- Sneath, P.H.A. (1957b) Some thoughts on bacterial classification. *J Gen Microbiol*, **17**, 184-200.
- Sneath, P.H.A. and Sokal, R.R. (1962) Numerical taxonomy. *Nature*, **193**, 855-860.
- Sneath, P.H.A. and Sokal, R.R. (1973) *Numerical taxonomy : the principles and practice of numerical classification*. W. H. Freeman, San Francisco.
- Sobel, E. and Martinez, H.M. (1986) A multiple sequence alignment program. *Nucleic Acids Res*, **14**, 363-374.
- Sokal, R.R. (1956) Quantification of systematic relationships and of phylogenetic trends. *Proc IXth Int Congr Ent*.
- Sokal, R.R. (1961) Distance as a measure of taxonomic similarity. *Syst Zool*, **10**, 70-79.
- Sokal, R.R. and Michener, C.D. (1958) A statistical method for evaluating systematic relationships. *University of Kansas Sci Bull*, **28**, 1409-1438.
- Sokal, R.R. and Sneath, P.H.A. (1963) *Principles of numerical taxonomy*. W. H. Freeman, San Francisco.

- Stephens, S.G. (1951) Possible significance of duplication in evolution. *Adv Genet*, **4**, 247-265.
- Strimmer, K. and von Haeseler, A. (1996) Quartet puzzling: A quartet maximum likelihood method for reconstructing tree topologies. *Mol Biol Evol*, **13**, 964-969.
- Stryer, L. (1995) *Biochemistry*. W. H. Freeman and Company, New York, NY.
- Studier, J.A. and Keppler, K.J. (1988) A note on the neighbor-joining algorithm of Saitou and Nei. *Mol Biol Evol*, **5**, 729-731.
- Swofford, D.L., Olsen, G.J., Waddell, P.J. and Hillis, D.M. (1996) Phylogenetic Inference. In Mable, B.K. (ed.), *Molecular systematics*. Sinauer Associates, Sunderland, MA.
- Szostak, J.W., Orr-Weaver, T.L., Rothstein, R.J. and Stahl, F.W. (1983) The double-strand-break repair model for recombination. *Cell*, **33**, 25-35.
- Tajima, F. (1993) Simple methods for testing the molecular evolutionary clock hypothesis. *Genetics*, **135**, 599-607.
- Tatusov, R.L., Koonin, E.V. and Lipman, D.J. (1997) A genomic perspective on protein families. *Science*, **278**, 631-637.
- 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.
- Thompson, J.D., Gibson, T.J., Plewniak, F., Jeanmougin, F. and Higgins, D.G. (1997) The CLUSTAL_X windows interface: flexible strategies for multiple sequence alignment aided by quality analysis tools. *Nucleic Acids Res*, **25**, 4876-4882.
- Thompson, J.D., Higgins, D.G. and Gibson, T.J. (1994) CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res*, **22**, 4673-4680.
- Thompson, J.D., Plewniak, F. and Poch, O. (1999) A comprehensive comparison of multiple sequence alignment programs. *Nucleic Acids Res*, **27**, 2682-2690.
- Tomb, J.F., White, O., Kerlavage, A.R. and ... (1997) The complete genome sequence of the gastric pathogen *Helicobacter pylori*. *Nature*, **388**, 539-547.
- Troemel, E.R. (1999) Chemosensory signaling in *C. elegans*. *Bioessays*, **21**, 1011-1020.

- van Zonneveld, A.J., Veerman, H., MacDonald, M.E., van Mourik, J.A. and Pannekoek, H. (1986) Structure and function of human tissue-type plasminogen activator (t-PA). *J Cell Biochem*, **32**, 169-178.
- Venter, J.C., Adams, M.D., Myers, E.W., Li, P.W., Mural, R.J., Sutton, G.G., Smith, H.O., Yandell, M., Evans, C.A., Holt, R.A., Gocayne, J.D., Amanatides, P., Ballew, R.M., Huson, D.H., Wortman, J.R., Zhang, Q., Kodira, C.D., Zheng, X.H., Chen, L., Skupski, M., Subramanian, G., Thomas, P.D., Zhang, J., Gabor Miklos, G.L., Nelson, C., Broder, S., Clark, A.G., Nadeau, J., McKusick, V.A., Zinder, N., Levine, A.J., Roberts, R.J., Simon, M., Slayman, C., Hunkapiller, M., Bolanos, R., Delcher, A., Dew, I., Fasulo, D., Flanigan, M., Florea, L., Halpern, A., Hannenhalli, S., Kravitz, S., Levy, S., Mobarry, C., Reinert, K., Remington, K., Abu-Threideh, J., Beasley, E., Biddick, K., Bonazzi, V., Brandon, R., Cargill, M., Chandramouliswaran, I., Charlab, R., Chaturvedi, K., Deng, Z., Di Francesco, V., Dunn, P., Eilbeck, K., Evangelista, C., Gabrielian, A.E., Gan, W., Ge, W., Gong, F., Gu, Z., Guan, P., Heiman, T.J., Higgins, M.E., Ji, R.R., Ke, Z., Ketchum, K.A., Lai, Z., Lei, Y., Li, Z., Li, J., Liang, Y., Lin, X., Lu, F., Merkulov, G.V., Milshina, N., Moore, H.M., Naik, A.K., Narayan, V.A., Neelam, B., Nusskern, D., Rusch, D.B., Salzberg, S., Shao, W., Shue, B., Sun, J., Wang, Z., Wang, A., Wang, X., Wang, J., Wei, M., Wides, R., Xiao, C., Yan, C., Yao, A., Ye, J., Zhan, M., Zhang, W., Zhang, H., Zhao, Q., Zheng, L., Zhong, F., Zhong, W., Zhu, S., Zhao, S., Gilbert, D., Baumhueter, S., Spier, G., Carter, C., Cravchik, A., Woodage, T., Ali, F., An, H., Awe, A., Baldwin, D., Baden, H., Barnstead, M., Barrow, I., Beeson, K., Busam, D., Carver, A., Center, A., Cheng, M.L., Curry, L., Danaher, S., Davenport, L., Desilets, R., Dietz, S., Dodson, K., Doup, L., Ferriera, S., Garg, N., Gluecksmann, A., Hart, B., Haynes, J., Haynes, C., Heiner, C., Hladun, S., Hostin, D., Houck, J., Howland, T., Ibegwam, C., Johnson, J., Kalush, F., Kline, L., Koduru, S., Love, A., Mann, F., May, D., McCawley, S., McIntosh, T., McMullen, I., Moy, M., Moy, L., Murphy, B., Nelson, K., Pfannkoch, C., Pratt, E., Puri, V., Qureshi, H., Reardon, M., Rodriguez, R., Rogers, Y.H., Romblad, D., Ruhfel, B., Scott, R., Sitter, C., Smallwood, M., Stewart, E., Strong, R., Suh, E., Thomas, R., Tint, N.N., Tse, S., Vech, C., Wang, G., Wetter, J., Williams, S., Williams, M., Windsor, S., Winn-Deen, E., Wolfe, K., Zaveri, J., Zaveri, K., Abril, J.F., Guigo, R., Campbell, M.J., Sjolander, K.V., Karlak, B., Kejariwal, A., Mi, H., Lazareva, B., Hatton, T., Narechania, A., Diemer, K., Muruganujan, A., Guo, N., Sato, S., Bafna, V., Istrail, S., Lippert, R., Schwartz, R., Walenz, B., Yooseph, S., Allen, D., Basu, A., Baxendale, J., Blick, L., Caminha, M., Carnes-Stine, J., Caulk, P., Chiang, Y.H., Coyne, M., Dahlke, C., Mays, A., Dombroski, M., Donnelly, M., Ely, D., Esparham, S., Fosler, C., Gire, H., Glanowski, S., Glasser, K., Glodek, A., Gorokhov, M., Graham, K., Gropman, B., Harris, M., Heil, J., Henderson, S., Hoover, J., Jennings, D., Jordan, C., Jordan, J., Kasha, J., Kagan, L., Kraft, C., Levitsky, A., Lewis, M., Liu, X., Lopez, J., Ma, D., Majoros, W., McDaniel, J., Murphy, S., Newman, M., Nguyen, T., Nguyen, N., Nodell, M., Pan, S., Peck, J., Peterson, M., Rowe, W., Sanders, R., Scott, J., Simpson, M., Smith, T., Sprague, A., Stockwell, T., Turner, R., Venter, E., Wang, M., Wen, M., Wu, D., Wu, M., Xia, A., Zandieh, A. and Zhu, X. (2001) The sequence of the human genome. *Science*, **291**, 1304-1351.

Webb, E.C. (1992) *Enzyme nomenclature*. Academic Press, San Diego.

- Whelan, S. and Goldman, N. (2001) A general empirical model of protein evolution derived from multiple protein families using a maximum-likelihood approach. *Mol Biol Evol*, **18**, 691-699.
- Wistow, G. (1993) Lens crystallins: gene recruitment and evolutionary dynamism. *Trends Biochem Sci*, **18**, 301-306.
- Wyrambik, D. and Grisebach, H. (1979) Enzymic synthesis of lignin precursors. Further studies on cinnamyl-alcohol dehydrogenase from soybean-cell-suspension cultures. *Eur. J. Biochem.*, **97**, 503-509.
- Yamada, Y., Avvedimento, V.E., Mudryj, M., Ohkubo, H., Vogeli, G., Irani, M., Pastan, I. and de Crombrughe, B. (1980) The collagen gene: evidence for its evolutionary assembly by amplification of a DNA segment containing an exon of 54 bp. *Cell*, **22**, 887-892.
- Yang, Z. (1993) Maximum-likelihood estimation of phylogeny from DNA sequences when substitution rates differ over sites. *Mol Biol Evol*, **10**, 1396-1401.
- Yang, Z. (1994) Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: approximate methods. *J Mol Evol*, **39**, 306-314.
- Yang, Z. (1995) A space-time process model for the evolution of DNA sequences. *Genetics*, **139**, 993-1005.
- 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. (2001a) ATV: display and manipulation of annotated phylogenetic trees. *Bioinformatics*, **17**, 383-384.
- Zmasek, C.M. and Eddy, S.R. (2001b) A simple algorithm to infer gene duplication and speciation events on a gene tree. *Bioinformatics*, **17**, 821-828.
- Zuckermandl, E. (1987) On the molecular evolutionary clock. *J Mol Evol*, **26**, 34-46.
- Zuckermandl, E., Jones, R.T. and Pauling, L. (1960) A comparison of animal hemoglobins by tryptic peptide pattern analysis. *Biochem*, **46**, 1349-1360.
- Zuckermandl, E. and Pauling, L. (1962) Molecular disease, evolution and genetic heterogeneity. In Pullman, B. (ed.), *Horizons in Biochemistry*. Academic Press, London.
- Zuckermandl, E. and Pauling, L. (1965a) Evolutionary divergence and convergence in proteins. In Vogel, H.J. (ed.), *Evolving Genes and Proteins*. Academic Press, New York.

Zuckerkandl, E. and Pauling, L. (1965b) Molecules as documents of evolutionary history.
J Theor Biol, **8**, 357-366.