Probabilistic Framework for Genome-wide Phylogeny and Ortholog Determination

by

Matthew D. Rasmussen

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2006
[June 2006]

Author_____
Department of Electrical Engineering and Computer Science
May 5, 2006

Certified by_____
Manolis Kellis
Assistant Professor, Distinguished Alumnus (1964) Career Development
Thesis Supervisor

Accepted by_____
Arthur C. Smith
Chairman, Department Committee on Graduate Theses

Probabilistic Framework for Genome-wide Phylogeny and Ortholog Determination

by

Matthew D. Rasmussen

Submitted to the Department of Electrical Engineering and Computer Science

May 2006

In Partial Fulfillment of the Requirements for the Degree of Master of Science in Electrical Engineering and Computer Science

ABSTRACT

Comparative genomics of multiple related species has emerged as a powerful tool for genome signal discovery. To that end, dozens of mammalian, fly, and fungal genomes have been fully sequenced. Making use of these genomes requires rigorous computational methods for determining the evolutionary history of every gene and region. In particular, comparative analysis requires the ability to distinguish between orthologous and paralogous regions. Current approaches to ortholog identification work adequately for pairs of species but are ineffective for multiple complete genomes. This thesis presents a new phylogenetic reconstruction method, SINDIR, that is designed specifically for genome-wide orthology determination. Unlike any other method, SINDIR exploits the known evolutionary history of a set of species to infer the history of their genes. This is done by learning a probabilistic model of evolution from a trusted set of unambiguous orthologs. Given this model, SINDIR can find the maximum likelihood phylogenetic tree for any set of the genes. In a novel technique, synteny maps are used to train and evaluate the evolutionary model on both simulated and real sequence data. SINDIR avoids errors commonly committed by current methods and achieves a significantly improved accuracy of orthology determination.

Thesis Supervisor: Manolis Kellis
Title: Assistant Professor, Distinguished Alumnus (1964) Career Development

Contents

Acknowledgments

To my loving family for their support and guidance: *Amy, Dave, Megan, George, Irene, Vivian, and Dillon*

Thanks to *Manolis* and the whole CompBio lab for their advice and insights.

Go as far as you can see; when you get there, you'll see farther.

— Thomas Carlyle

Go. Kill. Win.

— Basketball Coach

Chapter 1

Introduction

It took nearly a decade of work to sequence the human genome [8]. At the time, it was the first vertebrate to be completely sequenced, it was 80 times larger than any genome ever sequenced before it, and it took the effort of hundreds of researchers around the world to complete. Since then, technology has dramatically improved, enabling the sequencing of four additional mammalian genomes [4, 16, 9, 33], and within a year, that count will be 32.

The challenge for the next generation of computational biologists will be deciphering this treasure trove of genetic information. The human genome consists of just over 3 billion chemical bases denoted by the letters A, C, T and G. These letters come together to form all the instructions needed to construct a fully functioning human being. The most functionally active portions of the genome, genes, consist of just over 1.5% of these bases. Amazingly, only another 3.5% is estimated to have any functional importance [11]. Finding and describing these regions is a major goal of current research, but locating such small signals in a large genome has proved challenging.

In recent years, the most promising approach for genomic signal discovery has been comparative genomics [22, 37]. By using our understanding of evolution, we can learn how different regions within a genome mutate over millions of years, and use these differing patterns of mutation to classify yet unidentified regions. The general approach of comparative genomics is to consider a region of DNA along with its orthologous, or equivalent, regions in several other species. Given these regions, the series of mutations responsible for their differences can be inferred and compared with evolutionary models for various functional and non-functional elements. For example, a gene can be distinguished from non-coding DNA by ensuring its pattern of mutation is consistent with

models of gene evolution, such as a bias for silent mutations or reading frame preserving insertions and deletions.

The power to detect these patterns increases with the number of genomes compared. Therefore, the publishing of more mammalian, fungal, worm, and fly genomes in the coming months and years promises to accelerate our deciphering of the human genome. Accordingly, scalable methods for properly comparing such genomes are necessary and timely.

For my master's project, I have developed a new phylogenetic method, SINDIR, which is designed specifically for accurately identifying orthologous genes in several closely related species. This method provides a solution to a critical required step for any comparative analysis. In the past, fairly simplistic methods have been used for identifying orthologs, and their short-comings are now becoming more apparent with the increase in available sequence data. This thesis has the following structure:

- In chapter 2, we review existing methods and their drawbacks for determining orthology.
- In chapter 3, we introduce phylogenomics as a principled approach to determining orthology.
- In chapter 4, we present our main contribution, SINDIR, which is a new phylogenetic algorithm specifically designed for the phylogenomic setting.
- In chapter 5, we evaluate our algorithm on both real and simulated data sets. We also present a novel technique for evaluating phylogenetic algorithms on real data.
- In chapter 6, we outline several future directions for the SINDIR algorithm.
- In chapter 7, we conclude with a summary of our contributions.

Chapter 2

Existing methods of orthology determination

The need for reliable orthology determination is increasing with the growth of publicly available fully sequenced genomes. Many methods have been developed for orthology determination, but each has drawbacks. Some of these drawbacks are only now being noticed with the availability of large gene sets. In the following sections, we define the different kinds of ancestry genes can share and many methods that have been developed for inferring such ancestries.

2.1 Definitions of gene evolution

In evolution, nearly all genes are related to some extent. The phylogenetic literature has produced several terms that help distinguish the many ways genes can share ancestry. The most general genetic relationship is called *homology*. Two genes are said to be homologous if they both descend from a relatively recent common ancestral gene. When a species population *speciates*, that is diverges into two or more new and distinct species, each new species inherits a copy of the ancestral species's gene set. *Orthologs* are pairs of genes in different genomes that descend from an ancestral gene due to speciation. New genes can also arise by another process, *gene duplication*, where a gene sequence is copied and reinserted elsewhere in the genome. This process leads to two copies of the same gene, called *paralogs*, within the same genome.

Orthology is often the most sought after relationship, because orthologs commonly maintain

similar function after speciation. Thus, by knowing the the function of a gene in one species, orthology can be used to predict the function of genes in other species. However, this form of functional prediction can be complicated if a gene has a paralog. When two copies of a gene exist within a single genome, one gene may diverge more quickly to adapt a new function, while the other is selected to maintain the original function. Another possibility is that both genes specialize different sub-functions [10]. Because of these possibilities, it is important to identify when a gene has a paralog.

Sometimes the term *co-ortholog* is used to describe a gene that has both orthology and paralogy. This term helps signify genes that may have experienced neo-functionalization due to relaxed selection after gene duplication.

2.2 Orthologs by sequence similarity

In past, simplistic methods have been used for determining orthology. However, these methods often break down as more species are considered. The simplest and most widely used method is Best Bidirectional-BLAST Hits (BBH), also known as Best Reciprocal-BLAST Hits. This method is often viewed as conservative by identifying genes as orthologous if and only if they are reciprocally each other's highest scoring match in the BLAST algorithm. However, if BBH is applied between several genomes, the best hits will often disagree, leading to unlikely large sets of orthologous genes. Such difficulties have been observed elsewhere, indicating that the use of BLAST hits alone for inferring common sequence ancestry can be very misleading [23, 24].

Many clustering approaches have been developed to generalize the sequence similarity approach [36, 26, 29]. The main difficulty of these approaches is finding the correct similarity threshold for orthology. Rapidly expanding gene families can have a wide range of mutation rates and thus differing levels of similarity. In the Clusters of Orthologous Genes (COGs) database, for example, many clusters are forced to include entire gene families in order to avoid missing any orthology relationships [36]. Therefore, these clusters are too general for detection of subtle selective pressures at the ortholog level or for protein function assignment. Newer algorithms use more sophisticated techniques to refine gene clusters [25], but without a model of evolution, it is difficult to trust what such clusters represent [24].

2.3 Orthologs by synteny

In addition to sequence similarity, shared genetic ancestry can also be inferred by the location of genes in a genome. Over the course of millions of years, a genome can undergo large chromosomal rearrangements, where chromosomes break up and recombine in new ways. After a sufficient amount of time a genome will completely shuffle the locations of its genes. However, if relatively close species are compared, many of their genes will occur in the same order along a chromosome. Regions of conserved gene order are called *synteny blocks.* Such blocks represent orthologous chromosomal regions between genomes. Thus, the genes that appear within them are also orthologous.

The only exception to this logic is if a gene has a tandem duplication, that is a duplicate gene that appears next to the original. In this case, both genes will appear syntenically aligned and the synteny alone cannot determine if they are orthologs or paralogs. Genes that appear in regions of frequent chromosomal rearrangement will also have ambiguous orthology. Therefore, even though synteny provides a strong indication of orthology, it often cannot explain the orthology of all genes in a genome. The Ensembl genome database uses synteny along with BBH to make orthology calls [20].

2.4 Phylogenetic reconstruction

Phylogenetic algorithms have long been used to determine ancestral relationships and many successful software packages are available for reconstructing phylogenies [31, 30, 13]. Phylogenetic methods attempt to infer the order and rate of divergence of several genetic sequences. This information is often represented as a bifurcating tree, called a *phylogenetic tree*, where the leaves represent modern-day sequences, internal nodes represent ancestral sequences, and branch lengths represent the number of mutations needed to change one sequence into another.

Phylogenetic trees can either be *rooted* or *unrooted.* In an unrooted tree, no attempt is made to determine the relative age of the internal nodes. In a rooted tree, however, a node is picked as the oldest point of the tree. This oldest node, called a root, implies a directionality on every branch of the tree. A tree can also be rooted on a branch, by adding an additional root node to the midpoint of the branch.

The advantage of phylogenetic methods is that gene duplications can be inferred at specific

ACCTTTGGCAATCTGGCTACGC
ACCATTCCCAATCTTGCTACGC

Figure 2.1: The edit-distance between these two gene sequences is 4.

nodes within the binary tree, thereby properly identifying which genes are orthologs or paralogs. They can also handle varying mutation rates and complicated evolutionary events. There are two main approaches to phylogeny reconstruction: distance-based and character-based. The follow section will briefly review these approaches and introduce concepts that are relevant for our algorithm.

2.4.1 Distance-based phylogeny

One way to measure the differences between two sequences is as a distance. A simple notion of distance is *edit-distance*, which is the number of *edits* required to change one string into another. For example, the edit-distance between the two strings in Figure 2.1 is 4.

The same idea can be used to measure the number of possible mutations that have occurred since the divergence of two gene sequences. In addition to changing characters, genes can also have new characters inserted or substrings deleted. These events, although affecting multiple characters are often considered one edit. Another complexity is that mutation can occur twice in the same character, or site. This is called *back mutation* and can cause the number of edits to under predict the number of actual mutations that have occurred. Many models of evolution have been developed to estimate the likely number of mutations from an edit-distance [27].

By combining the distances between several genes, the ancestry of the genes can be inferred. Several of the fastest algorithms for phylogenetic reconstruction use this strategy. Among them are Neighbor Joining (NJ), Unweighted Pair Group Method by Arithmetic mean (UPGMA), and Least Squared Error [34, 31, 14].

These algorithms first calculate a distance matrix M, where $M_{i,j}$ is the evolutionary distance between genes i and j. Then they construct a binary tree with length labeled edges and gene labeled leaves, such that the length of the path between genes i and j is $M_{i,j}$. A distance matrix that can be represented by such a tree is called *additive*. If there is a point on the tree at which all leaves are equidistant, then the distance matrix is also called *ultrametric*. An ultrametric matrix

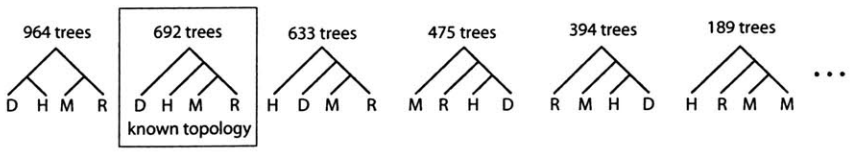


Figure 2.2: Most frequent subtrees of size four in 15,399 Neighbor Joining trees built from mammalian gene clusters (dog, human, mouse, rat). Subtrees were rooted by their connection with the rest of the gene family tree.

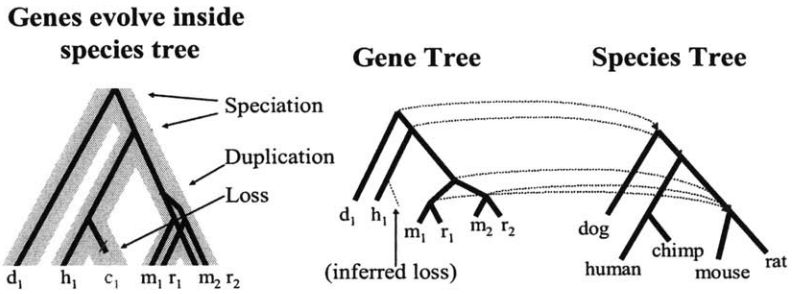


Figure 2.3: Left: example of how a gene tree evolves inside a species tree. Right: gene tree and species tree are drawn separately but a mapping between nodes, known as a reconciliation, indicates how the gene tree should fit inside the species tree.

or tree represents genes with constant mutation rates. In real gene sequences, mutation rates vary frequently. A matrix that is additive indicates the distance estimates are accurate, however, this is also rarely the case. UPGMA guarantees to construct the correct tree if the distances are ultrametric and NJ guarantees to construct the correct tree for an additive matrix. However, neither algorithm has any guarantees for distances that are not exactly ultrametric or additive.

In Figure 2.2 is an example of NJ applied to roughly 15,000 gene families in four mammalian species: dog, human, mouse, and rat. The known topology of the species is boxed in the figure, and it is expected that nearly all gene trees would reproduce this topology. However, due to a common error in NJ, known as long-branch attraction, the most frequent tree in fact has a topology that differs from the known species tree. These trees imply an unlikely number of gene duplications and losses as well as many other misleading conclusions.

2.4.2 Character-based phylogeny

One drawback of distance-based approaches is that they lose information when they summarize gene sequences into pair-wise distances. Other algorithms, such as Maximum Likelihood (ML) [13] and Bayesian inference [30], avoid this summarization by directly operating on the characters of the sequences. Using probabilistic models of evolution, these algorithms can express the uncertainty in estimating mutations and tree topologies.

ML algorithms must search through many possible trees to find the most likely one. Unfortunately, the number of possible trees increases exponentially with the number of genes. Therefore, only a small small fraction of possible trees can be analyzed. For each tree in the search, a model of evolution is applied to each branch and the likelihood of ancestral sequences and mutations are inferred. These likelihoods are used to calculate the likelihood of each proposed tree. The tree that achieves the maximum likelihood is the final answer.

In the Bayesian approach, the tree search is performed with Markov Chain Monte Carlo (MCMC), which ensures that the search samples trees from the posterior distribution. Using these samples, the posterior likelihood of a tree given the sequence data can be calculated. Algorithms, such as MrBayes, can find the Maximum A Posterior (MAP) tree.

Both of these algorithms must perform dramatically more computation than distance methods due to the tree search and because every character of the genes sequences must be considered. In the setting of constructing one gene tree, this additional computation is acceptable. However, for constructing trees of every gene family in dozens of species, current probabilistic formulations are prohibitively slow.

Chapter 3

Phylogenomics

Phylogenetic methods were developed to infer the divergence order of species within a clade or genes within a single gene family, but these methods can also be adapted for the genome-wide orthology problem. In a seminal paper [7], Eisen suggested such an approach and termed it *phylogenomics*. In this framework, genes are first clustered in into general gene families, then their relationships are further refined by building a phylogenetic tree within each cluster. By building phylogenies, orthologs and paralogs can be differentiated in a principled manner.

A phylogenomic approach has the following general structure.

1. annotate gene locations
2. reconstruct species phylogeny
3. cluster gene families
4. align gene families
5. reconstruct phylogenies (the focus of this thesis)
6. reconciling gene trees to species tree
7. generate orthology database

Many of the steps in this approach have suitable solutions. For example, there exist many successful techniques for gene finding and we can assume that such annotations will be available for our problem. As for the species tree, many of the techniques in Section 2.4 can successfully reconstruct a species tree from fully sequenced genomes. To cluster genes into families, any of the methods given in Section 2.2 are suitable. Also, multiple alignment programs now exist for aligning

hundreds of sequences [6]. The final steps of this pipeline, reconciliation and reading orthology, have also been developed [39] and we explain them in Section 3.1.

However, the most difficult step is the reconstruction of phylogenies. There are a few recent approaches to this problem, which we review in Section 3.2, but they face several difficult challenges. Currently, there is no accurate way to systematically reconstruct the ancestry of thousands of gene families with high accuracy. In this thesis, we present a new phylogenetic algorithm that is specifically designed for the phylogenomic problem. It addresses the challenges faced by existing techniques and has been shown to improve reconstruction accuracy by a large margin.

3.1 Reconciling gene trees to species trees

By comparing a gene tree to the species tree, we can locate gene duplication and loss events. This comparison, called a *reconciliation*, is formalized as finding a mapping from gene nodes to species nodes (See Figure 2.3). The mapping indicates to which species each gene belongs. For modern-day genes, the species is known, but for ancestral genes there is a choice for its species. The most common method of reconciliation is a parsimonious one: find the mapping of genes to species such that the fewest number of duplications and/or losses are inferred. Efficient algorithms have been developed to find such reconciliations [39]. In this section, we present how to infer duplications and losses from a reconciliation, an algorithm for finding an optimal reconciliation, and additional uses of reconciliation.

3.1.1 Speciation, duplication, and loss

In the gene trees that we consider, there are only two ways to create a new gene: speciation and duplication. Therefore, each internal node of a gene tree represents either a speciation or duplication event. A speciation event represents a species population A segregating into two new species B and C. Let a be a gene in species A. During speciation, a will bifurcate and have two children b and c which will be present in species B and C, respectively. The correct reconciliation should map gene a to species A, gene b to species B, and gene c to species C.

A gene duplication event occurs sometime between speciations, that is somewhere along the species branch. Since the reconciliation maps gene nodes to species nodes, we must approximate to which species the duplicating gene belongs. In most formulations, if the exact species of a gene is

```
FindGeneLoss(G, M):
    losses = 0

    foreach node ∈ inOrderTranversal(G):
        visited_species = {}
        internal_species = {}

        foreach child ∈ children(node):
            ptr = M[child]

            while ptr ≠ M[node]:
                visited_species = visited_species ∪ {ptr}
                ptr = parent(ptr)
                internal_species = internal_species ∪ {ptr}

        foreach species ∈ internal_species:
            if species ∉ visited_species:
                losses = losses + 1

    return losses
```

Figure 3.1: Pseudo-code of finding gene loss in a gene tree G with reconciliation M.

not present in a species tree, then the gene is mapped to the most recent possible species. Therefore, if two mice genes have a common parent that represents a duplication event, the parent is mapped to the mouse species, even though it actually belongs to a slightly more ancient mouse-like species.

Given the definitions above, an internal node can be labeled duplication or speciation according to the following rule: *a gene node is a duplication if and only if it is mapped to the same species as one of its children, otherwise it is a speciation.*

We can infer gene losses using a parsimonious rule. We are guaranteed a loss has occurred below g_i whenever a descendent species of $M[g_i]$ lacks a mapping within the subtree rooted at g_i. We can discover all such situations of loss in time linear to the gene tree, using the algorithm in Figure 3.1.

3.1.2 Reconciliation by minimum duplication

In [39], the following equation is given for finding a reconciliation that minimizes the number of inferred duplications. Let M be a reconciliation and $g_1, ..., g_n$ be the genes in a gene tree (both

modern and ancestral).

$$\begin{cases} M[g_i] = species(g_i) & \text{if } g_i \text{ is a modern gene} \\ M[g_i] = LCA(\{M[g_j] \mid g_j \in children(g_i)\}) & \text{otherwise} \end{cases} \tag{3.1}$$

Where $LCA(X)$ stands for the Least Common Ancestor of the set of nodes X, $children(g_i)$ gives the children of a node, and $species(g_i)$ gives the modern species for a modern gene. The first condition states that the reconciliation is initialized to map the leaves of the gene tree to proper leaves of the species tree. This can be done because we know from which species each gene was sequenced. The second condition can then be applied for g_i, once the mapping of the children of g_i is determined. Thus, a reconciliation can be solved in almost linear time using a post-order traversal of the gene tree. The run time is "almost linear" because of time needed to compute the LCA of a set of nodes. In worst case, the LCA of a set of nodes takes time linear to the number of species, if the species tree is extremely unbalanced. However, for the size of most trees that we consider, this time is almost constant.

3.1.3 Rooting by reconciliation

The most popular ways to root a tree are *outgroup rooting* and *midpoint rooting*. Outgroup rooting requires knowing at least one gene to be an outgroup of the rest, but this will not be known for every gene family that we consider. Midpoint rooting only works if we assume a molecular clock, however, this assumption rarely holds. Therefore, we need a different method of rooting a gene tree that can be done for any gene family. In this thesis, we use a method called *rooting by reconciliation*, that roots a tree such that the number of duplications and/or losses is minimized. An algorithm for doing this in almost linear time is given in [39]. We briefly outline it here as well.

The general idea is that we want to try rooting the tree on each branch, count the number of events, and keep the rooting that achieved the minimum number of events. This can be done in almost linear time by remembering the reconciliation of subtrees that reappear in different rootings. If we attempt rooting branches in an in-order traversal, then only the reconciliation of the node incident to both the old and new rooting branch can change. The reconciliation of the new implied root node will also need to be done. Therefore, only two LCAs must be done for each rooting attempt. By only computing the number of events gained or lost with each rooting, we can quickly

compute the total number of events for each rooting. This leads to an almost linear runtime algorithm for rooting by reconciliation.

3.1.4 Reading orthology and paralogy

Lastly, we ultimately want to read orthology and paralogy relationships from a gene tree. This can be done once the internal nodes are labeled as duplication or speciation using a reconciliation and the rule given in Section 3.1.1. Recall that two genes are orthologs if and only if their LCA is a speciation event. We can find all orthologs using the following algorithm.

For each node n labeled as a speciation in the gene tree, find all the leaves L_1 and L_2 of the subtrees rooted at the two children of n. Every pair of genes $l_1 \in L_1$ and $l_2 \in L_2$ are orthologs. All other gene pairs of the same genome are technically paralogs, but usually it is most useful to restrict the paralogy definition to only recent duplications. To find paralogs, a similar algorithm can be used on duplication nodes that are recent enough.

3.2 Challenges of existing phylogenomic methods

Currently, the main challenge in phylogenomics is automating phylogenetic reconstruction for thousands of trees. Phylogenies are often sensitive to the particular construction method and the most accurate methods are computationally expensive [32]. The most frustrating challenge is reading orthology from gene trees. By definition, two genes are orthologs, if and only if their divergence is due to speciation. However, the most common mistake of phylogenetic methods is to create a tree with a gene divergence order that differs slightly from the known species order [24]. If the tree is to be trusted, then the only logical interpretation is to infer that gene duplications and losses are responsible for the disagreement. Therefore, many genes that are obvious orthologs by other measures (high sequence similarity, syntenic in genome alignment) are instead inferred as unrelated.

Several phylogenomic approaches have been developed and each one has tackled these issues differently. One idea has been to use bootstrapping of Neighbor Joining (NJ) trees in order to estimate the ambiguity of not only the tree topology but also the inferred orthologs [38, 35]. The accuracy of such approaches have been tough to measure without a real data set on which to evaluate. However, in our experience NJ is greatly affected by poor distance estimates and suffers frequently from long branch attraction (Figure 2.2). In the case of long branch attraction, topology

errors are committed consistently and thus will not be avoided with use of bootstrapping.

Another approach has been to use a vaguer notion of reconciliation, such that a gene is not reconciled to a species but a set of close species and orthologies are given priority over duplications in parts of the tree where reconciliation is too vague [35]. This reduces the number of spurious duplications inferred, but the approach loses any ability to distinguish orthologs and paralogs between close species, such as the mammals.

The TreeFam database [24], which is a recently published public database of animal phylogenies and orthologies has taken the approach of manual curation. Since there currently exists no automatic method for reliably reconstructing gene phylogenies with sensible species reconciliations, they instead use human annotators to visually inspect gene phylogenies along with additional information such as functional annotations and synteny.

The cause of these reconstruction difficulties is that nearly all phylogenetic methods ignore the species tree during gene tree construction, and therefore do not account for the likelihood of duplications and losses they infer. Only recently, have evolutionary models been proposed to include the species tree [1], but these models are not yet computationally practical for application genome-wide.

In this thesis, I focus on a new method of phylogenetic reconstruction that is specifically designed for the phylogenomics problem. It uses the species tree during gene tree reconstruction in order to avoid unlikely trees. It also mimics the intuition of human annotators who have a general notion of what a phylogenetic tree "should" look like for a set of species. When the annotator sees a gene topology that doesn't match the species topology and notices that a branch is too long or short they will infer that the tree is likely to be wrong. This kind of observation is formulated mathematically in our algorithm as a probabilistic model. By training the algorithm on a data set of trusted phylogenies, our algorithm learns what a correct phylogeny "should" look like. The scope of this thesis is simply phylogenetic reconstruction. However, in the future, this algorithm can be adapted for a full phylogenomic system, including gene family clustering and generation of an orthology database.

Chapter 4

SINDIR methods

I have developed and prototyped a new phylogenetic reconstruction method, SINDIR (Species INformed DIstance-based Reconstruction), that is specifically designed for determining orthologs across multiple complete genomes. SINDIR contains a novel training procedure that learns a model of evolution for true orthologous gene sequences. Using this model, SINDIR finds the maximum likelihood phylogeny for any gene family. Once a phylogeny is obtained, orthologous and paralogous genes can be readily identified.

SINDIR will ultimately be used within a full phylogenomic framework. In such a framework, gene annotations are first clustered into gene families by information such as sequence similarity and conserved synteny. These gene families are simply sets that are large enough such that for every gene that is a member, its ortholog is also a member. In addition, these sets must also be small enough for application of phylogenetic methods. By using strong signals of orthology, such as synteny, a small subset of unambiguous orthologous genes can be identified. These genes are then used as a training set for SINDIR's evolutionary model. Once a model is learned, the other more ambiguous gene sets are then evaluated one at a time by SINDIR. For each gene set, SINDIR produces a maximum likelihood tree, from which orthologs and paralogs can be inferred.

4.1 Key features

Many algorithms have been developed for reconstructing phylogenies, however, the phylogenomics problem presents unique challenges. For example, a clade of mammalian genomes will have roughly

tens of thousands of gene families, for which phylogenetic reconstruction must be done. Therefore, in order to be practical, SINDIR must have a computationally efficient runtime. This is achieved by operating only on pair-wise sequence distances. As opposed to character-based methods, which must apply their models of evolution to every column in a sequence alignment, a distance-based method simplifies its calculations during reconstruction by first summarizing the alignment into a distance matrix as described in Section 2.4.1. However, SINDIR is not a traditional distance-based method. Methods such as NJ, UPGMA, and LSE construct trees that optimize a particular cost, such as distance distortion. Unfortunately, these costs are difficult to integrate with additional information about the species evolution. Therefore, instead of a cost function, SINDIR uses a novel probabilistic formulation of distances which provides a principled framework for integrating addition information from species evolution.

A second unique challenge of phylogenomics is the reconciliation of gene trees with species. As illustrated in Figure 2.2, gene trees often disagree with species trees. This is a problem that was commonly faced by previous attempts at phylogenomic reconstruction (Section 3.2). The cause of these disagreements is that there is limited information in gene sequences. When reconstructing species phylogenies, this limitation can be overcome by concatenating multiple gene sequences together, thereby increasing the number of phylogenetically informative characters. However, when the phylogeny of the genes themselves is desired, concatenation is not an option. Although existing phylogenomic algorithms assume a species tree is known, they do not use it in constructing their gene trees. It is only until after a gene tree is constructed, that it is compared to the species tree.

This, unfortunately, is too late. Often there are multiple gene trees that can explain the divergence of a set of sequences. By ordinary models of sequence evolution, these trees can have very similar likelihoods, implying that any one of them is a suitable answer. However, with the help of a species tree, we can locate rare evolutionary events, such as gene duplication and loss. When the probability of such events is incorporated, the number of likely gene trees is dramatically reduced. SINDIR exploits this fact, by using the species during the reconstruction of a gene tree. Not only are the probabilities of rare events considered, but SINDIR also incorporates the expected mutation rates for each branch of the tree. These expected mutation rates are represented as distributions, which are learned in SINDIR's training phase.

These features result in an phylogenomic algorithm that is uniquely suited for the phylogenomics

problem. As shown in Section 5, SINDIR is able to reconstruct gene trees more accurately than any other current method of phylogenetic reconstruction we tested. In the following sections, an outline of the algorithm is given (Section 4.2), followed by details regarding the model training (Section 4.3), likelihood calculation (Section 4.4), and tree search (Section 4.5).

4.2 Algorithm overview

An outline of the SINDIR algorithm is illustrated in Figure 4.1. The algorithm has two main phases: a training phase and a reconstruction phase. The training phase is given as input a set of gene trees built on unambiguous orthologs and a tree topology for the species. In our training of SINDIR presented in Section 5, syntenic genes with with no trace of duplication were discovered and maximum likelihood genes trees matching the species tree were constructed using the PHYLIP DNAML and PROML programs. After the training phase, a model is created that will be used to evaluate possible gene trees in the reconstruction phase.

The second phase, reconstruction, begins by accepting as input a distance matrix for a set of genes with ambiguous orthology. This distance matrix can be derived from a sequence alignment using any number of publicly available programs. In our analysis, we used the PHYLIP DNADIST and PROTDIST programs for genetic distance estimation. From this distance matrix, an initial proposed gene tree is constructed using the Neighbor Joining (NJ) algorithm. The reconstruct phase, then proceeds in a search loop, where the gene tree is slightly altered to produce new gene trees, and the likelihood of each gene tree is calculated according to the learned model. Each proposed gene tree is labeled with branch lengths that closely approximate the original distance matrix. After a sufficient search, the gene tree that achieved the maximum likelihood is outputted.

The basic assumption of the SINDIR algorithm is that the model of evolution for ambiguous and unambiguous orthologous genes is the same. It is this assumption that allows the information learned from the unambiguous genes to inform the reconstruction of ambiguous gene sets.

4.3 Evolutionary model and training

SINDIR's evolutionary model has been motivated by observations of trusted orthologous genes from four complete mammalian genomes: human, mouse, rat, and dog. These genomes will serve

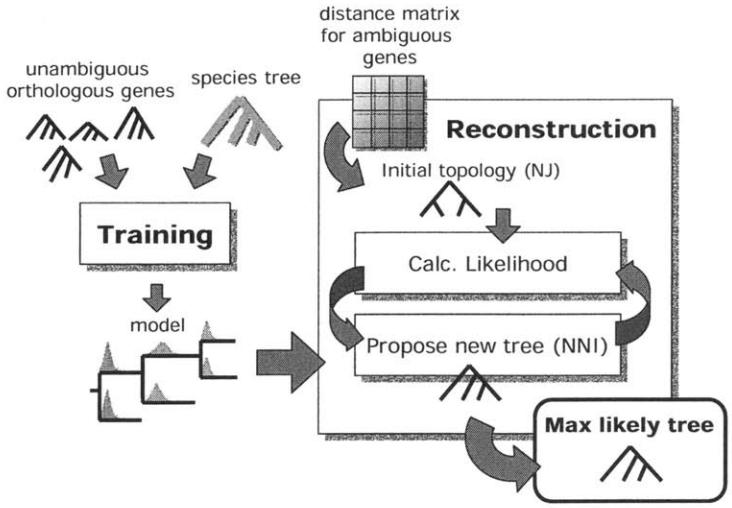


Figure 4.1: Outline of SINDIR algorithm.

as an example for this section. The fly and fungal genomes also have similar statistical properties and the observations described here apply to them as well.

In the training phase, SINDIR estimates a distribution for the branch lengths of trusted gene trees. A key assumption of the algorithm is that sequences on different branches evolve independently. This is a common assumption among all probabilistic phylogeny methods. However, in gene trees reconstructed from syntenic one-to-one orthologs, branch lengths are not independent and are in fact strongly correlated (Figure 4.2). This correlation is strongest between branch lengths and *total tree length*, the sum of all branch lengths in a tree. The distribution of total tree lengths strongly fits a gamma distribution (Figure 4.3).

Faced with this fact, we designed SINDIR to estimate the distribution of *relative branch lengths*. A relative branch length is found by dividing the original branch length, or rather *absolute branch length*, by the total tree length. This dramatically reduces the dependency of the branch lengths (Figure 4.2) and allows SINDIR to exploit the assumption of independence between branches.

4.3.1 Model of evolution

These observations lead to the following model. The model assumes that all genes in the set descend from a single gene placed at the root of the species tree. The genes in a gene tree have

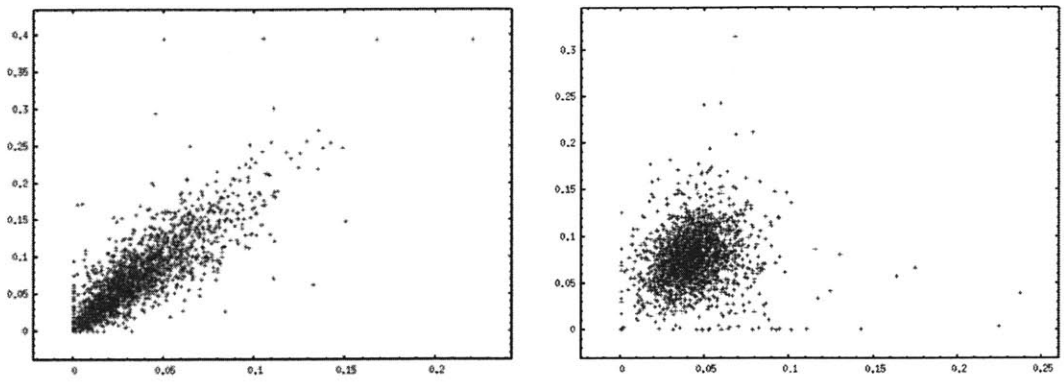


Figure 4.2: Left: correlation of absolute lengths of two mammalian branches. Right: correlation of relative lengths of two mammalian branches.

a common base mutation rate, b, sampled from a gamma distribution (Figure 4.3). As genes descend down the species tree, their mutation rates vary from the base rate by a factor f_i, which is sampled independently for each species branch i from a normal distribution F_i. The parameters of the normal distribution F_i are specific to each branch i of the species tree (Figure 4.4). Gene duplications are events that occur at midpoints along a species branch. Their distribution along the species branch is assumed to be uniform and do not influence the sampling of f_i.

4.3.2 Training

To train this model, we build a set of trusted gene trees. These can be found, for example, by building maximum likelihood trees from sets of genes found aligned in genomic synteny, since synteny is a strong indication of orthology. For each trusted tree, we estimate its base rate as the total tree length. By dividing branch lengths by the base rate, we derive the rate factors f_i. The distribution of rate factors from all trusted trees follows a normal distribution, whose parameters can be determined by any standard fitting procedure. SINDIR finds the maximum likelihood estimation of the mean, μ_i, and standard deviation, σ_i, parameters. An example of these parameters for the mammals is given in Figure 4.5.

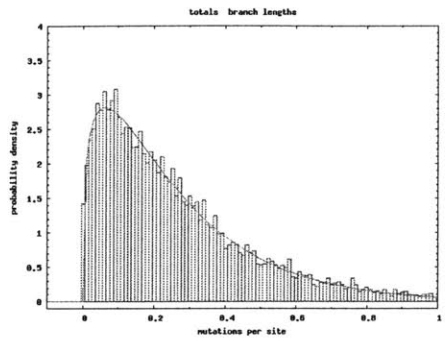


Figure 4.3: Distribution of total tree length in 1800 mammalian gene families. Fitted gamma parameters: $\alpha = 1.311, \beta = 4.949$

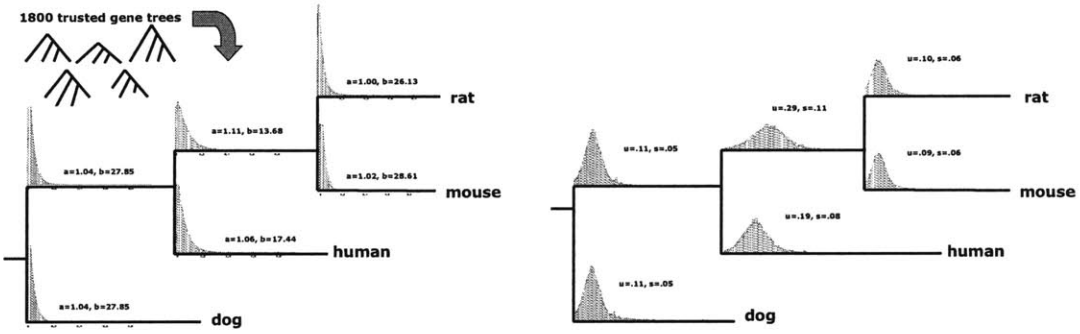
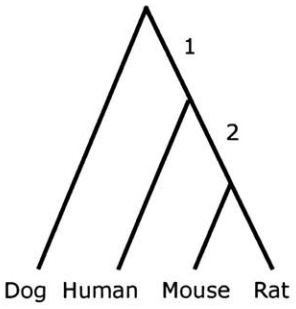


Figure 4.4: Absolute (left) and relative (right) branch lengths distributions found from 1800 trusted gene trees built on gene aligned in synteny, a strong indication of orthology. The branch lengths distributions were calculated for each species branch.



	gamma		normal	
branch	α	β	μ	σ
1	0.881	21.002	0.107	0.053
2	1.103	3.483	0.314	0.107
dog	0.881	21.002	0.107	0.053
human	0.988	14.595	0.171	0.083
mouse	0.753	15.810	0.078	0.063
rat	0.726	15.457	0.084	0.067

Figure 4.5: Fitted parameters on each branch of the mammalian species tree. The gamma is fitted to the absolute length distribution, while the normal distribution is fitted to the relative branch length distributions.

4.4 Estimating gene tree likelihood

Within the reconstruction phase, SINDIR must calculate the likelihood of thousands of trees. Each of these trees are proposed by a tree search algorithm described in Section 4.5. Given a proposed gene tree, SINDIR first estimates its base rate b, which can be used to derive the tree's rate factors f_i. The rate factors are then used to calculate the likelihood that the evolutionary model would produce such a gene tree. As the tree search procedure proposes different tree topologies, the estimated base rate will roughly remain the same but the inferred rate factors will vary greatly. Thus, in practice, SINDIR finds only a few proposed topologies with likely branch lengths. The gene tree that achieves the maximum likelihood is kept as the final answer.

4.4.1 Base rate estimation

SINDIR uses the maximum likelihood estimate (MLE) of the base rate $\hat{b}$ for each proposed tree. This is currently done as an approximation. Ideally, the gene tree likelihood should be integrated over all possible base rates. The MLE can be found using the known prior distribution of base rates $gamma(b|\alpha,\beta)$ and the distribution of rate factors $normal(f_i|\mu_i,\sigma_i^2)$ for each branch i of the tree. Using this information, $\hat{b}$ can be found by solving the following cubic equation.

$$\hat{b}^3 - \left(\frac{\alpha-1}{\beta}\right)\hat{b}^2 + \left(\frac{1}{\beta}\sum_i \frac{\mu_i x_i}{\sigma_i^2}\right)\hat{b} - \left(\frac{1}{\beta}\sum_i \frac{x_i^2}{\sigma_i^2}\right) = 0 \tag{4.1}$$

Where x_i is the absolute branch lengths in the proposed gene tree. The derivation of this equation is given in Section A.1. A convenient benefit of this equation is that one can find the MLE base rate even for gene trees with gene loss and duplication events. That is, the summations in the third and fourth terms normalize the base rate for branches that do not appear at all (due to loss) or appear multiple times (due to duplication). With a base rate estimate, rate factors f_i can be calculated by dividing each x_i by $\hat{b}$.

4.4.2 Reconciliation

SINDIR's evolutionary model contains relative branch length distributions for each species branch. SINDIR calculates the likelihood of seeing a particular gene branch length by reconciling the gene branch to a species branch and assuming the gene branch was sampled from the species branch's

distribution. Thus, the likelihood calculation decomposes into two cases: (1) a simple case, where no duplications or losses are needed, and (2) a complex case, where duplications or losses are required. We first address the simple case and then show how the complex case can be reduced to the simple one.

Case 1: Simple reconciliation

In simple gene trees, the reconciliation maps each gene branch to a unique species branch. Using the assumption that genes on different branches evolve independently, we can factor the gene tree likelihood into the likelihoods of each gene branch. Since we have a density estimation of relative gene branches found within each species branch, the likelihood of a gene branch with relative length f_i in species branch i is simply $normal(f_i|\mu_i, \sigma_i^2)$, where μ_i and σ_i are parameters learned during the training phase. Therefore, the likelihood of a simple gene tree G given a species tree S with parameters μ_i and σ_i is:

$$P(G|S) = \prod_{i \in branches(G)} normal(f_i|\mu_i, \sigma_i^2) \tag{4.2}$$

Case 2: Complex reconciliation

When topologies differ, several reconciliations between a gene and species tree may be possible. In SINDIR, we find the reconciliation that implies the fewest gene duplications and losses, using a method given in [39]. An example of such a reconciliation is depicted in Figure 2.3. Most reconciliation algorithms define a reconciliation as a mapping from the nodes of a gene tree to the nodes of a species tree. For speciation events this mapping is sufficient, because speciations in gene trees coincide with speciations in species trees. However, for gene duplications, we would like to know where along the species branch the duplication occurred. This requires a gene node to map to the middle of a species branch. To capture these kinds of mappings, SINDIR views a reconciliation as a mapping of gene branches to species branches. Unlike the simple case, where every gene branch maps to a single species branch, we must in general handle cases where (2.a) one gene branch must map onto multiple species branches and where (2.b) multiple gene branches map onto one species branch. These two cases are shown in Figure 4.6.

We can reduce case 2.a to simple case 1 by merging two or more species branches into one

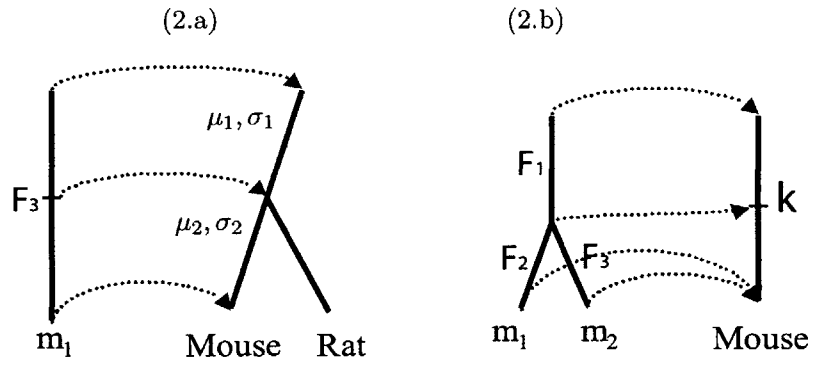


Figure 4.6: Example of two types of complex reconciliation. In case (2.a) multiple gene branches must reconcile to a single species branch. In case (2.b) a single gene branch must reconcile to multiple species branches.

species branch, thus recreating a one-to-one mapping of gene and species branches. Let F_3 be the random variable for the length of a gene branch mapping to multiple species branches. From our model, F_3 must be the result of adding the lengths of two smaller gene branches, F_3' and F_3'', where branch $3''$ is the only child of $3'$. The branching point between these branches is not known, because the second child of $3'$ has been lost. However, we can still construct the distribution for F_3 knowing that it is the sum of two smaller branches sampled from normals.

$$\begin{aligned} F_3' &\sim normal(\mu_1, \sigma_1^2) \\ F_3'' &\sim normal(\mu_2, \sigma_2^2) \\ F_3 &= F_3' + F_3'' \\ &\sim normal(\mu_1, \sigma_1^2) + normal(\mu_2, \sigma_2^2) \\ &= normal(\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2) \end{aligned}$$

Using this merged branch distribution we can now calculate the likelihood of f_3 using the normal distribution, just as we did in case 1. Notice that this strategy also extends to merging more than two species branches.

The reduction for case 2.b requires slightly more work. In case 2.b, we have multiple gene branches reconciling to a single species branch. We cannot merge gene branches, as we did with species branches in 2.a, because such merges would produce gene branches that share segments, and thus have lengths that are not independent (one of SINDIR's assumptions). Instead, SINDIR

breaks up species branches by adding midpoints that split a branch into multiple branches and produces a mapping where each gene branch maps to one, perhaps partial, species branch. Although this gives us a one-to-one mapping of branches, to complete the reduction, we must construct new distributions for the newly divided species branches. Let f_1 and f_2 be relative branch lengths depicted in case 2.b of Figure 4.6, that are sampled from random variables F_1 and F_2 respectively. Let μ and σ be the mean and standard deviation of the original species branch. Since in SINDIR's model we consider the branch length $f_1 + f_2$ to be sampled from a normal distribution, we would like the following equation to hold.

$$F_1 + F_2 \sim normal(\mu, \sigma^2)$$

By choosing a midpoint k along the species branch we can separate the distribution into two distributions from which F_1 and F_2 can be drawn independently. The following distributions for F_1 and F_2 satisfy our requirements.

$$\begin{aligned} F_1 &\sim normal(k\mu, k\sigma^2) \\ F_2 &\sim normal((1-k)\mu, (1-k)\sigma^2) \\ normal(k\mu, k\sigma^2) + normal((1-k)\mu, (1-k)\sigma^2) &= normal(\mu, \sigma^2) \end{aligned}$$

A general equation for gene branch that reconciles to a partial species branch is:

$$P(F_i = f_i | p_i) = normal(p_i\mu, p_i\sigma^2) \tag{4.3}$$

where p_i is the fraction of the species branch reconciled with gene branch i. To attain a likelihood for all three gene branches, we condition on the value of k and multiple by its prior probability. Since we do not know k we must integrate over all possible values of k between zero and one.

$$\begin{aligned} P(F_1, F_2, F_3|S) &= \int_0^1 P(F_1, F_2, F_3|S, k)P(k)dk \\ &= \int_0^1 P(F_1|S, k)P(F_2|S, k)P(F_3|S, k)P(k)dk \end{aligned}$$

Notice, that once k is given, the likelihood of $P(F_1, F_2, F_3|S)$ can be factored. SINDIR currently

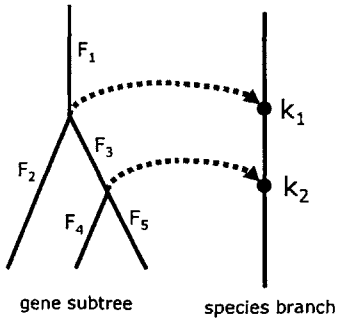


Figure 4.7: An example of multiple midpoints needed on one species branch

assumes that all positions of the midpoint along a species branch are equally likely, therefore the prior for k is uniform and $P(k) = 1$. The integral can be computed using numerical integration. Currently, SINDIR uses Gaussian quadrature as implemented in the numerical python package, SciPy.

More complex cases of 2.b are possible that require multiple midpoints k_i to be chosen along a single species branch. This occurs whenever a subtree G' of the gene tree G is reconciled to one species branch S' and G' contains more than one duplication node. Notice, each duplication node i has a one-to-one reconciliation with a midpoint k_i in S'. The midpoints k_i must be partially ordered, such that if j is a descendent of i then $k_j > k_i$. With this restriction we use the same factoring strategy as done in the single midpoint case.

Figure 4.7 illustrates a complex example of 2.b that has two duplications reconciled to the same species branch. For this example, we have the following likelihood:

$$\begin{aligned} P(F_1, F_2, F_3, F_4, F_5|S) = & \int_0^1 \int_{k_1}^1 P(F_1, F_2, F_3, F_4, F_5|S, k_1, k_2) P(k_1, k_2) dk_2 dk_1 \\ = & \int_0^1 \int_{k_1}^1 P(F_1|S, k_1) P(F_2|S, k_1) P(F_3|S, k_1, k_2) P(F_4|S, k_2) P(F_5|S, k_2) \\ & P(k_2|k_1) P(k_1) dk_2 dk_1 \end{aligned}$$

The species branch fraction p_i can be calculated for each gene branch i by simply taking the difference of the two surrounding midpoints for each gene branch. This allows us to calculate the likelihood of each gene branch using (4.3). In this example, the species branch fractions are:

$$p_1 = k1,\ p_2 = 1 - k1,\ p_3 = k2 - k1,\ p_4 = 1 - k2,\ p_5 = 1 - k2$$

Using SINDIR's assumptions of uniform priors for the location of duplications, both $P(k_1)$ and $P(k_2|k_1)$ are uniform. However, $P(k_2|k_1)$ is uniform over the range $(k_1, 1)$. In SINDIR, the total likelihood of the subtree G' is then calculated using double numerical integration. For cases where more midpoints are necessary, deeper nested integration is used.

4.4.3 Bringing it all together

Using the algorithms outlined in cases 1, 2.a, and 2.b, SINDIR can evaluate the likelihood of any proposed gene tree. First gene branches that reconcile to multiple species branches are handled by case 2.a. After this step, the gene tree can be decomposed into subtrees that reconcile to exactly one species branch. The likelihood of each of these subtrees can be computed using algorithms from either case 1 or case 2.b. Since each branch evolves independently, these subtree likelihoods can all be multiplied together to attain the likelihood of the entire gene tree.

4.4.4 Corner cases

SINDIR does not assume that the given set of genes are all orthologs with each other. Instead, it allows the possibility that the gene set represents a larger gene family. If this is the case, then the correct gene tree will have duplication nodes that proceed all speciation nodes. These duplication nodes will reconcile "above" the root of the species tree. Consequently, there will also be branches that reconcile above the species tree root. This posses a problem, because the learned evolutionary model does not have any distributions for branch lengths above the species tree root. To handle these kinds of branches, SINDIR assumes that any mutation rates before the species tree root are equally likely. Therefore, when comparing proposed gene trees the likelihood calculation of branches that reconcile above the species tree root can be ignored altogether. We call such branches *free branches*, because the branches get to grow for free (no cost). All other branches are called *non-free.*

Free branches can also occur in reconciliation case 2.a, where a gene branch not only reconciles above the species tree root but also below. We call such branches *partially free.* SINDIR splits partially free branches, such that one portion is completely free and the other reconciles completely below the species tree root. The midpoint is chosen such that the likelihood of the resulting branch is maximized. This is simply calculated by taking the minimum of the gene branch length and the mean of the branch distribution.

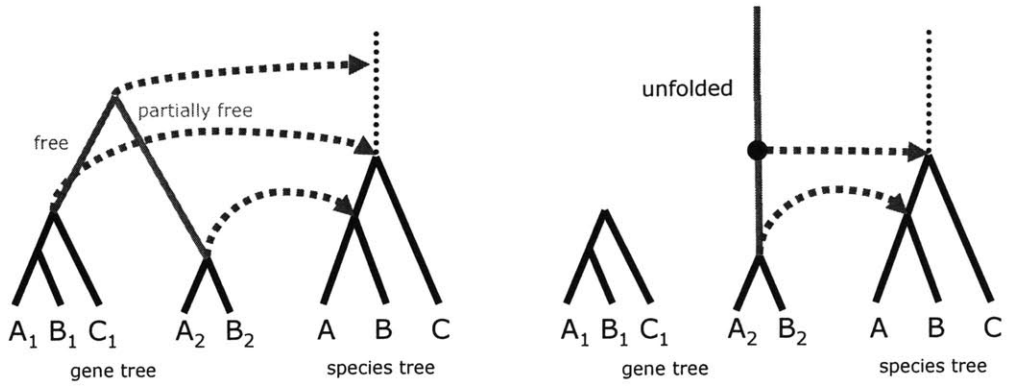


Figure 4.8: Left: example of free and partially free branches. Right: example of branch unfolding.

Partially free branches usually can be handled separately from all other branches, making their likelihood calculation easy to implement. There is only one case where this is not true. When rooting a tree, we know what branch to place the root, but not where along the branch to place it. SINDIR chooses to consistently place the root on the midpoint of the rooting branch in order to remove this ambiguity. One affect is that the distributions of the top two gene branches are always the same. This works out fine as long as both branches are either free or non-free. However, if only one branch is non-free, then positioning the root at the center of the rooting branch may be distorting how much branch length is non-free. We handle this case by *unfolding* the rooting branch Figure 4.8. Since only one child of the root needs the branch for calculating likelihood (the other child is free), we can extend the non-free child branch by a factor of two and then classify it as partially free. Now, the branch can be split such that its likelihood is maximized.

4.4.5 Evolutionary events: duplication and loss

The preceding sections, calculate the likelihood of the branch lengths of a gene tree. In addition, SINDIR also calculates the likelihood of the events occurring at each node in a gene tree. Every internal node of a gene tree is either a speciation or duplication. Currently, SINDIR implements a very simple notion of likelihood for such events. For each internal node, the likelihood that it is a duplication is d and a speciation is $1 - d$. It is also assumed that duplications occur independently. The probability d is a parameter to SINDIR that can be set by training on examples of gene

duplication.

To estimate the likelihood of loss, SINDIR first infers the minimum number of gene losses that could have occurred, using the algorithm in Section 3.1. Losses are assumed to occur independently. The likelihood of seeing a loss is a parameter l.

Duplications and losses are assumed to occur independently of mutation rates and therefore, these likelihoods can be multiplied together to attain the likelihood of the entire gene tree.

4.5 Tree search

The number of possible rooted trees grows exponentially with the number of leaves (Section A.2). Therefore, calculating the likelihood of every possible tree topology is impractical for even reasonably sized trees. Consequently, we can only propose a small fraction of the possible trees. This is done with a search through the space of possible trees. Several algorithms for tree topology search have developed [5, 27]

A good sampling of the tree space is important for finding the ML tree. If not enough space is explored, the ML tree may not be found. On the other hand, excessive tree searching is time-consuming. We implemented two types of tree search in order to experiment with these extremes: (1) a greedy tree search as used in the PHYLIP programs and (2) a Markov Chain Monte Carlo (MCMC) search. Both of these searches require two sub-procedures: Nearest Neighbor Interchange and Distance Fitting. The following sections will introduce these sub-procedures and the search strategies that use them.

4.5.1 Nearest Neighbor Interchange

Nearest Neighbor Interchange (NNI) is a procedure for slightly changing a tree topology. NNI is used in both our greedy and MCMC searches to explore the tree space. An *internal branch* is a branch which is not incident to a leaf. Every internal branch has four adjacent subtrees as illustrated in Figure 4.9. There exists three unique ways to attach adjacent subtrees to an internal branch in an unrooted tree. Given one topology, the other two can be obtained by swapping two subtrees, either A with C or A with D. In a tree of n leaves, there are $n-3$ internal branches. Therefore, there are a total of $2n-6$ possible NNIs for every unrooted tree.

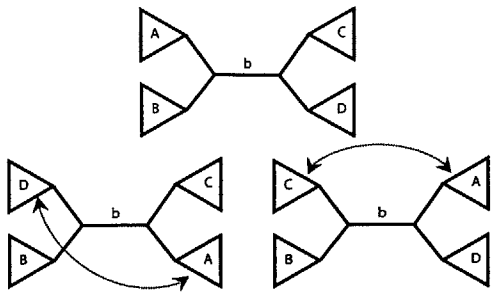


Figure 4.9: Example of the two possible Nearest Neighbor Interchanges for an internal branch b.

4.5.2 Fitting branch lengths

The NNI procedure can be used to propose new topologies, but in order to calculate the likelihood of any tree, we must also have lengths labelled on the branches. As input, SINDIR is given a distance matrix for the genes on the leaves of a proposed tree. Using Least Squared Error (LSE), SINDIR finds the labelling of lengths to a topology that best approximates the distances in the distance matrix. There exists an $O(n^2)$ run-time for Ordinary Least Squares (a form of LSE) [3]. By using LSE, SINDIR is able to constrain its tree search space by the distance matrix.

4.5.3 Search 1: Greedy Search

We implemented the greedy search used by the PHYLIP programs to find an optimal gene tree. The algorithm's pseudo-code is given in Figure 4.10. The search performs branch length fitting on every proposed topology and uses NNI to explore new trees. The total number of trees evaluated is:

$$\begin{aligned}\sum_{4}^{n}(i-1-3)+(2i-6) &= \sum_{4}^{n-1} 3i - 10 \\ &= \frac{3}{2}n^2 - \frac{17}{2}n - 36\end{aligned}$$

GreedySearch:

- order leaves randomly
- construct rooted tree T with first two leaves
- iteratively add leaf i to T, $i \in [3,n]$

  Add one branch

  - iterate over all branches, b, of T
    - bisect b, add branch connecting bisection and i
    - fit distances on T
    - calculate likelihood of T
  - let T be the ML tree thus far

  Optimize subtree

  - for each branch b of T
    - perform NNI on b
    - fit distances on T
    - calculate likelihood of T
  - Let T be ML tree thus far
- **return** ML tree T

Figure 4.10: Pseudo-code of greedy tree search used in PHYLIP.

4.5.4 Search 2: MCMC Search

Markov Chain Monte Carlo (MCMC) is a method for sampling from a probability distribution. We use it in this algorithm as a way to search the tree likelihood space. This proves to be fairly successful in finding the ML tree, since the ML tree is likely to get sampled the most number of times.

MCMC samples a probability distribution by recording the states of a random walk according to a Markov chain. As more samples are taken, the samples tend toward the stable distribution of the Markov chain. Thus, in order to sample the tree likelihood distribution, we need a Markov chain whose stable distribution is the same as the tree likelihood distribution. Such a Markov chain can be constructed using the Metropolis algorithm [5]. Given a current tree T_1 with likelihood p, use a method like NNI to randomly propose a new T_2 with likelihood q. If $q > p$, then accept T_2 as the next state in the Markov chain, otherwise accept T_2 with the probability q/p. As long as the proposal procedure proposes T_2 from T_1 with the same probability as proposing T_1 from T_2, then the accept/reject rule is guaranteed to sample the likelihood distribution of trees.

Our MCMC search begins on a tree found by the greedy search given in the previous section.

We propose new trees by applying two randomly chosen NNIs on the current tree. We find that running MCMC for roughly 2000 iterations for trees of a dozen genes is sufficient to find the ML tree nearly every time. We currently remember the likelihoods of visited trees by storing the likelihoods in a hash table where trees are hashed by their topologies. This hashing speeds the search greatly, when high likelihood tree spaces are sampled multiple times.

Chapter 5

SINDIR benchmark evaluation

We have tested SINDIR on a wide variety of real and simulated data in order to assess its ability to capture and correctly reconstruct phylogenies. Within this section, we introduce a novel method for evaluating reconstruction performance on real datasets, which has proven critical in understanding the true accuracy of current state of the art methods. This evaluation is general enough to be applied to any clade species that are reasonably close to each other. In particular, we demonstrate this approach on three diverse clades: fungi, flies, and mammals. Using a known case of Whole Genome Duplication (WGD) within the fungal species , we can also create a test set of real gene duplications. For simulated data, we introduce a new method of simulating gene trees using SINDIR's model of evolution. These simulations create gene trees that are likely to arise in the phylogenomics problem and allow testing of extremely complex cases of evolution. Together, these datasets provide a challenging and informative evaluation of SINDIR versus several state of the art techniques for phylogenetic reconstruction.

In nearly all of our tests, SINDIR out performs by a large margin. We believe this is due to SINDIR's ability to include information about the species tree. Most existing phylogenetic methods are statistically consistent. That is, as the length of the sequences increase the probability of reconstructing the correct phylogeny approaches one [12]. Therefore, the most likely explanation for the poor performance of existing methods, is that the average gene sequence is not long enough for current methods to reliably reconstruct. Therefore, in gene tree reconstruction, it becomes critical to incorporate more information, and according to our results, SINDIR's approach is very successful in this regard.

In the following sections, we describe our construction of real training and test sets for phylogenetic reconstruction (Section 5.1), the effect of SINDIR's duplication and loss parameters, and performance of SINDIR against state of the art phylogenetic methods on a diverse range of real and simulated datasets (Section 5.4.2)

5.1 Phylogeny evaluation by synteny

Evaluating phylogeny and orthology methods is often difficult because there exist very few datasets where the real answer is known. Some attempts have been made to create high confidence datasets, but this often requires stringent filtering of data that allows testing of only the most obvious cases of evolution or incomplete datasets (by leaving out paralogs) [35]. Instead, many evaluations are either qualitative or performed on simulated data, where the real answer is known because it is chosen. Simulated datasets are valuable, because they offer the ultimate control in testing different evolutionary scenarios. However, simulated data may not perfectly capture all the events that occur in reality. In this thesis, we developed a new method for determining high confident phylogenetic trees from real data. These trees can then be used for both SINDIR's training and evaluation. We build such trees using the following procedure.

First, we find syntenic regions between the species of interest. Many methods have been developed for finding synteny [20]. Nearly all methods use high scoring BLAST hits with some sort of heuristics to chain together hits of conserved order. We implemented our own synteny determination method that can be applied to multiple genomes. Using synteny, we can find genes that appear syntentically aligned in multiple genomes. Synteny is a strong signal of orthology. We consider a set of genes to be high confident orthologs, if they appear syntenically aligned in all genomes, and there are no other significant BLAST hits to any other genes in any genome.

Given a high confident set of orthologous genes, we can assume that no duplications occur within their phylogeny. There exists only one possible tree relating such genes. The tree can be determined exactly if the species tree is known. This is done by replacing each species in a species tree by the gene that comes from that species.

5.1.1 Measures of accuracy

We measure phylogenetic performance using two measures: tree correctness and orthology correctness. Tree correctness is the percent of trees reconstructed perfectly, such that their topology matches the known topology. Tree topologies are compared with unrooted trees, thus allowing evaluation of algorithms that produce unrooted trees. Orthology correctness is measured as the sensitivity and specificity of predicting a pair of genes to be orthologs. Orthologs are determined using the procedure given in Section 3.1.4

5.2 Data preparation

Once a set of genes are determined to syntentically align, we multiply align their derived protein sequences. Proteins are easier to accurately align than nucleotides because there are well known biases for substitutions that preserve hydrophobicity and polarity. If there are multiple splice forms for a gene, we use the longest. We use the MUSCLE software for multiple alignment [6]. Once a set of peptides are aligned, we reverse translate each sequence preserving the placement of gaps in order to produce a nucleotide alignment. For both peptide and nucleotide alignments, we calculate a distance matrix using the PHYLIP programs PROTDIST and DNADIST, respectively. The programs were run with default settings. The correct tree for each set of genes was derived differently depending on the dataset.

5.2.1 Existing phylogeny methods

We tested against the following phylogenetic reconstruction software packages:

- **PHYLIP (DNAML, PROML)** [13] We ran PROML with the JTT model of peptide evolution and PROML with Kimura two parameter with a transition/trasversion ratio of 2.
- **PHYML (DNA, Peptide)** [18] We ran PHYML for peptides with JTT model of evolution, estimated proportion of invariant sites, 4 rate categories, estimated alpha, and BIONJ as initial tree. For nucleotides, we used similar parameters along with an estimated kappa.
- **BIONJ (DNA, Peptide)** [15] We ran BIONJ with default parameters on both distance matrices derived from peptides and nucleotides.

5.3 Real data

We tested SINDIR against three other algorithms for phylogenetic reconstruction on real datasets from three different clades of species: fungi, flies, and mammals. In the following sections, we describe the methods used to derive our correct phylogenetic trees and the performance of our algorithm.

5.3.1 Fungal species

For our fungal dataset we used genes from seven species; four close species *S. cerevisiae* (scer), *S. paradoxus* (spar), *S. mikatae* (smik), and *S. bayanus* (sbay) (<2 mutations/site), which we call the *ingroup*, and three more distant species *K. lactis* (klac), *A. gossypii* (agos), and *K. waltii* (kwal) (>1.4 mutations/site), which we call the *outgroup*. The yeast genome has undergone Whole Genome Duplication (WGD) in its evolutionary history [21]. This means that every gene was duplicated simultaneously somewhere along the branch connecting the four close species and the three distant species, as shown in Figure 5.2. However, sometime after WGD many genes lost their paralogs, leaving most genes with only one copy.

Creating the one-to-one dataset

Genes annotations for *S. cerevisiae* were taken from SGD [19]. Other annotations were predicted by using TBLASTX to map *S. cerevisiae* genes to other genomes. In our synteny analysis we identified 2476 confident ortholog sets that contain exactly one gene from each of the 7 species. We call these ortholog sets *one-to-ones*. Care was taken in identifying orthologous genes from synteny, since the WGD is one of the few events that can produce syntenic paralogs that are not tandem. We first found genes from the ingroup with strong synteny, and then extended synteny groups where possible to all seven species. This prevented paralogs from being included in the ingroup species.

Determining species tree

The correct ancestry of the three outgroup species is still disputed, therefore, we ran PHYML on each of the 2476 nucleotide alignments in order discover the most likely species tree. The ten most frequent tree topologies are show in Figure 5.1. The most frequent tree was rooted on *K. waltii* and the remaining are rooted to closely match the first tree (for comparison only). Most of the

Tree topology	Count	Percent
((((((spar,scer),smik),sbay),klac),agos),kwal)	1,243	50.1%
(((((spar,scer),smik),sbay),kwal),(agos,klac))	708	28.6%
(((((spar,scer),smik),sbay),agos),(klac,kwal))	188	07.6%
((((((spar,scer),sbay),smik),klac),agos),kwal)	71	2.9%
(((((spar,scer),sbay),smik),kwal),(agos,klac))	34	1.4%
(((((spar,scer),smik),sbay),(agos,klac)),kwal)	28	1.1%
(((((sbay,smik),(spar,scer)),klac),agos),kwal)	26	1.0%
((((sbay,smik),(spar,scer)),kwal),(agos,klac))	21	0.8%
((((((smik,scer),spar),sbay),klac),agos),kwal)	17	0.7%
((((((smik,spar),scer),sbay),klac),agos),kwal)	16	0.6%

Figure 5.1: Ten most frequent yeast topologies by PHYML on 2576 DNA alignments. Trees are rooted to match first tree.

differences in these topologies are due to the placement of the outgroup species.

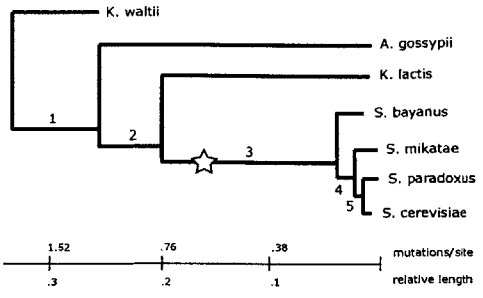
Training SINDIR on fungal species

We trained SINDIR on 1000 alignments from the 2476 dataset. These alignments were left out of our subsequent tests to avoid over fitting. The parameters found for each branch are given in Figure 5.2. The internal branches are numbered as indicated in the left panel. Notice that the variance is often less than half the mean. This indicates that >97.7% of the distribution is positive, meaning that even though the normal gives non-zero likelihood to negative branch lengths, it gives a very small likelihood. Therefore, a normal is a satisfactory approximation of the data.

Testing on fungal one-to-ones

In Figure 5.3, are the results of applying SINDIR along with three other phylogenetic algorithms on the 400 alignments of fungi one-to-one orthologs. SINDIR was run with duplication probability .1 and loss probability .1. These probabilities were set by running SINDIR on datasets with duplication and loss (Section 5.4.2).

All algorithms were run on both peptide and nucleotide alignments. For all algorithms, the nucleotide alignments are easier to reconstruct. Many of the mistakes made by the peptide reconstructions were in placing of the ingroup species. This is probably due to the fact that the ingroup species are very close and have not diverged enough at the amino-acid level to provide enough phylogenetically informative characters. SINDIR outperforms in both tree correctness and orthology



Branch	Mean	Standard deviation
1	0.0773	0.02482
2	0.0569	0.02619
3	0.1572	0.05070
4	0.0156	0.01104
5	0.0070	0.00561
scer	0.0151	0.00740
spar	0.0089	0.00535
smik	0.0219	0.00885
sbay	0.0241	0.01233
klac	0.1856	0.05274
agos	0.2461	0.06731
kwal	0.0773	0.02482

Figure 5.2: Left: The star represents the location of the Whole Genome Duplication. Branch lengths are proportional to the average mutations/site within coding regions. Right: SINDIR model parameters for 7 fungal species.

One-to-one fungi orthologs			
Method	Sequence	Correct Trees	Orth. Sn.
SINDIR	dna	198 (99.0%)	99.7%
PHYML	dna	87 (43.5%)	74.0%
PHYML	pep	28 (14.0%)	57.1%
DNAML	dna	98 (49.0%)	77.3%
PROML	pep	33 (16.5%)	60.8%
BIONJ	dna	103 (51.5%)	76.6%
BIONJ	pep	33 (16.5%)	57.1%

Figure 5.3: Phylogenetic reconstruction performance of one-to-one syntenic orthologs across 7 fungal species. Each method was applied to the same 200 randomly selected ortholog sets.

correctness. Only ortholog sensitivity is given in the Figure 5.3, because specificity for this dataset is always one; without duplicate genes, it is impossible to over predict orthology. Surprisingly, the NJ algorithm which is the fastest algorithm is also the most accurate of the competing methods.

Whole Genome Duplication in fungal species

SINDIR outperformed by a large margin for fungal one-to-one orthologs, however, this may be due to a bias that SINDIR has for gene trees that match the species tree. Therefore, we need to test against other datasets where the correct answer is a gene tree topology that actually differs from the species tree. The fungi genomes provide a great opportunity to test such trees.

Whole genome duplicated yeast ohnologs				
Method	Sequence	Correct Trees	Orth. Sn.	Orth. Sp.
SINDIR	dna	195 (97.5%)	99.7%	99.0%
PHYML	dna	96 (48.0%)	77.4%	100.0%
PHYML	pep	25 (12.5%)	56.3%	99.9%
DNAML	dna	83 (41.5%)	77.3%	100.0%
PROML	pep	26 (13.0%)	58.7%	99.9%
BIONJ	dna	78 (39.0%)	78.4%	100.0%
BIONJ	pep	29 (14.5%)	56.9%	99.9%

Figure 5.4: Phylogenetic reconstruction performance of Whole Genome Duplicated yeast ohnologs across 7 fungal species. Each method was applied to the same 200 randomly selected ortholog sets.

The ingroup species have all undergone WGD in their past, meaning that every gene was duplicated. This was first hypothesized by Ohno in 1970 as a mechanism for rapid evolution [28]. Accordingly, paralogs arising from WGD are often called *ohnologs*. The WGD appears to have been followed immediately by a period of rapid gene loss, most likely along branch 3 [21]. Therefore, most genes only appear with one copy. However, there is a small set of genes whose ohnolog has been kept. For such genes, we know where the duplication occurred because of the established dating of the WGD. This provides a perfect dataset of real gene duplications, where the correct tree can be inferred from synteny.

To identify genes whose ohnolog was kept, we first identified genes that appeared in syntenic alignment and had two syntenic regions for the ingroup species. We found 239 groups that passed our strict definitions of synteny across all seven species. We also filtered this set for cases of *gene conversion*, which is a phenomenon where one ohnolog's sequence is copied and replaces the other. This results in ohnologs with striking sequence similarity. This event also "erases" the history of one of the ohnologs. Therefore, we avoided such events by requiring that when NJ is applied to an ortholog set that two subtees of four ingroup genes each (one from each species) are created. This reduced our dataset to 225 ortholog groups.

In Figure 5.4 are the results of running the phylogenetic algorithms on the WGD dataset. The performance of the competing algorithms vary only slightly in comparison to the one-to-one test set. SINDIR drops in accuracy in comparison to the one-to-one dataset, but maintains a large improvement over competing algorithms. This is a very different dataset than the one-to-ones, because ohnologs are known to have varying mutation rates, that is one ohnolog often experiences

accelerated evolution. Most of this mutation occurs on branch 3. The fact that SINDIR can reconstruct these trees with such high accuracy indicates that SINDIR can handle the varying mutation rates within its model. This can be seen by the high variance allowed for branch 3 (Figure 5.2). Most of the errors committed by SINDIR in this test were due to placing *K. lactis* with the more divergent of the two ingroup genes. This most likely occurs because the accelerate branch is so long that SINDIR expects to see outgroup species branching off of it.

5.3.2 Fly species

The recently sequenced fly genomes offer a very interesting phylogeny to test against. Unlike the fungal species, the flies have many interior branching and thus have a complex tree topology. We ran SINDIR against this species to test SINDIR's ability to handle more complex trees.

For our fly dataset, we used genes from nine species: *D. melanogaster* (dmel), *D. simulans* (dsim), *D. erecta* (dere), *D. ananassae* (dana), *D. pseudoobscura* (dpse), *D. mojavensis* (dmoj), *D. virilis* (dvir), and *D. grimshawi* (dgri). Gene annotations for *D. melanogaster* were downloaded from FlyBase [17]. Gene annotations for the other species were found by mapping *D. melanogaster* genes using TBLASTX.

Reconstruction of one-to-one syntenic orthologs

We constructed a dataset of fly one-to-one orthologs using a strategy similar to the fungal one-to-one dataset. We found 3719 highly confident ortholog sets across all 9 species that had only one splice form each. There is currently a dispute over the correct species tree for the nine fly species, specifically the location of *D. erecta* and *D. yakuba*. Therefore, we used PHYML to find the ten most frequently reconstructed gene trees (Figure 5.5). The second topology shows a misplacement of a branch with high variance in mutation rate. The third topology shows the alternative branching of *D. erecta* and *D. yakuba*, but it is clearly a minority. We used the most frequent tree as our fly species tree.

From this dataset, we trained SINDIR on 1000 randomly picked alignments. These alignments were not used in testing. The parameters learned in training are shown in Figure 5.6. Again, the standard deviations are small compared to the mean branch lengths. We evaluated the phylogenetic algorithms on the one-to-one dataset similar to the fungal dataset. The results are given in Fig-

Tree topology	Count	Percent
(((((dmel,dsim),(dere,dyak)),dana),dpse),((dmoj,dvir),dgri))	1,460	39.2%
((((((dmel,dsim),(dere,dyak)),dana),dpse),dmoj),(dvir,dgri))	520	14.0%
((((((dmel,dsim),dere),dyak),dana),dpse),((dmoj,dvir),dgri))	424	11.4%
(((((((dmel,dsim),dere),dyak),dana),dpse),dmoj),(dvir,dgri))	231	6.2%
((((((dmel,dsim),dyak),dere),dana),dpse),((dmoj,dvir),dgri))	218	5.9%
(((((((dmel,dsim),dyak),dere),dana),dpse),dmoj),(dvir,dgri))	153	4.1%
((((((dmel,dsim),(dere,dyak)),dana),dpse),dvir),(dmoj,dgri))	131	3.5%
(((((((dmel,dsim),dere),dyak),dana),dpse),dvir),(dmoj,dgri))	57	1.5%
((((((dere,dyak),dmel),dsim),dana),dpse),((dmoj,dvir),dgri))	37	1.0%
(((((((dmel,dsim),dyak),dere),dana),dpse),dvir),(dmoj,dgri))	33	0.9%

Figure 5.5: Ten most frequent fly topologies by PHYML on 3719 DNA alignments. Trees are to match first tree.

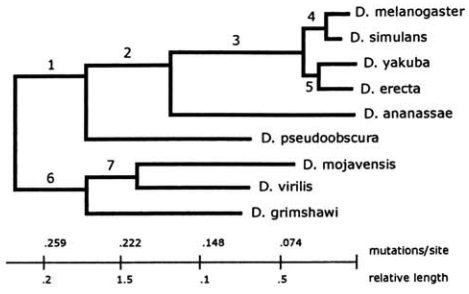
ure 5.7. With a larger more complex tree, the number of possible topologies increases. Accordingly, the performance of all the phylogenetic algorithms drop in comparison with the fungal evaluations. However, SINDIR maintains a high reconstruction accuracy.

Large fly gene families with simulated loss

To test the ability of SINDIR to handle duplications and loss in a large complex species tree, such as the flies, we modified the real fly orthologs sets. This was done by merging pairs of ortholog sets by average linkage of BLAST scores. This produced 624 sets of 18 genes each that represent gene families with a deep gene duplication that creates two subtrees of nine genes each. Then we randomly removed genes from the sets to simulate gene loss. The removal was done such that one gene from each species remains (Figure 5.8). This allows the possibility of proposing a tree that matches the species tree with no duplication or loss. However, SINDIR should notice that the branch lengths would be more likely generated from a topology with duplication and loss. As seen in Figure 5.9, these trees are difficult for all algorithms to reconstruct, probably because the branches leading the ancient duplication can be very long a difficult to estimate accurately. Again, SINDIR obtains the highest accuracy, even for a topology designed to be difficult.

5.3.3 Mammalian species

For the four fully sequenced mammals, human, dog, mouse, and rat, we wanted to test the specific case that confused Neighbor Joining (NJ) in Figure 2.2. From this dataset, we chose 1000 alignments



Branch	Mean	Standard deviation
1	0.0496	0.0140
2	0.0590	0.0149
3	0.0917	0.0190
4	0.0154	0.0072
5	0.0103	0.0051
6	0.0496	0.0140
7	0.0357	0.0113
dmel	0.0159	0.0061
dsim	0.0117	0.0063
dere	0.0240	0.0077
dyak	0.0264	0.0084
dana	0.1273	0.0242
dpse	0.1148	0.0232
dmoj	0.1089	0.0227
dvir	0.0778	0.0168
dgri	0.1073	0.0238

Figure 5.6: Left: Branch lengths are proportional to the average mutations/site within coding regions. Right: SINDIR model parameters for 9 species of fly.

One-to-one fly orthologs			
Method	Sequence	Correct Trees	Orth. Sn.
SINDIR	dna	372 (93.0%)	98.0%
PHYML	dna	174 (43.5%)	87.3%
PHYML	pep	121 (30.2%)	82.0%
DNAML	dna	166 (41.5%)	82.1%
PROML	pep	112 (28.0%)	79.9%
BIONJ	dna	193 (48.2%)	90.5%
BIONJ	pep	133 (33.2%)	83.2%

Figure 5.7: Phylogenetic reconstruction performance of one-to-one syntenic orthologs across 9 fly species. Each method was applied to the same 400 randomly selected ortholog sets.

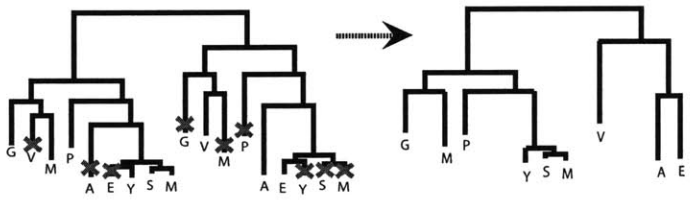


Figure 5.8: Example of how to construct large fly families with simulated loss.

Fly gene families with simulated lost					
Prob. of dup.	Probability of loss 1	.5	.1	.05	.01
.5	11%	18%	45%	48%	66%
.25	16%	23%	48%	58%	67%
.1	17%	28%	51%	56%	71%
.01	34%	38%	58%	64%	76%

Figure 5.9: Performance of SINDIR with several parameter settings on 200 simulated fly families with simulated loss.

of genes whose gene tree is strongly believed to be (dog,(human,(mouse,rat))). Recall that NJ actually found more trees matching ((dog,human),(mouse,rat)). In order for a tree to be of the second topology one duplication prior to speciation of mammals must occur followed by at least three gene losses. This would mean that the branch between the subtrees of (mouse,rat) and (dog,human) should be fairly long. Using SINDIR, we can ask which topology is more likely given the branch lengths of the tree.

To test for the specificity of predicting topologies, we also produced a dataset whose correct topology should be ((dog,human),(mouse,rat)). This was done using a similar strategy we used with the fly trees. Specifically, we paired together one-to-one orthologs sets that were most similar to each other and then removed mouse and rat from one set and dog and human from the other.

$$((\text{dog},(\text{human},\sout{(\text{mouse},\text{rat})})),(\sout{\text{dog}},(\sout{\text{human}},(\text{mouse},\text{rat})))) \rightarrow ((\text{dog},\text{human}),(\text{mouse},\text{rat}))$$

Using SINDIR, the likelihood of each topology for each cluster in both datasets was calculated and the most likely topology was chosen as the predicted topology. In Figure 5.11, the log likelihood of each topology for each gene cluster is shown. In Figure 5.10, the accuracy of SINDIR is calculated. SINDIR correctly predicts the topology in 90.2% of the gene clusters. Both its sensitivity and specificity are high with values 85.1% and 95.7%, respectively.

5.4 Simulated data

In addition to real datasets, we also tested our algorithm on simulated datasets. Simulated data allowed us to test more extreme cases of evolution that would be difficult to produce with our

One-to-one mammailan orthologs			
topologies	predict 1	predict 2	total
1. (dog,(human,(mouse,rat)))	**851 (85.1%)**	149 (14.9%)	1000
2. ((dog, human), (mouse, rat))	38 (4.3%)	**852 (95.7%)**	890

Sensitivity 85.1%, Specificity 95.7%

Figure 5.10: Comparison of mammalian toplogies using SINDIR

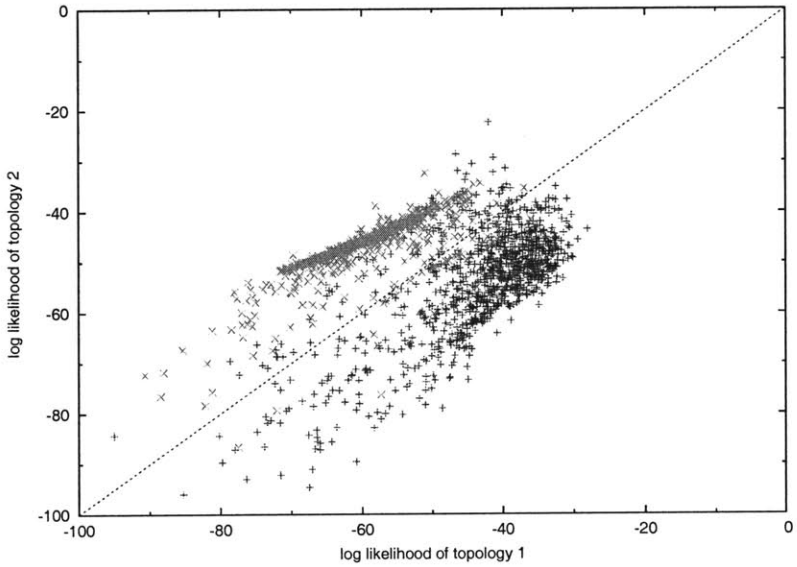


Figure 5.11: Comparison of two toplogies 1: (dog,(human,(mouse,rat))) and 2: ((dog,human),(mouse,rat)) for 1800 gene clusters. Clusters that are actually topology 1 are ploted red, and clusters of topology 2 are ploted green.

synteny technique on real data. Nucleotide alignments are simulated using the model of evolution that SINDIR assumes. The simulation model is described in Section 5.4.1 and test cases derived from simulations are presented in Section 5.4.2.

5.4.1 Simulation model

Since SINDIR reconstructs a gene tree by using a species tree, we must simulate gene trees that evolve inside species trees. Although simulation programs exist for evaluating phylogenetic algorithms, they all simulate sequence without knowledge of a species tree. Therefore, we had to produce our own.

The simulation takes as input a species tree and SINDIR's mutation model, which is the normal distribution of relative branch lengths for each species branch. The simulation begins with a random sequence at the root of the species tree. The base frequencies are chosen to match that of the fly genes. It then begins to evolve down each child branch of the species tree root. To evolve down a species branch, the simulation chooses a branch length from the normal distribution for that branch. It then mutates the sequence according the Kimura two parameter model. This process repeats as the sequences evolve down the species tree until the sequences reach the leaves. For simplicity, inserts and deletes are not simulated. Therefore, the final sequences are already perfectly aligned.

To simulate duplication and loss, there is a parameter for how many duplications D or losses L to create. Duplications are randomly assigned to D species tree branches. When a sequence reaches a branch assigned with a duplication, it chooses a duplication midpoint k between 0 and 1 uniformly, which represents how far down the species branch the sequence should evolve before it duplicates. This factor k is also multiplied with the chosen gene branch length. When the sequence reaches the duplication point, two new branches are created that recursively follow the same process. If more duplication midpoints are needed, they are chosen between k and 1. After a tree is fully simulated, L branches are randomly picked to be lost. These branches along with their descendents are removed. Also any leaves that are not reconciled to modern species are also pruned. The resulting tree is a simulated gene tree with at most D duplications and L losses.

One-to-one simulated fly orthologs			
Methods	slow-short	slow-med	slow-long
SINDIR	85.5%	97.5%	99.25%
DNAML	71.25%	96%	97.75%
	med-short	med-med	med-long
SINDIR	88.75%	98.75%	99%
DNAML	85.75%	97.25%	98.25%
	fast-short	fast-med	fast-long
SINDIR	90.75%	97.5%	98.25%
DNAML	87.5%	95%	96.5%

Figure 5.12: Performance on simulated fly one-to-ones

5.4.2 Simulated fly datasets

For our simulated datasets, we used the mutation model learned from real fly alignments. This created simulated fly gene trees. We ensured that the percent identity of our alignments equalled that of an average real fly alignment. We did this by adjusting the simulation's base rate parameter. We also ensured that the transition/transversion ratio was the same (.914).

Our first simulation tested the affect of mutation rate and sequence length. We identified the average mutation rate as 1.2 mutations/site (along the entire tree) and the average alignment length as 1500. We then chose a slower (.5) and a faster (3.0) rate of mutation along with a shorter (500) and longer (3000) alignment length. In Figure 5.12, we give the accuracies of reconstruction for SINDIR and DNAML on each possible conbination of speed and sequence length. Both algorithms can reconstruct these datasets faily well, however SINDIR has a higher performance. The slow and short dataset appears the most challenging.

Simulated duplication and loss

Next, we performed simulated tests with duplication and loss. In Figure 5.13, we tried three levels of loss (1, 2, and 3) and two levels of duplications (1 and 2). We also created two datasets with both duplication and loss. For these last datasets, we tested 20 different parameter settings for SINDIR (Figure 5.14 and Figure 5.15). From these results, we found that performance plateaued around duplication probability .1 and loss probability .1. These parameters were used in the rest of our tests.

Simulated fly orthologs with duplication and loss							
	Losses			Dups		Dups & Losses	
	1	2	3	1	2	2&3	3&3
SINDIR	99%	98%	99%	91%	87%	$\star$	$\star$
DNAML	96%	98%	98%	81%	75%	91%	82%
PHYML	96%	98%	98%	80%	66%	81%	79%
BIONJ	93%	97%	98%	75%	65%	83%	79%

Figure 5.13: Performance on 400 alignments of simulated fly duplication and loss. $\star$ For performance of SINDIR on tree with duplication and loss see Figure 5.15 and Figure 5.15.

Simulated fly orthologs (dup 2 & loss 3)					
Prob. of	Probability of loss				
dup.	1	.5	.1	.05	.01
.5	73%	81%	88%	90%	92%
.25	79%	81%	89%	89%	91%
.1	81%	83%	89%	91%	91%
.01	81%	85%	90%	89%	92%

Figure 5.14: SINDIR performance on 400 alignments of simulated fly with 2 duplications and 3 losses.

Simulated fly orthologs (dup 3 & loss 3)					
Prob. of	Probability of loss				
dup.	1	.5	.1	.05	.01
.5	64%	70%	81%	85%	88%
.25	66%	73%	84%	88%	87%
.1	70%	74%	81%	88%	87%
.01	74%	81%	84%	87%	87%

Figure 5.15: SINDIR performance on 400 alignments of simulated fly with 3 duplications and 3 losses.

Chapter 6

Future work

With more fully sequenced genomes becoming available, there will be an increasing need for phylogenomic approaches to relate all available genes and regulatory regions. SINDIR has been designed for reconstructing highly accurate gene trees for such situations. In the future, we plan to include SINDIR in a larger phylogenomic pipeline, which will include gene annotation, synteny construction, gene family clustering, and an interface for visualizing orthologies and phylogenies.

Along with this system, there are several other possible directions for improving SINDIR itself. First, SINDIR can be extended from reconstructing the phylogenies of genes to any arbitrary genomic region. Many of the same principles on which SINDIR operates, still apply in this context. This requires confident identification and clustering of such regions, as well as exploration of whether the mutation rates of such regions can be described with similar distributions as done with genes. For example, do intergenetic regions have a base rate, like the one observed for genes?

A second future direction, would be to improve SINDIR's ability to reconstruct larger gene families. The most common error SINDIR committed in reconstructing the whole genome duplicated yeast genes was the placing of *K.lactis* with the more accelerated post-duplication subtree containing *S. cerevisiea*, *S. paradoxus*, *S. mikatae*, and *S. bayanus*. This is most likely due to SINDIR's use of a single base rate for an entire tree. Usually this assumption is accurate, however, in gene families where there are deep paralogs or many ancient duplications, the tree may actually be better described with multiple base rates. However, adding the ability to infer multiple base rates must be done carefully. Multiple base rates add parameters to SINDIR's model and thus will always allow better fitting to the data. In the extreme, the data can be completely over fitted by

assigning an individual base rate for each branch. One possible way of balancing the addition of parameters with better fitting is to use a log-odds ratio test. That is, compare the log likelihood of a tree with one base rate versus two. Then only except two base rates, if two base rates increases the tree likelihood more than what would be expected at random.

Another improvement would be to use a better method of distance estimation. As seen in the evaluation section, nearly all algorithms had an improved accuracy using substitutions per nucleotide rather than amino-acid. This is because most of our datasets contained very close species, that had very few mutations in their peptides. However, when comparing more distant species the nucleotide mutations begin to saturate and conservation is only observed at the protein level. Perhaps we could use a distance measure that trades off between nucleotide and amino-acid mutations depending on the evolutionary distance. Another observation is that not all mutations are equally important. Mutations in critical portions of a protein domain are more rare than those in a linker region. Therefore, one possibility of improving distance estimation is using Hidden Markov Modeling of protein domains, and weighting mutations by there conservation of domain structures.

Lastly, a very promising extension to SINDIR is the use of the species tree in tree search. Many gene trees look very similar to the species tree. If the species tree could be used to focus the tree search, the search space can be greatly reduced, increasing the chances of find the maximum likelihood tree.

Chapter 7

Contributions

In this thesis, we presented four main contributions to the phylogenomics and phylogenetic problems.

1. **Novel method of phylogeny evaluation on real data using synteny**

   Most evaluations of existing algorithms have been on simulated data or have been based on confidence scores or posterior probabilities. This is because evaluating a phylogenetic algorithm requires a dataset where the correct answer is known, and for real data, it is very difficult to know the phylogenetic history independent of phylogenetic algorithms. However, with the publishing of multiple complete genomes, synteny can be found and used as an independent method of orthology determination. We showed how orthology can be combined with a known species tree to produce a high confident gene tree.

   Although, the tree for some species remains questionable, it is likely these issues can be eventually resolved by using more data. In the species tree reconstruction problem, potentially the entire genome can be used for reconstruction. Therefore, we feel it is reasonable to require the species tree as input to our algorithm.

2. **Novel distance-based probabilistic model for phylogeny**

   Existing methods can be roughly classified as either distance-based or character-based. Nearly all distance-based methods are formulated as a cost minimization, which is difficult to integrate with other information. We are aware of only one probabilistic distance-based algorithm, Weighbor, which models the distribution of errors in distance estimation [2]. On the other

hand, there exist many implementations of probabilistic character-based methods. Unfortunately, character-based methods usually require much more computation and are thus less desirable for phylogenomics.

In contrast, SINDIR combines the best features of both of distance-based and character-based methods, by using a probabilistic model of distances. This leads to a fast and accurate phylogenetic algorithm. SINDIR is also the first algorithm we are aware of that learns its model of evolution from a training set of real phylogenetic trees. This gives SINDIR an edge by informing the algorithm what mutation rates are expected for each branch.

3. **Implementation of model in a new phylogenetic algorithm**

   In this thesis, we implemented our model of evolution in a new phylogenetic algorithm called SINDIR, Species INformed DIstance-based Reconstruction. The algorithm was prototyped in python and evaluated on both real and simulated datasets. The prototype allowed quick exploration of various features and approaches. Now that we have a stable program, we plan to re-implement the algorithm in a faster native language such as C++. This will provide much needed speed for larger gene families and execution on entire genomes.

4. **Significantly improved phylogeny and orthology reconstruction**

   In Section 5, we evaluated our algorithm on a diverse range of both real and simulated datasets. In nearly all of our tests, SINDIR outperformed the current state-of-the-art methods by a large margin. We believe this is due to SINDIR's use of the species tree to help inform the reconstruction of the gene tree. Many of the errors committed by existing algorithms are minor in terms of topology difference. However, when the likelihood of evolutionary events such as duplication and loss is considered, these differences can be identified as extremely unlikely. When phylogenetic algorithms are applied tens of thousands of times across all the gene families of multiple genomes, it will become important to avoid such frequent and extremely dramatic errors. Therefore, SINDIR's accuracy is critically needed for the phylogenomics approach.

Appendix A

Derivations

A.1 Estimating gene tree base rates

Let B be the base rate. For each branch i, we have branch length x_i, rate factor F_i, relative rate mean μ_i and standard deviation σ_i.

Given several x_i we want to find maximum likely B.

$$X_i = F_i * B$$

$$P(X_i = x_i|\mu_i, \sigma_i) = normal(x_i|\mu_i, \sigma_i^2)$$

$$B = X_i/F_i \ \ \forall i$$

$$P(B = b|\alpha, \beta) = gamma(b|\alpha, \beta)$$

$$P(B = b|x_i, \mu_i, \sigma_i \ \forall i, \alpha, \beta) = P(x_1/F_1 = b, ..., x_n/F_n = b) * P(B = b|\alpha, \beta)$$

Since F_i are indenpendent from each other, we can factor this likelihood for each branch.

$$
\begin{aligned}
P(B = b|x_i, \mu_i, \sigma_i \ \forall i) &= P(B = b) \prod_i P(x_i/F_i = b) \\
&= P(B = b) \prod_i P(F_i = x_i/b) \\
&= gamma(b|\alpha, \beta) \prod_i normal(x_i/b|\mu_i, \sigma_i^2)
\end{aligned}
$$

The maximum likelihood estimator of B given α, β, x_i, μ_i, and σ_i for all i is then

$$
\begin{aligned}
arg\max_b P(B = b|x_i, \mu_i, \sigma_i \ \forall i, \alpha, \beta) &= arg\max_b gamma(b|\alpha, \beta) \prod_i normal(x_i/b|\mu_i, \sigma_i^2) \\
&= arg\max_b b^{\alpha-1} e^{-\beta b} \frac{\beta^\alpha}{\Gamma(\alpha)} \prod_i \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left(-\frac{(\frac{x_i}{b} - \mu_i)^2}{2\sigma_i^2}\right)
\end{aligned}
$$

Applying a natural log gives

$$
\begin{aligned}
&= arg\max_b (\alpha - 1) \log b - \beta b + \log \frac{\beta^\alpha}{\Gamma(\alpha)} + \sum_i \log \frac{1}{\sigma_i \sqrt{2\pi}} + \sum_i \left(-\frac{(\frac{x_i}{b} - \mu_i)^2}{2\sigma_i^2}\right) \\
&= arg\max_b (\alpha - 1) \log b - \beta b + K + \sum_i \left(-\frac{(\frac{x_i}{b} - \mu_i)^2}{2\sigma_i^2}\right)
\end{aligned}
$$

Where K is the constant $\log \frac{\beta^\alpha}{\Gamma(\alpha)} + \sum_i \log \frac{1}{\sigma_i \sqrt{2\pi}}$. Setting the derivate to zero gives

$$
\begin{aligned}
0 &= \frac{d}{db}(\alpha-1)\log b - \beta b + K + \sum_i \frac{-1}{2\sigma_i^2}\left(\frac{x_i}{b}-\mu_i\right)^2 \\
&= \frac{\alpha-1}{b} - \beta + \sum_i \frac{-1}{\sigma_i^2}\left(\frac{x_i}{b}-\mu_i\right)\left(-x_i b^{-2}\right) \\
&= \frac{\alpha-1}{b} - \beta + \sum_i \left(\frac{x_i^2}{b^3\sigma_i^2} - \frac{\mu_i x_i}{b^2\sigma_i^2}\right) \\
&= (\alpha-1)b^2 - \beta b^3 + \sum_i \frac{x_i^2}{\sigma_i^2} - \sum_i \frac{\mu_i x_i b}{\sigma_i^2}
\end{aligned}
$$

Which gives the following cubic polynomial

$$
b^3 - \frac{\alpha-1}{\beta}b^2 + \left(\frac{1}{\beta}\sum_i \frac{\mu_i x_i}{\sigma_i^2}\right)b - \frac{1}{\beta}\sum_i \frac{x_i^2}{\sigma_i^2} = 0
$$

A.2 Number of rooted and unrooted binary trees

First let us count the number of unrooted binary trees of n leaves. Number of the leaves in an arbitrary order 1 through n. For $n <= 3$ there is exactly one possible unrooted tree. For $n > 3$, use the following induction. Assume we have a tree of $n-1$ leaves. A tree of n leaves can be created by bisecting any of the $2(n-3)+1$ branches and attaching a new branch that connects the bisection and the leaf n. Each of these trees are unique, and if leaf n is added last, can only be created by starting with a single tree of $n-1$ leaves. Therefore, the number of topologies for a unrooted binary tree of n leaves is

$$
\begin{aligned}
U &= 3 * 5 * 7 * ... * (2n-5) \\
&= (2n-5)!!
\end{aligned}
$$

Each rooted tree is defined as an unrooted tree and a rooting branch. There are $2n-3$ branches in an unrooted tree of n leaves. Therefore, the number of rooted tree topologies is

$$R = (2n-3) * U = (2n-3) * (2n-5)!!$$

Bibliography

[1] Lars Arvestad, Ann-Charlotte Berglund, Jens Lagergren, and Bengt Sennblad. Gene tree reconstruction and orthology analysis based on an integrated model for duplications and sequence evolution. *RECOMB*, 2004.

[2] William Bruno, Nicholas Socci, and Aaron Halpern. Weighted neighbor joining: A likelihood-based approach to distance-based phylogeny reconstruction. *Molecular Biology Evolution*, 17, 2000.

[3] David Bryant and Peter Waddell. Rapid evaluation of least-squares and minimum-evolution criteria on phylogenetic trees. *Mol. Biol. Evol.*, 15:1346–1359, 1998.

[4] Mouse Genome Sequencing Consortium. Initial sequencing and comparative analysis of the mouse genome. *Nature*, 420, 2002.

[5] R. Durbin, S. Eddy, A. Krogh, and G.Mitchison. *Biological sequence analysis: Probabilistic models of proteins and nucleic acids.* Cambridge University Press, 1998. Metropolis explanation.

[6] Robert C. Edgar. Muscle: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research*, pages Vol. 32, No. 5 1792–1797, 2004.

[7] Jonathan A. Eisen. Phylogenomics: improving functional predictions for uncharacterized. *Genome Research*, pages 8(3):163–7, 1998.

[8] Eric S Lander et. al. Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921, February 2001.

[9] K Lindblad-Toh et al. Genome sequence, comparative analysis and haplotype structure of the domestic dog. *Nature*, 438, 2005.

[10] Olivier Jaillon et. al. Genome duplication in the teleost fish tetraodon nigroviridis. *Nature*, pages 431, 946 – 957, 21 October 2004.

[11] Robert H Waterston et. al. Initial sequencing and comparative analysis of the mouse genome. *Nature*, 420(6915):520–562, December 2002.

[12] J. Felsenstein. Phylogenies from molecular sequences: Inference and reliability. *Annu. Rev. Genet*, 22, 1988.

[13] J Felsenstein. Phylip (phylogeny inference package) version 3.6. *Distributed by the author. Department of Genome Sciences, University of*, 2005.

[14] W. M. Fitch and E. MARGOLIASH. Construction of phylogenetic trees. *Science*, 155:279–284, 1967.

[15] O Gascuel. Bionj: an improved version of the nj algorithm based on a simple model of sequence data. *Molecular Biology and Evolution*, 14:685–695, 1997.

[16] R Gibbs, G Weinstock, M Metzker, D Muzny, E Sodergren, and S Scherer et al. Genome sequence of the brown norway rat yields insights into mammalian evolution. *Nature*, 428, April 2004.

[17] G. Grumbling, V. Strelets, and The FlyBase Consortium. Flybase: anatomical data, images and queries. *Nucleic Acids Research*, 34, 2006.

[18] S Guindon and O Gascuel. A simple, fast, and accurate algorithm to estimate large phylogenies by maximum likelihood. *Systematic Biology*, 52, 2003.

[19] JE Hirschman, R Balakrishnan, KR Christie, MC Costanzo, SS Dwight, SR Engel, DG Fisk, and EL et al Hong. Genome snapshot: a new resource at the saccharomyces genome database (sgd) presenting an overview of the saccharomyces cerevisiae genome. *Nucleic Acids Research*, 34, 2006.

[20] T Hubbard, D Andrews, M Caccamo, G Cameron, Y Chen, M Clamp, L Clarke, G Coates, T Cox, and F et al Cunningham. Ensembl 2005. *Nucleic Acids Research*, 33, January 2005.

[21] Manolis Kellis, Bruce Birren, and Eric Lander. Proof and evolutionary analysis of ancient genome duplication in the yeast saccharomyces cerevisiae. *Nature*, April 2004.

[22] Manolis Kellis, Nick Patterson, Bruce Birren, Bonnie Berger, and Eric S Lander. Methods in comparative genomics: genome correspondence, gene identification and regulatory motif discovery. *J Comput Biol*, 11(2-3):319–355, 2004.

[23] Golding GB. Koski LB. The closest blast hit is often not the nearest neighbor. *J Mol Evol*, pages 52(6):540–2, 2001 Jun.

[24] Heng Li, Avril Coghlan, Jue Ruan, Lachlan james Coin, Jean-Karim Heriche, Lara Osmotherly, Ruiqiang Li, Tao Liu, Zhang Zhang, Lars Bolund, Gane Ka-Shu Wong, Weimou Zheng, Paramvir Dehal, Jun Wang, and Richard Durbin. Treefam: a curated database of phylogenetic trees of animal gene families. *Nucleic Acids Research*, 34, 2006.

[25] Li Li, Christian Stoeckert Jr, and David Roos. Orthomcl: Identification of ortholog groups for eukaryotic genomes genome research. 13, 2003.

[26] Li Li, Christian J Jr Stoeckert, and David S Roos. Orthomcl: identification of ortholog groups for eukaryotic genomes. *Genome Res*, 13(9):2178–2189, September 2003.

[27] Masatoshi Nei and Sudhir Kumar. *Molecular Evolution and Phylogenetics.* Oxford University Press, 2000. Tree topology search strategies.

[28] S Ohno. *Evolution by Gene Duplication.* Allen and Unwin, 1970.

[29] M Remm, CEV Storm, and Sonnhammer ELL. Automatic clustering of orthologs and in-paralogs from pairwise species comparisons. *JMB*, 314:1041–1052.

[30] Huelsenbeck JP. Ronquist F. Mrbayes 3: Bayesian phylogenetic inference under mixed models. *Bioinformatics*, pages 19(12):1572–4, 2003 Aug 12.

[31] N Saitou and M Nei. The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol Biol Evol*, 4(4):406–425, July 1987.

[32] Michael J. Sanderson and H. Bradley Shaffer. Troubleshooting molecular phylogenetic analyses. *Annual Reviews*, pages Vol. 33: 49–72, November 2002.

[33] Chimpanzee Sequencing and Analysis Consortium. Initial sequence of the chimpanzee genome and comparison with the human genome. *Nature*, September 2005.

[34] Sneath, P. H. A., and R. R. Sokal. *Numerical taxonomy.* W. H. Freeman, San Francisco, 1973.

[35] Christian E.V. Storm and Erik L.L. Sonnhammer. Comprehensive analysis of orthologous protein domains using the hops database. 2003.

[36] Roman L Tatusov, Natalie D Fedorova, John D Jackson, Aviva R Jacobs, Boris Kiryutin, Eugene V Koonin, Dmitri M Krylov, Raja Mazumder, Sergei L Mekhedov, Anastasia N Nikolskaya, B Sridhar Rao, Sergei Smirnov, Alexander V Sverdlov, Sona Vasudevan, Yuri I Wolf, Jodie J Yin, and Darren A Natale. The cog database: an updated version includes eukaryotes. *BMC Bioinformatics*, 4:41, September 2003.

[37] Xiaohui Xie, Jun Lu, EJ Kulbokas, Todd Golub, Vamsi Mootha, Kerstin Lindblad-Toh, Eric Lander, and Manolis Kellis. Systematic discovery of regulatory motifs in human promoters and 3útrs by comparison of several mammals. *Nature*, February 2005.

[38] Christian M Zmasek and Sean R Eddy. Rio: analyzing proteomes by automated phylogenomics using resampled inference of orthologs. *BMC Bioinformatics*, 3:14, May 2002.

[39] Eddy SR. Zmasek CM. A simple algorithm to infer gene duplication and speciation events. *Bioinformatics*, 2001.