Georg-August-Universität
Göttingen
Zentrum für Informatik

ISSN 1612-6793
Nummer ZFI-BM-2007-13

Masterarbeit

im Studiengang "Angewandte Informatik"

A phylogeny pipeline and its application to contribute to resolving the phylogeny of sponges (Phylum Porifera)

Fabian Schreiber

an der

Biologische Fakultät,
Abteilung Bioinformatik

Bachelor- und Masterarbeiten
des Zentrums für Informatik
an der Georg-August-Universität Göttingen

11. Mai 2007

Georg-August-Universität Göttingen
Zentrum für Informatik

Lotzestraße 16-18
37083 Göttingen
Germany

Tel. +49 (5 51) 39-1 44 02
Fax +49 (5 51) 39-1 44 03
Email office@informatik.uni-goettingen.de
WWW www.informatik.uni-goettingen.de

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Göttingen, den 11. Mai 2007

Master's thesis

A phylogeny pipeline and its application to contribute to resolving the phylogeny of sponges (Phylum Porifera)

Fabian Schreiber

May 11, 2007

supervised by Prof. Burkhard Morgenstern
Faculty of Biology, Department of Bioinformatics
Georg-August-University of Göttingen

co-supervised by Jun-Prof. Dr. Gert Wörheide
Geoscience Centre Göttingen, Department of Geobiology
Georg-August-University of Göttingen

Abstract

The study of the evolutionary history of organisms (Phylogeny) is a demanding task involving different steps of analyses. The automation of this steps will make phylogenetic analyses faster and easier to do.

We will develop a phylogenetic pipeline capable of carrying out all steps needed for a complete phylogenetic analysis. In a next step we will apply this pipeline to try to answer a fundamental question about the phylogeny of sponges.

Recent phylogenetic studies revealed that the group of sponges consists of an inferred common ancestor and but not all of its descendants. This is contrary to what people previously thought about the history of sponges.

The application of the phylogenetic pipeline to this problem should give an answer which hypothesis to corroborate.

Table of Contents

List of Figures

List of Tables

Abbreviations and Acronyms

AA	**A**mino **A**cid
BI	**B**ayesian **I**nference
BLAST	**B**asic **L**ocal **A**lignment **S**earch **T**ool
BLOSUM	**BLO**ck **SU**bstitution **M**atrices
cDNA	**c**omplementary **D**eoxyribo**N**ucleic **A**cid
DDBJ	**D**NA **D**ata **B**ank of **J**apan
DNA	**D**eoxyribo**N**ucleic **A**cid
EMBL	**E**uropean **M**olecular **B**iology **L**aboratory
EST	**E**xpressed **S**equence **T**ag
INSD	**I**nternational **N**ucleotide **S**equence **D**atabase Collaboration
LBA	**L**ong **B**ranch **A**ttraction
MCMC	**M**arkov **C**hain **M**onte **C**arlo
ML	**M**aximum **L**ikelihood
MRP	**M**atrix **R**epresentation using **P**arsimony
MSA	**M**ultiple **S**equence **A**lignment
NNI	**N**earest **N**eighbour **I**nterchange
PCR	**P**olymerase **C**hain **R**eaction
RNA	**R**ibo**Nu**cleic **A**cid
SPR	**S**ubtree **P**rune and **R**egraft
T-Coffee	**T**ree-based **C**onsistency **O**bjective **F**unction **f**or alignment **E**valuation
URL	**U**niform **R**esource **L**ocator

Chapter 1

Introduction

1.1 Evolutionary Theory and Diversity of Life

To prepare himself for the study of natural history in the tropics, *Charles Darwin* visited the *Galapagos Island* aboard *HMS Beagle* to investigate living organisms and fossils. He observed finches and noticed that they differed depending on the island they came from. The population of finches had adopted to the circumstances of each individual island. He assumed that some kind of selection process was acting on the populations. After returning home from the *Galapagos Islands*, *Darwin* published his

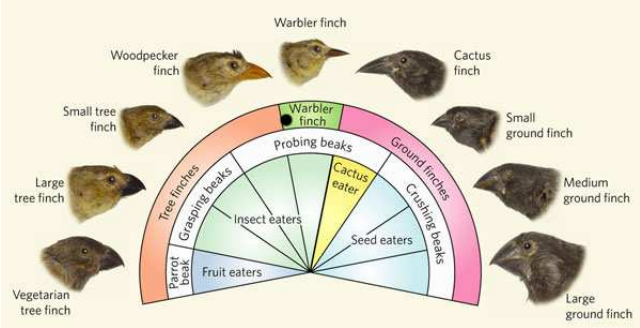

Figure 1.1: The fourteen different species of finches that were observed by *Darwin* on the *Galapagos Islands*. (Figure taken from (Wel00).)

new observations and assumptions in the book "The origin of Species by Means of Natural Selection" (Dar59). In the book he assumed that "all the organic beings which have ever lived on this earth have descended from some one primordial form" and supported the hypothesis with many examples. This knowledge formed the basis for the creation of a new field of science: The study of evolutionary history of organisms. *Darwins* ideas encouraged other scientists to use morphological, paleontological, and biogeographical information to infer relationships among species. *Darwin*, his colleagues, and his successors used this information to understand the relationship among species and the range of biodiversity on earth.

1.1.1 Natural Selection

Darwin called the main driving force of evolution "natural selection". This process acts within and between populations. It assumes that there are more individuals than the environment can support.

Therefore, there is a competition for survival within and between populations of individuals. Individuals that can adopt best to the environment have a higher likelihood to survive and produce

offsprings. In the course of time this process forms populations of individuals well adapted to an environment. As the environment can change, so can the process of natural selection. A change of the environment can lead to a change in the way natural selection favors traits. Changes to an environment can be e.g. a change of climate an volcanic eruption.

1.1.2 Modern Synthetic Theory

Darwin's idea of natural selection was fundamental, but he couldn't describe the genetic basis. Based on this knowledge *Gregor Meddel* introduced the impact of heredity and variability in populations from his observations on peas (Men66). In the 20th century *Huxley* united the fields of genetics and evolution with his publication about neo-Darwinism (Hux74). Now it was clear that genetic mutation was the basis of natural selection. In the early 1960s protein data showed that the variability among populations was much greater than expected (BS67). Various theories were proposed at that time. One such theory was the neutral theory of evolution, which assumes that most mutations are neutral and dependend on mutation rate and population size (Kim68). The neutral theory and the theory of natural selection both assume only a small proportion of mutations to affect the individuals of a population.

1.1.3 The Genomic Era

In the early 20th century, *William Bateson* introduced the name genetics to describe the study of heredity, variation and inheritance.

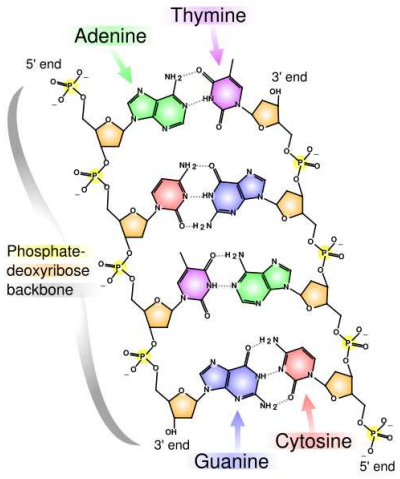

Figure 1.2: Two-dimensional structure of DNA. (Figure taken from (DNA).)

It was only in the year 1953, when Watson and Crick investigated the very basic of molecular genetics - the deoxyribonucleic acid (DNA). They were the first to describe its physical and chemical structure. The DNA contains the information for the development and functioning of all living organisms. It is a long polymer of of nucleotides (A - Adenine, G -Guanine, C - Cytosine, and T - Thymine) held together by a sugar-phosphate backbone (See Fig. 1.2). The DNA codes for the production of messenger RNA (mRNA) and ribosomes read the coded information carried by the mRNAs and use it for protein synthesis. These processes are called transcription and translation respectively. The

flow of information from DNA to RNA and to the Protein sequence is known as "the central dogma of molecular biology" (See Fig. (Dog)) The DNA can be copied and passed from one generation to the next one. Since the replication process is not always perfect, errors can be introduced to the sequence of nucleotides.

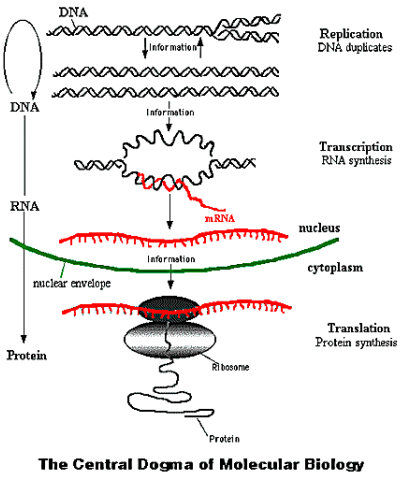

Figure 1.3: The information flow known as the central dogma of molecular biology. The sequence of DNA is first transcribed to RNA and then translated to a protein sequence. (Figure taken from (Dog).)

These errors include insertions and deletions of nucleotides (indels) as well as substitutions of nucleotides. Changes to this sequence are referred to as mutations. The invention of fast sequencing methods (SC75) and the polymerase chain reaction (SSF$^+$85) formed the basis for the automation of DNA sequencing. With DNA sequences becoming available scientist started to use molecular data rather than morphological to infer phylogeny. First they used single genes, but later whole genomes could be considered in phylogenetic studies.

1.1.4 Tree of Life

Darwins had the idea that all living things are related through common ancestry and that this relation could be expressed by a "great Tree of Life" (Darwin, 1859). Scientists used *Darwins* idea of representing the process of evolution as a tree structure to classify organisms. In the course of time this tree underwent some major changes: *Haeckel* (1866) divided all organism into two domains, plants and animals, *Copeland* (1938) into four and *Whittaker* (1959) extended to five domains to accommodate the fungi. *Carl Woese* and colleagues were the first to use the advancing molecular techniques and were able to identify the three domains of life: Prokaryotes, Archaea, and Eukaryotes (WF77). The study of evolutionary processes not only revealed the tree of life (See Fig. 1.4) but also helped in other fields of biology:
Evolutionary analyses have been applied to estimate the timing of a common ancestor of the HI virus (KMT$^+$00), investigating the origins of deadly *flu 5* strains (WRP$^+$02) and the genetic mechanisms of malaria (KEI$^+$02).

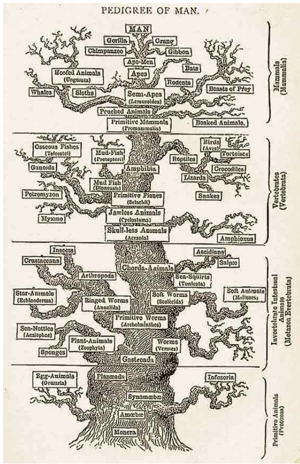

Figure 1.4: The Tree of Life as proposed by *Ernst Haeckel* in "The Evolution of Man" (1879). (Figure taken from (Tre).)

1.2 Reconstruction of Phylogenies

The study of molecular phylogenetics can be seen as the problem of finding the tree diagram that represents the relationship for a set of amino-acid or nucleotide sequences best. In the course of time a great variety of approaches to this problem have been proposed. In general, these approaches split up the reconstruction of phylogenies into two main steps - the alignment and the inferring of the tree itself. The first step for a successful phylogenetic analysis is the appropriate choice of a dataset.

While morphological data (size, biochemical properties, etc) were used in the beginning, scientists now use molecular data (DNA or protein sequences). E.g. rRNA sequences as the 18S rRNA or the 28S rRNA are useful markers in phylogenetic analysis, because genes that encode the rRNA are found in every organism and these genes are the most conserved genes in all cells. The focus of phylogenetic studies is to compare organisms of interest to other well-studied organism. Tools like *BLAST* (See Sec. 2.1.2) try to find similar sequences in databases. This is an important step, as the choice of inappropriate sequences can lead to a misleading result of the analysis (See Sec. 2.2). Although sequencing methods have been improved, only a small proportion of genes for a small number of organism have been sequenced and scientist have to decide which taxa / characters to include or to exclude from the analysis. This is another point to deal with in the process of finding an appropriate dataset (See Sec. 2.2) ready to be aligned in the next step.

The alignment (See Sec. 2.3) is the lining up of the sequences in a way that parts of the sequences that seem to be evolutionary related are grouped. Although it is possible to do alignments by hand, only automated methods will be used in this thesis. The evolutionary process has shaped molecular sequences by inserting / deleting or substituting nucleotides and therewith amino acids. Alignment methods try to maximise the similarity between a set of input sequences by introducing gaps. The computation of an optimal - according to some cost function - alignment for two sequences of length n requires time $O(n^2)$ using dynamic programming algorithms. Nowadays datasets consists of several tens of sequences and the time to align these sequences is $O(n^m)$, where m is the number of sequences, and heuristics have to be used to master this problem. Some of these heuristics simplify the computation of multiple sequence alignments by pairwise aligning the most similar sequences and adding

the next similar sequence (See Sec. 2.3.5.1). More advanced heuristics allow heterogeneous inputs (See Sec. 2.3.5.2), extend the side-by-side comparison to a segment-by-segment comparison (See Sec. 2.3.5.3) or use k-mer counting and profiles for the alignment (See Sec. 2.3.5.4). Once the alignment has been computed it can be curated either automatically or manually. Whereas the manual curation requires expert knowledge, the automated curation is based on the selection of similar (conserved) blocks for further analysis (See Sec. 2.3.6).

Darwin was the first to postulate the use of a tree to represent the process of evolution (1859). With the course of time, scientists invented different method of inferring trees. In the beginning the simply counted the number of nucleotide / amino acid matches between sequences to get a measure of similarity and used distance methods as neighbour-joining (See Sec. 2.4.2) to infer a tree. But these methods had a number of drawbacks, which methods like maximum parsimony try to compensate(See Sec. 2.4.1). This method is based on the assumption that the most plausible hypothesis is to be preferred and searches for the tree that minimises the number of implied evolutionary changes (e.g. indels, substitutions). Since parsimony is not able to detect multiple substitution it faces the problem of long branch attraction - the phenomenon when rapidly evolving lineages are inferred to be closely related, regardless of their true evolutionary relationships (See (San02) for up-to-date discussion). The next development in phylogenetic methodology was the idea to apply the problem of inference of phylogenies to other statistical inference problems. With the postulation of probabilistic models of evolution by Felsenstein (Fel81) it became possible to compute the likelihood of a phylogenetic tree. The likelihood of a statistical model is defined as the probability of the observed data given the model. The idea of maximum likelihood methods (See Sec. 2.4.4.4) is that the model which makes the observed data most probable is to be preferred. There are a variety of different substitution models (See Sec. 2.4.3.1) - describing the process of changes in sequences - and methods to select the best model for further analysis. Bayesian methods (See Sec. 2.4.4.5) are very similar to maximum likelihood methods because the make use of the same probabilistic model (the likelihood, $P(data|model)$[1]). The equation

$$P(model|data) = \frac{P(data|model) \times P(models)}{P(data)}$$

is called Bayes' theorem. A strength of bayesian methods is its ability to include additional parameters such as branch lengths or substitution rates by using Markov chain Monte Carlo (MCMC) . Since heuristics are used to carry out phylogenetic analysis, it is no guaranteed to get the best result. Confidence for the result can be achieved using permutation test (e.g.Bootstrapping (See Sec. 2.7)).

1.3 The History of Sponges

Sponges are primitive, sessile, mostly marine, water dwelling, filter feeders . They pump water through their bodies to filter out particles of food matter. Sponges (their scientific name is: Porifera) represent the simplest of animals (metazoans). Sponges are exclusively aquatic (water dwelling), most marine, found from deepest oceans to the coast of the sea. They play an important role in many marine habitats but little is known about their diversity, biology and ecology as compared with most other

[1]This is the probability of the data, given the occurrence of a probabilistic model

animal groups. Sponges are able to actively pump up to 10 times their body volume per hour. That makes them the most efficient vacuum cleaners of the sea. During the late 19th century and the mid 20th century many publications have contributed to gain knowledge about sponges. This knowledge has helped to understand structural, physiological, and biochemical mechanisms of the sponges and provided answers to fundamental biological questions (e.g. the biosynthesis of chemicals, the evolution of eukaryotic immunology, cellular theory, etc.). About 7,000 species are currently known and grouped as Demospongiae, Hexactinallida, and Calcarea. Demospongiae (See Fig. 1.5 as an example for a sponge from the class Demospongiae) are by far the biggest group (90% of sponges are demosponges). Their skeletons are composed of spongin fibers and/or siliceous spicules (Demb).

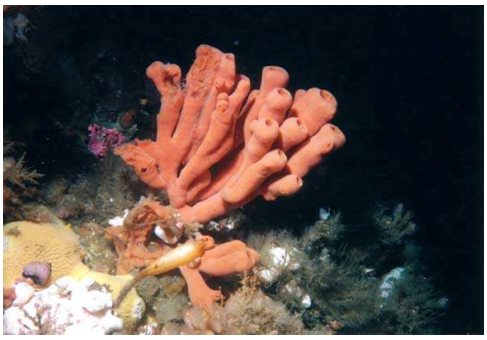

Figure 1.5: Orange Finger Sponge (Neoesperiopsis rigida (Demospongiae)). (Figure taken from (Dema).)

The hexactinellids (Glass sponges) are characterized by siliceous spicules consisting of six rays intersecting at right angles, much like a toy jack (Hex). Members of the group Calcarea are the only sponges that possess spicules composed of calcium carbonate. These spicules do not have hollow axial canals (Cal). Beyond their role as water cleaners and reef builders in the marine ecosysem, they are used as "bath sponges" (since early Greek civilization) or sources of therapeutic drugs. Scientist tried to classify sponges by dividing them into different classes. Gray was the first to use morphological markers to subdivided the sponges into the classes *Silicea* (Demospongiae + Hexactinellida) and *Calcarea* (See Fig. 1.6 (left part)). This distinction was maintained until *Reiswig* and *Mackie* subdivided the sponges into "Symplasma"(Hexactinellida) and "Cellularia"(Demospongiae + Calcarea). The use of molecular data (28S rDNA (LBEVC92) and 18S rDNA (BMA^{+}01)) showed paraphyly within the sponges, whereby *Calcarea* seemed to be closer related to other metazoans than to *Silicea* (See Fig. 1.6 (right part)). Further analysis supported the closer relationship of *Calcarea* to *Ctenophora* (Jellyfish). This is contrary to the former assumption of the monophyly of sponges, that means that the group of sponges consists of an inferred common ancestor and all its descendants as well. Medina et al used full-length 28S and 18S rDNA to find strong support for the clade (Demospongiae + Hexactinellida), but could not decide about the paraphyly of Porifera - the assumption that this group contains its most recent common ancestor, but does not contain all the descendants of that ancestor. Further investigations - with a larger *Taxon Sampling* - will hopefully reveal more about the history of the sponges.

1.4 Scope of this Thesis

The goal of this theis is to help to contribute to resolving the phylogeny of sponges. Several studies have revealed new insights in the history of sponges. Condflicting results from phylogenetic analyses based on 28S and 18S rDNA make the use of larger datasets necessary.

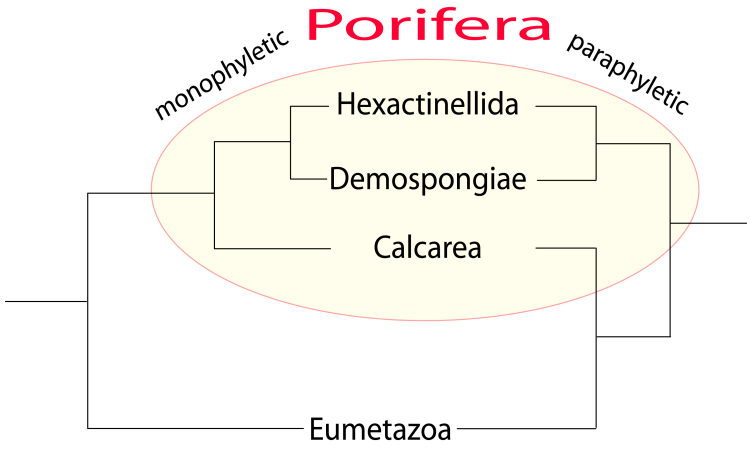

Figure 1.6: The two hypothesis about the history of phylum porifera. On the left side the hyphothesis that sponges are a monophyletic group. And on the right side the hypothesis that the subphylum "Calcarea" more closer related other eumetazoa like ctenophora or cnidara than it is to the two other poriferan classes.

This thesis tries to answer the question if the group of sponges is monophyletic or paraphyletic (See Fig. 1.6 (left)). The general opinion is that phylum porifera is a monophyletic group. The use of extended datasets (multiple gene sequences - instead of just one or a few genes) should give us an answer or at least a tendency if we can trust this hypothesis or not.

This dataset will be based on existing datasets from *Rokas* and *Baurain* (See Sec. 4). Both studied the phylogeny of metazoans using large datasets. We will use these datasets as a framework to carry out our analysis. Homology searches (using BLAST) will update the datasets by adding sponge sequences from public databases. The use of popular alignment methods and the selection of conserved regions from alignments afterwards will lead to high quality alignments ready for supermatrix or supertree analysis. The parallel use of several methods in each step of the analysis and the bootstrapping of the phylogenetic trees will measure the confidence in the data.

To automate the complete analysis a phylogeny pipeline will be developed to carry out all different steps of the analysis automatically. This also includes an iterative approach to estimate the right selection of parameters for the homology search.

Chapter 2

Current Methods in Phylogenetic Analysis

The study of the evolutionary history of a group of organisms involves different steps of analyses. In this chapter we would like to introduce the basics of phylogenetic analyses, starting with the search for homologous genes, the multiple sequence alignment, the automated improvement of the alignments and two approaches to reconstruct the phylogenetic tree

2.1 Search for Homologous Sequences

A single sequence of nucleotides or amino acids alone is not informative in the phylogenetic context, it has to be compared with to sequences to be able to make assumptions about its evolutionary history (SIM). This is usually done by comparing the sequence of interest to homologous sequences.

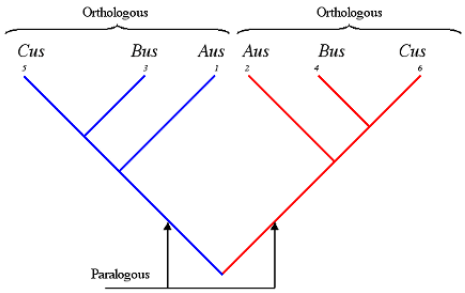

Figure 2.1: A gene is duplicated (at the root) to produce two paralogous genes. The process of speciation produces orthologous genes in the branches *Aus*, *Bus*, *Cus*. (Figure taken from (Hom).)

Homologous sequences are thought to share a common ancestor; i.e at some point of time in the evolutionary history, there was a protein which through processes of speciation or gene duplication produced two homologous proteins (See Fig. 2.1). In the case of speciation events the resulting genes are *orthologs*, in case of duplication events they are called *paralogs*[1].

An appropriate algorithm is needed to automatically find homologous sequences for a given sequence. This algorithm should take a (set of) sequence of interest and try to find similar sequences in databases. An example for such an algorithm is BLAST (See Sec. 2.1.2) and examples for popular databases are GenBank, dbEST or the Trace Archive which will be described in the next section.

[1]The focus of this thesis is the study of similarities of sequences on the amino acid level and not the similarity of functions, so no distinction between ortholog and paralog sequences will be made.

2.1.1 Molecular Databases

All molecular sequences as well as their annotations[2] and additional information (e.g. publications) are stored in databases. The three main primary databases (See Fig. 2.2) are DDBJ (DNA Data Bank of Japan) (DDB), EMBL Nucleotide DB (European Molecular Biology Laboratory) (EMB), and GenBank (National Centre for Biotechnology Information) (GENa). They are organised as the International Nucleotide Sequence Database Collaboration (INSD) (INS). Three useful databases

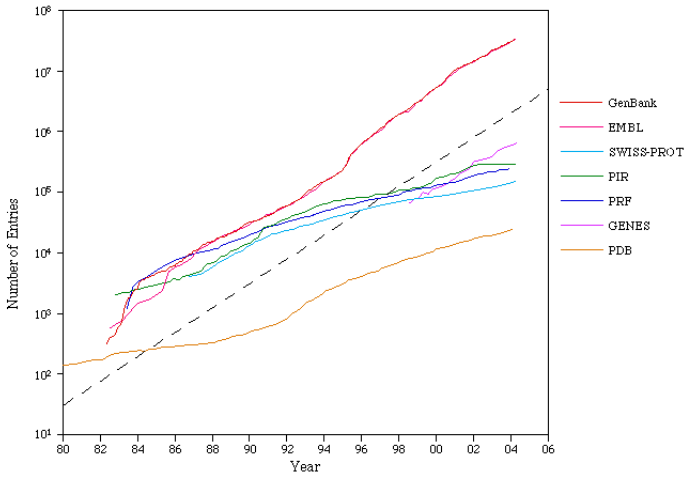

Figure 2.2: The growth statistics in number of sequences of the largest public databases. The years are on the horizontal, the number of entries in the databases are on the vertical axes. (Figure taken from (The).)

for phylogenetic analysis are *GenBank*, *dbEST* and the *Trace Archive* hosted by the NCBI. These databases contain protein and EST sequences and Trace files. Protein sequences (See Sec. 2.1.1.1) are the first and best choice in the process of finding homologous sequences, because are usually annotated and give scientist a high degree of confidence with homology searches. Additionally, EST databases (See Sec. 2.1.1.2) and the Trace Archive (See Sec. 2.1.1.3) can be searched to increase the amount of useful data for the phylogenetic study.

2.1.1.1 GenBank

Genbank is a public database for nucleotide and protein sequences, supporting bibliographic and biological annotations (BBL+98). It was launched in 1982 by Walter Goad and colleagues[3] and is maintained by the NCBI. Sequence data are submitted from authors, sequencing centers, the US Office of Patents and Trademarks (USPTO) [4] and daily exchanged with DDBJ and EMBL. Sequence data from GenBank can be obtained in three possible ways:
The ENTREZ System is a database retrieval system that has access to over 20 biological databases containing DNA and protein sequences and their annotations and related information[5].

[2] additional information about the sequence. e.g. function of a gene, author who submitted the sequence, etc
[3] `http://www.ncbi.nlm.nih.gov/`
[4] `http://www.uspto.gov/`
[5] `http://www.ncbi.nlm.nih.gov/Entrez/`

BLAST is a set of programs to find local similarities between a query sequence and database sequences[6] (See 2.1.2).
FTP is the third possibility of retrieving data from GenBank. The NCBI offers the full bimonthly GenBank release in different file formats[7].

2.1.1.2 dbEST

With the beginning of the high-throughput sequencing era , initiated 1991 by Venter and colleagues (HAN), great interest has been put in a complete gene list for an organism. With this list researchers should be able to broaden the knowledge about biochemical pathways, which would help in e.g. drug design. One of the first steps after sequencing an organism is the collection of cDNA[8] to identify new genes. cDNA-clones are randomly chosen. They are short parts of the sequences of both ends of the inserts (See Fig. 2.3). These sequences are called expressed sequence tags (EST). They are very short

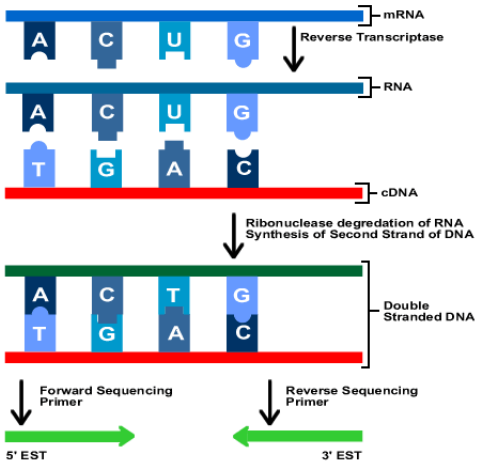

Figure 2.3: The process of *EST* generation. A single-stranded mRNA sequence is first reverse transcribed into a doubled-stranded cDNA sequence. Short parts of both ends of the complementary strand are then sequenced. (Figure taken from (NCB).)

(400-600 bases), relatively inaccurate (2% error), and identified by comparing them to known genes or other EST sequences. By comparing the EST sequences to known gene and assigning a putative function they become useful in the search for homologous sequences for phylogenetic studies In 1992 *GenBank* established its own EST database, called *dbEST* (BLT93). It currently includes 42,050,137 entries[9].

2.1.1.3 Trace Archive

The Trace Archive provides an database for DNA sequencing reads, associated Traces, and quality values. These data come from whole-genome shotgun projects, EST projects, and other large-scale sequencing projects.(Trab)

[6] http://www.ncbi.nlm.nih.gov/BLAST/
[7] `ftp.ncbi.nih.gov`
[8] complementary DNA (cDNA) is DNA synthesized from a mature mRNA template
[9] dbEST release 030907

Table 2.1: The five organisms with the most EST sequences available in *dbEST*

Organism	Number of ESTs
Homo sapiens (human)	4,070,035
Mus musculus (mouse)	2,522,776
Rattus norvegicus (rat)	326,707
Drosophila melanogaster (fruit fly)	255,456
Glycine max (soybean)	234,900

.

Traces are primary data from sequencing machines including quality values (probability of a base being in error) for each base and ancillary information (like source DNA, size of insert, etc.) (BO05). The Trace Archive is also hosted at NCBI. It currently contains 1,521,222,251 entries from 800 organism[10](dbE)

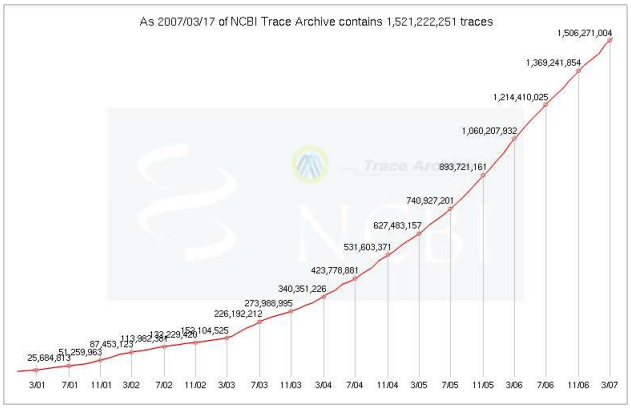

Figure 2.4: The increase in number of sequences over the last 6 years in the Trace Archive. (Figure taken from (Traa).)

2.1.2 BLAST - how to search databases

The number of available sequences rapidly increases (See Fig. 2.5)- mainly because of ongoing genome sequencing project - automated searches for homologs in sequence databases (e.g. *BLAST*) and the following alignment of best hits from these searches are becoming a standard technique.

A quick way of obtaining a wealth (depending on the e-value) of similar sequences is the use of comparison algorithms such as *BLAST*[11] (AGM$^+$90), an approximation of the Smith-Waterman algorithm (See Sec. 2.3.1.1).

BLAST contains a set of programs for different types of similarity searches (See Fig. 2.6). The programs differ in the type of query sequence (nucleotide or protein), the type of database to be searched (nucleotide, translated nucleotide or protein), and the method to be used (standard or iterative). The BLAST searches databases for local optimal local alignments (See Sec. 2.3.1.1). BLAST starts to search for words of length W with a score of at least T when compared to the query sequence using a substitution matrix (See Sec. 2.4.3.1). Since proteins consist of functional domains that are repeated

[10] 2007/03/17

[11] Although BLAST is a set of programs, this thesis refers to BLAST as only one program

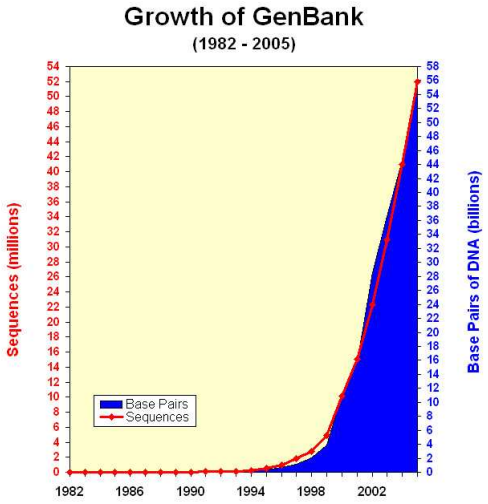

Figure 2.5: Increase in number of sequences from 1985 - 2005 at Genbank(NCBI). The years are on the horizontal axes, the number of sequences in millions (left) and the number of base pairs in billions (right) are on the vertical axes. (Figure taken from (GENb).)

within the same protein and across different proteins from different species, its best to use a local alignment tool to find these short streches ("word") rather than complete matching sequences.
The word hits are then extended - without introducing gaps - to try to generate an alignment with a score of at least S. Parameter T ist mainly important for the speed and sensitivity of the algorithm. The results of *BLAST* are presented as Figure 2.7 shows. Each Alignment is given an Score S and an expectation value (e-value), that is "the number of different alignments with scores equivalent to or better than S that are expected to occur in a database search by chance. The lower the e-value, the more significant the score" (E-V). The e-value is defined as

$$E = Kmn\, e^{\lambda S}$$

where K and λ represent natural scales for the search space and the scoring system. m is the size of query sequence, n the size of the database, and S the score of the alignment. The e-value is first indicator for a sequence being homologous to the query sequence or not (int):

- E < e-100: identical sequence (same gene/protein and organism)
- e-50 < E < e-100: almost identical sequence
- e-10 < E < e-50: closely related sequence
- 1 < E < e-5: could be a true homologue
- E > 1: most likely not related

Remember that the e-value is just an indicator for homology, but not a guarantee. So each alignment should be viewed with a critical eye in order to declare it as useful for the further analysis or not.

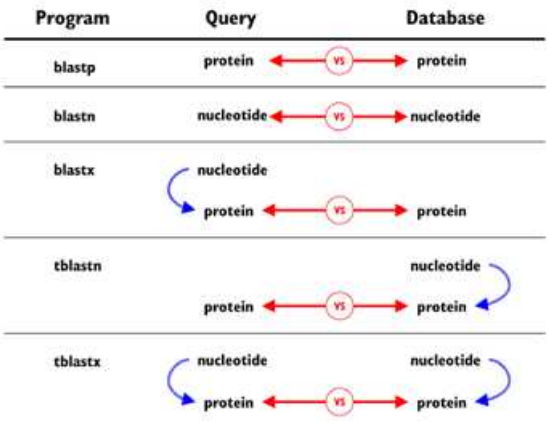

Figure 2.6: The different *BLAST* programs available for similarity searching. The horizontal arrows indicate the query of a database using a query sequence. The other arrows indicate the translation of a nucleotide sequence into a protein sequence. (Figure taken from (BLAa).)

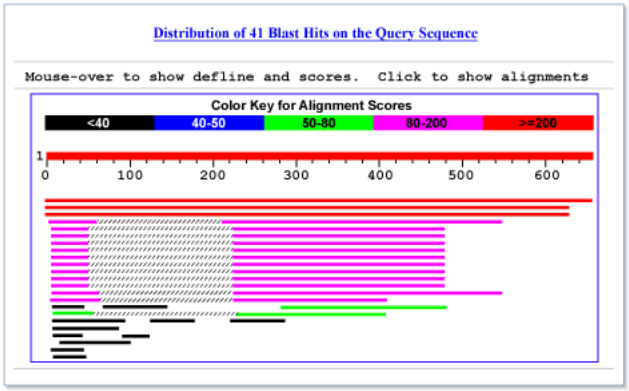

Figure 2.7: The query sequence is represented by the numbered red bar at the top. Database hits are shown aligned to the query, below the red bar. Of the aligned sequences, the most similar are shown closest to the query. In this case, there are three high-scoring database matches that align to most of the query sequence. (Figure taken from (BLAb).)

2.2 Taxon Sampling

The amount of sequence data has increased significantly during the past decade (Figure 2.5). This is the consequence of new techniques such as the polymerase chain reaction (PCR) or faster sequencing methods (e.g. 454 Sequencing).

This wealth of data has increased the number of phylogenetic analysis and led to new challenges and research areas. One such area is called *Taxon Sampling*. The term *Taxon Sampling* was first used by *Hasegewa* (HKY85) and *Hillis* introduced it for the use in phylogenetic analyses (Hil96). Taxon sampling is the process of choosing which taxa / characters to include and which not in phylogenetic studies. Since methods such as sequence alignment and the reconstruction of phylogenetic trees are based on the initial set of homologous genes, *Taxon Sampling* is of extreme importance for the success of every phylogenetic analysis. A selection of inappropriate taxa / characters can lead to misleading estimations of phylogeny and should be taken very seriously.

Taxon sampling is "driven by resource limitation" (RK03). This means that after the search for homologous a given data matrix - with the taxa vertically and the characters horizontally lined up - would contain many empty cells (no data or taxon available) (See Fig. 2.9). The taxonomic sampling approach favors the use of large datasets - no matter if they contain empty cells or not -, because large datasets will reduce the chance of running into trees with very long branches (e.g. those in the *Felsenstein* zone (See Fig. 2.8)). Otherwise, the delection of taxa / characters reduces the

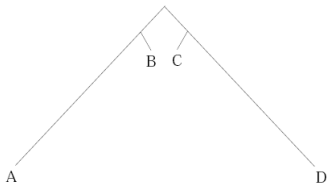

Figure 2.8: The effect of concurrent evolution along multiple branches. This can cause sequences to appear very close, although they are very divert. Trees with this structure are said to be in the *Felsenstein zone*. (Figure adopted from (Fel).)

computational burden, simplifies the inference process, and decreases the effect of error propagation in the inferred tree.

Countless empirical studies have been published about this topic in last decade and led to a heated debate among the supporters and opponents. The following two quotations summarize the different opinions about *Taxon Sampling* for phylogenetic analyses:

> *"If the evolutionary question of interest does not require a large number of taxa, it seems best to use fewer taxa because larger trees are more likely to contain inconsistent branches".* (Kim96)

> *"Including large number of taxa in an analysis may be the best way to ensure phylogenetic accuracy".* (Hil96)

Scientists have different opinions about the right selection of taxa. Therefore, no rule is available for this *Taxon Sampling* problem. In the following we will take a closer look at how to handle this situation.

Consider the situation of (for example) 10 taxa based on a combined analysis of two genes regions. Five taxa are lacking data for the second gene as in Figure 2.9. The corresponding entries in the data matrix are coded as either missing or unknown ('?').

A researcher might choose to deal with this situation by deleting these taxa, deleting characters 3 and 4, or by simply including all the characters and taxa (taxonomic sampling approach). The first two options are based on the implicit assumption that including these five taxa and characters 3 and 4 will somehow be problematic because of the effects of missing data.

2.2.1 The Taxonomic Sampling Approach: Including all characters and taxa

Although sequencing methods have been improved and many genes from different organisms have been sequenced, the "sampling of genes and taxa for a given group of organisms [...] still is quite

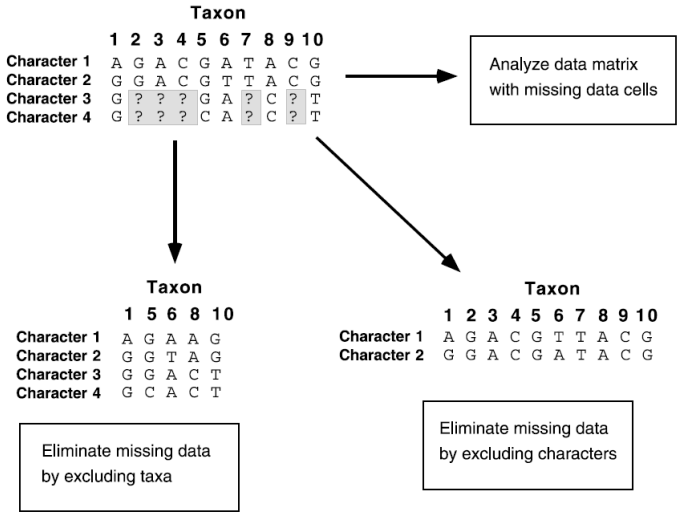

Figure 2.9: Missing data in phylogenetic analysis. Taxa two, three, four, seven, and nine lack data for characters 3 and 4. The researcher can deal with this situation by excluding problematic data or including all data. (Figure taken from (Wie06).)

sparse" (SDR+03). For most taxa only a small proportion of sequences is available, so a combination of available gene sequences for any taxonomic group would led to the situation of a data matrix with many empty cells (See Fig. 2.9). A naive solution would be to exclude all taxa with missing data from the data matrix. But the fear of eliminating taxa / characters and with it (important) phylogenetic signal remains. Since eliminating missing data cells also means eliminating non-missing data, its hard to balance between the benefit of excluding missing data cells and the cost of excluding taxa / characters. A possible approach to a solution was the development of the supertree approach (See Sec. 2.6) that combines the information (topologies) from different source trees to one single tree.

2.2.1.1 Adding taxa or characters

The general assumption in many fields of science is that more data lead to a stronger supported hypothesis. This is only partly true for the field of phylogentic analysis, because there are special cases where the opposite is true, e.g. the Felsenstein Zone (also called *long branch attraction* (LBA)) (See Fig. 2.8). In these cases adding taxa increases the probability to estimate the wrong tree (HP89). In general there is no consensus about the benefit of adding taxa or characters. Different authors yield different hypotheses although each one is well founded wihtin its individual study[12]. One general statement can give a idea on how to handle this: "most accurate reconstruction are based on a large amount of characters - if the data is of high quality" (NEK99).

2.2.1.2 Adding both taxa and characters

Adding taxa and characters lead to an increase of the size of the data matrix. This step will certainly also increase the comlexity and with it the running time of phylogeny methods, but a positive effect

[12]See (Wie06) for more information

that small datasets - where crucial data are absent and lead to erroneous results could be reached - can be avoided.

2.2.1.3 The Impact of Incomplete Taxa: Deleting taxa and characters

Before incomplete taxa[13] are removed from further analysis, its important to take a closer look at the impact of these data on phylogenetic analyses.

There are good reasons for not excluding these data. Simulations showed that there is little support for excluding dubious taxa based on the amount of missing data they bear. It has been shown that incomplete taxa can be accurately placed in a phylogeny (through simulations) and with strong statistical support (through emperical analyses). They can even increase the phylogenetic accuracy for complete taxa. This has been shown by (Wie06). He also proved that the essential condition for the correct placing in the phylogeny is the amount of characters present and not those that are absent.

Usually there is no knowledge about the "true" tree and therefore no justification for making an a priori decision on what data is reliable or not. All data can be seen as evidence of evolution. But if there is detailed knowledge about the taxa, an a posteriori approach can be used to compare with an expected result. This can lead to the exclusion of taxa if a known monophyly of a clade is violated in the inferred tree (NC96).

[13] $> 75\%$ missing data

2.3 Sequence Alignment

Whether the aim is the phylogenetic analysis of several homologous, the identification of motifs or patterns, or structural modelling, multiple sequence alignments give the researcher the possibility go gather more information than a single sequence can offer. The goal of the sequence alignment problem is to take a set of amino acid (Protein) or nucleotide (DNA/RNA) sequences and arrange them in a way to minimise/maximise a given cost function. If two sequences are to be aligned, we speak of pairwise sequence alignment and multiple sequence alignment with more than two sequences.

2.3.1 Pairwise Sequence Alignment

Given two sequences, an optimal alignment can be computed using dynamic programming. The two most popular approaches calculate global and local alignments (See Fig. 2.10).

```
Global FTFTALILLAVAV
       F--TAL-LLA-AV

Local  FTFTALILL-AVAV
       --FTAL-LLAAV--
```

Figure 2.10: The difference between global and local alignments.

2.3.1.1 Needleman-Wunsch

The Needleman-Wunsch algorithms has been first described in 1970 by Needleman and Wunsch (NW70). This approach tries to maximise a *similarity score* by computing global alignments. The algorithm computes a global alignment in three steps:

- A matrix represents all possible pairings of two sequences and their similarity with a similarity score. This similarity score can be based on biochemical or evolutionary information.
- A path trough the matrix starts in the upper-left corner and ends in the lower-right corner. At every step of the path the algorithm tries to find the best alignment that ends there.
- The best alignment is the alignment with the highest total score.

2.3.1.2 Smith-Waterman

The Smith-Waterman algorithm, proposed by Smith and Waterman in 1981 (SWF81), is based on the Needleman-Wunsch algorithm, but in contrast it calculates a local alignment. The algorithm allows to align subsequences (with all possible lengths) to find local similarities between sequences. Additionally to the *similarity score* a *gap penalty* - for occurrences of gaps in the alignment - is subtracted from the total score.

2.3.2 Multiple Sequence Alignment (MSA)

The Needleman-Wunsch algorithm was originally designed for only two sequences. In principle, it is possible to extend the Needleman-Wunsch algorithm to deal with more than two sequences. But since

the number of cells in a multiple alignment matrix growth exponentially with the number of sequences and their lengths, simultaneous alignments for more than a few sequences are a big computational problem. The use of heuristics makes tries the solve the problem of MSA.

2.3.3 Example of an multiple sequence alignment

Figure 2.11 (upper part) shows a way of looking at problem of multiple sequence alignments. These sequences represent only parts of protein sequences. The Symbols correspond to amino acids. Figure

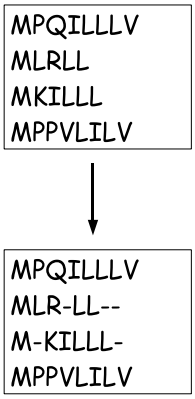

Figure 2.11: Example: Protein sequences before (upper part) and after (lower part) the alignment step.

2.11 (lower part) now shows a possible alignment for the given set of protein sequences. All sequences have the same length. This has been achieved by introducing gaps ('-') to put similar parts of the sequences in the same column (and therefore minimise a given cost function).

2.3.4 Basics

Definition 1 (Multiple Sequence Alignment, MSA)

Let $\sum$ be a finite alphabet without gap ('-') and $\sum' = \sum \cup \{'-'\}$. Let further be $s_1, \ldots, s_k$ k sequences over $\sum$ with lengths $l_1, \ldots, l_k$. A (global) multiple alignment A of $s_1, \ldots, s_k$ is a matrix of dimension $k \times l$ with the following constraints:

- $max(l_1, \ldots, l_k) \leq l \leq \sum_{i=1}^{k} l_i$
- $A[i][j] \in \sum' \forall\, 1 \leq i \leq k,\ 1 \leq j \leq l$
- *For each two symbols of a sequence $s_{k,i}, s_{k,j}$ with $i \leq j$ the relative order throughout the alignment process stays the same. If their new positions in the MSA are $s_{k,i'}$ and $s_{k,j'}$, then $i' \leq j'$*

- *No column consists of only gaps*
- *The number of sequences is $k \geq 2$ ($k = 2$ is a pairwise alignment)*

Definition 2 (Optimal Alignment) *An optimal Alignment is the one that minimises/maximises a given cost function. One such function is the Sum of Pairs Score. It is defined as follows:*

$$SP = \sum_{h=1}^{l} \sum_{(i,j,i<j)} c(s_{i,h}, s_{j,h})$$

with:

- *$s_{i,h}$ represents elements of the matrix;*
- *c: $\sum' \times \sum' \rightarrow \mathbf{R}$ is a cost function for pairs of symbols;*
- *$c(-,-) = 0$;*

It can be shown that the multiple sequence alignments with the *Sum of Pairs Score* is a NP-complete problem (BV01). To reduce the computational complexity many approaches to the MSA problem are based on pairwise sequence alignments.

2.3.5 Overview

Many approaches to solving the MSA problem have been developed in the past years. According to *Notredame* (Not02) these approaches can be grouped as follows:

- *Progressive algorithms:* These class of algorithms is one of the easiest and most effective ways to solve the MSA problem. Sequences are added one-by-one to a MSA with the progressive method. The order of addition of the sequences is given by a precomputed criteria, such as the similarity of every two sequences. A popular representative of this class is *ClustalW* (See Sec. 2.3.5.1.
- *Exact algorithms:* In contrast to the three other classes, exact algorithms always calculate an optimal alignment. But the fact that the time complexity increases exponentially with increasing number of sequences, the number of input sequences is drastically limited. A member of this class is the Branch-and-Bound approach (Kec93).
- *Iterative algorithms:* Iterative algorithms start with an alignment and tries to refine it until no improvement can be achieved. This class can be subdivided into algorithms using *Simulated Annealing*, *genetic algorithms* or similar (*stochastic iterative approaches*) and algorithms based on dynamic programming (*non-stochastic iterative approaches*). *Muscle* is a member of this class (See Sec. 2.3.5.4).
- *Consistency-based algorithms:* Algorithms of this class try to use independent observations in a way that keeps them consistent. Pairwise alignments are build and a solution is generated (with an affection to consistent subalignments), that resembles the pairwise alignments most. A mixture of progressive and consistency-based algorithms is T-Coffee (See Sec. 2.3.5.2).

2.3.5.1 ClustalW

ClustalW is a widely used multiple sequence alignment program and was first described by Thompson in 1994 (THG94). The basic algorithm of Clustalw consists of three parts, following the basic progressive alignment procedure:

- **Pairing of sequences to form a distance matrix**
  Pairwise distances for all sequences are calculated using standard methods for pairwise alignments. The score for the pairwise alignment is calculated using a word method described in Sec. 2.4.2 minus a penalty for gaps. Scores are organised in an $n \times n$ table to represent similarities between the sequences.

- **Using the distance matrix to calculate a guide tree**
  With the distance matrix as an input, a phylogenetic tree is created using the neighbour-joining method (See Sec. 2.4.2). This tree is called guide tree, because it will guide through the process of aligning the sequences in the next step. The primary resulting tree is unrooted and but will be rooted so that also a weight for each sequence can be derived.

- **Progressive alignment of sequences according to the guide tree**
  The branching order of the guide tree is used to align sequences according to the position of the sequence in the tree. The method starts from the leaves of the rooted tree and works until it reaches the root itself. At each step a pairwise alignment method (See Sec. 2.3.1) calculates the score using a substitution matrix and a penalty for gaps. Once gaps have been introduced at an early stage they stay fixed.

2.3.5.2 T-Coffee

T-Coffee (Tree-based Consistency Objective Function for alignment Evaluation) is a method mainly based on a progressive alignment and was introduced in 2000 by *Notredame* (Not02).
Disadvantages of *ClustalW* are that a once introduced gap could not be removed throughout the alignment process and that it used global alignments in to align sequences.

T-Coffee computes local and global alignments and summarizes them in a library. The global information come from *ClustalW*, the local from *Lalign*, a variant of the *Smith-Waterman* algorithm, The library is then extended by comparing every pairwise alignment to get information to guide through the alignment process. A guide tree is then constructed and used to carry out the multiple alignment process.

2.3.5.3 DiAlign

DiAlign, first described in 1997 by Morgenstern (MFDW98), is another alignment method that tries to use local information to guide a global alignment.
DiAlign extends the residue-by-residue comparison of previous alignment programs by taking whole segments - that are uninterrupted stretches of residues - into consideration.

The program DiAlign constructs alignments from gapfree pairs of similar segments of the sequences. Such segment pairs are referred to as diagonals.

Every possible diagonal is given a so-called weight reflecting the degree of similarity among the two segments involved. The overall score of an alignment is then defined as the sum of weights of the diagonals it consists of and the program finds an alignment with a maximum score – in other words: the program tries to find a consistent collection of diagonals with a maximum sum of weights. For the multiple alignment a greedy approach is used. DiAlign is especially suited to detect local similarities in otherwise completely unrelated sequences.

2.3.5.4 Muscle

In 2004 Robert C. Edgar introduced Muscle, a "multi sequence alignment with high accuracy and high throughput" (Edg04).
The algorithm, implemented in Muscle, involves a fast estimation of distances using a k-mer[14] counting[15] to reduce the computation time, a progressive alignment with a new object function based on profiles, which is called "log-expectation score" and refining the current alignment with tree-based partitioning.

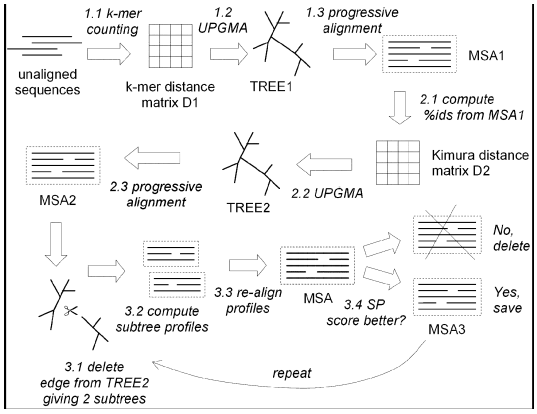

Figure 2.12: This diagram summaries the flow of the MUSCLE algorithm. There are three main stages: Stage 1 (draft progressive), Stage 2 (improved progressive) and Stage 3 (refinement). A multiple alignment is available at the completion of each stage, at which point the algorithm may terminate. (Figure taken from (Edg04).)

The muscle algorithm is described as follows (See Fig. 2.12):

1. **Distance measures and guide tree estimation**
   Similarities between the input sequences are computed using k-mer distances, a contiguous subsequence of length k. This approach for counting similarities is used because the frequency of k-mers in related sequences is higher than by chance. The resulting matrix is then clustered by UPGMA (See Sec. 2.4.2) and a first multiple alignment is constructed according to the branching order of the UPGMA-tree.

2. **Profile alignment**
   As the use of the K-mer distance can be a cause for errors, the tree is re-estimated using the Kimura distance[16]. For each pair of sequences from MSA1 the Kimura distance is computed, the

[14] A k-mer is a contiguous subsequence of length k
[15] in contrast to the pairwise alignment other methods use
[16] similar to the K-mer distance, but uses additional information from alignment sources

resulting matrix (D2) is again clustered by UPGMA and the resulting UPGMA-tree (TREE2) is used as a guide tree to carry out a progressive alignment (MSA2).

3. **Refinement**
   TREE2 is divided into two subtrees by deleting the edge with the least distance to the root. A profile of each multiple alignment in the subtree is generated. Realigning the two profiles produces a new multiple alignment, whose score is compared to the score of the previous multiple alignment (MSA2) in order to decide if the score is improved and the current alignment is to be kept or redo the refinement steps until convergence or a user-defined threshold is reached.

2.3.6 Selection of Conserved Blocks from Multiple Alignments

The use of multiple sequence alignments in phylogenetic analysis, particularly those that are not very well conserved, requires the elimination of poorly aligned positions and divergent regions, since they may not be homologous or may have been saturated by multiple substitutions (Cas00). Not all alignments are useful for phylogenetic reconstruction. Sequences that are too similar therefore include almost no phylogenetic signal[17] or sequences that are so divert that they might contain multiple substitutions and are not useful to build a reliable phylogenetic tree. Usually not all parts of a gene evolve at the same rate, some parts evolve faster others slower. Parts evolving slower are well (usually) conserved (e.g. functional domains) and therefore suitable for further analysis, whereas faster-evolving parts are less conserved, full of gaps, and too divert to be included in further analysis. The latter ones can be excluded (Lak91; OW93; SOWH96). Optimising the alignment is an important step, because "it has been shown that the alignment method may have more impact than does the type of tree-building method used" (ME97).

Gblocks (Cas00) is a method that is capable of optimising multiple alignments. It defines conserved blocks in a multiple alignment according to a set of requirements and some thresholds, *IF, FS, CP, BL1, BL2*, and excludes the rest of the alignment from further analysis. Gblocks defines several criteria

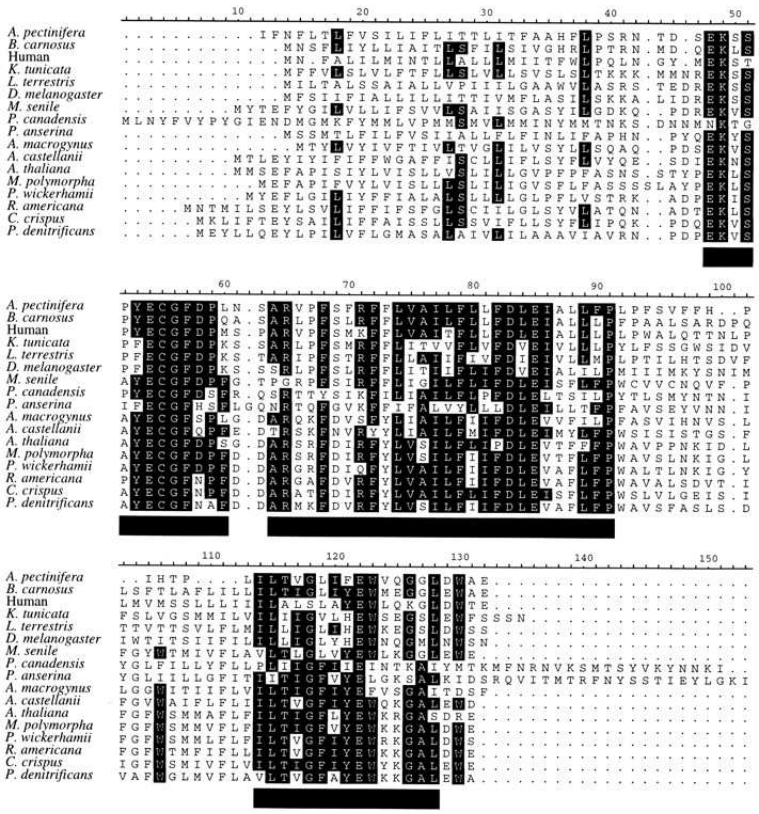

Figure 2.13: Alignment of ND3 sequences from several eukaryotes and a bacterial outgroup with the blocks selected by the Gblocks program with default parameters underlined. Positions at which more than 50% of the residues are identical and have no gaps are shaded. (Figure taken from (Cas00).)

for selecting a region of the alignment as a conserved block.

[17] that are differences between sequences be they on nucleic or amino acid level

1. Positions are classified into three types: non-consevered ($\leq$ IS identical residues or there is a gap), conserved ($\geq IS$ and $\leq FS$ identical residues) and highly-conserved ($\geq$FS identical residues).
2. Long ($> CP$),contiguous and non-conserved regions are rejected, because these regions are usually ambiguous.
3. Flanks of the remaining regions regions are examined and positions are removed until all highly conserved regions are adjacent.
4. Conserved regions with length $\leq$ *BL1* are also rejected, because the quality of an alignment is hard to assess in smaller regions.
5. Positions with gaps and neighboring non-conserved positions are removed until a conserved region is reached.
6. In the end remaining small blocks are kept unless they are $\leq$ *BL2*.

The defined blocks are then concatenated to an alignment and can be used for the further analysis.

2.4 Methods of Phylogenetic Estimation

A fundamental problem in computational biology is the reconstruction of evolutionary tree (also called *phylogenetic trees*). According to (HL03)) Methods trying to solve this problem can help in study of evolutionary processes: Detection of orthology and paralogy, estimation of divergence times, the Reconstruction of ancient proteins, the Finding of residues important to *Natural Selection*, the Detection of recombinant points, the identification of mutations likely to be linked to diseases, and the Determining of identity of new pathogens.

The problem of reconstructing phylogeny can be formulated as a computer science problem on binary trees (adopted from (KW99)).

Definition 3 (A Phylogenetic Tree) *A phylogenetic tree $T = (V, E)$ is a connected, acyclic graph with a set of vertices V, a set of edges E. A rooted phylogenetic binary (also called bifurcating) tree is a directed tree with a unique node corresponding to the most recent common ancestor of all entities on the leaves[18]. Each internal node has exactly two children, every edge $e \in E$ is labelled with a positive real number $|e|$, called its length, and each leaf is labelled with a taxon.*

Current evidence shows that simultaneous speciation is quite rare, therefore, one is able to approximately describe most phylogenetic relationship using binary trees. All methods describe in this thesis can be extended to deal with multifurcating trees (where the degree of an internal node can be ≥ 3). The problem of phylogenetic reconstruction can be stated as:

Definition 4 (The Problem of Phylogenetic Reconstruction) *Let $T = t_1, \ldots, t_n$ be a set of taxa and a sequence s_i corresponding to each taxon t_i. The problem is to find a phylogenetic tree with leaves labelled $t_1, \ldots, t_n$ that fits the data best.*

In order to decide which tree fits the data best, some criteria are needed to decide why one tree is "better" than another tree. Criteria like *parsimony*, *maximum likelihood*, or *Bayesian inference* are able to decide this question using different assumptions. Methods using this criteria belong to the class of NP-hard problems and therefore, are computationally intractable (KW99). The use of heuristics allows to find a solution for the problem of phylogenetic reconstruction using some criteria. There are three classes of heuristics known. These are *parsimony*, *distance-based* and *statistical* methods.

2.4.1 Parsimony

The general idea of parsimony analysis is that a evolutionary tree is to be preferred that involves "the minimum net amount of evolution" (ECS64). It means that the most plausible phylogeny (for a given dataset) is the one which requires as few evolutionary events (e.g. insertions, deletions,..) as possible. Parsimony implies that simpler hypotheses are preferable to more complicated ones. A parsimony analysis run is straightforward. The different trees (in tree space) are scored according to the number of evolutionary changes they require. That means that the algorithm counts the number of evolutionary transitions required to explain the distribution of a character. The most parsimonious

[18] An unrooted tree is undirected and makes no assumptions about ancestry

tree for a given dataset is seen as the preferred hypothesis of relationship among the taxa in the analysis.

Parsimony methods make use of methods for searching the tree space (See Sec. 2.4.4) to reduce complexity. An advantage of parsimony methods over other methods, is its running time, a disadvantage is that it assumes low rates in character chance which can lead to long branch attraction (HL03).

2.4.2 Distance-based Methods

The main idea about distance-based approaches is that relatedness between the data can be first transformed into a distance matrix representing the dissimilarities between each input sequence. These distance matrix can be computed using a specific model of evolution (See Sec. 2.4.3.1). Various algorithms can then be used to determine the best tree given the distance matrix. The three most common approaches are UPGMA (SM58), neighbour-joining (SN87) and minimum evolution (DG04). An advantage of the distance-based methods is that they are fast. A major drawback is that a inappropriate selection of a model of evolution for the construction of the distance matrix can affect the tree topology (DeB92). Distance-based methods are often used to estimate a guide tree for multiple sequence alignments (See Sec. 2.3) or a starting tree for further analysis (e.g.maximum likelihood).

2.4.3 Statistical Methods

Statistical methods use of the likelihood criterion to measure the probability of different trees (in tree space) given a tree and an substitution model (Fel81). The goal in methods under a maximum likehood (ML) framework is to find the tree and evolutionary model that maximise the this probability.

2.4.3.1 Amino Acid Substitution Models

Substitution Models describe the process of changes from characters in one state to another state. The divergence among sequences can be modeled with a mutation matrix. This matrix (M) contains the probabilities that an amino acid mutates. The i-th row and the j-th column contains the probability that amino acid i (aa_i) changes to amino acid j (aa_j)

$$M_{ij} = P(aa_i \rightarrow aa_j)$$

This corresponds to a model of evolution assuming that amino acid mutations occur randomly and independently, but with some predefined probability. This is a Markovian model of evolution. The predifined properties depend on the physico-chemical properties (hydrophobicity, size, charge, etc.) of the amino acids and can be modeled as a matrix. Amino acids appear with different frequencies in nature. These are denoted by M_i, where M_i is a vector of the Markov state matrix M Since this model is symmetric, the propability of amino acid i mutating to amino acid j is the same as amino acid j mutating to amino acid i.

$$M_{ij} = M_{ji}$$

Examples for widely used substitution models are Dayhoff (DSO78), BLOSUM, and CAT (CAT).

2.4.4 Searching the Tree Space

Inferring phylogenetic trees is a computational challenge (CHHP00). The number of unrooted binary trees for n taxa is

$$t(n) = \frac{(2n-5)!}{2^{n-3}(n-3)!}$$

where t is the number of possible trees (Fel04). For example there are 2.8×10^{76} possible trees for a dataset of 50 taxa. This number is growing exponentially with increasing number of taxa. The problem of finding the optimal tree is computational intractable (See Tab. 2.2).

Table 2.2: This table shows the increase in number of possible trees with the increasing number of taxa

Number of Taxa	Number of unrooted trees	Number of rooted trees
2	1	1
3	1	3
4	3	15
5	15	105
6	105	945
7	954	10.395
8	10.395	135.135
9	135.135	34.459.425
10	34.459.425	2.13E15
15	2.13E15	8.E21

Heuristics can be used to make the computation feasible, but they do not guarantee to find the optimal tree (CHHP00). Three popular heuristics in this context are listed below.

2.4.4.1 Nearest Neighbour Interchange (NNI)

The NNI uses an optimisation method called hill-climbing to search for the best tree in the tree space. The basic idea is to define a neighbourhood criterion for trees and use a heuristic algorithm (e.g. the greedy algorithm) to find a tree given by a local maximum.

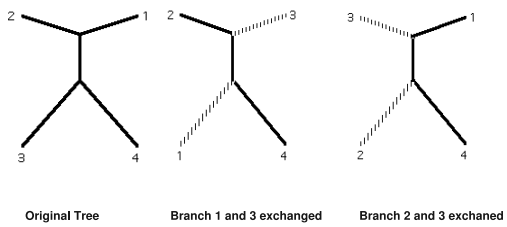

Figure 2.14: The NNI algorithm. Branches are exchanged to yield new trees. (Figure taken and changed from (HyPa).)

The neighbour area is the set of trees that can be reached by exchanging branches in the current tree[19] (See Fig. 2.14). The tree with the highest likelihood is selected as the best local tree and the procedure is repeated until no better tree can be found.

2.4.4.2 Subtree Prune and Regraft (SPR)

SPR is similar to NNI, but uses a more extensive rearrangement algorithm. It cuts subtrees and pastes them in distant parts of the tree (Kea06).

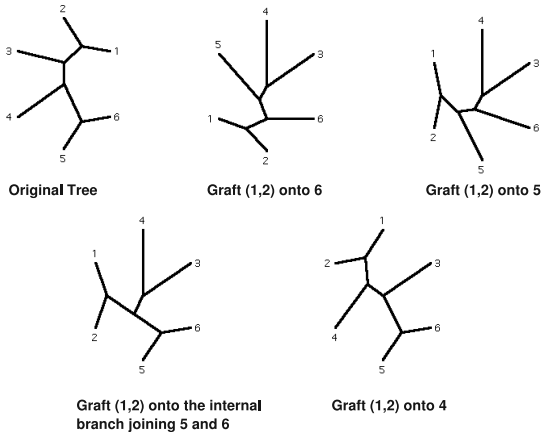

Figure 2.15: The selection of the subtree with leaves 1 and 2 from the original tree and the 4 possibilities to paste it. (Figure taken and changed from (HyPb).)

For example, given a tree as in 2.15, the subtree with leaves 1 and 2 can be selected from the original tree. Now there are 4 possibilities to paste the subtree resulting in a new tree. These possibilities are branch 4, 5, 6, and the internal branch joining 5 and 6. The tree with the highest score is taken as the local best tree. Again this procedure is repeated until no further improvement can be achieved.

2.4.4.3 Tree-Bisection-Reconnection (TBR)

The third algorithm presented introduces an even more extensive rearrangement method. It breaks internal branches and treats the resulting subtrees as independent trees. All possible connections between the two trees are examined to find the best tree (See Fig. 2.16). This method is more robust than NNI and SPR, because it tries to find a global optimal tree.

2.4.4.4 Maximum Likelihood

Maximum Likelihood is a statistical method for the inference of phylogeny. The first method to calculate a likelihood of a tree was given by Felsenstein (Fel81). The general idea is that the likelihood is the probability of seeing the observed data (D, e.g. an alignment) given a model of evolution (T).

$$P(D|T)$$

[19]The starting tree or the local best tree

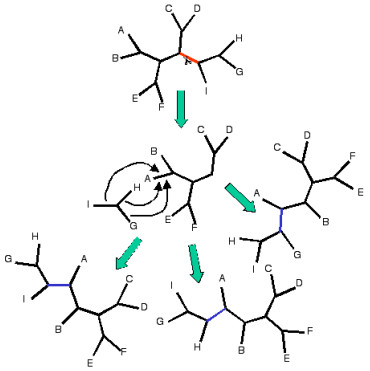

Figure 2.16: The Tree Bisection and Reconnection method. The original tree is broken apart and the two resulting subtrees are connected to yield the best score. (Figure taken from (Phyb).)

Input to the likelihood method is a set of (nucleotide or amino acid) sequences and a substitution model. Given a branch with the state i and its length t, the probability of observing state j at the end of that branch is then denoted as $P_{ij}(t)$. Current maximum likelihood methods are based ontwo assumptions. The first is the independence of evolution for each site and the second assumption is the independence of evolution of a branch from other branches. The probability of observing a single (possible different) site at each of a leaf node of a tree given a substitution model is called site likelihood. Using the first assumption the likelihood computation can be formulated as the product of individual site likelihoods (D^i)

$$P(D|T) = \prod_{i=0}^{n} P(D^i|T)$$

where T is the tree and $P(D|T)$ the likelihood of the tree. The knowledge of the likelihood for each single site is enough to compute the likelihood for the whole alignment. The following example is taken from (Kea06), because it describes the computation of the likelihood for a single site very well. Given a tree like Figure 2.17

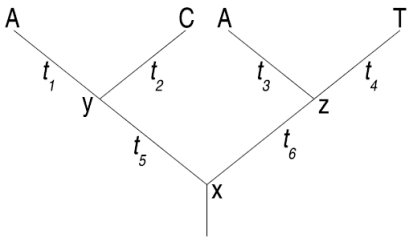

Figure 2.17: A phylogenetic tree. t_{0-6} denote the branch lengths, $x - z$ internal nodes, and the character states are given at the tips of the branches. (Figure taken from (Kea06).)

and an alphabet $A = A, C, G, T$, where A, C, G, T are nucleotides, then the likelihood of a site can be computed as

$$P(D^i|T) = \sum_{x \in A} \sum_{y = \in A} \sum z \in P(A, C, A, T, x, y, z|T) \tag{2.1}$$

This equation calculates the probability to observe the nucleotides A,C,A,T at the leaves of a tree and three unknown nucleotides (x,y, and z) at internal nodes of Figure 2.17. The independence of evolution along all branches lead to equation 2.1.

$$\begin{aligned} P(A,C,A,T,x,y,z|T) =& P(x)P(y|x,t_5)P(z|x,t_6)P(A|y,t1) \\ & P(C|y,t_2)P(A|z,t_3)P(T|z,t_4) \end{aligned} \tag{2.2}$$

where t_{1-6} describe the length of the branches in Figure 2.17. Due to the summation sign and the unknown internal nucleotides (x,y,z) in equation 2.1 the number of terms exponentially increase in the equation. Therefore the computation is practible impossible (e.g. for 50 taxa, we have 49 internal nodes $=> 4^{49}$ terms in the equation). Felsenstein used dynamic programming to calculate the likelihood of a tree recursively - from the the tips of the tree down to the root (Fel81). The probability of the events from internal node n to the leaves of a tree at site s assuming that the site state is $\in$ (A,C,G,T) and is called conditional likelihood and is denoted as

$$\sum_{y=A,C,G,T} P(C|y,t_2) \times P(A|y,t_1) \tag{2.3}$$

The conditional likelihood of an internal site only requires the knowledge of the conditional likelihoods of sibling nodes. According to equation 2.1 and equation 2.2 this can be rewritten to:

$$\begin{aligned} P(D^i|T) = & \sum_{x=A,C,G,T} (P(x) \\ & (\sum_{y=A,C,G,T} P(x|y,t_5)P(y|A,t_1)P(y|C,t_2) \\ & (\sum_{z=A,C,G,T} P(z|x,t_6)P(z|A,t_3)P(z|T,t_4)))) \end{aligned} \tag{2.4}$$

The computation follows a path from the leaves of the tree to the root. Assigning a *1* for an observed base and a *0* otherwise, a walk from the the leaves of the tree will compute alphabetically ordered 4-tuples. An observed A at a particular site will be noted as (1,0,0,0) and sites with sequencing errors can be treated as unknown (1,1,1,1) or e.g. being a purine (1,0,1,0). Since likelihood values are usually very small they are presented as the negative natural log of the likelihood value.

2.4.4.5 Bayesian Inference

The maximum likelihood methods calculates the probability of seeing the observed data (D) given a model/theory (T)

$$P(D|T)$$

Bayesian methods on the other hand calculate the probability that the model/theory is correct given the observed data.

$$P(T|D)$$

Since the bayesian methods are based on the likelihood function they inherit many of its properties. This includes robustness to long branch attraction (AS04) and heterogeneous evolution (GK05). It is possible to include prior knowledge such as a prior distribution of trees and to have a measure of support for the phylogenetic hypothesis through the posterior probability. This posterior probability is the probability of the *ith* tree conditional on the available data. Bayes's theorem is used to calculate this probability

$$P(T_i|D) = \frac{P(D|T_i) \times P(T_i)}{\sum_{j=0}^{S} P(D|T_j) \times P(T_j)}$$

where $P(T_i|D)$ is the posterior probability of tree i, $P(D|T_i)$ is the likelihood of tree i, $P(T_i)$ is the prior probability of tree i and S is the number of possible trees. As Huelsenbeck stated (HLMR02) equation 2.4.4.5 shows that to calculate the posterior probability it is necessary to sum up over all possible trees and integrate over all possible branch length and model parameters. Since this calculation can be very expensive a Markov chain combined with monte carlo integration (MCMC) is used for approximation (MNL99; YR97). The MCMC first starts with a random tree and creates new trees based on the current tree (using e.g. SPR). This is done to examine a wide range of trees in the tree space. A high degree of convergence for the phylogeny can be reached by sampling millions of trees (RH03), but there always remains uncertainty of convergence (MV05). A possible solution is the comparison of outputs from different runs, because different runs from different starting points in the parameter space should lead to the same phylogeny to get strong support (HLMR02). Support for the results can be achieved by periodically (e.g every 1000^{th} generation) taking the current tree topology and use the resulting set of trees to build up a set of probabilities for clades over the entire run (HLMR02)[20].

[20]It should be considered that posterior probabilities are often higher than bootstrap (See Sec. 2.7)values.

2.5 Supermatrix Approach

The concatenation of (incompletely) overlapping datasets is called supermatrix approach. These concatenated datasets form a matrix, which can be analysed using known phylogeny methods to yield the best tree.

Supermatrix methods attempting to sample large groups of organisms (usually) face the problem of a large amount of missing data. The effects of missing data and possible ways to deal with it are described in Sec. 2.2.

The supermatrix approach (also known as 'simultaneous analysis', 'combined-analysis' or 'total-evidence approach') use all character evidence from all taxa directly and simultaneously (See Fig. 2.18).

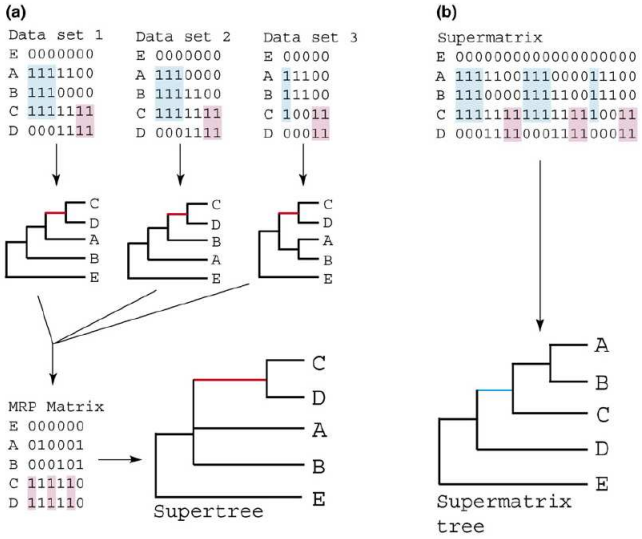

Figure 2.18: Schematic representation of a MRP supertree (left) and a parsimony supermatrix (right). Note that the two approaches are based on the same dataset but yield different trees. (Figure taken from (dQG06).)

An advantage of supermatrices over supertrees (See next Sec.) is that they reveal hidden support, that is, the increased support for a clade in simultaneous analysis as compared to separate analysis of that data. Moreover, supermatrices can support relationships that are contradicted by supertree methods. The full evidence from all characters can be assessed and used to estimate the phylogeny.

The increasing number of sequences available (See Fig. 2.2) favours the supermatrix approach in the future systematic analysis, because supermatrices analyse data simultaneously and individual characters are treated as phylogenetic evidence rather than just the topology of the trees.

2.6 Supertree Approach

Supertree construction is a phylogenetic approach to combine overlapping source trees - and not the character data used to derive those trees - to build a single, large supertree (BE04). One of the main justification for the use of supertree construction is the limited amount of data. It is still unclear whether the entire history of a group of organism can be reconstructed correctly using only a few

genes (KFC+98). By being able to indirectly combine heterogeneous forms of phylogenetic information - the raw data ("total evidence" (Klu98)) or the tree topology derived from them ("taxonomic congruence" (Mic78)), supertrees are a good method for constructing complete phylogenies of groups with hundreds of species (BE05).[21]. Through the combination of different, heterogeneous datasets, the use of supertrees make it possible to get new insights into evolutionary methods and history.

The first idea of supertrees date as far back as the field of systematic itself. In the beginning nested trees were just pasted together as a kind of taxonomic substitution to yield a more informative tree (informal supertree). The development of informal supertrees brought along new insights in evolutionary processes. (e.g. Only through informal supertrees it was possible to imagine the "Tree of Life"[22] as a whole.) Figure 2.19 shows the evolution of supertrees from informal to formal supertrees.

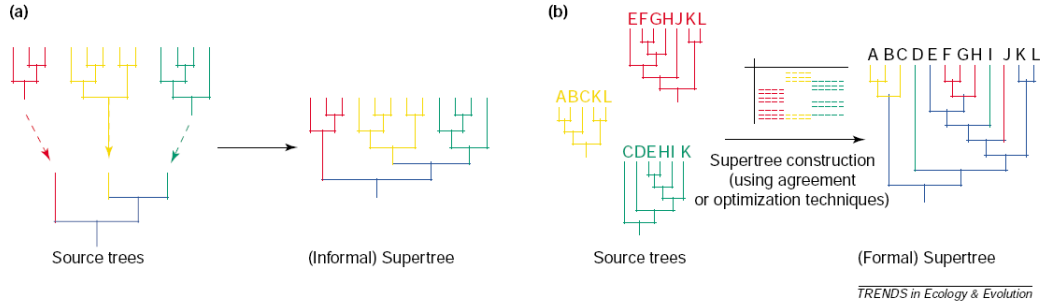

Figure 2.19: Supertree techniques from past and present. **(a)** In the past, hierarchically nested trees were crafted together to yield the supertree. Overlapping positions are shown in the same color. **(b)** In the present, overlapping source trees are combined to yield the supertree. In this example a matrix representation is used. Portions of the supertree determined from a single source tree are displayed in the colour of that source tree. (Figure taken from (BE04).)

A big disadvantage of informal supertrees was that they are not able to include multiple source trees of the same group, but the development of formal supertree methods has improved this situation. After the first formal supertree method was introduced by (Gor86) - an analogy of strict consensus - various methods have been proposed (See Tab. 2.3), but non of them could compete with the total evidence approach (Supermatrix). With the introduction of matrix representation using parsimony, indecently described by (Gor86) and (Rag92), an universal method for combining even incompatible source trees was there to compete with the Supermatrix approach.

2.6.1 Types of supertrees

Supertree methods can be classified broadly as either direct or indirect methods (WTLB01). Direct supertree methods are similar to classical consensus techniques. The supertree is derived directly from the source trees without an intermediate step (See Fig. 2.20). Compatible trees (i.e. without conflicting nodes and full taxon overlap) are required as input for direct supertree methods. Incompatible trees cannot be incorporated in the analysis. The following table gives a short overview over the existing direct (agreement) and indirect (optimisation) supertree methods.

[21] In contrast, the use of the supermatrix approach would have to deal with a matrix full of empty cells
[22] `http://tolweb.org/tree/`

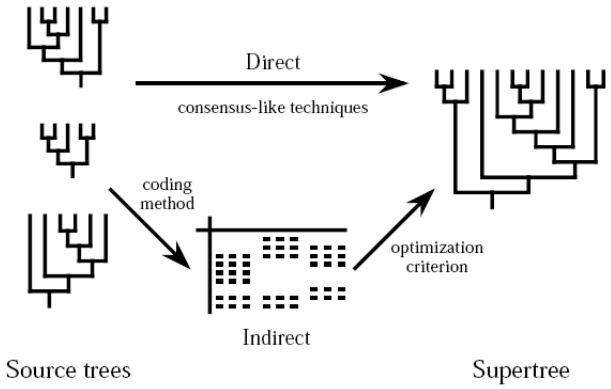

Figure 2.20: Diagrammatic representation of supertree construction, illustrating both direct and indirect methods. (Figure taken from (BE04).)

Table 2.3: Overview of existing direct and indirect supertree methods.

Direct Supertrees	**Indirect Supertrees**
MinCutSupertree	Average consensus (Matrix representation using distances, MRD)
Modified mincut Supertree	Bayesian supertrees
RankedTree	Gene tree parsimony
Semi-Labelled- and AncestralBuild	Matrix representation using compability (MRC)
Semi-strict	Matrix representation using flipping (MRF)
Strict	Matrix representation using parsimony (MRP)
Strict consensus merger	Most similar supertree method (dfit)
	Quartet supertrees

Indirect supertree methods use an intermediate step. The individual source trees topologies are encoded and combined using a form of matrix representation. The matrix is then analysed using an optimisation criterion (BEGS02). These criteria can be *compatibility*, *likelihood*, *least-squares*, *Bayesian methods*, or *parsimony*.
The latter one is used in the supertree method called Matrix representation using Parsimony (MRP) (Bau92; Rag92). A great advantage of indirect supertree methods is that the source trees need not to be compatible.

2.6.2 Matrix Representation using Parsimony (MRP)

The use of a matrix as an intermediate step in supertree construction requires that the hierarchical structure of the source trees is encoded in the matrix. For each internal branch of a source tree, those taxa are listed that appear on one side of the branch and not on the other. The matrix representation in its most basic form scores taxa descendend from that node as *1* and taxa not descendend scored as *0*. This coding is called additive binary coding.

This coding scheme leaves a one-to-one correspondence between the matrix and the tree. Using different optimisation techniques the matrix can be converted back to a tree (See Fig. 2.21). The matrices for each single source tree are then concatenated into a single matrix. Taxa that are not

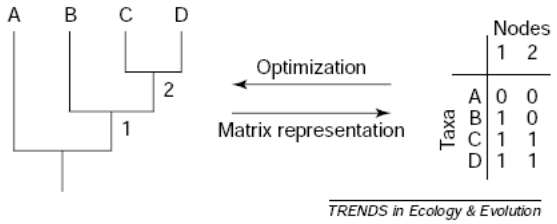

Figure 2.21: The one-to-one representation between a tree and its matrix representation. (Figure taken from (BE04).)

present in a given source tree are coded as missing (*?*) and all trees are virtually rooted with an unique outgroup (Bau92; Rag92). The use of parsimony as a optimisation criterion yield the final supertree. Contrary to the supermatrix approach the elements created by the matrix representation are statements of membership and are only functionally equivalent to characters (BE04).

2.7 Tree Evaluation

Current phylogenetic methods use heuristic to find an optimal tree. Since heuristics do not guarantee to find the best tree and methods for searching tree space can get stuck in a local maximum (and not a global maximum[23]), scientists need a measure of confidence to evaluate results. There are several procedures available to evaluate the phylogenetic signal in the data and the robustness of the tree (SOWH96). The most popular class of tests is the test of data signal versus randomised data.

2.7.1 Bootstrapping

This method belongs to the class of randomised character data (Permutation tests). The idea is to randomise parts of the data and rebuild the tree (See Fig. 2.22). This is done several times (e.g. 500) and estimates the accuracy of the tree by comparing the topology of parts of the tree build with randomised data and the real data. The bootstrap method was invented by Efron in 1979 (Efr79) and introduced as a tree evaluation method by Felsenstein in 1985 (Fel85).

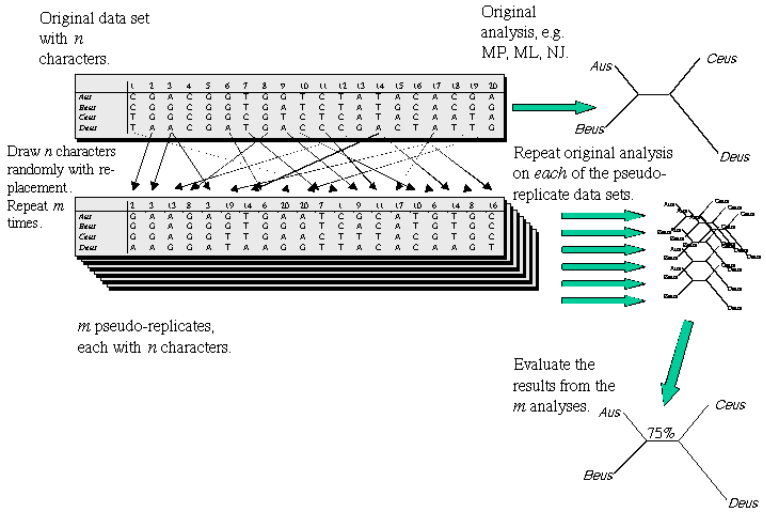

Figure 2.22: The process of bootstrapping. Parts of the dataset are randomised, the tree is inferred and the number of matching topologies determines the bootstrap value. (Figure taken from (Boo).)

The output of the bootstrap method - given a tree as input - is a number associated with a branch. This number is the proportion of bootstrap replicates supporting the monophyly of that clade.
The bootstrapping method consists of two main steps. The first includes the generation of new datasets using randomly sampled columns of characters. The total number of positions of all datasets is the same. The second step is the computation of a number that gives the proportion of times a particular branch appears in a tree (Bootstrap value) (BO05). A value of $> 75\%$ is regarded as good support for the topology, a value of $>95\%$ is a desirable degree of confidence in the data..

[23]That is the tree with the highest score given some optimisation function

Chapter 3

The Phylogeny Pipeline

The process of phylogenetic reconstruction consists of several steps of analysis. These steps are the search for homologs for a given set of sequences (See Sec. 2.1), the alignment of these sequence (See Sec. 2.3), the selection of conserved parts from the alignments (See Sec. 2.3.6), the construction of a phylogenetic tree using either a supermatrix (See Sec. 2.5) or a supertree approach (See Sec. 2.6) and recieving confidence in the data using e.g. bootstrapping (See Sec. 2.7.1). Each of these steps requires the selection of a program - to perform the requested task - and its usage. The use of a scripting languages like *Perl*[1] allows the construction of a phylogenetic pipeline to automate this process of phylogenetic reconstruction. This pipeline accepts a set of sequence, executes all steps of the analyis, and - as a result - outputs a phylogenetic tree. It offers the use of different popular programs in each analysis step, performes data conversions between the different programs (if needed), and handles the organisation of the data throughout the analysis.

The following section explains the phylogenetic pipeline in detail.

[1] `www.perl.org`

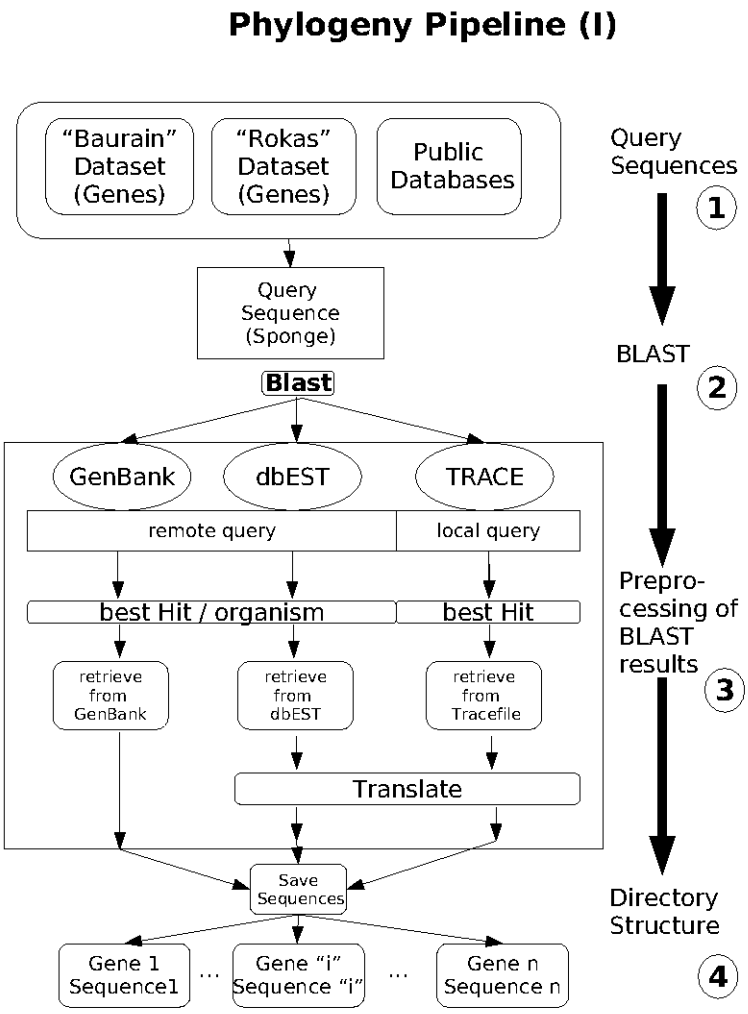

Figure 3.1: The first part of the phylogeny pipeline - the homology search. It takes a query sequences from a file in a given directory structure, uses BLAST to perform the homology search against different databases, parses the BLAST result and retrieves the best hits for each BLAST search. Additionally it translates the resulting DNA sequences (in case of BLAST searches against dbEST and Trace Archive) and organises the sequences in a directory structure.

3.1 Searching for Homologs

The phylogeny pipeline performs a homology search against public databases to find closely related sequences from other organism given a sequence of interest. The results of the different BLAST searches are postprocessed, the best hits obtained from public databases and saved in an appropriate directory.

1. **Selection of a Query Sequence**
   A good selection of a query sequence for a BLAST search is essential and can decide about success in finding homologous sequences (See Sec. 2.1.2). Since we are looking for homologous sequences of sponge genes, appropriate query sequences for the BLAST search are sponge sequences. The

pipeline needs the query sequences organised in a directory structure like Fig. 3.2. Depending on

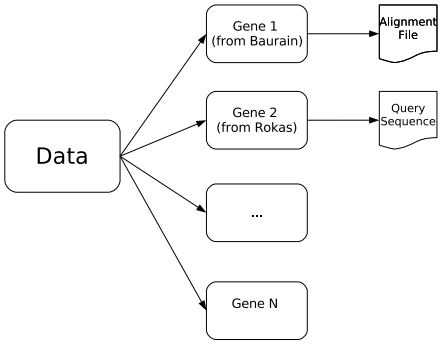

Figure 3.2: The data organisation needed for the first step of the analysis - the homology search. Each subdirectory of the directory "Data" contains either an alignment file or a single sequence file. These can be used as a query sequence for the BLAST searches.

the type of sequence in the directories, the pipeline can either update an existing dataset - given an alignment file - or create a new dataset - given single sequence file. This is done by either parsing the alignment for an appropriate candidate sequence or just using the single sequence as a query sequence.

2. **BLAST**
   Three different databases can be blasted to establish a large Taxon Sampling. These databases are nr (non-redundant) (See Section 2.1.1.1), dbEST (See Section 2.1.1.2), and a local database created from a Trace file for the Organism Reniera (See Section 2.1.1.3). For each database different versions of BLAST are needed to perform the search.

   Blasting the different databases
   GenBank (*nr* and *dbEST*) offers a nice tool to submit BLAST searches to the NCBI servers and obtain the results after the search. This tool is called netblast[2] and can be used given the type of BLAST program "-p", the query sequence "-i", the output file "-o" and all other parameters BLAST uses.

```
blastcl3 -p blastn -i seq.txt -d dbEST -o n_refm.out
```

Start a remote BLAST Search

   The third database can be created locally. The Trace file can be obtained from the Trace archive (Trab) and the tool *blastdb* used to convert the fasta file into a BLAST database. **Different BLAST programs**
   Since dbEST and the local *Trace* database consist of nucleotide sequences, the BLAST program *tblastn* performes a protein query against a tranlated nucleotid database in a six reading frames.

[2] `http://www.ncbi.nlm.nih.gov/staff/tao/URLAPI/netblast.html`

In the case of searching against the database *nr* the program *blastp* will perfom a protein query against a protein database.

3. **Postprocessing the BLAST results**
   All results - either recieved from NCBI via netblast or from the local BLAST search - will be temporarely stored and parsed using special BLAST modules offered by the programming language *Perl* and its extension *Bioperl*[3].

   The program considers only the best hits for each organism as a homologous sequence. The next step is to obtain the sequences from GenBank (nr and dbEST) or from the local Trace file. The sequences can be obtain from the Trace file or using Bioperl from GenBank. E.g. the following command retrieves the protein sequence for the *RAS-related protein P23* (*Rattus norvegicus*, Acession number: gi|171001) from "genbank" in "fasta" format:

```
perl -MBio::Perl -e '                #Use Module Bio::Perl and execute command
$database="genbank";                 #Search at GenBank
$id="gi|1710001";                        #The accession number of the sequence
$format="fasta";                        #The sequence in fasta format
$sequence=get_sequence($database,$id);        #Retrieve sequence
write_sequence(">-",$format,$sequence)';          #Print to STDOUT
```

Retrieve a Sequence from the Database

   Since two out of three BLAST searches will be performed against translated nucleotid databases, the best BLAST hits have to be translated according to the right reading frame from the BLAST result. This is done using the program *transeq* from the EMBOSS package (RLB00).

4. **Set of Homologous Genes**
   Each best BLAST hit for each organism and each gene is saved in a single file. The name of the file correspondes to the database searched and the name of the organism and is saved in the directory with the name of the gene the sequence describes (See Fig. 3.3).

3.2 The Analysis of the Data

[3] http://www.bioperl.org/wiki/BLAST

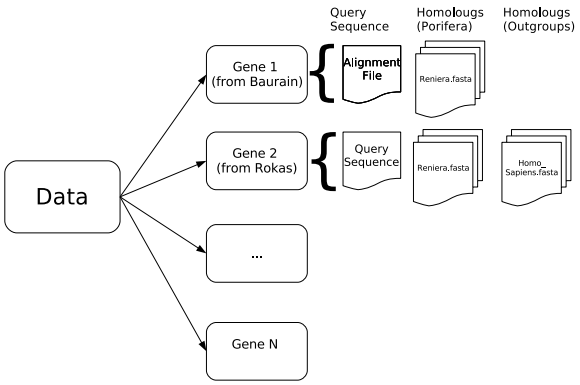

Figure 3.3: The directory structure of after the search for homologous sequences. Based on the initial directory structure in Fig. 3.2 the subdirectories now contain the query sequences and the homologous sequences found by BLAST.

The dataset is complete, homologous sequences to the query sequences have been found, and the analysis of the evolutionary relatedness can start. The result will be a phylogenetic tree.

The data analysis consists of several steps:

1. **Sequence Alignments**
   The query sequence and the homologous sequences of each gene will be merged into a single file. This file can be input for several alignment programs. These programs are *ClustaW*, *Dialign*, *T-Coffee* and *Muscle* (See Sec. 2.3). The alignment programs will take the input files and and arrange the sequences to identify regions of similarity that may be evolutionary related.

   The idea of using different programs is that it will broaden the scope of analysis. The results will be compared at the end. If all data contain the same statement about evolutionary history, then this an additional factor of convidence of the data.

2. **Postprocessing**
   The program *Gblocks* (See Sec. 2.3.6) takes alignments and delete parts that are not useful for further phylogenetic analysis. The deleted parts contain either too less or too much phylogenetic information. A side-effect is that shorter alignments reduce the computational complexity the analysis.

3. **Supertree / Supermatrix methods**
   There are two popular approaches for phylogenetic analysis of multiple gene sets. These are the supertree and the supermatrix approach.

   - Supertree
     In the case of supertree analysis, the alignments - foreach alignment method and foreach gene separately - are taken, the adequate evolutionary model selected using the program *Modelgenerator* (AZP05) [4] and phylogenetic methods will be used to infer the best tree

[4]Note that this step done manually by the user, because it requires the use of the webinterface at `http://www.cs.nuim.ie/distributed/multiphyl.php`

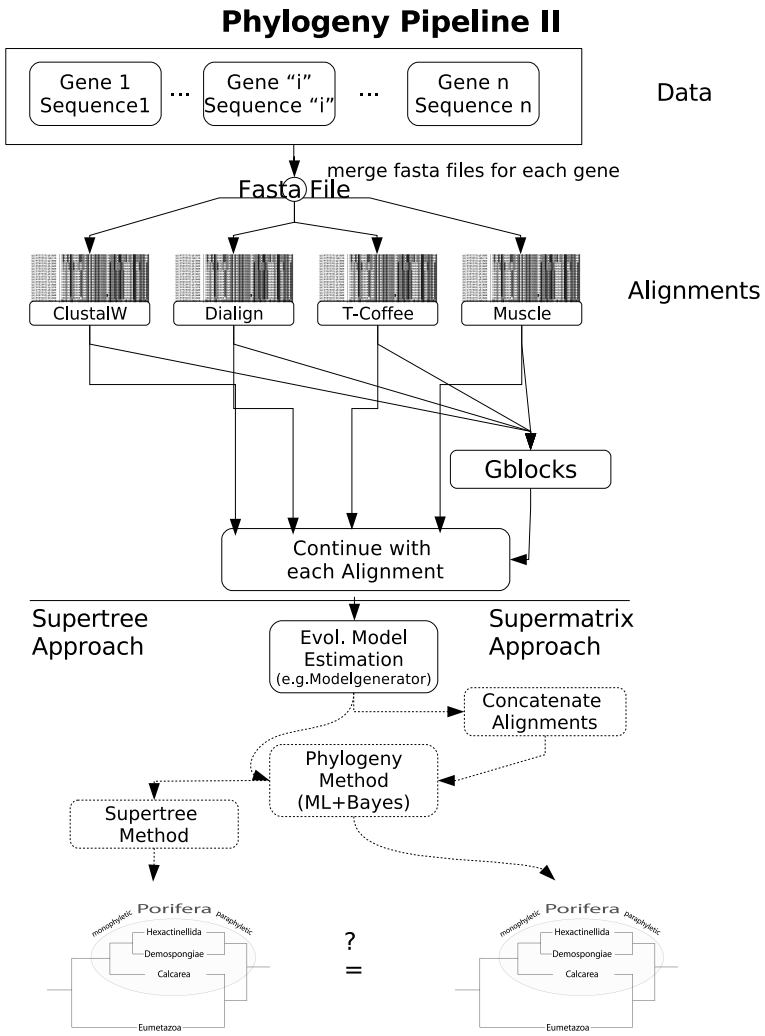

Figure 3.4: The second part of the workflow of the phylogeny pipeline. All sequences for each gene are concatenated to carry out the alignment process using different programs. Evolutionary models are estimated for the complete and the alignments containing only conserved parts (selected by *Gblocks*). The analyis can then be finished with the Supertree or the Supermatrix approach.

according to some optimality criterion. The different programs are *MrBayes*, *PhyloBayes*, *PHYML* and *RAxML*.

The resulting tree files will then be concatenated to a single tree file for the final supertree construction using the program *clann* (cla).

- Supermatrix

  The Supermatrix approach starts with the concatenation of the alignment into a supermatrix using *Scafos*, a program "for selection, concatenation and fusion of sequences for phylogenomics" (RREP07)[5]. Foreach single alignment in the supermatrix the best model of protein evolution will be estimated using *Modelgenerator* via a webinterface[6] (Kea06). The supermatrix along with the a list of protein evolution models serve as an input for

[5]This step is also done manually, because the user interface of scafos offers complex functionality
[6]`http://www.cs.nuim.ie/distributed/multiphyl.php`

the final phylogenetic analysis, which will be carried out using the programs MrBayes, PhyloBayes, PHYML, and RAxML.

3.3 The Final Directory Structure

Phylogenetic analysis are complex tasks. They include several points where the researcher have to take a closer look on the data to evaluate them. So a great interest is to design a directory structure that allows the researcher to easily find the results from the different analyses and evaluate them (See Fig. 3.5).

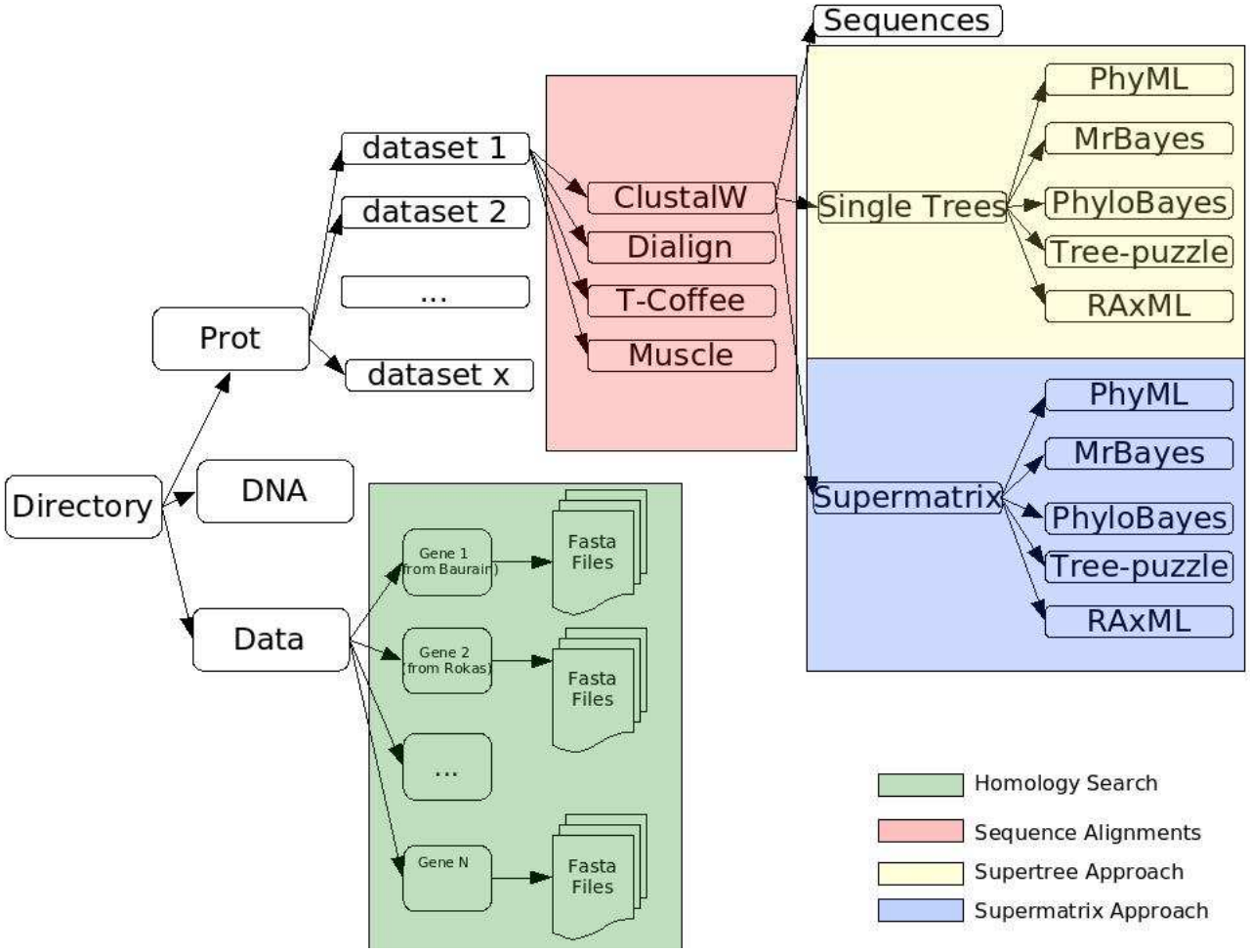

Figure 3.5: The directory structure after the phylogenetic analysis has completed. The "Data" branch contains all sequence data after the homology search, the "Prot" branch is used if the protein sequences are to be analysed, the "DNA" branch for DNA sequences. The subbranches in the "Prot" directory inherit several user defined datasets. Each of which contains the full sub structure including applied alignment and phylogeny methods.

The directory structure is designed to reflect a straighforward run of the pipeline. E.g. The researcher wants to view the alignments created with *Clustalw* for his dataset "fish", he just has to follows the path

```
/prot/fish/clustalw/sequences/
```

or the researcher wants to take a look at the phylogenetic tree for the gene "catalase" from his dataset "human" created with the phylogeny method *RAxML* and *Muscle* as the multiple sequence alignment program we will find it in

```
/prot/human/muscle/trees/raxml/
```

3.4 The User Interface

In the following we will shortly describe how the user can use the phylogeny pipeline to carry out a complete analysis or just single analysis steps.

3.4.1 The Command line tool

The command line tool provides simple access to the full functionality of the program. A straighforward run of the pipeline includes the following steps: Homology Search, Construction of Multiple Sequence Alignments, Selection of Conserved Regions from the Alignments, and Phylogenetic Methods.

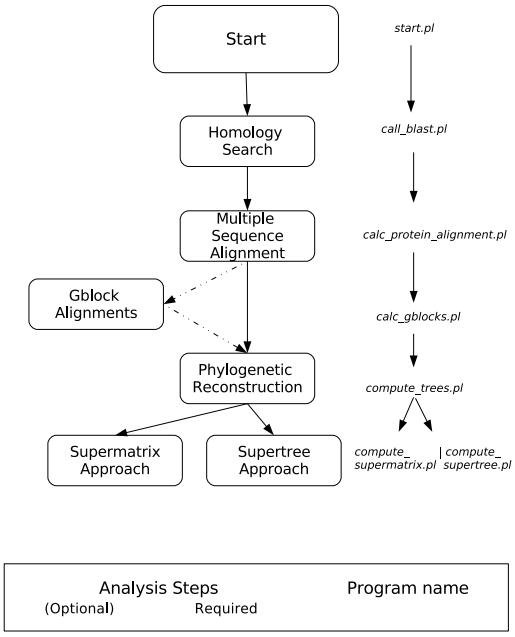

Figure 3.6: The different steps of the phylogenetic analysis (left) and the name of the corresponding programs (right). Note that the selection of conserved regions of alignments (using Gblocks) is optional.

Fig. 3.6 gives an overview of the different programs related to the different steps of the analysis: The user can start a complete analysis or can enter the analysis at every step. The documentation for all programs can be found in the appendix (See App. A.2).

3.4.2 The Graphical User Interface (GUI)

A GUI was developed using the Perl extension *TK*[7]. This GUI allows the use the phylogeny pipeline without any knowlegde of the unix command line. All available parameters for the different programs of the pipeline can be entered and a help function is available. A screenshot of the GUI is in Fig. 3.7.

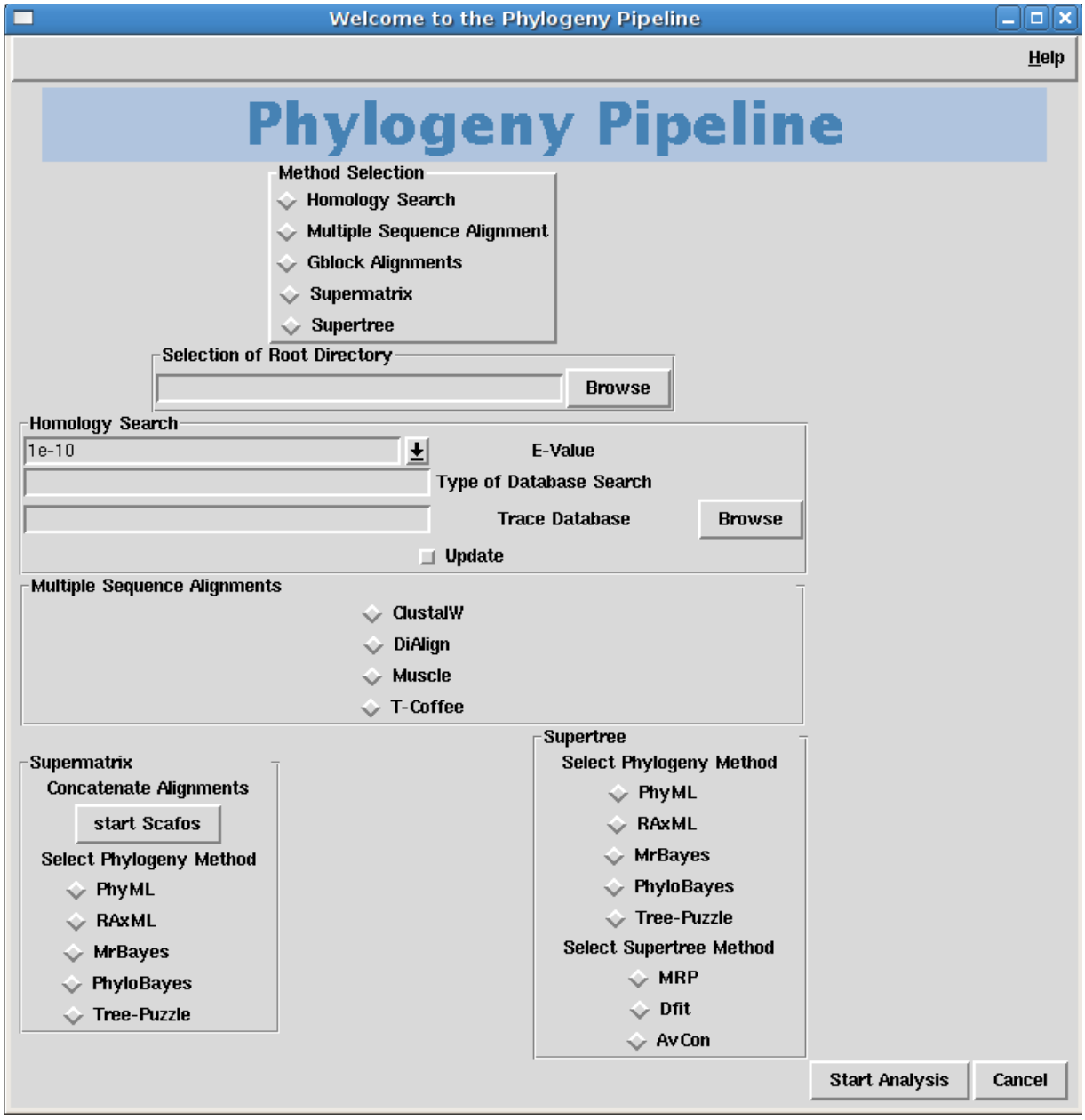

Figure 3.7: The screenshot of the GUI of the phylogenetic pipeline. The GUI offers the access to all steps of the phylogenetic analysis, including the selection of a root directory (required for each analysis step) and analysis-dependent parameters (e.g. e-value with the homology search).

[7] http://www.perltk.org/

Chapter 4

The Dataset

The right selection of taxa and genes is essential for getting reliable results. For that reason, we will use two existing datasets as a basis and update these so that they fit our goal of analysis. These datasets were used in studies from Rokas (RKC05) and Baurain (BBP07)[1] The interest of Rokas was the investigation of metazoa at the base of the metazoan tree and within protostomes. He selected metazoans and closely related eukaryotes including representatives from choanoflagellates, poriferans, cnidarians, platyhelminths, priapulids, annelids, mollusks, arthropods, nematodes, urochordates and vertebrates (RKC05). Rokas used a large dataset, but his taxon sampling and the selection of a model of evolution lead to misleading results (BBP07). Baurain updated this dataset to decrease the possibility of these errors. He extended the dataset both in the number of genes and in the number of taxa.

The goal of our study is to contribute to answering the question if the phylum porifera - with its three subphyla demospongiae, hexactinallida, and calcarea - is *monophyletic* or - as some studies using molecular markers suggest - *paraphyletic* (See Sec. 1.3). To answer this question additional homologous sequences from phylum porifera were added to the existing dataset by BLAST searches.

The Rokas dataset is not publicly available, but there is a list of genes which can be used to search for homologs. The *Baurain* dataset already includes sponge sequences from two organisms: Suberites and Reniera (both demospongiae). Taking one of these sponge sequences we will search for homologous sponge sequences. The goal here is to get at least one sequences for each of the three poriferan classes. The *Baurain* dataset comprises only selected genes. We will add additional genes from public databases to increase the number of sponge sequences. Additionally we will create a completely new dataset to extend the *Taxon Sampling*. This will be done getting all the sequences for phylum porifera from GenBank.

[1] These dataset are called *Rokas* dataset and *Baurain* dataset hereafter.

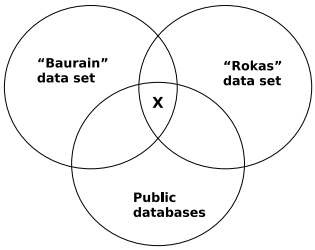

Figure 4.1: The combined datasets from Rokas, Baurain and the manually generated dataset from GenBank. The intersection includes genes with sponge sequences from all poriferan groups available (marked with an X).

The most interesting subset of these datasets consists of all genes with at least one sequences from each poriferan class available (See Fig. 4.1), because the goal of our study is to clarify the relationship of this three classes and a tree with sequences covering just two of the three classes is not informative.

Chapter 5

Results

In this chapter we present the results from our phylogenetic analysis to help to answer the question if sponges are a monophyletic or paraphyletic group. We developed a phylogenetic pipeline as described in Chapter 3 and used it to carry out the different steps of the analysis. If not mentioned otherwise all steps of the analysis were performed using default parameters.

5.1 Data Assembly

5.1.1 Construction of dataset

We started our analysis with the Baurain dataset. The dataset from Baurain contains 133 genes (12,942 amino acid positions) from 57 taxa (See App.). Each of the protein alignments from the Baurain dataset (BBP07) was updated using our phylogenetic pipeline. All fungi sequences were excluded from the analysis, because they are not in the focus of our study of sponges. For each gene, a sponge sequence (*Suberites*, *Reniera*) or the sequence closest to the sponges (*Monosiga ovata*) was selected from the protein alignment from the Baurain dataset to query the non-redundant (*nr*) and the EST database (*dbEST*) using *BLAST*. To find an e-value and a substitution matrix that fits our data best, iterative BLAST searches were performed. According to the recommendation to interpret e-values (See Sec. 2.1.2) we used e-values ranging from $1e-10$ to $1e-40$. The tested subsitution matrices were *BLOSUM62* and *PAM250*. BLOSUM62 is the standard matrix for BLAST searches, PAM250 performes well with distant sequences. For each e-value and substitution matrix a single dataset was created (See Tab. 5.1).

5.1.2 Gene Selection

For each dataset, only those genes were kept in the analysis further with at least one sequences for each sponge group available. This drastically reduced the number of genes in the study from 133 to 10-15 genes. An overview of the genes used in this study is in App. A.2.

After the evaluation of the BLAST searches we decided to continue the analysis with the two most different datasets for each substitution matrix to reduce the computational complexity. These are ds_10 and ds_40 for the substitution matrix BLOSUM62 and ds_10_pam and ds_40_pam for PAM250, respectively.

Table 5.1: Overview of the different dataset, e-values, substitution matrices used for the homology search and resulting number of genes.

Dataset	E-Value	Substitution Matrix	Number of Genes
ds_10	1e-10	BLOSUM62	15
ds_20	1e-20	BLOSUM62	13
ds_30	1e-30	BLOSUM62	10
ds_40	1e-40	BLOSUM62	9
ds_10_pam	1e-10	PAM250	13
ds_20_pam	1e-20	PAM250	10
ds_30_pam	1e-30	PAM250	7
ds_40_pam	1e-40	PAM250	5

5.1.3 Construction of gene alignments of single genes

Alignments were constructed using ClustalW, Muscle, T-Coffee, and DiAlign[1]. Ambiguously aligned regions were automatically deleted with Gblocks (Cas00).

5.1.4 Chimerical Operational Taxonomic Units (OTUs)

To increase the amount of data, we created chimerical sequences by merging sequences from closely related taxa using the program *SCAFOS*. Sequences are incorporated into the chimerical sequence in descending order of sequence length as shown in Fig. 5.1.

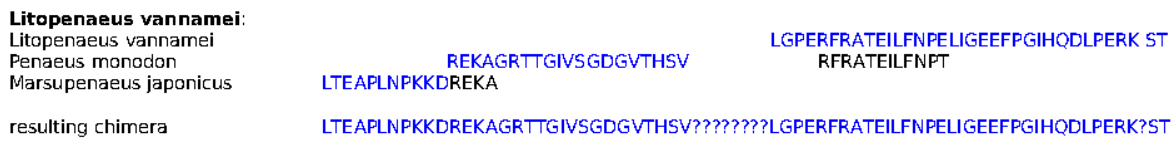

Figure 5.1: Sequence parts are incorporated from longest to shortest. Selected parts are coloured in blue. The chimerical sequences is the concatenated of the selected parts of the sequences.

Chimeric OTUs have been named after the inclusive species that was most represented (See Tab. A.3).

5.2 Phylogenetic and Evolutionary Analyses

5.2.1 Selection of Evolutionary Model

The evolutionary models for each gene were estimated using the program *ModelGenerator*[2]. ModelGenerator selected *RtREV* (DRMG02) to be the best-fit amino acid substitution model for all genes.

[1]Since the alignments do not differ significantly, only ClustalW alignments will be considered in the following.
[2]`http://www.cs.nuim.ie/distributed/multiphyl.php`

5.2.2 Creating different datasets by excluding taxa

For each dataset we created a reduced dataset, including only those taxa present in > 50% the genes. Those dataset get the postfix "red" (for reduced). E.g. The reduced dataset from ds_10 is ds_10_red. This reduced the number of sponge taxa to four. These taxa are *Suberites sp.*, *Reniera sp.*[3] (both Demospongiae), *Aphrocallistes vastus* (Hexactinellida), and *Leucosolenia sp.* (Calcarea).

5.2.3 Supermatrix

The gene alignments were concatenated into a supermatrix using *SCAFOS*.

Table 5.2: The number of amino acid positions, genes, OTUs with the different datasets, and a reference to a more detailed statistic of each dataset.

Dataset	#AA positions	Number of Genes	Number of OTUs	Full Statistics
ds_10	12526	15	68	App. A.4
ds_10_red	12526	15	48	App. A.5
ds_40	7278	9	68	App. A.6
ds_40_red	7278	9	47	App. A.7
ds_10_pam	8906	13	68	App. A.8
ds_10_pam_red	8906	13	43	App. A.9
ds_40_pam	4655	5	68	App. A.10
ds_40_pam_red	4655	5	45	App. A.11

The Maximum likelihood analysis were performed using the parallel versions of *PHYML* (GG03) and *RAxML* (Sta06). Support values for maximum likelihood analysis were obtained after 100 bootstrap replicates.

Bayesian Inference was performed using *PhyloBayes* (phya) and the parallel version of *MrBayes* (ADHR04). Trees were sampled every 1000 generation to get posterior probabilities. The burn-in value was set to 300 trees. This is the level at which all variable parameters reached a stable value in a preliminary run. The total number of generations was set to 300,000 generations. Four parallel chains (one cold and three heated) were used to exhaustively search the tree space.

All analysis were done using the RtREV model of amino acid evolution, except for the analysis using PhyloBayes. Here we used the *CAT* model (LP04) in a MCMC framework.

5.2.4 Supertree

The single gene trees were inferred using PHYML with the RtREV model of amino acid evolution. The concatenated trees were then used to carry out the supertree analysis using the program *clann* and the supertree methods dfit (cla) and MRP (See Sec. 2.6.2).

[3]Note that the term *Reniera sp.* is not used any more, instead this species is called *Amphimedon queenslandica*

5.3 Phylogenetic Trees

In the following section we present our resulting phylogenetic trees. They are presented in a way to show the effects of different parameters and settings we applied throughout the analysis. The effects we investigated were:

- The reduction of datasets by setting a cut-off value for missing data and its effect on the resulting phylogenetic tree (See Fig. 5.2 and Fig. 5.3).
- The use of differente-values and their effect on the resulting phylogenetic tree (See Fig. 5.4 and Fig. 5.5).
- The use of different substitution matrices for the homology search and their effect on the resulting phylogenetic tree (See Fig. 5.6).

The taxa names have been prefixed with the a two-letter code corresponding to the taxonomic group they are a part of (AN = Annelids, AR = Arthropoda, CH = Choanoflagellata, NE = Nematodes, MO = Mollusca, PL = Plathelminthes, PO = Porifera, UR = Urochordates, VE = Vertrebratae). The taxa names of the sponges have one additional character showing to which of the three poriferan groups they belong (D = Demospongia, H= Hexactinellida, and C = Calcarea).

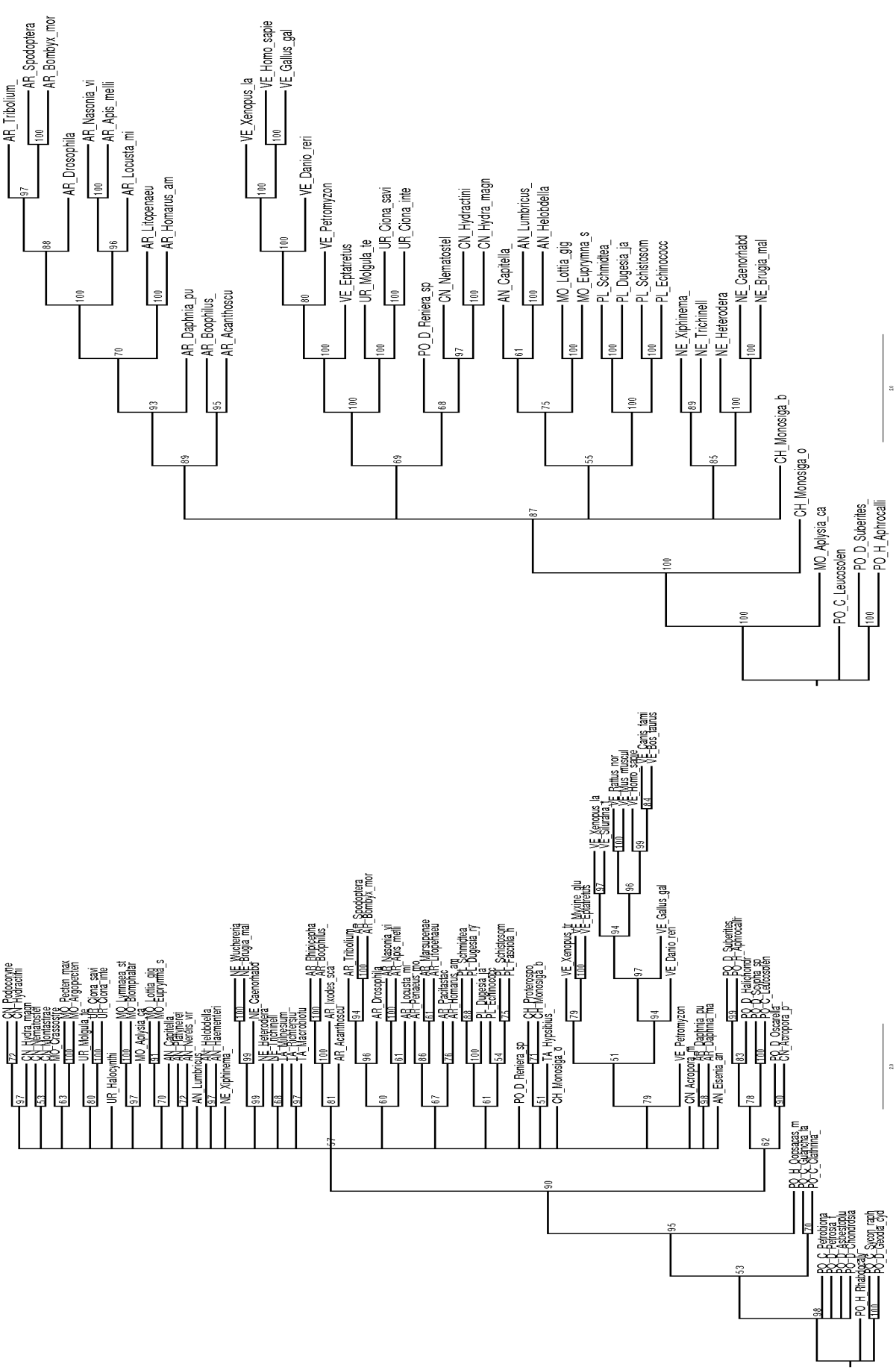

Figure 5.2: The effects of using a cut-off value to exclude taxa lacking sequences for more than 50% of the genes. On the left is the tree build with the whole dataset (ds_10), on the right is the tree build with the reduced dataset (ds_10_red).

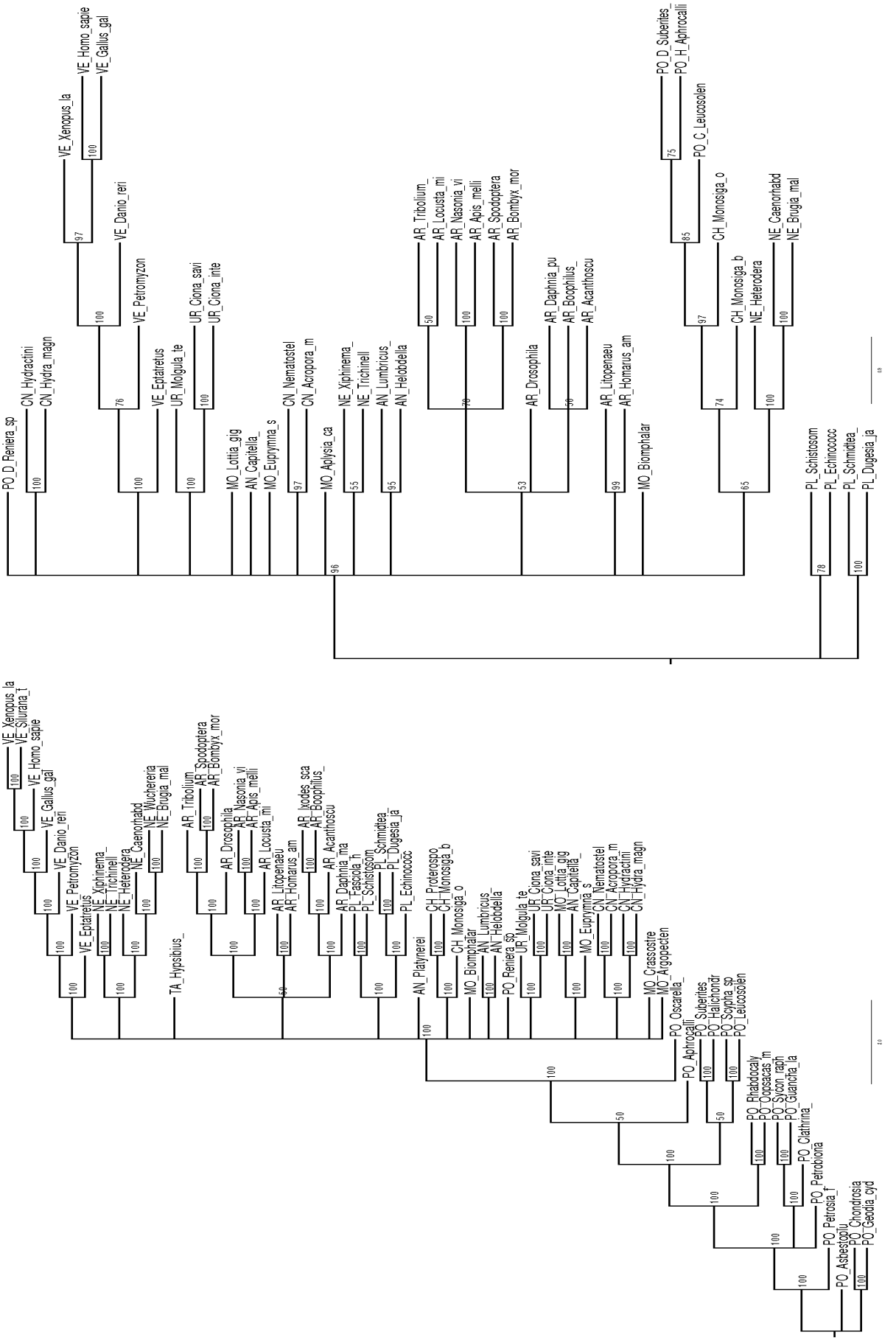

Figure 5.3: The effects of using a cut-off value to exclude taxa lacking sequences for more than 50% of the genes. On the left is the tree build with the whole dataset (ds_40), on the right is the tree build with the reduced dataset (ds_40_red).

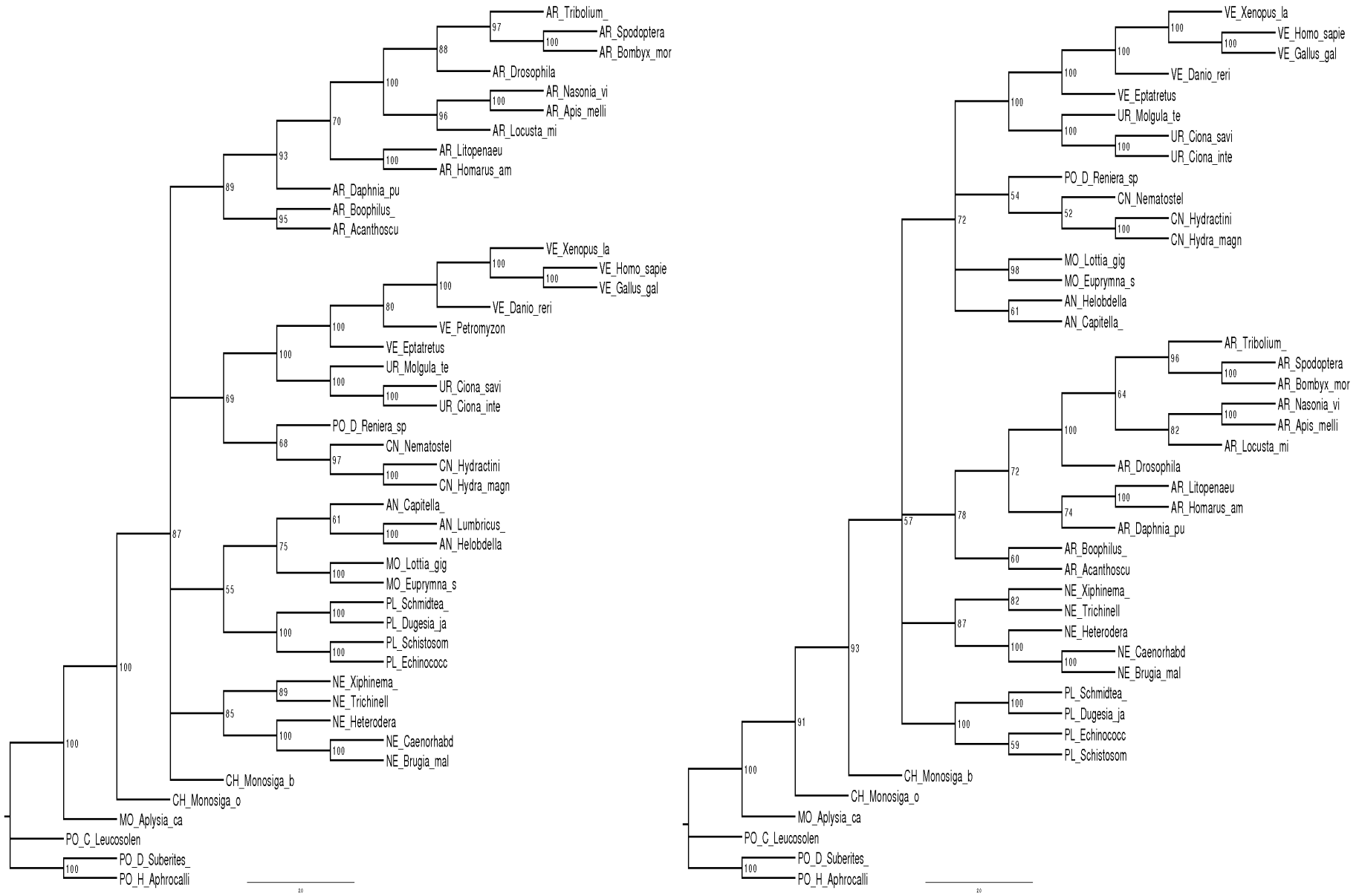

Figure 5.4: The effects of different substitution matrices for the homology search. On the left side the tree for the BLOSUM62 (dataset ds_10_red), on the right side the tree for the PAM250 matrix (ds_10_pam_red).

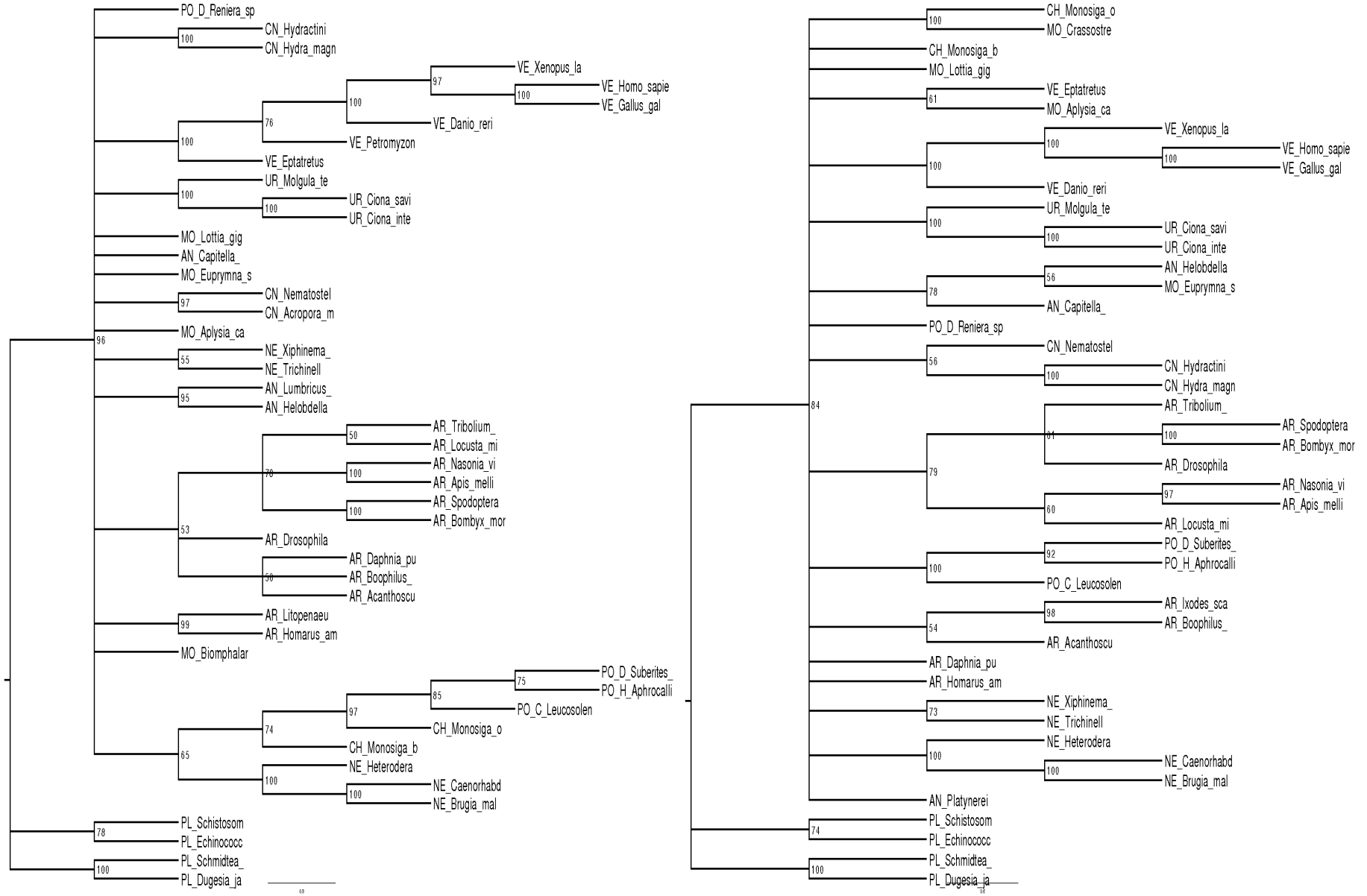

Figure 5.5: The effects of different substitution matrices for the homology search. On the left side the tree for the BLOSUM62 (dataset ds_40_red), on the right side the tree for the PAM250 matrix (ds_40_pam_red).

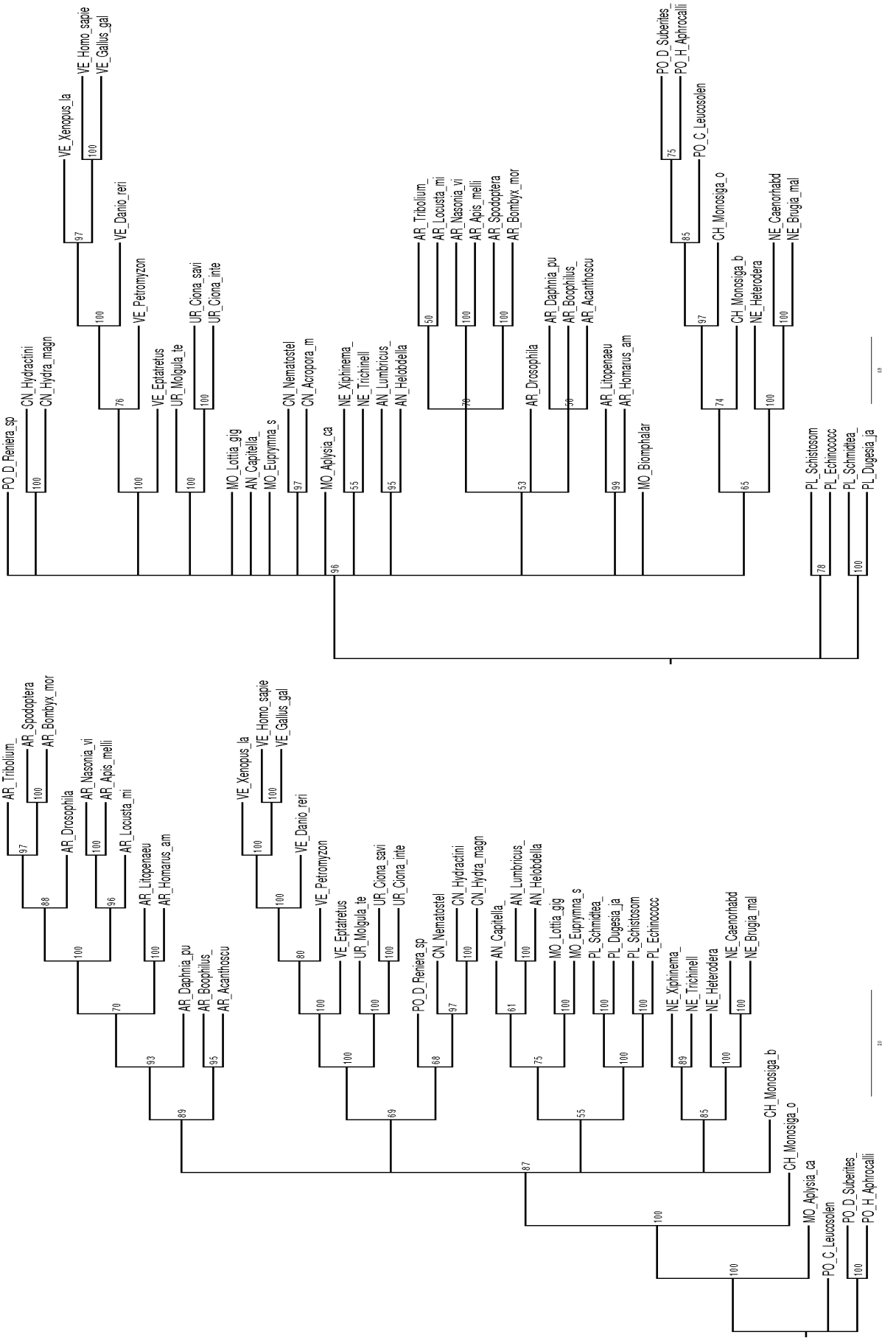

Figure 5.6: The effects of different e-values for the Homology search and their effect on the resulting trees. On the left side the tree inferred from the dataset ds_10_red, on the right side the tree inferred from the dataset ds_40_red.

Chapter 6

Discussion & Outlook

The phylogenetic pipeline (See Sec. 3) has been successfully implemented and used to carry out a phylogenetic analysis.

The goal of this analysis was to support one out of two hypotheses about the history of sponges: Either that sponges are a monophyletic or paraphyletic group (See Sec. 1.3).

The results we achieved so far do not clearly corroborate one of these hypotheses. Due to the limited amount of time and the long runtime of the phylogeny programs for bayesian inference (MrBayes and PhyloBayes) and for maximum likelihood analysis (RAxML), our results are based on the maximum likelihood analysis using PHYML only.

PHYML uses the heuristic NNI (See Sec. 2.4.4.1) to search tree space and trees constructed with this method usually have lower support values (BBP07) than other methods. Therefore, our results should be treated with caution.

The inferred trees from the reduced datasets all grouped the sponges together (except for *Reniera sp.* whose role is phylogeny is doubtful (HvS06)) and this may be seen as a first hint to the monophyly of sponges. A larger taxon sampling and the use of other methods will give more evidence for one of the two hypotheses. Our results also show evidence for the monophyly of *Silicea* (Demospongiae + Hexactinellida), since the sponges *Suberites sp.* (Demospongia) and *Aphrocallistes vastus* (Hexactinellida) were grouped together in each tree (See Sec. 5.3).

The main focus of this study was to decide whether the sponges are mono- or polyphyletic. But the fact the we used different parameters and methods throughout the analysis gives room for further insights:

- The results from the iterative search for a best-fit e-value and an appropriate substitution matrix (See Sec. 5.1) suggests that the choice of an e-value in the range of 1e-10 - 1e-40 (See Fig. 5.6) as well as the choice of either BLOSUM62 or PAM250 as a substitution matrix (See Fig. 5.5) does not influence our results significantly.

  The choice of a lower e-value (1e-40) resulted in a more stringent phylogenetic tree including sequences more likely to be homologs, but with the a general lower resolution. The basal part of the tree remained unchanged with different substitution matrices with the dataset ds_10 (See Fig. 5.4) and the same groups of organisms were clustered together to form monophyletic clades in both trees. Even the support values were almost equal.

- The results from different alignment methods (See Sec. 5.1.3) on the same dataset do not differ significantly. As suggested (RK03), the choice of an alignment method is not essential for the success of a phylogenetic analysis.

- According to (Wie06) taxa with data present in less than 50% of the genes can be excluded from further analysis. We can support this suggestion. After the exclusion of these taxa, we could reduce artefacts (e.g. the vertebrate *silurana* clustered with the sponges in Fig. 5.2) in the tree and received higher support values for the different clades (See Fig. 5.3 for the effects of taxa exclusion on dataset ds_40_red). The more taxa we excluded the more stringenter our resulting phylogenetic tree became. E.g. the complete dataset ds_10 included an artefact, the cnidaria Arcopora palmata clustered with the sponges. After the exclusion we could find the sponges grouped together and also the monophyly of other groups (e.g. the arthropods) reconstructed (See Fig. 5.2).

Although we could not clearly answer the question of the monophyly of the sponges, we found some evidence for it. This evidence will be the basis for further analyses.

On the one hand, we are waiting for the three other phylogeny programs to finish and on the other hand we will extend our dataset to continue our study with an increased taxon sampling and more phylogenetic signal included.

The results from RAxML and the Bayesian Inference programs (MrBayes and PhyloBayes) should give us more information whether our data include enough phylogenetic information to answer our initial goal of the study - are sponges mono- or polyphyletic? The reason for this is that MrBayes and RAxML generally perform well and are more accepted in the phylogeny community, because they use improved heuristics for searching the tree space and that improves the likelihood of finding the best tree in tree space and therefore getting better results. Additionally, PhyloBayes uses the CAT model which showed to perform well by obtaining a better statistical fit, and alleviating phylogenetic artefacts, due to long branch attraction (LP06).

If the taxon sampling we used is to sparse, we will extend the existing dataset with the genes from Rokas and genes from public databases as described in Sec. 4 to increase the number of genes and taxa in the study. Additionally we will include data from ongoing sequencing projects. In addition to the supermatrix approach (See Sec. 2.5), we will also focus on the supertree approach (See Sec. 2.6) to improve the quality of the results and to find out if its useful for our analysis.

Chapter 7

Conclusion

In this thesis we presented a phylogenetic pipeline for the use in phylogenetic analyses. The pipeline covers all important parts of a complete phylogenetic analysis: The search for homologous sequences (given a sequence of interest), the construction of sequence alignments to bring evolutionary related parts of the sequence in correspondence, and the application of two approaches to construct a phylogenetic tree showing the evolutionary relatedness of organisms.

We showed the practical use of this pipeline with a concrete example. The application of the pipeline to find a solution for the question whether the most recent common ancester and all descendants of the sponges are sponges (monophyly) or not (paraphyly). We used our pipeline to carry out all steps required for this phylogenetic analysis. The data and methods we used could not clearly answer this question, but indicates that the use of more data an additional methods could lead to an answer of this fundamental question in sponge history.

Index

Bibliography

ADHR04 Gautam Altekar, Sandhya Dwarkadas, John P. Huelsenbeck, and Fredrik Ronquist. Parallel metropolis coupled markov chain monte carlo for bayesian phylogenetic inference. *Bioinformatics*, 20(3):407–415, February 2004.

AGM+90 S.F. Altschul, W. Gish, W. Miller, E.W. Myers, and D. Lipman. Basic local alignment search tool. *J. Mol. Biol.*, 215:403–410, 1990.

AS04 Frank E. Anderson and David L. Swofford. Should we be worried about long-branch attraction in real data sets? investigations using metazoan 18s rdna. *Molecular Phylogenetics and Evolution*, 33(2):440–451, November 2004.

AZP05 Federico Abascal, Rafael Zardoya, and David Posada. Prottest: selection of best-fit models of protein evolution. *Bioinformatics*, 21(9):2104–2105, May 2005.

Bau92 B.R. Baum. Combining trees as a way of combining data sets for phylogenetic inference, and the desirability of combining gene trees. *Taxon*, 41:3–10, 1992.

BBL+98 DA Benson, MS Boguski, DJ Lipman, J Ostell, and BF Ouellette. Genbank. *Nucl. Acids Res.*, 26(1):1–7, January 1998.

BBP07 Denis Baurain, Henner Brinkmann, and Herve Philippe. Lack of resolution in the animal phylogeny: Closely spaced cladogeneses or undetected systematic errors? *Mol Biol Evol*, 24(1):6–9, January 2007.

BE04 Olaf R. P. Bininda-Emonds. The evolution of supertrees. *Trends in Ecology and Evolution*, 19:315–322, 2004.

BE05 Olaf R. P. Bininda-Emonds. Supertree construction in the genomic age. *Methods in Enzymology*, 395:745–757, 2005.

BEGS02 Olaf R. P. Bininda-Emonds, John L. Gittleman, and Mike A. Steel. The (super)tree of life: Procedures, problems and prospects. *Annu. Rev. Ecol. Syst.*, 33:265–289, 2002.

BLAa Overview of the different blast programs available for similarity searches `http://www.ch.embnet.org/CoursEMBnet/Exercises03/blast.png`.

BLAb Graphical overview of blast results `http://www.ncbi.nlm.nih.gov/entrez/`.

BLT93 Mark S. Boguski, Todd M.J. Lowe, and Carolyn M. Tolstoshev. dbest [mdash] database for [ldquo]expressed sequence tags[rdquo]. *Nat Genet*, 4(4):332–333, August 1993.

BMA+01 C. Borchiellini, M. Manuel, E. Alivon, N. Boury-Esnault, J. Vacelet, and Y. Le Parco. Sponge paraphyly and the origin of metazoa. *Journal of Evolutionary Biology*, 14(1):171–179, 2001.

BO05 A. D. Baxevanis and B. F. F Ouellette. *Bioinformatics: A pratical guide to the analysis of genes and proteins*. Wiley-Interscience, 2005.

Boo The bootstrap process `http://artedi.ebc.uu.se/course/X3-2004/Phylogeny/Phylogeny-Credibility/`.

BS67 B. G. Barrell and F. Sanger. The sequence of phenylalanine trna from e. coli. *Sanger FEBS Letters*, 3:275–278, 1967.

BV01 Paola Bonizzoni and Gianluca Della Vedova. The complexity of multiple sequence alignment with sp-score that is a metric. *Theoretical Computer Science*, 259(1-2):63–79, May 2001.

Cal Porifera - calcarea `http://www.ucmp.berkeley.edu/porifera/calcarea.html`.

Cas00 J. Castresana. Selection of conserved blocks from multiple alignments for their use in phylogenetic analysis. *Mol Biol Evol*, 17(4):540–552, April 2000.

CAT The cat model `http://www.lirmm.fr/mab/article.php3?id_article=420`.

CHHP00 Benny Chor, Michael D. Hendy, Barbara R. Holland, and David Penny. Multiple maxima of likelihood in phylogenetic trees: An analytic approach. *Mol Biol Evol*, 17(10):1529–1541, October 2000.

cla Clann - supertree software `http://bioinf.may.ie/software/clann/`.

Dar59 C. Darwin. *On the origin of species by means of natural selection or the preservation of favoured races in the struggle for life.* Murray, London., 1859.

dbE dbest - statistics `http://www.ncbi.nlm.nih.gov/dbEST/dbEST_summary.html`.

DDB Dna data bank of japan `http://www.ddbj.nig.ac.jp/Welcome-e.html`.

DeB92 RW DeBry. The consistency of several phylogeny-inference methods under varying evolutionary rates. *Mol Biol Evol*, 9(3):537–551, May 1992.

Dema Porifera - demospongiae (picture) `http://www.palaeos.com/Invertebrates/Porifera/Images/rigida.jpg`.

Demb Porifera - demospongiae `http://www.ucmp.berkeley.edu/porifera/demospongia.html`.

DG04 Richard Desper and Olivier Gascuel. Theoretical foundation of the balanced minimum evolution method of phylogenetic inference and its relationship to weighted least-squares tree fitting. *Mol Biol Evol*, 21(3):587–598, March 2004.

DNA The two-dimensional structure of dna
`http://en.wikipedia.org/wiki/DNA`.

Dog The central dogma of molecular biology
`http://www.accessexcellence.org/RC/VL/GG/images/central.gif`.

dQG06 A. de Queriroz and J. Gatesy. The supermatrix approach to systematics. *Trends in Ecology and Evolution*, 22, 2006.

DRMG02 Matthew W. Dimmic, Joshua S. Rest, David P. Mindell, and Richard A. Goldstein. rtrev: An amino acid substitution matrix for inference of retrovirus and reverse transcriptase phylogeny. *Journal of Molecular Evolution*, 55(1):65–73, July 2002.

DSO78 M. O. Dayhoff, R. M. Schwartz, and B. C. Orcutt. *A model of evolutionary change in proteins*. National Biomedical Research Foundation, 1978.

E-V Definition: E-value
`http://www.ncbi.nlm.nih.gov/Education/BLASTinfo/glossary2.html`.

ECS64 A. W. F. Edwards and L. L. Cavalli-Sforza. *The reconstruction of evolution*. Systematics Association, London, 1964.

Edg04 Robert C. Edgar. Muscle: multiple sequence alignment with high accuracy and high throughput. *Nucl. Acids Res.*, 32(5):1792–1797, 2004.

Efr79 B. Efron. Bootstrapping methods: another look at the jackknife. *Ann. Stat.*, 7:1–26, 1979.

EMB Embl nucleotide sequence database
`http://www.ebi.ac.uk/embl/index.html`.

Fel Felsenstein zone
`http://www.biomedcentral.com/content/figures/1471-2148-5-50-1.jpg`.

Fel81 Joseph Felsenstein. Evolutionary trees from dna sequences: A maximum likelihood approach. *Journal of Molecular Evolution*, 17(6):368–376, November 1981.

Fel85 J. Felsenstein. Confidence intervals on phylogenies: an approach using the bootstrap. *Evolution*, 39:783–791, 1985.

Fel04 J. Felsenstein. *Felsenstein, J.* Sinauer Associates, Sunderland, MA, 2004.

GENa Entrez nucleotide
`http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=Nucleotide`.

GENb Growth of genbank (1982 - 2005)
`http://www.ncbi.nlm.nih.gov/Genbank/genbankstats.html`.

GG03 Stéphane Guindon and Olivier Gascuel. A simple, fast, and accurate algorithm to estimate large phylogenies by maximum likelihood. *Systematic Biology*, 52(5):696–704, 2003.

GK05 Sudhindra R. Gadagkar and Sudhir Kumar. Maximum likelihood outperforms maximum parsimony even when evolutionary rates are heterotachous. *Mol Biol Evol*, 22(11):2139–2141, November 2005.

Gor86 A.D. Gordon. Consensus supertrees: the synthesis of rooted trees containing overlapping sets of labeled leaves. *J. Classif.*, 3:31–39, 1986.

HAN Ncbi - handbook
`http://www.ncbi.nlm.nih.gov/books/bv.fcgi?rid=handbook.section.858`.

Hex Porifera - hexactinellidae
`http://www.ucmp.berkeley.edu/porifera/hexactinellida.html`.

Hil96 D.M. Hillis. Inferring complex phylogenies. *Nature*, 383:130, 1996.

HKY85 Masami Hasegawa, Hirohisa Kishino, and Taka-aki Yano. Dating of the human-ape splitting by a molecular clock of mitochondrial dna. *Journal of Molecular Evolution*, 22(2):160–174, October 1985.

HL03 M. Holder and P. O. Lewis. Phylogeny estimation: traditional and bayesian approaches. *Nature Reviews Genetics*, 4:275–284, 2003.

HLMR02 John P. Huelsenbeck, Bret Larget, Richard E. Miller, and Fredrik Ronquist. Potential applications and pitfalls of bayesian inference of phylogeny. *Systematic Biology*, 51(5):673–688, September 2002.

Hom Definition: Homology
`http://artedi.ebc.uu.se/course/BioInfo-10p-2001/Phylogeny/`.

HP89 M. D. Hendy and D. Penny. A framework for the quantitative study of evolutionarz trees. *Systematic Zoologgy*, 38:297–309, 1989.

Hux74 J. Huxley. *Evolution - The Modern Synthesis*. George Allan & Unwin, London., 1974.

HvS06 J. N. A. Hooper and R. W. M. van Soest. A new species of amphimedon (porifera, demospongiae, haplosclerida, niphatidae) from the capricorn-bunker group of islands, great barrier reef, australia: target species for the sponge genome project. *Zootaxa*, 1314:31–39, 2006.

HyPa Hyphy documentation-nni
`http://www.hyphy.org/docs/analyses/methods/nni.html`.

HyPb Hyphy documentation
`http://www.hyphy.org/docs/analyses/methods/spr.html`.

INS International nucleotide sequence database collaboration
`http://insdc.org/page.php?page=home`.

int Basic sequence analysis using blast
`http://www.bioinfo.no/cookbook/sequences/`.

Kea06 T.M. Keane. *Computational Methods for Statistical Phylogenetic Inference*. PhD thesis, Department of Biology, National University of Ireland, Maynooth, County Kildare, Ireland, 2006.

Kec93 John D. Kececioglu. The maximum weight trace problem in multiple sequence alignment. In *CPM '93: Proceedings of the 4th Annual Symposium on Combinatorial Pattern Matching*, pages 106–119, London, UK, 1993. Springer-Verlag.

KEI+02 Lukasz Kedzierski, Ananias A. Escalante, Raul Isea, Casilda G. Black, John W. Barnwell, and Ross L. Coppel. Phylogenetic analysis of the genus plasmodium based on the gene encoding adenylosuccinate lyase. *Infection, Genetics and Evolution*, 1(4):297–301, July 2002.

KFC+98 Mari Kaellersjoe, James S. Farris, Mark W. Chase, Birgitta Bremer, Michael F. Fay, Christopher J. Humphries, Gitte Petersen, Ole Seberg, and Kaere Bremer. Simultaneous parsimony jackknife analysis of 2538rbcl dna sequences reveals support for major clades of green plants, land plants, seed plants and flowering plants. *Plant Systematics and Evolution*, 213(3):259–287, September 1998.

Kim68 M. Kimura. Evolutionary rate at the molecular level. *Nature*, 217:624–626, 1968.

Kim96 J. Kim. General inconsistency conditions for maximum parsimony: Effects of branch lengths and increasing numbers of taxa. *Syst. Biol.*, 45:363–374, 1996.

Klu98 Arnold G. Kluge. Total evidence or taxonomic congruence: Cladistics or consensus classification. *Cladistics*, 14(2):151–158, 1998.

KMT+00 B. Korber, M. Muldoon, J. Theiler, F. Gao, R. Gupta, A. Lapedes, B. H. Hahn, S. Wolinsky, and T. Bhattacharya. Timing the ancestor of the hiv-1 pandemic strains. *Science*, 288(5472):1789–1796, June 2000.

KW99 J. Kim and T. Warnow. Tutorial on phylogenetic tree estimation, 1999.

Lak91 JA Lake. The order of sequence alignment can bias the selection of tree topology. *Mol Biol Evol*, 8(3):378–385, May 1991.

LBEVC92 B. Lafay, N. Boury-Esnault, J. Vacelet, and R. Christen. An analysis of partial 28s ribosomal rna sequences suggests early radiations of sponges. *Biosystems*, 1992.

LP04 Nicolas Lartillot and Herve Philippe. A bayesian mixture model for across-site heterogeneities in the amino-acid replacement process. *Mol Biol Evol*, 21(6):1095–1109, June 2004.

LP06 N. LARTILLOT and H. PHILIPPE. Computing bayes factors using thermodynamic integration. *Systematic Biology*, 55:195–207, 2006.

ME97 DA Morrison and JT Ellis. Effects of nucleotide sequence alignment on phylogeny estimation: a case study of 18s rdnas of apicomplexa. *Mol Biol Evol*, 14(4):428–441, April 1997.

Men66 G. Mendel. Versuche uber plflanzenhybriden. in verhandlungen des naturforschenden vereines. *Brunn.*, pages 3–47, 1866.

MFDW98 B Morgenstern, K Frech, A Dress, and T Werner. DIALIGN: finding local similarities by multiple sequence alignment. *Bioinformatics*, 14(3):290–294, 1998.

Mic78 M. F. Mickevich. Taxonomic congruence. *Syst. ZooL*, 27:143–158, 1978.

MNL99 Bob Mau, Michael A. Newton, and Bret Larget. Bayesian phylogenetic inference via markov chain monte carlo methods. *Biometrics*, 55(1):1–12, 1999.

MV05 Elchanan Mossel and Eric Vigoda. Phylogenetic mcmc algorithms are misleading on mixtures of trees. *Science*, 309(5744):2207–2209, September 2005.

NC96 Kevin C. Nixon and James M. Carpenter. On simultaneous analysis. *Cladistics*, 12(3):221–241, 1996.

NCB Expressed sequence tags (ests) `http://www.ncbi.nlm.nih.gov/books/bv.fcgi?rid=handbook.section.858`.

NEK99 Johan A. A. Nylander, Christer Erséus, and Mari Källersjö. A test of monophyly of the gutless Phallodrilinae (Oligochaeta, Tubificidae) and the use of a 573-bp region of the mitochondrial cytochrome oxidase i gene in analysis of annelid phylogeny. 28:305–313, 1999.

Not02 Cedric Notredame. Recent progress in multiple sequence alignment: a survey. *Pharmacogenomics*, 3(1):131–144, 2002.

NW70 Saul B. Needleman and Christian D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443–453, March 1970.

OW93 GJ Olsen and CR Woese. Ribosomal rna: a key to phylogeny. *FASEB J.*, 7(1):113–123, January 1993.

phya Phylobayes homepage `http://www.lirmm.fr/mab/`.

Phyb Phylogeny-tree search
`http://artedi.ebc.uu.se/course/BioInfo-10p-2001/Phylogeny/`.

Rag92 M.A. Ragan. Phylogenetic inference based on matrix representation of trees. *Mol. Phylogen. Evol*, 1:53–58, 1992.

RH03 Fredrik Ronquist and John P. Huelsenbeck. Mrbayes 3: Bayesian phylogenetic inference under mixed models. *Bioinformatics*, 19(12):1572–1574, August 2003.

RK03 M. S. Rosenberg and S. Kumar. Taxon sampling, bioinformatics, and phylogenomics. *Syst. Biol.*, 52:119–124, 2003.

RKC05 Antonis Rokas, Dirk Kruger, and Sean B. Carroll. Animal evolution and the molecular signature of radiations compressed in time. *Science*, 310(5756):1933–1938, December 2005.

RLB00 P. Rice, I. Longden, and A. Bleasby. Emboss: The eurpean molecular open software suite. *Trends in Genetics*, 16:276–277, 2000.

RREP07 B. Roure, N Rodriguez-Ezpeleta, and H. Philippe. Scafos: a tool for selection, concatenation and fusion of sequences for phylogenomics. *BMC Evolutionary Biology*, 7(Suppl 1):S2, 2007.

San02 Michael J. Sanderson. Estimating absolute rates of molecular evolution and divergence times: A penalized likelihood approach. *Mol Biol Evol*, 19(1):101–109, January 2002.

SC75 F. Sanger and A. R. Coulson. A rapid method for determining sequences in dna by primed synthesis with dna polymerase. *Journal of Molecular Biology*, 94(3):441–446, May 1975.

SDR$^+$03 Michael J. Sanderson, Amy C. Driskell, Richard H. Ree, Oliver Eulenstein, and Sasha Langley. Obtaining maximal concatenated phylogenetic data sets from large sequence databases. *Mol Biol Evol*, 20(7):1036–1042, July 2003.

SIM Introduction to the similarity page
`http://www.ncbi.nlm.nih.gov/Education/BLASTinfo/similarity.html`.

SM58 R. R. Sokal and C. D. Michener. A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin*, 38:1409–1438, 1958.

SN87 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.

SOWH96 D.L. Swofford, G.J. Olsen, P.J. Waddell, and D.M. Hillis. *Phylogenetic Inference*. Sinauer, Sunderland, Mass, 1996.

SSF$^+$85 RK Saiki, S Scharf, F Faloona, KB Mullis, GT Horn, HA Erlich, and N Arnheim. Enzymatic amplification of beta-globin genomic sequences and restriction site analysis for diagnosis of sickle cell anemia. *Science*, 230(4732):1350–1354, December 1985.

Sta06 Alexandros Stamatakis. Raxml-vi-hpc: maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models. *Bioinformatics*, 22(21):2688–2690, November 2006.

SWF81 T. F. Smith, M. S. Waterman, and W. M. Fitch. Comparative biosequence metrics. *Journal of Molecular Evolution*, 18(1):38–46, January 1981.

The The growth statistics of the largest public databases `http://www.kokocinski.net/bioinformatics/databases.php`.

THG94 Julie D. Thompson, Desmond G. Higgins, and Toby J. Gibson. Clustal w: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucl. Acids Res.*, 22(22):4673–4680, 1994.

Traa Graphical view of trace archive statistics `http://www.ncbi.nlm.nih.gov/Traces/`.

Trab Trace server at ebi `http://www.ncbi.nlm.nih.gov/Traces/trace.cgi`.

Tre The tree of life `http://www.scienceinschool.org/repository/images/issue2tree3_large.jpg`.

Wel00 J. Wells. *Icons of Evolution: Science or Myth?* Regnery Publishing, Washington, D.C., 2000.

WF77 Carl R. Woese and George E. Fox. Phylogenetic structure of the prokaryotic domain: The primary kingdoms. *PNAS*, 74(11):5088–5090, November 1977.

Wie06 John J. Wiens. Missing data and the design of phylogenetic analyses. *Journal of Biomedical Informatics*, 39(1):34–42, February 2006.

WRP+02 Michael Worobey, Andrew Rambaut, Oliver G. Pybus, David L. Robertson, Mark J. Gibbs, John S. Armstrong, and Adrian J. Gibbs. Questioning the evidence for genetic recombination in the 1918 "spanish flu" virus. *Science*, 296(5566):211a–, April 2002.

WTLB01 M. Wilkinson, J.L. Thorley, D.T.J. Littlewood, and R.A. Bray. Towards a phylogenetic supertree of platyhelminthes. *Interrelationships of the Platyhelminthes*, pages 292–301, 2001.

YR97 Z Yang and B Rannala. Bayesian phylogenetic inference using dna sequences: a markov chain monte carlo method. *Mol Biol Evol*, 14(7):717–724, July 1997.

All stated URLs in this thesis are finally reviewed for its accessibility on May 11, 2007. A future accessability of linked contents can not be assured. If any URL is no longer available, copies of contents can be requested by Email: *fab.schreiber@gmail.com*.

Appendix A

Appendix

A.1 The Baurain Data Set

Table A.1: Summary of the frequency of missing data per taxa used by Baurain

OTU	# of AA present	% of AA missing
Acanthoscurria gomesiana	9770	24.5
Acropora millepora	7998	38.2
Apis mellifera	12812	1.0
Argopecten irradians	11431	11.7
Biomphalaria glabrata	11096	14.3
Blastocladiella emersonii	12415	4.1
Bombyx mori	12935	0.1
Boophilus microplus	11542	10.8
Caenorhabditis elegans	12936	0.0
Capitella sp.	11718	9.5
Ciona intestinalis	12879	0.5
Ciona savignyi	12878	0.5
Crassostrea virginica	11893	8.1
Cryptococcus neoformans	12911	0.2
Danio rerio	12795	1.1
Daphnia pulex	12901	0.3
Dugesia ryukyuensis	11778	9.0
Echinococcus granulosus	11242	13.1
Eptatretus burgeri	12502	3.4
Euprymna scolopes	10346	20.1
Fasciola hepatica	5299	59.1
Gallus gallus	11371	12.1
Glomus intraradices	6210	52.0
Helobdella robusta	11631	10.1
Homarus americanus	10339	20.1
Homo sapiens	12942	0.0
Hydractinia echinata	11868	8.3
Hydra magnipapillata	12938	0.0

Hypsibius dujardini	10910	15.7
Ixodes scapularis	10476	19.1
Litopenaeus vannamei	12471	3.6
Locusta migratoria	12336	4.7
Lottia gigantea	8195	36.7
Lumbricus rubellus	12422	4.0
Molgula tectiformis	12715	1.8
Monosiga ovata	12441	3.9
Monosiga brevicollis	11081	14.4
Nasonia vitripennis	10284	20.5
Nematostella vectensis	12030	7.0
Neocallimastix patriciarum	9841	24.0
Petromyzon marinus	12035	7.0
Platynereis dumerilii	6567	49.3
Proterospongia sp.	7293	43.6
Reniera sp.	11671	9.8
Rhizopus oryzae	12888	0.4
Saccharomyces cerevisiae	12910	0.2
Schistosoma mansoni	12770	1.3
Schistosoma japonicum	12364	4.5
Schizosaccharomyces pombe	12906	0.3
Schmidtea mediterranea	12781	1.2
Spodoptera frugiperda	10467	19.1
Suberites domuncula	9773	24.5
Tribolium castaneum	12934	0.1
Ustilago maydis	12908	0.3
Xenopus tropicalis	12942	0.0
Xiphinema index	11010	14.9
Yarrowia lipolytica	12504	3.4
mean	11426	11.7

A.2 The Phylogeny Pipeline - Documentation

The documentation of the single programs of the phylogenetic pipeline as described in Sec. 3.

start.pl

start.pl - This script starts several steps of the phylogenetic analysis

SYNOPSIS

```
start.pl [-m] [-d] (-t) (-n) (-a) (-b) (-u)
```

Options

```
-m      :Method:
                        h       :Homology search
                        d       :Dataset selection
                        m       :Multiple Alignment
                        g       :Selection of Conserved Regions
                        p       :Reconstruction of Phylogeny
-d      :Root Directory of Analysis
-t      :Type of Database Search: (n)r, (e)st, (t)race or (a)ll
-n      :Blast-Database
-p      :Phylogeny Method: (p)hyml, (r)axml, (m)rBayes, or (a)ll
-s      :Type of Dataset
-a      :Alignment Methods:  (c)lustalw, (t)-coffee, (d)ialign, (m)uscle
-b      :Bootstrap option
-u      :Update existing dataset
```

DESCRIPTION

This is the starting script for the phylogeny pipeline. All steps of the analysis can be started using this script. For detailed information see the describtion of the executed programs.

An Example

```
Homology Search: perl start.pl -m h [-d] [-t] (-e) (-n) -u
Dataset Selection: perl start.pl -m d [-d] [-s]
Multiple Alignment: perl start.pl -m m [-d] [-a]
Selection of Conserved Regions: perl start.pl -m g [-d]
Reconstruction of Phylogeny: perl -m p [-d] [-p] (-b);
```

start_blast.pl

start_blast.pl - A script for starting a batch of Blast searches.

SYNOPSIS

```
start_blast.pl (-e) [-d] [-u] (-t) (-e) (-n)
      -e E-value
      -d Root Directory of Analysis (with subdirectories containing query sequences)
      -u (u)pdate existing dataset
      -t Type of Database to search ((n)on-redundant, db(e)st, (t)race or (a)ll)
      -n Name of Blast-Database (for Blast searches against traces)
```

DESCRIPTION

This script starts a batch of Blast searches. The script searches the subdirectories of ”data” for fasta files. For each subdirectory (for each gene) the first sponge entry will be used as the query sequence and a Blast search with the specified E-Value against the specified database(s) is performed. The Blast results are parsed by appropriate parser scripts.

An Example

```
perl start_blast.pl -e 1-e20 -d data -t n
```

Parse_blast_est.pl

parse_blast_est.pl - A Parser for Blast outputs from dbEST searches

SYNOPSIS

```
parse_blast_est.pl [-i] (-f)
    -i Seaching for Sponge hits ?(1=yes,0=no)]
    -f Name of Gene
```

DESCRIPTION

This script parses blast outputs and saves each best hit per organism in a seperate file.

The parser takes one hit for each organism and retrieves the sequence from dbEST. Each single best blast hit is given the prefix ”po”(Porifera) or ”out”(Outgroup) and then stored in a directory ”Name of Gene”.

An Example

```
cat blastresult.blast | parse_blast_est.pl -i 1 -f catalase
```

The blast result in blastresult.blast is input to the parser. Each single best blast hit is given a prefix ”po”(Porifera) and then stored in the directory ”catalase”.

Parse_blast_nr.pl

parse_blast_nr.pl - A Parser for Blast outputs from nr (non-redundant) searches

SYNOPSIS

```
parse_blast_nr.pl [-i] (-f)
    -i Seaching for Sponge hits ?(1=yes,0=no)
    -f Name of Gene
```

DESCRIPTION

This script parses blast outputs and saves each best hit per organism in a seperate file.

The parser takes one hit for each organism and retrieves the sequence from GenBank. Each single best blast hit is given the prefix ”po”(Porifera) or ”out”(Outgroup) and then stored in a directory ”Name of Gene”.

An Example

```
cat blastresult.blast | parse_blast_nr.pl -i 1 -f catalase
```

The blast result in blastresult.blast is input to the parser. The parser takes one hit for each organism and gets the sequence from dbEST. Each single best blast hit is given a prefix ”po”(Porifera) and then stored in a directory ”catalase”.

Parse_blast_trace.pl

parse_blast_trace.pl - A Parser for Blast outputs from local Trace Archive searches

SYNOPSIS

```
parse_blast_trace.pl [-i] (-f)
    -i Seaching for Sponge hits ?(1=yes,0=no)
    -f Name of Gene
```

DESCRIPTION

This script parses blast outputs and saves each best hit per organism in a seperate file.

The parser takes one hit for each organism and gets the sequence from the local Trace file. Each single best blast hit is given the prefix ”po”(Porifera) or ”out”(Outgroup) and then stored in a directory ”Name of Gene”.

An Example

```
cat blastresult.blast | parse_blast_trace.pl -i 1 -f catalase
```

The blast result in blastresult.blast is input to the parser. Each single best blast hit is given a prefix ”po”(Porifera) and then stored in a directory ”catalase”.

Calc_protein_alignments.pl

calc_protein_alignments.pl - A script for starting a batch of multiple alignments

SYNOPSIS

```
calc_protein_alignment.pl [-d] (-m)
      -d Dataset (organised in a directory with subdirectories containing the sequence files
      -m Alignment method: (c)lustalw, (d)ialign, (t)-Coffee, (m)uscle or (a)ll
```

DESCRIPTION

This script starts a batch of multiple alignments. Given a dataset and an alignment method, the script first concatenated all sequences from one subdirectory of the dataset to one fasta file and executes the selected multiple alignment method. The results are saved in a subdirectory of the method that was applied and the dataset that was used.

An Example

```
perl calc_protein_alignments.pl -d fish -m t
```

Calculates T-Coffee alignments for all sequences of the dataset ”fish”.

Compute_trees.pl

Compute_trees.pl - A script for starting the reconstruction of phylogeny for a set of alignments

SYNOPSIS

```
compute_trees.pl (-d) (-m) [-b]
      -d Dataset (organised in a directory with subdirectories containing the sequence files
      -m Phylogeny method: (p)hyml, (r)axml, (m)rbayes
      -b Switches the bootstrap option on/off
```

DESCRIPTION

This script starts the reconstructon of phylogeny for a set of alignments. Each alignment is first converted into phylip and nexus format. Given a dataset and a phylogeny method, the script executes the selected phylogeny method. The results are saved in a subdirectory of the method that was applied and the dataset that was used.

An Example

```
perl compute_trees.pl -d fish -m p -b 1000
```

Calculates the PHYML tree and bootstraps them with 1000 replicates for all alignments of the dataset ”fish”.

Compute_supertree.pl

compute_supertree.pl - A script for the calculation of a supertrees given a set of gene trees.

SYNOPSIS

```
compute_supertree.pl -d -m
      -d      :Directory containing the source trees
      -m      :Supertree method (avcon, mrp, or dfit)
```

DESCRIPTION

The script concatenates all source trees from the directory into a single file and starts the the supertree method using the program clann.

An Example

`perl compute_supertree.pl -d fish -m a`

Computes a supertree with the method ”avcon” from all source trees in the directory ”fish”.

Compute_supermatrix.pl

compute_supermatrix.pl - A script for the calculation of a phylogenetic tree from a supermatrix

SYNOPSIS

```
compute_supermatrix.pl [-d] (-m)
     -d Dataset (organised in a directory with subdirectories containing the sequence files
     -m Phylogeny method: (p)hyml, (r)axml, (m)rbayes
     -b Switches the bootstrap option on/off
```

DESCRIPTION

Given a directory, the script starts the analysis of the supermatrix - contained in the directory - with the the given phylogeny method.

An Example

```
perl compute_supermatrix.pl -d fish -m m
```

Constructs a phylogenetic tree of the supermatrix from the directory ”fish” using the program mrbayes.

A.3 Our Dataset

A.3.1 List of Genes used in the Study

After the reduction of the dataset (as described in Sec. 5.1.2)

Table A.2: Overview of the genes used in this study. In the first column are the abbreviations of the gene names and in the second colum their full scientific names.

Abbr.	Full Gene Name
cct-A	T complex protein 1 alpha subunit
cct-B	T complex protein 1 beta subunit
cct-D	T complex protein 1 delta subunit
cct-E	T complex protein 1 epsilon subunit
cct-G	T complex protein 1 gamma subunit
cct-Z	T complex protein 1 ? subunit
ef2-EF2	Elongation factor EF2
ef2-U5	Elongation factor Tu family U5 snRNP specific protein
hsp70-E	Heat shock 70kDa protein form E
hsp70-mt	Heat shock 70kDa protein, mitochondrial form
hsp70-SSE	Heat shock 70kDa protein subfamily SSE1
if2g	Eukaryotic translation initiation factor 2g
mcm-B	Minichromosome family maintenance protein 2
rpl5	60S ribosomal Protein 5
rps2	40S ribosomal Protein 2

A.3.2 List of Chimerical Operational Taxonomic Units OTUs

Table A.3: Overview of the chimerical operational taxonomic units as used in the study. The most present species are in bold letters.

Chimeric OTU	**Species included**
Acropora millepora	**Acropora millepora** , Acropora palmata, Montastraea faveolata
Argopecten irradiens	**Argopecten irradiens**, Pecten maximus
Biomphalaria glabrata	**Biomphalaria glabrata**, Aplysia californica, Lymnaea stagnalis
Boophilus microplus	**Boophilus microplus**, Rhipicephalus appendiculatus
Daphnia magna	**Daphnia magna**, Daphnia pulex
Dugesia japonica	**Dugesia japonica**, Dugesia ryukyuensis
Eptatretus burgeri	**Eptatretus burgeri**, Myxine glutinosa
Helobdella robusta	**Helobdella robusta** , Haementeria depressa
Homarus americanus	**Homarus americanus**, Pacifastacus leniusculus
Homo sapiens	**Homo sapiens** , Canis familiaris, Mus musculus, Bos taurus, Rattus norvegicus
Hydractinia echinata	**Hydractinia echinata**, Podocoryne carnea
Hypsibius dujardini	**Hypsibius dujardini**, Macrobiotus islandicus, Richtersius coronifer
Litopenaeus vannamei	**Litopenaeus vannamei**, Penaeus monodon, Marsupenaeus japonicus
Lumbricus rubellus	**Lumbricus rubellus**, Eisenia andrei
Molgula tectiformis	**Molgula tectiformis**, Halocynthia roretzi
Platynereis dumerilii	**Platynereis dumerilii**, Nereis virens
Xenopus laevis	**Xenopus laevis**, Xenopus tropicalis

A.3.3 Statistics of the different Datasets

A.3.3.1 ds_10

Table A.4: Overview of the genes used in dataset ds_10, their lengths and the number of OTUs.

Name of Gene	Begin	End	Length	Number of OTUs
cct-A	1	704	704	44
cct-B	705	1250	546	47
cct-D	1251	1839	589	46
cct-E	1840	2426	587	46
cct-G	2427	3078	652	47
cct-Z	3079	3736	658	44
ef2-EF2	3737	4686	950	55
ef2-U5	4687	6052	1366	38
hsp70-E	6053	6876	824	54
hsp70-SSE	6877	8934	2058	44
hsp70-mt	8935	9870	936	55
if2g	9871	10422	552	44
mcm-B	10423	11684	1262	36
rpl5	11685	12030	346	53
rps2	12031	12526	496	52

Table A.5: Overview of the genes used in dataset ds_10_red, their lengths and the number of OTUs.

Name of Gene	Begin	End	Length	Number of OTUs
cct-A	1	704	704	42
cct-B	705	1250	546	45
cct-D	1251	1839	589	43
cct-E	1840	2426	587	43
cct-G	2427	3078	652	43
cct-Z	3079	3736	658	41
ef2-EF2	3737	4686	950	48
ef2-U5	4687	6052	1366	36
hsp70-E	6053	6876	824	43
hsp70-SSE	6877	8934	2058	37
hsp70-mt	8935	9870	936	42
if2g	9871	10422	552	43
mcm-B	10423	11684	1262	35
rpl5	11685	12030	346	47
rps2	12031	12526	496	46

A.3.3.2 ds_40

Table A.6: Overview of the genes used in dataset ds_40, their lengths and the number of OTUs.

Name of Gene	Begin	End	Length	Number of OTUs
cct-B	1	546	546	47
ef2-EF2	547	1496	950	55
ef2-U5	1497	2862	1366	38
hsp70-E	2863	3686	824	54
hsp70-mt	3687	4622	936	55
if2g	4623	5174	552	44
mcm-B	5175	6436	1262	36
rpl5	6437	6782	346	53
rps2	6783	7278	496	52

Table A.7: Overview of the genes used in dataset ds_40_red, their lengths and the number of OTUs.

Name of Gene	Begin	End	Length	Number of OTUs
cct-B	1	546	546	44
if2g	547	1098	552	44
mcm-B	1099	2360	1262	35
rpl5	2361	2706	346	47
rps2	2707	3202	496	47

A.3.3.3 ds_10_pam

Table A.8: Overview of the genes used in dataset ds_10_pam, their lengths and the number of OTUs.

Name of Gene	Begin	End	Length	Number of OTUs
cct-A	1	704	704	44
cct-B	705	1250	546	47
cct-D	1251	1839	589	46
cct-E	1840	2426	587	46
cct-Z	2427	3084	658	44
ef2-U5	3085	4450	1366	38
hsp70-E	4451	5274	824	54
hsp70-mt	5275	6250	976	55
if2g	6251	6802	552	44
mcm-B	6803	8064	1262	36
rpl5	8065	8410	346	53
rps2	8411	8906	496	52

Table A.9: Overview of the genes used in dataset ds_10_pam_red, their lengths and the number of OTUs.

Name of Gene	Begin	End	Length	Number of OTUs
cct-A	1	704	704	39
cct-B	705	1250	546	42
cct-D	1251	1839	589	38
cct-E	1840	2426	587	40
cct-Z	2427	3084	658	40
ef2-U5	3085	4450	1366	34
hsp70-E	4451	5274	824	38
hsp70-mt	5275	6250	976	39
if2g	6251	6802	552	41
mcm-B	6803	8064	1262	34
rpl5	8065	8410	346	43
rps2	8411	8906	496	43

A.3.3.4 ds_40_pam

Table A.10: Overview of the genes used in dataset ds_40_pam, their lengths and the number of OTUs.

Name of Gene	Begin	End	Length	Number of OTUs
cct-B	1	546	546	47
ef2-EF2	547	1496	950	55
ef2-U5	1497	2855	1359	37
hsp70-E	2856	3679	824	54
hsp70-mt	3680	4655	976	55

Table A.11: Overview of the genes used in dataset ds_40_pam_red, their lengths and the number of OTUs.

Name of Gene	Begin	End	Length	Number of OTUs
cct-B	1	546	546	43
ef2-EF2	547	1496	950	45
ef2-U5	1497	2855	1359	35
hsp70-E	2856	3679	824	41
hsp70-mt	3680	4655	976	42

Acknowledgments

I thank Prof. Dr. Burkhard Morgenstern and Jun-Prof. Dr. Gert Wörheide for the idea, supervision and support of this project.
I also thank my department, the department of bioinformatics, for their support and taking me up so kindly as a new member of the group.
I am grateful to Markus Hsi-Yang Fritz for proofreading this work.
I would like to express my gratitude to my parents for patiently financing my studies and supporting me all the time.
Last but not least, i would like to thank my girlfriend, Pascale, for her love and continuous support.