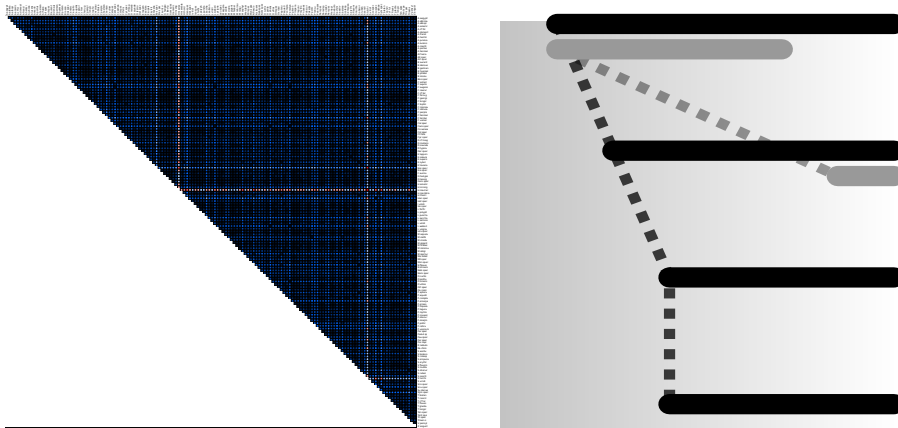


# Homology Assessment in Molecular Phylogenetics

Evaluation, Improvement, and Influence of Data Quality  
on Tree Reconstruction



## Dissertation

zur Erlangung des Doktorgrades (Dr. rer. nat.)  
der Mathematisch-Naturwissenschaftlichen Fakultät  
an der Rheinischen Friedrich-Wilhelms-Universität Bonn

vorgelegt von  
Patrick Kück  
aus Bonn

September 2011

Angefertigt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Rheinischen Friedrich-Wilhelms-Universität Bonn.



Die Dissertation wurde am Zoologischen Forschungsmuseum Alexander Koenig (ZFMK), Bonn durchgeführt.

1. Prüfer: Prof. Dr. Johann-Wolfgang Wägele
2. Prüfer: Prof. Dr. Bernhard Misof
3. Prüfer: Prof. Dr. Wolfgang Alt
4. Prüfer: Prof. Dr. Jes Rust

Tag der Mündlichen Prüfung: 09.12.2011

Erscheinungsjahr: 2012

*“If phylogenetic inference is to be a science, we must consider its methods guilty until proven innocent.”*

(Joseph Felsenstein, 1978)

For the two people I love most.  
Thank you for showing me the beauty of life.

MY SON LUCA & MY GIRLFRIEND BIRTHE.

# Contents

<b>1</b>	<b>General Introduction</b>	<b>1</b>
1.1	The Importance of Alignment Quality . . . . .	1
1.2	Alignment Algorithms – Advantages and Disadvantages . . . . .	2
1.2.1	Progressive algorithms . . . . .	3
1.2.2	Consistency based algorithms . . . . .	3
1.2.3	Incorporation of secondary structure information . . . . .	4
1.2.4	Phylogenomic data . . . . .	4
1.3	Alignment Masking – An interface between alignment and tree reconstruction . . . . .	4
1.4	The Ant Tree of Life – Grown on different alignments seeded from identical data . . . . .	5
1.5	AliGROOVE – Phylogenetic topologies in the light of alignment quality . . . . .	5
1.6	Learning by Doing – Simulations to search for systematic errors . . . . .	6
1.7	New software tools . . . . .	7
<b>2</b>	<b>Masking of randomness in sequence alignments can be improved and leads to better resolved trees</b>	<b>9</b>
2.1	Introduction . . . . .	11
2.2	Methods . . . . .	12
2.2.1	Data sets . . . . .	12
2.2.2	Alignments . . . . .	13
2.2.3	Split Networks . . . . .	13
2.2.4	Tree reconstructions . . . . .	14
2.3	Results . . . . .	14
2.3.1	ALISCORE algorithm for amino acid data . . . . .	14
2.3.2	Testing performance on real data . . . . .	15
2.4	Discussion . . . . .	18
2.5	Additional Files . . . . .	23
<b>3</b>	<b>Improved phylogenetic analyses corroborate a plausible position of <i>Martialis heureka</i> in the ant tree of life</b>	<b>25</b>
3.1	Introduction . . . . .	27
3.2	Materials and Methods . . . . .	28
3.2.1	Data set . . . . .	28
3.2.2	Alignment . . . . .	28
3.2.3	Phylogenetic reconstructions . . . . .	29
3.3	Results . . . . .	30
3.3.1	Alignment masking, number of bootstrap replicates and likelihood scores . . . . .	30
3.3.2	Phylogenetic relationships . . . . .	32

---

3.4	Discussion . . . . .	36
3.5	Additional Files . . . . .	38
<b>4</b>	<b>AliGROOVE: a new tool to visualize the extent of sequence similarity and alignment ambiguity in multiple alignments</b>	<b>41</b>
4.1	Introduction . . . . .	43
4.1.1	AliGROOVE algorithm . . . . .	45
4.2	Material and Methods . . . . .	47
4.2.1	Simulated data . . . . .	47
4.2.2	Empirical data . . . . .	47
4.3	Results . . . . .	48
4.3.1	Testing performance on simulated data . . . . .	48
4.3.2	Testing performance on empirical data . . . . .	51
4.4	Discussion . . . . .	51
<b>5</b>	<b>Long branch effects distort Maximum Likelihood phylogenies in simulations despite selection of the correct model</b>	<b>55</b>
5.1	Introduction . . . . .	57
5.2	Material and Methods . . . . .	58
5.2.1	Simulations . . . . .	58
5.2.2	Maximum Likelihood Analyses . . . . .	61
5.2.3	Scoring . . . . .	61
5.3	Results . . . . .	61
5.3.1	Topology A . . . . .	62
5.3.2	Topology B . . . . .	62
5.3.3	Maximum Likelihood Values . . . . .	62
5.4	Discussion . . . . .	67
5.5	Additional Files . . . . .	68
<b>6</b>	<b>Developed Software and help scripts (published/unpublished)</b>	<b>71</b>
6.1	FASconCAT: Convenient handling of data matrices . . . . .	71
6.1.1	Introduction . . . . .	71
6.1.2	Concatenation of data . . . . .	72
6.1.3	Data conversion . . . . .	74
6.1.4	Discussion . . . . .	74
6.2	ALICUT . . . . .	76
6.3	BHoEMe . . . . .	76
6.4	SusEX . . . . .	77
6.5	ESTa . . . . .	77
6.6	TaxEd . . . . .	77
6.7	LoBraTe . . . . .	77
6.8	RAxTAX . . . . .	77
6.9	SecSITE . . . . .	78
6.10	SPIPES . . . . .	78

---

6.11 Additional Files . . . . .	78
<b>7 General Discussion</b>	<b>81</b>
7.1 The Effect of Alignment Masking on Phylogenetic Analyses . . . . .	81
7.2 The Effect of Long Branches and chosen Model Parameters on Maximum Likelihood Reconstructions . . . . .	82
7.3 Perspectives . . . . .	83
<b>Bibliography</b>	<b>87</b>
<b>A Additional Information Chapter 3</b>	<b>103</b>
A.1 Bayesian majority rule consensus topologies . . . . .	103
A.2 Maximum Likelihood majority rule consensus topologies . . . . .	106
<b>B LoBraTe</b>	<b>109</b>
B.1 Flowchart of the LoBraTe Process Pipeline . . . . .	109
<b>C RAxTAX</b>	<b>111</b>
C.1 Flowchart of the RAxTAX Process Pipeline . . . . .	111
<b>D Manual FASconCAT</b>	<b>113</b>
D.1 Introduction . . . . .	113
D.2 Usage/Options . . . . .	114
D.2.1 Start FASconCAT via menu . . . . .	114
D.2.2 Start FASconCAT via single command line . . . . .	115
D.2.3 Options . . . . .	116
D.3 Internals . . . . .	120
D.3.1 Input/Output . . . . .	120
D.3.2 Computation time . . . . .	120
D.3.3 Error reports . . . . .	122
D.4 Important Notes . . . . .	125
D.5 License/Help-Desk/Citation . . . . .	126
<b>E Manual ALICUT</b>	<b>127</b>
E.1 Introduction . . . . .	127
E.2 Usage/Options . . . . .	128
E.2.1 Start ALICUT via menu . . . . .	128
E.2.2 Start ALICUT via single command line . . . . .	128
E.2.3 Additional Information files . . . . .	130
E.3 Input/Output . . . . .	130
E.4 License/Help-Desk/Citation . . . . .	132
E.5 Copyright . . . . .	132

---

<b>F</b>	<b>Short Documentation ESTa</b>	<b>133</b>
F.1	General Information . . . . .	133
F.2	General Usage . . . . .	133
F.3	Output-files . . . . .	133
F.3.1	“EST-request_(NCBI).txt” . . . . .	133
F.3.2	“New_EST_entries.txt” . . . . .	134
<b>G</b>	<b>List of Abbreviations</b>	<b>135</b>
<b>H</b>	<b>List of Electronic Supplementary Files</b>	<b>137</b>
<b>I</b>	<b>Summary</b>	<b>139</b>
<b>J</b>	<b>Erklärung</b>	<b>141</b>

# General Introduction

---

## Contents

---

<b>1.1</b>	<b>The Importance of Alignment Quality . . . . .</b>	<b>1</b>
<b>1.2</b>	<b>Alignment Algorithms – Advantages and Disadvantages . .</b>	<b>2</b>
1.2.1	Progressive algorithms . . . . .	3
1.2.2	Consistency based algorithms . . . . .	3
1.2.3	Incorporation of secondary structure information . . . . .	4
1.2.4	Phylogenomic data . . . . .	4
<b>1.3</b>	<b>Alignment Masking – An interface between alignment and tree reconstruction . . . . .</b>	<b>4</b>
<b>1.4</b>	<b>The Ant Tree of Life – Grown on different alignments seeded from identical data . . . . .</b>	<b>5</b>
<b>1.5</b>	<b>AliGROOVE – Phylogenetic topologies in the light of alignment quality . . . . .</b>	<b>5</b>
<b>1.6</b>	<b>Learning by Doing – Simulations to search for systematic errors . . . . .</b>	<b>6</b>
<b>1.7</b>	<b>New software tools . . . . .</b>	<b>7</b>

---

## 1.1 The Importance of Alignment Quality

The final goal of every phylogenetic analysis is to reconstruct most efficiently taxon relationships from underlying data. Yet, little attention has been paid to the role of alignment accuracy and its impact on tree reconstruction [1]. The success of phylogenetic analyses depends strongly on the algorithmic assumptions of the primary and secondary homology assessment [2]. The primary homology assessment in molecular phylogenetics comprises two main steps: i) the identification of homologous sequences, and ii) the classification of positional homology among them [1,2]. In both steps of the primary assessment, alignment algorithms are used to determine the respective homology hypotheses. In the first step of the primary homology assessment, similar sequences are identified through sequence comparisons by alignment algorithms like BLAST (Basic Local Alignment Search Tool) [3–5]. Subsequently, efficient alignment algorithms are used to allocate positional similarity among sequences [2]. Therefore, multiple sequence alignments are statements of primary homology in phylogenetic analyses [1,2,6,7].



Unfortunately, similarity and homology are not necessarily corresponding [2, 8]. Similarity of character states can either be due to common ancestry or due to convergence [2, 9, 10]. Alignment algorithms can not differentiate between positional similarity of sequences and evolutionary homology. Most of them rely by necessity on maximizing sequence similarity [2], but the maximization of sequence similarity without considering evolutionary homology can lead to incorrectly aligned sequence positions due to random similarity among sequences.

The primary homology assessment forms the basis for the derivation of the secondary homology hypotheses, a result of the tree reconstruction process. Due to the dependence of tree reconstruction on the primary homology assessment, the influence of incorrect alignment sections can not be completely corrected by tree model algorithms [2]. As consequence, incorrect alignment sections can distort tree topologies even if model assumptions are chosen correctly. It is important to keep in mind that each alignment itself is a set of homology hypotheses. Some hypotheses are correct, others might not [11]. The degree of alignment accuracy is strongly influenced by the chosen alignment algorithm and its parameter settings. As mentioned by Ogden and Rosenberg [1], topological accuracy decreases if alignment errors increase. Ogden and Rosenberg [1] have also shown that alignment inaccuracy has a stronger negative impact on tree reconstruction if data sets are derived from more pectinate topologies with unequal branch lengths than on balanced, ultrametric topologies with equal branch lengths. For that reason, it can be concluded that the success of phylogenetic analyses depends as strong on alignment accuracy than on model assumptions of the phylogenetic reconstruction itself (e.g. [1, 12–15]). Alignment quality should therefore receive the most possible attention and concern in phylogenetic analyses.

## 1.2 Alignment Algorithms – Advantages and Disadvantages

There are different kinds of sequence data in molecular phylogenetics. Some data consist of conservative or highly variable sequences, others of sequences with highly variable and conserved sequence regions (mosaic genes like ribosomal structure genes). Some sequences are long, others are short, some data sets have missing data, like EST libraries, and some display a much higher degree of substitution rates than others. To infer positional homology among sequences, alignment algorithms have to convert raw sequences of different length to sequences of equal length (or raw sequences of equal length to longer sequences of equal length) [1]. For this purpose, alignment algorithms have to place gaps to compensate insertions and deletions among sequences [1]. The decision to place a gap or not depends on the respective algorithm and its setup.

The diversity of different published alignment algorithms is enormous. For that reason, it is important to find an appropriate alignment algorithm and an appropriate parameter setup most suitable for the respective data set. Over the last two

decades, the area of multiple sequence alignments (MSA) has undergone a major transformation [16]. Especially progressive, iterative optimization strategies and the use of consistency-based scoring algorithms have become mainstream trends in phylogenetics [16].

### 1.2.1 Progressive algorithms

Progressive alignment algorithms [17–19] consist of simple, but computationally very efficient alignment heuristics [16]. They align given sequences pairwise to each other in the order given by a pre-calculated distance topology [2, 16] and are implemented in most recent alignment methods [16, 20], like ClustalW [21], HMMER [22], MUSCLE [23], MAFFT [24], and T-COFFEE [20]. A main disadvantage of the progressive algorithm is that sequences once aligned will not be re-aligned in the further alignment progress, even if sequences later added stand in conflict with previously aligned ones [2, 16]. This is especially a problem of progressive, non-iterative alignment methods like ClustalW [21]. Progressive, iterative methods like MAFFT [24] or MUSCLE [23] re-align each sequence of a multiple sequence alignment on the basis of a new topology until the iteration steps consistently fail to improve the alignment [2, 16, 23, 24]. The implementation of iterative alignment steps to progressive algorithms has led to a strong improvement of alignment accuracy in benchmark tests [2, 25, 26]. Another disadvantage of all progressive alignments lies in the use of predefined gap penalties. Different penalty values of mismatch, gap opening, gap extension, and affine gap costs can lead to different alignments [11].

### 1.2.2 Consistency based algorithms

Consistency based alignment algorithms try to find the alignment that agrees the most with different pairwise alignments [16]. T-COFFEE [20] for example, a progressive, consistency based alignment method, creates a primary library of weights relative to pairwise sequence identity obtained from a global (ClustalW [21]) and a local (Lalign [27]) alignment. Followed by an extension phase, T-COFFEE generates an extended library of final weights to find the multiple alignment that best fits the alignments in the primary library [16, 20, 25]. Afterwards, the T-COFFEE algorithm uses the information of the extended library to make a progressive alignment which considers all single executed pairwise alignments [20]. Another consistency based alignment method is Dialign-T, a segment-based alignment approach. Dialign-T combines also local and global alignment features [28, 29], but aligns only statistically significant and consistent similarities of sequences. Sequence parts without observable similarity at the primary sequence level are left unaligned [30].

While T-COFFEE seems to perform better for global alignments, Dialign-T tends to produce better local ones [30]. An advantage of both methods towards progressive, iterative or non-iterative alignments is the avoidance of arbitrary gap costs. Furthermore, both consistency based methods have shown good performance in many benchmark tests [16, 25, 26, 29, 30], but did not outperform iterative, progres-

sive alignment methods. For example, T-COFFEE performed "detectably worse" in a benchmark test of Morrison [2] if sequence identity was lower than 50%. A main disadvantage of consistency based approaches is their high need of computational memory [16, 30]. The use of T-COFFEE, for example, is actually limited to 50 taxa on a normal desktop computer [20].

### 1.2.3 Incorporation of secondary structure information

Another new generation of alignment methods like MXSCARNA [31] or RNAsalsa [32] includes functional information of secondary structure sequences into the alignment process. For genes with conserved secondary RNA structure, e.g. ribosomal RNA genes, it was shown that an inclusion of secondary structure information can lead to considerably improved alignment quality [33].

### 1.2.4 Phylogenomic data

Phylogenomic sequences pose further challenges: i) Large sequence size makes it impossible to apply standard alignment methods where computation time is proportional to sequence length, and ii) genomic rearrangements have to be taken into account [30].

## 1.3 Alignment Masking – An interface between alignment and tree reconstruction

As described in section 1.1 and 1.2, no alignment method is perfect, because all methods have to use heuristics [16]. As mentioned, the best choice of an appropriate alignment method is not only dependent on the alignment algorithm itself, but also on the chosen gap penalty values [11]. Highly variable sequence regions (e.g. loop regions of secondary structure genes) are more difficult to align. The same applies to sequences of unequal lengths or to data sets which contain a high amount of missing data (e.g. EST data). Random sequence similarity due to convergent character states of strongly derived taxa can also reduce alignment quality and therefore distort the identification of positional homologies [34, 35].

As described in section 1.1, random sequence similarity or ambiguously aligned sequence regions are derived from the primary homology assessment. As a consequence of the dependence on the primary assessment, the effect of erroneously aligned sequence sections cannot be fully compensated by the tree reconstruction method. Therefore, ambiguously aligned sequence sections and random sequence similarity can negatively influence phylogenetic reconstructions and lead to defective estimation of substitution model parameters [34]. Especially if data sets are very large (e.g. phylogenomic data), the negative alignment effects on model estimation and tree reconstructions do not disappear, but become evident more intensely [34]. Therefore, it is necessary to detect and remove erroneously aligned sections before tree reconstruction. Alignment masking approaches are methods which meet this

requirement. The effect of two masking methods, ALISCORE [34] (a parametric approach) and GBLOCKS [36] (a non parametric approach), on alignment quality and tree reconstruction is described in chapter 2: "Masking of randomness in sequence alignments can be improved and leads to better resolved trees". This section gives furthermore the first comprehensive characterisation of the most recent amino-acid masking algorithm implemented in ALISCORE [35].

## **1.4 The Ant Tree of Life – Grown on different alignments seeded from identical data**

Despite the attempts to propose a robust sister group of all extant ants [37–41], it is still doubtful which ant subfamily constitutes the first split in the ant tree of life. Rabeling et al. [40] presented a Bayesian tree with resolved single inter- and intra subfamily relationships and a nearly unresolved Maximum Likelihood topology which proposed Martialinae as the earliest branch within the ant tree of life. While the position of Martialinae was highly supported by Bayesian analyses, the best Maximum Likelihood tree could resolve this placement only with moderate bootstrap support. Previous molecular studies had proposed the subfamily Leptanillinae as a sister group of all other extant ants [37–39]. Rabeling et al. [40] did not name the used alignment method, nor the way in which they identified an excluded ambiguously aligned sequence section before tree reconstruction. Therefore, it is possible that the placement of Martialinae suggested by Rabeling et al. [40] could be due to i) inferior sequence alignments or confounding effects of randomized alignment sections, or ii) an insufficient number of bootstrap replicates (ML approach) and/or an insufficient number of Bayesian generations.

Chapter 3, "Improved phylogenetic analyses corroborate a plausible position of *Martialis heureka* in the ant tree of life", describes a re-analysis of Rabeling et al.'s data. The re-analysis is coupled with parametric alignment masking and thoroughly performed phylogenetic analyses which comes to different conclusions for the ant tree of life than Rabeling et al. [40]. The study of chapter 3 is another example about the positive impact of alignment masking on data quality and gives an impression of how results from the tree reconstruction should be handled.

## **1.5 AliGROOVE – Phylogenetic topologies in the light of alignment quality**

As shown in chapter 2 and 3, alignment masking increases tree-likeness of given data by reducing the influence of data noise on tree reconstructions. However, while masking methods are commonly efficient in detecting ambiguously aligned sequence blocks, all methods more or less lack the ability to detect heterogeneous sequence divergence within sequence alignments. The sliding window approach of ALISCORE as described by Misof and Misof [34] and Kück et al. [35], for example

is unable to identify randomized alignment blocks if ambiguously aligned positions are not present in more than  $\approx 20\%$  of sequences [34]. This is a main disadvantage of masking approaches, because undetected heterogeneous sequence divergence can result in a strong bias in tree reconstructions, like long branch attraction (first described by Felsenstein [42] on a four taxon case).

AliGROOVE implements an adaption of the ALISCORE masking algorithm which can help to detect strongly derived sequence regions that can have a negative influence on tree reconstruction methods. Therefore, the AliGROOVE algorithm provides the possibility to highlight taxa which will most likely be misplaced in trees and thus negatively influence the tree-likeness of given data. Chapter 4, "AliGROOVE: a new tool to visualize the extent of sequence similarity and alignment ambiguity in multiple alignments", gives a detailed description of the AliGROOVE algorithm and the possibility of tagging branches as an indirect estimation of reliability of a subset of possible splits guided by a topology. The performance of the AliGROOVE algorithm was tested on simulated and empirical data. First test results are already shown and discussed in chapter 4.

## 1.6 Learning by Doing – Simulations to search for systematic errors

Considering the tree reconstruction process, the first task is the choice of an appropriate tree reconstruction method. The method should be robust to model violations and efficiently recover the topology of the underlying tree [43]. There are four main groups of reconstruction methods which are commonly used in phylogenetic analyses: Neighbor Joining, Maximum Parsimony, Maximum Likelihood (ML) and Bayesian approaches. Maximum Likelihood and Bayesian analyses are normally more accurate in tree reconstruction than Maximum Parsimony and Neighbor Joining methods [1].

Maximum Likelihood and Bayesian analyses clearly outperform Maximum Parsimony if the data include heterogeneous or heterotachous substitution rates [44–46]. Maximum Parsimony does not account for multiple substitutions and among-site rate variation (ASRV) of substitution rates and becomes inconsistent if evolutionary rates are heterogeneous. This applies especially for distantly related sequences [47, 48]. Although statistical properties of Maximum Parsimony are not completely understood, it is commonly assumed that Maximum Parsimony will find the correct topology under a finite number of characters when the evolutionary rate is constant [49]. Nevertheless, Maximum Parsimony can be inconsistent under that condition, because the probability of a single substitution on a short interior branch is often lower than multiple parallel substitutions on longer branches [50]. This case of inconsistency is true with a small extent of sequence divergence, too.

However, examining theoretical studies and comparative tests on Maximum Likelihood and Bayesian analyses, Maximum Likelihood turns out as the first choice for phylogenetic tree reconstructions. As mentioned in chapter 3, Bayesian analyses

tend to overestimate signal and give high support values even if the data is uninformative [51, 52]. It is shown from simulated data, that Bayesian analyses have a much higher type I error rate than Maximum Likelihood, especially in cases of model misspecification [52]. Another disadvantage of Bayesian analyses is the unknown influence of subjective prior assumptions on Bayesian tree reconstructions [53].

Chapter 5: "Long branch effects distort Maximum Likelihood phylogenies in simulations despite selection of the correct model" shows that the success of Maximum Likelihood depends not only on the degree of alignment quality, but also on the relation of branch length differences of underlying topologies. This is especially the case if branch length relations are strongly divergent in the true topology that shall be reconstructed. To avoid long branch effects it is important to know the influence of internal and terminal long branches on Maximum Likelihood behavior under various model violations. The study of chapter 5 tested the robustness of Maximum Likelihood towards different classes of long branch effects in multiple taxon topologies. To test the robustness of Maximum Likelihood, one must know the true evolutionary history of sequences. Therefore, the study of chapter 5 used simulated fixed data sets under two different 11-taxon trees and a broad range of different branch length conditions to infer the reconstruction success of Maximum Likelihood with sequence alignments of different length. The data was then re-analysed with Maximum Likelihood under i) true-, ii) estimated-, and iii) violated model assumptions about among-site rate variation. Simulation studies have previously been used by numerous studies to examine tree reconstruction success under various conditions (e.g. [43, 52, 54–61]), but the study of long branch effects is new.

Although the simulation study of chapter 5 gives no information on the impact of alignment accuracy on tree reconstruction, it shows the influence of branch length differences on tree reconstruction if the underlying alignment is completely correct. As perfect alignments will never be available in reality, it can be suspected, that the negative effects of incorrect model assumptions on tree reconstruction will be much more dramatic in empirical data.

## 1.7 New software tools

The realization of the studies described in chapter 2–5 would not have been possible without the development of numerous scripts. Smaller scripts were used for data handling like data extraction, data summary, data concatenation, data conversion, or program execution. Larger pipeline scripts were needed to execute complete data analyses, starting with data simulation, performing of phylogenetic analyses, till data evaluation and result plotting. All programs developed for this thesis are written in Perl. Some of the most important scripts and pipelines which have been written for the accomplishment of this thesis or which have been written for other studies are listed and described in chapter 6: "Developed Software and help scripts (published/unpublished)" and attached as electronic appendix.



# Masking of randomness in sequence alignments can be improved and leads to better resolved trees

---

## Contents

---

<b>2.1</b>	<b>Introduction</b>	<b>11</b>
<b>2.2</b>	<b>Methods</b>	<b>12</b>
2.2.1	Data sets	12
2.2.2	Alignments	13
2.2.3	Split Networks	13
2.2.4	Tree reconstructions	14
<b>2.3</b>	<b>Results</b>	<b>14</b>
2.3.1	ALISCORE algorithm for amino acid data	14
2.3.2	Testing performance on real data	15
<b>2.4</b>	<b>Discussion</b>	<b>18</b>
<b>2.5</b>	<b>Additional Files</b>	<b>23</b>

---



---

**Abstract:** Methods of alignment masking, which refers to the technique of excluding alignment blocks prior to tree reconstructions, have been successful in improving the signal-to-noise ratio in sequence alignments. However, the lack of formally well defined methods to identify randomness in sequence alignments has prevented a routine application of alignment masking. In this study, we compared the effects on tree reconstructions of the most commonly used profiling method (GBLOCKS) which uses a predefined set of rules in combination with alignment masking, with a new profiling approach (ALISCORE) based on Monte Carlo resampling within a sliding window, using different data sets and alignment methods. While the GBLOCKS approach excludes variable sections above a certain threshold which choice is left arbitrary, the ALISCORE algorithm is free of a *priori* rating of parameter space and therefore more objective.

ALISCORE was successfully extended to amino acids using a proportional model and empirical substitution matrices to score randomness in multiple sequence alignments. A complex bootstrap resampling leads to an even distribution of scores of randomly similar sequences to assess randomness of the observed sequence similarity. Testing performance on real data, both masking methods, GBLOCKS and ALISCORE, helped to improve tree resolution. The sliding window approach was less sensitive to different alignments of identical data sets and performed equally well on all data sets. Concurrently ALISCORE is capable of dealing with different substitution patterns and heterogeneous base composition. ALISCORE and the most relaxed GBLOCKS gap parameter setting performed best on all data sets. Correspondingly Neighbor-Net analyses showed the most decrease in conflict.

Alignment masking improves signal-to-noise ratio in multiple sequence alignments prior to phylogenetic reconstruction. Given the robust performance of alignment profiling, alignment masking should routinely be used to improve tree reconstructions. Parametric methods of alignment profiling can be easily extended to more complex likelihood based models of sequence evolution which opens the possibility of further improvements.

**Keywords:** Alignment Masking, ALISCORE, GBLOCKS, Data Quality, Signal-to-Noise Ratio

---

## 2.1 Introduction

Multiple sequence alignments are an essential prerequisite in alignment based phylogenetic reconstructions, because they establish fundamental homology assessments of primary sequence characters. In consequence, alignment errors can influence the correctness of tree reconstructions [1, 62, 63]. To deal with this problem at the level of sequence alignment, different approaches and alignment software tools have been developed, but despite major advances, alignment quality is still mostly dependent on arbitrary user-given parameters, e.g. gap costs, and inherent features of the data [2, 16]. In particular when sequences are highly divergent and/or length variable, sequence alignment and the introduction of gaps become a more and more complex enterprise and can currently not be fully governed by formal algorithms. The major problem is that finding the most accurate alignment parameters in progressive and consistency based alignment approaches is difficult due to the incomplete knowledge of the evolutionary history of sequences and/or heterogeneous processes along sequences [25]. As a result, problematic sequence alignments will contain sections of ambiguous indel positions and random similarity.

To improve the signal-to-noise ratio, a selection of unambiguous alignment sections can be used. It has been shown that a selection of unambiguously aligned sections, or alignment masking [64], improves phylogenetic reconstructions in many cases [62, 65, 66]. However, a formally well defined criterion of selecting unambiguous alignment sections or profiling multiple sequence alignments was not available. To fill this gap, different automated heuristic profiling approaches of protein and nucleotide alignments have been developed. GBLOCKS [36] is currently the most frequently used tool. The implemented method is based on a set of simple predefined rules with respect to the number of contiguous conserved positions, lack of gaps, and extensive conservation of flanking positions, suggesting a final selection of alignment blocks more “suitable” for phylogenetic analysis [36, 67]. The approach does not make explicit use of models of sequence evolution and is subsequently referred to as a “non-parametric” approach.

The recently introduced alternative profiling method, ALISCORE [34], identifies randomness in multiple sequence alignments using parametric Monte Carlo resampling within a sliding window and was successfully tested on simulated data. ALISCORE was first developed for nucleotide data, but has been extended here to amino acid sequences. The program is freely available from <http://aliscore.zfnk.de>. In short, within a sliding window an expected similarity score of randomized sequences is generated using a simple match/mismatch scoring for nucleotide or an empirical scoring matrix for amino acid sequences (see Methods), actual base composition, and an adapted Poisson model of site mutation. The observed similarity score is subsequently compared with the expected range of similarity scores of randomized sequences. Like GBLOCKS it is independent of tree reconstruction methods, but also independent of a *a priori* rating of sequence variation within a multiple sequence alignment. Because of its explicit use of, although rather simple, models of sequence evolution, ALISCORE can be called a parametric method of alignment masking.

Table 2.1: **Data sets used for analyses.** mtI: mitochondrial data set I; mt II: mitochondrial data set II; EST: EST data set; 12S + 16S rRNA: mitochondrial ribosomal data set. Type: Kind of sequence type. AA: Amino acid sequences; NUC: Nucleotide sequences. N genes: Number of genes per data set. N species: Number of species per data set. N cons. clades: Number of considered clades (selected). Data source: dbEST: EST database of NCBI; unpublished sequences provided by KM (K. Meusemann), BMvR (B. Reumont), FR (F. Roeding), TB (T. Burmester) and JD (J. Dambach).

Data set	Type	N genes	Taxon	N species	N cons. clades	Data source
mtI	AA	11	Eukaryota	17	12	NCBI/SwissProt
mtII	AA	5	Eukaryota	24	15	NCBI/SwissProt
EST	AA	51	Arthropoda	26	7	dbEST; KM/BMvR/FR/TB
12S + 16S	NUC	2	Arthropoda	63	9	NCBI/JD

It has been demonstrated that both methods correctly identify randomness in sequence alignments, although to a very different extent [34, 36, 67]. A comparison of their performance on real data is however missing. Both masking methods suggest a set of alignment blocks suitable for tree reconstructions. These alignment blocks should have a better signal-to-noise ratio and this should lead to better resolved trees and increased support values. Therefore, we used these predictions to assess the performance of both masking methods by comparing reconstructed Maximum Likelihood (ML) trees. Additionally, our analyses compared the sensitivity of tree reconstruction given both profiling approaches in relation to different data and alignment methods. Different test data sets were aligned with commonly used alignment software (CLUSTALX 1.81 [21], MAFFT 6.240 [24], MUSCLE 3.52 [23], T-COFFEE 5.56 [20], and PCMA 2.0 [68]).

For protein alignments, we used two data sets of mitochondrial protein coding genes that differ in their sequence variability and number of taxa, and an EST data set of mainly ribosomal protein coding genes, including missing data of single taxa. For nucleotide alignments, we tested the performance of ALISCOPE and GBLOCKS on highly variable 12S + 16S rRNA sequence alignments (Tab. 2.1).

## 2.2 Methods

### 2.2.1 Data sets

We used four different types of real data sets in combination with different alignment approaches, three mitochondrial (mt) and one nuclear (nu) data set (Fig. 1). Complete mt protein coding sequences of 11 genes were downloaded for eukaryotes from SwissProt and GenBank. Six genes (*COII*, *COIII*, *ND2*, *ND3*, *ND4L*, *ND6*) show high sequence variability compared to the less variable genes (*COI*, *Cytb*, *ND1*, *ND4*, *ND5*). The first mt data set (mtI) included protein sequences of all chosen mitochondrial genes of 17 taxa. The second mt data set (mtII) comprised the five less variable genes out of data set mtI but with 24 taxa, corresponding to Talavera & Castresana [67]. The third mitochondrial data set (12S + 16S) included nearly

complete 12S + 16S rRNA sequences for 63 arthropod taxa. The nuclear data set (EST) was compiled from 51 mainly ribosomal protein coding genes from Expressed Sequence Tags (ESTs) of 26 arthropod taxa. These were selected from published (dbEST, NCBI) and unpublished EST data (Meusemann, v. Reumont, Burmester, Roeding, unpubl.). The data comprised representatives of all major arthropod clades including water bears (Tardigrada) and velvet worms (Onychophora). A definitive tree of arthropods has not been established yet, therefore we restricted our comparison on tree resolution and bootstrap support values for selected clades. We remark that increased resolution and support might not reflect a real improvement of phylogenetic signal-to-noise ratio, but we consider this comparison as a good approximation in which the bootstrap values are used as approximation of tree-likeness in the data.

### 2.2.2 Alignments

All genes were aligned separately, each data set using MAFFT 6.240 [24], MUSCLE 3.52 [23], CLUSTALX 1.81 [21], and T-COFFEE 5.56 [20] with default parameters. Since the number of taxa of the rRNA data was too high for T-COFFEE, PCMA 2.0 [68] was used instead which aligns more similar sequences with the CLUSTAL algorithm and less similar sequences with the T-COFFEE algorithm. Each alternative alignment was profiled once with ALISCOPE and with all three possible gap predefinitions of GBLOCKS in which either no gaps (GBLOCKS(none)), all gaps (GBLOCKS(all)), or positions which have in less than 50% of sequences a gap (GBLOCKS(half)) are allowed. Thus, five different sets per alignment method were used in tree reconstructions: a) unmasked, b) three different GBLOCKS masked, and c) ALISCOPE masked. This was conducted for all four data sets (mtI, mtII, 12S + 16S, EST). Using ALISCOPE, alignments were screened separately with 2,000 randomly drawn pairwise comparisons and a window size  $w = 6$ . Within its scoring function gaps were treated like ambiguous characters on nucleotide level. On amino acid level we used the BLOSSUM62 substitution matrix. Positions identified by ALISCOPE or suggested by GBLOCKS as randomly similar were removed and single genes were concatenated for each data set and each approach. Percentage of remaining positions after masking was plotted for each alignment and masking approach (Fig. 2.1), in total for 1,104 single alignments (see electronic supplementary File ES1).

### 2.2.3 Split Networks

Split decomposition patterns were analyzed with SplitsTree 4 [69], version 4.10. We used the Neighbor-Net algorithm [70] and uncorrected p-distances to generate Neighbor-Net graphs from concatenated alignments of each data set before and after exclusion of randomly similar sections.

## 2.2.4 Tree reconstructions

Maximum likelihood (ML) trees were estimated with RAxML 7.0.0 [71] and the RAxML PTHREADS version [72]. We conducted rapid bootstrap analyses and search for the best ML tree with the GTRMIX model for rRNA data and the PROT MIX model with the BLOSUM62 substitution matrix for amino acid data with 100 bootstrap replicates each. Twenty topologies with bootstrap support values of all three GBLOCKS masked, ALISCORE masked, and unmasked alignments were compared for each single data set. Majority rule was applied for all GBLOCKS masked, ALISCORE masked, and unmasked topologies to investigate consistency of selected clades. Clades below 50% bootstrap support were considered as unresolved.

## 2.3 Results

### 2.3.1 ALISCORE algorithm for amino acid data

As for nucleotide sequences [34], ALISCORE uses a sliding window approach on pairs of amino acid sequences to generate a profile of random similarity between two sequences. In contrast to the algorithm with nucleotide data, ALISCORE employs the empirical BLOSUM62 matrix,  $Q$ , (or alternatives of it, PAM250, PAM500, MATCH) to score differences between amino acids,  $Q_{ij}$ . Pairs containing indels and any amino acid are defined by using the value of a comparison of stop codons and any amino acid defined within  $Q$ . The observed score within a window of pairwise comparisons is generated by summing scores of single site comparisons. Starting from a multiple sequence alignment of length  $L$ , sequence pairs  $(i, j)$  are selected for which the following procedure is executed: In a sliding window of size  $w$  at position  $k$ , a similarity score  $S(k)$  is calculated comparing positions  $(i(k), j(k)), \forall k \in (1, 2, \dots, L)$ , using the following simple objective function:

$$S(k) = \sum_{p=k}^{k+w-1} Q_{ij}(p)$$

Observed scores are compared to a frequency distribution of scores of randomly similar amino acid sequences with length given by the window size. The generation of randomly similar sequences follows the Proportional model [73], which is an adaptation of a simple Poisson model of change probability, adapted for observed amino acid frequencies, but still assuming that the relative frequencies of amino acids are constant across sites:

$$\text{Proportional} : P_{ij}(t) = \begin{cases} \pi_j + (1 - \pi_j)e^{-\mu t} & (i = j) \\ \pi_j + (1 - e^{-\mu t}) & (i \neq j) \end{cases}$$

with  $P_{ij}(t)$  as the probability of change from amino acid  $i$  to  $j$ ,  $\pi_j$  the frequency of amino acid  $j$ ,  $\mu$  the instantaneous rate of change, and  $t$  the branch length/time. Different to the algorithm used with nucleotide sequences in which scores are adapted to varying base composition along sequences and among sequences, the frequency

distribution of scores of randomly similar sequences is only produced once for amino acid data. The frequency distribution is generated by: 1) collecting frequencies of amino acids of the complete observed data set, 2) generating 100 bootstrap resamples of this amino acid frequency distribution and 100 delete-half bootstrap resamples of each of the 100 complete bootstrap resamples, and 3) by using these 10,000 delete-half bootstrap resamples to generate 1,000,000 scores of randomly similar amino acid sequences with length given by the window size. This complex resampling leads to an even distribution of scores of randomly similar sequences. The frequency distribution of randomly similar sequences is used to define a cutoff  $c(\alpha = 0.95)$  to assess randomness of the observed sequence similarity within the sliding window. Matching indels are defined as  $Q_{ij} = c/w$ . The principle of the complete scoring process is described in [34].

### 2.3.2 Testing performance on real data

#### 2.3.2.1 Extent of identified randomly similar blocks

Compared to GBLOCKS, using ALISCORE resulted in the exclusion of fewer positions in most data sets (Fig. 2.1). GBLOCKS identified fewer randomized positions only for the highly diverse 12S + 16S rRNA data with the GBLOCKS(all) option. For each data set, the percentage of identified randomly similar sections differed on average between 1% and 5% for each multiple sequence alignment when ALISCORE was applied, and between 1% and 9% when GBLOCKS was used. Most alignment sites were discarded by the default option GBLOCKS(none).

#### 2.3.2.2 ML trees and Neighbor-Net analyses

Resulting ML trees and Neighbor-Net graphs were examined under two different aspects: 1) We compared trees of all unmasked alignments with trees of differently masked alignments per data set to analyze the influence of each masking method on data structure and presence/absence of selected clades (Fig. 2.2).

2) We compared bootstrap values of corresponding trees (Tab. 2.2) and Neighbor-Net graphs (Fig. 2.3) of unmasked and differently masked alignments to see if alignment masking improves the signal-to-noise ratio in the predicted way.

In general, ALISCORE masked alignments resulted in consistent ML topologies among identical but differently aligned sequence data. The ALISCORE algorithm performed in most cases better or at least equal well than the best GBLOCKS settings (GBLOCKS(all), GBLOCKS(half)). Application of GBLOCKS(none) yielded less congruent trees.

**Amino acid data** While plants, fungi, metazoans, and included subtaxa were fully resolved in unmasked trees of dataset mtI, sister group relationships between major clades (Fungi, Metazoa, Amoebozoa) could not be resolved without alignment masking. If alignments were masked according to the ALISCORE profile, all ML trees showed a sister group relationship between fungi and metazoans. In the

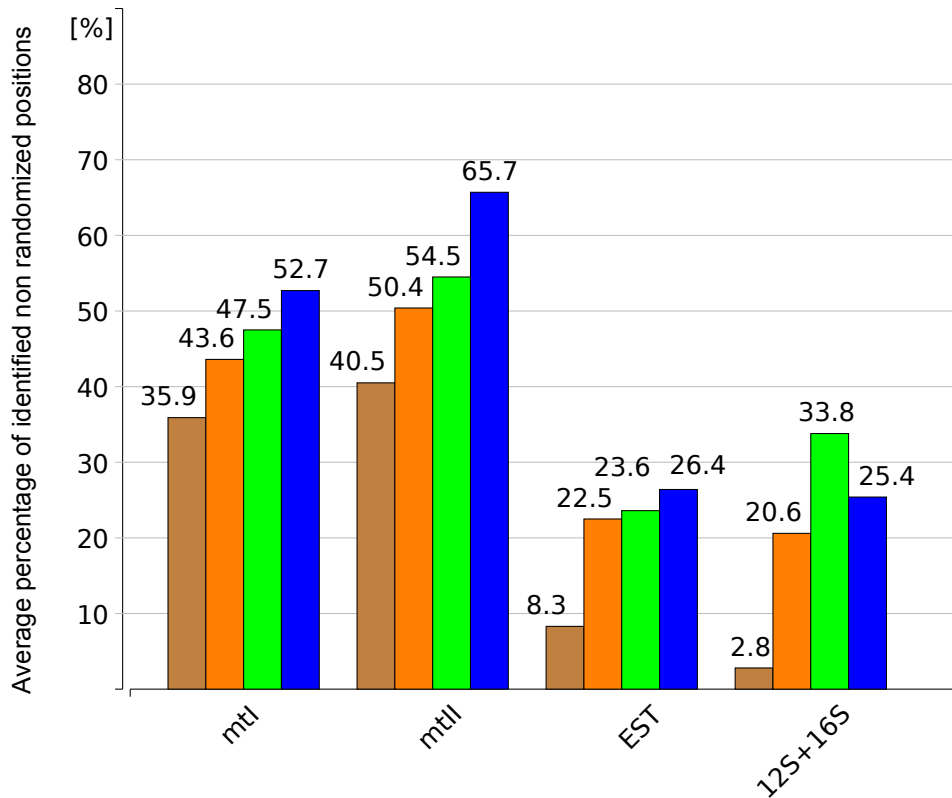


Figure 2.1: **Percentage of identified non randomized positions** X-axis: Used data sets. Y-axis: Average percentage of “non randomly similar” positions per data set after alignment masking. GBLOCKS(none): brown; GBLOCKS(half): orange; GBLOCKS(all): green; ALISCORE (blue). A list of all single values is given in the electronic supplementary File ES1.

Table 2.2: Averaged bootstrap support [%] of selected clades of each data set (mtI, mtII, EST, 12S + 16S). Inferred from majority rule ML trees for all GBLOCKS profiles (Gb(none), Gb(half), Gb(all)), ALISCORE (Al), and Unmasked (Unm). Values are averaged across different alignment methods per masking approach.

Data set	Selected clades	Gb(none)	Gb(half)	Gb(all)	Al	Unm
<b>mtI</b>						
	Plant	63.8	82.0	77.5	81.5	97.0
	Viridiplantae	99.8	99.5	99.8	100.0	100.0
	Streptophyta	100.0	100.0	100.0	100.0	100.0
	(Rhodophyta,Plant)	68.5	92.0	88.5	82.8	96.8
	Fungi	100.0	100.0	100.0	100.0	100.0
	(Ascomycota,Blastocladiomycota)	100.0	100.0	100.0	100.0	100.0
	Metazoa	100.0	99.5	99.3	100.0	85.8
	Bilateria	93.3	100.0	100.0	100.0	100.0
	Gastroneuralia	100.0	100.0	100.0	100.0	100.0
	Deuterostomia	73.0	99.8	100.0	99.8	100.0
	(Fungi,Metazoa)	100.0	62.5	75.0	76.0	0.0
	((Fungi,Metazoa),Amoebozoa)	41.3	17.3	46.3	24.0	0.0
<b>mtII</b>						
	Plant	0.0	0.0	0.0	0.0	0.0
	Viridiplantae	0.0	0.0	0.0	0.0	0.0
	Streptophyta	100.0	100.0	100.0	100.0	100.0
	Chlorophyta	0.0	0.0	0.0	0.0	0.0
	Rhodophyta	99.5	93.5	98.5	97.5	97.8
	(Rhodophyta,Plant)	0.0	0.0	0.0	0.0	0.0
	Amoebozoa	0.0	0.0	0.0	0.0	14.3
	Fungi	100.0	100.0	100.0	100.0	93.3
	(Ascomycota,Blastocladiomycota)	100.0	100.0	100.0	100.0	96.2
	Metazoa	100.0	100.0	100.0	100.0	100.0
	Bilateria	100.0	100.0	100.0	100.0	100.0
	Gastroneuralia	98.5	100.0	100.0	100.0	88.5
	Deuterostomia	69.3	94.5	98.8	95.5	73.3
	(Fungi,Metazoa)	87.3	70.5	64.8	79.3	67.5
	((Fungi,Metazoa),Amoebozoa)	0.0	0.0	0.0	0.0	0.0
<b>EST</b>						
	Chelicerata	0.0	50.8	57.5	24.8	0.0
	Pancrustacea	97.8	99.5	100.0	100.0	0.0
	(Cirripedia,Malacostraca)	56.0	58.5	79.5	85.0	0.0
	Hexapoda	0.0	46.5	66.8	52.8	0.0
	Collembola	98.8	99.5	99.8	100.0	0.0
	Nonoculata	18.8	74.5	72.8	84.3	0.0
	Ectognatha	88.0	55.8	76.8	75.3	0.0
<b>12S + 16S</b>						
	Campodeidae	17.5	89.3	98.8	97.0	99.8
	Diplura	0.0	87.8	96.8	92.5	94.3
	Archaeognatha	0.0	59.8	85.3	75.8	47.5
	Decapoda	0.0	38.5	38.0	81.8	73.3
	Dictyoptera	0.0	38.0	39.3	45.3	49.5
	Collembola	0.0	97.5	99.3	97.5	96.0
	Odonata	57.8	100.0	100.0	99.3	100.0
	Japygidae	29.8	99.5	99.0	100.0	100.0
	Hymenoptera	0.0	88.0	84.3	83.8	24.8



case of the T-COFFEE alignment, the Amoebozoa were placed as sister group to Fungi + Metazoa. The alignment masking of GBLOCKS(all) and GBLOCKS(half) led to comparatively resolved topologies. The GBLOCKS(none) option reduced signal in the data (Fig. 2.2). Bootstrap values as measurement of data structure increased after alignment masking in particular for deep nodes (clade (Fungi, Metazoa) and ((Fungi, Metazoa), Amoebozoa), see Tab. 2.2). After alignment masking, Neighbor-Net graphs showed less conflict (Fig. 2.3).

For the mtII data set we were not able to recover monophyletic plants and Amoebozoa as sister group to Fungi + Metazoa. The sister group relationship between Fungi and Metazoa was fully resolved in all ALISCORE, GBLOCKS(all), and GBLOCKS(none) masked data sets. GBLOCKS(none) and GBLOCKS(half) masked alignments supported in several instances implausible clades (Fig. 2.2). Bootstrap support values marginally increased after alignment masking (Tab. 2.2). Neighbor-Net graphs as well showed only marginal reduction of conflicts after alignment masking (Fig. 2.3).

Unmasked EST data did not yield well supported resolved trees. Most ALISCORE masked alignments led to clearly improved resolution of ‘traditionally’ recognized clades (e.g. Chelicerata, Hexapoda, Pancrustacea). If alignments were masked using GBLOCKS(all) or GBLOCKS(half), tree resolution increased likewise. Using the GBLOCKS(none) masking option did not improve resolution compared to other masked alignments (Fig. 2.2). Considering bootstrap values as measurement of tree-likeness, GBLOCKS(all), GBLOCKS(half), and ALISCORE improved tree-likeness of the data (Tab. 2.2). Except for the default GBLOCKS(none) setting, Neighbor-Net graphs showed a substantial decrease of conflict after alignment masking (Fig. 2.3).

**Nucleotide data** Again, ALISCORE and GBLOCKS(all) masking improved tree-likeness of the 12S + 16S nucleotide alignments at the taxonomically ordinal level. ALISCORE outperformed GBLOCKS(all) and GBLOCKS(half) in all instances. GBLOCKS(none) clearly performed worst (Fig. 2.2, Tab. 2.2).

## 2.4 Discussion

Parametric and non-parametric masking methods were successful in identifying ‘problematic’ alignment blocks. In general removal of these blocks prior to tree reconstruction improved resolution and bootstrap support. We interpret these results as an improvement in signal-to-noise ratio. For data set mtI and mtII we assumed clade validity congruently to Talavera & Castresana [67]. For the EST data set, traditionally accepted clades were only recovered for masked data sets in contrast to the unmasked approach, e.g. Pancrustacea [74–82], Malacostraca [83–85], Hexapoda [74, 75, 77, 80–82, 86–90], Ectognatha [77, 81, 86, 87, 89, 91] or Collembola [77, 81, 86–90], see (Fig. 2.4). A detailed review on these clades including morphological, neuro-anatomical and palaeontological evidence has been recently

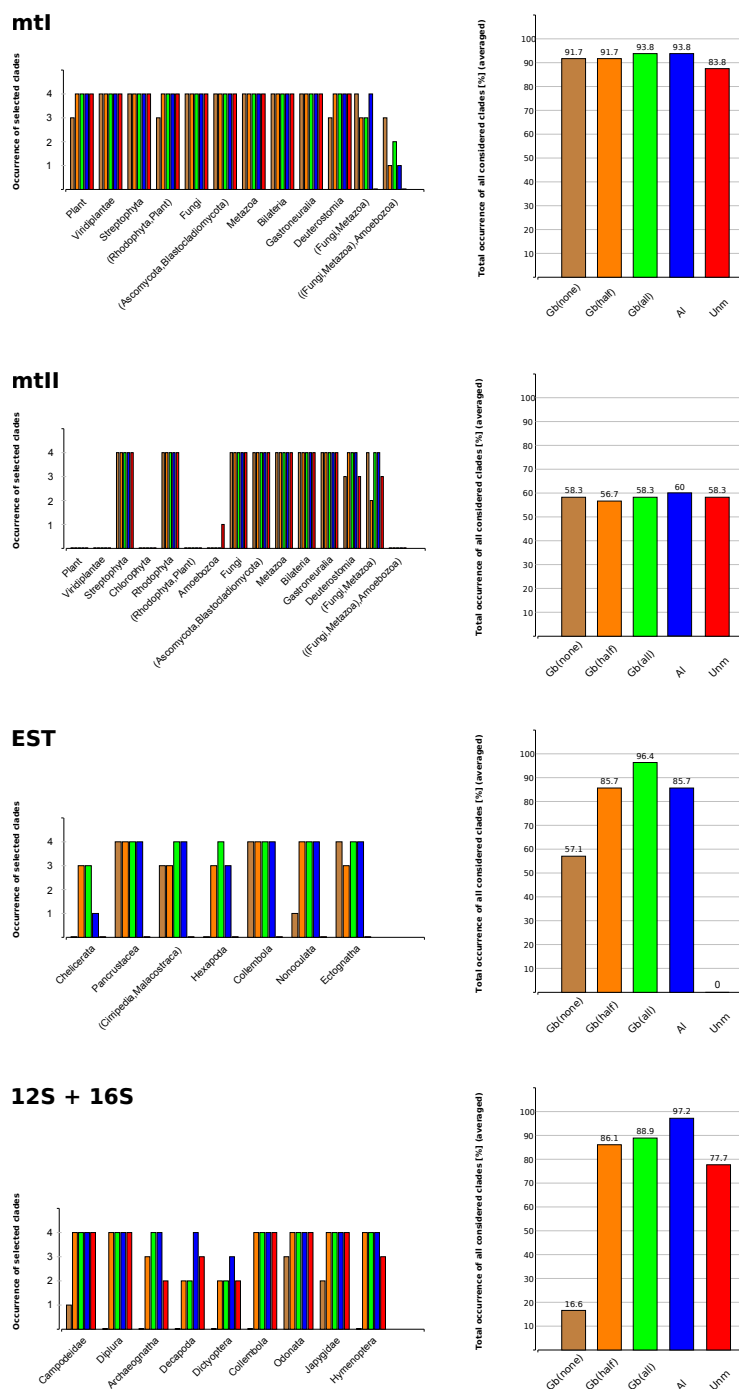


Figure 2.2: **Average percentage of resolved clades within single data sets.** On the left: Occurrence of selected clades (Tab. 2.2–2.3) of each data set (mtI, mtII, EST, 12S + 16S), inferred from majority rule ML trees. On the right: Total occurrence of all considered clades [%] for each data set, averaged across all four alignment methods. GBLOCKS(none): brown; GBLOCKS(half): orange; GBLOCKS(all): green; ALISCORE: blue; Unmasked: red.

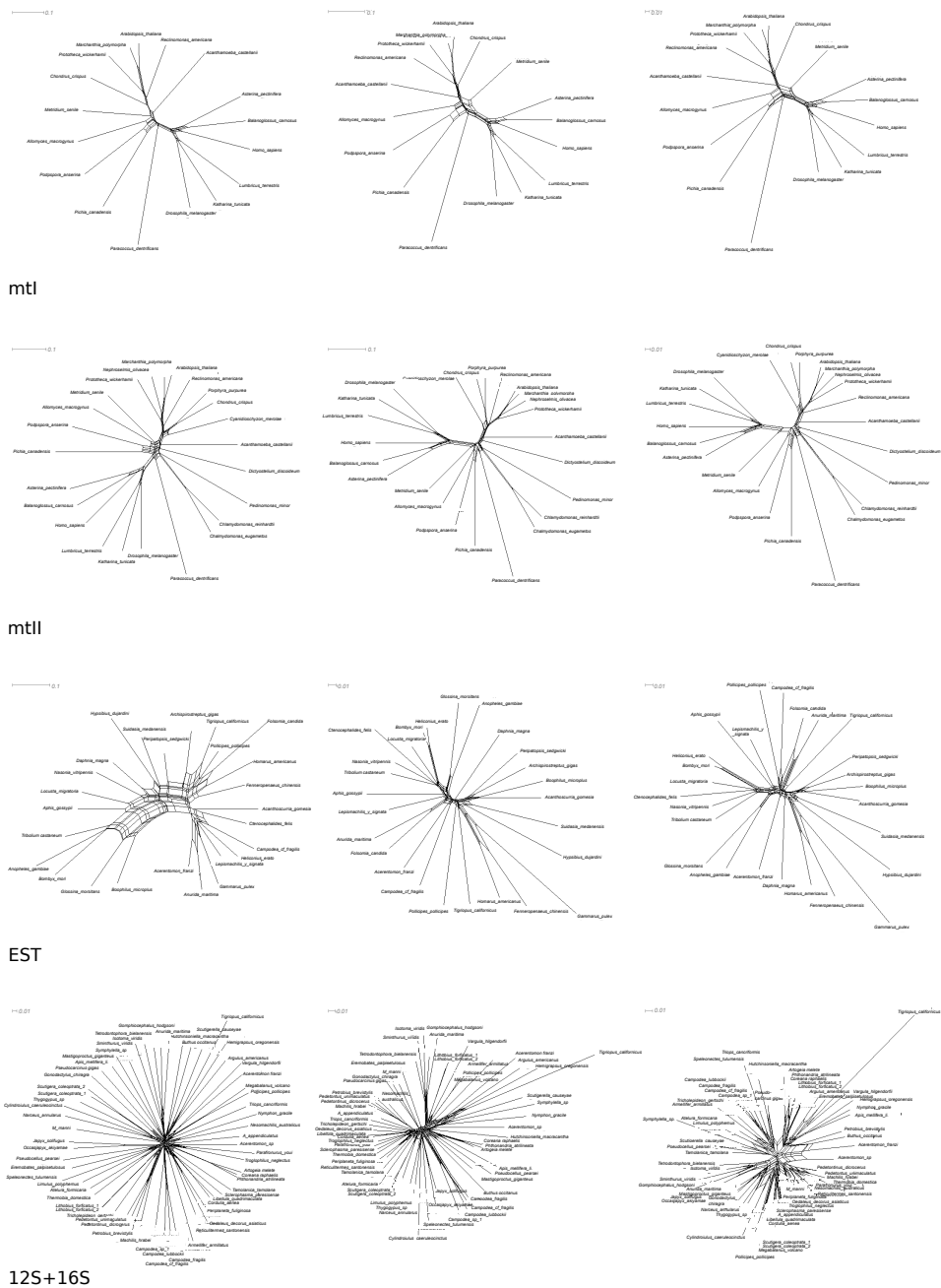


Figure 2.3: **Neighbor-Net graphs.** Neighbor-Net graphs generated with SplitsTree 4.10 based on concatenated supermatrices of unmasked (left), ALISCORE masked (middle), and GBLOCKS(none) masked (right) data. mtI, mtII and EST networks depend on a T-COFFEE alignment, the 12S + 16S rRNA network on a PCMA alignment. Neighbor-Nets were calculated with uncorrected p-distances. All inferred Neighbor-Net graphs are given in the electronic supplementary File ES1. Tree like structures in these graphs indicate distinct signal-like patterns in the corresponding alignment. Graphs generated from ALISCORE data sets are more tree-like. Lack of information leads to star-like graphs, conflicting signal produces cobwebs.

published in Edgecombe [92] and Grimaldi [93]. An improved data structure after alignment masking is also supported by more distinct split patterns (Fig. 2.3).

Alignment masking further reduced sensitivity of tree reconstructions to different alignment methods. The method implemented in GBLOCKS has the potential to overestimate the extent of divergent or ambiguously aligned positions, especially in partial gene sequences and gappy multiple sequence alignments like EST data or rRNA loop regions. Masking with the GBLOCKS(none) option tended to result in suboptimal node resolution and support values (Fig. 2.2 and Tab. 2.2–2.3). In the case of the 12S + 16S rRNA data, GBLOCKS(none) masking even reduced signal strength. This phenomenon is clearly evident in Figure 2.3, where the split decomposition pattern appears most fuzzy in the GBLOCKS(none) Neighbor-Net graph. We conclude that the incongruence between GBLOCKS(none)– and remaining masked trees may have resulted from conservative and stringent default parameters settings of GBLOCKS, in which all gap including positions were removed and only large conserved blocks were left. While the higher amount of conflicting signal in unmasked multiple sequence alignments clearly based on noisy data, it seems that the GBLOCKS(none) masking discarded too many informative positions.

The ALISCORE and GBLOCKS(all) approach performed quite similar and best on all data sets. This demonstrates that even a predefined set of rules suffices to extract randomness within sequence alignments. Talavera & Castresana [67] showed this already in their extensive analyses of GBLOCKSs performance.

The use of large data sets in phylogenomic analyses resulted in a tremendous increase of molecular data, but also in an increase of sampling error which could even bias seemingly robust phylogenetic inference [94]. Several such cases have been reported [95–97]. Therefore, it is important to establish a reliable alignment masking approach to cope with systematic errors in multiple sequence alignments. Our analyses showed that the sliding window approach will be a useful profiling tool to guide alignment masking.

ALISCORE optionally uses a BLOSUM62 or various PAM matrices to score differences between amino acid sequences, or a simple match/mismatch score for differences between nucleotide or amino acid sequences. It uses a simple modified Poisson model of character state change (called Proportional for amino acids [73], adapted for uneven base composition and sequence selection) in its resampling procedure to generate a null distribution of expected scores of randomly similar sequences. These scoring models and resampling processes are not very realistic, but however performed well in our analyses.

A recently published alternative approach, NOISY, uses a `qnet`-graph of sequence relationships to assess randomness of single positions [62]. The approach uses Monte Carlo resampling of single columns to compare fit of random data columns on a `qnet`-graph with the fit of observed data columns. The NOISY method appears as a fast and better alternative to the GBLOCKS approach, but a comparative analyses of its performance with the sliding window approach remains to be done.

We demonstrate for empirical data that alignment masking is a powerful tool to improve signal-to-noise ratio in multiple sequence alignments prior to phylogenetic

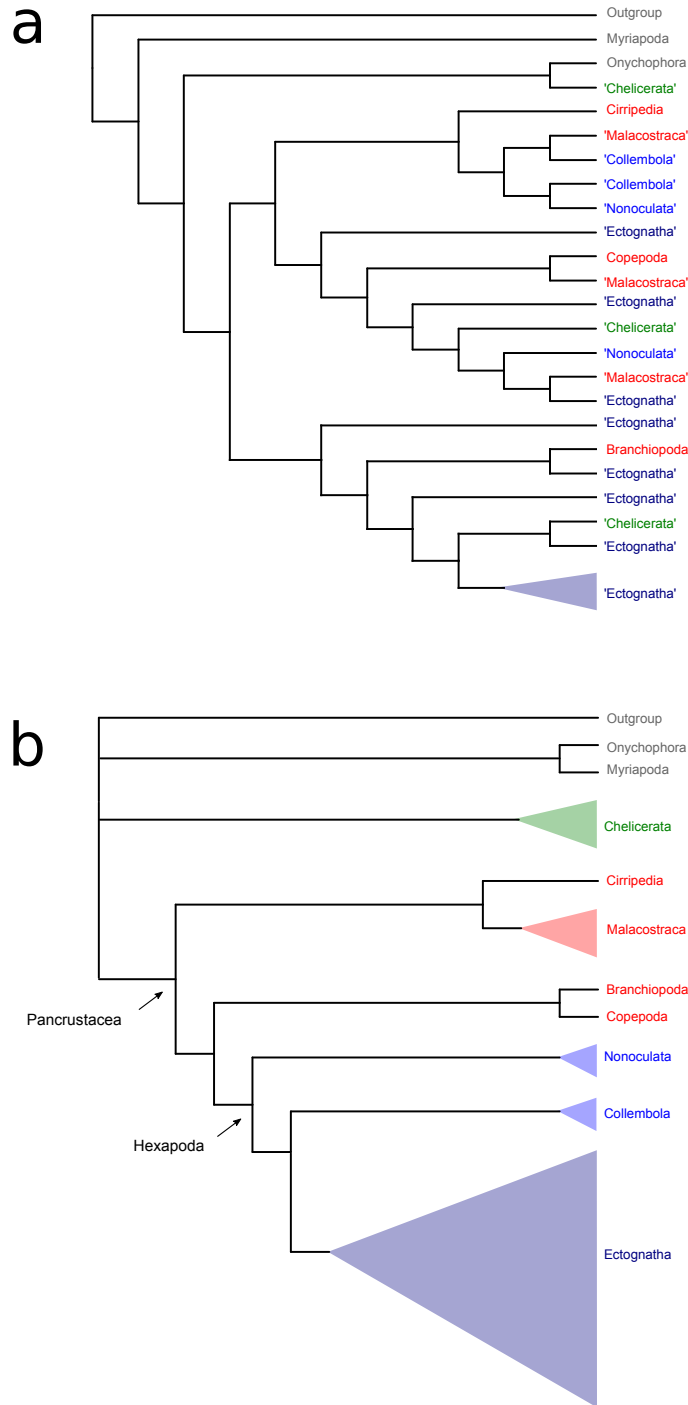


Figure 2.4: **Schematized cladograms inferred from the unmasked and masked EST data set.** Schematized cladograms (best ML trees, majority rule) inferred from the T-COFFEE aligned EST data set a) unmasked b) masked with ALISCORE considering selected clades. Quotation marks indicate non-monophyly of clades. Color code: Outgroup (tardigrades), onychophorans, myriapods: grey; chelicerates: green; crustaceans: red; hexapods: blue (proturans, diplurans, collembolans: royal-blue; ectognath hexapods: dark-blue).

Table 2.3: **Average percentage of resolved clades (selected)**. Inferred from majority rule ML trees, averaged across different alignment methods. Values are given for all GBLOCKS profiles (Gb(none), Gb(half), Gb(all)), ALISCORE (Al), and Unmasked (Unm).

<b>Data</b>	Gb(none)	Gb(half)	Gb(all)	Al	Unm
mtI	91.7	91.7	93.8	93.8	83.8
mtII	58.3	56.7	58.3	60	58.3
EST	57.1	85.7	96.4	85.7	0
12S + 16S	16.6	86.1	88.9	97.2	77.7

reconstruction. Masking multiple sequence alignments makes them additionally less sensitive towards different alignment algorithms. Our study also shows, that the scoring algorithm for amino acid data implemented in ALISCORE performs well.

The ALISCORE (parametric) approach is independent of a *priori* rating of sequence variation and seems to be more capable to handle automatically different substitution patterns and heterogeneous base composition.

It will be a matter of further analyses, whether an extension of the sliding window approach to more realistic likelihood models of change and Monte Carlo resampling will further improve the performance. However, it would be conceivable to implement a more explicit model based approach in GBLOCKS as well. The advantage of improved parameterizing GBLOCKS could be a significant gain in speed compared to the sliding window approach. The best approach should be the most efficient one in terms of computational time and increased reliability of trees, the latter one admittedly hard to assess.

## 2.5 Additional Files

- **Electronic supplementary file ES1 — Detailed analytical results of chapter 2**
  - Detailed results including lists of the percentage of remained positions after alignment masking per data set and alignment method. Given are all considered clades and corresponding bootstrap values (> 50%) per data set, alignment method and (un)masked approach as well as all Neighbor-Net graphs
  - **Format:** XLS
  - **Size:** 757 KB
  - **View:** Excel Viewer or Libre Office Calculator
- **Electronic supplementary file ES2 — Presentation of the ALISCORE algorithm and the results of chapter 2**

- The presentation was given 2009 within the status seminar of the “Deep Metazoan Phylogeny (DMP)” project and describes the ALISCORE algorithm, results of chapter 2, and gives a perspective of the AliGROOVE algorithm described in chapter 4
- **Format:** PDF
- **Size:** 1.6 MB
- **View:** PDF Viewer
- **Electronic supplementary file ES3 — Publication (Kück et al. (2010) [35])**
  - Corresponding publication to the study of chapter 2
  - **Format:** PDF
  - **Size:** 875.6 KB
  - **View:** PDF Viewer

# Improved phylogenetic analyses corroborate a plausible position of *Martialis heureka* in the ant tree of life

---

## Contents

---

<b>3.1</b>	<b>Introduction</b>	<b>27</b>
<b>3.2</b>	<b>Materials and Methods</b>	<b>28</b>
3.2.1	Data set	28
3.2.2	Alignment	28
3.2.3	Phylogenetic reconstructions	29
<b>3.3</b>	<b>Results</b>	<b>30</b>
3.3.1	Alignment masking, number of bootstrap replicates and likelihood scores	30
3.3.2	Phylogenetic relationships	32
<b>3.4</b>	<b>Discussion</b>	<b>36</b>
<b>3.5</b>	<b>Additional Files</b>	<b>38</b>

---



---

**Abstract:** Martialinae are pale, eyeless and probably hypogaecic predatory ants. Morphological character sets suggest a close relationship to the ant subfamily Leptanillinae. Recent analyses based on molecular sequence data suggest that Martialinae are the sister group to all extant ants. However, by comparing molecular studies and different reconstruction methods, the position of Martialinae remains ambiguous. While this sister group relationship was well supported by Bayesian partitioned analyses, Maximum Likelihood approaches could not unequivocally resolve the position of Martialinae. By re-analysing a previous published molecular data set, we show that the Maximum Likelihood approach is highly appropriate to resolve deep ant relationships, especially between Leptanillinae, Martialinae and the remaining ant subfamilies. Based on improved alignments, alignment masking, and tree reconstructions with a sufficient number of bootstrap replicates, our results strongly reject a placement of Martialinae at the first split within the ant tree of life. Instead, we suggest that Leptanillinae are a sister group to all other extant ant subfamilies, whereas Martialinae branch off as a second lineage. This assumption is backed by approximately unbiased (AU) tests, additional Bayesian analyses and split networks. Our results demonstrate clear effects of improved alignment approaches, alignment masking and data partitioning. We hope that our study illustrates the importance of thorough, comprehensible phylogenetic analyses using the example of ant relationships.

**Keywords:** Maximum Likelihood, Ant Tree of Life, Bayesian Analyses, Martialinae

---

### 3.1 Introduction

Recently, a spectacular and rare new subfamily of ants was described from the Brazilian Amazon with new implications for the ant tree of life. The monotypic subfamily, Martialinae was characterized by a single worker that shows remarkable morphological features [40]. It is a small, blind, pale, and most likely hypogaecic predator that lives either in the leaf-litter stratum or directly within the soil. Some morphological characters, such as the absence of eyes and frontal lobes, fully exposed antennal sockets, and a flexible promesonotal suture, indicate a closer relationship to the also small, eyeless, subterranean, and predatory ant subfamily, Leptanillinae [98]. Other characters, like a strongly reduced clypeus and long forceps-like mandibles, justify the establishment of a taxon Martialinae [40]. More important, this new subfamily was presented as a putative sister group to all other extant ants on the basis of the molecular analyses of three nuclear genes, the small and large nuclear subunits 18S and 28S rRNA and elongation factor EF1aF2 [40]. Previous molecular studies had proposed the subfamily Leptanillinae as a sister group of all other extant ants [37–39]. The proposed sister group relationship of leptanillines suggested in these studies, as well as the one presented for Martialinae by Rabeling et al. [40], is of high significance for a better understanding of ant relationships and ground plan characters. These results strongly support the scenario of a small, eyeless, and hypogaecic predator as an ancestor of modern ants [37,38,40], but contradict previous morphological studies, which assumed that ancestral ants were larger, more wasp-like, epigaecic foragers with well-developed eyes [99–102]. Therefore, the phylogenetic position of Martialinae and Leptanillinae within the ant tree of life still awaits a clear resolution.

Rabeling et al. [40] presented a Bayesian tree with resolved single inter- and intra subfamily relationships and proposed Martialinae as the earliest branch (posterior probability 0.91) within the ant tree of life. Recent studies have shown that Bayesian analyses tend to overestimate the potential signal within data and provide high support values, even if the data is completely uninformative [51,52]. Furthermore, Bayesian approaches show a much higher type I error rate (the possibility that erroneous conclusions will be drawn more often), especially in the case of model misspecification [52]. Bayesian posterior probability values are substantially higher than corresponding bootstrap values [51,52,103,104]. Suzuki, Glazko & Nei [51] showed in simulation studies that Bayesian support values “can be excessively liberal when concatenated gene sequences are used”. Bootstrap values are in general more conservative and more reliable in assessing the robustness of phylogenetic trees which should be preferable in phylogenetic analyses [51,52,104]. Therefore, we suggest that topologies inferred with Maximum Likelihood (ML) analyses in combination with a sufficient number of bootstrap replicates provide a more realistic picture of the underlying signal.

We re-analysed the data of Rabeling et al. [40] using partitioned and unpartitioned ML approaches with a sufficient number of bootstrap replicates. Despite the mentioned criticisms on Bayesian analyses, we additionally conducted compa-

rable Bayesian analyses to see whether any of our Bayesian topologies support the relationships found by Rabeling et al. [40], especially with respect to deep splits. For alignment masking we applied the software ALISCORE. Recent studies have shown that alignment masking of positions that can not be aligned unambiguously is strongly recommended to improve the signal-to-noise ratio in multiple sequence alignments prior to tree reconstruction. Several automated software tools have been developed [34–36, 62, 64] that offer a more comprehensible alignment masking than a manual exclusion of sites. ALISCORE is a parametric masking approach that identifies randomised alignment sections by using a Monte Carlo resampling within a sliding window [34, 35]. The approach assumes that the score of inaccurate and ambiguous alignment sections will not be distinguishable from randomly similar aligned sequences. Therefore, ALISCORE compares the score of originally aligned sequences with scores of randomly drawn sequences of similar character composition. ALISCORE has been successfully tested both in simulations [34] and on real data sets [35], and has been used in recent molecular phylogenetic studies [105–109].

## 3.2 Materials and Methods

### 3.2.1 Data set

We used molecular data previously published by Rabeling et al. [40]. In accordance to [40], we used the data matrix of Brady et al. [37] kindly provided by S. Brady. We added respective sequences of *Martialis heureka* [40] from GenBank (<http://www.ncbi.nlm.nih.gov/>). The data set comprised three genes of 152 taxa subdivided into 21 ant subfamilies and 11 outgroup taxa. Sequence data included elongation factor 1-alpha F2 (EF1aF2, nuclear protein coding gene), 18S rRNA and 28S rRNA (nuclear ribosomal genes).

### 3.2.2 Alignment

Single genes were aligned separately using the local L-ins-i algorithm of MAFFT version 6.717 [110]. The L-ins-i algorithm is an iterative progressive algorithm which outperformed other methods in benchmark tests [25, 26]. Each of the three sequence alignments (18S, 28S, and EF1aF2) was screened for randomised sections with ALISCORE [34] using all possible pairwise comparisons and a window size  $w = 6$ . Within ALISCORE, gaps were treated as ambiguous characters. Randomised sections (28S rRNA: 725 base positions (bp); 18S rRNA: 14 bp) were excluded with ALICUT [111]. In the EF1aF2 alignment, no randomised positions were detected. Single genes were concatenated using FASconCAT version 1.0 [112]. The concatenated supermatrix of the masked approach included 4,315 characters while the unmasked supermatrix comprised 5,054 characters. All alignments (phylip format) and the respective character partitions are provided as electronic supplementary files ES4 – ES7.

### 3.2.3 Phylogenetic reconstructions

#### 3.2.3.1 Split networks

We computed NeighbourNetworks [69,70,113] with SplitsTree 4.10 [69] to visualise the data structure of the unmasked and masked alignments. NeighbourNetworks were calculated applying uncorrected p-distances for the unmasked alignment and the masked alignment used for the masked-partitioned analyses. NeighbourNetwork graphs give an indication of noise, signal-like patterns and conflicts within a multiple sequence alignments.

#### 3.2.3.2 Maximum Likelihood Analyses

We estimated a Maximum Likelihood (ML) topology for the unmasked supermatrix and the masked supermatrix in non-partitioned analyses with RAxML [71] using RAxMLHPC-PTHREADS [72], version 7.2.6. A third topology was reconstructed from the masked supermatrix with four partitions according to the setup described for the Bayesian analyses in Rabeling et al. [40] with the RAxMLHPC-HYBRID [114], version 7.2.6. The first partition included the 18S, the second partition the 28S. The third partition comprised the 1st and 2nd codon position of EF1aF2, the fourth partition included the 3rd codon position of EF1aF2. We identified the correct reading frame and excluded the first position of the EF1aF2-alignment. Therefore, the EF1aF2-alignment was 1 bp shorter (516 bp) than that described in Rabeling et al. [40].

We conducted rapid bootstrap analyses and a thorough search for the best ML tree using GTR +  $\Gamma$  with 5,000 bootstrap replicates. We evaluated the number of necessary bootstrap replicates *a posteriori* for each data set according to the bootstrap criteria based on the Weighted Robinson-Foulds (WRF) distance criterion [115] using RAxML 7.2.6 for the extended majority-rule (MRE) consensus tree criterion. We chose a cutoff value of 0.01 to ensure a sufficient number of bootstrap replicates. In final trees, clades with a bootstrap support (bs) below 50% were considered unresolved. All analyses were performed on HPC LINUX clusters of the ZFMK, Bonn, Germany. Trees were edited with the software TreeGraph 2 [116].

To test alternative placements of Martialinae and Leptanillinae as suggested by Rabeling et al. [40], we exchanged the position of Martialinae and Leptanillinae in our best trees (unmasked, masked-unpartitioned and masked-partitioned). We compared alternative tree topologies by performing an AU test [117] for each data set. Therefore, we optimised branch lengths for alternative topologies. Subsequently, we calculated per site log Likelihood scores using RAxML 7.2.6. AU tests were performed with CONSEL [118], version v0.1i.

#### 3.2.3.3 Bayesian Analyses

Bayesian phylogenies were calculated using MrBayes [119,120] for three data sets also used in our ML analyses. Topologies were inferred from (i) the unmasked

superalignment (ii) the masked superalignment, non-partitioned and (iii) the masked superalignment with four partitions according to [40] and our ML analyses. Similar to Rabeling et al., we used MrBayes v3.2 (an unreleased version of MrBayes; the source code was downloaded from the current version system in January, 2011). Convergence of parameters of the Bayesian analyses was assessed with the software Tracer v1.5 [121].

We chose the sequence evolution model GTR +  $\Gamma$  for all three data sets (i) – (iii) for accuracy of comparison with our ML analyses. Parameters of the model (i.e., base frequencies, transition/transversion ratio, and rate variation shape parameter) were unlinked across partitions. According to Rabeling et al., Metropolis coupling was used with eight chains per analysis and a temperature increment of 0.05 [40]. For analysis (i) and (ii) we ran 30 million generations with a sample frequency of 200. For analysis (iii) we ran 28,130,500 generations with a sample frequency of 100. After checking all analyses for parameter convergence in Tracer v1.5, we discarded a burn-in of 10% for each analysis. After discarding the burn-in, majority rule consensus trees with posterior probabilities were calculated from all sampled trees within MrBayes. All analyses were performed on HPC LINUX clusters of the ZFMK, Bonn, Germany. Trees were edited with the software TreeGraph 2 [116].

### 3.3 Results

#### 3.3.1 Alignment masking, number of bootstrap replicates and likelihood scores

Alignment masking remarkably improved data structure, which is visualised by comparing split networks derived from the unmasked and masked alignments. The split (NeighborNet) network [69, 70, 113] from the masked alignment obviously showed less conflict than the split network from the unmasked alignment, especially within subfamilies of formicoids. Nevertheless, conflicting signal is obvious, e.g. within poneroids or dorylomorphs (Fig. 3.1).

We determined the number of sufficient bootstrap replicates for our ML analyses using the ‘bootstopping criterion’ according to Pattengale et al. [115] (see method section). Our unmasked data set converged after 2,400 bootstrap replicates, our masked-unpartitioned data set after 3,400 bootstrap replicates, and the masked-partitioned data set after 4,100 bootstrap replicates applying the Weighted Robinson-Foulds (WRF) distance criterion [115] with an extended majority-rule (MRE) consensus tree criterion and a cutoff value of 0.01. Thus, the number of 5,000 bootstrap replicates chosen for our ML analyses had been sufficient for all of our data sets.

Our partitioned ML analysis of the masked data set clearly outperformed the masked-unpartitioned data set in terms of likelihood scores (masked-partitioned:  $ln = -49230.716$ ; masked-unpartitioned:  $ln = -52002.229$ ).

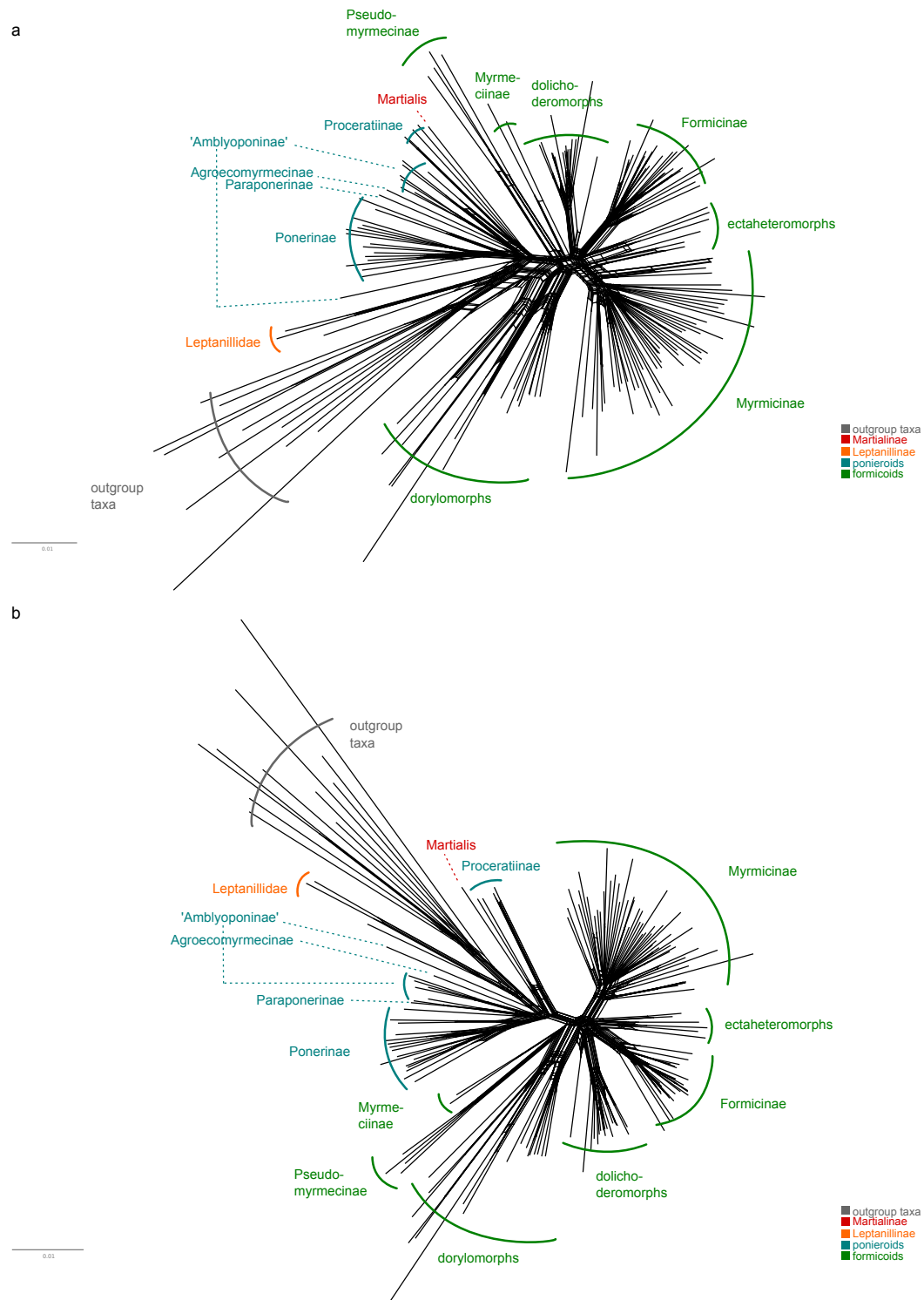


Figure 3.1: NeighborNet graphs With uncorrected  $p$  distances inferred with **Splitstree** version 4.10. a: Split network based on the unmasked alignment. b: Split network based on the masked alignment which was used for the masked-partitioned analyses.

### 3.3.2 Phylogenetic relationships

#### 3.3.2.1 Placement of Leptanillinae and Martialinae

All ML and Bayesian topologies suggested a clade including Leptanillinae + all remaining ant subfamilies with maximum support (Fig. 3.2–3.4, Tab. 3.1, and App. A: Fig. A.1–A.6). Martialinae always split off as a second branch and form a clade with poneroids and monophyletic formicoids. Applying an approximately unbiased test (AU test) [117] for all ML topologies, the Null hypothesis ( $H_0$ ) assumes that either Leptanillinae as a sister group of remaining Formicidae and Martialinae as second branch in the ant tree of life or vice versa, are not significantly different. While  $H_0$  was not significantly rejected for our unmasked data set ( $p = 0.120$ ), both ML topologies of our masked data sets significantly outperformed  $H_0$ . Both AU tests of the masked and the masked-partitioned data set significantly rejected  $H_0$  (masked:  $p < 0.0001$ ; masked-partitioned:  $p = 0.046$ ). Leptanillinae as the first split within the ant tree of life was also supported by our split network analyses. Both split networks (masked and unmasked) showed less conflict for Leptanillinae as the first split than for Martialinae (Fig. 3.1).

#### 3.3.2.2 Relationships of poneroids and formicoids

None of our topologies recovered a clade poneroids, except the Bayesian topology derived from the unmasked data set (0.86 bpp, see App. A: Fig. A.1). Further, all ML and Bayesian topologies failed to resolve the relationships between Agroecomyrmecinae, Amblyoponinae, Paraponerinae, and Proceratiinae. Conflicting signal among these subfamilies is seen in both split networks, but the masked network shows less conflict (Fig. 3.1b). In contrast to our unmasked data, all masked approaches resolved a (Ponerinae, formicoids) clade with weak bootstrap and high Bayesian support values (masked-unpartitioned: 57% bs, 0.97 bpp; masked-partitioned: 68% bs, 1 bpp; Fig. 3.3,3.4, Tab. 3.1, and App. A: Fig. A.2–A.3). A formicoid clade was maximally supported in all topologies (100% bs, 1 bpp).

Within formicoids, a dorylomorph clade was recovered in all our trees (100% bs, 1 bpp; Fig. 3.2–3.4, Tab. 3.1 and App. A: Fig. A.1–A.6). Four of six topologies suggested a clade dorylomorphs + formicoids. However, in the ML masked-unpartitioned topology, the placement of dorylomorphs remained unresolved. In the unmasked Bayesian topology, a clade dorylomorphs + Pseudomyrmecinae was present, but with weak support (see App. A: Fig. A.1). Concerning the relationships between dolichoderomorphs, Myrmeciinae, and Pseudomyrmecinae, we did not obtain an unequivocal resolution from any topology. The relationships between Formicinae, Myrmicinae and ectaheteromorphs were not resolved by our ML topology of the unmasked data set, whereas the trees of both masked approaches showed weak node support for a clade Myrmicinae + ectaheteromorphs (unpartitioned: 73% bs; partitioned: 67% bs). This clade was also resolved in all Bayesian topologies with moderate support (see App. A: Fig. A.1–A.3).

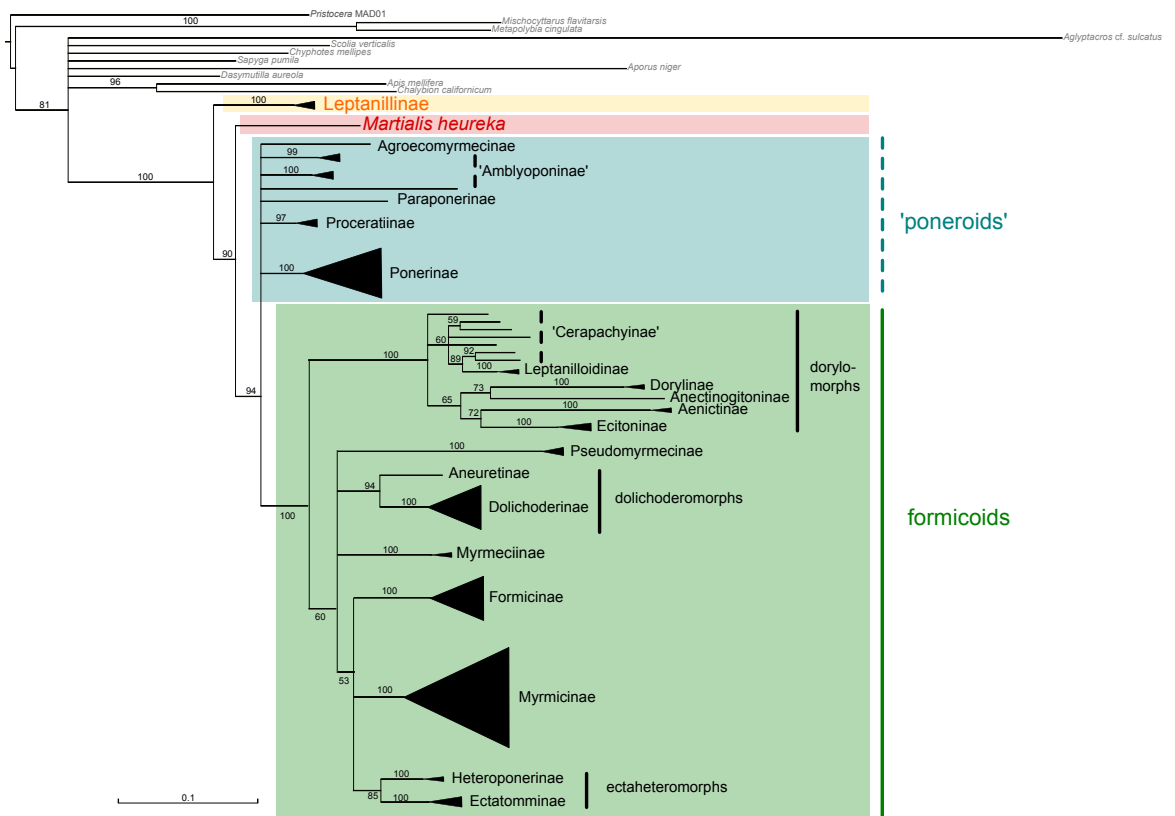


Figure 3.2: ML topology inferred from the unmasked, unpartitioned data set. Schematised ML topology with branch lengths inferred from the unmasked supermatrix (best ML tree, majority rule, 5,000 bootstrap replicates). Quotation marks indicate non-monophyly.



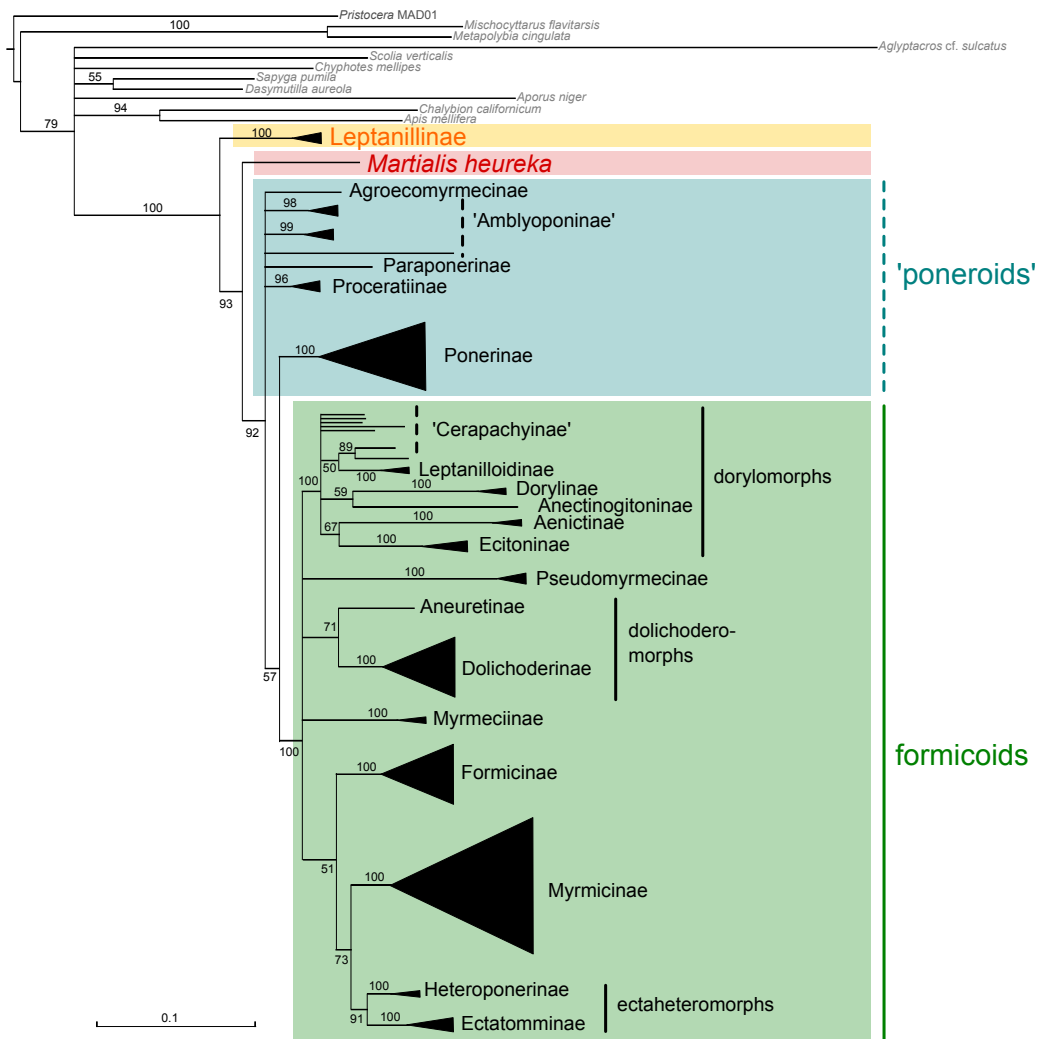


Figure 3.3: **ML topology inferred from the masked-unpartitioned data set.** Schematised ML topologies with branch lengths inferred from the masked supermatrix. Best ML tree of the masked-unpartitioned analysis (739 positions excluded from the unmasked alignment), majority rule, 5,000 bootstrap replicates. Quotation marks indicate non-monophyly.

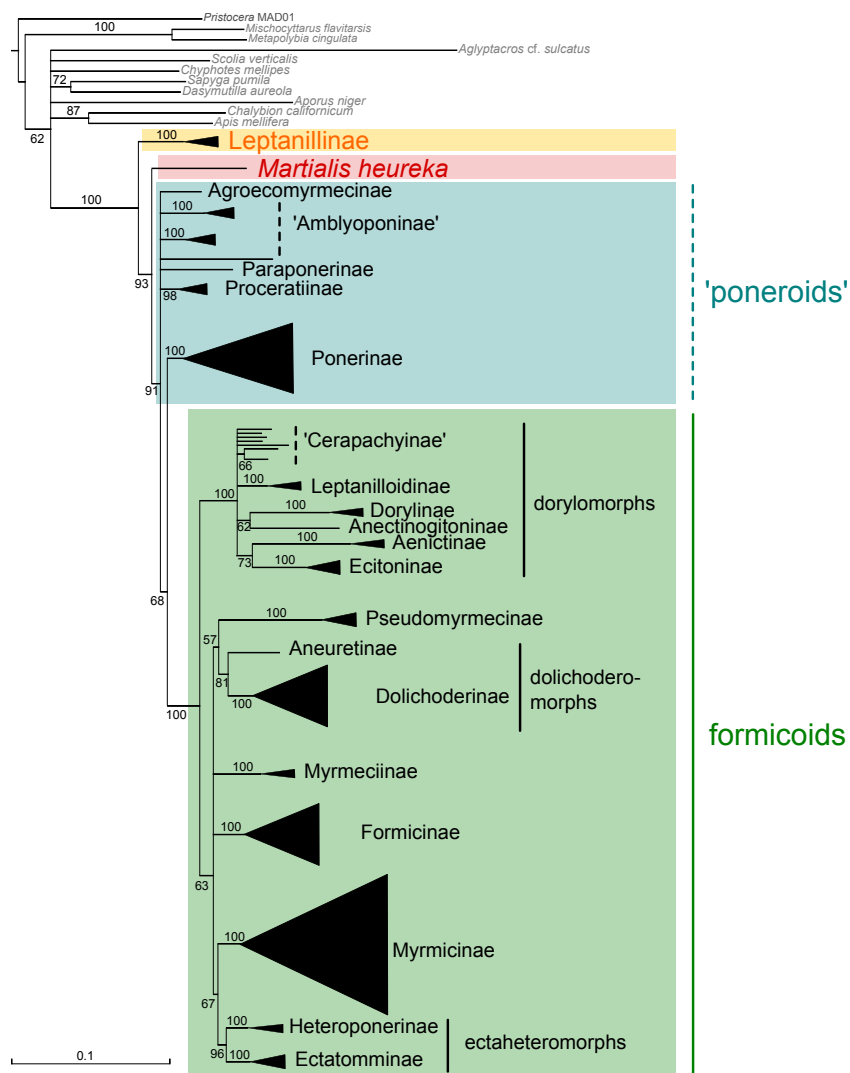


Figure 3.4: **ML topology inferred from the masked-partitioned data set.** Schematised ML topologies with branch lengths inferred from the masked supermatrix. Best ML tree, of the masked-partitioned analysis (739 positions excluded from the unmasked alignment + one bp to correct the reading frame), majority rule, 5,000 bootstrap replicates. Quotation marks indicate non-monophyly.

Table 3.1: Selected clades with bayesian posterior probability [bpp] and bootstrap support [bs] values recovered in our Bayesian (Bayes) and Maximum Likelihood (ML) topologies. Clade 1 (Leptanillinae,(Martialinae, remaining ants)) and (Martialinae(poneroid/formicoid clade)) are resolved in all Bayesian and ML topologies. Poneroids are not monophyletic with the exception of the unmasked, Bayesian topology (weakly supported). Amblyoponinae are only monophyletic within the Bayesian masked-partitioned topology. A clade (Ponerinae, formicoids) with a subsequent paraphyly of poneroids, is suggested by all masked topologies with high Bayesian posterior probability (bpp) but low bootstrap (bs) support. Dorylomorphs are monophyletic with exception of the masked-unpartitioned ML topology.

	Bayes posterior probabilities [bpp]			ML bootstrap support [bs]		
	unmasked	masked	masked-part.	unmasked	masked	masked-part.
Clade 1	1	1	1	100	100	100
Clade 2	1	1	1	90	93	93
poneroids	0.86	–	–	–	–	–
Amblyoponinae	–	–	0.77	–	–	–
(Ponerinae,formicoids)	–	0.97	1	–	57	68
formicoids	1	1	1	100	100	100
dorylomorphs	1	1	1	100	100	100

### 3.4 Discussion

A clade Leptanillinae + all remaining ant subfamilies is highly supported in all our ML and Bayesian analyses. This result is significant with AU tests for the masked-unpartitioned and masked-partitioned approach. Our split network analyses similarly corroborate this scenario. This is also congruent to earlier molecular studies [37, 38], but contradicts the results of Rabeling et al. [40]. Based on our re-analyses of the respective data set [40] and other molecular studies [37–39, 41, 122], we suggest that, at present, it seems unlikely that Martialinae are the sister group to all other recent ant subfamilies.

The placement of Martialinae suggested by Rabeling et al. [40] could be due to inferior sequence alignments or confounding effects of randomized alignment sections. The MAFFT-L-*ins-i* algorithm applied in our study was shown to be one of the most accurate available alignment algorithms, and can be considered to be the best choice for sequence alignments [25, 26]. Still, 739 alignment positions were identified by ALISCORE as potentially randomised and therefore excluded. ALISCORE and subsequent alignment masking increased the signal-to-noise ratio within the data, but influenced our tree topologies only marginally. However, a positive effect of the masking approach is clearly shown by a strong decrease of contradictory signal within the masked alignment, especially for deeper splits (Fig. 3.1). Partitioning of the masked data set leads to an increased likelihood score, and higher node resolution within formicoids. Martialinae are again resolved as the second branch (cf. Fig. 3.2–3.4, Tab. 3.1, and App. A: Fig. A.1–A.6) avoiding possible artifacts due to noise.

Discrepancies between our results and the results of Rabeling et al. [40] could

further be explained by an insufficient number of bootstrap replicates (ML approach) and an insufficient number of Bayesian generations. They conducted 500 bootstrap replicates for the ML approach [40] *versus* 5,000 bootstrap replicates in our study. Pattengale et al. [115] showed in a recent study on 'bootstopping' that the number of bootstrap replicates for accurate confidence values is strongly dependent on the data set. In testing the performance and accuracy of bootstrap criteria on real DNA alignments, they showed that a range of 100 – 500 bootstrap replicates is usually sufficient. Still, in some cases a much higher number of up to 1,200 replicates was necessary to deliver support values that are equally robust as those in the reference tree with 10,000 replicates. Most differences between reference and 'bootstopped' topologies occurred on poorly supported branches (< 75% bs). Since the bootstrap support in the ML tree of Rabeling et al. [40] for a clade Martialinae + remaining ants is only 76.2%, 500 replicates might have been insufficient. In contrast, our support values derived from 5,000 bootstrap replicates are evaluated and confirmed by *a posteriori* 'bootstop tests' (see results). As mentioned above, single data sets of earlier studies [37, 38] propose Leptanillinae as a sister lineage to all other ants. However, it should be considered that the subfamily Martialinae was just discovered in 2008. Therefore, Moreau [41] combined data sets of Brady et al. [37], Moreau et al. [38], and Rabeling et al. [40] to a supermatrix in which the relationship of Leptanillinae and Martialinae was unresolved.

Our analyses showed that an exclusion of randomised sections improved the resolution between Ponerinae and the formicoids (Fig. 3.3, 3.4, 3.1, and App. A: Fig. A.2, A.3, A.5, A.6). Alignment masking led to a placement of Ponerinae next to formicoids (Tab. 3.1). Discrepancies between low bs and high bpp support values seem to confirm typical observations considering Bayesian analyses [51, 52, 103, 104]. The relationships between the Amblyoponinae, Agroecomyrmecinae, Paraponerinae, and Proceratiinae remain unresolved in most of our topologies. Only the Bayesian topology of the masked-partitioned data set show monophyletic Amblyoponinae with weak support (Tab. 3.1). Thereby, Amblyoponinae branch off as a third split (0.84 bpp) within the ant tree of life. The monophyly of Amblyoponinae has been favoured by earlier studies [37–39, 41]. Therefore, we conclude that more genes are necessary to robustly resolve an amblyoponine clade as well as relationships between Amblyoponinae, Agroecomyrmecinae, Paraponerinae, and Proceratiinae. All our topologies highly support a dorylomorph clade. Our unmasked and masked-partitioned topology and both Bayesian topologies derived from our masked approaches corroborate a placement of the dorylomorphs next to the remaining formicoids. This hypothesis stands in concordance with other studies [37, 38, 40]. Finally, the non-monophyly of cerapachyines within the dorylomorphs is consistent with these studies.

Compared with Brady et al. [37], the inclusion of Martialinae reduce the branch lengths for leptanillines and formicoids, although the branch separating ants from the aculeate outgroup Hymenoptera still remains relatively long. However, with current methods and the available data, it is not possible to assess putative long branch artifacts like discussed in Brady et al. [37]. It is possible that new molecular sequence data might 'improve' the current ant tree of life. It is possible that a data

set with most signal coming from rRNA genes might not be sufficient to support a robust ant tree (cf. Fig. 3.1). For a deeper insight into subfamily relationships, multi-gene analyses of genomic/EST data and a more exhaustive taxon sampling combined with improved phylogenetic approaches seem indispensable.

### 3.5 Additional Files

- **Electronic supplementary file ES4 — Unmasked alignment file**
  - The unmasked supermatrix alignment in phylip format (18S, 28S, and EF1aF2), generated with the local L-ins-i algorithm of MAFFT version 6.717 [110]
  - **Format:** PHY
  - **Size:** 808.1 KB
  - **View:** Bioedit, Seaview or Texteditor
- **Electronic supplementary file ES5 — Masked alignment file for the masked-unpartitioned analyses**
  - The masked supermatrix alignment in phylip format (18S, 28S, and EF1aF2), generated with the local L-ins-i algorithm of MAFFT version 6.717 [110] and screened for randomised sections with ALISCORE [34]
  - **Format:** PHY
  - **Size:** 691.6 KB
  - **View:** Bioedit, Seaview or Texteditor
- **Electronic supplementary file ES6 — Masked alignment file for the masked-partitioned analyses**
  - The masked supermatrix alignment in phylip format (18S, 28S, and EF1aF2), generated with the local L-ins-i algorithm of MAFFT version 6.717 [110] and screened for randomised sections with ALISCORE [34]
  - **Format:** PHY
  - **Size:** 691.5 KB
  - **View:** Bioedit, Seaview or Texteditor
- **Electronic supplementary file ES7 — Character partition file**
  - Character partition file (plain text format) for the masked alignment used for the masked-partitioned analyses
  - **Format:** TXT
  - **Size:** 691.6 KB
  - **View:** Texteditor

- **Electronic supplementary file ES8 — Publication (Kück et al. (2011) [123])**
  - Corresponding publication to the study of chapter 3
  - **Format:** PDF
  - **Size:** 386 KB
  - **View:** PDF Viewer



# AliGROOVE: a new tool to visualize the extent of sequence similarity and alignment ambiguity in multiple alignments

---

## Contents

---

<b>4.1</b>	<b>Introduction</b>	<b>43</b>
4.1.1	AliGROOVE algorithm	45
<b>4.2</b>	<b>Material and Methods</b>	<b>47</b>
4.2.1	Simulated data	47
4.2.2	Empirical data	47
<b>4.3</b>	<b>Results</b>	<b>48</b>
4.3.1	Testing performance on simulated data	48
4.3.2	Testing performance on empirical data	51
<b>4.4</b>	<b>Discussion</b>	<b>51</b>

---



---

**Abstract:** The detection of ambiguously aligned sequence sections through alignment masking methods has become a widely accepted tool to reduce signal noise and to increase tree-likeness of given data sets. A main disadvantage of all masking methods is their insensitivity in detecting heterogenous sequence divergence within sequence alignments. With AliGROOVE, we propose a tool that can visualize heterogeneous sequence divergence or alignment ambiguity related to single taxa or subsets of taxa within alignments. The method prepares profiles of sequence similarity for all pairwise comparisons by using an adaptive implementation of the sliding window approach which was first introduced in the ALISCORE masking method. The sliding window approach offers the possibility to identify taxa which are robustly supported in topologies, but show predominantly randomized sequence similarity in comparison to other taxa. The removal of these taxa can lead to an increase of alignment quality and tree-likeness of data which in turn improve the reliability of tree reconstructions. AliGROOVE was tested on simulated and empirical data. The results show that the sliding window approach has some predictive power, therefore we consider this characteristic as a major advantage over all character based masking approaches in phylogenetics.

**Keywords:** Alignment Masking, ALISCORE, Data Quality, Tree-Likeness, Sequence Similarity, Alignment Ambiguity

---

## 4.1 Introduction

Alignment masking as a measure of reducing noise in sequence alignments is regularly applied in phylogenetics. The idea behind masking blocks of sequence alignments is that the influence of missing and/or ambiguously aligned blocks of sequence alignments in subsequent tree reconstructions are reduced [34–36, 62, 67] by increasing the tree-likeness of the data. Simulations and analysis of alignment masking of empirical data corroborate the correctness of this idea. Basically, complete blocks of alignments are masked applying either arbitrarily chosen thresholds of sequence variability within alignment columns (e.g. Gblocks [36, 67] and REAP [64]) or applying a sliding window approach to identify blocks of predominately high alignment ambiguity (Aliscore [34, 35]). All methods inherently exclude complete alignment blocks instead of subset of taxa blocks, thus masking potentially valuable data for subsets of taxa.

Additionally, all methods are relatively insensitive in detecting heterogeneous sequence divergence within sequence alignments. This is an important deficiency of masking methods, because heterogeneous sequence divergence can cause strong biases in tree reconstructions, for example long branch effects. Therefore, a method which can visualize heterogeneous sequence divergence or alignment ambiguity related to single taxa or subsets of taxa within alignments would thus be a useful complement to masking approaches. It offers the chance to identify taxa which will most likely be misplaced in trees and which negatively influence the tree-likeness of the data. An ideal would be to be able to place a question mark at suspicious branches within a tree.

For this purpose, we developed AliGROOVE, a new tool to visualize the extent of sequence similarity and alignment ambiguity in multiple alignments which can help to detect strongly derived sequences that, most probably, will negatively influence tree reconstruction methods. We implemented an adaptation of the recently published ALISCORE masking algorithm [34, 35]. ALISCORE uses a parametric Monte Carlo resampling within a sliding window to generate profiles of sequence similarity for all pairwise sequence comparisons. These profiles consist of site scores ranging from -1 indicating full random similarity to +1, non-random similarity. AliGROOVE summarises site scores of profiles of sequence similarity normalized over the whole alignment length from each pairwise comparison and translates the obtained scoring distances between sequences into a similarity matrix (Fig. 4.1). It thus delivers information on heterogeneous sequence similarity within the alignment. The colour of each box in the matrix represents the obtained sum of similarity scores between two sequences. Red indicates that ambiguously aligned sequence positions dominate between two sequences while blue indicates the opposite. The more positive or negative the total similarity score between two sequences, the darker the corresponding colour.

The ALISCORE algorithm has been successfully tested in simulations and on real data sets [34, 35]. As a result, ALISCORE was used for alignment masking in recent molecular phylogenetic studies [105–109]. We used simulated data to see

## Chapter 4. AliGROOVE: a new tool to visualize the extent of sequence similarity and alignment ambiguity in multiple alignments

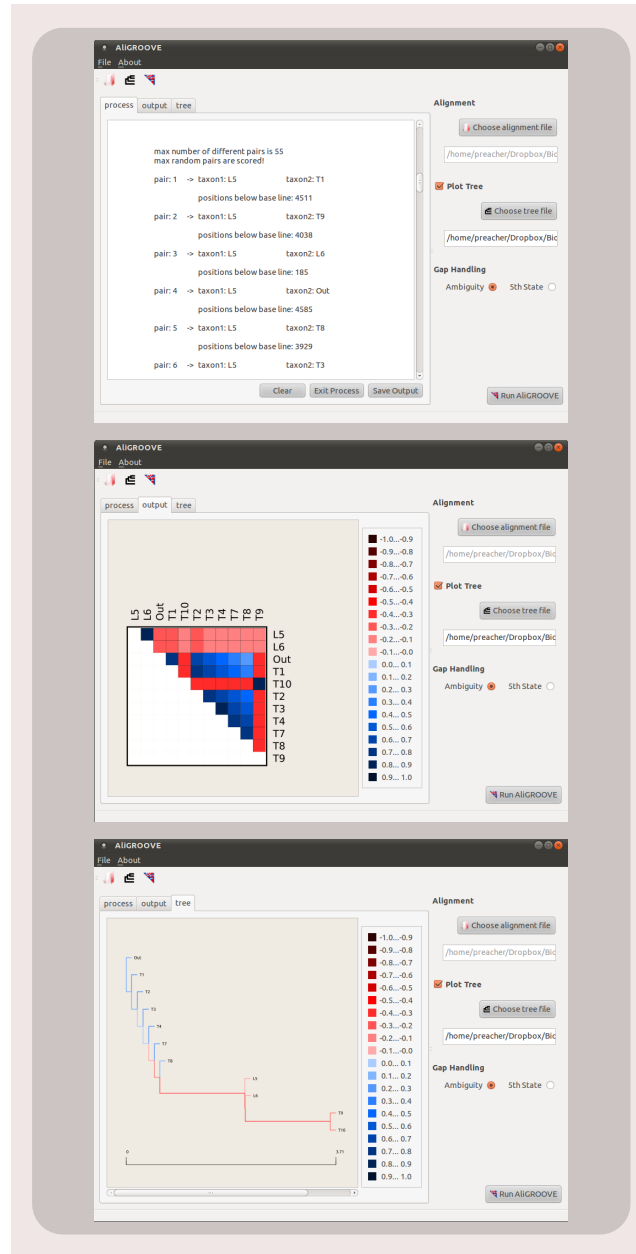


Figure 4.1: **Graphical User Interface (GUI) of AliGROOVE.** AliGROOVE can be directly started via command line or by use of a graphical user interface (GUI). Gaps in multiple sequence alignments can be treated either as fifth state or as ambiguity character. When AliGROOVE is used with a GUI, single process information will be shown in separate process window and can be directly saved as textfile (above). Sequence divergences obtained from single pairwise comparisons are shown in the output window after the process run (bottom).

whether our extension to AliGROOVE is sensitive enough to pick up predominantly, ambiguously aligned single taxa or groups of taxa. Additionally, we applied AliGROOVE on two empirical data (one mitochondrial and one nuclear data set).

#### 4.1.1 AliGROOVE algorithm

The algorithm of AliGROOVE is based on the scoring scheme of ALISCORE [34,35]. ALISCORE uses a sliding window approach to compare two sequences for random similarity within the sliding window. In short, first, the observed mismatch within the sliding window is recorded and secondly, compared with scores of same window size generated by permutations of character states within the sliding window and a predefined neighborhood. If the observed score is better than 95% of all generated permutations, it is considered non-random, otherwise indistinguishable from random similarity. Positions within the sliding window receive a positive sign if non-random and a negative if random. Each position will receive a number of signs corresponding to the size of the sliding window which will finally be summed up and normalized by the sliding window size for each position. A profile of sequence similarity between two sequences will thus show sections in which these two sequences might show non-random similarity and sections of random similarity expressed by negative signs. The AliGROOVE algorithm generates an average over all sites for each pairwise comparison excluding globally invariant sites within the alignment and records these values in a similarity matrix for all pairwise comparisons for a given set of sequences. The entries in this similarity matrix express the average amount of non-random versus random similarity in pairwise comparisons and can thus illustrate heterogeneous signal in the data.

The algorithm is based on either simple match/mismatch scores for nucleotide sequences or on the BLOSUM62 matrix to score aminoacid matches/mismatches. It is thus a relatively simple scoring regime but turned out efficient in simulations and empirical data [34,35,105–109,123].

The AliGROOVE pairwise similarity scores can be directly used to tag potentially unreliable relationships within topologies. To define the reliability of single internal branches, AliGROOVE calculates the average similarity score from all single pairwise similarity scores between taxa which are connected by the respective branch. To determine the reliability of terminal branches, AliGROOVE calculates the average pairwise similarity score from all single similarity scores between the terminal branch and remaining taxa. The tagging of branches is effectively an indirect estimation of reliability of a subset of all possible splits guided by a topology. Calculated reliabilities of single branches are shown colorized in a new tree outfile. The colouring of each branch depends on the obtained similarity score. The tagging colour scheme is identic with the colour scheme that is used for the sequence similarity matrix. An example of the AliGROOVE tagging algorithm is given in Figure 4.2.

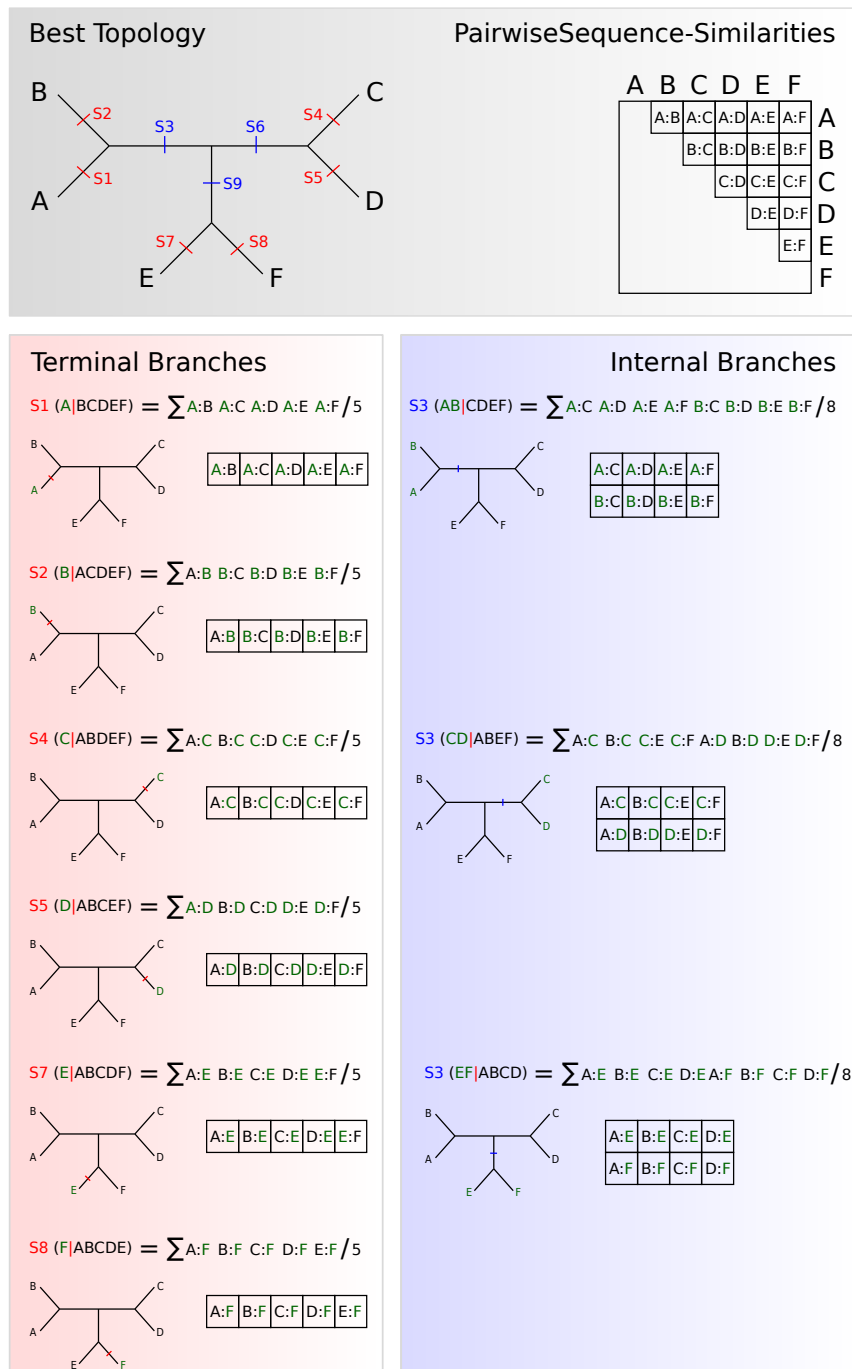


Figure 4.2: **Example of the AliGROOVE tagging algorithm.** Single Branch reliabilities of the given six taxon topology are calculated by using the corresponding sequence similarity matrix. Single reliabilities of the six terminal branches (S1, S2, S4, S5, S7, and S8) are calculated from all single similarity scores between a terminal taxon and the remaining taxa. Single Reliabilities of the three internal branches (S3, S6, S9) are calculated by averaging the total pairwise similarity scores between taxa which are connected by the respective branch.

## 4.2 Material and Methods

### 4.2.1 Simulated data

To test the efficiency of AliGROOVE to detect heterogeneous taxa, we designed two sets of nucleotide data (set A and B) under different 11-taxon topologies (Fig. 4.3–4.4). The topology of the first set up (set A) contains two long terminal branches (*LtB*) (Fig. 4.3). The second setup (set B) contained two long internal branches (*LiB*), separated by one short internal branch (*SiB*) (Fig. 4.4). While lengths of *SiB* and remaining branches (*RB*) are kept constant ( $L_{SiB} = 0.01$ ,  $L_{RB} = 0.1$ ), alignments are generated for each of two different branch lengths of either *LiB* or *LtB* ( $L_{LiB} \vee L_{LtB} = 0.9, 1.5$ ). Sequence length of each alignment was set to 10,000 base positions (bp). All alignments were generated with INDELible v.1.01 [124] using the Jukes-Cantor model (JC) of sequence evolution and a mixed-distribution model of  $\Gamma + I$  for ASRV. All data were simulated with ASRV, shape parameter  $\alpha = 1.0$ , and a proportion of invariant sites  $\rho_{inv} = 0.3$ . ASRV was modelled using a continuous  $\Gamma$ -rate distribution while indel events were not simulated.

Trees of simulated data were inferred with PhyML\_3.0\_linux64 [125, 126]. We analyzed the data with a mixed-distribution model (JC+ $\Gamma$ +I) and correct parameter values ( $\alpha = 1.0$ ,  $\rho_{inv} = 0.3$ ). The number of relative substitution rate categories was set to four ( $c = 4$ ) and tree topologies and branch lengths were optimized. Maximum Likelihood analyses were performed and evaluated with a Perl pipeline, and ran on a Linux Cluster with HP ProLiant DL380 G5 blades (Dual quad core Intel Xeon E5345, 2.33 GHz, 2x 4MB L2-cache, 1333 MHz Bus, 32 GB RAM). For each branch length-combination, we generated 100 data replicates and recorded the frequencies of correct and incorrect tree reconstructions using correct alignments and substitution models.

### 4.2.2 Empirical data

We used AliGROOVE on two kinds of empirical data sets: i) on a masked nucleotide alignment of 148 concatenated nuclear 18S and 28S rRNA arthropod sequences (4102 bp) published by Reumont et al. [81], and ii) on a concatenated unmasked and masked supermatrix alignment (5082 bp) of five mitochondrial genes (Atp6, CoxI, CoxII, CoxIII, and Cytb) downloaded from the NCBI genome data base for 53 chelicerate ingroup taxa and 8 myriapod outgroup taxa. Single mitochondrial genes were aligned with ClustalW [21]. The best ML topology of the mitochondrial data set was estimated using PhyML\_3.0\_linux64 [125, 126] and the GTR+ $\Gamma$ +I model with 1,000 bootstrap replicates. For the mt data, we compared the results of the AliGROOVE approach on the unmasked and masked data. Additionally, we compared tree reconstructions with all taxa and with a data subset in which most divergent taxa identified by AliGROOVE had been removed. We used a resolution score, the total sum of bootstrap values above 50 divided by the number of internal nodes, to compare resolution of trees.

## Implementation of AliGROOVE

AliGROOVE is implemented in Perl and runs on Linux, Mac OS, and Windows operating systems. It can be used via command line or graphical user interface (GUI) (Fig. 4.1). The GUI of AliGROOVE is based on QT, a cross-platform application and GUI framework in C++. AliGROOVE is freely available from <http://software.zfnk.de> or upon authors request.

## 4.3 Results

### 4.3.1 Testing performance on simulated data

Our goal was to show that suspiciously placed taxa or nodes can be associated with high scores of randomness. We simulated sequence alignments under two different topological conditions (see Material and Methods) and applied the AliGROOVE algorithm. Both settings represented 11-taxa trees containing either (A) terminal or (B) long internal branches.

In set A, we simulated multiple data with increasing terminal branch lengths of two taxa and recorded the frequencies of correct and incorrect tree reconstructions. It turns out that as long as terminal branches are correctly placed in the tree using the ML approach with correct model specifications, these terminal branches (L5, L6) show positive (non-random) scores with taxon neighbors in the tree (Fig. 4.3). However, at a terminal branch length of 90x in relation to the internal branch lengths the average similarity score between these long terminal branches and taxon neighbors in the tree drops to only slightly positive scores and the frequency of inferring the correct tree is at 0.53 compared to inferring an incorrect tree with 0.47. It is thus a matter of chance to infer the correct tree, despite correct sequence alignment and the application of the correct substitution model. If taxa with long terminal branches do not have positive scores with other taxa, they are most frequently misplaced. Using the AliGROOVE approach, we would tag these branches as suspicious.

In set B, we simulated multiple data increasing two internal branch lengths and recorded possible errors in tree reconstructions. Again, as soon as these two internal branches join taxa with on average negative scores, tree reconstructions were predominantly not correct (Fig. 4.4). For example in set B taxon T7 or T8 are connected to taxon T5 or T6 via a suspiciously long branch. In set B taxa T7 and T8 become monophyletic instead of being paraphyletic in relation to taxa T9 and T10. In this special case, the two long internal branches overwrite the signal between taxon T7 and T8. Using the ALIGROOVE approach, we would tag the two long internal branches as suspicious and would also place a tag on the common branch of T7 and T8.

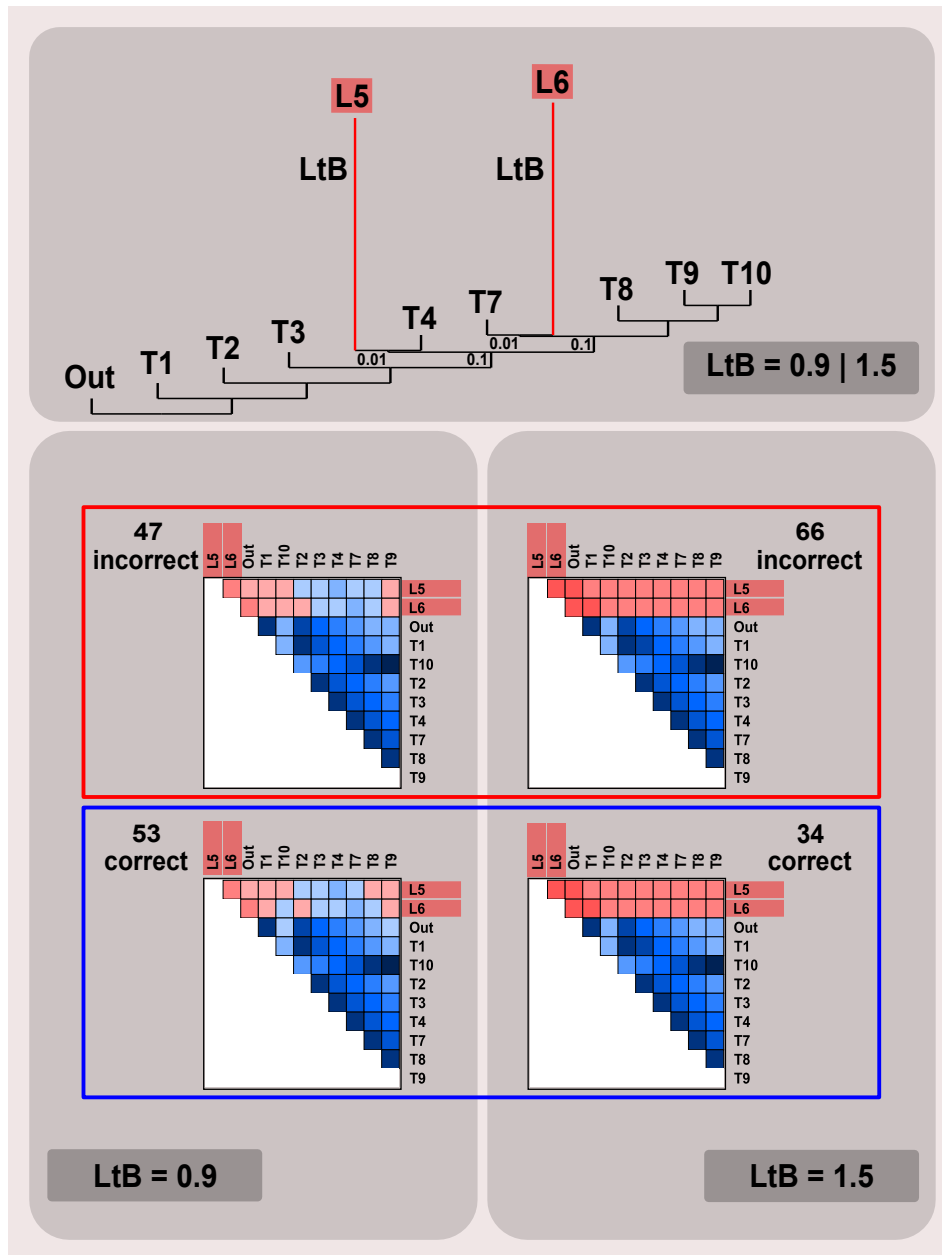


Figure 4.3: AliGROOVE performance on simulated data of topology A. If the length of both long terminal branches ( $LtB$ ) is 90x in relation to the internal branch lengths ( $LtB = 0.9$ ) the average similarity score between these long terminal branches and taxon neighbors in the tree drops to only slightly positive scores. Despite correct substitution model assumptions the frequency of inferring the correct topology is dropped to 0.53. If both  $LtB$ 's are strongly increased ( $LtB = 1.5$ ), both long branches do not have positive scores with other taxa and are most frequently misplaced despite correct substitution model assumptions (0.66).



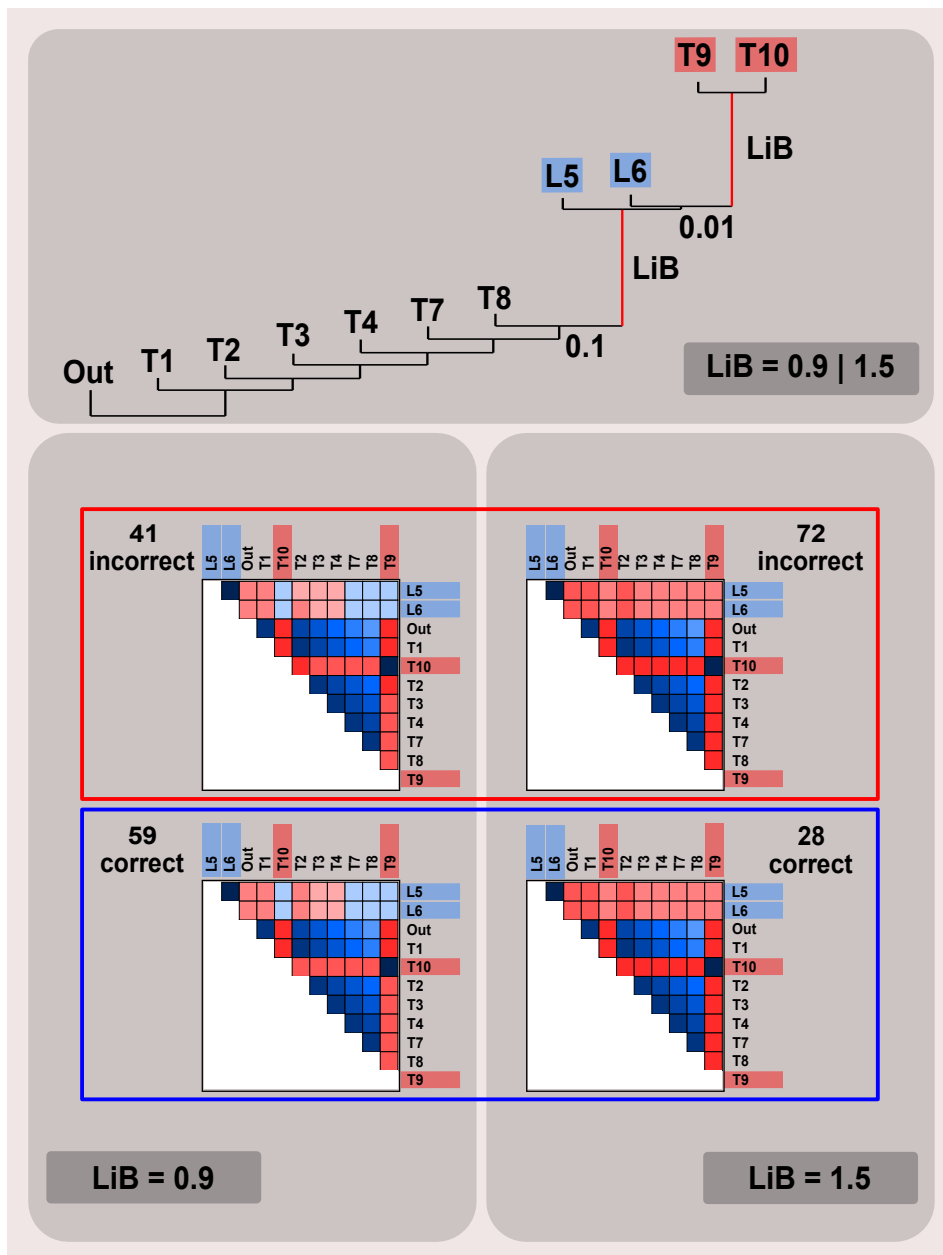


Figure 4.4: **AliGROOVE** performance on simulated data of topology **B**. If the length of both long internal branches (LiB) is 90x in relation to interior internal branch length (LiB = 0.9) the average similarity score of taxa (L5 and L6) between these long internal branches and taxon neighbors in the tree drops to only slightly positive scores. Despite correct substitution model assumptions the frequency of inferring the correct topology is dropped to 0.59. If both LiB's are strongly increased (LiB = 1.5), both taxa (L5 and L6) do not have positive scores with other taxa (except to themselves) and are most frequently misplaced as monophyletic group despite correct substitution model assumptions (0.72).

### 4.3.2 Testing performance on empirical data

#### 4.3.2.1 Mitochondrial Data

In the unmasked data set AliGROOVE analyses identified nearly all Acariformes sequences as strongly derived to each other and to the remaining chelicerate sequences (Fig. 4.5), while pairwise comparisons without Acariformes sequences achieved in most cases positive similarity scores. The bootstrap support for the resolved clade 'Acariformes and Ricinulei' was below 50%. Only *Unionicola* and *Walchia* received positive similarity scores if compared with non-Acariformes sequences. While *Walchia* showed weak positive similarity scores to three other Acariformes genera (*Unionicola*, *Ascoschoengastia*, and *Leptotrombidium*), sequences of the *Unionicola* were only scored positive in comparisons with Acariformes sequences if compared to *Walchia*. Our Maximum Likelihood (ML) topology received maximum bootstrap support for a sister group relationship of *Walchia* and *Ascoschoengastia*, next to *Leptotrombidium*. The resolution score (RS) of this tree was  $RS = 77.33$ . Removing the red branches of the unmasked data and reaping tree reconstructions did not improve the resolution score ( $RS = 77.1$ ). Several additional branches appear not tagged in red, indicating that there is still quite some noise in the data. In comparison, the masked data contained much less noise, but is also characterized by a lower resolution score.

#### 4.3.2.2 Nuclear Data

AliGROOVE identified the sequences of Remipedia and Cephalocarida as most divergent within the masked alignment (Fig. 4.6). Both taxa show long branches in the time-heterogeneous consensus tree of Reumont et al. [81] and are placed with only moderate support in this tree. The Cephalocarida clustered even within Hexapods in the time-homogeneous consensus tree of Reumont et al. [81]. While the remipede sequence scored weakly positive in most sequence comparisons, nearly all pairwise comparisons with Cephalocarida received negative similarity scores. The highest extent of random similarity was found between Remipedia and Cephalocarida. The AliGROOVE algorithm clearly identified the most problematic sequences in the data set.

## 4.4 Discussion

Tree reconstruction approaches in particular Maximum Likelihood approaches are extremely efficient in translating structure in sequence alignment data into trees. However, even these best available approaches can become inconsistent if signal in the data is heterogeneous or if assumptions about substitution processes are clearly misspecified. In these cases, seemingly robust reconstructions might be biased. Another matter of concern is the decrease of robustness values of tree reconstructions using bootstrapping or posterior probabilities if the data is very noisy or not tree-like. Alignment masking has been put forward to improve the tree-likeness of the

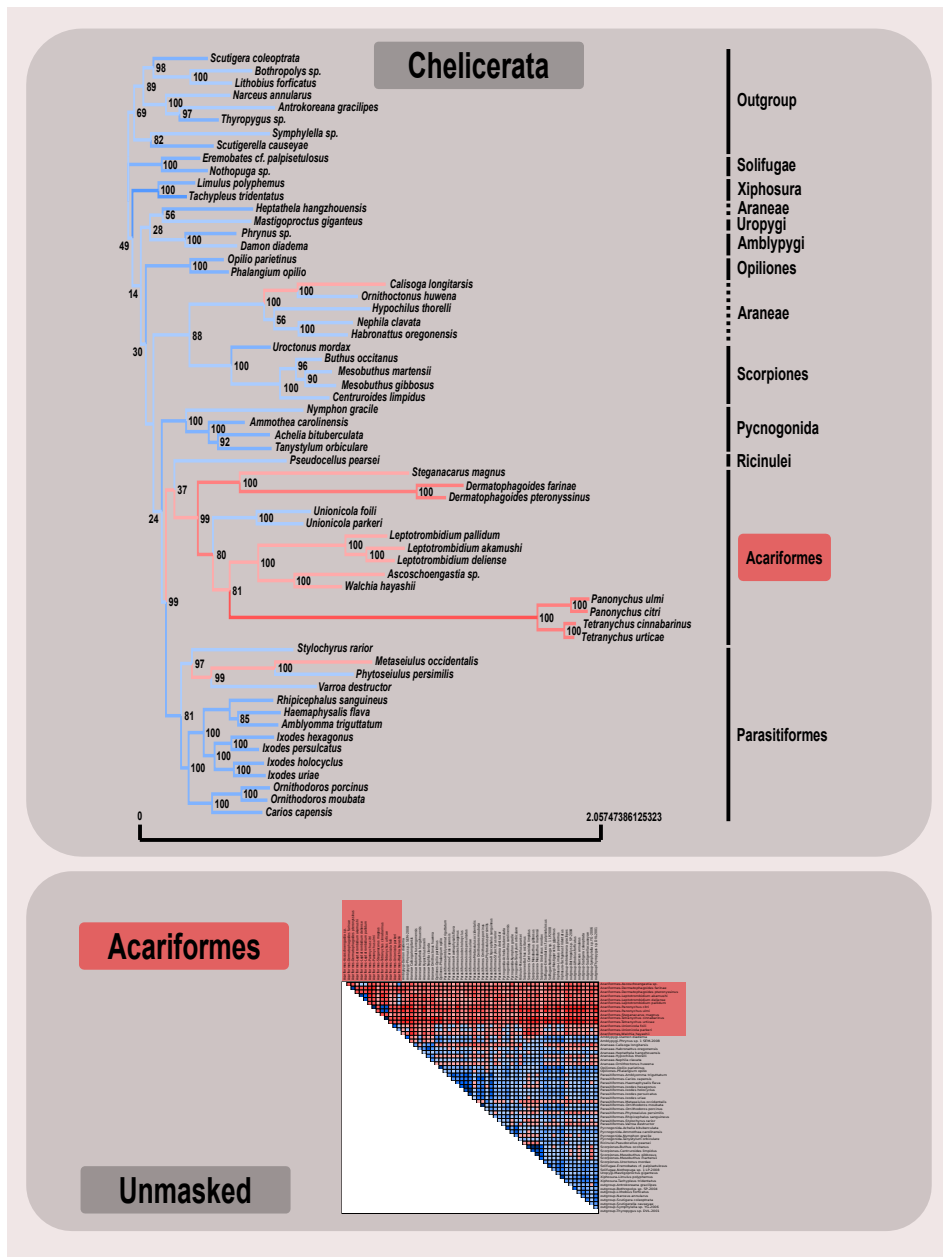


Figure 4.5: AliGROOVE performance on a unmasked mitochondrial alignment of five mitochondrial genes. Nearly all Acariformes sequences are identified as strongly derived to each other and to the remaining chelicerate sequences (below). The clade 'Acariformes and Ricinulei' is also not sufficient supported (bootstrap support below 50%).

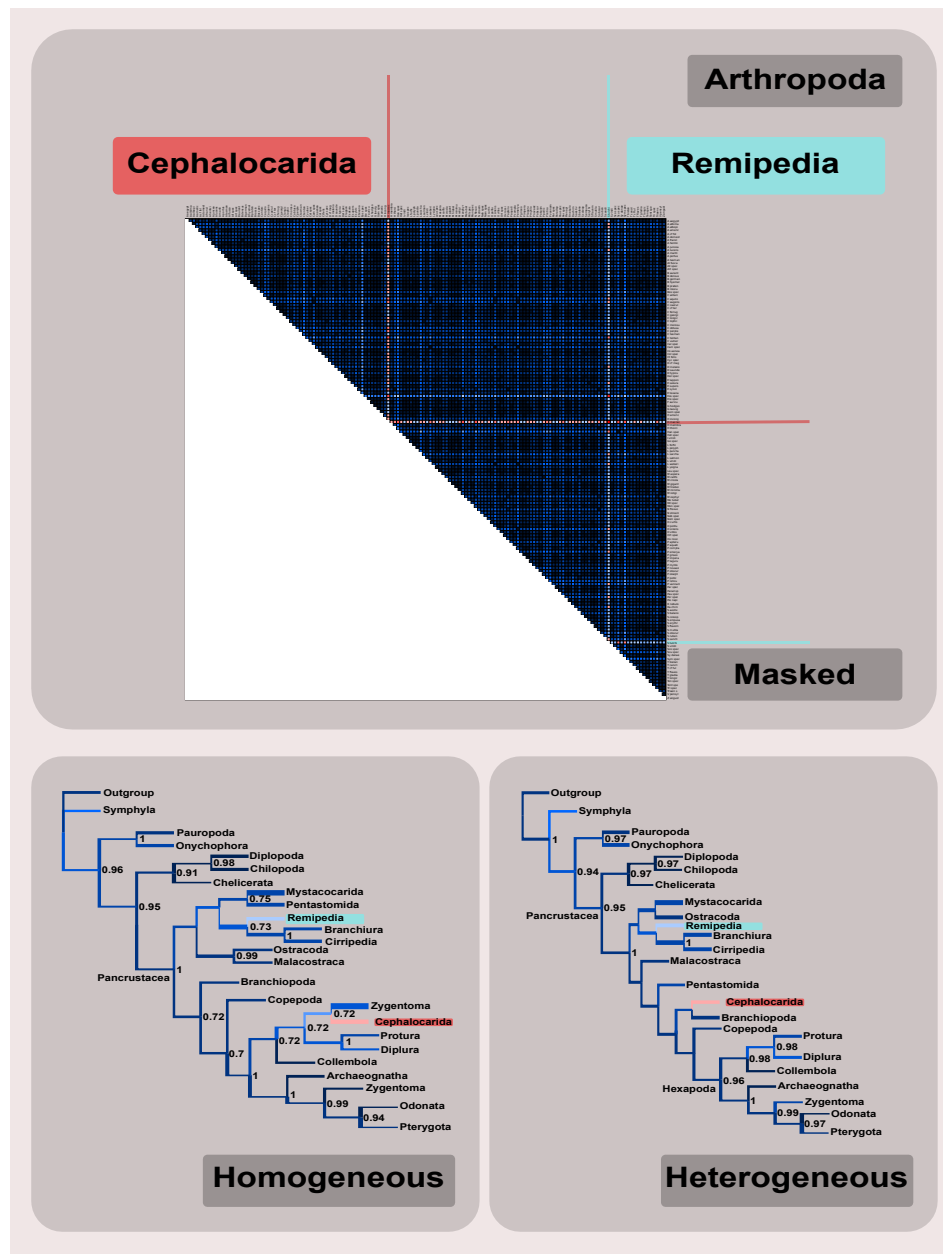


Figure 4.6: AliGROOVE performance on a masked nucleotide alignment of 148 concatenated nuclear 18S and 28S rRNA arthropod sequences. While the sequence of Cephalocarida was identified as strongly derived in all sequence comparisons, the sequence of the Remipedia emerged as little more similar to remaining sequences. Both taxa show long branches in the time-heterogeneous and time-homogeneous consensus tree of Reumont et al. 2009 [81] (branch length relations and support values below 0.7 bayesian posterior probability are not shown here). While the Remipedia clustered always next to other Crustaceans, the Cephalocarida clustered even within Hexapods in the time-homogeneous consensus tree. In both topologies, the placement of Cephalocarida and Remipedia shows only weak support (below 0.7 bayesian posterior probability).

alignment data. The basic idea of masking is the filtering of noisy alignment blocks prior to tree reconstructions. It has been shown that this masking is successful in improving the signal-to-noise ratio in sequence alignments. Here we show that the sliding window approach as it is used in ALISCORE [34] can be used to identify single taxa or subsets of taxa which show predominantly randomized sequence similarity in comparison to other taxa. Removal of these taxa can potentially also increase the tree-likeness of the data and thus help to improve the reliability of tree reconstructions. The basic idea is that single taxa can be misplaced or induce strong biases in tree reconstructions due to their strong sequence divergence. This misplacement can even be robustly supported by bootstrapping or posterior probability values. Our approach offers the chance to identify taxa which are robustly placed in trees but show predominantly randomized sequence similarity to other taxa.

The sequences of the Cephalocarida specimen of the study of Reumont et al. [81] shows predominantly randomized sequence similarity in most pairwise comparisons. We would therefore predict, that these sequences do not help to robustly place the taxon in the tree of arthropods. Reumont et al. [81] report exactly this as the time-homogeneous approach placed Cephalocarida within Enthognatha next to Nonoculata (Protura + Diplura), the time-heterogeneous analysis clustered Cephalocarida as sister group to Branchiopoda (Fig. 4.6). Although congruent with some morphological data [127], the clade Branchiopoda + Cephalocarida is only weak supported in the time-heterogeneous consensus tree of Reumont et al. [81] and is in conflict with other recent molecular studies [128,129]. In the data of Reumont et al. [81] sequences of Remipedia also show high sequence divergence. The position of this taxon as a sistergroup to Branchiura and Cirripedia received low node support in the analyses of Reumont et al. [81] as it would have been predicted from the AliGROOVE similarity matrix. This phylogenetic position is also in conflict with another recent molecular studies [130]. Thus, the AliGROOVE approach demonstrates its usability with this data.

The mitochondrial data of chelicerates clearly shows strong heterogeneity in the similarity matrix. Specimens of Acariformes display mostly random similarity to all other sequences in the data and it would have been predicted from this pattern, that these sequences can not be robustly placed in the tree or are potentially misplaced despite robust support (Fig. 4.5). Again, this is exactly what we see in the tree reconstructions, as Acariformes are sistergroup to Ricinulei and together with Parasitiformes sistergroup to Pycogonida with low support which is considered implausible by many specialists. However, removal of these sequences did not improve resolution in this case.

The simulation results and the analyses of empirical data show that the sliding window approach has some predictive power, therefore we consider this characteristic a major advantage over all character based masking approaches in phylogenetics. It also offers the possibility of excluding taxa based on a formal argument in comparison with excluding taxa based exclusively on the evaluation of branch lengths.

# Long branch effects distort Maximum Likelihood phylogenies in simulations despite selection of the correct model

---

## Contents

---

<b>5.1</b>	<b>Introduction</b>	<b>57</b>
<b>5.2</b>	<b>Material and Methods</b>	<b>58</b>
5.2.1	Simulations	58
5.2.2	Maximum Likelihood Analyses	61
5.2.3	Scoring	61
<b>5.3</b>	<b>Results</b>	<b>61</b>
5.3.1	Topology A	62
5.3.2	Topology B	62
5.3.3	Maximum Likelihood Values	62
<b>5.4</b>	<b>Discussion</b>	<b>67</b>
<b>5.5</b>	<b>Additional Files</b>	<b>68</b>

---

---

**Abstract:** The aim of this study was to test the robustness and efficiency of Maximum Likelihood with respect to different long branch effects on multiple taxon trees. We simulated data of different alignment lengths under two different 11-taxon trees and a broad range of different branch length conditions. The data was analyzed with Maximum Likelihood under the true model parameters as well as estimated and incorrect assumptions about among-site rate variation. If length differences between connected branches strongly increase, tree inference with the correct likelihood model assumptions can fail. Incorporating among-site rate estimates of mixed-distribution models ( $\Gamma+I$ ) increases the robustness of Maximum Likelihood in comparison with models using only  $\Gamma$ . The results show that for some topologies and branch lengths the reconstruction success of Maximum Likelihood under the correct model is still low for alignments with a length of 100,000 base positions. Interestingly, too low values of the shape parameters can lead to a reduction of long branch effects. Altogether, the high confidence that is put in Maximum Likelihood trees is not always justified even if alignment lengths exceed 10,000 base positions.

**Keywords:** Maximum Likelihood, Long Branch Attraction, Model Assumptions, Rate Heterogeneity, Parameter Estimation

---

## 5.1 Introduction

Maximum likelihood (ML) tree inference has been shown to be statistically consistent for binary trees with finite branch and infinite sequence lengths when model and model parameter assumptions are correct [42, 131–134]. Thus, ML tree inference will converge on the true tree as more and more data are accumulated [60, 134]. Additionally, ML is said to be robust against model violations [42, 43, 60, 135–138] and thus, even oversimplified likelihood models are said to find the correct tree in most instances if branch lengths are well balanced [139].

Undoubtedly, the ML method is more robust and more efficient than other methods (e.g. [42, 43, 45, 58, 60, 135, 136, 140–146]). This has led to a widespread application and acceptance of ML tree inference. The degree of robustness and efficiency has however mainly been assessed using 4-taxon tree simulations. Setups in which ML methods can potentially fail or become inefficient on trees with more than four taxa have not been studied in great detail (e.g. [54, 56, 61, 147]). Thus, we address the robustness and efficiency of ML methods to different long branch effects in an 11-taxon setup. We show that ML methods indeed reconstruct correct topologies in a wide parameter range, but we also discovered instances in which ML methods reconstruct the wrong tree for relatively long alignments even under correct model assumptions. Exactly these effects have not been studied previously and are probably frequent in empirical data.

It is well known that if among-site rate variation (ASRV) is ignored in tree reconstruction, the ML approach underestimates substitution rates, which becomes progressively worse with increasing evolutionary distances [148]. Ignoring ASRV makes ML tree inference susceptible to long branch attraction [43, 47, 55, 57, 58, 60, 133, 138, 140, 142, 145]. Therefore, ASRV is, apart from considering multiple substitutions, the most important advance brought by model-based reconstruction methods. Three possibilities to account for rate variation are the “invariant sites model (I)”, the “ $\Gamma$  distributed rates model” ( $\alpha$  shape parameter) and a combination of both models ( $\Gamma$ +I). The invariant sites parameter assumes an estimated fraction of sites as invariable while remaining sites are assumed to evolve at an equal rate. Under the  $\Gamma$ -model, rate variation among sites is modelled using a  $\Gamma$  distribution.

Older studies argue that combining both models ( $\Gamma$ +I) to a mixed-distribution model should lead to a significant improvement of the heterogeneity estimation in comparison to invariable sites- or  $\Gamma$ -model estimates alone [58, 59, 136, 149, 150]. However, recently published studies relied on the exclusive application of the restricted  $\Gamma$ -model (e.g. [108, 130, 151–153]). One argument is, that parameters of the  $\Gamma$ - and invariant sites model cannot be optimized independently. This can lead to problems during model parameter optimization due to multiple optima in the likelihood function [154, 155]. The shape parameter of the  $\Gamma$  distribution and the invariant sites estimation are indeed strongly correlated and subject to large sample variance [59, 149, 156]. The correlation makes it difficult to distinguish between truly invariable and slowly evolving sites, especially in the case of alignments with a small number of sequences. However, if many taxa are included ( $N > 20$ ), the mixed-



distribution model can be reliably estimated [59, 156]. Parameter correlation can also be seen as an advantage. Erroneous estimates of one parameter can be compensated by the other. Erroneous estimates of both together can fit the data such that the likelihood score changes only marginally [59]. Such an error compensation of rate estimates is not possible under  $\Gamma$ -model estimation alone. We have addressed the important question whether  $\Gamma$ +I models are superior over pure  $\Gamma$  models and whether the parameters could be estimated correctly for a taxon set of just 11 taxa. Furthermore, we investigated how deviations from the simulated  $\Gamma$  parameter effects the reconstruction success.

No model can be assumed to be entirely correct for real data [145]. Effects of Long branch attraction (LBA) are therefore not only theoretical concepts, but also real phenomena [65, 147, 157]. The “classical case of long branch attraction” (Fig. 5.1a) which is caused by the misleading effect of parallel substitutions on long branches [42] is well studied and affects mainly the maximum parsimony method. In a topology of more than four taxa, the classical case can be categorized into different subclasses. (i) The case in which two short terminal branches are grouped together because the rest of the tree constitutes two long branches on either side of the two short branches, shall be referred to LBA class I (Fig. 5.1b). In this case, the long branch effect might not be immediately obvious since the long branches are hidden and are made up of larger groups of taxa. (ii) The case in which two long terminal taxa or large groups of taxa lead to random errors in the topology is referred to as LBA class II. Finally, the case in which the two long terminal branches are incorrectly grouped together shall be referred to as LBA class III (Fig. 5.1c). For infinitely long sequences, ML should still reconstruct the correct tree in all of these scenarios, but the robustness and efficiency might vary. Even though it seems at first sight that the three cases should yield comparable reconstruction successes under the ML method, considerably different reconstruction successes are obtained in this study.

## 5.2 Material and Methods

### 5.2.1 Simulations

We designed two sets of data simulations under different topologies (Fig. 5.2). The first set was characterized by a stepwise elongation of two terminal non neighboring branches (*BI2*) for different internal branch lengths (*BI1*) (Fig. 5.2a), the second set was characterized by a stepwise elongation of two internal branches (*BI2*) for different lengths of an intermediate internal branch (*BI1*) (Fig. 5.2b). Trees consisted of 11 taxa in which lengths of all remaining branches (*RB*) are kept constant ( $L_{RB} = 0.1$ ). For each length of *BI1* (0.01, 0.05, 0.1, 0.3, 0.5), we increased the length of *BI2* from 0.1 to 1.5 in steps of 0.2. Thus, branch length ratios *BI2/BI1* ranged from three to 150. All alignments were generated with INDELible v.1.01 [124] using the Jukes-Cantor model (JC) of sequence evolution and a mixed-distribution model of  $\Gamma$ +I for ASRV. All data were simulated with ASRV, shape parameter

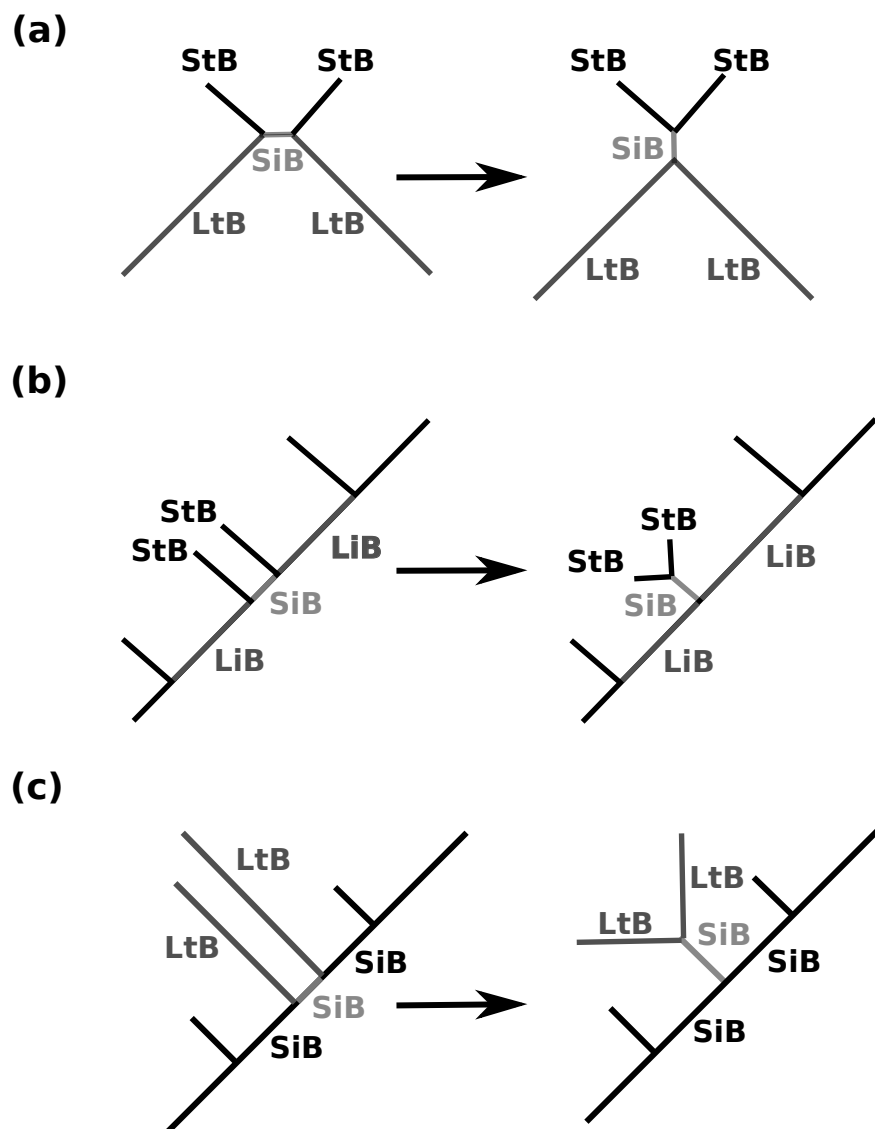
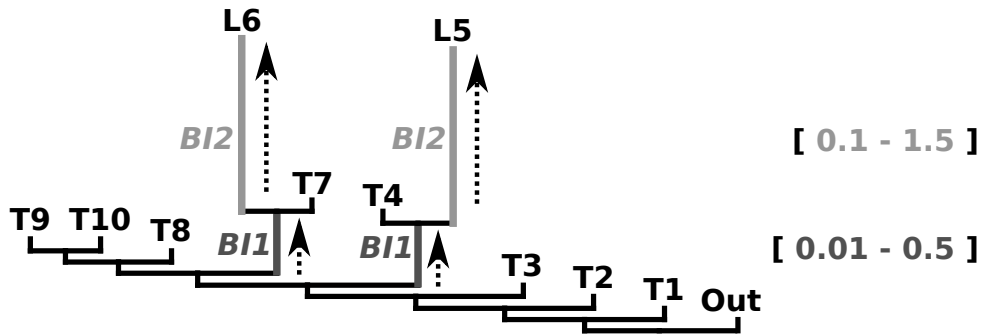


Figure 5.1: **Subclasses of the “classical long branch attraction”.** (a) The “classical long branch attraction” case and three subclasses in the presence of more than 4 taxa: (b) class I effect: two short terminal branches (StB), separated by a short internal branch (SiB) are grouped together, the rest of the tree is found at the ends of two long internal branches (LiB) on either side of the two short branches. (c) class III effect: Two long terminal branches (LtB) are attracted in direct analogy to the “classical” case (a).

$\alpha = 1.0$ , and a proportion of invariant sites  $\rho_{inv} = 0.3$ . ASRV was modelled using a continuous  $\Gamma$ -rate distribution while indel events were not simulated. For each branch length-combination of  $BI1$  and  $BI2$ , we simulated the evolution of 100 data replicates for each sequence length (2,000, 3,000, 4,000, 10,000 and 100,000 bp). The JC model has been chosen for the simulations (i) since it is better understood than other models of sequence evolution and (ii) since LBA effects are expected to be worse under more complicated models.

**(a) Topology A**



**(b) Topology B**

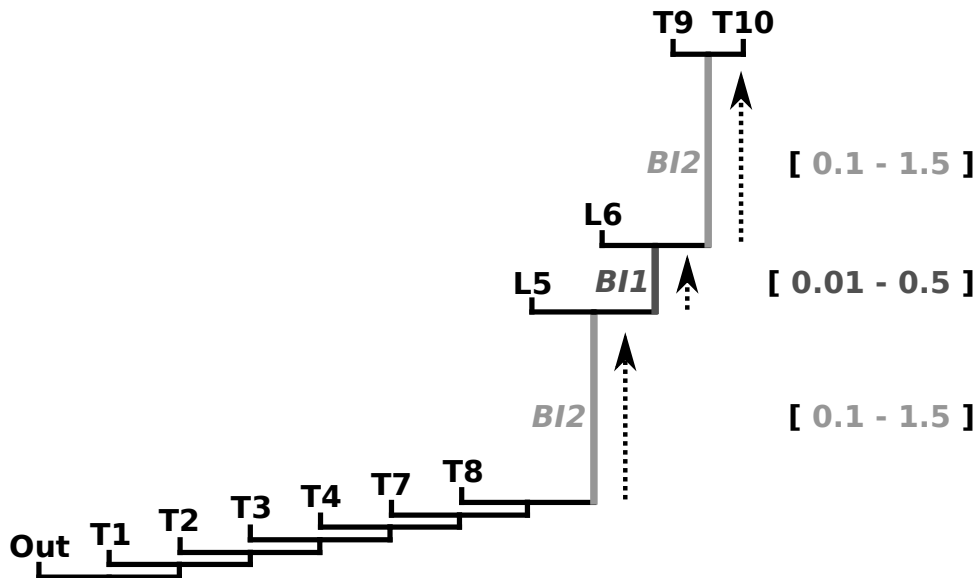


Figure 5.2: **Two sets of simulations.** Given model topology for a) Topology A: stepwise elongation of two terminal branches ( $BI2$ ) under different ancestral branch lengths ( $BI1$ ) and b) Topology B: stepwise elongation of two internal branches ( $BI2$ ) under different lengths of an intermediate branch ( $BI1$ ).

Table 5.1: **The used model parameter settings of ASRV for Maximum Likelihood analyses.** Single settings included either  $\Gamma$  or  $\Gamma$ +I parameters (fixed or estimated). Simulated ASRV as well as model parameter setting for additional simulations/analyses of alignment length of 100,000 base positions are highlighted bold.

		$\Gamma$	$I$	
JC	+	0.1		
JC	+	0.1	+	0.3
JC	+	1.0		
<b>JC</b>	+	<b>1.0</b>	+	<b>0.3</b>
JC	+	100		
JC	+	100	+	0.3
JC	+	estimate		
JC	+	estimate	+	0.3
JC	+	estimate	+	estimate

### 5.2.2 Maximum Likelihood Analyses

Trees were inferred with the Jukes-Cantor (JC) model under different parameter settings using PhyML\_3.0\_linux64 [125, 126] (Tab. 5.1). We analyzed the data either (i) with a mixed-distribution model (JC+ $\Gamma$ +I) or (ii) with  $\Gamma$  distributed rates, but without estimating a fraction of invariant sites (JC+ $\Gamma$ ). The  $\Gamma$  shape parameter  $\alpha$  was either estimated from the data or set to  $\alpha = 0.1$ ,  $\alpha = 100$ , or  $\alpha = 1.0$  (correct simulated value) in which  $\alpha = 100$  is assumed as an approximation to non-ASRV. The fraction of invariant sites was set to  $\rho_{inv} = 0.3$  or estimated (Tab. 5.1). For the alignment length of 100,000 bp, reconstruction was only performed under the correct model parameters. The number of relative substitution rate categories was set to four ( $c = 4$ ) and tree topologies and branch lengths were optimized (heuristic search). Maximum likelihood analyses were performed and evaluated with a Perl pipeline for automated long branch tests, and ran for three months on a Linux Cluster with HP ProLiant DL380 G5 blades (Dual quad core Intel Xeon E5345, 2.33 GHz, 2x 4MB L2-cache, 1333 MHz Bus, 32 GB RAM).

### 5.2.3 Scoring

Wrong topologies were classified into LBA class I, II and III effects (Fig. 5.2). Wrong topologies for which only one branch had been misplaced were collectively classified as “random topological errors” (class II).

## 5.3 Results

Selected results of ML reconstructions for  $\alpha = 0.1$  under the mixed-distribution model (Jukes-Cantor+ $\Gamma$ +I) and the  $\Gamma$  distributed model (JC+ $\Gamma$ ) are shown in Fig-

ure 5.3a and Figure 5.3b. Each individual plot corresponds to a fixed internal branch length of  $BI1$  (Fig. 5.2), specific model assumptions and an increase of two neighbouring branches ( $BI2$ ) for alignment lengths of 2,000, 3,000, 4,000, and 10,000 base positions (bp). Complete results are presented as electronic supplementary file ES9. Branch length combinations for which class I-III effects predominate among observed model assumptions are listed in Table 5.2 and plotted in Figure 5.4. Figure 5.5 illustrates the reconstruction success for topologies A and B when model assumptions are correct and the alignment length is 100,000 bp.

### 5.3.1 Topology A

If the true proportions of invariant sites ( $\rho_{inv} = 0.3$ ) and ASRV ( $\alpha = 1.0$ ) are assumed for datasets of topology A, ML is able to infer predominantly correct trees (Fig. 5.2a) under most of the ancestral branch lengths ( $BI1 > 0.01$ ) even if terminal branch lengths are extremely long. However, ML performs worse if the proportion of invariant sites and/or ASRV are not assumed (Fig. 5.4 and Tab. 5.2). ML also performs worse if a proportion of invariant sites has not been included at all. In both instances, “classical long branch effects” of long terminal branches (class III) and other random topological errors (class II) are present independent of alignment lengths.

While topological random errors (class II) predominate tree inference even under correct model assumptions ( $\alpha = 1.0$ ;  $\rho_{inv} = 0.3$ ) and moderate sequence lengths of 10,000 bp when  $BI1$  is very small ( $BI1 = 0.1$ ), ML correctly resolves nearly all trees under these conditions when sequence lengths are extended to 100,000 bp (Fig. 5.5a). In general, the performance of ML inference is mostly afflicted by distinct branch length differences, less so by wrong model assumptions.

### 5.3.2 Topology B

Even if the true proportion of invariant sites ( $\rho_{inv} = 0.3$ ) and ASRV ( $\alpha = 1.0$ ) are assumed, ML is not able to infer correct trees of topology B (Fig. 5.3b) if the internal branch lengths  $BI1$  are small ( $BI1 = 0.01$ ) and the internal branch lengths  $BI2$  are large ( $BI2 \geq 1.3$ ) (Fig. 5.2b, 5.4, and Tab. 5.2). Wrong trees do not disappear when sequence alignment lengths rise to 100,000 bp (Fig. 5.5b). The occurrence of LBA class I is more frequent when model assumptions are misspecified, in particular when the proportion of invariant sites is not estimated.

### 5.3.3 Maximum Likelihood Values

Likelihood values of single trees become higher if among site rate variation is considered. Within single parameter setups all trees affected by long branch artifacts show nearly identical likelihood scores as correct resolved topologies of corresponding sequence lengths. Likelihood values of all reconstructed trees corresponding to the results of Figure 5.4 are shown in the electronic supplementary file ES10. Distinct

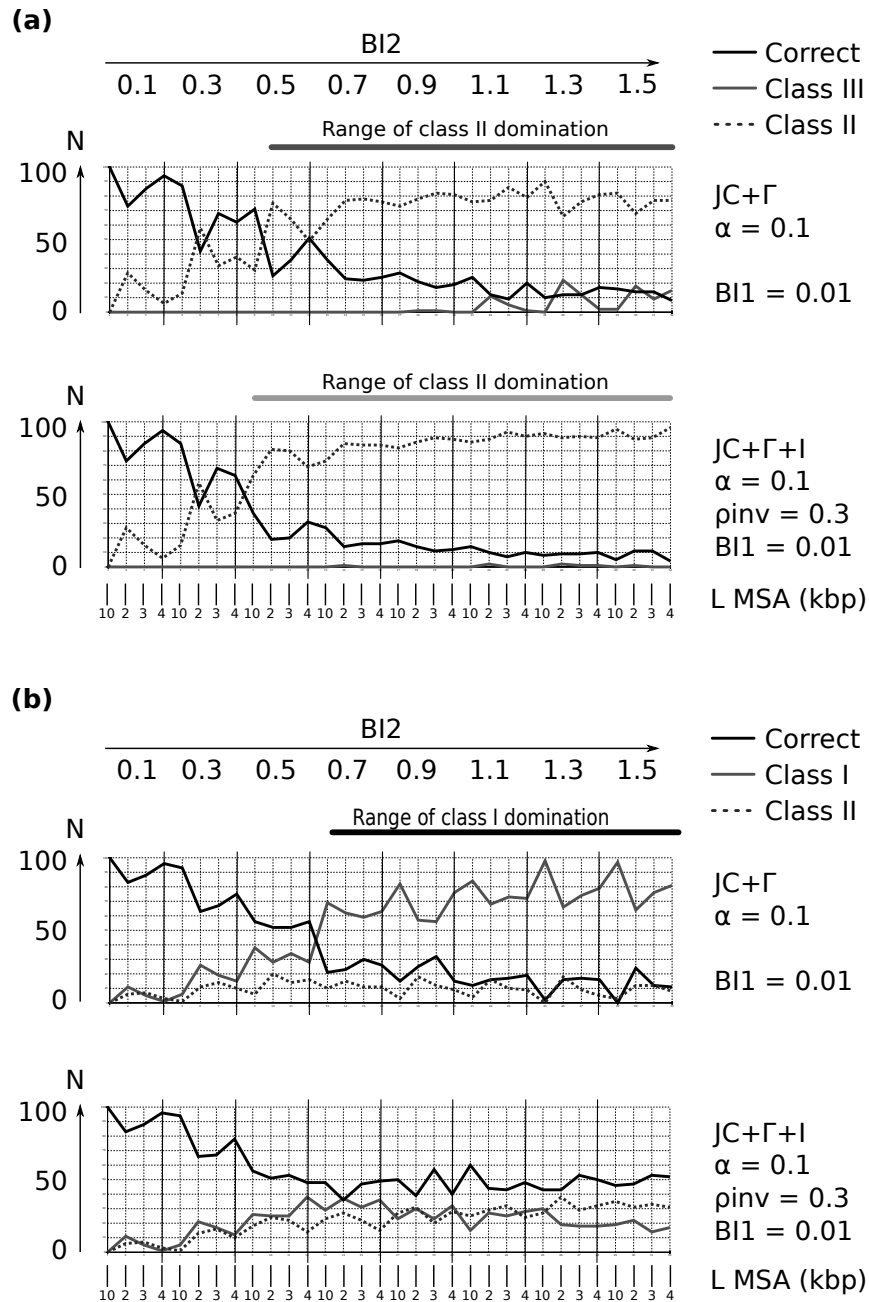


Figure 5.3: Typical examples for correct and wrong topologies (“classic long branch effects” (class III), “hidden long branch effects” (class I), and “random errors” (class II)). Inferred from 100 simulation repeats for each branch length combination and alignment length. Each individual plot corresponds to a fixed branch length of  $BI1 = 0.01$  (Fig. 5.2) and fixed reconstruction scheme with the models JC+ $\Gamma$  ( $\alpha = 0.1$ ) or JC+ $\Gamma$ +I ( $\alpha = 0.1$ ;  $\rho_{inv} = 0.3$ ). Branch length differences increase from left to right by increasing  $BI2$  in discrete elongation steps (0.1-1.5). Four successive data points (belonging to one cell in the plot) correspond to four alignment lengths (10,000, 2,000, 3,000, 4,000). Alignment corresponding branch lengths of  $BI2$  are shown above each subfigure. The y-axis depicts the reconstruction success of the 100 simulation repeats (N) for a) Topology A (Fig. 5.2a) and b) Topology B (Fig. 5.2b).

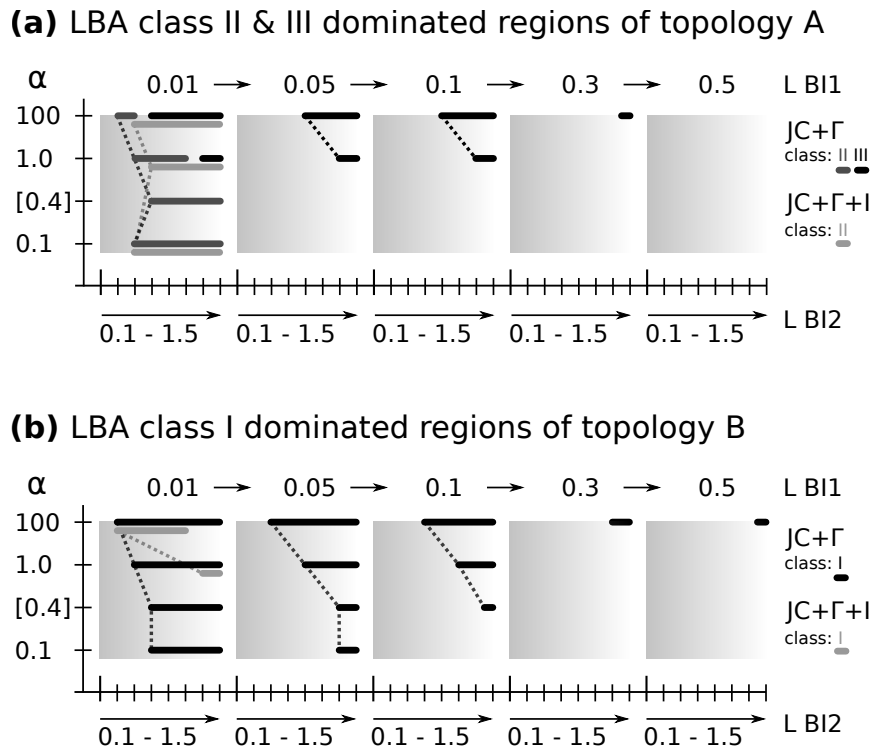


Figure 5.4: **Plots of long branch attraction (LBA)**. Ranges of branch length differences between *BI1* and *BI2* (see Fig. 5.2) in which LBA dominated tree reconstruction under investigated model assumptions are summarized over all alignment lengths. Dominated ranges are displayed in bar charts, based on the selected  $\alpha$  value (100, 1.0, 0.4, 0.1  $\leftrightarrow$  y-axis) and the inclusion or exclusion of an invariant sites model (JC+ $\Gamma$  or JC+ $\Gamma$ +I). Note that if  $\Gamma$  was estimated alone,  $\alpha$  was accurately estimated to 0.4. Domination ranges found under JC+ $\Gamma$  are shown for each  $\alpha$  in the upper bars, domination ranges investigated under JC+ $\Gamma$ +I in the lower bars. Single bars correspond to fixed ranges of lengths for *BI1* and *BI2* in which lengths of *BI2* increase from 0.1-1.5 within each box (x-axis; lower scale). Length of *BI1* increases with each box from 0.01-0.5 (x-axis; upper scale). a) Domination of “classical long branch effects” (class III) and “random errors” (class II) found in topology A (Fig. 5.2a), b) domination of “hidden long branch effects” (class I) found in topology B (Fig. 5.2b).

Table 5.2: **Single domination ranges refer to specific  $\alpha$  values.** Length categories of *BI2* to fix lengths of *BI1* (0.01, 0.05, 0.1, 0.3, 0.5) in which LBA class I-III dominate tree reconstructions of corresponding topologies A and B (Fig. 5.2) when the invariant sites model is ignored (JC+ $\Gamma$ ) or included (JC+ $\Gamma$ +I). If  $\Gamma$  was estimated alone,  $\alpha$  was estimated to 0.4. Domination of “classical long branch effects” (class III) and “random errors” (class II) appears in reconstructions of Topology A (Fig. 5.2a) “hidden long branch effects” (class I) dominate tree reconstructions under certain branch lengths of Topology B (Fig. 5.2b).

<u>JC+<math>\Gamma</math></u>	<u>class I</u>	<u><math>\Gamma</math></u>	<u>BI1</u>	<u>BI2</u>
		100	0.01	0.3 $\rightarrow$ 1.5
			0.05	0.5 $\rightarrow$ 1.5
			0.1	0.7 $\rightarrow$ 1.5
			0.3	1.3 $\rightarrow$ 1.5
			0.5	1.5 $\rightarrow$ 1.5
		1.0	0.01	0.5 $\rightarrow$ 1.5
			0.05	0.9 $\rightarrow$ 1.5
			0.1	1.1 $\rightarrow$ 1.5
	JC+ $\Gamma$ (estimated)	0.4	0.01	0.7 $\rightarrow$ 1.5
			0.05	1.3 $\rightarrow$ 1.5
			0.1	1.5 $\rightarrow$ 1.5
		0.1	0.01	0.7 $\rightarrow$ 1.5
			0.05	1.3 $\rightarrow$ 1.5
	<u>class II</u>	<u><math>\Gamma</math></u>	<u>BI1</u>	<u>BI2</u>
		100	0.01	0.3 $\rightarrow$ 0.5
		1.0	0.01	0.5 $\rightarrow$ 1.1
	JC+ $\Gamma$ (estimated)	0.4	0.01	0.7 $\rightarrow$ 1.5
		0.1	0.01	0.5 $\rightarrow$ 1.5
	<u>class III</u>	<u><math>\Gamma</math></u>	<u>BI1</u>	<u>BI2</u>
		100	0.01	0.7 $\rightarrow$ 1.5
			0.05	0.9 $\rightarrow$ 1.5
			0.1	0.9 $\rightarrow$ 1.5
			0.3	1.3 $\rightarrow$ 1.5
		1.0	0.01	1.3 $\rightarrow$ 1.5
			0.05	1.3 $\rightarrow$ 1.5
			0.1	1.3 $\rightarrow$ 1.5
<u>JC+<math>\Gamma</math>+I</u>	<u>class I</u>	<u><math>\Gamma</math></u>	<u>BI1</u>	<u>BI2</u>
		100	0.01	0.3 $\rightarrow$ 1.1
		1.0	0.01	1.3 $\rightarrow$ 1.5
	<u>class II</u>	<u><math>\Gamma</math></u>	<u>BI1</u>	<u>BI2</u>
		100	0.01	0.5 $\rightarrow$ 1.5
		1.0	0.01	0.7 $\rightarrow$ 1.5
	JC+ $\Gamma$ (estimated)	0.4	0.01	0.7 $\rightarrow$ 1.5
		0.1	0.01	0.5 $\rightarrow$ 1.5
			0.05	1.5 $\rightarrow$ 1.5



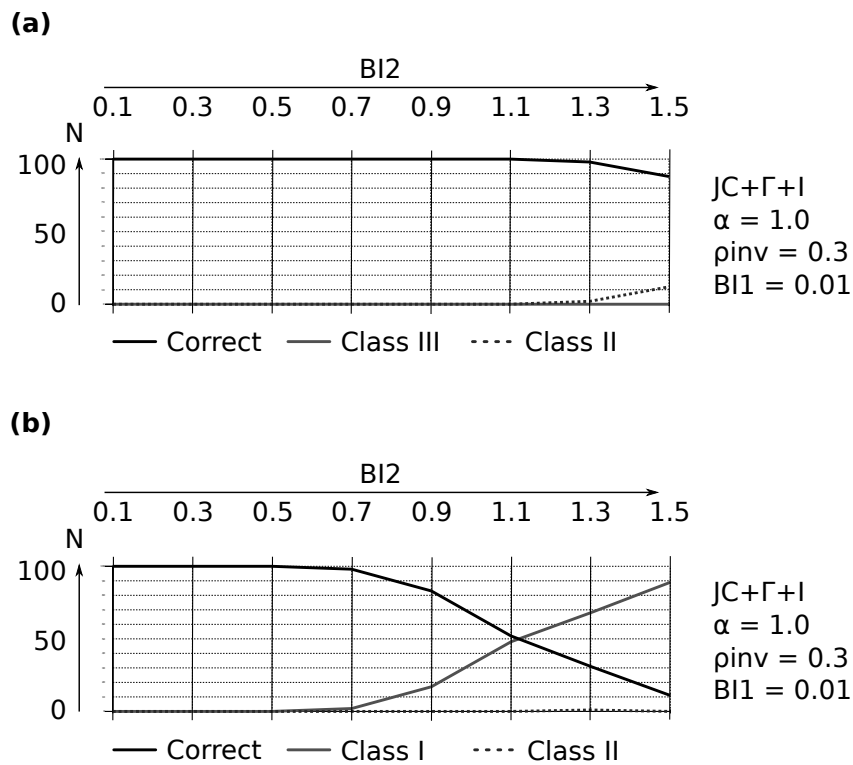


Figure 5.5: **Reconstruction success of Maximum Likelihood.** a) Topology A (Fig. 5.2a) and b) Topology B (Fig. 5.2b) under alignment lengths of 100,000 base positions if model assumptions are identical to the simulated parameters ( $\alpha = 1.0$ ;  $\rho_{inv} = 0.3$ ). Branch lengths differences increase from left to right by increasing  $BI2$  in discrete elongation steps (0.1-1.5) while  $BI1$  is kept constant (0.01). The y-axis depicts the reconstruction success of the 100 simulation repeats ( $N$ ).

differences in likelihood scores between wrong and correct topologies could not be observed in many cases if the ML parameters were equal.

## 5.4 Discussion

For alignment lengths in the range of 2,000-10,000, the reconstruction success was investigated for different fixed as well as estimated model parameters ( $\rho_{inv}$  and  $\alpha$ ). As expected, our results show that incorporating rate heterogeneity into analyses leads to an increased reconstruction success of ML, which has also been observed in previous studies (e.g. [59, 136, 139, 147, 150]).

If ASRV is considered either by the  $\Gamma$ -distribution (JC+ $\Gamma$ ) or the invariant sites model ( $\alpha = 100$ ;  $\rho_{inv} = 0.3$ ), ML performs better with the invariant sites proportion model than with the  $\Gamma$  distribution model. The inclusion of a mixed-distribution model (JC+ $\Gamma$ +I) again fits our data much better than a  $\Gamma$  distribution or invariant sites proportion model alone. By using a mixed-distribution model, ML recovered the correct topologies under a wide range of branch lengths. This supports the results of Sullivan et al. 1999 [59] as well as Anderson and Swofford [136] who showed that ML recovers topologies best if a  $\Gamma$ +I model is used.

For a combination of very short *BI1* and long *BI2*, ML performs poorly, even if a mixed-distribution model is used in the reconstruction (Fig. 5.4). Interestingly, we found that ML becomes more robust and efficient when analyses are performed with a lower than the correct value of  $\alpha$  even in ( $\Gamma$ +I) models than simulated, especially in cases of “hidden” (class I) and “classical long branch effects” (class III). It could be envisaged, that a lower value of  $\alpha$  leads to an overcompensation for multiple substitutions which in turn leads to the effect that convergent substitutions on long branches are not erroneously identified as homologies.

ML is not able to recover the true tree for the topology B with large length differences between short (*BI1*) and long branches (*BI2*) (Figure 5b), even if the correct model is specified. This class of topologies has not been investigated before and constitutes a new example for which ML efficiency is low even for long alignments. For the case of 4 taxa, the “inverse Felsenstein zone” is a well known example of reduced ML efficiency where alignment lengths of 100,000 bp are required for an 80% chance to recover the correct topology [60]. It can be expected, that our topology and setup yields an similarly “inefficient valley of death” to the one found for the “inverse Felsenstein zone” [60]. Since we can soon regularly expect data sets of the size of complete genomes, it would be interesting to investigate the extents of this valley, i.e. the necessary alignment length for which ML will reliably find the correct tree. For the topology A which corresponds to the classical Felsenstein zone (Fig. 5.2a), ML recovers the true tree efficiently. Our results for this topology are consistent with those found by Swofford et al. [60].

It is also interesting to note, that estimates of model parameters are very accurate for the  $\Gamma$ +I models used in the reconstruction. This high accuracy is found for all branch lengths and topologies even in those cases for which the reconstruction

success is low (see electronic supplementary file ES11). This excludes model misspecification as the source of phylogenetic inaccuracy in our study and is consistent with the observation that wrong topologies also occur if the true model parameters have been specified. Sullivan et al. [59] argued, that the number of taxa is important for the correct estimate of the shape parameter and the number of invariable sites, mainly due to stochastic errors in small samples. The observation that 11 taxa already allow us to find good estimates of the parameters in question could be explained by longer alignments in this study. Further, Sullivan et al. [158] demonstrated on 4-taxon trees that estimates of the  $\Gamma$  distribution can be strongly influenced by topologies which involve long internal branches. This correlation was not found in our analyses.

This study also confirms the expected correlation between the proportion of invariant sites and the shape parameter  $\alpha$  if parameters are estimated. If no invariant sites are assumed in the reconstruction, this model deficiency is partially compensated by a lower estimated value of the shape parameter, which results in an increased number of sites with low and very low substitution rates. Since this compensation is only partial and leads to an overestimation of substitution rates for a certain number of sites, the reconstruction success is lower compared with the application of a  $\Gamma+I$  model.

Our results show that the risk of obtaining a wrong topology increases even if ML is used in the reconstruction process and that this risk highly depends on branch length relations in the true topology that shall be reconstructed. Putting this together, we assume that Phillippe et al.'s [159] statement “probably most of the deep phylogenetic events are misplaced through artifacts” is not entirely wrong.

## 5.5 Additional Files

- **Electronic supplementary file ES9 — Detailed results of all ML tree reconstructions**
  - Complete overview of the reconstruction success of all ML analyses
  - **Format:** PDF
  - **Size:** 219.1 KB
  - **View:** PDF Viewer
  
- **Electronic supplementary file ES10 — Detailed results of investigated likelihood scores**
  - Investigated Likelihood values of all ML analyses
  - **Format:** PDF
  - **Size:** 313.9 KB
  - **View:** PDF Viewer

- 
- **Electronic supplementary file ES11 — Detailed results of model parameter estimates**
    - Investigated parameter estimates of all ML analyses
    - **Format:** TDF
    - **Size:** 114.1 KB
    - **View:** PDF Viewer
  
  - **Electronic supplementary file ES12 — Presentation of the results of chapter 5**
    - The presentation about the study of chapter 5 was given 2011 within the Systematics conference in Berlin, 2011
    - **Format:** PDF
    - **Size:** 2.3 MB
    - **View:** PDF Viewer



# Developed Software and help scripts (published/unpublished)

---

## Contents

---

<b>6.1 FASconCAT: Convenient handling of data matrices . . . . .</b>	<b>71</b>
6.1.1 Introduction . . . . .	71
6.1.2 Concatenation of data . . . . .	72
6.1.3 Data conversion . . . . .	74
6.1.4 Discussion . . . . .	74
<b>6.2 ALICUT . . . . .</b>	<b>76</b>
<b>6.3 BHoEMe . . . . .</b>	<b>76</b>
<b>6.4 SusEX . . . . .</b>	<b>77</b>
<b>6.5 ESTa . . . . .</b>	<b>77</b>
<b>6.6 TaxEd . . . . .</b>	<b>77</b>
<b>6.7 LoBraTe . . . . .</b>	<b>77</b>
<b>6.8 RAxTAX . . . . .</b>	<b>77</b>
<b>6.9 SecSITE . . . . .</b>	<b>78</b>
<b>6.10 SPIPES . . . . .</b>	<b>78</b>
<b>6.11 Additional Files . . . . .</b>	<b>78</b>

---

## 6.1 FASconCAT: Convenient handling of data matrices

### 6.1.1 Introduction

Today, data concatenation into supermatrices is a frequently used task in phylogenetic approaches. Data concatenation has been employed in rRNA analyses [81, 160], in analyses using 'mixed' nucleotide alignments combining rRNA sequences like 18S and 28S as well as protein coding genes [37, 38, 161], in analyses based on nucleotide and amino acid alignments or in phylogenomic studies [106, 162, 163]. The handling of different required file formats is often extensive and time consuming and different scripts or programs are often necessary. Most common formats are FASTA [164], NEXUS [165], CLUSTAL [166] and PHYLIP [167]. To consider structure information of unpaired (loop) and paired (stem) regions using e.g. ribosomal RNA genes, most programs like RNAsalsa [32], MrBayes [120], PHASE [168] and RAxML [71]

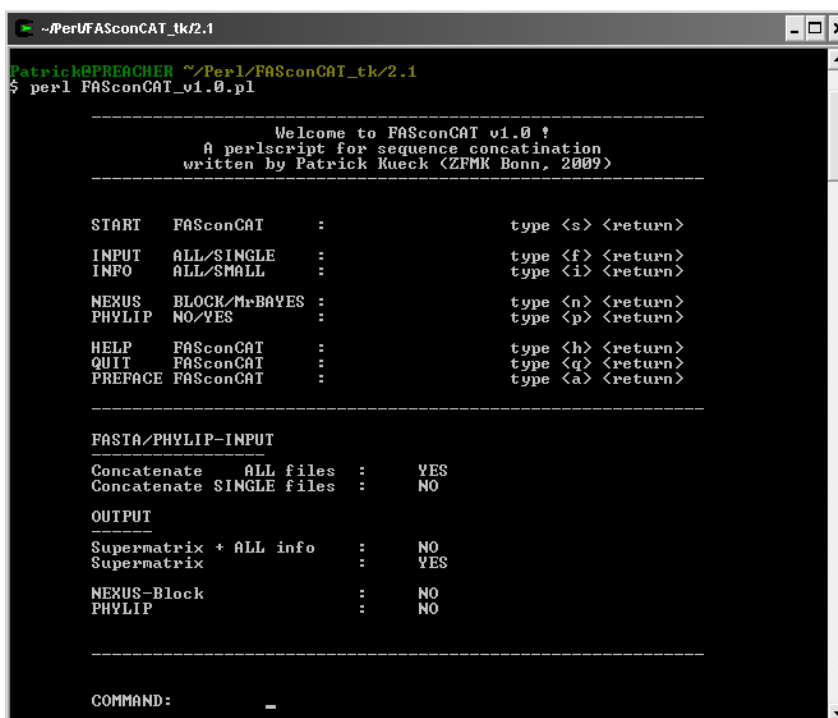
accept structure information in 'dot-bracket' format. Recent concatenation tools like CONCATENATOR [169] can only concatenate and convert sequence data from FASTA to NEXUS and vice versa and are unable to handle additional structure information. Moreover, concatenation is mostly restricted to a limited number of gene alignments. With FASconCAT, we provide a new software tool for easy and fast data handling.

FASconCAT is implemented in Perl and runs on Windows PCs, Mac OS and Linux operating systems. It can be used via command line or by terminal menu options. The main menu of FASconCAT is subdivided into two parts, separated by a dashed line (Fig. 6.1). The upper component constitutes of a list of all possible options and their associated commands for adjustment. The lower part shows the actual parameter settings of FASconCAT. All default parameters can be optionally changed, and the new setting configuration will be displayed in the lower part of the menu.

The software is designed to concatenate different data formats of nucleotide and amino acid alignments (sequence or artificially, e.g. RY coded) as well as "dot-bracket" structure information of identical taxa into one supermatrix file. It can also be used as a simple data converter if just one file is provided. FASconCAT can handle FASTA, CLUSTAL and PHYLIP input files. No unique input format is required. Sequences must have equal length within each file. FASTA, NEXUS and PHYLIP can be chosen either as multiple or single output format. The output files can be directly implemented into software like PAUP\* [170], MrBayes [120] or RAxML [71]. FASconCAT optionally creates NEXUS files with command blocks applicable in MrBayes [120]. Among other things this option is very convenient for partitioned or mixed DNA/RNA analyses. Furthermore, it provides information about supermatrix partitions (single ranges) which can be used in partitioned analyses.

### 6.1.2 Concatenation of data

Sequence data, with or without structure information, are concatenated either by taking all appropriate files in the folder or by user specification. With FASconCAT, it is also possible to concatenate amino acid and nucleotide alignments into one supermatrix. Missing taxon sequences in single files are considered and replaced either by 'N' (nucleotide sequences), 'X' (amino acid sequences) or by '.' (dots, structure strings in 'dot-bracket' format), dependent on their associated data level. FASconCAT can read sequences in interleaved and non-interleaved format. The number of files for concatenation is not limited. The computation time rather depends on the computer hardware and the random access memory (RAM). For example, the concatenation of ten files comprising 108 taxa with a length of 1,000 bp each requires between 0.5 (default option) and 3.4 seconds ('NEXUS' option) on a normal desktop computer (see Appendix D (manual) for more information). Creating NEXUS files is the most time consuming option. Every user can individually choose favoured options to optimize time performance. If no options are specified, FASconCAT runs



```
--PerlFASconCAT_tk/2.1
Patrick@PREACHER ~/Perl/FASconCAT_tk/2.1
$ perl FASconCAT_v1.0.pl

-----
                Welcome to FASconCAT v1.0 !
                A perlscript for sequence concatenation
                written by Patrick Kueck (ZFMK Bonn, 2009)
-----

START  FASconCAT      :                type <s> <return>
INPUT  ALL/SINGLE     :                type <f> <return>
INFO   ALL/SMALL     :                type <i> <return>
NEXUS  BLOCK/MsBAYES :                type <n> <return>
PHYLIP NO/YES       :                type <p> <return>
HELP   FASconCAT     :                type <h> <return>
QUIT   FASconCAT     :                type <q> <return>
PREFACE FASconCAT    :                type <a> <return>
-----

FASTA/PHYLIP-INPUT
Concatenate ALL files : YES
Concatenate SINGLE files : NO

OUTPUT
Supermatrix + ALL info : NO
Supermatrix              : YES
NEXUS-Block              : NO
PHYLIP                   : NO
-----

COMMAND: -
```

Figure 6.1: **Main menu of FASconCAT.** The menu is subdivided into a command block (upper half) and a setting block (lower half). Users can specify their setting by using single commands via menu options or by typing multiple commands directly via the start command line of FASconCAT.



under default which is the most time saving setting.

FASconCAT delivers useful accompanying information about the supermatrix and all single input files. As default, information is given for the partitions of the concatenated data set (fragment range) and the number of concatenated sequences per taxon. Additional information is provided by specifying several options, for example the number of sequence characters, sequence-type, number of gaps, a list of unpaired (loop) and paired (stem) positions (see Appendix D (manual) for detailed instructions). A schematic overview is given in Figure 6.2.

#### 6.1.2.1 Default options

With standard options, FASconCAT takes all available input files (CLUSTAL, FASTA, PHYLIP) within the script placed folder and concatenates them into a supermatrix in FASTA format. Provided structure sequences in 'dot-bracket' format (one per file) are concatenated as well. Default information are accessorially provided (see above).

#### 6.1.2.2 Additional options: -f, -i, -n and -p

With option -f, individual input files can be defined by the user. Additional information on the supermatrix and the input files, e.g. base composition of nucleotide sequences or the amount of gaps, can be activated by option -i. With -n, NEXUS files are generated that can be directly used in PAUP\* [170] or MrBayes [120]. With typing -n -n, a complete set up for MrBayes is created. It can be easily modified as favoured by the user. With option -p, FASconCAT additionally provides an output in PHYLIP format, either with non-interleaved sequences and restricted taxon names up to ten signs (-p) or relaxed, with non-interleaved sequences and no restriction for taxon names (-p -p).

An example for FASconCAT usage could be: The user has three sequence alignment files in the same folder where FASconCAT is located, one in FASTA, the second in PHYLIP and the third in CLUSTAL format. The user wants to concatenate all alignments into a supermatrix in FASTA format and obtain all possible information via command line in a terminal on a LINUX system. FASconCAT has to be started as follows:

```
perl FASconCAT.pl -i -s <enter>
```

#### 6.1.3 Data conversion

Sequence formats can be simply converted by running FASconCAT just with one input file.

#### 6.1.4 Discussion

FASconCAT is a new, convenient tool for concatenation of sequence files. FASconCAT is easy to use and not limited in number of input files or input sequences.

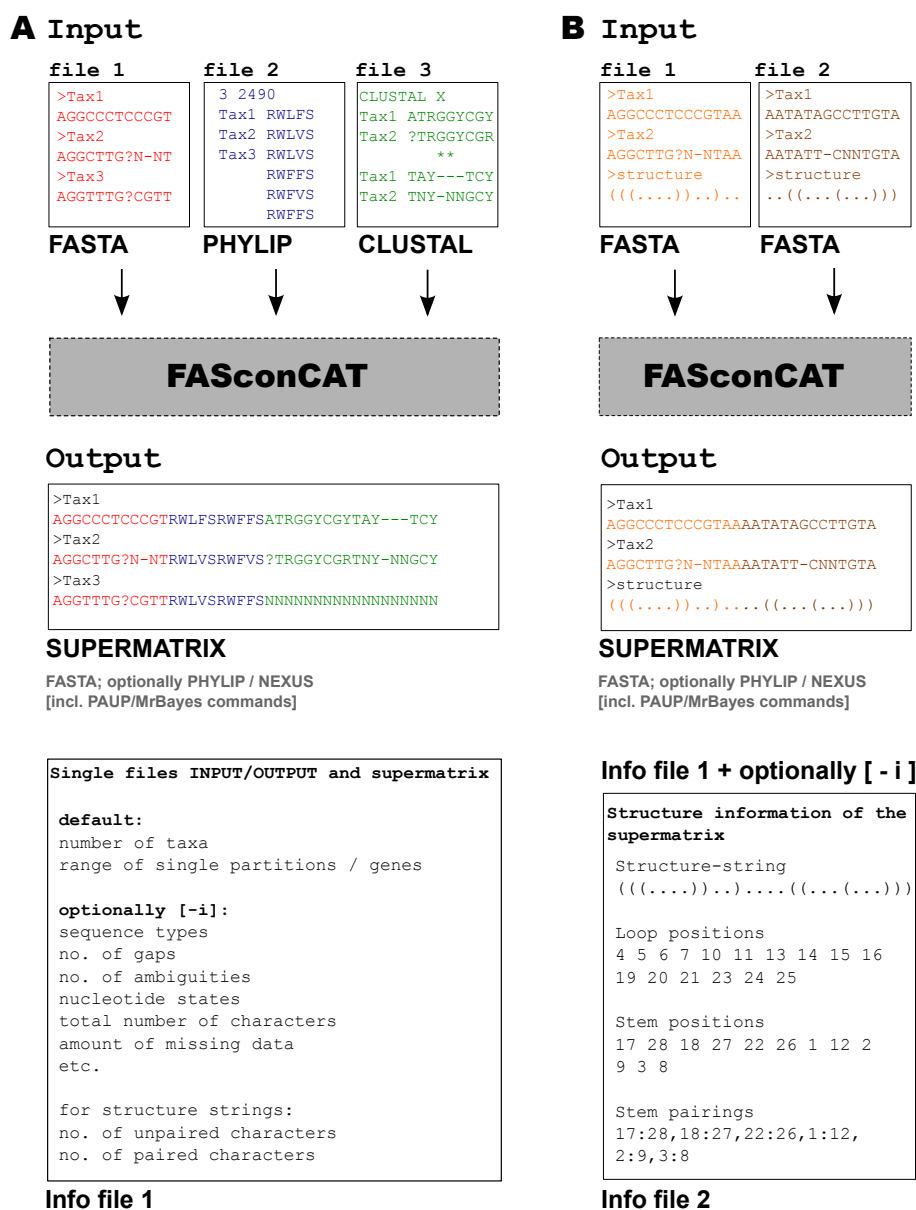


Figure 6.2: **Schematic overview of FASconCAT.** A: Three input files with different format (FASTA; PHYLIP, NEXUS), a nucleotide sequence alignment, an amino acid alignment and a nucleotide alignment with the third position RY recoded, are concatenated into a supermatrix (FASTA format, default). Additionally, an information file (Info file 1) is provided containing a list of concatenated sequences (taxa) and range information of single genes in the supermatrix (default). Optionally, additional information can be obtained by specific commands. B: Two input files, nucleotide alignments with a structure string are concatenated into a supermatrix. Specifying the `-i` option, additional information about the percentage of unpaired (loop) and paired (stem) positions, is provided (Info file 1). A second information file is obtained, containing the concatenated structure string, the position of loop and stem positions and related stem pairings (Info file 2).

Running on UNIX and Windows operating systems, the software reads several input formats, considers structure information, provides several output formats and optionally complete set up blocks at once, e.g. for analyses in MrBayes. It facilitates data handling, it is time saving in generating data matrices and in converting file formats and delivers many useful additional information about the input sequences. Detailed information and instructions are provided in the manual of FASconCAT (Appendix D). The manual (Appendix D) also includes some tests about computation time of FASconCAT on a normal desktop computer. Help is provided for every option. FASconCAT is simple to use and freely available from <http://fasconcat.zfmk.de> or upon request from the corresponding author. The FASconCAT script and a corresponding presentation from a FASconCAT lecture of a regular course within the molecular biology department of the ZFMK are also added as electronic supplementary files ES13 and ES14.

## 6.2 ALICUT

ALICUT is an additional tool to the ALISCORE software which removes all ALISCORE identified random sequence similarity positions (rss) in multiple sequence alignments of given FASTA file(s) that are listed in the FASTA file corresponding "List" outputfile(s) of ALISCORE. Additionally, ALICUT generates a listfile including percentages of remaining positions of each input file. If structure sequences are implemented, ALICUT can automatically remain randomised stem positions or replace their corresponding character position by dots if characters are identified as non-randomised (see Appendix E (manual) for more information). Actual version: ALICUT v1.0 [111]. Processes of ALICUT were used inter alia in the studies of Reumont et al. [81], Letsch et al. [33, 160], Meusemann et al. [107], and Kück et al. [35, 123]. The script and a presentation of ALICUT are deposited as electronic supplement files ES16 and ES17.

## 6.3 BHoEMe

BHoEMe ("bootstrap homoplasy excess method") is an automatized approach of a method which was first described by Seehausen [171]. The script is written in Perl and enables an identification of phylogenomic inconsistency of a hybrid taxon which introduces homoplasies in parts of a phylogenetic tree constructed from AFLP genotypes. By reconstructing a tree with and without the putative hybrid taxon, and recording the difference that its inclusion makes to the bootstrap support for each node, the homoplasy effect of a putative hybrid taxon can be investigated for each node and compared to a distribution of non-hybrid effects generated by adding and removing in turn each other taxa in the tree. Actual version: BHoEMe v0.1 beta.

## 6.4 SusEX

SusEX (“subsequence extractor”) finds subsequences in FASTA alignments specified by user predefined start and end strings. Identified substrings are printed out in single FASTA files. Actual version: SusEX v0.1 beta [172].

## 6.5 ESTa

ESTa (“EST adjustment”) downloads the actual EST-summary list for all taxa with more than 1.000 EST database entries from the NCBI data base and compares them with entries from an earlier downloaded listfile. New EST entries are given out as separate textfile and the new listfile provides the basis for the next EST adjustment from NCBI. Actual version: ESTa v0.1 beta [173]. Processes of ESTa was used in the study of Meusemann et al. [107]. The ESTa script is deposited as electronic supplement file ES18. A short documentation of ESTa is given in Appendix F.

## 6.6 TaxEd

TaxEd (“taxon editor”) converts all taxa information of downloaded NCBI sequences into a more simplified and standardised information format. Actual version: TaxEd v0.1 beta [174]. Processes of ESTa was used in the study of Meusemann et al. [107].

## 6.7 LoBraTe

LoBraTe (“long branch test”, [175]) handles simulations of different long branch effects (LBA) under different Maximum Likelihood model and parameter settings to identify reasons of inconsistencies of the Maximum Likelihood approach relative to LBA. Beside the stepwise increase of terminal and internal branches of given asymmetric and symmetric topologies and the simulation of sequence alignments based on each topologie with INDELible, the pipeline starts Maximum Likelihood analysis under PhyML, identifies long branch topologies of classes I, II and III, numbers of symmetric splits for the right and wrong relationships of each calculated topology, executes special likelihood ratio and chi square tests and performs a parametric bootstrap analyses. All results are given out as vector grafik plots. LoBraTe was used for the simulation analyses of chapter 5 and 4. Actually, a mathematical algorithm is tested with LoBraTe for its efficiency to detect long branch attraction between strongly derived taxa. A floatchart of the LoBraTe process pipeline and its output plots is shown in Appendix B.

## 6.8 RAxTAX

RAxTAX [176] is designed to execute a full phylogenetic analyses starting from raw sequence data and ending by a full Maximum Likelihood analysis. Included

steps are execution of a multiple sequence alignment (selectively MAFFT-L-ins-i, MAFFT-G-ins-i, MAFFT-E-ins-i, T-COFFEE, DIALIGN-TX or MUSCLE), alignment refinement, alignment masking (ALISCORE), removing of ALISCORE identified random sequence similarities (ALICUT), gene concatenation (FASconCAT) and Maximum Likelihood tree reconstruction (RAxML). RAxTAX is also able to read in a user specified textfile in which single listed sequences are excluded after each reconstruction step to identify sequences which are potential “problematic” in reference to tree reconstruction, e.g. long branch effects. Processes of RAxTAX was used inter alia by the study of Kück et al. [123]. Flowcharts about single process steps, starting commands, and an overview of input and output file formats are given in Appendix C.

## 6.9 SecSITE

SecSITE (“secondary structure impact test”, [177]) is a pipeline to identify the impact of rRNA secondary structure consideration in alignment and tree reconstruction. SecSITE can be used for simulated and real data sets. Processes of SecSITE was used by Letsch et al. [33].

## 6.10 SPIPES

SPIPES (“small pipe”, [178]) is a pipeline to identify the effect of alignment masking approaches in regard to alignment structure improvement. SPIPES executes from raw data four different alignment approaches (MAFFT-L-ins-i, MUSCLE, T-COFFEE, ClustalW), afterwards it masks each alignment either with GBLOCKS or ALISCORE, cuts out randomised sequence sections, concatenates each gene relative to alignment method and masking approach and finally executes a Maximum Likelihood method (RAxML) of each supermatrix. Summarised info are sampled in extra textfiles. SPIPES was used for the processes publicated by Kück et al. [35].

## 6.11 Additional Files

- **Electronic supplementary file ES13 — FASconCAT v1.0**
  - Sequence Concatenation Software
  - **Format:** PL
  - **Size:** 46.9 KB
  - **View:** Texteditor
- **Electronic supplementary file ES14 — Presentation of the Software FASconCAT v1.0**

- 
- The presentation was used to introduce FASconCAT within a regular course of the molecular biology department of the ZFMK (2010), specified on phylogenetic methods, algorithms and analyses
  - **Format:** PDF
  - **Size:** 625.6 KB
  - **View:** PDF Viewer
  - **Electronic supplementary file ES15 — Publication (Kück & Meusemann (2010) [112])**
    - Publication of the FASconCAT sequence concatenation software described in chapter 3 and appendix D
    - **Format:** PDF
    - **Size:** 619.4 KB
    - **View:** PDF Viewer
  - **Electronic supplementary file ES16 — ALICUT v1.0.pl [111]**
    - An ALISCORE complementation script to remove ALISCORE identified randomized sequence sections.
    - **Format:** PL
    - **Size:** 46.9 KB
    - **View:** Texteditor
  - **Electronic supplementary file ES17 — Presentation of ALICUT v1.0**
    - The presentation was used to introduce ALICUT within a regular course of the molecular biology department of the ZFMK (2010), specified on phylogenetic methods, algorithms and analyses
    - **Format:** PDF
    - **Size:** 180.4 KB
    - **View:** PDF Viewer
  - **Electronic supplementary file ES18 — ESTa v0.1.beta**
    - A script to download database entries from the NCBI data base
    - **Format:** PL
    - **Size:** 4 KB
    - **View:** Texteditor



# General Discussion

## Contents

<b>7.1</b>	<b>The Effect of Alignment Masking on Phylogenetic Analyses</b>	<b>81</b>
<b>7.2</b>	<b>The Effect of Long Branches and chosen Model Parameters on Maximum Likelihood Reconstructions . . . . .</b>	<b>82</b>
<b>7.3</b>	<b>Perspectives . . . . .</b>	<b>83</b>

## 7.1 The Effect of Alignment Masking on Phylogenetic Analyses

It has been shown in chapter 2, "Masking of randomness in sequence alignments can be improved and leads to better resolved trees", that alignment masking is a powerful approach to improve signal-to-noise ratio in multiple sequence alignments before tree reconstruction. Parametric and non-parametric methods have successfully identified incorrectly aligned sequence sections. The identification and removal of these sections make alignment accuracy less dependent on chosen alignment algorithms and lead to improved node resolution and bootstrap support values in tree reconstructions. The improvement of node resolution and bootstrap support values was especially noticeable in reconstructions of deep node relationships. Therefore, the selection of more reliable alignment sections through alignment masking reduces the sensitivity of substitution models, which was additionally demonstrated in the analysis in chapter 3, "Improved phylogenetic analyses corroborate a plausible position of *Martialis heureka* in the ant tree of life". Given the robust performance of alignment masking on alignment accuracy, it should routinely be used to improve tree reconstructions.

The parametric masking approach of ALISCORE is, in opposite to the non-parametric GBLOCKS approach, independent of *a priori* rating of sequence variation and seems to be more capable to handle automatically different substitution patterns and heterogeneous base composition. Furthermore, the ALISCORE algorithm does not overestimate the amount of randomized aligned sequence sections [34] which seems especially common through the exclusion of all gap containing alignment sites under the conservative GBLOCKS default setting. The ALISCORE algorithm can be easily extended to more complex likelihood based models of sequence evolution which opens the possibility of further improvements.



Despite all advantages of masking methods, chapter 4, "AliGROOVE: a new tool to visualize the extent of sequence similarity and alignment ambiguity in multiple alignments", describes also the inability of recent masking algorithms to detect strong heterogeneous sequence divergence in sequence alignments. This is an important drawback of recent masking algorithms, because negative effects of undetected heterogeneous sequence divergence like effects of long branch attraction are still the most problematic kinds of biases in recent phylogenies (e.g. [81, 150, 151, 179–183]). As shown in chapter 5, "Long branch effects distort Maximum Likelihood phylogenies in simulations despite selection of the correct model", even Maximum Likelihood methods can fail to resolve the correct topology if signal in the data is heterogeneous. As shown in chapter 4, the sliding window approach as it is used in ALISCOPE can be used to identify single taxa or subsets of taxa which show predominantly random sequence similarity in comparison to other taxa. Removal of these taxa can potentially increase the tree-likeness of the data as well and thus help to improve the reliability of tree reconstructions. The simulation results and the analyses of empirical data of chapter 4 show that the sliding window approach has some predictive power. This characteristic is considered as a major advantage over all character based masking approaches in molecular phylogenetics. It also offers the possibility of excluding taxa based on a formal argument in comparison with excluding taxa based exclusively on the subjective evaluation of branch lengths. However, the results of chapter 4 and 2 indicate also, that the scoring scheme of the ALISCOPE algorithm, which is based either on simple match/mismatch scores for nucleotide sequences or on the BLOSUM62 matrix to score aminoacid matches/mismatches, is a relatively simple scoring regime. It turned out to be efficient in simulations and empirical data [34, 35, 105–109], but it will be a matter of further analyses if an extension of the sliding window approach to more realistic likelihood models of change and Monte Carlo resampling will further improve the performance of ALISCOPE as well as AliGROOVE.

Anyway, through the use of large, phylogenomic data sets, which will be common in near future, the associated danger of decreased alignment accuracy and phylogenetic inference makes it important to establish a reliable alignment masking approach to cope with systematic errors in multiple sequence alignments. The analyses of chapter 2 and 3 demonstrate that the sliding window approach will be a useful profiling tool to guide alignment masking.

## 7.2 The Effect of Long Branches and chosen Model Parameters on Maximum Likelihood Reconstructions

The simulation results of chapter 5 show that the risk of obtaining a wrong topology increases with branch length differences, even if Maximum Likelihood is used in the reconstruction process with the correct simulation model. The study demonstrates that this risk highly depends on branch length relations in the true topology that shall be reconstructed. Long branches lead to substitutional saturation along

sequences, which increases the risk that the loss of conserved signal reduces the chances for correct tree reconstruction. Maximum Likelihood was not able to recover the correct tree for a topology with large length differences between short and long internal branches, even if the correct model was specified. Wrong topologies did not even disappear when sequence alignment length rises to 100,000 base positions. Altogether, the study seems to confirm the assumption of Phillipe and Laurent (1998) [159] that “probably most of deep phylogenetic events are misplaced through artifacts”. However, the study of chapter 5 demonstrates also, that Maximum Likelihood is very robust to fluctuations of conserved signal if rate heterogeneity is incorporated into the analysis. The new observations agree with previously performed studies (e.g. [59, 136, 139, 147, 150]). If among-site rate variation is considered by a mixed-distribution model, Maximum Likelihood recovered the correct topology under a wide range of branch lengths in accordance with older publications [58, 59, 136, 149, 150]. The study of chapter 5 is consistent with the expected positive correlation between the proportion of invariant sites and the shape parameter  $\alpha$  if parameters are estimated.

Recently published studies relied on the exclusive application of the restricted  $\Gamma$ -model [108, 130, 151–155]). The assumption that the estimation of ASRV is much more predictable if the  $\Gamma$ -model is used as single estimator could not be confirmed by the study of chapter 5. If no invariant sites are assumed for the reconstruction, this model deficiency is partially compensated by a lower estimated value for the shape parameter, which implies an increased number of sites with low and very low substitution rates. Since this compensation is only partial and leads to an overestimation of substitution rates for a certain number of sites, the reconstruction success is lower compared with the application of a  $\Gamma+I$  model. If among-site rate variation is considered either by the  $\Gamma$ -distribution or the invariant sites model, ML performs even better with the invariant sites proportion model than with the  $\Gamma$ -distribution model.

The assumption that estimates of the  $\Gamma$ -distribution can be strongly influenced by topologies which involve long internal branches [158] was not confirmed in the study of chapter 5. Estimates of model parameters are found to be very accurate for the  $\Gamma+I$  models used in the tree reconstructions, even in those cases for which the reconstruction success is low. This excludes model misspecification as the source of phylogenetic inaccuracy in this study and is consistent with the observation that wrong topologies also occur if correct model assumptions have been specified. What is lacking to the model are additional information about the “true” branch lengths.

Anyway, for a combination of very short and long branches, ML performs poorly, even if a mixed-distribution model is used in the reconstruction.

## 7.3 Perspectives

Yet, the focal point in molecular phylogenetics has been given rather on the extraction of data quantity than on the production of data quality. A high accumulation

of sequence data, like phylogenomic data, is not capable of suppressing systematic biases. Indeed quite the opposite is true. The risk that systematic biases negatively influence phylogenetic analyses increases as more data become available. The influence of systematic biases increases again if evolutionary substitution models are not chosen suitably for underlying data. Substitution models with too many parameters can lead to an overestimate of the evolutionary history of underlying data while the assumption of undercharged parameters can result into the opposite. In both cases, the chance of identifying an incorrect topology continues to increase. Evolutionary substitution models which most fit with underlying data are often hard to detect. Even the most suitable model represents only a rough approximation to the actual evolutionary history of sequence data. Therefore, it is not surprising that current reconstruction methods are not able to perfectly differentiate between phylogenetic informative and non-informative signal. Inconsistencies of reconstruction methods on finite data sets due to incorrect model assumptions can be seen as a major drawback of molecular phylogenetic analyses [184]. Another problem of currently tree reconstruction methods is the dependency on arbitrary tree and model parameter search heuristics which often lead to suboptimal trees instead of the correct topology.

For that purpose, a reliable identification of erroneously placed taxa after the tree reconstruction process would be desirable. Whether the sliding window approach can be sensitive enough to safely fill this gap has to be shown in further analyses. Another attempt to identify incorrect taxon placements in topologies could be the identification of random similarity between closely resolved taxa via inferred branch length distances. The possibility of random sequence similarities increases with increased branch length distances. In the most extreme case, the exponent  $\mu * t$  (substitution rate multiplied by divergence time) of evolutionary models goes into infinity. In this case, the probability that similar character states are due to common ancestral states is, e.g. under Jukes Cantor, decreased to 1/4. This means that sequence similarity of corresponding taxa is indistinguishable from random. A comparison of likelihood scores obtained for similar character states under two conditions: i) with the inclusion of investigated branch length distances and ii) under infinite branch length assumptions, could provide information whether sequence similarity between taxa depends potentially on common ancestral states or on random processes. First test analyses on simulated data have been performed with LoBraTe, but further investigations are necessary to give a clear statement.

However, even if erroneously placed taxa could be reliably identified after tree reconstruction, it is left unclear if a definite phylogenetic assignment of these taxa can be made certain for ever. With the use of recent evolutionary substitution models, systematic bias will always have a negatively influence on tree reconstruction methods, especially if taxa are highly derived.

While most of the theoretical research in molecular systematics concentrates on the accuracy and improvement of phylogenetic reconstruction methods, the influence of alignment accuracy on tree reconstructions has been paid comparatively little attention. This is astonishing if one assumes that alignment accuracy can have a strong influence on tree reconstruction. Therefore, a better understanding and

further development of alignment and masking heuristics should be rather essential tasks in phylogenetics for the future. Especially as millions of euros are invested each year in Europe for phylogenetic research.

On the basis of high quality alignments, the use of invariants, hadamard conjugations, or split techniques could be good alternatives to recently used tree reconstruction methods like Maximum Likelihood or Bayesian approaches. Unfortunately, invariants and hadamard-conjugations are currently computationally not feasible for larger data sets (e.g. hadamard-conjugations can only handle data sets with less than 16 taxa). Search heuristics like the quasi-biclique techniques could be a good starting point to make invariants and hadamard-conjugations less complex and time consuming. Surely, the most important advantage of invariants to hadamard-conjugations is certainly the resignation of evolutionary models. Another target for phylogenetic analyses offer split techniques in which an improved scoring scheme for split detection could lead to robust and reliable split-topologies.



# Bibliography

- [1] Ogden TH, Rosenberg M (2006) Multiple sequence alignment accuracy and phylogenetic inference. *Syst Biol* 55: 314-328. (Cited on pages 1, 2, 6 and 11.)
- [2] Morrison DA (2006) Multiple sequence alignments for phylogenetic purposes. *Aust Syst Bot* 19: 479-539. (Cited on pages 1, 2, 3, 4 and 11.)
- [3] Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, et al. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res* 25: 3389-3402. (Cited on page 1.)
- [4] Kent WJ (2002) BLAT-The BLAST-like alignment tool. *Genome Res* 12: 656-664. (Cited on page 1.)
- [5] Tatusova TA, Madden TL (1999) BLAST 2 Sequences, a new tool for comparing protein and nucleotide sequences. *FEMS Microbiol Lett* 174: 247-250. (Cited on page 1.)
- [6] DePinna MCC (1991) Concepts and tests of homology in the cladistic paradigm. *Cladistics* 7: 367-394. (Cited on page 1.)
- [7] Phillips A, Janies D, Wheeler W (2000) Multiple sequence alignment in phylogenetic analysis. *Mol Phylogenet Evol* 16: 317-330. (Cited on page 1.)
- [8] Reeck GR, de Haën C, Teller DC, Doolittle RF, Fitch WM (1987) "Homology" in proteins and nucleic acids: a terminology muddle and a way out of it. *Cell* 50: 667. (Cited on page 2.)
- [9] Wegnez M (1987) Letter to the editor. *Cell* 51: 516. (Cited on page 2.)
- [10] Patterson C (1988) Homology in classical and molecular biology. *Mol Biol Evol* 5: 603-625. (Cited on page 2.)
- [11] Rosenberg MS (2009) *Sequence Alignment. Methods, Models, Concepts, and strategies.* University of California Press. (Cited on pages 2, 3 and 4.)
- [12] Kjer KM (1995) Use of rRNA secondary structure in phylogenetic studies to identify homologous positions: An example of alignment and data presentation from the frogs. *Mol Phylogenet Evol* 4: 314-330. (Cited on page 2.)
- [13] Hwang UW, Kim W, Tautz D, Friedrich M (1998) Molecular phylogenetics at the Felsenstein Zone: Approaching the Strepsiptera problem using 5.8S and 28S rRNA sequences. *Mol Phylogenet Evol* 9: 470-480. (Cited on page 2.)

- 
- [14] Cammarano P, Creti R, Sanangelantoni AM, Palm P (1999) The Archea monophyly issue: A phylogeny of translational elongation factor G(2) sequences inferred from an optimized selection of alignment positions. *J Mol Evol* 49: 524-537. (Cited on page 2.)
- [15] Ogden TH, Whiting M (2003) The problem with "the Palaeoptera problem": Sense and sensitivity. *Cladistics* 19: 432-442. (Cited on page 2.)
- [16] Notredame C (2002) Recent progresses in multiple sequence alignment: a survey. *Pharmacogenomics* 3: 1-14. (Cited on pages 3, 4 and 11.)
- [17] Hogeweg P, Hesper B (1984) The alignment of sets of sequences and the construction of phylogenetic trees. An integrated method. *J Mol Evol* 20: 175-186. (Cited on page 3.)
- [18] Feng DF, Doolittle RF (1987) Progressive sequence alignment as a prerequisite to correct phylogenetic trees. *J Mol Evol* 25: 351-360. (Cited on page 3.)
- [19] Taylor WR (1988) A flexible method to align large numbers of biological sequences. *J Mol Evol* 28: 161-169. (Cited on page 3.)
- [20] Notredame C, Higgins DG, Heringa J (2000) T-COFFEE: A novel method for fast and accurate multiple sequence alignment. *J Mol Biol* 302: 205-217. (Cited on pages 3, 4, 12 and 13.)
- [21] Thompson JD, Higgins DG, Gibson TJ (1994) CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res* 22: 4673-4680. (Cited on pages 3, 12, 13 and 47.)
- [22] Eddy SR (1998) Profile Hidden Markov Models. *Bioinformatics* 14: 755-763. (Cited on page 3.)
- [23] Edgar RC (2004) MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res* 32: 1792-1797. (Cited on pages 3, 12 and 13.)
- [24] Katoh K, Kuma Ki, Hiroyuki T, Miyata T (2005) MAFFT version 5: Improvement in accuracy of multiple sequence alignment. *Nucleic Acids Res* 33: 511-518. (Cited on pages 3, 12 and 13.)
- [25] Nuin PAS, Wang Z, Tellier ERM (2006) The accuracy of several multiple sequence alignment programs for proteins. *BMC Bioinformatics* 7: 471. (Cited on pages 3, 11, 28 and 36.)
- [26] Golubchik T, Wise M, Eastal S, Jermin L (2007) Mind the gaps: evidence of bias in estimates of multiple sequence alignments. *Mol Biol Evol* 24: 2433-2442. (Cited on pages 3, 28 and 36.)

- [27] Huang X, Miller W (1991) A time-efficient linear-space local similarity algorithm. *Adv Appl Math* 12: 337-357. (Cited on page 3.)
- [28] Morgenstern B (2002) A simple and space-efficient fragment-chaining algorithm for alignment of DNA and protein sequences. *Appl Math Lett* 15: 11-16. (Cited on page 3.)
- [29] Subramanian AR, Weyer-Menkhoff J, Kaufmann M, Morgenstern B (2005) DIALIGN-T: An improved algorithm for segment-based multiple sequence alignment. *BMC Bioinformatics* 6: 13. (Cited on page 3.)
- [30] Morgenstern B (2009) *Sequence Alignment. Methods, Models, Concepts, and strategies.* University of California Press. (Cited on pages 3 and 4.)
- [31] Tabei Y, Kiryu H, Asai K (2008) A fast structural multiple alignment method for long RNA sequences. *BMC Bioinformatics* 9: 33-49. (Cited on page 4.)
- [32] Stocsits RR, Letsch HO, Hertel J, Misof B, Stadler PF (2009) Accurate and efficient reconstruction of deep phylogenies from structured RNAs. *Nucleic Acids Res* 37: 6184-6193. (Cited on pages 4 and 71.)
- [33] Letsch HO, Kück P, Stocsits RR, Misof B (2010) The impact of rRNA secondary structure consideration in alignment and tree reconstruction: simulated data and a case study on the phylogeny of hexapods. *Mol Biol Evol* 27: 2507-2521. (Cited on pages 4, 76 and 78.)
- [34] Misof B, Misof K (2009) A Monte Carlo approach successfully identifies randomness in multiple sequence alignments: a more objective means of data exclusion. *Syst Biol* 58: 21-34. (Cited on pages 4, 5, 6, 11, 12, 14, 15, 28, 38, 43, 45, 54, 81, 82 and 127.)
- [35] Kück P, Meusemann K, Dambach J, Thormann B, von Reumont B, et al. (2010) Parametric and non-parametric masking of randomness in sequence alignments can be improved and leads to better resolved trees. *Front Zool* 7: 10. (Cited on pages 4, 5, 24, 28, 43, 45, 76, 78, 82, 127 and 137.)
- [36] Castresana J (2000) Selection of conserved blocks from multiple alignments for their use in phylogenetic analysis. *Mol Biol Evol* 17: 540-552. (Cited on pages 5, 11, 12, 28 and 43.)
- [37] Brady S, Schultz T, Fisher B, Ward P (2006) Evaluating alternative hypotheses for the early evolution and diversification of ants. *Proc Natl Acad Sci U S A* 103: 18172-18177. (Cited on pages 5, 27, 28, 36, 37 and 71.)
- [38] Moreau C, Bell C, Vila R, Archibald S, Pierce N (2006) Phylogeny of the ants: diversification in the age of angiosperms. *Science* 312: 101-104. (Cited on pages 5, 27, 36, 37 and 71.)



- [39] Ouellette G, Fisher B, Girman D (2006) Molecular systematics of basal subfamilies of ants using 28S rRNA (Hymenoptera: Formicidae). *Mol Phylogenet Evol* 40: 359-369. (Cited on pages 5, 27, 36 and 37.)
- [40] Rabeling C, Brown J, Verhaagh M (2010) Newly discovered sister lineage sheds light on early ant evolution. *Proc Natl Acad Sci U S A* 105: 14913-14917. (Cited on pages 5, 27, 28, 29, 30, 36 and 37.)
- [41] Moreau C (2009) Inferring ant evolution in the age of molecular data (Hymenoptera: Formicidae). *Myrmecol News* 12: 201-210. (Cited on pages 5, 36 and 37.)
- [42] Felsenstein J (1978) Cases in which parsimony or compatibility methods will be positively misleading. *Syst Biol* 27: 401-410. (Cited on pages 6, 57 and 58.)
- [43] Gaut S, Lewis PO (1995) Success of maximum likelihood phylogeny inference in the four-taxon case. *Mol Biol Evol* 12: 152-162. (Cited on pages 6, 7 and 57.)
- [44] Gadagkar SR, Kumar S (2005) Maximum likelihood outperforms maximum parsimony even when evolutionary rates are heterotachous. *Mol Biol Evol* 22: 2139-2141. (Cited on page 6.)
- [45] Gaucher EA, Miyamoto MM (2005) A call for likelihood phylogenetics even when the process of sequence evolution is heterogeneous. *Mol Phylogenet Evol* 37: 928-931. (Cited on pages 6 and 57.)
- [46] Spencer M, Susko E, Roger AJ (2005) Likelihood, parsimony and heterogeneous evolution. *Mol Biol Evol* 22: 1161-1164. (Cited on page 6.)
- [47] Yang Z (1996) Among-site rate variation and its impact on phylogenetic analyses. *Tree* 11: 367-372. (Cited on pages 6 and 57.)
- [48] Rannala B, Hulsenbeck JP, Yang Z, Nielsen R (4) Taxon sampling and the accuracy of large phylogenies. *Syst Biol* 47: 702-710. (Cited on page 6.)
- [49] Felsenstein J, Sober E (1986) Parsimony and likelihood: an exchange. *Syst Zool* 35: 617-626. (Cited on page 6.)
- [50] Takezaki N, Nei M (1994) Inconsistency of the maximum parsimony method when the rate of nucleotide substitution is constant. *J Mol Evol* 39: 210-218. (Cited on page 6.)
- [51] Suzuki Y, Glazko GV, Nei M (2002) Overcredibility of molecular phylogenies obtained by Bayesian phylogenetics. *Proc Natl Acad Sci U S A* 99: 16138-16143. (Cited on pages 7, 27 and 37.)
- [52] Erixon P, Svennblad B, Britton T, Oxelman B (2003) Reliability of Bayesian posterior probabilities and bootstrap frequencies in phylogenetics. *Syst Biol* 52: 665-673. (Cited on pages 7, 27 and 37.)

- [53] Alfaro ME, Holder MT (2006) The posterior and the prior in bayesian phylogenetics. *Annu Rev Ecol Evol Syst* 37: 19-42. (Cited on page 7.)
- [54] Fukami-Kobayashi K, Tateno Y (1991) Robustness of maximum likelihood tree estimation against different patterns of base substitutions. *J Mol Evol* 32: 79-91. (Cited on pages 7 and 57.)
- [55] Huelsenbeck JP (1995) Performance of phylogenetic methods in simulation. *Syst Biol* 44: 17-48. (Cited on pages 7 and 57.)
- [56] Gu X, Zhang J (1997) A simple method for estimating the parameter of substitution rate variation among sites. *Mol Biol Evol* 14: 1106-1113. (Cited on pages 7 and 57.)
- [57] Lockhart PJ, Larkum AW, Steel MA, Waddell PJ, Penny D (1996) Evolution of chlorophyll and bacteriochlorophyll: The problem of invariant sites in sequence analysis. *Proc Natl Acad Sci U S A* 93: 1930-1934. (Cited on pages 7 and 57.)
- [58] Sullivan J, Swofford DL (1997) Are guinea pigs rodents? The importance of adequate models in molecular phylogenetics. *J Mammal Evol* 4: 77-86. (Cited on pages 7, 57 and 83.)
- [59] Sullivan J, Swofford DL, Naylor GJP (1999) The effect of taxon sampling on estimating rate heterogeneity parameters of maximum-likelihood models. *Mol Biol Evol* 16: 1347-1356. (Cited on pages 7, 57, 58, 67, 68 and 83.)
- [60] Swofford DL, Waddell PJ, Huelsenbeck JP, Foster PG, Lewis PO, et al. (2001) Bias in phylogenetic estimation and its relevance to the choice between parsimony and likelihood methods. *Syst Biol* 50: 525-539. (Cited on pages 7, 57 and 67.)
- [61] Pol D, Siddal ME (2001) Biases in maximum likelihood and parsimony: a simulation approach to a 10-taxon case. *Cladistics* 17: 266-281. (Cited on pages 7 and 57.)
- [62] Dress AWM, Flamm C, Fritsch G, Grünewald S, Kruspe M, et al. (2008) Noisy: identification of problematic columns in multiple sequence alignments. *Algorithms Mol Biol* 3: 7. (Cited on pages 11, 21, 28 and 43.)
- [63] Löytynoja A, Goldman N (2005) An algorithm for progressive multiple alignment of sequences with insertions. *Proc Natl Acad Sci USA* 102: 10557-10562. (Cited on page 11.)
- [64] Hartmann S, Vision TJ (2008) Using ESTs for phylogenomics: Can one accurately infer a phylogenetic tree from a gappy alignment? *BMC Evol Biol* 8:95: S13. (Cited on pages 11, 28 and 43.)

- [65] Wägele JW, Mayer C (2007) Visualizing differences in phylogenetic information content of alignments and distinction of three classes of long-branch effects. *BMC Evol Biol* 7: 147. (Cited on pages 11 and 58.)
- [66] Wong KMA, Suchard MA, Huelsenbeck JP (2008) Alignment uncertainty and genomic analysis. *Science* 319: 473-476. (Cited on page 11.)
- [67] Talavera G, Castresana J (2007) Improvement of phylogenies after removing divergent and ambiguously aligned blocks from protein sequence alignments. *Syst Biol* 56: 564-577. (Cited on pages 11, 12, 18, 21 and 43.)
- [68] Pei J, Sadreyev R, Grishin NV (2003) PCMA: fast and accurate multiple sequence alignment based on profile consistency. *Bioinformatics* 19: 427-428. (Cited on pages 12 and 13.)
- [69] Huson DH, Bryant D (2006) Application of phylogenetic networks in evolutionary studies. *Mol Biol Evol* 23: 254-267. (Cited on pages 13, 29 and 30.)
- [70] Bryant D, Moulton V (2004) Neighbor-Net: An agglomerative method for the construction of phylogenetic networks. *Mol Biol Evol* 21: 255-265. (Cited on pages 13, 29 and 30.)
- [71] Stamatakis A (2006) RAxML-VI-HPC: maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models. *Bioinformatics* 22: 2688-2690. (Cited on pages 14, 29, 71 and 72.)
- [72] Ott M, Zola J, Aluru S, Stamatakis A (2007) Large-scale Maximum Likelihood-based Phylogenetic Analysis on the IBM BlueGene/L. In: *Proceedings of ACM/IEEE Supercomputing conference 2007*. Reno NV., ACM, New York, NY, USA. (Cited on pages 14 and 29.)
- [73] Hasegawa M, Fujiwara M (1994) Relative efficiencies of the maximum likelihood, maximum parsimony, and neighbor-joining methods. *Mol Phylogenet Evol* 2: 1-5. (Cited on pages 14 and 21.)
- [74] Zrzavý J, Štys P (1997) The basic body plan of arthropods: insights from evolutionary morphology and developmental biology. *J Evol Biol* 10: 653-367. (Cited on page 18.)
- [75] Dohle W (2001) Are the insects terrestrial crustaceans? A discussion of some new facts and arguments and the proposal of the proper name "Tetraconata" for the monophyletic unit Crustacea + Hexapoda. *Ann Soc Entomol Fr (New Series)* 37: 85-103. (Cited on page 18.)
- [76] Richter S (2002) The Tetraconata concept: hexapod-crustacean relationships and the phylogeny of Crustacea. *Org Divers Evol* 2: 217-237. (Cited on page 18.)

- [77] Mallatt J, Giribet G (2006) Further use of nearly complete 28S and 18S rRNA genes to classify Ecdysozoa: 37 more arthropods and a kinorhynch. *Mol Phylogenet Evol* 3: 772-794. (Cited on page 18.)
- [78] Harzsch S (2006) Neurophylogeny: Architecture of the nervous system and a fresh view on arthropod phylogeny. *Integr Comp Biol* 46: 162-194. (Cited on page 18.)
- [79] Ungerer P, Scholtz G (2008) Filling the gap between identified neuroblasts and neurons in crustaceans adds new support for Tetraconata. *Proc Biol Sci* 275: 369-376. (Cited on page 18.)
- [80] Regier JC, Shultz JW, Ganley ARD, Hussey A, Shi D, et al. (2008) Resolving arthropod phylogeny: Exploring phylogenetic signal within 41 kb of protein-coding nuclear gene sequence. *Syst Biol* 57: 920-938. (Cited on page 18.)
- [81] von Reumont BM, Meusemann K, Szucsich NU, Dell'Ampio E, Bartel D, et al. (2009) Can comprehensive background knowledge be incorporated into substitution models to improve phylogenetic analyses? A case study on major arthropod relationships. *BMC Evol Biol* 9: 119. (Cited on pages 18, 47, 51, 53, 54, 71, 76 and 82.)
- [82] Mallatt J, Craig CW, Yoder MJ (2010) Nearly complete rRNA genes assembled from across the metazoan animals: Effects of more taxa, a structure-based alignment, and paired-sites evolutionary models on phylogeny reconstruction. *Mol Phylogenet Evol* 55: 1-17. (Cited on page 18.)
- [83] Richter S, Scholtz G (2001) Phylogenetic analysis of the Malacostraca (Crustacea). *J Zool Syst Evol Res* 39: 113-116. (Cited on page 18.)
- [84] Martin JW, Davis GW (2001) An updated classification of the recent Crustacea. Number 39 in Science Series. Natural History Museum of Los Angeles County. (Cited on page 18.)
- [85] Jenner RA, Ní Dhubhghaill C, Ferla MP, Wills MA (2009) Eumalacostracan phylogeny and total evidence: limitations of the usual suspects. *BMC Evol Biol* 9: 21. (Cited on page 18.)
- [86] Kjer KM, Carle FL, Litman J, Ware J (2006) A Molecular Phylogeny of Hexapoda. *Arthropod Syst Phylogeny* 64: 35-44. (Cited on page 18.)
- [87] Kjer KM (2004) Aligned 18S and insect phylogeny. *Syst Biol* 53: 506-514. (Cited on page 18.)
- [88] Luan Yx, Mallatt JM, Xie Rd, Yang Ym, Yin Wy (2005) The phylogenetic positions of three basal-hexapod groups (Protura, Diplura, and Collembola) based on on ribosomal RNA gene sequences. *Mol Biol Evol* 22: 1579-1592. (Cited on page 18.)

- [89] Gao Y, Bu Y, Luan YX (2008) Phylogenetic relationships of basal hexapods reconstructed from nearly complete 18S and 28S rRNA gene sequences. *Zoolog Sci* 25: 1139-1145. (Cited on page 18.)
- [90] Timmermans MJ, D Roelofs D, Mariën J, van Straalen NM (2008) Revealing pancrustacean relationships: Phylogenetic analysis of ribosomal protein genes places Collembola (springtails) in a monophyletic Hexapoda and reinforces the discrepancy between mitochondrial and nuclear DNA markers. *BMC Evol Biol* 8: 83. (Cited on page 18.)
- [91] Misof B, Niehuis O, Bischoff I, Rickert A, Erpenbeck D, et al. (2007) Towards an 18S phylogeny of hexapods: Accounting for group-specific character covariance in optimized mixed nucleotide/doublet models. *Zoology (Jena)* 110: 409-429. (Cited on page 18.)
- [92] Edgecombe GD (2009) Arthropod phylogeny: An overview from the perspectives of morphology, molecular data and the fossil record. *Arthropod Struct Dev* 39: 74-87. (Cited on page 21.)
- [93] Grimaldi DA (2009) 400 million years on six legs: On the origin and early evolution of Hexapoda. *Arthropod Struct Dev* 39: 191-203. (Cited on page 21.)
- [94] Nishihara H, Okada N, Hasegawa M (2007) Rooting the eutherian tree: the power and pitfalls of phylogenomics. *Genome Biol* 8: R199. (Cited on page 21.)
- [95] Blair JE, Ikeo K, Gojoberi T (2002) The evolutionary position of nematodes. *BMC Evol Biol* 2: 7. (Cited on page 21.)
- [96] Phillips MJ, Delsuc F, Penny D (2004) Genome-scale phylogeny and the detection of systematic biases. *Mol Biol Evol* 21: 1455-1458. (Cited on page 21.)
- [97] Delsuc F, Brinkmann H, Phillippe H (2005) Phylogenomics and the reconstruction of the tree of life. *Nat Rev Genet* 6: 361-375. (Cited on page 21.)
- [98] Bolton B (1990) The higher classification of the ant subfamily Leptanillinae (Hymenoptera: Formicidae). *Syst Ent* 15: 267-282. (Cited on page 27.)
- [99] Baroni Urbani C, Bolton B, Ward PS (1992) The internal phylogeny of ants (Hymenoptera: Formicidae). *Syst Ent* 17: 301-329. (Cited on page 27.)
- [100] Grimaldi D, Agosti D, Carpenter J (1997) New and rediscovered primitive ants (Hymenoptera: Formicidae) in Cretaceous amber from New Jersey, and their phylogenetic relationship. *Am Mus Novit* 3208: 1-43. (Cited on page 27.)
- [101] Grimaldi D, Agosti D (2000) A formicine in New Jersey Cretaceous amber (Hymenoptera: Formicidae) and early evolution of the ants. *Proc Natl Acad Sci U S A* 97: 13678-13683. (Cited on page 27.)

- [102] Wilson EO, Hölldobler B (2005) The rise of the ants: a phylogenetic and ecological explanation. *Proc Natl Acad Sci U S A* 102: 7411-7414. (Cited on page 27.)
- [103] Huelsenbeck J, Larget B, Miller R, Ronquist F (2002) Potential applications and pitfalls of Bayesian inference of phylogeny. *Syst Biol* 51: 673-688. (Cited on pages 27 and 37.)
- [104] Douady C, Delsuc F, Boucher Y, Doolittle W, Douzery E (2003) Comparison of Bayesian and Maximum Likelihood bootstrap measures of phylogenetic reliability. *Mol Biol Evol* 20: 248-254. (Cited on pages 27 and 37.)
- [105] Schwarzer J, Misof B, Tautz D, Schlieven UK (2009) The root of the East African cichlid radiations. *BMC Evol Biol* 9: 186. (Cited on pages 28, 43, 45 and 82.)
- [106] Simon S, Strauss S, von Haeseler A, Hadrys H (2009) A phylogenomic approach to resolve the basal pterygote divergence. *Mol Biol Evol* 26: 2719-2730. (Cited on pages 28, 43, 45, 71 and 82.)
- [107] Meusemann K, von Reumont BM, Simon S, Roeding F, Kück P, et al. (2010) A phylogenomic approach to resolve the arthropod tree of life. *Mol Biol Evol* 27: 2451-2464. (Cited on pages 28, 43, 45, 76, 77 and 82.)
- [108] Muriene J, Edgecombe G, Giribet G (2010) Including secondary structure, fossils and molecular dating in the centipede tree of life. *Mol Phylogenet Evol* 57: 301-313. (Cited on pages 28, 43, 45, 57, 82 and 83.)
- [109] Dinapoli A, Zinssmeister C, Klussmann-Kolb A (2011) New insights into the phylogeny of the Pyramidellidae (Gastropoda). *J Mollus Stud* 77: 1-7. (Cited on pages 28, 43, 45 and 82.)
- [110] Katoh K, Toh H (2008) Recent developments in the MAFFT multiple sequence alignment program. *Brief Bioinform* 9: 286-298. (Cited on pages 28 and 38.)
- [111] Kück P (2009) ALICUT: a Perlscript which cuts ALISCOPE identified RSS. Department of Bioinformatics, Zoologisches Forschungsmuseum A. Koenig (ZFMK), Bonn, Germany, version 1.0 edition. (Cited on pages 28, 76, 79, 127, 132 and 138.)
- [112] Kück P, Meusemann K (2010) FASconCAT: Convenient handling of data matrices. *Mol Phylogenet Evol* 56: 1115-1118. (Cited on pages 28, 79 and 138.)
- [113] Bryant D, Moulton V, Spillner A (2007) Consistency of the neighbor-net algorithm. *Algorithms Mol Biol* 2: 8. (Cited on pages 29 and 30.)
- [114] Pratas F, Trancoso P, Stamatakis A, Sousa L (2009) Fine-grain parallelism using multi-core, Cell/BE, and GPU systems: Accelerating the phylogenetic

- likelihood function. In: Proceedings of ICPP 2009. Vienna, Austria, IEEE Computer Society, Los Alamitos, CA, USA. (Cited on page 29.)
- [115] Pattengale N, Alipour M, Bininda-Emonds ORP, Moret BME, Gottlieb EJ, et al. (2010) How many bootstrap replicates are necessary? *J Comput Biol* 17: 337-354. (Cited on pages 29, 30 and 37.)
- [116] Stöver BC, Müller KF (2010) TreeGraph 2: combining and visualizing evidence from different phylogenetic analyses. *BMC Bioinformatics* 11: 7. (Cited on pages 29 and 30.)
- [117] Shimodaira H (2002) An approximately unbiased test of phylogenetic tree selection. *Syst Biol* 51: 492-508. (Cited on pages 29 and 32.)
- [118] Shimodaira H, Hasegawa M (2001) CONSEL: for assessing the confidence of phylogenetic tree selection. *Bioinformatics* 17: 1246-1247. (Cited on page 29.)
- [119] Huelsenbeck J, Ronquist F (2001) MrBayes: Bayesian inference of phylogenetic trees. *Bioinformatics* 17: 754-755. (Cited on page 29.)
- [120] Ronquist F, Huelsenbeck J (2003) MrBayes 3: Bayesian phylogenetic inference under mixed models. *Bioinformatics* 19: 1572-1574. (Cited on pages 29, 71, 72 and 74.)
- [121] Rambaut A, J DA (2009) Tracer. Available at <http://beast.bio.ed.ac.uk/software/tracer/>, version 1.5 edition. (Cited on page 30.)
- [122] Ward PS (2010) Taxonomy, phylogenetics, and evolution. In: Lach L, Parr CL, Abbott K, editors, *Ant Ecology*, Oxford: Oxford University Press, chapter 1. pp. 3-17. (Cited on page 36.)
- [123] Kück P, Hita-Garcia F, Misof B, Meusemann K (2011) Improved phylogenetic analyses corroborate a plausible position of *Martialis Heureka* in the ant tree of life. *PLoS ONE* 6: e21031. (Cited on pages 39, 45, 76, 78 and 137.)
- [124] Fletcher W, Yang Z (2009) INDELible: A flexible simulator of biological sequence evolution. *Mol Biol Evol* 26: 1879-1888. (Cited on pages 47 and 58.)
- [125] Guindon S, Gascuel O (2003) A simple, fast, and accurate algorithm to estimate large phylogenies by maximum likelihood. *Syst Biol* 52: 696-704. (Cited on pages 47 and 61.)
- [126] Guindon S, Dufayard JF, Lefort V, Anisimova M, Hordijk W, et al. (2010) PhyML 3.0: New algorithms and methods to estimate maximum-likelihood phylogenies: assessing the performance of PhyML 3.0. *Syst Biol* 59: 307-321. (Cited on pages 47 and 61.)

- [127] Wallosek D (1998) On the Cambrian diversity of Crustacea. In: vVaupel Klein FRSJC, editor, *Crustaceans and the biodiversity crisis*, Proceedings of the fourth international crustacean congress, Leiden, Netherlands: Brill Academic Publishers. pp. 3-27. (Cited on page 54.)
- [128] Giribet G, Edgecombe GD, Wheeler WC (2001) Arthropod phylogeny based on eight molecular loci and morphology. *Nature* 413: 157-161. (Cited on page 54.)
- [129] Regier JC, Shultz JW, Kambic RE (2005) Pancrustacean phylogeny: hexapods are terrestrial crustaceans and maxillopods are not monophyletic. *Proc Biol Sci* 272: 395-401. (Cited on page 54.)
- [130] Regier JC, Shultz JW, Zwick A, Hussey A, Ball B, et al. (2010) Arthropod relationships revealed by phylogenomic analysis of nuclear protein-coding sequences. *Nature* 319: 473-476. (Cited on pages 54, 57 and 83.)
- [131] Felsenstein J (1973) Maximum likelihood and minimum-steps methods for estimating evolutionary trees from data on discrete characters. *Syst Zool* 22: 240-249. (Cited on page 57.)
- [132] Felsenstein J (1981) Evolutionary trees from DNA sequences: a maximum likelihood approach. *J Mol Evol* 17: 368-376. (Cited on page 57.)
- [133] Chang JT (1996) Full reconstruction of Markov models on evolutionary trees: identifiability and consistency. *Math Biosci* 137: 51-73. (Cited on page 57.)
- [134] Rogers JS (1997) On the consistency of maximum likelihood estimation of phylogenetic trees from nucleotide sequences. *Syst Biol* 46: 354-357. (Cited on page 57.)
- [135] Bruno WJ, Halpern AL (1998) Topological bias and inconsistency in maximum likelihood using wrong models. *Mol Biol Evol* 16: 564-566. (Cited on page 57.)
- [136] Anderson FE, Swofford DL (2004) Should we be worried about long-branch attraction in real data sets? Investigations using metazoan 18S rDNA. *Mol Phylogenet Evol* 33: 440-451. (Cited on pages 57, 67 and 83.)
- [137] Kolaczkowski B, Thornton JW (2004) Performance of maximum parsimony and likelihood phylogenetics when evolution is heterogeneous. *Nature* 431: 980-984. (Cited on page 57.)
- [138] Kelchner SA, Thomas MA (2006) Model use in phylogenetics: nine key questions. *Trends Ecol Evol* 22: 87-94. (Cited on page 57.)
- [139] Yang Z, Goldman N, Friday A (1994) Comparison of models for nucleotide substitution used in Maximum-Likelihood phylogenetic estimation. *Mol Biol Evol* 11: 316-324. (Cited on pages 57, 67 and 83.)



- [140] Huelsenbeck JP, Hillis DM (1993) Success of phylogenetic methods in the four-taxon case. *Syst Zool* 42: 247-264. (Cited on page 57.)
- [141] Yang Z (1993) Maximum likelihood estimation of phylogeny from DNA sequences when substitution rates differ over time. *Mol Biol Evol* 10: 1396-1401. (Cited on page 57.)
- [142] Yang Z, Goldman N, Friday AE (1995) Maximum likelihood trees from DNA sequences: a peculiar statistical estimation problem. *Syst Biol* 44: 384-399. (Cited on page 57.)
- [143] Yang Z (1997) How often do wrong models produce better phylogenies? *Mol Biol Evol* 14: 105-108. (Cited on page 57.)
- [144] Siddal ME (1998) Success of parsimony in the four-taxon case: Long branch repulsion by likelihood in the Farris zone. *Cladistics* 14: 209-220. (Cited on page 57.)
- [145] Sullivan J, Swofford DL (2001) Should we use model-based methods for phylogenetic inference when we know that assumptions about among-site rate variation and nucleotide substitution pattern are violated? *Syst Biol* 50: 723-729. (Cited on pages 57 and 58.)
- [146] Fischer M, Steel M (2009) Sequence length bounds for resolving a deep phylogenetic divergence. *J Theor Biol* 256: 247-252. (Cited on page 57.)
- [147] Huelsenbeck JP (1997) Is the Felsenstein zone a fly trap? *Syst Biol* 46: 69-74. (Cited on pages 57, 58, 67 and 83.)
- [148] Felsenstein J (1984) Distance methods for inferring phylogenies: a justification. *Evolution* 38: 16-24. (Cited on page 57.)
- [149] Gu X, Fu YX, Li WH (1995) Maximum likelihood estimation of the heterogeneity of substitution rate among nucleotide sites. *Mol Biol Evol* 12: 546-557. (Cited on pages 57 and 83.)
- [150] Phillipe H, Germot A (2000) Phylogeny of eukaryotes based on ribosomal RNA: Long-Branch Attraction and models of sequence evolution. *Mol Biol Evol* 17: 830-834. (Cited on pages 57, 67, 82 and 83.)
- [151] Sanderson MJ, Wojciechowski MF, Hu JM, Sher-Khan T, Brady SG (2000) Error, bias, and long-branch attraction in data for two chloroplast photosystem genes in seed plants. *Mol Biol Evol* 17: 782-797. (Cited on pages 57, 82 and 83.)
- [152] Savard J, Tautz D, Richards S, Weinstock GM, Gibbs RA, et al. (2006) Phylogenomic analysis reveals bees and wasps (Hymenoptera) at the base of the radiation of Holometabolous insects. *Genome Res* 16: 1334-1338. (Cited on pages 57 and 83.)

- [153] Rota-Stabelli O, Campbell L, Brinkmann H, Edgecombe GD, Longhorn SJ, et al. (2010) A congruent solution to arthropod phylogeny: phylogenomics, microRNAs and morphology support monophyletic Mandibulata. *Proc R Soc B* 278: 298-306. (Cited on pages 57 and 83.)
- [154] Mayrose I, Friedman N, Pupko T (2005) A gamma mixture model better accounts for among site heterogeneity. *Bioinformatics* 21: 151-158. (Cited on pages 57 and 83.)
- [155] Ren F, Tanaka H, Yang Z (2005) An empirical examination of the utility of codon-substitution models in phylogeny reconstruction. *Syst Biol* 54: 808-818. (Cited on pages 57 and 83.)
- [156] Tourasse NJ, Gouy M (1997) Evolutionary distances between nucleotide sequences based on the distribution of substitution rates among sites as estimated by parsimony. *Mol Biol Evol* 14: 287-298. (Cited on pages 57 and 58.)
- [157] Huelsenbeck JP, Crandall KA (1997) Phylogeny estimation and hypotheses testing using maximum likelihood. *Annu Rev Ecol Syst* 28: 437-466. (Cited on page 58.)
- [158] Sullivan J, Holsinger KE, Simon C (1996) The effect of topology on estimates of among-site rate variation. *J Mol Evol* 42: 308-312. (Cited on pages 68 and 83.)
- [159] Phillipe H, Laurent J (1998) How good are deep phylogenetic trees? *Curr Opin Genet Dev* 8: 616-623. (Cited on pages 68 and 83.)
- [160] Letsch HO, Greve C, Kück P, Fleck G, Stocsits RR, et al. (2009) Simultaneous alignment and folding of 28S rRNA sequences uncovers phylogenetic signal in structure variation. *Mol Phylogenet Evol* 53: 758-771. (Cited on pages 71 and 76.)
- [161] Dinapoli A, Klussmann-Kolb A (2010) The long way to diversity – Phylogeny and evolution of the Heterobranchia (Mollusca:Gastropoda). *Mol Phylogenet Evol* 55: 60-76. (Cited on page 71.)
- [162] Dunn CW, Hejnol A, Matus DQ, Pang K, Browne WE, et al. (2008) Broad phylogenomic sampling improves resolution of the animal tree of life. *Nature* 452: 745-749. (Cited on page 71.)
- [163] Phillipe H, Derelle R, Lopez P, Pick K, Borchiellini C, et al. (2009) Phylogenomics revives traditional views on deep animal relationships. *Curr Biol* 19: 706-712. (Cited on page 71.)
- [164] Pearson WR, Lipman DJ (1988) Improved tools for biological sequence comparison. *Biochemistry* 85: 2444-2448. (Cited on page 71.)

- [165] Maddison DR, Swofford DL, Maddison WP (1997) NEXUS: an extensible file format for systematic information. *Syst Biol* 46: 590-621. (Cited on page 71.)
- [166] Higgins DG, Sharp PM (1988) CLUSTAL: a package for performing multiple sequence alignment on a microcomputer. *Gene* 73: 237-244. (Cited on page 71.)
- [167] Felsenstein J (1989) PHYLIP - phylogeny inference package (version 3.2). *Cladistics* 5: 164-166. (Cited on page 71.)
- [168] Gowri-Shankar V, Jow H (2006) *PHASE*: a software package for *Phylogenetics And Sequence Evolution*. University of Manchester, 2.0 edition. (Cited on page 71.)
- [169] Pina-Martins F, Paulo OS (2008) CONCATENATOR: sequence data matrices handling made easy. *Mol Ecol Resour* 8: 1254-1255. (Cited on page 72.)
- [170] Swofford DL (2003) *PAUP\**. Phylogenetic Analysis Using Parsimony (\*and other methods). Sinauer Associates, Sunderland, Massachusetts, version 4 edition. (Cited on pages 72 and 74.)
- [171] Seehausen O (2004) Hybridization and adaptive radiation. *Trends Ecol Evol* 19: 198-207. (Cited on page 76.)
- [172] Kück P (2009) SusEX: a Perlscript which finds subsequences in FASTA alignments specified by user predefined start and end strings. Department of Bioinformatics, Zoologisches Forschungsmuseum A. Koenig (ZFMK), Bonn, Germany, version 0.1 beta edition. (Cited on page 77.)
- [173] Kück P (2008) ESTa: a Perlscript which downloads the actual EST-summary list for all taxa with more than 1.000 EST database entries from the NCBI data base. Department of Bioinformatics, Zoologisches Forschungsmuseum A. Koenig (ZFMK), Bonn, Germany, version 0.1 beta edition. (Cited on page 77.)
- [174] Kück P (2007) TaxEd: a Perlscript which converts downloaded NCBI sequences into a more simplified and standardised information format. Department of Bioinformatics, Zoologisches Forschungsmuseum A. Koenig (ZFMK), Bonn, Germany, version 0.1 beta edition. (Cited on page 77.)
- [175] Kück P (2010) LoBraTe: a Perlpipeline to handle simulations of different long branch effects (LBA) under different maximum likelihood model and parameter settings to identify reasons of inconsistencies of the maximum likelihood approach relative to LBA. Department of Bioinformatics, Zoologisches Forschungsmuseum A. Koenig (ZFMK), Bonn, Germany, version 1.0 edition. (Cited on page 77.)

- [176] Kück P (2010) RAxTAX: a Perlpipeline to execute a full phylogenetic analyses starting from raw sequence data and ending by a full maximum likelihood analysis. Department of Bioinformatics, Zoologisches Forschungsmuseum A. Koenig (ZFMK), Bonn, Germany, version 1.0 edition. (Cited on page 77.)
- [177] Kück P (2009) SecSITE: a Perlpipeline to identify the impact of rRNA secondary structure consideration in alignment and tree reconstruction. Department of Bioinformatics, Zoologisches Forschungsmuseum A. Koenig (ZFMK), Bonn, Germany, version 1.0 edition. (Cited on page 78.)
- [178] Kück P (2009) SPIPES: a Perlpipeline to identify the effect of alignment masking approaches in regard to alignment structure improvement. Department of Bioinformatics, Zoologisches Forschungsmuseum A. Koenig (ZFMK), Bonn, Germany, version 1.0 edition. (Cited on page 78.)
- [179] Inagaki Y, Susko E, Fast NM, Roger AJ (2004) Covarion shifts cause a long-branch attraction artifact that unites Microsporidia and Archaeobacteria in EF-1 $\alpha$  phylogenies. *Mol Biol Evol* 21: 1340-1349. (Cited on page 82.)
- [180] Fares MA, Byrne KP, Wolfe KH (2006) Rate asymmetry after genome duplication causes substantial long-branch attraction artifacts in the phylogeny of *Saccharomyces* species. *Mol Biol Evol* 23: 245-253. (Cited on page 82.)
- [181] Brinkmann H, Phillipe H (2008) Animal phylogeny and large-scale sequencing: progress and pitfalls. *J Syst Evol* 46: 274-286. (Cited on page 82.)
- [182] Thomson RC, Shaffer BH (2010) Sparse supermatrices for phylogenetic inference: taxonomy, alignment, rogue taxa, and the phylogeny of living turtles. *Syst Biol* 59: 42-58. (Cited on page 82.)
- [183] Pattengale ND, Aberer AJ, Swenson KM, Stamatakis A, Moret BME (2011) Uncovering hidden phylogenetic consensus in large data sets. *Comput Biol Bioinf* 8: 902-911. (Cited on page 82.)
- [184] Phillipe H, Delsuc F, Brinkmann H, Lartillot N (2005) Phylogenomics. *Annu Rev Ecol Evol Syst* 36: 541-562. (Cited on page 84.)



# Additional Information Chapter 3

## A.1 Bayesian majority rule consensus topologies

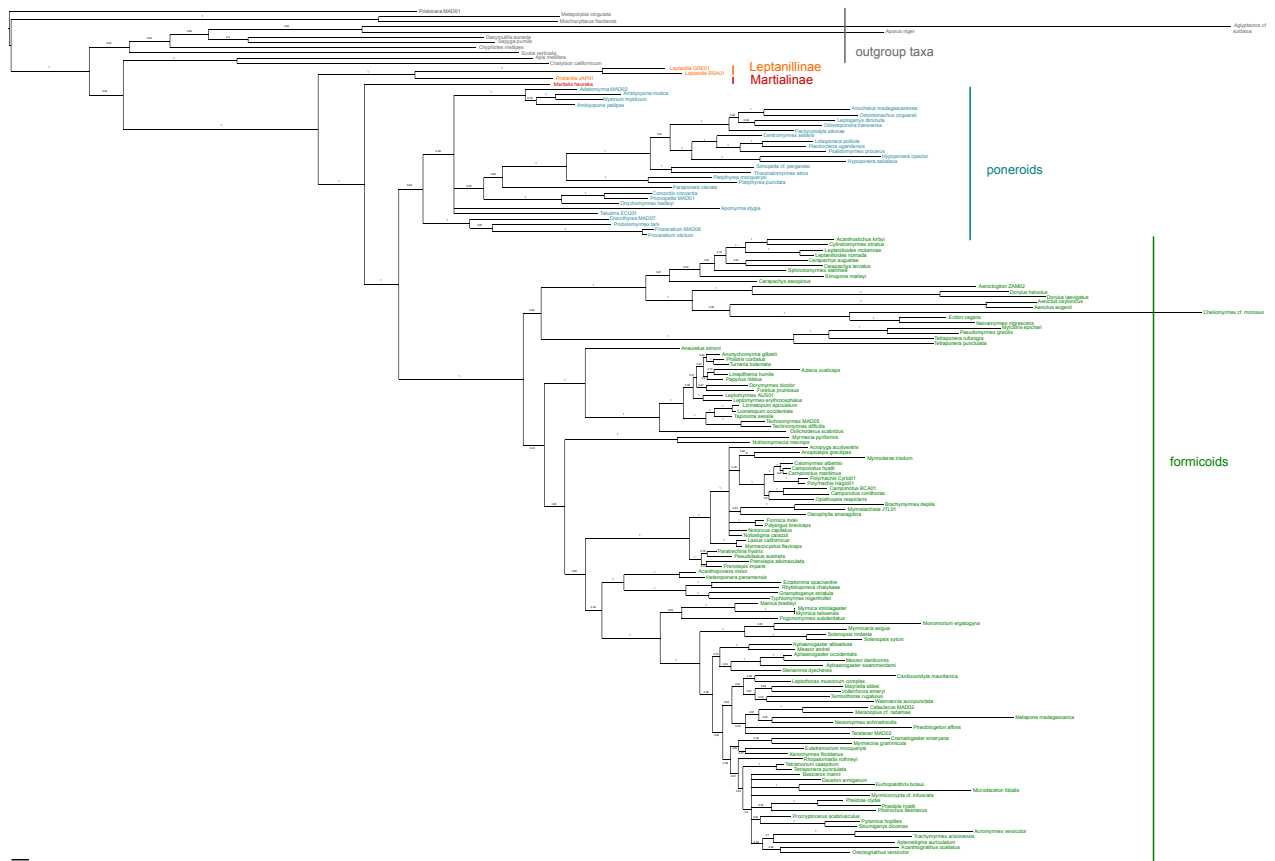


Figure A.1: **Unmasked, unpartitioned data set.** Bayesian (majority rule consensus) topology inferred from the unmasked, unpartitioned data set with 5,000 bootstrap replicates (GTR+ $\Gamma$ , 28,130,500 generations, sample frequency 100, burn-in (10%) discarded; see method section Chapter 3). The tree was rooted with *Pristocera*.



Figure A.2: **Unmasked, unpartitioned data set.** Bayesian (majority rule consensus) topology inferred from the masked, unpartitioned data set with 5,000 bootstrap replicates (GTR+ $\Gamma$ , 30 million generations, sample frequency 200, burn-in (10%) discarded; see method section Chapter 3). The tree was rooted with *Pristocera*.

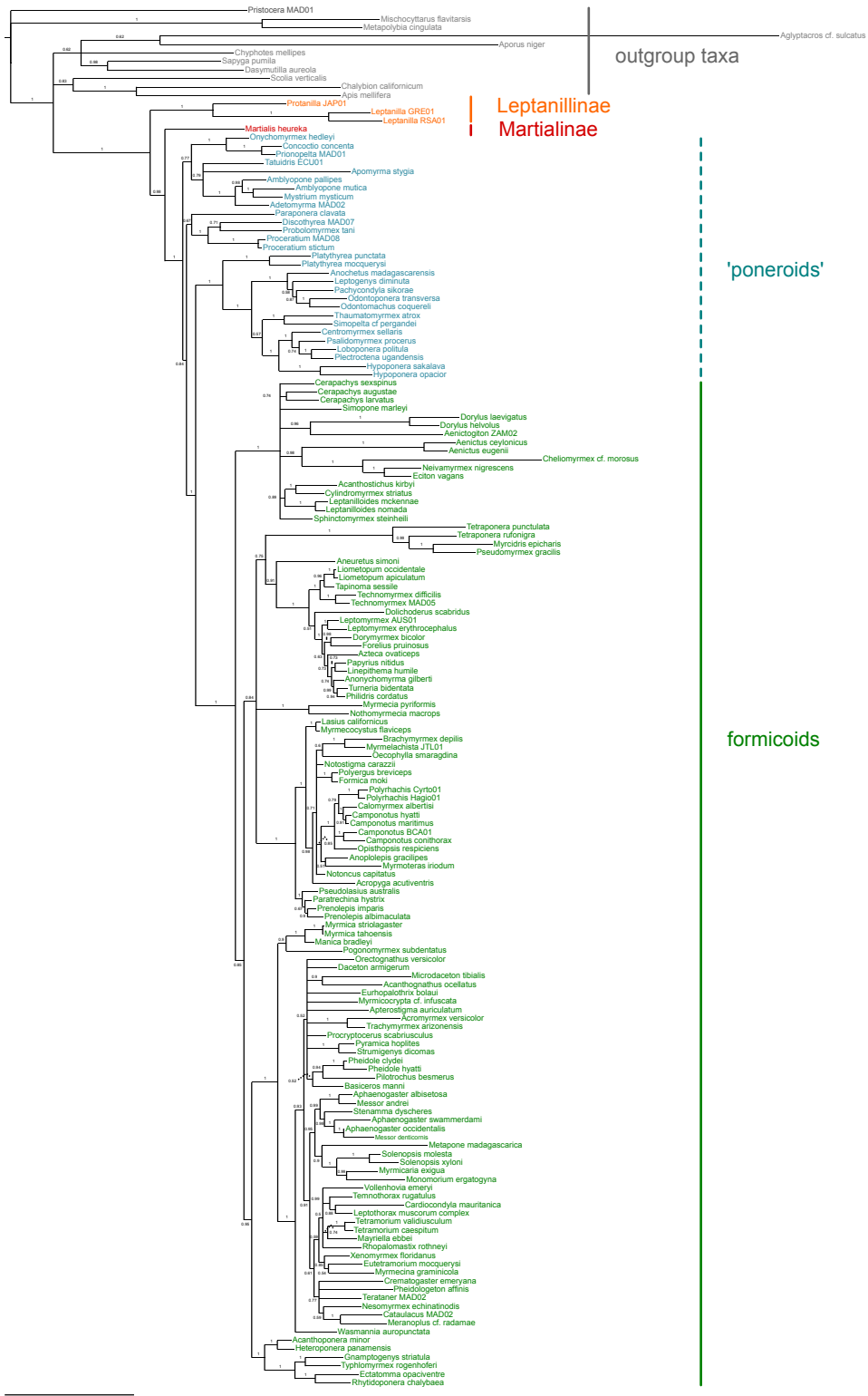


Figure A.3: Unmasked, unpartitioned data set. Bayesian (majority rule consensus) topology inferred from the masked, partitioned data set with 5,000 bootstrap replicates (GTR+ $\Gamma$ , 30 million generations, sample frequency 200, burn-in (10%) discarded; see method section Chapter 3). The tree was rooted with *Pristocera*.



## A.2 Maximum Likelihood majority rule consensus topologies



Figure A.4: **Unmasked, unpartitioned data set.** Maximum Likelihood (majority rule consensus) topology inferred from the unmasked, unpartitioned data set with 5,000 bootstrap replicates (-f a; GTR+ $\Gamma$ , see method section Chapter 3). The tree was rooted with *Pristocera*.

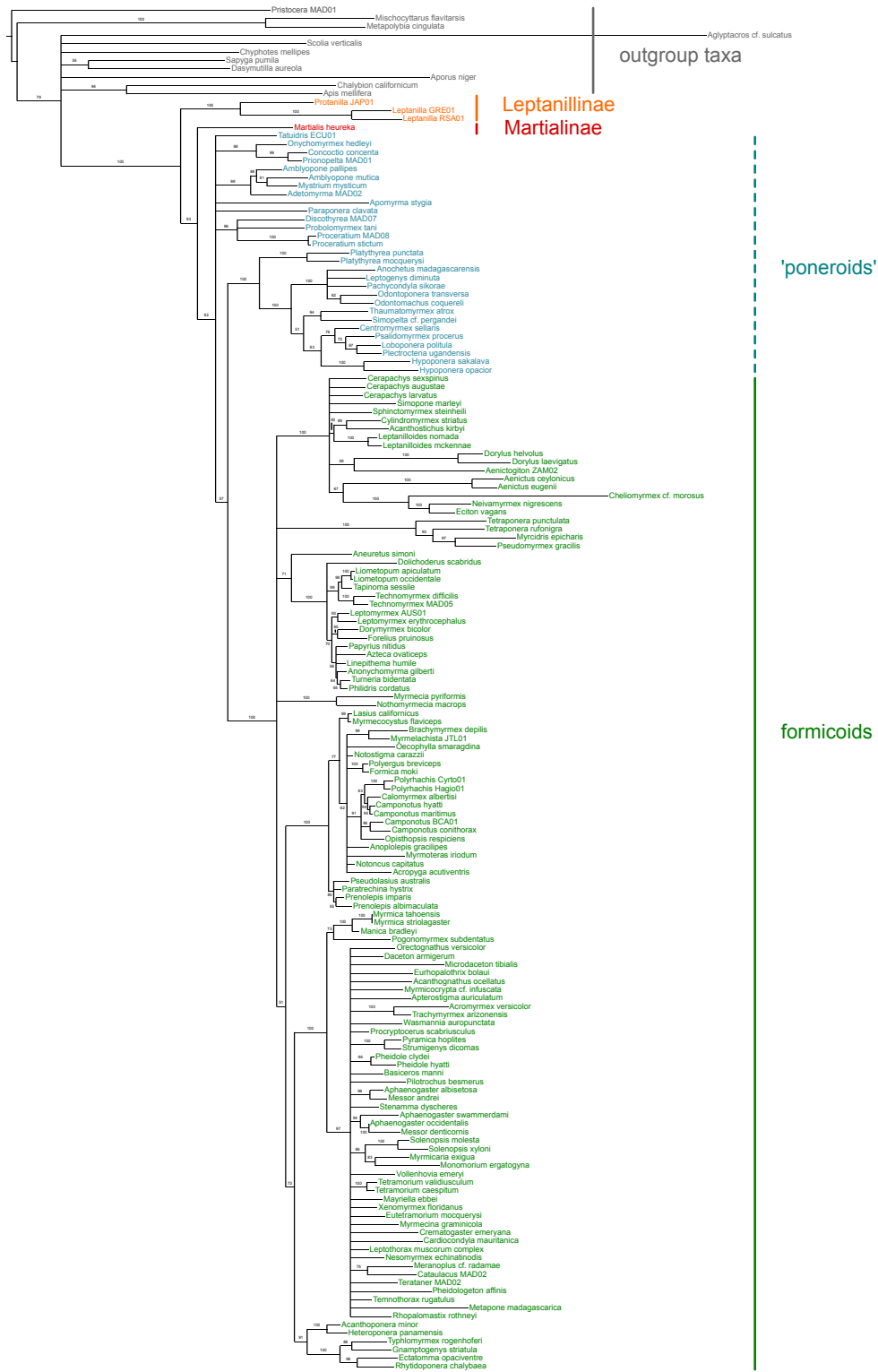


Figure A.5: **Masked, unpartitioned data set.** Maximum Likelihood (majority rule consensus) topology inferred from the masked, unpartitioned data set with 5,000 bootstrap replicates (-f a; GTR+ $\Gamma$ , see method section Chapter 3). The tree was rooted with *Pristocera*.

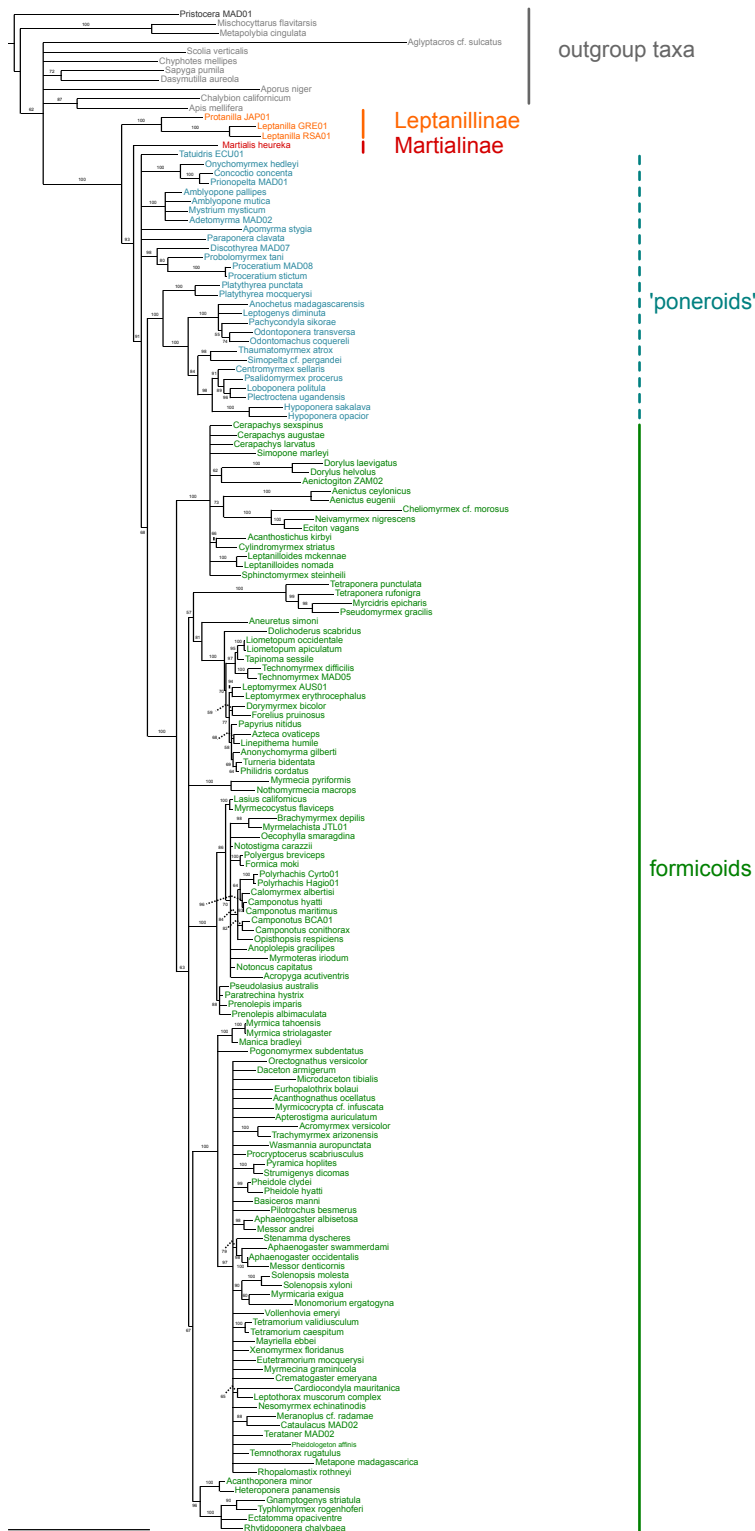


Figure A.6: **Masked, partitioned data set.** Maximum Likelihood (majority rule consensus) topology inferred from the masked, partitioned data set with 5,000 bootstrap replicates (-f a; GTR+ $\Gamma$ , see method section Chapter 3). The tree was rooted with *Pristocera*.

## B.1 Flowchart of the LoBraTe Process Pipeline

LoBraTe (Long Branch Test) is a process pipeline designed to infer the behaviour of different branch lengths on Maximum Likelihood inference under different evolutionary model assumptions. Additionally, LoBraTe calculates branch length relations of correct and incorrect relationships with a special mathematical algorithm including a likelihood ratio test and chi square test. LoBraTe is actually used to test the mathematical algorithm for its efficiency to identify long branch attraction between strongly derived taxa. LoBraTe was also used for the simulation analyses of chapter 5 and 4. For chapter 5, over 800,000 simulations are automatically analyzed with LoBraTe.

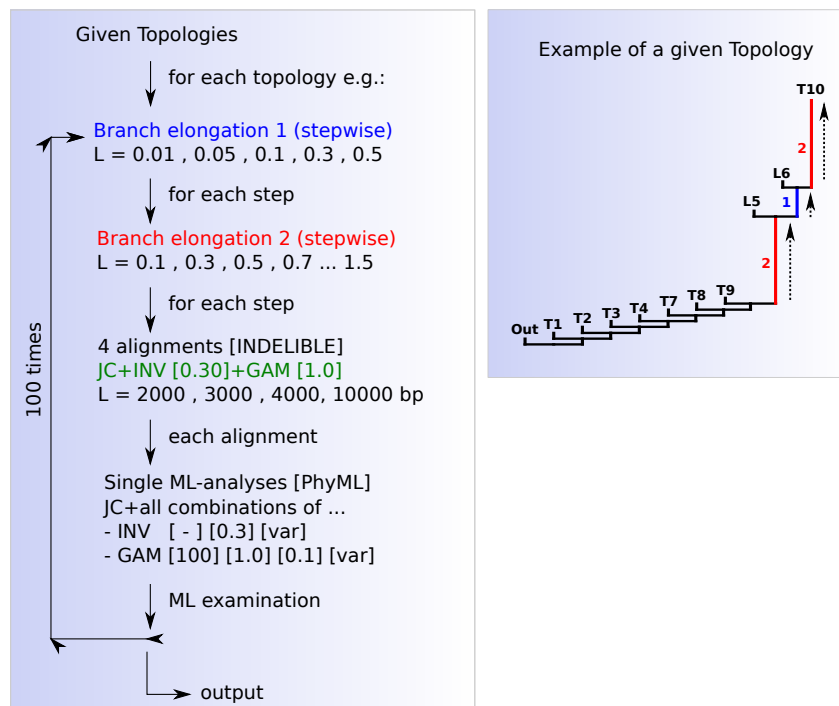


Figure B.1: Overview of the LoBraTe simulation and analyse processes.

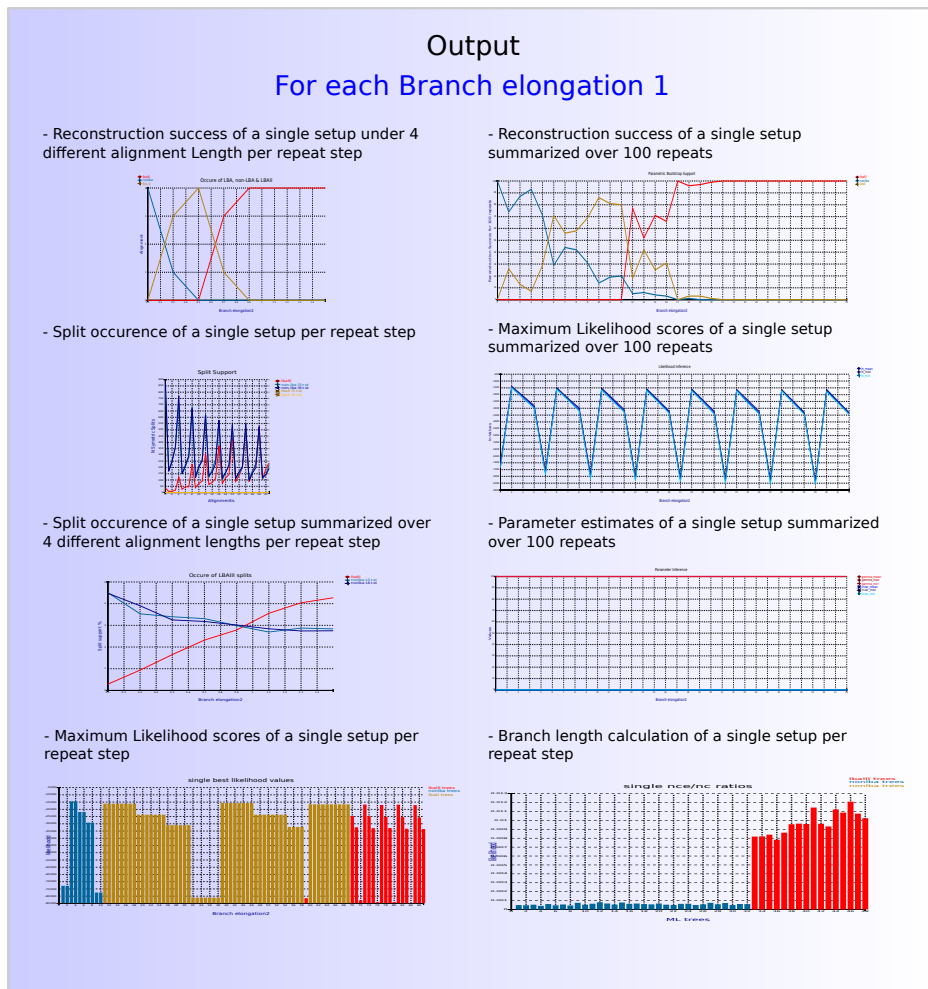


Figure B.2: Overview of single LoBraTe output plots.

APPENDIX C

# RAXTAX

## C.1 Flowchart of the RAXTAX Process Pipeline

RAXTAX is a process pipeline designed to execute a full phylogenetic analysis starting from raw sequence data and ending by a full Maximum Likelihood analysis. Figure C.1 gives an schematic overview about optional and stringent starting commands and the handling of an optional given taxon-restriction inputfile. Figure C.2 shows all single subprocesses of a full RAXTAX analysis in which only concatenated data is completely analysed. Parallel to this, RAXTAX can completely analyse all single masked and unmasked files within the same process run.

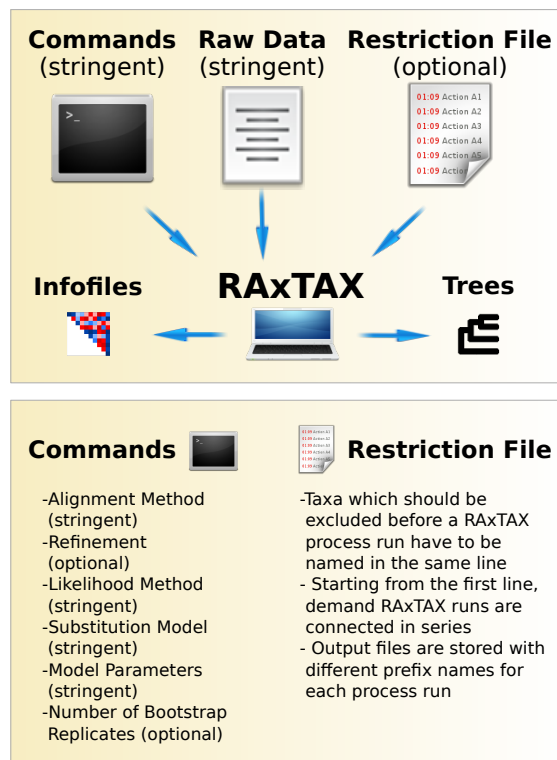


Figure C.1: Schematic overview about input and output of RAXTAX.

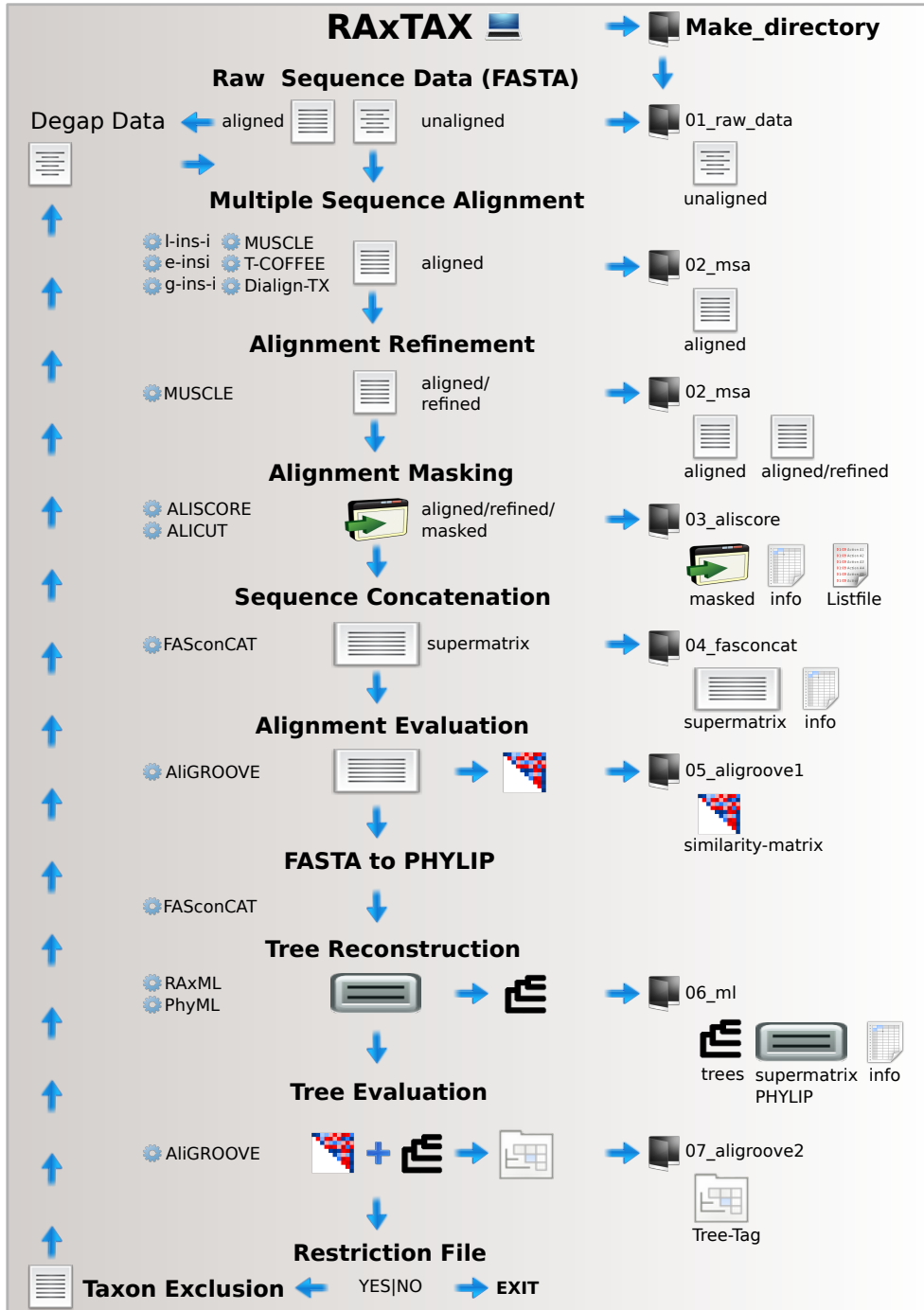


Figure C.2: Overview of all subprocesses during a complete RAXTAX analysis.

# Manual FASconCAT

---

## D.1 Introduction

FASconCAT is designed to concatenate sequence alignment files into one supermatrix file in a convenient manner. The supermatrix, for which different output formats are selectable (FASTA, PHYLIP, NEXUS) can be directly used for phylogenetic purposes. It considers standard nucleotide sequence alignments, recoded nucleotide sequences (e.g. with the third position of a codon RY coded), and amino acid alignments. Provided structure strings (in dot-bracket format), often used e.g. in ribosomal RNA analyses, are recognized and concatenated as well. FASconCAT can handle input files in PHYLIP, CLUSTAL and FASTA format in one single run, there has to be no unique input format. Within a sequence file, sequences must have equal length. The software extracts taxon specific associated gene- or structure sequences out of given input files and links them to one string. Missing taxon sequences in single files are replaced either by 'N' (nucleotide data), 'X' (amino acid data) or by '.' (structure strings in 'dot-bracket' format), dependent on their taxon associated data level. It is possible to concatenate nucleotide and amino acid files into one supermatrix file. FASconCAT can read sequences in interleaved and non-interleaved format. For given FASTA files, the program tolerates line breaks in sequences, but not in sequence (taxon) names. Sequence names may only include alphanumeric signs, underscores (`_`) and blanks. FASconCAT will issue an error prompt and die if any non-alphanumeric sign is encountered in sequence names.

FASconCAT was written on Linux and works on WindowsPCs, Mac OS and Linux running systems. Input files originating from Windows, CRFL line feeds should be converted into Unix (LF) line feeds in advance, especially, if the user changes the operating system. This can be done in several editors like e.g. **Bioedit**, **Notepad++** or **Scite**. FASconCAT usually replaces them, but might not succeed in every instance.

Ambiguities and indels are allowed. Any other sign in sequences, except for those covered by the universal DNA/RNA or amino acid code, will also lead to an error prompt. Structure information (e.g. of ribosomal RNA sequences) are also recognized, analyzed and concatenated. Structure information should be present in each file once and associated with equal taxon names, e.g. "structure". Otherwise, the software will interrupt with a specific error prompt. FASconCAT provides additionally information about each input file and the new concatenated supermatrix in .xls format. The file includes single range information of each gene (gene fragment or partition) and a list of all concatenated sequences. If structure strings have



been included, it lists the number and percentage of unpaired and paired alignment positions of each single file and the supermatrix file. Optionally, extended information is provided. The extended information setting includes reports about e.g. base composition of single files and the supermatrix file for nucleotide data. Further, if structure strings in dot-bracket format have been included, the concatenated structure composition of loop and stem positions are printed in a separate .txt file (-i, see below). For a more detailed report about additional information see section 'Usage/Options'.

As another option, FASconCAT can generate NEXUS files of concatenated sequences, either with commands which can be directly executed in PAUP or MrBayes, or without any specific commands. It is also possible to generate output files in PHYLIP format with relaxed- (unlimited signs) or strict (limited up to ten signs) sequence names while sequences are always printed out as non-interleaved. FASconCAT can be started directly via command line or indirectly, guided by menu options.

## D.2 Usage/Options

To run FASconCAT, open the terminal of your running system. Move through your directory path to the folder where FASconCAT and input files are placed. Type the name of your FASconCAT version, followed either by a blank and a) your demand options in one row to start FASconCAT directly or b) followed by pressing <enter> to get into the FASconCAT menu. Notice that all input files have to be located in the FASconCAT including folder. To execute FASconCAT, a Perl interpreter must be installed on the current system. Linux and Mac OS systems do not need a subsequent installation because the Perl interpreter is usually included as a standard tool. Unfortunately, Windows users have to install a Perl interpreter *ex post*. We recommend the ActivePerl interpreter which can be downloaded for free under:

- <http://activeperl.softonic.de/>

### D.2.1 Start FASconCAT via menu

#### D.2.1.1 Open the menu under Windows

Open a prompt (DOS) terminal on your Windows system and navigate to the folder where FASconCAT and files are located <cd your\_path...>. Then open FASconCAT:

- C:\FASconCAT\_Folder> FASconCAT\_v1.0.pl <enter>

#### D.2.1.2 Open the menu under Linux/Mac

Open a terminal and navigate to the folder where FASconCAT and files are located <cd your\_path...>. Then open FASconCAT:

- user@user:~/FASconCAT\_Folder> perl FASconCAT\_v1.0.pl <enter>

### D.2.1.3 Menu handling

The main menu of FASconCAT is subdivided into two parts separated by a dashed line. The upper component constitutes all possible options and their associated commands for adjustment. The lower part shows the current parameter setting of FASconCAT.

```

~/.Perl/FASconCAT_tk/2.1
Patrick@PREACHER ~/Perl/FASconCAT_tk/2.1
$ perl FASconCAT_v1.0.pl

-----
Welcome to FASconCAT v1.0 !
A perlscript for sequence concatenation
written by Patrick Kueck (ZPMK Bonn, 2010)
-----

START  FASconCAT      :          type <s> <enter>
INPUT  ALL/SINGLE     :          type <f> <enter>
INFO   ALL/SMALL     :          type <i> <enter>
NEXUS  BLOCK/Mr-BAYES :          type <n> <enter>
PHYLLIP NO/YES       :          type <p> <enter>
HELP   FASconCAT    :          type <h> <enter>
QUIT   FASconCAT    :          type <q> <enter>
PREFACE FASconCAT    :          type <a> <enter>
-----

FASTA/PHYLLIP-INPUT
Concatenate ALL files : YES
Concatenate SINGLE files : NO

OUTPUT
Supermatrix + ALL info : NO
Supermatrix             : YES
NEXUS-Block            : NO
PHYLLIP                 : NO
-----

COMMAND:  -

```

Figure D.1: Menu of FASconCAT

If you like to change the default parameter setting, type the option associated command into the command line and press <enter>. The new setting configuration will be displayed in the lower part of the menu. If the parameter configuration is completed, FASconCAT can be started by typing “s” and pressing <enter>. For getting help type “h” and press <enter>, to return to the main menu type “b” and press <enter>. To quit the program type “q” and press <enter>.

### D.2.2 Start FASconCAT via single command line

FASconCAT can be directly started via command line. Therefore, commands and chosen options has to be typed in one row in the terminal. Start FASconCAT via command line simplifies the implementation of FASconCAT into complex process pipelines. Move through your directory path to the folder where FASconCAT and your files are located and type the name of the FASconCAT version, followed by a blank and the demand options with a minus (-) sign in front of each. Then press <enter>. Make sure you write the input options correctly, for example “-i” and not “- i”. Otherwise FASconCAT will not start working but instead open the main menu.

- C:\FASconCAT\_Folder> perl FASconCAT\_v1.0.pl -h <enter> ↔ help menu
- C:\FASconCAT\_Folder> perl FASconCAT\_v1.0.pl -s <enter> ↔ start FASconCAT under default

Table D.1: FASconCAT via command line: options

General options	Command	
Help menu	-help	
Preface	-a	
Start FASconCAT	-s	
Parameter options	Command	Default
Defined input files	-f	none
Dispense all infos	-i	none
PHYLIP output (strict)	-p	none
PHYLIP output (relaxed)	-p -p	none
NEXUS output (blank)	-n	none
NEXUS output (MrBayes)	-n -n	none

### D.2.3 Options

FASconCAT runs with several additional parameter options. Unknown commands are ignored.

*\* NOTE: Described commands are valid if the single command line is used. Working menu guided, type all options without “-”, for example “i” instead of “-i”.*

#### D.2.3.1 -f option

FASconCAT asks for user defined input files before concatenation starts. After starting the program via the “s” command, it will display a list of all files in FASTA (.fas), PHYLIP (.phy) and CLUSTAL (.aln) format which are located in the software folder with an associated list number (Table D.2). The user can define specific files for concatenation, regardless of the file format! Type the file associated number of selected files, separated by comma without blanks, in one row and press <enter>. If only one input file is chosen, FASconCAT converts it to the selected output format. By typing b and <enter>, FASconCAT will skip back to the main menu.

- COMMAND: 2,3,4 <enter> ↔ only the PHYLIP and CLUSTAL files will access the concatenation process

Table D.2: Example list of selectable files for specific file concatenation.

Listnumber	Filename
1	example_file_1.fas
2	example_file_2.phy
3	example_file_3.aln
4	example_file_4.aln
5	example_file_5.aln

### D.2.3.2 -i option

FASconCAT provides useful additional information about the supermatrix file and all single input files, e.g. base composition of nucleotide sequence files, the amount of gaps of each file or the amount of missing data. This option needs a little more computation time, depending on the data set. Therefore, this option is not included within the default setting. All additional information is listed in Table D.3.

### D.2.3.3 -n option

With the “-n” option, FASconCAT generates an additional NEXUS file (.nex) which can be directly loaded into PAUP, MrBayes or other NEXUS file using programs. With the “-n-n” option, FASconCAT generates not only a NEXUS file with implemented taxa sequence blocks, but rather an executable file for Bayesian analyses with the software MrBayes. For that reason we integrated a presetting of parameters which seems to be a good start point for Bayesian analyses. This can be easily changed manually by using any text editor. If a structure string in dot-bracket format is given while dots code unpaired (loop) positions and brackets code paired positions (stems), FASconCAT automatically compiles a partition set for MrBayes with separate charset for stem and loop regions. Table D.4 gives an overview of the integrated setup for MrBayes. To choose the MrBayes option via the FASconCAT menu, the “n” command has to be selected twice. If FASconCAT is started directly via command line, type “-n”, respectively “-n-n”.

*\* NOTE: Currently, partition blocks for different gene partitions are not implemented! For this purpose, the user have to modify the NEXUS file manually. However, it is planned to implement command blocks for a partitioned gene analysis in the next version.*

### D.2.3.4 -p option

With the “-p” option, FASconCAT additionally generates an output in PHYLIP (.phy) format. The PHYLIP format can be printed either strictly with non-interleaved sequences and restricted sequence names (up to 10 signs) or relaxed (no restriction in sign number for sequence names). To choose the strict PHYLIP option the “p” command has to be selected once, for the relaxed PHYLIP format twice.

Table D.3: List of default &amp; additional information within the .xls outputfile under the -i option

Default information	Supermatrix file	input files
Single fragment ranges	yes	no
Number of concatenated sequences per taxon	yes	no
<b>Additional information</b>		
Number of taxa	yes	yes
Number of sequence characters	yes	yes
Data type (nucleotide/amino acid)	yes	yes
Number of single nucleotide characters	yes	yes
Number of gaps	yes	yes
Number of ambiguity characters	yes	yes
Number of inserted replacement characters	yes	yes
Number of missing taxa per fragment	no	yes
Number of inserted replacement strings	yes	yes
Number of characters in total	yes	no
Number of amino acid characters	no	yes
Percent & total number of nucleotides	yes	no
Percent & total number of gaps	yes	no
Percent & total number of ambiguities	yes	no
Percent & total number of inserted replacements	yes	no
Percent & total number of loop characters	yes	yes
Percent & total number of stem characters	yes	yes
Percent & total number of missing data (?)	yes	yes
List of loop positions	yes	no
List of stem pairing positions	yes	no

\* **NOTE:** The number of ambiguities is set = 0 if nucleotide AND amino acid files are concatenated, since it is currently not possible to distinguish between amino acids and ambiguities.

Table D.4: Overview of all **MrBayes** setup parameters in the NEXUS output under the “-n-n” option. Structure partition parameters are only printed out by given structure information.

<b>MrBayes commands</b>	<b>Setup</b>
Number of generations	2000
Print frequency	100
Sample frequency	100
Number of chains	4
Save branch lengths	yes
Set autoclose	yes
No warnings	yes
Unlink statefrequency	all
tratio	all
Shape	all
Number of substitution	6
Rates	gamma
Sump burnin	20
Number of sump runs	2
Sumt burnin	20
Number of sumt runs	2
Inputfilename	FcC_smatrix.nex
<b>Structure partition</b>	
Set partition	looms
partition looms	2: loops, stems
lset 1	nucmodel= 4by4
lset 2	nucmodel= dublet

If FASconCAT is started directly via command line, type “-p”, respectively “-p-p”.

*\* NOTE: Bioedit can only properly open PHYLIP files where a) no or maximally one blank is in front of the number of sequences (first line of a PHYLIP file) and where b) all sequence names have exactly 10 signs (including blanks). It is not possible to edit a PHYLIP file with short (< 10 signs) or with relaxed sequence names (> 10 signs). See section D.4.*

## D.3 Internals

### D.3.1 Input/Output

FASconCAT is able to import three different file formats. The number and formats of the output files depend on chosen parameter settings. Table D.5 gives a summary of possible input and output formats.

Table D.5: Overview of possible input and output formats under given parameter options.

<u>Input format</u>	<u>Ending</u>	
FASTA	.fas/.fasta	
PHYLIP	.phy	
CLUSTAL	.aln	
<u>Chosen options</u>	<u>Output files</u>	<u>Contens</u>
all options	FcC_smatrix.fas	Supermatrix in FASTA format
-p or -p-p	FcC_smatrix.phy	Supermatrix in PHYLIP format
-n or -n-n	FcC_smatrix.nex	Supermatrix in NEXUS format
all options	FcC_info.xls	Concatenation information (restricted)
-i	FcC_info.xls	Concatenation information
-i	FcC_structure.txt	Structure information *

(\* structure strings were present in input files)

### D.3.2 Computation time

The computation time of FASconCAT depends on the data amount and on chosen options. Even for phylogenomic data sets, the computation time will be in acceptable manner on a normal desktop computer. Providing an additional PHYLIP output does not prolong the computation time. The most time consuming step is the compilation of NEXUS output files. Choosing all possible information (“-i”) is only little more time expensive than the default setup. Following examples give an impression about the computation time with different kinds of data amount and usage options. We simulated two series of tests using INDELIBLE, with different numbers of sequences.

The first test includes 26 nucleotide sequences, the second 108. Per test series, seven concatenation processes were conducted. They differed in the length of used data sets (100 - 100,000 bp). This was repeated for five different output options. Per concatenation process, ten data sets with identical alignment length were used. The computation time was measured for each single concatenation process and output option (Table D.6 and D.7).

Table D.6: Computation time of FASconCAT considering different sequence lengths and output options for 26 sequences per data set (test 1).

	Distinct concatenation processes						
	10	10	10	10	10	10	10
N data sets	10	10	10	10	10	10	10
Single lengths [bp]	100	500	1,000	10,000	25,000	50,000	100,000
Supermatrix [bp]	1,000	5,000	10,000	100,000	250,000	500,000	1,000,000
Output options	Computation time [sec]						
Default	0.2	0.1	0.1	0.5	1.2	2.4	4.8
PHYLIP	0.1	0.1	0.2	0.6	1.2	2.4	4.9
Default + all info	0.2	0.3	0.5	4	9.7	19.7	40.1
PHYLIP + all info	0.1	0.3	0.5	3.9	9.7	19.8	40.1
NEXUS	0.2	0.4	0.9	16.1	75.8	281.9	1321.6

Table D.7: Computation time of FASconCAT considering different sequence lengths and output options for 108 sequences per data set (test 2).

	Distinct concatenation processes						
	10	10	10	10	10	10	10
N data sets	10	10	10	10	10	10	10
Single lengths [bp]	100	500	1,000	10,000	25,000	50,000	100,000
Supermatrix [bp]	1,000	5,000	10,000	100,000	250,000	500,000	1,000,000
Output options	Computation time [sec]						
Default	0.3	0.4	0.5	2.3	5.5	11.3	21.4
PHYLIP	0.3	0.4	0.5	2.3	5.5	11	21.9
Default + all info	0.5	1.1	1.9	16.6	42.8	89.1	180.5
PHYLIP + all info	0.4	1.1	1.9	16.8	43.2	88.7	156.8
NEXUS	0.5	1.7	3.4	69.3	320.5	1172.5	5583.4



### D.3.3 Error reports

FASconCAT checks each input file according to correct format and forbidden sequence and structure characters. This subsection gives a short explanation for possible reasons to all implemented error reports.

*\* NOTE: Each error allocates FASconCAT to stop all running processes; FASconCAT will abort with an specified error message.*

#### D.3.3.1 Taxon in *filename.fas* not in FASTA format!

The *filename.fas* file is not in a FASTA format typical manner. FASconCAT reads sequences of FASTA files, either if they are in one line, or with line interruptions (blocks). Sequence names must be in one line and start with an “>”. Each line must end with a line break. Table D.8 gives an example of both acceptable FASTA formats.

Table D.8: Known FASTA formats in non-interleaved (format 1) and interleaved format (format 2).

---

<b>FASTA format 1</b>
>Name_sequence_1
AGCTCCCGTCCTTTG-AGA-GTGCCTTTCCT
>Name_sequence_2
AGCTCCGGCCCTTTG-AGA-GTGCCTTTCCT
>Name_sequence_n
AGCTCCCGTCCTTTGGAGAGGTGCCTTTCCT
<b>FASTA format 2</b>
>Name_sequence_1
AGCTGTCCTTTCTTG-AGA-GTGCCTTTCCT
GGGGCCCTTTC-GGTTTTCCCCGCCTTTCCT
>Name_sequence_n
AGCTGTCCTTTCTTGCAGACGTGCCTTTCCT
GGGGCTTCAAGTTTTCCCCGGGCCTTTCCT

---

#### D.3.3.2 *filename.aln* is not a CLUSTAL format!

The *filename.aln* file is not in a CLUSTAL format typical manner. Each line must end with a line break. Table D.9 shows a typical CLUSTAL format.

Table D.9: Example of a CLUSTAL formatted input file.

---

**CLUSTAL format**

---

```

CLUSTAL X (1.81) multiple sequence alignment
<line break>
<line break>
Name_sequence_1   AGGGCCCTTGCGCTTGCTTCC
Name_sequence_2   AGGGCCCTTGCGCCTTGCTTCC
Name_sequence_n   AGGGCCCTTGCGCCGGCTTCC
<line break>
<line break>
Name_sequence_1   ATTTCCCTTGGGCTTGCTTCC
Name_sequence_2   ATTTCCCTTGGGCCTTGCTTCC
Name_sequence_n   ATCTCCCTTGGGCCGGCTTCC

```

---

### D.3.3.3 *filename.phy* is not a PHYLIP format!

The *filename.phy* file is not in a PHYLIP format typical manner. Each line must end with a line break. Table D.10 shows a typical PHYLIP file in interleaved format with restricted sequence names (10 signs at maximum).

Table D.10: Example of a interleaved PHYLIP formatted input file.

---

**PHYLIP format (interleaved)**

---

```

6 40
Name_sequence_1   AGGGCCCTTG   CGCTTGGCCC
Name_sequence_2   AGGGCCCTTG   CGCCTCCCCC
Name_sequence_n   AGGGCCCTTG   CGCCGCCCGG
<line break>
                   ATTTCCCTTG   GGCTTCCCCC
                   ATTTCCCTTG   GGGGGCCTCC
                   ATCTCCCTTG   GGCCGGGGGC

```

---

### D.3.3.4 Unknown input format of *filename!*

Something in your input file is completely wrong. Please check your input file for correct format.

### D.3.3.5 Sequence name missing in *filename!*

Maybe you have forgotten the sequence name, the “>” in front of the sequence name, or your FASTA format is completely wrong. See also Table D.8 for known FASTA formats.

**D.3.3.6 Sequence missing in *filename*!**

Either you have forgotten the sequence or an additional line-break in your FASTA file, or your FASTA format is completely wrong. See Table D.8 for known FASTA formats.

**D.3.3.7 Sequence name *sequence\_name* in *filename* involves forbidden signs!**

Sequence names may only include alphanumeric signs, underscores ( `_` ) and blanks, everything else is not allowed. If the sequence names are correct, check the input format in common.

**D.3.3.8 Sequences of *filename* have no equal length!**

FASconCAT allows sequences within the same input file only if they have equal length.

**D.3.3.9 Multiple sequence names of *sequence\_name* in *filename*!**

Identical sequence names are not allowed in the same input file, because FASconCAT concatenates sequences on the basis of them. Two equal names in one file cannot be assigned correctly.

**D.3.3.10 Sequence of *filename* involves forbidden signs in *sequence\_name*!**

Ambiguities and indels are recognized. Any other sign in sequences, except for those covered by the universal DNA/RNA or amino acid code, is not allowed. If the sequence signs are correct, check the input format in common.

**D.3.3.11 *filename* involves multiple structure sequences!**

Multiple structure strings in one input file are not allowed. FASconCAT can concatenate only one structure string per file, which is sufficient for most phylogenetic analyses.

**D.3.3.12 Additional structure sequence of *sequence\_name* in *filename* not allowed!**

FASconCAT can handle only one structure strings per file. For that reason, single structure strings must have identical names. Maybe your files have one structure string, but the names are not identical. Check the names of the structure strings.

## D.4 Important Notes

Please recognize following notes. Most points will be implemented into FASconCAT in the future.

- Since the current version cannot distinguish between ambiguities and amino acids, the amount of ambiguities is not calculated, if nucleotide AND amino acid alignments are concatenated. Then, the amount of ambiguities is set = 0%.
- Currently, command blocks for a partitioned gene analyses are not implemented. Only a command block for a partitioned analyses of stems and loops is implemented. The implementation of command blocks for a partitioned gene analysis is planned, currently the user has to manually modify the Nexus file.
- The read of input alignments in NEXUS format is currently not implemented, but planned in the near future.
- **Bioedit** can only properly open PHYLIP files if a) no or maximally one blank is in front of the number of sequences (first line of a PHYLIP file) and where b) all sequence names have exactly 10 signs (including blanks). It is not possible to edit a PHYLIP file with short (< 10 signs) or relaxed sequence names (> 10 signs). If more than **one blank is in front of the number of sequences**, **Bioedit** gives an error prompt (Unknown file format). If sequence names are < 10 signs, a part of the sequence is written into the sequence name! If one (or more) sequence name is > 10 signs, **Bioedit** crashes. Therefore, the PHYLIP output of FASconCAT a) has no blank or any other sign in front of the sequence number! (Several editors might include more blanks or a tab in front of the sequence number.) b) FASconCAT fills up sequence names with blanks to exactly 10 signs. Using option (-pp), FASconCAT fills up all sequence names with blanks which are shorter than the longest sequence name. To watch the supermatrix in the relaxed PHYLIP format, use e.g.:

– **Seaview**

<http://pbil.univ-lyon1.fr/software/seaview.html>

Seaview can display PHYLIP files, regardless of sequence names length (shorter or longer than 10 signs), without any problem.

– **mesquite**

<http://mesquiteproject.org/mesquite/mesquite.html>

– **MEGA**

<http://www.megasoftware.net/>

– **geneious**

<http://www.geneious.com/>

A test version if for free available.

Relaxed sequence names (length not restricted) can be handled properly by e.g. the Maximum likelihood software RAxML.

- The output of the information (.xls) file is programmed along to English language standards. Therefore, users with systems or programs in German, should open this in an appropriate editor and replace '.' (dot) by a ',' (comma) (change e.g. 58.5 to 58,5). Otherwise, numbers with decimals might be wrongly displayed. Alternatively, users can edit the software preferences and change the language into English.

## D.5 License/Help-Desk/Citation

FASconCAT v1.0 was developed by Patrick Kück in 2010. It is implemented in Perl and freely available from <http://fasconcat.zfmk.de>. It can be distributed and/or modified under the terms of the GNU General Public License as published by the Free Software Foundation; either 2 of the license, or (at your option) any later version.

This program is distributed with the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

If you have any problems, error-reports or other questions about FASconCAT, feel free and write an email to [fasconcat@web.de](mailto:fasconcat@web.de) which is the official help desk email account for FASconCAT.

For further free downloadable programs from our institute visit:  
<http://software.zfmk.de>.

If you use FASconCAT please cite:

Kück P, Meusemann K (2010) FASconCAT: Convenient handling of data matrices. *Mol Phylogenet Evol* 56:1115-1118.



## E.2 Usage/Options

To start ALICUT, open the terminal window of your running system. Navigate through your directory path to the folder where ALICUT is located. Type the name of your ALICUT version, followed either by a blank and your demand options in one row to start ALICUT directly or followed by pressing enter to get into the ALICUT menu. Notice that all input files have to be located in the ALICUT including folder. To execute ALICUT a Perl interpreter must be installed on the current run system. Linux and Mac systems do normally not need a subsequent installation because the interpreter is a standard tool included in that systems in advance. Unfortunately, Windows users have to install a Perl interpreter ex post. We would recommend the ActivePerl interpreter which can be downloaded for free under:

- <http://activeperl.softonic.de/>

### E.2.1 Start ALICUT via menu

#### E.2.1.1 Open the menu under Windows

Open a prompt (DOS) terminal on your Windows system and navigate to the folder where ALICUT and the ALISCORE input and output files are located `<cd your_path...>`. Then open ALICUT:

- `C:\ALICUT_Folder> ALICUT_v1.0.pl <enter>`

#### E.2.1.2 Open the menu under Linux/Mac

Open a terminal and navigate to the folder where ALICUT and the ALISCORE input and output files are located `<cd your_path...>`. Then open ALICUT:

- `user@user:~/ALICUT_Folder> perl ALICUT_v1.0.pl <enter>`

#### E.2.1.3 Menu handling

The main menu of ALICUT (Fig. E.2) constitutes all possible options and their associated commands for adjustment.

To change the default parameter setting type the option associated command into the command line and press `<enter>`. After finishing parameter configuration FASconCAT can be started by typing “s” and pressing `<enter>`. For getting help type “h” and press `<enter>`, to return to FASconCAT type press `<enter>`, to quit the program type “q” and press `<enter>`.

### E.2.2 Start ALICUT via single command line

ALICUT can directly started by command line commands in one row which simplifies the implementation of ALICUT into complex process pipelines. Move through your directory path to the folder where ALICUT and your ALISCORE input and

```
preacher@preacher: /media/Wylie Times/Institut/Perl/skripte/ALICUT/ALICUT
preacher@preacher:~$ cd '/media/Wylie Times/Institut/Perl/skripte/ALICUT/ALICUT'
preacher@preacher:/media/Wylie Times/Institut/Perl/skripte/ALICUT$ perl ALICUT_v1.0.pl

-----
                Welcome to ALICUT V1.0 !
                a Perlscript to cut ALISCORE identified RSS
                written by Patrick Kueck (ZFMK, Bonn)
-----

START ALICUT:      type <s> <return>
QUIT ALICUT:      type <q> <return>
REMAIN STEMS:     type <r> <return>
HELP:             type <h> <return>
PREFACE:          type <p> <return>
ERROR REPORT:     type <x> <RETURN>

(X = Error associated number: 1 or 2)

-----

COMMAND:          █
```

Figure E.2: Menu of ALICUT.

output files are located and type the name of the ALICUT version, followed by a blank and the demand options with a minus (-) sign in front of each. Then press <enter>. Make sure you write the input options correctly. Otherwise ALICUT will not start working but instead open the menu. An overview of all ALICUT options is shown in Table E.1.

- `user@user:~/ALICUT_Folder> perl ALICUT_v1.0.pl -h <enter>` ↔ help menu
- `user@user:~/ALICUT_Folder> perl ALICUT_v1.0.pl -s <enter>` ↔ start under default

Table E.1: Overview of option codes via single command line start

Info options	Command	
Help menu	-h	
Preface	-p	
Start	-s	
Parameter option	Default	
Remain stem positions	-f	none
Exclude stem positions	-r -r	yes

### E.2.2.1 -r Option

If structure sequences are implemented into the ALISCORE masked FASTA infiles, ALICUT can automatically remain randomized stem positions. To remain random-



Table E.2: Example of the additional information file “ALICUT\_info.xls”.

Used List File	Used FASTA File	bp before	bp after	Rem. bp %
Hex_16S_LIST_random.txt	Hex_16S.fas	487	394	80,9
Hex_28S_LIST_random.txt	Hex_28S.fas	1221	1056	86,5
Hex_COI_LIST_random.txt	Hex_COI.fas	436	436	100

ized stem positions use the “-r” option via single command line or type “r” <enter> via menu options. Under default, ALICUT replaces the corresponding character positions of randomized stem positions by dots if these characters are identified as non-randomised.

*\* NOTE: Only one structure sequence per FASTA input file is allowed.*

### E.2.3 Additional Information files

ALICUT provides useful additional information of all single restricted FASTA files including percentages of remaining positions (“ALICUT\_info.xls”) and structure sequence information (“ALICUT\_Struc\_info.txt”). Table E.2 and E.3 gives an example of the construction of the additional information files.

## E.3 Input/Output

ALICUT is able to import FASTA files and ALISCORE “LIST” outfiles. The ALISCORE inputfile(s) and ALISCORE “List” outputfile(s) must be together in the same folder as ALICUT. The ALISCORE “List” outfile(s) must contain the ALISCORE identified randomized sequence similarity (RSS) positions in one single line, separated by one whitespace sign. ALICUT can handle unlimited FASTA files in one single run. ALICUT reads the FASTA infile(s) and ALISCORE “List” outfile(s), excludes the ALISCORE identified RSS positions listed, and saves the restricted sequences as a new FASTA file marked by the prefix “ALICUT\_”. An important condition for the restriction of the masked FASTA infile(s) is, that the ALISCORE “List” outfile(s) have the corresponding FASTA infile name as prefix (see for example Tab. E.2). In the best case, the FASTA infile(s) and ALISCORE “List” outfile(s) are not changed since the execution of ALISCORE. Table E.4 and E.5 give a summary of correct input formats.

*\* NOTE: If two “List” files are generated from an equal named FASTA file (e.g. the first “List” file includes ALSICORE identified RSS along a NJ tree while the second “List” file includes identified RSS positions identified by all single comparisons) the first ALICUT outfile will be overwritten by the second one.*

Table E.3: Example of the additional information file “ALICUT\_Struc\_info.txt”.

---

**Original structure information identified in testfile.fas**

---

-Number of characters:	19
-Number of single loop characters:	9
-Number of paired stem characters:	5
-Paired stem positions:	1:19
	2:18
	3:16
	6:14
	7:13
-Loop positions:	4
	5
	8
	9
	10
	11
	12
	15
	17

---

Table E.4: Known FASTA formats in non-interleaved (format 1) and interleaved format (format 2).

---

**FASTA format 1**

>Name\_sequence\_1  
AGCTCCCGTCCTTTG-AGA-GTGCCTTTTCCT

>Name\_sequence\_2  
AGCTCCGGCCCTTTG-AGA-GTGCCTTTTCCT

>Name\_sequence\_n  
AGCTCCCGTCCTTTGGAGAGGTGCCTTTTCCT

**FASTA format 2**

>Name\_sequence\_1  
AGCTGTCCTTTCTTG-AGA-GTGCCTTTTCCT  
GGGGCCCTTTC-GGTTTTCCCCGCCTTTTCCT

>Name\_sequence\_n  
AGCTGTCCTTTCTTGCAGACGTGTCCTTTTCCT  
GGGGCTTCAAGTTTTCCCCGGGTCCTTTTCCT

---

Table E.5: Known ALISCORE “List” outfile” format.

---

1	12	13	14	15	22	23	24	25	26	30	...	1000
---	----	----	----	----	----	----	----	----	----	----	-----	------

---

## E.4 License/Help-Desk/Citation

ALICUT v1.0 is written/developed in Perl by Patrick Kück in 2009. It is implemented in Perl and a free software. It can be distributed and/or modified under the terms of the GNU General Public License as published by the Free Software Foundation; either 2 of the license, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

If you have any problems, error-reports or other questions about ALICUT feel free and write an email to [ali\\_score@web.de](mailto:ali_score@web.de) which is the official help desk email account for the software. For further free downloadable programs from our institute visit:

<http://software.zfmk.de>.

If you use ALICUT please cite:

Kück P., ALICUT, a Perlscript which cuts ALISCORE identified RSS. Department of Bioinformatics, Zoologisches Forschungsmuseum A. Koenig (ZFMK), Bonn, Germany, version 1.0 edition [111].

## E.5 Copyright

© by Patrick Kück, October 2009

# Short Documentation ESTa

---

## F.1 General Information

“ESTa.pl” is written in PERL and downloads the actual EST-summary list from the NCBI-Genbank ([http://www.ncbi.nlm.nih.gov/dbEST/dbEST\\_summary.html](http://www.ncbi.nlm.nih.gov/dbEST/dbEST_summary.html)) for taxa with more than 1.000 EST database entries. The actual list will be stored in a EST database listfile (“EST-request\_(NCBI).txt”). Database entries of an earlier downloaded database listfile are replaced if an earlier version and the ESTa perlscript are placed in the same folder. In addition to the “EST-request\_(NCBI).txt” file, new database entries are printed out in a separate text file “New\_EST\_entries.txt”, too.

## F.2 General Usage

To start ESTa open the terminal window of your running system, navigate through your directory path to the folder where ESTa is located, and execute ESTa by typing the name of the ESTa version followed by <enter>. Be sure that an internet connection exist.

- C:\ESTa\_Folder> perl ESTa\_v0.1.beta.pl <enter> ↔ Windows
- user@user:~/ESTa\_Folder> perl ESTa\_v0.1.beta.pl <enter> ↔ Linux/Mac

## F.3 Output-files

### F.3.1 “EST-request\_(NCBI).txt”

- The file “EST-request\_(NCBI).txt” contains the following structure:  
“Taxon” <space> => <space> “Number of EST entries found in the NCBI database <newline>”
- “EST-request\_(NCBI).txt” lists all taxa with more than 1.000 entries in the NCBI-database
- The “EST-request\_(NCBI).txt” entries are listed alphabetically by taxon name

**F.3.2 “New\_EST\_entries.txt”**

- The file “New\_EST\_entries.txt” contains the following structure:  
“Taxon” <space> => <space> “Number of EST entries found in the NCBI database <newline>”
- The “New\_EST\_entries.txt” file lists all EST entries which which have changed since the last database update
- The “New\_EST\_entries.txt” entries are listed alphabetically by taxon name

# List of Abbreviations

Table G.1: List of abbreviations used in this thesis

Abbreviation	Definition
$\alpha$	Shape Parameter of the Gamma Distribution Model
$\Gamma$	Gamma Distribution Model
$\rho$	Proportion of Invariant Sites
12S	Small Mitochondrial Ribosomal Subunit
16S	Large Mitochondrial Ribosomal Subunit
AA	Amino Acid Sequences
Al	ALISCOPE (Alignment Masking Software)
AU test	Approximately Unbiased Test
ASRV	Among-Site Rate Variation
BI1	Branch Increase 1
BI2	Branch Increase 2
bs	Bootstrap Support Value
bpp	Bayesian Posterior Probability Value
bp	Base Positions
c	Relative Substitution Rate Categories
COI	Cytochrom Oxidase I
COII	Cytochrom Oxidase II
COIII	Cytochrom Oxidase III
Cytb	Cytochrom b
dbEST	Expressed Sequence Tags Database
EST	Expressed Sequence Tags
EF1aF2	Elongation Factor 1-alpha F2
Fig	Figure
GB	Giga byte
Gb	GBLOCKS (Alignment Masking Software)
GHz	Gigahertz
GUI	Graphical User Interface
$H_0$	Null Hypothesis
HPC	High performance computing
I	Invariant Sites Model
JC	Jukes Cantor
ML	Maximum likelihood
MB	Megabyte
MSA	Multiple Sequence Alignment
LBA	Long Branch Attraction
LiB	Long Internal Branch

*Continued on next page*

Table G.1 – continued from previous page

Abbreviation	Definition
LtB	Long Terminal Branch
MRE	Extended Majority-Rule Consensus Tree
mtI	Mitochondrial Data Set 1
mtII	Mitochondrial Data Set 2
N	Number
NC	Nucleotid Sequences
NCBI	National Center for Biotechnology Information
ND1	NADH Dehydrogenase Subunit 1
ND2	NADH Dehydrogenase Subunit 2
ND3	NADH Dehydrogenase Subunit 3
ND4	NADH Dehydrogenase Subunit 4
ND4L	NADH Dehydrogenase Subunit 4L
ND5	NADH Dehydrogenase Subunit 5
ND6	NADH Dehydrogenase Subunit 6
RB	Remaining Branches
RS	Resolution Score
RNA	Ribonucleine acid
rRNA	ribosomal RNA
sec	Seconds
SiB	Small internal Branch
StB	Small Terminal Branch
Tab	Table
Un	Unmasked
ZFMK	Zoologisches Forschungsmuseum A. Koenig, Bonn

# List of Electronic Supplementary Files

---

- Electronic supplementary file ES1 — Detailed analytical results (chapter 2)
- Electronic supplementary file ES2 — Presentation of the ALIS-CORE algorithm and the results (chapter 2)
- Electronic supplementary file ES3 — Publication (Kück et al. [35]) (chapter 2)
- Electronic supplementary file ES4 — Unmasked alignment file (chapter 3)
- Electronic supplementary file ES5 — Masked alignment file for the masked-unpartitioned analyses (chapter 3)
- Electronic supplementary file ES6 — Masked alignment file for the masked-partitioned analyses (chapter 3)
- Electronic supplementary file ES7 — Character partition file (chapter 3)
- Electronic supplementary file ES8 — Publication (Kück et al. [123]) (chapter 3)
- Electronic supplementary file ES9 — Detailed results of all ML tree reconstructions (chapter 5)
- Electronic supplementary file ES10 — Detailed results of investigated likelihood scores (chapter 5)
- Electronic supplementary file ES11 — Detailed results of model parameter estimates (chapter 5)
- Electronic supplementary file ES12 — Presentation of the results (chapter 5)
- Electronic supplementary file ES13 — Perlscript FASconCAT v1.0 (chapter 6)



- Electronic supplementary file ES14 — Presentation of the Software FASconCAT v1.0 (chapter 6)
- Electronic supplementary file ES15 — Publication of FASconCAT (Kück & Meusemann [112]) (chapter 6)
- Electronic supplementary file ES16 — Perlscript ALICUT v1.0.pl [111] (chapter 6)
- Electronic supplementary file ES17 — Presentation of ALICUT v1.0 (chapter 6)
- Electronic supplementary file ES18 — Perlscript ESTa v0.1.beta (chapter 6)

# Summary

---

Considering the final goal of every phylogenetic analysis, the reconstruction of taxon relationships from underlying data, little attention has been paid to the role of alignment accuracy and its impact on tree reconstruction. Multiple sequence alignments are statements of primary homology in phylogenetic analyses. In the first step of the primary homology assessment, similar sequences are identified through sequence comparisons by alignment algorithms like BLAST (Basic Local Alignment Search Tool), while subsequently efficient alignment algorithms are used to allocate positional similarity among sequences. Unfortunately, alignment algorithms can not differentiate between positional similarity of sequences and evolutionary homology, which can lead to incorrectly aligned sequence positions due to random similarity among sequences. Due to the dependence of tree reconstruction on the primary homology assessment, the influence of incorrect alignment sections can negatively influence phylogenetic reconstructions and lead to defective estimation of substitution model parameters, especially if data sets are very large. The degree of alignment accuracy is strongly influenced by the chosen alignment algorithm and its parameter settings. Ambiguously aligned sequence sections and random sequence similarity can negatively influence phylogenetic reconstructions and lead to defective estimation of substitution model parameters.

Alignment masking approaches are methods which detect and remove erroneously aligned sections before tree reconstruction. The effect of two masking methods on alignment quality and tree reconstruction is described in chapter 2 of my PhD thesis. This section gives furthermore the first comprehensive characterisation of the most recent amino-acid masking algorithm implemented in ALISCORE, one of the two masking approaches tested in this chapter. Another example about the positive impact of alignment masking on data quality is given in chapter 3 which describes a re-analysis of a previously published data set to resolve the Ant Tree of Life. The re-analysis is coupled with parametric alignment masking and thoroughly performed phylogenetic analyses which comes to different conclusions than the previously published study. While masking methods are commonly efficient in detecting ambiguously aligned sequence blocks, all methods more or less lack the ability to detect heterogeneous sequence divergence within sequence alignments. This is a main disadvantage of masking approaches, because undetected heterogeneous sequence divergence can result in a strong bias in tree reconstructions. Chapter 4 gives a detailed description of a new developed algorithm and the possibility of tagging branches as an indirect estimation of reliability of a subset of possible splits guided by a topology. The performance of the new algorithm was tested on simulated and

empirical data.

Considering the tree reconstruction process, the first task is the choice of an appropriate tree reconstruction method. Examining theoretical studies and comparative tests Maximum Likelihood turns out as the first choice for phylogenetic tree reconstructions. Chapter 5 shows that the success of Maximum Likelihood depends not only on the degree of alignment quality, but also on the relation of branch length differences of underlying topologies. The study of chapter 5 tested the robustness of Maximum Likelihood towards different classes of long branch effects in multiple taxon topologies by using simulated fixed data sets under two different 11-taxon trees and a broad range of different branch length conditions with sequence alignments of different length.

The realization of the studies described in chapter 2–5 would not have been possible without the development of numerous scripts. Some of the most important scripts and pipelines which have been written for the accomplishment of this thesis or which have been written for other studies are listed and described in chapter 6.

APPENDIX J

# Erklärung

---

Ich versichere, dass ich diese Arbeit selbständig verfasst, keine anderen Quellen und Hilfsmittel als die angegebenen benutzt und die Stellen der Arbeit, die anderen Werken dem Wortlaut oder Sinn nach entnommen sind, kenntlich gemacht habe.

Diese Arbeit hat in dieser oder ähnlichen Form keiner anderen Prüfungsbehörde vorgelegen.

December 15, 2011



