Diss. ETH No. 18177

Comparative Genomics Using Pairwise Evolutionary Distances

A dissertation submitted to the
ETH Zurich

for the degree of
Doctor of Sciences

presented by
Christophe Dessimoz
Dipl. Natw. ETH

born on 21 November 1980
citizen of Conthey (VS) and Vétroz (VS), Switzerland

accepted on the recommendation of
Prof. Dr. Gaston H. Gonnet, examiner
Prof. Dr. Amos Bairoch, co-examiner
Prof. Dr. Niko Beerenwinkel, co-examiner
Prof. Dr. Jörg Stelling, co-examiner
Prof. Dr. Martin Vingron, co-examiner

2009

Abstract

Comparative genomics is the study of genome structure, function, and evolution across species enabled by the recent availability of complete genome sequences. In this context, and assuming a common origin, one of the fundamental notion is the degree of evolutionary relatedness among sequences. This is usually expressed either as a tree structure, or as the matrix of distances between pairs of sequences. In this thesis, we explore and develop comparative genomics methods that rely on pairwise distances.

Part one of the thesis reviews and extends methods to estimate and combine distances between pairs of biological sequences. To estimate distances, the homologous characters must be identified; thus we begin with an evaluation of sequence alignment methods using simulation and real biological sequences. The main other contribution of part one consists in two methods to estimate the covariance of pairwise distances obtained from independent alignments (as can be the case in large-scale studies). The first method approximates the variance of the difference between two distances under empirical amino-acid substitution models. This is equivalent to computing the covariance between two distances that involve a common sequence. The second method is more general in the sense that it works with arbitrary Markovian models, and it also estimates the covariance of distances involving four distinct sequences, thereby allowing for the computation of full covariance matrices.

Part two applies these tools to large-scale comparative genomics analyses. The first application is an algorithm to detect differential gene losses, which often lead to misclassifications in orthologs detection methods based on genome-specific best hits, and in particular in the COGs database. Using it, we show that about a third of all COG groups include non-orthologs. These results motivate the second application, our own orthology inference effort, OMA. Besides the ability to detect differential gene losses, OMA has several other distinctive features: it computes evolutionary distances from pairwise maximum likelihood alignments, it uses confidence intervals to account for statistical inference uncertainty and to allow for many-to-many orthology, and clusters groups of orthologs using an edge-weighted clique algorithm. Our results are validated and compared with those resulting from 11 other projects and methods. The results of OMA are among the best in the phylogenetic tests. In functional tests, OMA, strict by design, performs comparatively well if high functional specificity is required. In terms of size, OMA is by a wide margin the largest effort of orthology inference; as of August 2008, we had performed all-against-all alignments of 670 complete genomes and had built orthologous matrices for the major clades of genomes. The third application of part two is an algorithm to detect lateral gene transfer based on pairwise distances. At its core, it computes a likelihood ratio between hypotheses of LGT and no LGT. As opposed to explicit phylogenetic LGT detection approaches, it avoids the high computational cost and

pitfalls associated with gene tree inference and reconciliation, while maintaining the high level of characterization (species involved in LGT, direction, distance to the LGT event in the past) associated with such methods. The results and validation of the algorithm, both through simulated and real data, show that the method outperforms common LGT detection approaches.

Résumé

Le génome est l'ensemble du matériel génétique d'un organisme encodé dans son ADN, et la génomique comparative est l'étude de la structure, de la fonction et de l'évolution des génomes au travers des espèces, rendue possible par le récent séquençage de génomes entiers. Dans ce contexte, et en supposant une origine commune à toutes les espèces, l'une des notions fondamentales c'est le degré de parenté entre espèces. Celle-ci s'exprime généralement soit sous forme d'arbre phylogénétique, soit sous forme de matrice de distances entre paires de séquences. Dans la présente thèse, nous explorons et développons particulièrement les méthodes de génomique comparative qui se basent sur les distances entre paires. Nous le faisons en deux parties principales.

Dans la première partie du mémoire, nous passons en revue et développons les méthodes actuelles pour l'estimation de distances entre paires de séquences biologiques. Afin d'estimer ces distances, les caractères homologues doivent au préalable être identifiés; nous commençons donc par une évaluation des méthodes d'alignement de séquence, d'abord en simulation puis par référence à des séquences biologiques réelles. Notre principale autre contribution dans cette première partie comprend deux méthodes pour estimer la covariance de distances obtenues en partant de paires alignées indépendamment les unes des autres (comme c'est souvent le cas dans les études génomiques à large échelle). La première méthode calcule la variance de la différence entre deux telles distances pour des modèles empiriques de substitution de caractères. Ceci équivaut à l'estimation de la covariance entre distances qui impliquent une séquence commune. La deuxième méthode est plus générale dans le sens qu'elle fonctionne pour tous les modèles Markoviens; elle permet aussi l'estimation de covariance entre deux distances qui impliquent quatre séquences distinctes, et peut ainsi s'employer pour calculer des matrices de covariance.

La seconde partie de la thèse consiste à appliquer ces méthodes dans le cadre d'études de génomique comparative à large échelle. La première application est un algorithme pour détecter des cas de perte différentielle de gènes, qui résultent souvent en des erreurs de classification au niveau de la prédiction d'orthologues, comme en témoigne le taux élevé d'erreurs dans la base de donnée COGs. En effet, nous démontrons la présence de séquences non-orthologues dans environ un tiers des groupes COG. Ce résultat motive la deuxième application, c'est-à-dire le développement de notre propre système d'identification d'orthologues, OMA. Mis à part sa capacité à détecter les pertes différentielles de gènes, OMA innove à plusieurs autres niveaux: l'algorithme se base sur les distances évolutives entre paires de séquences au lieu de scores souvent autrement utilisés par ailleurs; il emploie des intervalles de confiance pour tenir compte de l'incertitude statistique sur les distances estimées et pour permettre l'inférence de relations orthologues n-à-n ("many-to-many"); le regroupement d'orthologues se réalise par une approche de clique maximale pondérée. Nos résultats sont validés et comparés

avec ceux de 11 autres projets et méthodes. Les résultats d'OMA sont parmi les meilleurs dans les tests phylogénétiques. En terme de conservation de fonction, OMA, strict de par sa conception, assure de bonnes performances dans les cas où une spécificité fonctionnelle élevée est requise. En terme de taille, OMA recouvre de loin le plus grand nombre de génomes séquencés: au mois d'août 2008, nous avions aligné les gènes de 670 génomes complets, et construit des matrices d'orthologues pour les toutes les clades majeures. Enfin, la troisième application de cette deuxième partie est un algorithme pour détecter des transferts de gènes latéraux ("LGT") sur la base de distances des paires. En bref, il calcule un rapport de vraisemblance entre les hypothèses "transfert" et "pas de transfert". Contrairement aux approches phylogénétiques explicites, il évite les coûts de calcul élevés et les difficultés associées à l'inférence et la réconciliation d'arbres de gènes, tout en gardant leurs avantages en terme de caractérisation du cas en question (espèces impliquées dans le LGT, direction, distance évolutive au transfert). Les résultats et la validation de l'algorithme, au travers de données simulées et réelles, démontrent que cette méthode surpasse les méthodes standards de détection de LGT.

Acknowledgments

First, I would like to express my profound gratitude toward Prof. Gaston H. Gonnet, my thesis supervisor. He hired me. He provided me with invaluable ideas and constant support. He created many exciting opportunities along the way, but also gave me freedom and trust to pursue my own projects. I could not have hoped for a better mentor.

I am also greatly indebted to my coauthors. Like much of today's Science, this work is the result of collaborative efforts. Without their contributions, this thesis would be very different, if possible at all. I had the most intense exchanges with my fellow PhD students. In particular, I was fortunate to start my PhD together with Manuel Gil, Adrian Schneider, and Daniel Margadant. We learned from each other and together we grew. I also very much appreciated working with Brigitte Boeckmann from the Swiss-Prot group, whose patient data dissection and constructive criticism have been enormous assets to developing algorithms of biological relevance.

I am very grateful to the undergraduate and graduate students I had the privilege to mentor and collaborate with. I have had particularily rewarding relationships with Christian Ledergerber and Adrian Altenhoff. Thank you for your confidence and diligence, and keep up with the great work!

Looking back, there was not a single day when I was not looking forward to go to work; that says a lot about the friendly, fun and uncomplicated atmosphere around me at work. I would like to express warmest thanks to my colleagues and friends in the CBRG group and at ETH in general, including Adam, Alex, Bettina, Daniel, Dirk, Gina, Hervé, Marco, Maria, Markus, Pedro, and Peter. The sacred coffee breaks, the Friday lunch expeditions, the after-work beers; you made it all worthwhile!

Completing this thesis also gives me the chance to show my appreciation to my previous mentors, who accompanied me along my first steps in research: Prof. Bernard Witholt, Prof. Matthew R. Parsek, Prof. Vassily Hatzimanikatis, Prof. Guo-Qiang Chen, and Prof. Chidchanok Lursinsap. Though indirect, their contribution to this thesis is essential.

I also extend my gratitude to Prof. Amos Bairoch, Prof. Niko Beerenwinkel, Prof. Jörg Stelling, and Prof. Martin Vingron for their participation to my thesis committee, and for their valuable feedback.

Lastly, and most importantly, I would like to thank my family: my parents Nisa and Jean-Daniel for their love, caring and advice all along; my sisters Lyne and Claire for their affection; my godparents Jasmine and Christian, not least for regularly asking when I would be done with this thesis! But above all, I thank Shumin, my beloved wife, for her encouragements, patience, support and companionship in the last four years. To her this thesis is dedicated.

Zurich in December 2008
Christophe Dessimoz

Contents

Abstract iii

Acknowledgments vii

Table of Contents ix

List of Abbreviations xiii

1 Introduction 1

I Pairwise Distances 3

2 Models of Evolution 5
- 2.1 Markovian model of sequence evolution . . . 5
- 2.2 Substitution matrices . . . 6

3 Sequence Alignment 7
- 3.1 Methods . . . 8
  - 3.1.1 Packages under consideration . . . 8
  - 3.1.2 Simulation strategy . . . 9
- 3.2 Results . . . 10
  - 3.2.1 Time Complexity . . . 10
  - 3.2.2 Accuracy on simulated data . . . 10
  - 3.2.3 Accuracy on real data . . . 14
- 3.3 Discussion and Conclusions . . . 17

4 Distance estimation 19
- 4.1 ML Estimation . . . 22
  - 4.1.1 ML distance estimator: pair of sequences . . . 22
  - 4.1.2 ML distance estimator: triplet of sequences . . . 23
- 4.2 Difference between two distances . . . 23
  - 4.2.1 Methods . . . 24
  - 4.2.2 Results and discussion . . . 26

4.2.3 Conclusions About Variance of Distance Estimator 35
4.3 Covariance of distances estimated pairwise 36
4.3.1 Methods 37
4.3.2 Results and Discussion 41
4.3.3 Conclusions About Covariance Estimator 45

II Applications in Comparative Genomics 47

5 Detection of Non-Orthology 49
5.1 Material and Methods 51
5.1.1 Input data 53
5.1.2 Comparison of Evolutionary Distances 53
5.1.3 Algorithm 54
5.1.4 Phylogenetic Analysis 54
5.1.5 Validation 55
5.2 Results and Discussion 56
5.2.1 Phylogenetic analysis of selected COG groups 57
5.3 Conclusions 63

6 OMA: Large Scale Orthology Inference 65
6.1 Algorithm 67
6.1.1 All Against All Alignments 68
6.1.2 Formation of Stable Pairs 70
6.1.3 Verification of Stable Pairs 72
6.1.4 Clustering of Orthologs 74
6.2 Validation and Comparison 75
6.2.1 Results and Discussion 79
6.2.2 Conclusions of Comparison Study 87
6.3 OMA Browser 88
6.3.1 Implementation 88
6.3.2 Protein-centric view 88
6.3.3 OMA group-centric view 88
6.3.4 Data export and integration 90

7 Lateral Gene Transfer Detection 91
7.1 Introduction 91
7.2 Method 92
7.2.1 Preliminaries 92
7.2.2 Algorithm 94
7.2.3 Model Violations and Test of Multivariate Normality . . . 98
7.2.4 Combination of Results and Correction for Multiple Testing 98

7.3 Validation and Results 99
7.3.1 *In Silico* Evolution Scenarios 99
7.3.2 Artificial LGT Events in Real Data 100
7.3.3 Real Biological Data 101
7.3.4 Comparison with Previous Results 103
7.4 Conclusion 104

8 Conclusions **107**

Appendix **109**
A1 Complexity of the analytical solution of k-states model for triplets 109
A2 Comparison of Orthology: Methods 111
A2.1 Input data 111
A2.2 Phylogenetic Reconstruction Test 112
A2.3 Benchmarks from literature 113
A2.4 Functional based definition 113
A3 Alternative LGT scoring functions 116
A3.1 GC Content 116
A3.2 Best Hit Approach 116
A3.3 Perturbed-Distances Approach 116

Bibliography **117**

Index **131**

Curriculum Vitae **137**

List of Abbreviations

AA	amino-acid
AP	all pairs
BBH	best bi-directional hits
BP	broken pairs
CDS	coding sequence features
EC	enzyme classification
GLS	generalized least-squares
GO	Gene Ontology
GP	group pairs
indel	insertion-deletion
IPA	induced pairwise alignment
LGT	lateral gene transfer
MCMC	Markov-chain Monte Carlo
ML	maximum likelihood
MSA	multiple sequence alignment
OLS	ordinary least-squares
OPA	optimal sequence alignment
PAM	point accepted mutation
RSD	reciprocal shortest distance
SP	stable pairs
VP	verified pairs
WLS	weighted least-squares

Please refer to the index (p. 131) for the list of occurrence of these terms.

1 Introduction

Recent years have witnessed an exponential increase in the sequencing of genomic data. In 1995, the first entire genome, *Haemophilus influenzae* (Fleischmann et al., 1995), was sequenced. By September 2000, 31 organisms, including 3 Eukaryotes, had been completely sequenced (Iliopoulos et al., 2001). As of March 2008, 663 genomes have been completed, and 1,280 genomes are in various stage of sequencing or assembly[1]. With new sequencing techniques promising up to hundredfold speed increases (Margulies et al., 2005) and an apparently inexhaustible supply of species and strains, this rapid increase is likely to sustain in the foreseeable future.

The sequencing and characterization of complete genomes opens the way to quantitative, systematic and ideally unbiased comparisons between species. Such investigations, commonly grouped under the label *comparative genomics* (O'Brien et al., 1999; Koonin et al., 2000), aim at improving our understanding of the evolutionary forces that shape genes, biological functions, species, and Life in general.

The perhaps biggest challenge faced by comparative genomics is the complexity of the problems it attempts to solve. Even under simplistic models, many seemingly elementary tasks are computationally hard if they are to be solved optimally. For example, the time complexity of aligning sequences, even under coarse models of evolution, is exponential in the number of sequences under consideration (Carillo and Lipman, 1988). Reconstruction of a phylogenetic tree under the parsimony criterion, that is, the minimum number of character changes along the tree, is NP-complete (Graham and Foulds, 1982).

[1] http://www.ncbi.nlm.nih.gov/genomes/static/gpstat.html

Thus, in practice, the effective application of comparative genomics involves a delicate trade-off between model complexity (which ideally improves accuracy) and quantity of data that can be processed (constrained by speed). The optimality of this trade-off depends on the question under investigation and the data available. This explains and justifies the coexistence of more than one methods for the same type of problem.

In this thesis, I explore and extend the class of comparative genomics methods that uses as input evolutionary distances between pairs of sequences only. This may appear inferior to using as input a tree, whose topology and branch lengths relate all sequences simultaneously. In some contexts however, such as large-scale analyses, we shall see that ignoring higher-order combinations of sequence information can be beneficial in terms of the aforementioned speed–accuracy trade-off.

Outline of the thesis

The first part reviews and develops methods to estimate and work with evolutionary distances between pairs of sequences. Chapter 2 reviews the fundamentals of Markovian models of evolution. In Chapter 3, we evaluate common approaches and tools to identify homologous characters (characters having common ancestry), a prerequisite for distances estimation. Chapter 4 treats the problem of estimating the evolutionary distance between two sequences, and presents our contributions to estimate the distribution of the difference between two distance estimates (Chapter 4.2), and more generally the covariance of two distances (Chapter 4.3).

The second part presents large-scale comparative genomics applications that leverage the tools presented in the first part. In Chapter 5, we show how pairwise distances among quartet of sequences can be used to detect differential gene losses. This is useful, because most methods identifying orthologs on the basis of pairwise gene comparison misclassify sequences resulting from differential gene losses as orthologs. In Chapter 6, we describe our own orthology identification project, OMA, and present a detailed comparison of its results with other efforts. The third application, presented in Chapter 7, uses pairwise distances to efficiently identify lateral gene transfer events.

Most of the original contributions presented in this thesis were the object of refereed publications in journals or conferences. A reference, as well as the name of my co-authors, are presented at the beginning of the relevant sections.

Part I

Pairwise Distances

2 Models of Evolution

In this chapter, we briefly review the model of evolution that will be used throughout the thesis.

2.1 Markovian model of sequence evolution

The evolutionary distance between two biological sequences is generally based on the assumption of a first-order Markovian process of evolution. This implies two biological assumptions, common to most standard models of evolution: dependency on current state only (i.e. no memory) and position-independence. The substitution processes are described in the form of substitution matrices, defining mutation probabilities from each character to every other character for a given evolutionary distance. These matrices are either parametric models of sequence evolution or empirically based substitution matrices. Parametric models are often employed for nucleotide substitution (e.g. Jukes and Cantor 1969 or Hasegawa et al. 1985), while empirical matrices (based on counted substitutions of large sets of sequence alignments) are widely used for amino-acid (AA) replacements in proteins. Pioneered by Dayhoff in the 1970s (Dayhoff et al., 1978), these models have been improved with more sequence data becoming available in the 1990s (e.g. the updated Dayhoff matrices by Gonnet et al. 1992 or Jones et al. 1992). Codon substitutions have been described by parametric (e.g. Goldman and Yang 1994) as well as empirical (e.g. Schneider et al. 2005) matrices.

2.2 Substitution matrices

Because of the additivity of distances computed under the Markovian model of sequence evolution, substitution matrices for a wide range of evolutionary distances can be derived from a single substitution matrix $M(d_0)$ through the equation $M(d_0)^x = M(xd_0)$, which is a special form of the Chapman-Kolmogorov equation for Markov chains. It is common and computationally more efficient to formulate this process in terms of a rate matrix Q from which the probability matrices for distance d are derived as $M(d) = e^{dQ}$. In the present work, we measure d in PAM units (Dayhoff et al., 1978), which completely defines Q.

3
Sequence Alignment

This section is joint work with Manuel Gil

The study of biological sequences almost inevitably begins with the process of alignment. In most cases, the goal of this process is to match homologous characters, that is, characters that have common ancestry. In the present context, the quality of this process is crucial because it affects all subsequent analysis, and in particular distance estimation between pairs of sequences. Thus, our main concern here is to have accurate alignments for pairwise distance estimation.

If we have n homologous sequences under consideration, the $\binom{n}{2}$ pairwise alignments can be computed either independently, usually using algorithms such as Needleman and Wunsch (1970) for global alignments or Smith and Waterman (1981a) for local alignments. Since such algorithms are guaranteed to find the optimal matching under a particular model (the matching with highest score), we refer to such alignments as optimal pairwise alignments (*OPA*s). Alternatively, they can be computed using a multiple alignment of all n sequences (*MSA*), from which all pairwise alignments are induced (*IPA*s).

The accuracy of multiple alignment methods has been evaluated in numerous studies; see e.g. reviews of Blackshields et al. (2006); Edgar and Batzoglou (2006); Notredame (2007). They usually rely on curated reference alignments, *gold standards*, as benchmark, such as Balibase (Thompson et al., 2005), Prefab (Edgar, 2004) or Homstrad (Stebbings and Mizuguchi, 2004). Most of these alignments are obtained using protein structure information, because in general, structure tends to evolve more slowly than sequence (Chotia and Lesk, 1986).

Yet, this quasi-exclusive reliance upon structural alignments has some drawbacks. First, structural alignments have their own difficulties: the statistics of structural alignment scores is less well understood than that of their sequence-based counterparts. Indeed, studies have reported that their statistical significance could be overestimated by up to an order of magnitude (Sierk and Pearson, 2004). Cases of convergent evolution of protein structure can cause errors in homology inference, though the extent of this phenomenon remains debated (Pearson and Sierk, 2005; Gough, 2005; Gherardini et al., 2007). Second, the different applications of sequence alignments have at times irreconciliable optimality criteria: depending on the context, a correctly aligned position could be expected to have spatial proximity, or common ancestry, or conserved function. Thus, alignment methods should be evaluated in the context of the different applications, too.

Much of the previous work investigating the effect of alignment accuracy on evolutionary distance estimation and phylogenetic inference was done by Rosenberg (2005a,b) and co-authors (Ogden and Rosenberg, 2006), but these articles are generally restricted to small sets of simulated data, up to 16 sequences. In addition, they do not compare results from both pairwise and multiple sequence alignment methods. Yet, both approaches can be used for the purpose of pairwise distance estimation.

In this chapter, we compare pairwise and multiple sequence alignments packages for the purpose of evolutionary distance estimation. Our assessment is based both on simulation and real biological sequences. And since most of the methods introduced in this thesis involve large datasets, the analysis extends to sets of several hundred sequences.

3.1 Methods

3.1.1 Packages under consideration

1. Needleman–Wunsch: implementation in *Darwin* (Gonnet et al., 2000). The scoring matrix is the updated 70-PAM Dayhoff matrix (Gonnet et al., 1992).

2. *Maximum likelihood pairwise alignments:* implementation in *Darwin*. This procedure extends Needleman–Wunsch by optimizing the score in terms of both the alignment and the PAM distance of the Dayhoff matrix (see Thorne et al. (1991) for a similar approach). Since scores are log of odd ratios, this procedure yields the ML alignment. In this implementation, the maximization is performed iteratively: they first align the sequences based on an initial distance; they then iterate between distance estimation (Chapter 4) and alignment until convergence. To avoid being stuck

in local maxima, they repeat the procedure starting from several initial distances.

3. *Muscle:* the first MSA package evaluated here is *Muscle* (Edgar, 2004), version 3.2, default settings.

4. *Muscle optimized:* same as above, but using the updated 70-PAM Dayhoff matrix.

5. *Mafft:* the second MSA package evaluated here is *Mafft* (Katoh et al., 2005), version 6.24, default settings.

6. *Mafft optimized:* same as above, but using the updated 70-PAM Dayhoff matrix and using the option "local pair", which refines the MSA using local pairwise alignments.

3.1.2 Simulation strategy

To generate a set of n homologs, a random subtree of size n is sampled from the least-squares distance tree of life built from orthologs according to *OMA* (Chapter 6). A sequence of length m is generated according to AA frequencies from the stationary distribution of the PAM matrices, and is mutated and duplicated along the subtree according to the PAM model with indel according to a Zipfian distribution (Benner et al., 1993), while tracking all events. The resulting sequences are then used as input for the different methods, and their output can be compared to the known history.

3.2 Results

The results are divided into three parts. First, we look at the computational cost of the different methods, then their accuracy based on simulation, and finally their accuracy based on real data.

3.2.1 Time Complexity

The time complexity of the different methods is mainly influenced by the number of sequences (n) and the starting sequence length (m). In the case of Needleman–Wunsch, $O(n^2)$ pairs are computed, each requiring $O(m^2)$ time, so the overall time complexity is $O(n^2m^2)$. In the case of ML pairwise estimation, the optimal distance (and consequently scoring matrix) must also be determined. This approach is clearly more time-consuming than Needleman–Wunsch, but since the number of iterations required depends on the structure of the problem, we cannot provide an asymptotic bound. As for MSAs, an optimal solution to the problem is $O(m^n)$, which is clearly too expensive in practice. Thus, most MSA packages use heuristics, which makes it difficult to obtain asymptotic bounds.

The CPU time required by the different methods was evaluated empirically. Results are shown in Figure 3.1. *Mafft* using default settings is significantly faster than all other methods, and also scales best in terms of the number of sequences. The other methods scale roughly quadratically with the number of sequences. The slowest method by far is our ML pairwise alignment procedure. As for the impact of sequence divergence, pairwise methods are not affected, whereas MSA programs align closer sequences significantly faster than more distant sequences. Finally, time complexity scales with the square of sequence length, except in the case of both *Muscle* variants, which appear to scale sub-quadratically.

3.2.2 Accuracy on simulated data

The accuracy of the different packages was first tested by simulation (see Section 3.1.2). We used two measures of accuracy. First, we used the average percentage of correctly aligned positions over all pairwise alignments. Second, we computed the average bias of distances from all pairwise alignments.

Correctly aligned positions

Figure 3.2a plots the average percentage of correctly aligned positions versus the number of sequences. It appears that in all cases, the accuracy does not depend (at least in terms of mean and variance) on the number of sequences analyzed. This is not surprising in the case of pairwise methods, since each pair is processed independently, but in the case of IPAs, this result disagrees with the common idea that more sequences (i.e. more data) increase the accuracy of

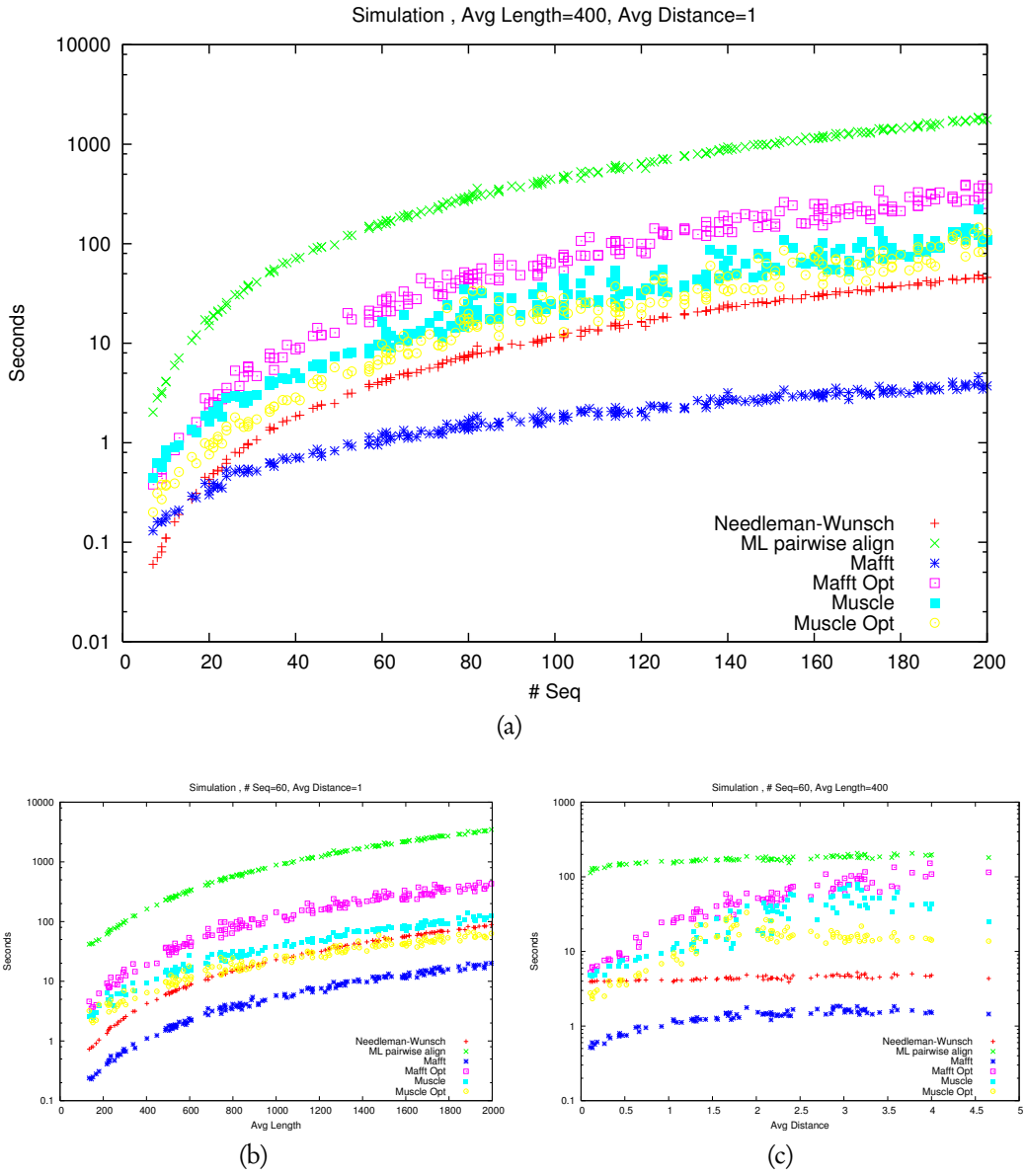


Figure 3.1: Comparison of the run-time (CPU time) of the different methods as a (a) function of number of sequences, (b) average sequence length, and (c) average evolutionary distance. Note the log-scale in the Y-axis

multiple sequence alignment (McClure et al., 1994; Thompson et al., 1999). Figure 3.2b reveals that the accuracy does not depend on sequence length either. Figure 3.2c plots the accuracy versus the average PAM distance. Unsurprisingly, closer sequences are easier to align, and hence for all methods, the mean accuracy has strong negative correlation with the average distance. Overall, *Mafft optimized* performs best, then *Muscle* and *Mafft* in their default settings. The pairwise methods are inferior, with the *ML pairwise alignment* method performing better than *Needleman–Wunsch*.

Distance estimates from alignments

In the context of this thesis, alignments are mainly used as a basis for distance estimation (Chapter 4.1.1); therefore, the impact of alignment errors on distance estimation is of particular relevance. In this part, we compare the different methods in terms of the distance estimated from their alignments. For reference, we also compare the results to distance estimates obtained from perfect alignments, that is, alignments with 100% correctly aligned positions known from the simulation approach. Figure 3.3a plots the average bias of estimated distances versus the number of sequences. First, we observe that there is no significant dependency between the average error and the number of sequences. With all methods, the variance decreases as the number of sequences increases, but this is a consequence of the quadratic growth in pairs of distances, which reduces the standard error of the mean estimates. A minor method-specific bias is noticeable, with *Needleman–Wunsch* leading to overestimating distances, and *ML pairwise alignments* leading to underestimates. Surprisingly, Figure 3.3b reveals no dependency of average distance accuracy on sequence length. This is unexpected because the variance of ML distance estimation decreases with large sequence. This suggests that the main source of variance past a certain length originates from the sampling process, here the different underlying subtrees. Overall, to see a difference among the methods, we need to increase significantly the average distance, as shown in Figure 3.3c. There, when sequences are far apart, distance estimation becomes difficult. The fast MSA methods are particularly affected by very long distances. On the other hand, *ML pairwise alignment* and *Mafft Opt*, which also rely on pairwise alignments, reveal themselves to be more robust. The good performances of *ML pairwise alignment* here are unexpected if one considers its poor results in the previous test: how are such relatively accurate distances estimated from such poor alignments? We provide the following asymptotic argument: in time-reversible Markov models, the stationary distribution is equal to the background frequency of characters. Therefore, at saturation (that is, as distance approaches infinity), the probability of homologous pair of characters converges to that of unrelated characters. This is consistent with the intuition that infinitely distant homologs cannot be distinguished from unrelated sequences. Thus, as distance increases, alignment mismatches have a

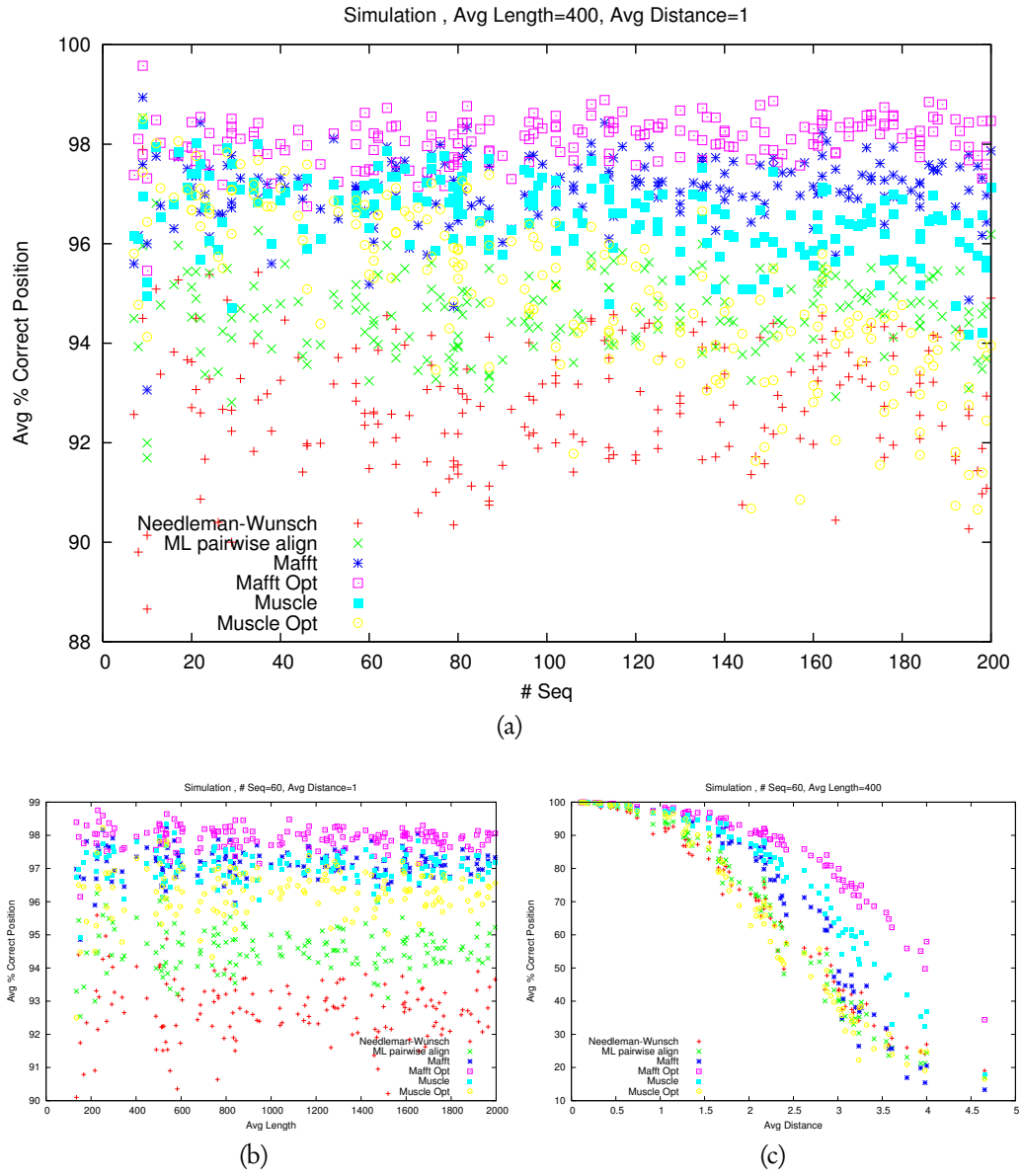


Figure 3.2: Comparison of the accuracy of the different methods in % of correctly aligned positions in pairwise alignments, as (a) function of sequences, (b) average sequence length, and (c) average evolutionary distance

diminishing effect on distance estimation.

This prompted us to investigate the effect of inclusion of unrelated sequences to the set of sequences to align. We repeated the simulation with a varying number of random, unrelated sequences in addition to a set of homologous sequences (Fig. 3.4). Under these circumstances, *ML pairwise alignment* is clearly the most robust to inclusion of non-homologous sequences.

3.2.3 Accuracy on real data

In addition to the above simulation, we also performed two tests based on real data. This is important, because no model can fully capture the complexity of biological evolution. By testing the alignment procedures on real data, we can hope to gauge the impact of departures from our model of evolution (Chapter 2) on alignment accuracy. In the first test, we estimate the fraction of correctly aligned positions relying on Balibase (Thompson et al., 2005), a curated database of reference alignments from structural data mentioned at the beginning of this chapter. In the second test, we take the evolutionary perspective and assess the quality of distances estimated from the different alignments by looking at trees with known and undisputed topology, reconstructed from the alignments.

Correctly aligned positions

We test accuracy on real data using version 3 of Balibase (Thompson et al., 2005). In Figure 3.5, we plot the percentage of correct position, normalized with respect to *Needleman–Wunsch*, versus the number of sequence. The ranking is very similar to that obtained by simulation, with *Mafft optimized* performing best, then the other MSA methods, and finally the two pairwise methods.

Distance Estimates and Tree Building from Alignments

To assess the performances of the various approaches on real data, we reconstructed trees from orthologous genes of species with well-resolved branching order, aligned by the different packages (Altenhoff and Dessimoz, 2009). Since the only difference in methodology stems from the alignment process, the level of congruence between orthologs and species trees can be used to rank the different alignment methods. Overall, the MSA methods did better, albeit not always significantly so (Fig. 3.6). For sequences spanning all Eukaryotes, the difference between the two types was significant (Figure 3.6, left). In contrast, of the two pairwise method, only "ML pairwise" was dominated in the Fungi dataset (Figure 3.6, right). The bias compensation that we observed in some simulations above does not seem to be relevant when dealing with real data. Among the MSA packages, *Mafft Optimized* performed slightly better, but the difference to the other MSA variants is not significant.

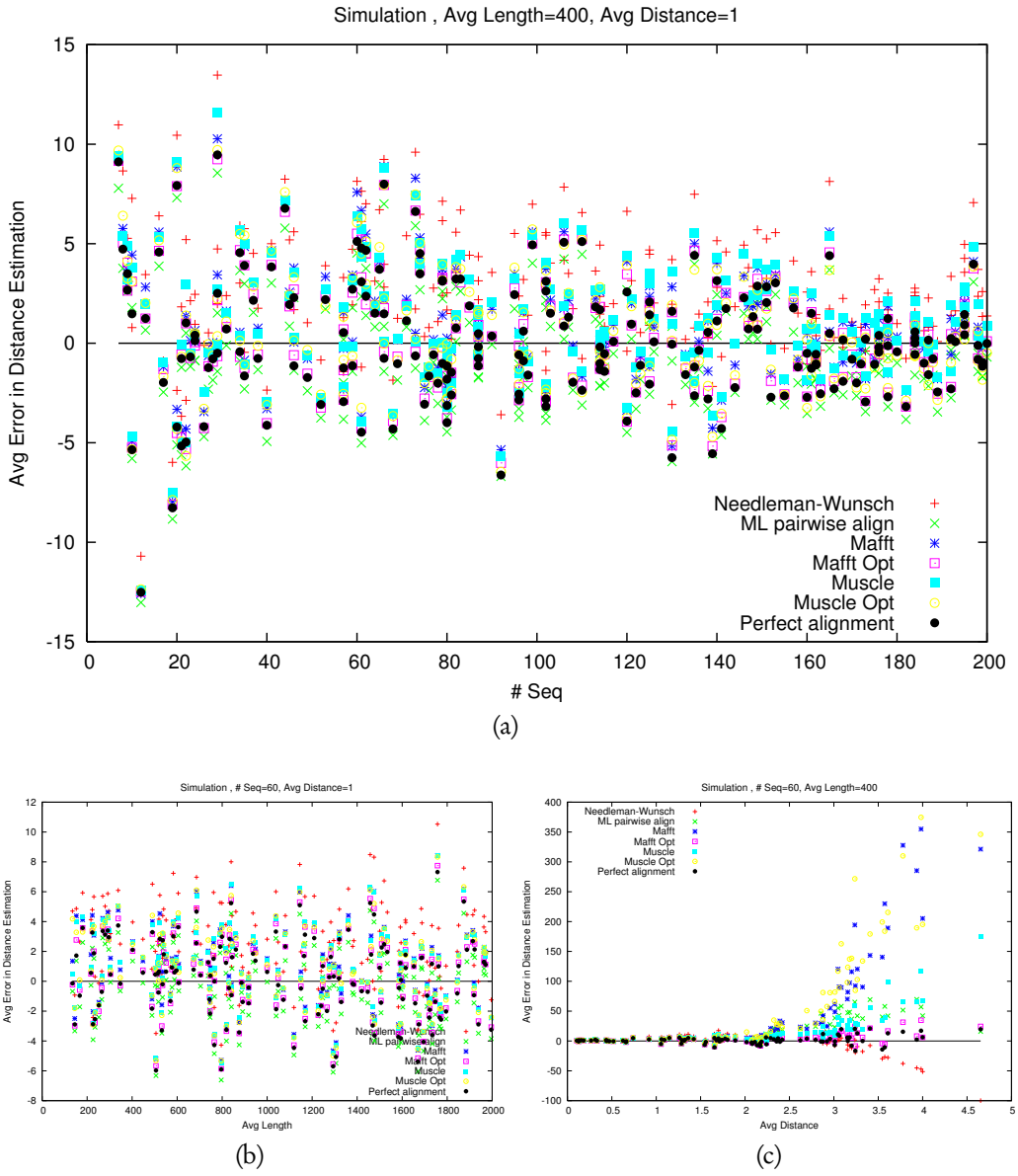


Figure 3.3: Comparison of average error in pairwise distance estimation from alignment computed by the different methods, as a function of sequences (a), average sequence length (b), and average evolutionary distance (c)

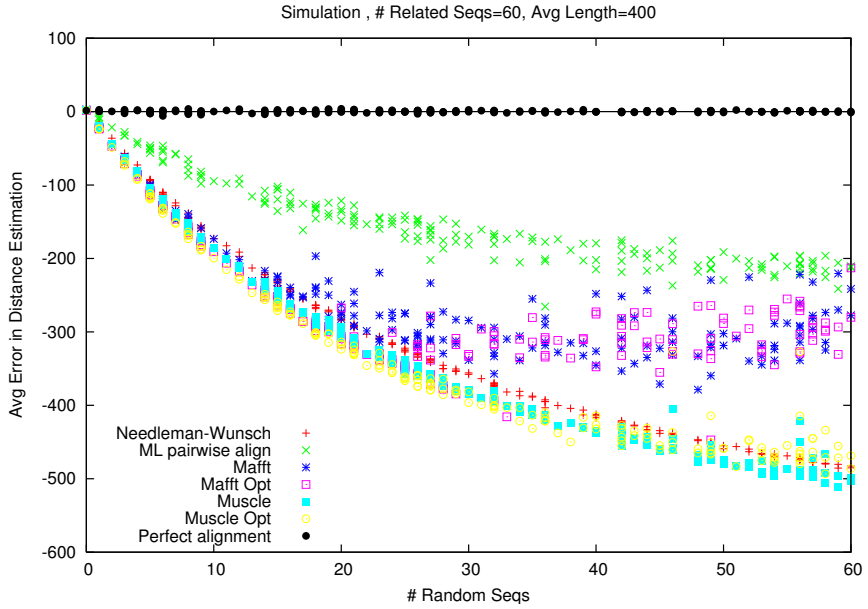


Figure 3.4: Comparison of average error in pairwise distance estimation from alignment computed by the different methods, as a function of the number of unrelated sequences included in the analysis

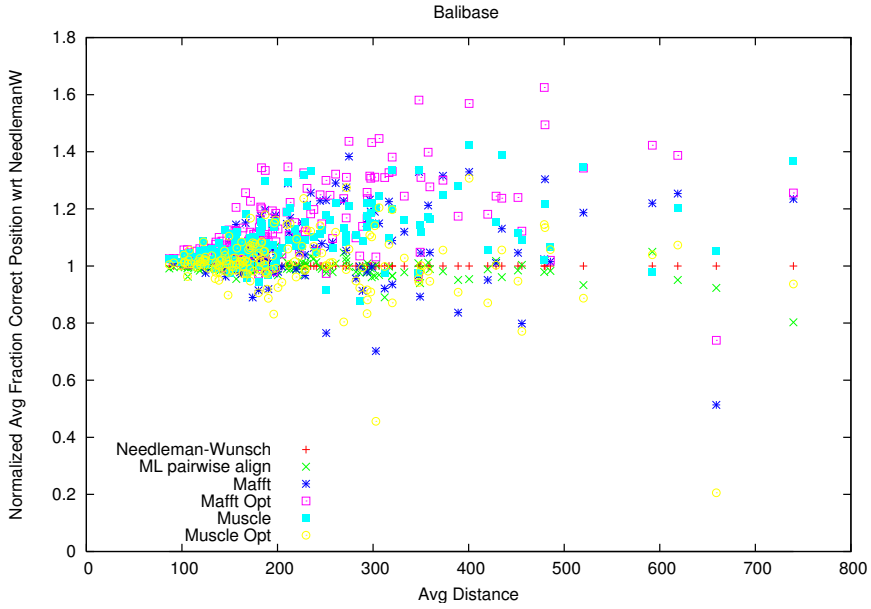


Figure 3.5: Comparison of the relative accuracy of the different methods in terms of correctly aligned positions normalized with respect to the accuracy obtained using *Needleman–Wunsch*, using the reference alignments of Balibase version 3

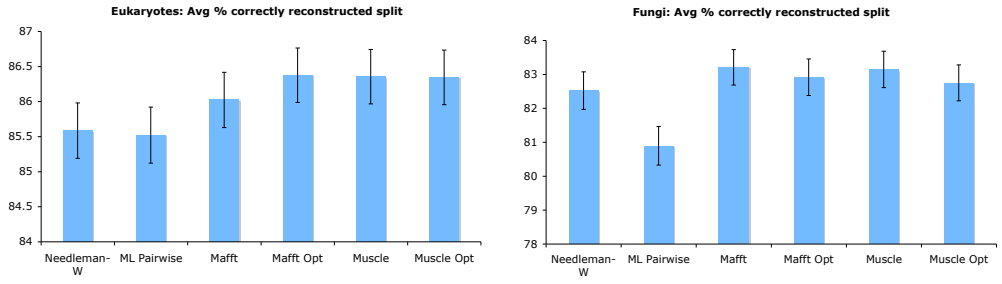


Figure 3.6: Comparison of the degree of correctly reconstructed phylogenetic tree topologies (1 minus Robinson-Foulds distance) based on the alignments from the different methods. Error bars depict 95% confidence intervals.

3.3 Discussion and Conclusions

In the context of estimating pairwise distances, we compared the runtime and accuracy of pairwise alignment versus MSA methods, both on simulated and real biological data. The results obtained consistently suggest that current MSA packages outperform pairwise alignment procedure, both in terms of time and correctly aligned positions. Moreover, MSA packages are likely to continue improving, whereas pairwise alignment already yields optimal results in terms of the evolutionary model.

However, distance estimation is surprisingly robust with respect to misaligned characters (as previously observed in Rosenberg 2005a); it appears that distance computed from IPAs are not significantly more accurate than from OPAs. In fact, the best estimates are obtained from *ML pairwise alignment* and *Mafft Opt* (which also computes pairwise alignments).

Finally, if we compare the two pairwise alignment methods, the results also show that joint alignment and distance estimation in the ML pairwise alignment method is an order of magnitude more computationally intensive than separate alignment and distance estimation, but yields better distance estimates, especially when distances are large.

4

Distance estimation

The estimation of evolutionary distances between gene and protein sequences is one of the most important problems in molecular evolution. it plays a particularly important role in phylogenetic tree inference (Swofford et al., 1996; Felsenstein, 2004b) and in an increasing number of comparative genomics analyses over large sets of genes or proteins (e.g. Kanehisa et al., 2004; Dessimoz et al., 2005; DeLuca et al., 2006).

The estimation of such distances is a two step process: first, homologous characters are identified, then the distances are estimated from the character substitution patterns. As we saw in Chapter 3, the matching of homologous positions can be done through a multiple sequence alignments (MSAs), which considers all sequences simultaneously, or through pairwise sequence alignment, using an algorithm such as Smith and Waterman (1981a). Though the former approach yields better results in theory and practice, the latter approach is often taken by large-scale comparative genomics analysis such as MIPS, RoundUp, or OMA (Mewes et al. 2004; DeLuca et al. 2006; Chapter 6), which analyze the sequences pairwise due to computational constraints.

Once the homologous characters are identified, the second step of distance estimation can proceed. The method of choice is a maximum likelihood (ML) estimation based on some model of evolution. There too, the distances can either be estimated simultaneously from all sequences using a combination of tree topology inference and joint optimization of all branches, or pairwise, by estimating the distances between every pair of sequences. Joint estimation requires MSAs, while pairwise distance estimation can be done from either OPAs or from the pairwise alignments induced by an MSA (IPAs). Fig. 4.1 provides an overview of the different approaches.

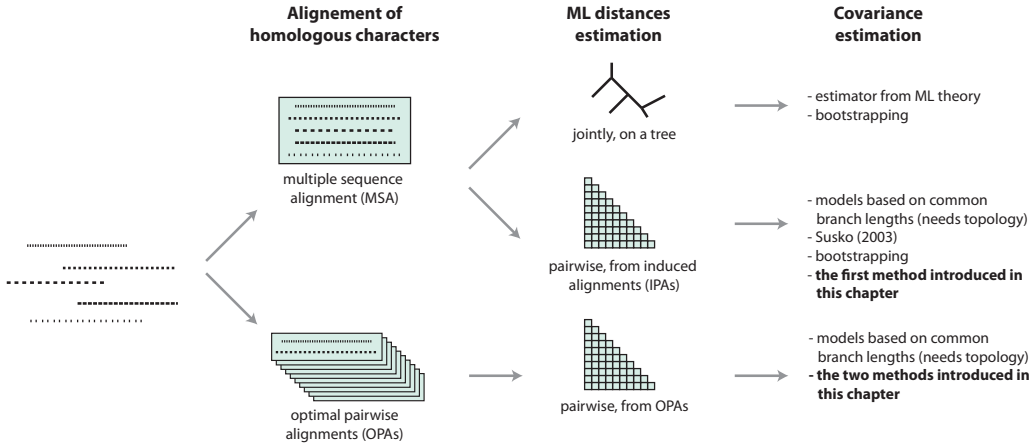


Figure 4.1: Overview of approaches to estimate evolutionary distances and their covariances: a set of n sequences can be aligned jointly to obtain an MSA or in a pairwise optimal manner resulting in $\binom{n}{2}$ optimal pairwise alignments (OPAs). Given a hypothesis of character homology, distance estimation per ML can essentially be done in two ways: jointly on a tree or pairwise. In the first case a tree's branch-lengths are estimated simultaneously. This requires an MSA. In the second case pairwise distances are estimated either from MSA induced pairwise alignments (IPAs) or from the OPAs. The distance estimators are afflicted with an error expressed by their variances and covariances. In all cases, the covariances can be modeled as a function of shared branch lengths, but this requires a phylogenetic tree. When distances are estimated based on an MSA, the variances and covariances can be obtained from ML theory or by bootstrapping over the MSA's columns. In the case of OPAs, these techniques cannot be directly applied. In this chapter, we present a covariance estimator for the case where the two OPAs in question share a sequence (i.e. for triplets), as well as an estimator for the general case.

In all cases, the estimation of evolutionary distances is subject to inference uncertainty, which is commonly quantified by their variances and covariances. Indeed, the distance variance information can be used to build confidence intervals around the estimate; covariances of pairs of distances can be used to build the confidence intervals of combinations of distances. Examples of applications include generalized least squares (GLS) phylogenetic tree building (Cavalli-Sforza and Edwards, 1967) construction of confidence sets of trees (Bulmer, 1991), test for monophyly using likelihood ratios (Huelsenbeck et al., 1996), comparison of evolutionary distances for orthology inference (Chapter 6), or distance-based lateral gene transfer detection (Chapter 7).

As we shall see shortly, variance estimates are provided by ML theory in both joint and pairwise distances estimation. However, ML theory only provides covariance estimates if all distances are estimated jointly. Covariance estimates for distances computed from IPAs in the context of specific parametric substitution models have been reported by Hasegawa et al. (1985) and Bulmer (1991), and were generalized by Susko (2003) to all Markovian models of evolution. Furthermore, the covariance of distances from IPAs can also be estimated — though much more slowly — through bootstrapping (Efron and Tibshirani, 1993). As for the covariance of distances obtained from OPAs, the main difficulty in computing them is that, since sequence pairs are aligned individually, they usually have inconsistencies in their inference of the homologous characters (or else, computing an MSA from pairwise alignments would be trivial). Thus, the alignments cannot be partitioned in consistent "columns" of characters, and neither Susko's method nor re-sampling approaches such as bootstrapping can be applied. Indeed, in the case of analyses relying exclusively on pairwise comparison and distance estimation, i.e. where no MSA computation can be afforded, we are not aware of any previously published estimator for the covariance of distances estimates from pairwise alignments.

Following a short review of ML distance and variance estimation, we introduce two methods to estimate the covariance of pairwise distances estimated from independent alignments (as can be the case in large-scale studies). The first method computes the variance of the difference between two distances approximation in empirical amino-acid substitution models. This is equivalent to computing the covariance between two distances that involve a common sequence. The second method is more general in the sense that it works with arbitrary Markovian models, and it also estimates the covariance of distances involving four distinct sequences, thereby allowing the computation of full covariance matrices.

4.1 ML Estimation

Evolutionary distances are best estimated by maximum likelihood (ML). In case of a pair of sequences, the ML estimation is well known and practical. When more sequences are under consideration, the complexity of distance estimation by ML increases very steeply, not only because the increased dimensionality of the optimization space, but also because such optimization should also be done simultaneously at the level of the multiple sequence alignment (MSA) and the phylogenetic tree topology (unrooted topologies for time-reversible models such as in the case of PAM matrices), two difficult procedures for which the optimal solution can currently only be computed in exponential time with respect to the number of sequences. In practice, the optimization of the MSA is often performed as a separate process prior to tree inference. In the following, we review how to compute ML distances for pairs and triplets of sequences, given correct alignments. While the case of a pair is central to the work of this thesis, the case of a triplet will be used in Sect. 4.2. These cases are also of particular interest because the underlying tree topology is trivial: for two and tree sequences, there is only one possible unrooted tree topology. For joint ML distance estimation in the case of more than 3 sequences, refer to Felsenstein (2004a) or Yang (2006).

4.1.1 ML distance estimator: pair of sequences

The unrooted tree topology underlying any pair of homologs is always a single branch connecting both sequences whose length must be estimated. Thus, the correct topology is trivial to obtain, and the problem in one dimension only. Given a correct pairwise alignment A and substitution matrices $M(d)$ (see 2.2), the likelihood of having two two sequences separated by evolutionary distance d is (Felsenstein, 1981; Gonnet, 1994; Muller and Vingron, 2000)

$$\begin{aligned} L(A \mid d) &= \prod_{[x,y]\in A} f(x) M_{x,y}(d) \\ &= \prod_{[x,y]\in A} f(x) \left[e^{dQ} \right]_{x,y} \end{aligned}$$

with x and y being aligned characters (e.g. amino acids, bases, but no deletions), and $f(x)$ the background frequency of the character x. Maximizing $L(A \mid d)$ in terms of d gives the ML estimator $\hat{d}$ of the evolutionary distance. This is usually done numerically using the Newton-Raphson method. The variance of the ML estimator $\hat{d}$ can be computed from the second derivative of the log-likelihood:

$$\sigma^2(\hat{d}) = -\left(\frac{\partial^2 l(A|\hat{d})}{\partial d^2} \right)^{-1}$$

Notice that the variance is obtained for free as it is already computed in Newton's iteration.

4.1.2 ML distance estimator: triplet of sequences

When more than three sequences are under consideration, the number of distances to estimate are the 3 inner-branch lengths of the (unique) underlying unrooted tree topology. In almost all models, it is assumed that the evolution in different lineages (i.e. branches) is independent (Felsenstein, 1981). The likelihood of the observed data can be computed, given the correct MSA and a family of substitution matrices $M(d)$ (see 2.2) as follows:

$$L(MSA|d_{OX},d_{OY},d_{OZ}) = \prod_{[x,y,z]} \sum_{o \in C} f(o) \left[e^{d_{OX}Q}\right]_{o,x} \left[e^{d_{OY}Q}\right]_{o,y} \left[e^{d_{OZ}Q}\right]_{o,z}$$

where C is the set of characters, and $f(o)$ the background frequency of the character o. Consequently, the log-likelihood function l is

$$l(MSA|d_{OX},d_{OY},d_{OZ}) = \sum_{[x,y,z]} \log \sum_{o \in C} f(o) \left[e^{d_{OX}Q}\right]_{o,x} \left[e^{d_{OY}Q}\right]_{o,y} \left[e^{d_{OZ}Q}\right]_{o,z}$$

If the log-likelihood is maximum, then its gradient disappears:

$$\nabla l = \begin{pmatrix} \partial l/\partial d_{OX} \\ \partial l/\partial d_{OY} \\ \partial l/\partial d_{OZ} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

There again, the problem can be solved efficiently by Newton's iteration

$$\begin{pmatrix} \hat{d}_{OX} \\ \hat{d}_{OY} \\ \hat{d}_{OZ} \end{pmatrix}_{i+1} = \begin{pmatrix} \hat{d}_{OX} \\ \hat{d}_{OY} \\ \hat{d}_{OZ} \end{pmatrix}_{i} - (\nabla^2 l_i)^{-1} \nabla l_i$$

where $(\nabla^2 l)^{-1}$ is the inverse of the Hessian (derivable in the same fashion as the gradient, not shown here). The inverse of the Hessian also yields the variance-covariance matrix of the estimates $\hat{d}_{OX}, \hat{d}_{OY}, \hat{d}_{OZ}$ when multiplied by -1. A final use of the Hessian is to check that its complement is positive definite, a condition necessary to ensure that the solution found is indeed a maximum and not a minimum or a saddle point.

4.2 Difference between two distances

This section is joint-work with Manuel Gil, Adrian Schneider, and Gaston Gonnet. It was published in Dessimoz et al. (2006b)

The estimation of the difference between two evolutionary distances within a triplet of homologs is a common operation that is used for example to determine which of two sequences is closer to a third one. The most accurate method is currently maximum likelihood (ML) over the entire triplet. However, this approach is relatively time consuming. We show that an alternative estimator, based on pairwise estimates and therefore much faster to compute, has almost the same statistical power as the maximum likelihood estimator. We also provide a numerical approximation for its variance, which could otherwise only be estimated through an expensive re-sampling approach such as bootstrapping. An extensive simulation demonstrates that the approximation delivers precise confidence intervals. To illustrate the possible applications of these results, we show how they improve the detection of asymmetric evolution, and the identification of the closest relative to a given sequence in a group of homologs. The results presented in this section constitute a basis for large-scale protein cross-comparisons of pairwise evolutionary distances.

The most accurate way of estimating evolutionary distances is currently ML, but the procedure is so time-consuming that is hardly practical when dealing with large datasets. In such cases, complexity is often tackled by working on the basis of individual pairs, such as in distance tree methods or in the "all-against-all" at the beginning of many comparative genomics analyses (see e.g. Chapter 6.1.1). However, by estimating an evolutionary distance for each pair individually, no knowledge about the covariance of distance estimates with common evolution can be directly obtained. Thus, when comparing pairwise distances among related sequences, for instance to infer which of two homologs is closer to a third one, confidence intervals cannot be derived directly from the pairwise estimates.

The present section investigates this fundamental problem of estimating the difference between two distances in a triplet of homologs (Fig. 4.2). We compare the standard multivariate ML approach with a much faster estimator based on pairwise distances, and present a formula to estimate its variance. As two examples of applications, we show how our results improve the detection of asymmetric evolution and the identification of the closest relative in a group of homologs.

4.2.1 Methods

In this section, we focus on the specific problem of estimating, in a triplet of homologs X, Y, Z (Fig. 4.2), the difference Δ between two distances d_{XY} and d_{XZ}. In such case, the multidimensional ML approach over the triplet is still practical. We call the estimator of Δ obtained by this method $\hat{\Delta}_{triplet}$:

$$\hat{\Delta}_{triplet} = \hat{d}_{OY} - \hat{d}_{OZ}$$

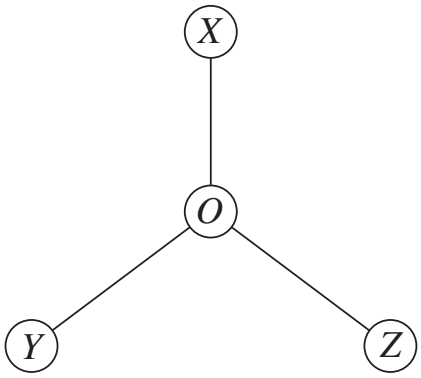


Figure 4.2: Unrooted tree topology of all triplets of homologs Sequences X, Y and Z originating from O. The problem addressed here is the estimation of the difference $\Delta = d_{XY} - d_{XZ} = d_{OY} - d_{OZ}$

Therefore, we obtain the variance of $\hat{\Delta}_{triplet}$ from the variance-covariance matrix:

$$\sigma^2(\hat{\Delta}_{triplet}) = \sigma^2(\hat{d}_{OY}) + \sigma^2(\hat{d}_{OZ}) - 2cov(\hat{d}_{OY}, \hat{d}_{OZ})$$

$$= [0, 1, -1](-\nabla^2 l)^{-1} \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix}$$

Alternatively, one can estimate Δ by performing pairwise alignments between X and Y, and between X and Z. The ML method for pairs of homologs, which was described above, computes the estimates $\hat{d}_{XY}$ and $\hat{d}_{XZ}$. By subtracting the first from the second, the alternative estimator $\hat{\Delta}_{pairwise}$ for the difference is obtained:

$$\hat{\Delta}_{pairwise} = \hat{d}_{XY} - \hat{d}_{XZ}$$

Since the two pairwise distance estimators are asymptotically unbiased and normally distributed, and considering the linearity of the expected value and the fact that the difference of two normally distributed variables is also normally distributed, the pairwise estimator $\hat{\Delta}_{pairwise}$ is also asymptotically unbiased and normally distributed, with variance

$$\sigma^2(\hat{\Delta}_{pairwise}) = \sigma^2(\hat{d}_{XY}) + \sigma^2(\hat{d}_{XZ}) - 2cov(\hat{d}_{XY}, \hat{d}_{XZ})$$

As described above, we obtain $\sigma^2(\hat{d}_{XY})$ and $\sigma^2(\hat{d}_{XZ})$ from the ML distance estimation, but since the two distances are estimated in two independent processes, we cannot use the Hessian to estimate their covariance. If the two distances span over non-overlapping paths on the tree, which is only the case if

$d_{OX} = 0$, the distance estimators are independant, the covariance is zero, and the variance $\sigma^2_{ind}(\hat{\Delta}_{pairwise}) = \sigma^2(\hat{d}_{XY}) + \sigma^2(\hat{d}_{XZ})$ can be computed. In all other cases, $\hat{d}_{XY}$ and $\hat{d}_{XZ}$ covary and the variance of their difference is smaller than the sum of their variances. Therefore, we only have an upper bound for the variance of our estimator:

$$\sigma^2(\hat{\Delta}_{pairwise}) \leq \sigma^2_{ind}(\hat{\Delta}_{pairwise})$$

Note that previous work on covariance estimation (e.g. Hasegawa et al. 1985; Bulmer 1991) do not apply here, because they require 3-way sequence alignments and are constrained to parametric models of evolution such as Jukes-Cantor and its generalizations.

4.2.2 Results and discussion

In the present section, we compare the estimators $\hat{\Delta}_{triplet}$ and $\hat{\Delta}_{pairwise}$, and introduce a numerical approximation to estimate the variance of $\hat{\Delta}_{pairwise}$, and show that it gives accurate confidence intervals. Finally, we describe two applications of the results.

Comparison between the two estimators

In terms of computational complexity, the two estimators differ significantly. Given m sequences of length n, $\hat{\Delta}_{triplet}$ requires the separate treatment of each $O(m^3)$ triplet, and considering that an optimal 3-way alignment by dynamic programming (DP) is $O(n^3)$, the time complexity is $O(m^3 n^3)$. In contrast, all $\hat{\Delta}_{pairwise}$ can be computed on the basis of $O(m^2)$ pairs of sequences aligned by DP in $O(n^2)$, yielding a time complexity of $O(n^2 m^2)$. Typically, whenever an analysis involves more than a few thousand proteins, millions of triplets have to be considered and $\hat{\Delta}_{pairwise}$ is the only practical approach of the two.

In terms of accuracy, both estimators are asymptotically unbiased: in the case of $\hat{\Delta}_{triplet}$, it is a property of the ML estimator, while in the case of $\hat{\Delta}_{pairwise}$, it is the consequence of the linearity of the expected value (Sect. 4.2.1). We compared the two estimators by simulation over a large number of triplets (length: 300 AA), generated randomly according to the PAM model of evolution with different distances d_{OX}, d_{OY}, d_{OZ} (Fig. 4.2). In each experiment, both estimators were converging toward the correct value for the difference, which confirms that the asymptotic behavior is a reasonable assumption for protein sequences of typical length. In terms of statistical power, surprisingly, the observed variance of the estimates obtained by $\hat{\Delta}_{pairwise}$ was on average less than 1% larger than the observed variance of the ML estimator over the triplet, suggesting that $\hat{\Delta}_{pairwise}$, although much faster to compute, is on average almost as accurate as $\hat{\Delta}_{triplet}$.

The variance of $\hat{\Delta}_{triplet}$ can be computed exactly (Sect. 4.2.1). But there is no direct estimator of the variance of $\hat{\Delta}_{pairwise}$, since it results from an algebraic relation over pairwise distances estimated individually, whose covariances are therefore unknown. There are indirect ways of estimating that variance, through the sampling distribution when doing simulation such as the one mentioned above, or bootstrapping when handling real data. However, such procedures are very time consuming. To overcome this problem, we devised a numerical approximation of $\sigma^2(\hat{\Delta}_{pairwise})$ as function of the pairwise distance estimates.

Numerical approximation of $\sigma^2(\hat{\Delta}_{pairwise})$

In essence, the numerical approximation described here was obtained through regression over a large number of samples. We settled for this approach after discovering that the analytical solution to this problem, even when using a simpler model of evolution (all AA mutations with equal probability), requires solving a polynomial of degree 23. The details of this investigation are reported in the Appendix A1. In view of this inherent complexity, the regression cannot be exact, but it turns out to be a surprisingly precise numerical approximation for comparisons that involve proteins that have an evolutionary distance smaller than 250 PAM units, which corresponds to percentage sequence identity greater or equal to 19.68%.

We generated random triplets in the following way: a random-length (uniform 100..500) sequence was chosen as the origin O. Three random PAM distances (uniform 1..125) were selected for d_{OX}, d_{OY} and d_{OZ}. The sequence O was mutated according to these distances to obtain X, Y and Z, our triplet. We generated about 30,000 triplets for three types of scoring matrix: updated Dayhoff matrices (Gonnet et al., 1992), DNA for coding genes and JTT (Jones et al., 1992). The DNA scoring matrices were computed from a very large set of entire coding gene alignments from mammals. It is used in the OMA project (Chapter 6) to align entire coding genes and is based on a 4-symbol alphabet. For each triplet, we computed pairwise distance estimates and their variances as input for the approximation. Given that $\hat{\Delta}_{pairwise}$ is almost as powerful as $\hat{\Delta}_{triplet}$, we computed and used $\sigma^2(\hat{\Delta}_{triplet})$ as reference value for $\sigma^2(\hat{\Delta}_{pairwise})$.

We examined a large number of regressions and one approximation stood out of the rest due to its efficiency, low average error and other minor indications. Table 4.1 shows the coefficients of the approximation for the three types of scoring matrices.

For example, the approximation for DNA variances is

$$\tilde{\sigma}^2(\hat{\Delta}_{pairwise}) = \frac{\hat{d}_{YZ}^{1.3182}}{\sigma^2(\hat{d}_{YZ})^{0.3026}} \cdot \frac{(\sigma^2(\hat{d}_{XY}) + \sigma^2(\hat{d}_{XZ}))^{1.0933} (\sigma^2(\hat{d}_{XY})\sigma^2(\hat{d}_{XZ}))^{0.1181}}{(\hat{d}_{XY} + \hat{d}_{XZ})^{1.2449}}$$

Type	$\hat{d}_{XY}+\hat{d}_{XZ}$	$\sigma^2(\hat{d}_{XY})+\sigma^2(\hat{d}_{XZ})$	$\hat{d}^2_{YZ}$	$\sigma^2(\hat{d}_{YZ})$	$\sigma^2(\hat{d}_{XY})\sigma^2(\hat{d}_{XZ})$	error	dim
Day	-1.3090	1.0435	0.6895	-0.3339	0.1590	0.087	2.13
DNA	-1.2449	1.0933	0.6591	-0.3026	0.1181	0.098	2.13
JTT	-1.2921	1.0978	0.6741	-0.3065	0.1144	0.080	2.10

Table 4.1: $\sigma^2(\hat{\Delta}_{pairwise})$: coefficients of the regression on the logarithms for the three types of scoring matrices. The error column shows the mean error, which by virtue of being a regression on logarithms is very close to the relative error.

Readers familiar with numerical analysis will find an analogy between the approximation presented here and standard approximations for transcendental functions. For example, it is customary to approximate e^x through a quotient of polynomials $p(x)/q(x)$, for some limited range of x.

The relative error is in all the three cases less than 10%. Furthermore, since we normally use the square root of the variance, the relative error is in such cases half of the indicated. The last column indicates the dimension of the approximation which should be 2 in perfect conditions, and is indeed quite close. The fact that very different matrices have very similar coefficients, the low error and the almost correct dimensionality reassures us of the quality of the approximation.

To test the accuracy/applicability of the approximation, as well as the other two methods to obtain the variance, we compared the 95 and 99% confidence level obtained using the appropriate number of standard deviations to the actual percentage of correct decisions obtained in a simulation over 400,000 protein triplets generated as described above. The results are shown in Table 4.2.

	$k = 1.960$	$k = 2.576$
$\lvert\hat{\Delta}_{triplet}-\Delta\rvert > k\cdot\sigma(\hat{\Delta}_{triplet})$	0.95129 ± 0.00067	0.99062 ± 0.00030
$\lvert\hat{\Delta}_{pairwise}-\Delta\rvert > k\cdot\sigma_{bootstrap}(\hat{\Delta}_{pairwise})$	0.9511 ± 0.0020	0.99001 ± 0.00091
$\lvert\hat{\Delta}_{pairwise}-\Delta\rvert > k\cdot\sigma(\hat{\Delta}_{triplet})$	0.94641 ± 0.00070	0.98896 ± 0.00032
$\lvert\hat{\Delta}_{pairwise}-\Delta\rvert > k\cdot\tilde{\sigma}(\hat{\Delta}_{pairwise})$	0.94808 ± 0.00069	0.98953 ± 0.00032
$\lvert\hat{\Delta}_{pairwise}-\Delta\rvert > k\cdot\sigma_{ind}(\hat{\Delta}_{pairwise})$	0.98137 ± 0.00042	0.99774 ± 0.00015

Table 4.2: Verification of accuracy of confidence intervals: comparison among the different methods to estimate the variance of the two estimators $\hat{\Delta}_{triplet}$ and $\hat{\Delta}_{pairwise}$, resulting from a simulation using updated Dayhoff matrices over 400,000 proteins triplets, except for the bootstrapping method, based on 40,000 samples. The first column tests the 95% confidence interval, the second the 99% confidence interval.

As expected, the ML estimator over the entire triplet (first row) yields a precise variance estimate. On the other hand, we see that assuming independence for the estimation of the variance (last row) leads to very inaccurate confidence intervals. Estimating the variance of $\hat{\Delta}_{pairwise}$ by bootstrapping (10,000 re-samples) gives good confidence intervals, but the procedure is even more computationally intensive than $\hat{\Delta}_{triplet}$, and therefore of little practical use in the present context. Using $\tilde{\sigma}^2(\hat{\Delta}_{pairwise})$ in conjunction with the variance of the

ML estimator works remarkably well (third and fourth row). And surprisingly, applying the numerical approximation (fourth row) happened to give slightly more accurate results than the exact triplet variance (third row).

Finally, we compared the different estimators on real biological sequences, using data obtained from the OMA orthologs project (Chapter 6). Triplets of orthologous sequences from various eukaryotes were randomly selected and aligned using the multiple sequence alignment package from *Darwin* (Gonnet et al., 2000). All positions containing gaps were excluded, and variances were then estimated on the ungapped triplets using the various Estimators (Fig. 4.3). The variance estimates from the approximation formula deviate very little from the results obtained by the two more expensive methods — for simulated as well as empirical alignments. Additionally, the plots illustrate the high correspondence between the results from the ML estimation and the bootstrapping, and show that the estimator based on an assumption of independence often yields overestimates of the variance. The difference between simulated and empirical data probably arises from the limitations of the Markovian model of evolution. Worth noticing is that the agreement of our estimator with bootstrapping is comparable to the one of the ML variance estimator: this implies that our approximation has a similar robustness when applied to real data.

Applications

In the following, we provide two examples of applications that benefit from the increase in statistical power of the estimator $\hat{\Delta}_{pairwise}$ enabled by the approximation: detection of asymmetric evolution and identification of the closest relative in a set of homologs. Furthermore, in Chapter 5, we show how our result can be used in the context of paralogy detection.

We first define three indicator functions that will be used in these comparisons. They decide whether the pair of proteins X, Y is significantly closer than X, Z at the confidence level expressed by the number of standard deviations k. The first and second ones both use the estimator $\hat{\Delta}_{pairwise}$, but the first definition uses as variance of the estimate the upper bound that is obtained by assuming independence of $\hat{d}_{XY}$ and $\hat{d}_{XZ}$ (Section 4.2.1), whereas the second use the approximation $\tilde{\sigma}^2(\hat{\Delta}_{pairwise})$ of the variance. The third indicator function uses the estimator $\hat{\Delta}_{triplet}$.

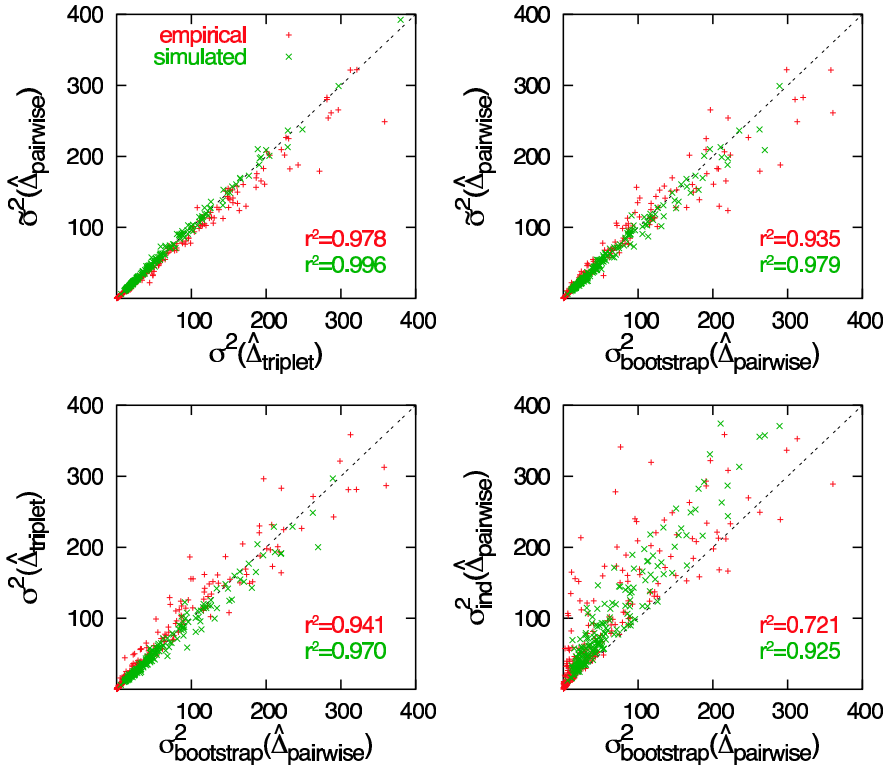


Figure 4.3: Scatter plots comparing the variance estimators The upper-left plot shows the strong agreement between $\sigma^2(\hat{\Delta}_{triplet})$ and our approximation $\sigma^2(\hat{\Delta}_{pairwise})$. From the upper-right and the lower-left plots, it can be seen that both have similar correlation with $\sigma^2_{bootstrap}(\hat{\Delta}_{pairwise})$. Finally, the lower-right plot confirms that variance estimation under the assumption of independence can yield a large overestimation of the correct variance.

$$\text{closer}_{ind}(X,Y,Z,k) = \begin{cases} \text{true} & \text{if } \hat{\Delta}_{pairwise} < -k \cdot \sigma_{ind}(\hat{\Delta}_{pairwise}) \\ \text{false} & \text{otherwise} \end{cases}$$

$$\text{closer}_{\text{app}}(X,Y,Z,k) = \begin{cases} \text{true} & \text{if } \hat{\Delta}_{pairwise} < -k \cdot \tilde{\sigma}(\hat{\Delta}_{pairwise}) \\ \text{false} & \text{otherwise} \end{cases}$$

$$\text{closer}_{\text{triplet}}(X,Y,Z,k) = \begin{cases} \text{true} & \text{if } \hat{\Delta}_{triplet} < -k \cdot \sigma(\hat{\Delta}_{triplet}) \\ \text{false} & \text{otherwise} \end{cases}$$

Asymmetric evolution After a gene duplication, the two copies can evolve independently. It has been suggested that in many cases, one duplicate maintains the ancestral function while the other is free to evolve and acquire novel functionality (Ohno, 1970). This scenario implies that the protein with conserved functionality will undergo less sequence evolution than the one exploring new functionalities.

Detecting this asymmetric evolution after duplication is an important factor not only for function prediction or orthologs assignment, but also for bringing new insights in our understanding of genome evolution in general (e.g. Van de Peer et al. 2001; Dermitzakis and Clark 2001; Li and Tsoi 2002; Wagner 2002).

In order to identify cases of asymmetric evolution, one typically considers three sequences – the two duplicates (Y and Z) and an out-group (X). Several methods have been developed to test the significance of the unequal lengths of the branches leading from the common ancestor to the two duplicated sequences. Tests on simulated and real data from *Arabidopsis thaliana* for two of such methods have suggested very low statistical power to detect asymmetric evolution of duplicates (Seoighe and Scheffler, 2005).

The closer indicator function can be used to detect asymmetric evolution. With d_{XY} being the distance from the out-group to the closer of the two duplicates and d_{XZ} the distance to the other one, $\text{closer}(X,Y,Z,k)$ decides if the two duplicated proteins have evolved at significantly different rates. The parameter k can be chosen to reflect the confidence level, e.g. 1.96 for the 95% level.

We tested the method using all three variants of closer ($k = 1.96$) on a protein set from a recent publication about whole genome duplication in *S. cerevisiae* (Kellis et al., 2004). From a set of 450 genes pairs that arose by whole genome duplication, they report 115 cases of one paralog evolving at least 50% faster than the other paralog. The position of the ancestral gene was determined by an out-group gene from *K. waltii*. Additionally, a set of 76 gene pairs is given where at least one of the *S. cerevisiae* genes evolved at least 50% faster than the *K. waltii* homolog.

The results are summarized in Fig. 4.4. We first discuss the differences among three variants of closer. As expected, the overestimation of the variance of the estimator in $\text{closer}_{\text{ind}}$ considerably reduces the cases of asymmetry detected in

comparison with $\text{closer}_{\text{app}}$. As for $\text{closer}_{\text{app}}$ and $\text{closer}_{\text{triplet}}$, they agree on 400 of 450 cases, with 21 cases only reported by $\text{closer}_{\text{app}}$ and 29 only by $\text{closer}_{\text{triplet}}$. This discrepancy results from the error introduced by the approximation for the estimation of the variance of $\hat{\Delta}_{pairwise}$, but mostly from the inherent differences in the predictions of the two estimators $\hat{\Delta}_{pairwise}$ and $\hat{\Delta}_{triplet}$.

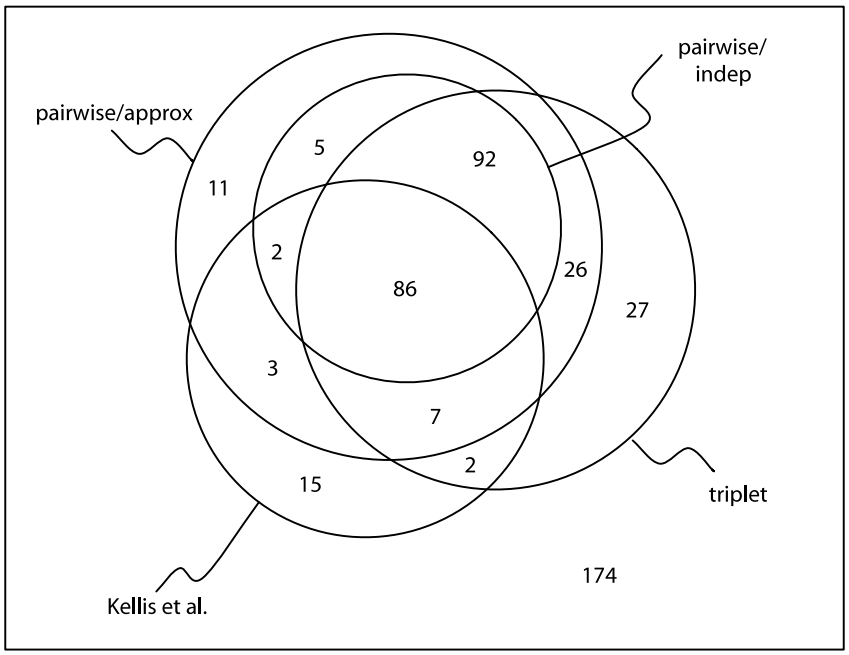


Figure 4.4: Detection of Asymmetric Evolution. Comparison between the results of Kellis et al. and the three variants of closer, with $k = 1.96$. The circles separate cases of significant asymmetry (inside) from insignificant asymmetry (outside). For instance, there were 92 cases where all three variants of closer reported significant asymmetry, while the method of Kellis et al. did not detect significant asymmetry. Note that the 0.05 significance level applies to independent tests on each triplet; a more global conclusion would require a multiple test correction.

If we now compare the predictions of Kellis and colleagues with our results, it appears that in 98 out of 115 cases, their prediction of asymmetric evolution could be confirmed by $\text{closer}_{\text{app}}$, while with the remaining 17 pairs, our method did not support the asymmetry prediction. It is remarkable, however, that all these 17 pairs belong to the group of the 76 pairs with a fast evolving *K. waltii* homolog. We believe that the higher uncertainty in locating the origin of the triplet arising from a longer branch to the out-group may cause rate-based methods as used in (Kellis et al., 2004) to wrongly conclude asymmetric divergence. As opposed to that, the distance-based methods presented here, by incorporating the variance of the estimates explicitly, take the uncertainty about the point

of origin into account, and therefore give more conservative predictions in these cases.

Furthermore, $\text{closer}_{\text{app}}$ found 134 additional cases of asymmetry among the remaining 335 gene pairs in the data set. Together with the 98 cases above, this results in 51.6% of all genes arising from the genome duplication event. This is clearly more than the 5% that could be expected from random chance and agrees with previous studies were significant amounts of asymmetrically evolving duplicates have been reported (e.g. Blanc et al. 2000; Conant and Wagner 2003).

Closest homolog without phylogenetic reconstruction The identification of the closest relative of a protein (or gene) in a set of homologs traditionally requires the reconstruction of the corresponding phylogenetic tree. However, building gene trees remains a time consuming and error-prone task, thus methods based on pairwise evolutionary distance estimates are attractive. In this section, we show that using the variance approximation presented above can boost the statistical power of PAM distance comparisons to determine the closest homolog.

In simple contexts, or when accuracy is not a concern, the problem of identifying the closest relative can be solved reasonably well by coarse approaches, such as the top blast hit, or even the sequence with highest percentage identity. As the number of proteins grows larger and the number of homologs with similar distances increase, these methods show their limits. Indeed, it has been previously shown that the top blast hit is often not the closest relative (Koski and Golding, 2001). At least two ideas lead to better results: the use of evolutionary distance estimates such as PAM distances, and accounting for confidence intervals, so that whenever there is not enough information to reliably discriminate among several distances, all of them are kept, presumably for further analysis.

Since the comparison of the methods requires precise and unbiased knowledge of the closest homolog, we use simulated data generated in the same way as in the section above, according to the PAM model. Families of homologs were created through mutation and duplication following random phylogenetic trees (Fig. 4.5) with the following properties: (i) each branch has a random mutation rate from a uniform distribution between 0 and 1, (ii) duplication occurs only along the leftmost branch, at random intervals, on average about every 6 PAM units, (iii) the generation is performed in 60 steps and results in trees with an average number of leaves of 13.04 ($\sigma = 3.1$). The very asymmetric duplication process is used to explore efficiently the parameter space, both in terms of distance magnitude to the closest homolog as in the number of homologs with similar distances.

For each protein X belonging to such a family, the closest homolog predictions using the following three criteria were compared to the actual closest ho-

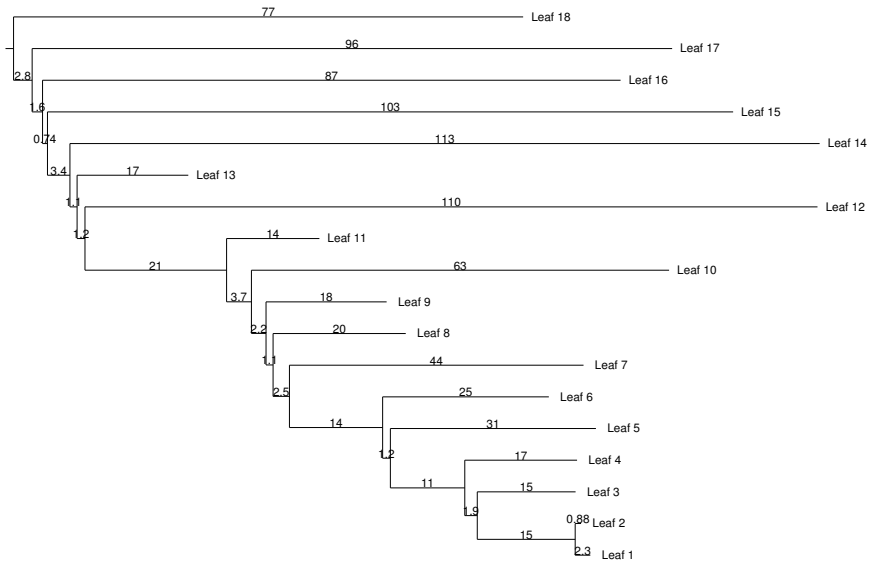


Figure 4.5: Tree randomly generated for closest homolog simulation Example of a random tree (see text for description of the procedure) used to compare the different methods to infer the closest homolog to each leaf. Distances indicated are in PAM units.

molog. The first one computes the subset of homologous sequences H that align with X with score higher than a particular fraction of the top score.

$$Set_1 = \left\{ Y \in H \;\middle|\; \text{Score}(X,Y) \geq (1-k_1) \cdot \max_{Z \in H}(\text{Score}(X,Z)) \right\}$$

The second method computes the set of closest homologs, without using our variance approximation, formally

$$Set_2 = \{ Y \in H \mid \nexists\, Z \in H, Z \neq Y, \text{closer}_{\text{ind}}(X,Z,Y,k_2) \}$$

The third method computes the set of closest homologs using our approximation, formally

$$Set_3 = \left\{ Y \in H \;\middle|\; \nexists\, Y \in H, Z \neq Y, \text{closer}_{\text{app}}(X,Z,Y,k_3) \right\}$$

The cut-off parameters k_1, k_2, k_3 can be set according to the desired level of confidence. At $k = 0$, only the top score, respectively the shortest expected distance, is returned. Higher k values correspond to more conservative predictions, with increasing number of closest homolog candidates. For the evaluation of the methods, we vary k_1 between 0 and 0.25, while k_2, k_3 are varied between 0 and 3. Note that only k_3 corresponds to the number of standard deviations from the expected value.

The resulting curves are presented in Fig. 4.6. At low cut-off values, all three methods perform similarly, but as k increases, the method using $\text{closer}_{\text{app}}$ gives better results.

4.2.3 Conclusions About Variance of Distance Estimator

Computing the difference of two evolutionary distances that are not independent is a common operation in an increasing number of bioinformatics analyses. We presented and compared two estimators for the difference of two evolutionary distances in a triplet of homologs, one estimator based on pairwise distance estimates and the ML estimator. Surprisingly, the estimator based on pairwise distance is almost as powerful as the ML estimator. But in terms of time complexity, it scales much better than the ML estimator and is therefore better suited at large-scale analyses. However, since its variance is not easy to estimate, we introduced a numerical approximation that allows the computation of accurate confidence intervals. Finally, we showed how these results can be used to test for asymmetrical evolution, and to identify the closest relative of a sequence in a group of homologs without phylogenetic reconstruction.

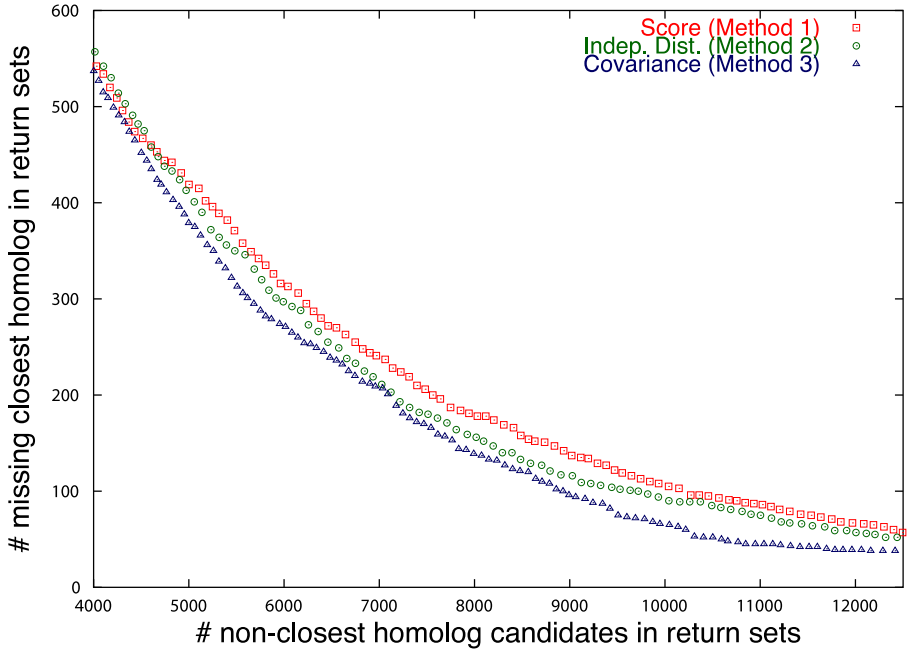


Figure 4.6: Identification of the closest homolog: comparison between methods using alignment score (1), distance with assumption of independence (2) and distance using our variance approximation (3), on simulated data.

4.3 Covariance of distances estimated pairwise

This section is joint-work with Manuel Gil. It was published in Dessimoz and Gil (2008)

In the previous section, we have introduced a numerical approximation for the constrained case of the covariance of two OPA distances involving a common sequence (i.e. on a triplet of sequences), for empirical substitution models such as PAM or JTT. In this section, we present an estimator for the covariance of ML distances estimated from OPAs that works on triplets and quartets of sequences. This solves the problem of sets of sequences of arbitrary size, because each covariance involves at most four sequences at a time. Thus, the full covariance matrix is naturally obtained through quartet analysis. In the following, we first derive our estimator, then analyze its performances in terms of bias and variance. Finally, we compare the results obtained on triplets of sequences to our previous work.

4.3.1 Methods

Covariance of Distances from OPAs

In this section we derive a covariance estimator for ML distances from OPAs.

Preliminaries The columns of an MSA are a consistent hypothesis of character homology for a set of sequences. With OPAs on the other hand, we have the problem that for a set sequences, the resulting pairwise alignments are not always consistent in their inference of the homologous characters. Fig. 4.7 depicts an example. Let $s_{i,j}$ be the character at position j in a sequence s_i. Only characters in bold, for example $\{s_{1,1}, s_{2,1}, s_{3,1}, s_{4,1}\}$, are consistently aligned in the OPAs. We call such a consistent set of characters an *anchor*. On the other hand, $s_{1,2}$ is aligned to $s_{2,2}$ and to $s_{3,2}$, so in a consistent situation it would follow that $s_{2,2}$ and $s_{3,2}$ should be aligned, but it is not the case.

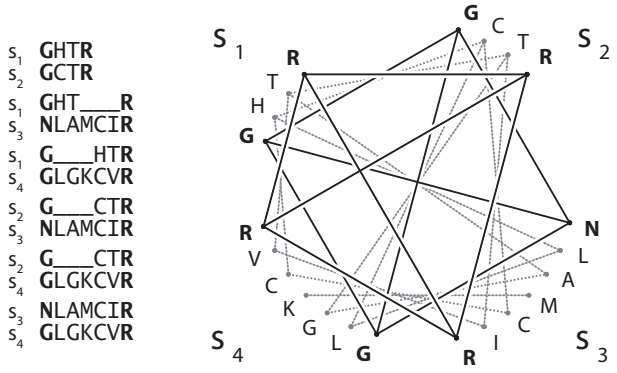


Figure 4.7: Example of anchors: the six pairwise alignments of four example sequences (left) and the corresponding graph-representation (right). The consistent positions are in bold.

Given m sequences, the anchors can formally be defined as follows: Define a graph $G(\{s_i\})$ with $\sum_i^m |s_i|$ vertices labeled by $s_{i,j}$. We join vertices s_{i_1,j_1} and s_{i_2,j_2} if the corresponding characters are aligned in the $\mathrm{OPA}(s_{i_1}, s_{i_2})$. The set of anchors for the $\binom{m}{2}$ OPAs is defined as the set of all cliques of size $\binom{m}{2}$ in $G(\{s_i\})$. By construction, the sub-alignments induced by the anchors define an MSA. In the derivation of our covariance estimator, we assume that the anchor positions are correctly aligned. For the non-anchor positions, we know that some proportion is wrongly aligned in at least one of the $\binom{m}{2}$ OPAs. We do not know, though, which positions and in which alignments. In this paper we are interested in the covariance of distances from two OPAs. In each case the anchors are determined from the particular sequences involved in the corresponding covariance estimation. If the two OPAs share a sequence $m = 3$, otherwise $m = 4$.

The following pseudocode shows how the anchors can be found for $m = 4$. It uses a function $\mathtt{M}(s_{i_1}, s_{i_2}, j_1)$ which returns the index j_2 of the character s_{i_2,j_2} of s_{i2} aligned to s_{i_1,j_1} in $\mathrm{OPA}(s_{i_1}, s_{i_2})$.

```
Anchors ← {}
for j1 ← 1 to length(s1) do
    j2 ← M(s1,s2,j1); j3 ← M(s1,s3,j1); j4 ← M(s1,s4,j1)
    if M(s2,s3,i2) = j3 and M(s2,s4,i2) = j4 and M(s3,s4,i3) = j4 then
        Anchors ← Anchors ∪ {[s1,j1, s2,j2, s3,j3, s4,j4]}
    end
end
```

Formulation of the Covariance Estimator Let $p(X_j, d)$ denote the probability of a homologous character-pair X_j for the j-th OPA when the distance is taken to be d. We assume that the gap-positions have been removed from the alignments and that the j-th OPA has length n_j. Denote $\hat{d}_j$ the distance obtained by ML and d_j the true distance. It is well known from ML theory (see e.g. Rice 2001) that under appropriate smoothness conditions, the variance of $\hat{d}_j$ is

$$V_j = \frac{1}{n_j}\left(E\left[-\frac{\partial^2}{\partial d_j^2}\ln\left(p(X_j, d_j)\right)\right]\right)^{-1}. \tag{4.1}$$

Let the score function for the j-th OPA be

$$u_j(d) = \sum_{l=1}^{n_j}\frac{\partial}{\partial d}\ln\left(p(x_{j,l}, d)\right), \tag{4.2}$$

where $x_{j,l}$ is the realization of X_j at position l. To abbreviate, we set $u_{j,l}(d) = \frac{\partial}{\partial d}\ln\left(p(x_{j,l}, d)\right)$. As mentioned by Susko (2003), ML results yield

$$\sqrt{n_j}(\hat{d}_j - d_j) = -\sqrt{n_j}V_j u_j(d_j) + o_p(1). \tag{4.3}$$

Based on equation (4.3) we derive now an expression for the covariance of two distance estimates $\hat{d}_j$ and $\hat{d}_k$. Along this paper, variables with a superscript A refer to anchors, N refer to non-anchors. Since virtually all Markovian models of evolution assume independent positions, we can split the score functions in a part corresponding to the anchor positions and a non-anchor part:

$$u_j(d) = u_j^A(d) + u_j^N(d). \tag{4.4}$$

We assume that the sums in $u_j^A(d)$ and $u_k^A(d)$ are ordered such that $u_{j,l}^A(d)$ and $u_{k,m}^A(d)$ are part of the same anchor iff $l = m$. Since, up to high order terms, $(\hat{d}_j - d_j)$ is equal to $-V_j u_j(d_j)$ we can write for the covariance of $\hat{d}_j$ and $\hat{d}_k$

$$cov\left(\hat{d}_j, \hat{d}_k\right) = cov\left((\hat{d}_j - d_j), (\hat{d}_k - d_k)\right) \tag{4.5}$$

$$\approx cov\left(-V_j\left\{u_j^A(d_j) + u_j^N(d_j)\right\}, -V_k\left\{u_k^A(d_k) + u_k^N(d_k)\right\}\right) \tag{4.6}$$

$$\begin{aligned} = V_j V_k \; &\left\{cov(u_j^A(d_j), u_k^A(d_k)) + cov(u_j^N(d_j), u_k^N(d_k))\right. \\ &\left. + cov(u_j^A(d_j), u_k^N(d_k)) + cov(u_j^N(d_j), u_k^A(d_k))\right\}. \end{aligned} \tag{4.7}$$

Correlations between distance arise from common mutation events (on common branches on the "true" tree). As mentioned above, positions in a sequence are stochastically independent from one another. We assume that the anchors are correctly aligned. Consequently, characters in the anchor and non-anchor parts cannot be homologous to each other. Therefore $cov(u_j^A(d_j), u_k^N(d_k))$ and $cov(u_j^N(d_j), u_k^A(d_k))$ are both zero. The expression becomes

$$V_j V_k \left\{cov(u_j^A(d_j), u_k^A(d_k)) + cov(u_j^N(d_j), u_k^N(d_k))\right\} \tag{4.8}$$

$$= V_j V_k \left\{ cov\left(\sum_{l=1}^{n_A} u_{j,l}^A(d_j), \sum_{m=1}^{n_A} u_{k,m}^A(d_k)\right) + cov\left(\sum_{l=1}^{n_j - n_A} u_{j,l}^N(d_j), \sum_{m=1}^{n_k - n_A} u_{k,m}^N(d_k)\right)\right\} \tag{4.9}$$

$$= V_j V_k \left\{ \sum_{l=1}^{n_A}\sum_{m=1}^{n_A} cov\left(u_{j,l}^A(d_j), u_{k,m}^A(d_k)\right) + \sum_{l=1}^{n_j - n_A}\sum_{m=1}^{n_k - n_A} cov\left(u_{j,l}^N(d_j), u_{k,m}^N(d_k)\right)\right\}, \tag{4.10}$$

where n_A is the number of anchors. Because of the correctness assumption of the anchors, all pairs that are not part of the same anchor are non-homologous, and therefore, their covariance is zero, i.e. for $l \neq m$, $cov\left(u_{j,l}^A(d_j), u_{k,m}^A(d_k)\right) = 0$ and we get

$$V_j V_k \left\{ \sum_{l=1}^{n_A} cov\left(u_{j,l}^A(d_j), u_{k,l}^A(d_k)\right) + \sum_{l=1}^{n_j - n_A}\sum_{m=1}^{n_k - n_A} cov\left(u_{j,l}^N(d_j), u_{k,m}^N(d_k)\right)\right\}. \tag{4.11}$$

We assumes that the $u_{j,l}^A(d_j)$ are i.i.d. We denote the corresponding random variables U_j^A. The assumption is justified due to the Markov model and the

correctness assumption of the anchors. As to the $u^N_{j,l}(d_j)$ some proportion may be homologous, but we do not know which one. Determining the homologous pairs would solve the problem of MSA construction (known to be hard and not our goal here). Instead, we take the working assumption that the $u^N_{j,l}(d_j)$ and $u^N_{k,m}(d_k)$ do not covary. With the two assumptions the expression of the covariance approximation becomes:

$$cov\left(\hat{d}_j, \hat{d}_k\right) \approx V_j V_k n_A cov(U^A_j, U^A_k). \tag{4.12}$$

By using the form of equation (4.12), we obtain an estimator for the covariance. The variance V_j is estimated by

$$\hat{V}_j = \frac{1}{n_j}\left(\frac{1}{n_j}\sum_{l=1}^{n_j} -\frac{\partial^2}{\partial d_j^2}\ln\left(p(x_{j,l}, \hat{d}_j)\right)\right)^{-1}. \tag{4.13}$$

The estimate for the covariance of the anchor part is the well-known unbiased estimator

$$\widehat{cov}(U^A_j, U^A_k) = \frac{1}{n_A - 1}\sum_{l=1}^{n_A}(u^A_{j,l}(\hat{d}_j) - \bar{u}^A_j(\hat{d}_j))(u^A_{k,l}(\hat{d}_k) - \bar{u}^A_k(\hat{d}_k)), \tag{4.14}$$

where $\bar{u}^A_j$ denotes the sample mean.

Simulation Methods

To evaluate the performance of the covariance estimator we performed a Monte Carlo simulation on quartets and compared our estimator to the sample covariance (also referred to as the Monte Carlo covariance).

Sampling of Quartets The quartets were sampled uniformly from a variance weighted least-squares (WLS) tree on 352 species. The WLS tree was inferred by the *LeastSquaresTree* function in Darwin (Gonnet et al., 2000). To obtain the input distance and variance matrices for *LeastSquaresTree* we used data from the OMA project (Chapter 6). The inter-species distances were determined as average PAM distances over sets of groups of orthologs. A total of 100 quartets were sampled, each one contributing one data-point to the plots shown here.

Simulation Procedure for One Quartet To explore the branch-length space, while preserving the relative branch-length structure given by the WLS tree we applied an uniformly distributed U(0.5,2) expansion/contraction factor on each quartet. Then, we generated 40,000 times three random sequences of length

$m = \{200, 500, 800\}$ and mutated each of them along the dilated model quartet. We assumed a Markovian model of evolution using the updated PAM matrices (Gonnet et al., 1992) and introduced gaps of Zipfian distributed length (Benner et al., 1993).

We applied our covariance estimator on each of the 40,000 quartets and estimated its expected value and variance to compare it against the sample covariance which we also refer to as *Monte Carlo covariance*. In the analysis of the results, we treated the sample covariance as a reference value, as it constitutes an unbiased estimator for the true covariance. The biases reported in the result section are defined as the estimate of the expected value of our covariance estimator minus the Monte Carlo covariance. Note that being an estimator itself, the sample covariance's variance had also to be taken into account in the analysis of the results.

4.3.2 Results and Discussion

In the following, we present and analyze the performances of the estimator for the covariance of two distances. For this purpose, it is informative to analyze the results separately for the following three different underlying topological relations, illustrated in Fig. 4.8:

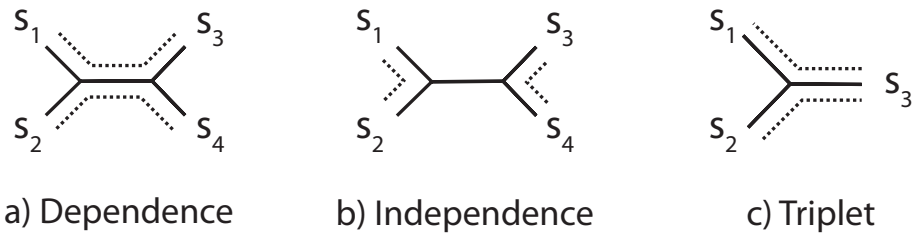


Figure 4.8: Possible topological relations of sequences: for two pairwise distances, one can distinguish three possible underlying topological configurations relating them. If they are estimated between four sequences, there are two possible configurations. Either they share some common evolution (a) or they are independent (b). In the third configuration, the two distances are estimated from two OPAs that share a sequence (c).

Case of Dependence: the two distances are estimated between four distinct sequences, and they have some evolution in common (i.e. the two distance involve a common branch on the tree). With such an evolutionary history, the two distances estimates covary positively.

Case of Independence: the two distances are estimated between four distinct sequences, but they have no evolution in common (i.e. the two distance involve

distinct branches on the tree). This case is informative, because a central assumption in most evolutionary models is that evolution on different branches is independent (Felsenstein, 2004a). With no branch in common, the distances should not covary (Bulmer, 1991). Thus, such a topology can be used to test the estimators as negative control.

Case of Triplet: the two distances involve a common sequence, and have some evolution in common. This case is of special interest, because we have just presented earlier in this chapter an alternate estimator for this particular case using a different approach 4.2. Thus, we can compare our results to this approach, hereafter called "the numerical approximation".

Note that the covariances are estimated using the same algorithm in all three cases: we only distinguish them from each another for the purpose of this analysis.

To assess the performance of the covariance estimator, it was compared against the Monte Carlo covariance estimator. In short, each point shown in the figures was obtained from 40,000 sets of sequences mutated along a random quartet subtree of the tree of life (see Section 4.3.1 above). That way, the evaluation is based on tree samples that are distributed as closely as possible to real biological data. To account for gene families with varying rates, each quartet was scaled with a random factor uniformly distributed between 0.5 and 2. Note that results corresponding to very large distance constitute extreme cases; for instance, when sequences are 150 PAM units apart, each position has, on average, mutated 1.5 times.

Fig. 4.9a shows the mean of our estimator versus the Monte Carlo estimator in nine scatterplots arising from combining the topologies mentioned above (rows) with three different sequence lengths (columns). In the case of dependence, the first row, we see that our estimator lies in about 80% of the cases within the 95% confidence interval of the Monte Carlo estimator. In the case of independence, both estimators are close to zero, though our estimator shows a minor upward bias in some cases. The third row gives the result of both the covariance estimator introduced here, as well as the numerical approximation from our other estimator (Section 4.2). Here, we see that though the former performs well in cases of lower covariance values, it shows a clear downward bias in cases of larger covariances. The numerical approximation does not present any apparent sign of bias, which is hardly surprising, given that it was obtained through regression. What is however surprising, is that, given its simple structure, it performs better than the covariance estimator, which takes into account more data and is backed by a more detailed model.

It is instructive to compare the absolute bias of the covariance estimator to the well-known ML variance estimator (see e.g. Rice 2001). As can be seen in Fig. 4.9b, the ML variance is also biased for high variance values. We conjecture

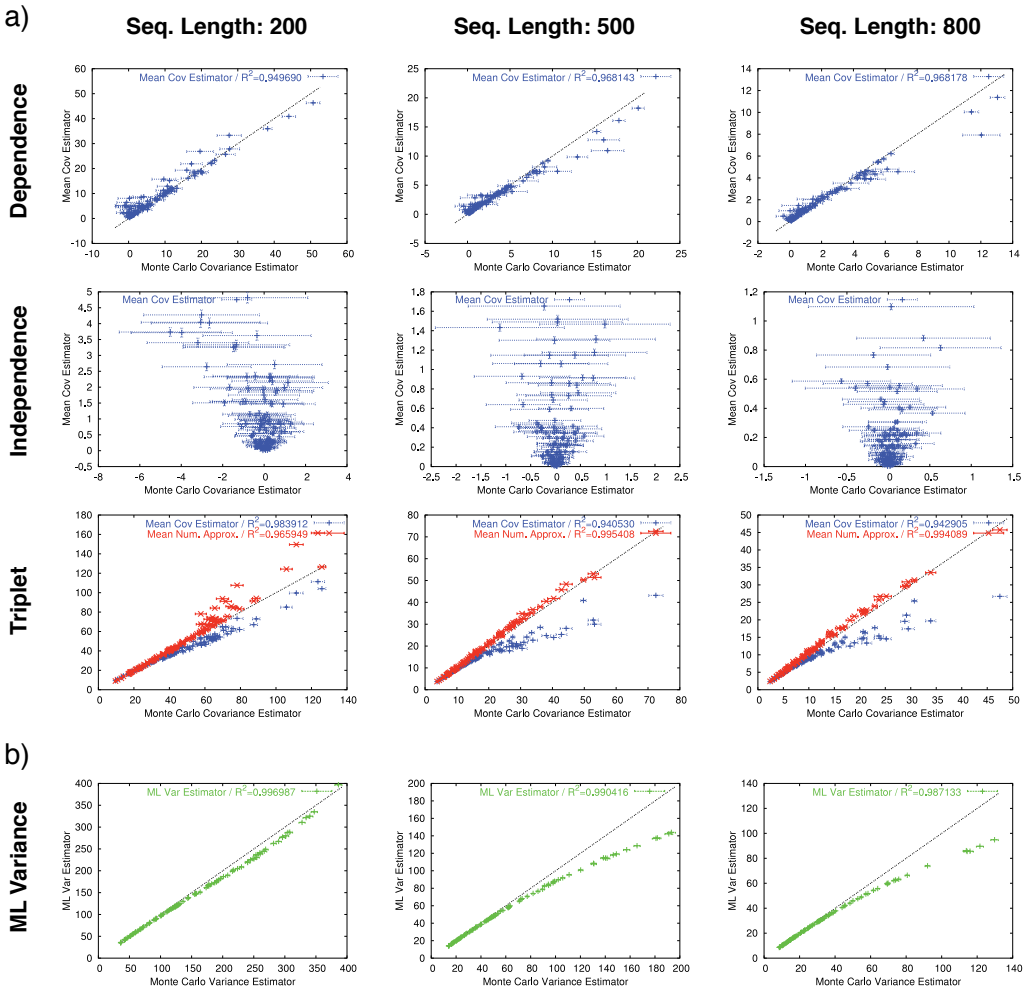


Figure 4.9: Comparison of the Covariance Estimator and the ML Variance Estimator with their Monte Carlo Counterparts: error-bars indicate 95% confidence intervals. **a)** Monte Carlo covariance estimator vs. average of the covariance estimator for sequence lengths of {200, 500, 800} AA. In the dependence case, the estimator appears unbiased in most cases. In the independence case, the estimator shows a slight upward bias, but the absolute values are close to zero. In the triplet case, a downward bias with increasing covariance is visible. **b)** Monte Carlo variance estimator vs. average of ML variance estimator. A downward bias with increasing variance is visible.

that this is mainly due to misaligned positions, which cause model violations in the parameter estimation. This problem is also likely to affect the covariance estimator. Even more directly, the ML variance estimator is a factor in the expression of the covariance estimator (see Section 4.3.1), so any error in the ML variance is propagated to the covariance estimator. At this point, improving the ML estimator for cases of high divergence will require better alignments (see Chapter 3 for a study of the influence of alignment errors on distance estimation), or an explicit modeling of the misaligned positions, which is beyond the scope of the present work.

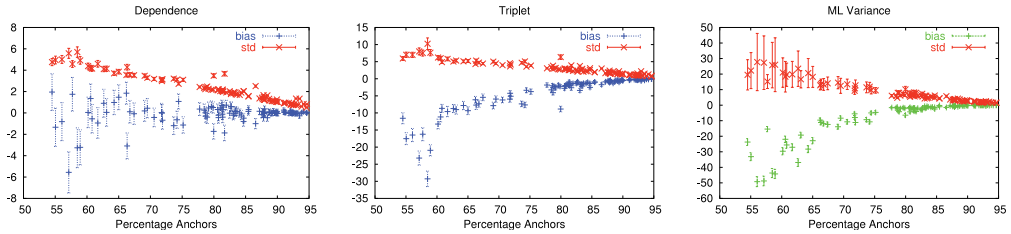


Figure 4.10: Bias and standard deviation of the covariance and ML variance estimators: average percentage of anchors vs. bias and standard deviation of the covariance estimator for sequence length of 500 AA. Error-bars indicate the 95% confidence intervals. The bias increases with decreasing fraction of anchors. The bias is smaller than the standard deviation when percentage of anchors is greater than 65% (dependence), 80% (triplet) and 75% (ML variance).

Further, to put the bias of the covariance estimator into perspective, we compared it to the standard deviation of the estimator. Fig. 4.10 presents the bias and standard deviation as function of the average number of anchors for sequence length of 500. The anchors are the positions that are consistently aligned in the OPAs involved (see Section 4.3.1 for precise definition). Both bias and standard deviation strongly depend on the fraction of anchors, which can be thought of as a measure of alignment quality. Fig. 4.11 depicts the dependency between percentage of anchors and average distance. As one would expect, the fraction of anchors decreases as divergence increases. For a fraction of anchor positions below 60%, the average of the two distances involved in the covariance computation is always greater than 150 PAM.

In Fig. 4.10, we first consider the bias and standard deviation for the case of dependence. When the fraction of anchor positions is above 60% (this is the case for approximately 85% of the quartets of sequences in families of orthologs in OMA (Chapter 6), data not shown), the bias is far smaller than the standard deviation, and is therefore likely to have little negative impact in practice.

In the case of triplets, the bias exceeds the standard deviation already when the fraction of anchors is about 80%. The ML variance estimator has this transition around 75% of anchors. In the case of independence, where we expect our

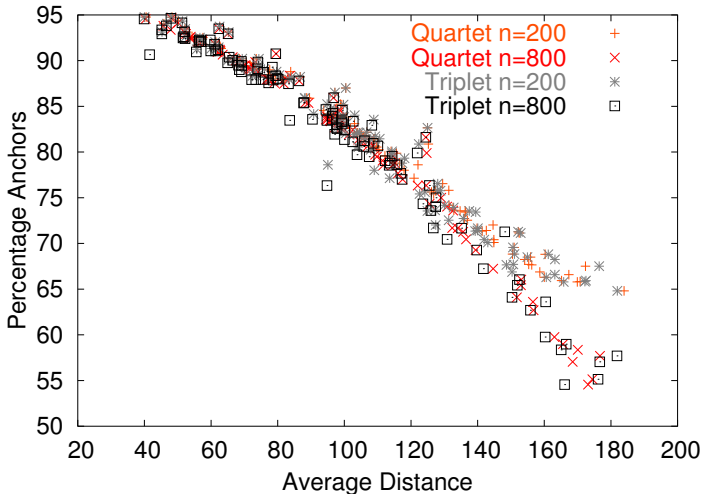


Figure 4.11: Relation between distance and percentage of anchors: Horizontal axis: average of the two distances for which the covariance has been estimated. Vertical axis: Average percentage of anchors. The *Quartet* labels refer to the dependence case. The fraction of anchors decreases with increasing distance.

covariance estimator to be zero, its bias is always much smaller than its standard deviation (data not shown).

Most applications of the covariance estimator involve the covariance matrix. Let $\hat{A}$ be an approximation to the matrix A[1]. Fig. 4.12 shows the relative error of the 2×2 variance-covariance matrices computed with the ML variance estimator in the diagonal entries and our covariance estimator in the off-diagonal entries, and the same 2×2 matrices with only diagonal entries. The plots show that for the dependence case the the matrices with both covariance and ML variance estimators have a equal or lower relative error than the matrices with the ML variance only, except for a few cases in the region with a high fraction of anchors. In the triplet case, the variance-covariance matrices have always lower error then variance matrices. Finally, in the case of independence, the matrices with covariance do not always have lower relative error, but this is expected, because the true covariance is null in this special case.

4.3.3 Conclusions About Covariance Estimator

We have presented a method to estimate the covariances of distances estimated from pairwise alignments. It does not require the construction of MSAs, which are hard to compute and therefore are only approximated in practice. Furthermore, it does not rely on phylogenetic trees as it is the case with covariance

[1]We use the expression $\frac{||\hat{A}-A||_2}{||A||_2}$ as relative error of the estimator $\hat{A}$, where $||\cdot||_2$ denotes the two-norm.

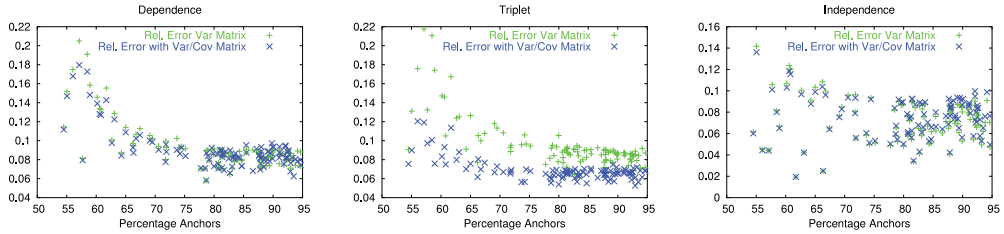


Figure 4.12: Relative error of covariance matrix: average relative error of variance matrices and variance/covariance matrices for a sequence length of 500 AA. Dependence and independence cases: variance matrices and variance-covariance matrices have comparable error. Triplet case: variance-covariance matrices have lower error.

estimation from joint ML, or in covariance estimation methods that model the covariances as a function of shared branch lengths (Nei et al., 1985; Gascuel, 1997). Tree building is not only a costly process, but is also subject to inference errors.

The accuracy of our estimator is comparable to the ML variance estimator. Both estimators are biased but in both cases the bias is, for distances below 150 PAM, far smaller than their standard deviation. The bias of the covariance estimator (as well as the ML variance, and to some extent the distance estimators) becomes worse with declining percentage of anchors. These biases arise because the alignment positions under scrutiny do not constitute an unbiased subsample of the true homologous positions. Note that misaligned positions are likely to affect distances from MSAs too. A solution to this problem would lead to better distance estimates in the first place. In the meanwhile, it is probably best to issue a warning if the percentage of anchors falls below some threshold.

The estimation of evolutionary distances is a very important process in molecular evolution, and therefore the covariance estimator presented here will be of use for various applications, such as the construction of GLS trees on OPA distances, the construction of confidence sets of trees based on the GLS test statistic, relative-rate tests, distance-based lateral gene transfer detection, and in general in any process that needs to estimate confidence of distance combinations.

Part II

Applications in Comparative Genomics

5

Detection of Non-Orthology

This chapter is joint-work with Brigitte Boeckmann, Alexander Roth and Gaston Gonnet. It was published in Dessimoz et al. (2006a)

The identification of orthologous genes is a central problem in bioinformatics. Orthologs are genes that evolve from a common ancestor through speciation events, as opposed to paralogs, that result from gene duplication (Fitch, 1970). Discriminating orthologs from paralogs is an important, but non-trivial task. It is important, because function conservation is considerably higher among orthologs (Koonin, 2005), and also because only orthologs reflect the history of their species (Fitch, 1970), meaning that phylogeny inferences must be based on orthologs. It is non-trivial because this distinction requires precise estimates of evolutionary distances from data that are often noisy. Other complications include gene deletion, variations in evolutionary rates, lateral gene transfer, or simply the fact that orthology and paralogy are non-transitive relations, meaning that the relation of every pair of genes must be analyzed separately.

So far, several projects have addressed this problem systematically. Of those, the COGs database (Tatusov et al., 1997, 2003) is by far the best established, not least due to its early inception, its wide scope, and its presence on the NCBI website. The significance of COG in the community is reflected by hundreds of references in scientific articles. Even more importantly, most current initiatives for the identification of orthologs use ideas derived from the methodology of COG, in particular the idea of genome-specific best hit (Fujibuchi et al., 2000;

Remm et al., 2001; Lee et al., 2002). Of all those projects depending either on the methods or results from COG, few question the accuracy of them.

In its last accessible release (2003), the COGs database groups 138,458 proteins from 66 prokaryotes into 4,873 groups that consist of orthologs and in-paralogs. The term in-paralog was coined by Remm and coworkers (Remm et al., 2001) and describes in this context paralogs inside the same species ("trivial paralogs"), as opposed to out-paralogs that result from a duplication event prior to the last speciation event[1]. The inclusion of in-paralogs is usually justified by the fact that such sequences are orthologous to every other sequence within their group. Consequently, the relation of every pair of sequences inside the same COG is unambiguous: pairs of sequences from the same species are paralogs, otherwise, they are expected to be orthologous. The construction of COG groups is based on the fact that orthologous genes almost always have a higher level of sequence conservation than paralogs. Hence, bi-directional best hits ("BBHs") are likely to be formed between orthologs. Yet, if the corresponding ortholog is missing, a BBH might link paralogous sequences. That problem is partly taken care of by COG's approach: BBH are only grouped when they form triangles, and triangles are merged only when they have a common side. However, if more than one species have lost the corresponding ortholog, the construction over triangles will *not* suffice to prevent paralogs from being clustered together. This scenario is far from being unlikely, because losses occurring before speciation events get replicated, and therefore the problem becomes very significant as more species and strains are included for analysis. In fact, simple situations such as the one illustrated on Fig. 5.1 are sufficient to have paralogs clustered together. It is then up to the human curation step at the end of the COG building process (Tatusov et al., 1997) to resolve all such cases.

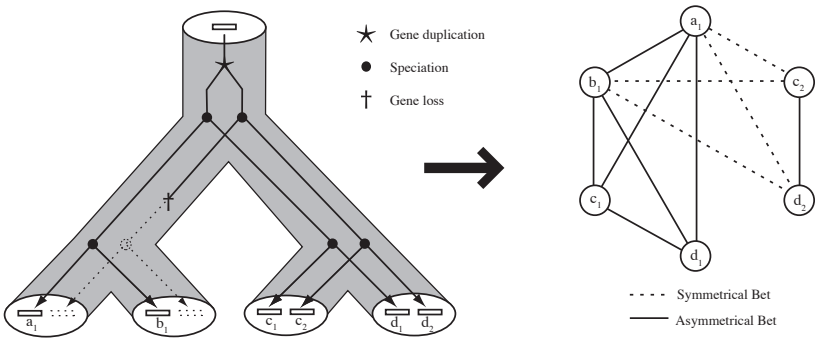


Figure 5.1: A simple evolutionary scenario under which the COG algorithm groups paralogous sequences

[1] Strictly speaking, in/out-paralogy is a relation defined over two sequences and a speciation event of reference. When that event is omitted, it is here the last speciation event that is implied.

The difficulty caused by a single missing ortholog can be easily avoided by requiring that all BBHs be symmetrical, which is what most other projects do. However, if the corresponding ortholog is missing in *both* genomes, even a symmetrical BBH will link paralogs. Therefore, BBHs, even symmetrical, are not necessarily linking orthologs.

This problem could be solved through phylogenetic analysis of the relevant gene families, in particular tree reconciliation (Goodman et al., 1979), but this procedure is not yet practical in large-scale, automated contexts (Koonin, 2005). In the following, we present an algorithm that detects non-orthology without the need of gene tree construction, then report its application on the last version of the COGs database.

5.1 Material and Methods

The algorithm presented here is designed to detect non-trivial paralogous relations within groups of orthologs such as COG groups. Knowing that a paralogous relation within a group is likely to be caused by the loss of the corresponding ortholog in both species, the algorithm looks for a third-party species, which we call the "witness of non-orthology", in which both corresponding orthologs are present (Fig. 5.2). Under the assumptions of good and complete data, and similar evolutionary rates among orthologs, such a situation is characterized by the following three requirements on the evolutionary distances: i) In Z, z_3 is the closest protein to x_1 and z_4 is the closest protein to y_2. ii) The pair (x_1, z_3) must be significantly closer than (x_1, z_4), and conversely, (y_2, z_4) must be significantly closer than (y_2, z_3). That excludes cases where z_3 and z_4 are in-paralogs (Fig. 5.3, left), because for in-paralogs to fulfill those conditions, convergent evolution *at the sequence level* would be required, a phenomenon that is so unlikely that we ignore it (Doolittle, 1994). iii) the distance between (x_1, z_4) must be similar to (y_2, z_3). That excludes cases where X (resp. Y) speciated before the duplication event, in which case x_1 (resp. y_2) is orthologous to all three other genes (Fig. 5.3, right).

We finish this overview of the algorithm by considering the impact of lateral gene transfer (LGT) and gene fusion/fission. Clearly, the algorithm presented here was not designed to detect LGT events between x_1 and y_2, an interesting problem in itself that remains largely unsolved. More importantly here, an LGT in a third-party species Z can lead to a situation where Z wrongly appears to be witness of non-orthology: consider 3 orthologous proteins x_1, y_2 and z_3 in three species X, Y and Z. At some point, Z acquires through LGT a member of that orthologous family, which we now refer to as z_4. Z keeps both copies z_3 and z_4. Furthermore, Z happens to be closer to X than Y, while the donor of z_4 is closer to Y than X. This situation leads to a misclassification by our algorithm. Although such cases cannot be ruled out, we did not encounter any among the

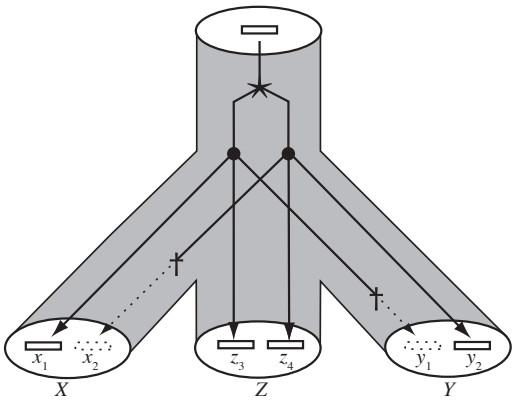


Figure 5.2: Suitable case of a witness. A duplication occurred before all speciations and Z is a witness of the non-orthology between the sequences x_1 and y_2.

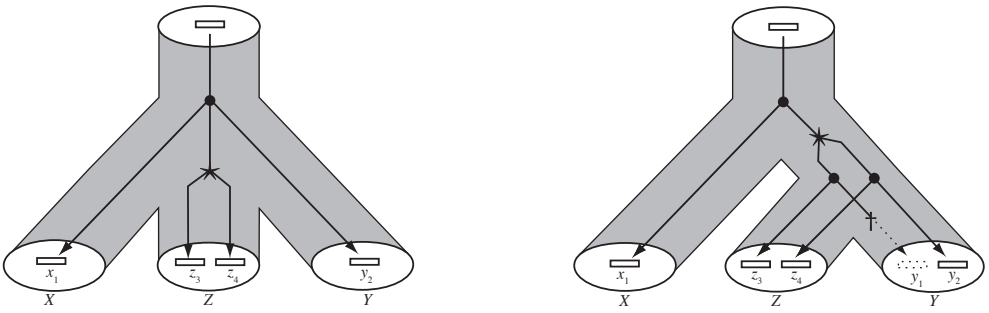


Figure 5.3: Unsuitable cases of witnesses. To the left, duplication occurred only in Z, and therefore z_3 and z_4 are in-paralogs with respect to (X, Y), and cannot act as witness of non-orthology. To the right, X speciated before the duplication event. Hence, x_1 is orthologous to all 3 other proteins and cannot act as witness of non-orthology.

numerous case-by-case analysis performed on the results. It could be that orthologous gene displacement of z_3 by z_4 through homologous recombination is a much more likely scenario, and besides, the frequency of LGT appears to be higher among closely related species (Lawrence and Hendrickson, 2003). As for gene fusion or gene fission, the units for amino acid sequence analysis are no longer proteins but domains. Even though the analysis of homologous domains from distinct proteins is scientifically meaningful, our analysis remains at the level of entire proteins to simplify matters.

Note that the complications caused by LGT events and, probably to a lesser extent, by gene fusion/fission are not specific to our method and pose challenges to other approaches as well, in particular tree reconciliation.

5.1.1 Input data

The algorithm uses two inputs: the COGs database and pairwise sequence alignments between all proteins involved in the analysis. As introduced above, the orthology of two sequences is verified through an exhaustive search of the corresponding sequences in complete, third-party genome. Therefore, a large number of genomes is desirable. However, since the relation between every pair of sequence is needed, such searches require the computation of a very large number of pairwise alignments. For practical reasons, all results presented here use results from the Smith and Waterman (1981a) all-against-all protein alignments precomputed in the scope of the OMA project (Chapter 6).

For each alignment, a PAM distance estimate and the corresponding variance is computed using ML and numeric integration (Gonnet, 1994; Muller and Vingron, 2000).

5.1.2 Comparison of Evolutionary Distances

The algorithm uses evolutionary distances to detect paralogs. However, the distances estimates are subject to perturbation, which must be taken into account when comparing them. Therefore, assuming that errors are normally distributed, the difference $\Delta(d_1, d_2)$ of two distances d_1, d_2 has expected value:

$$\mathbb{E}(\Delta(d_1, d_2)) = \mathbb{E}(d_1) - \mathbb{E}(d_2)$$

with variance

$$\sigma^2(\Delta(d_1, d_2)) = \sigma^2(d_1) + \sigma^2(d_2) - 2Cov(d_1, d_2)$$

If the two distances are independent, the covariance term disappears and the variance of the difference can be obtained directly from the individual variances. But more often than not, d_1 and d_2 involve a common protein and are therefore

not independent, meaning that not taking the covariance into account overestimates the error. In Chapter 4, we have presented methods to approximate the covariance of two evolutionary distances.

5.1.3 Algorithm

The algorithm goes through each COG group, and verifies inside each of them that every two genes x_1, y_2 coming from different species have a significant alignment, and are indeed orthologs. Alignments are considered significant if the score is above 130 (47 bits, which typically corresponds to an E-value around 2e-6) and the length of the alignment not less than 50% of the smallest sequence. The verification of orthology is performed through the search, in each third-party genome Z, of two genes z_3 and z_4 that fulfill the three conditions (i-iii) presented at the beginning of this section:

$$\begin{aligned} &\forall z_i \neq z_3 : \ \Delta(x_1 z_3, x_1 z_i) < k \cdot \sigma(\Delta(x_1 z_3, x_1 z_i)) \\ &\forall z_j \neq z_4 : \ \Delta(y_2 z_4, y_2 z_j) < k \cdot \sigma(\Delta(y_2 z_4, y_2 z_j)) \end{aligned} \tag{5.1}$$

$$\begin{aligned} \Delta(x_1 z_4, x_1 z_3) &> k \cdot \sigma(\Delta(x_1 z_4, x_1 z_3)) \\ \Delta(y_2 z_3, y_2 z_4) &> k \cdot \sigma(\Delta(y_2 z_3, y_2 z_4)) \end{aligned} \tag{5.2}$$

$$| \Delta(x_1 z_4, y_2 z_3) | < k \cdot \sqrt{(\sigma^2(x_1 z_4) + \sigma^2(y_2 z_3))} \tag{5.3}$$

where k is the confidence level, which we set to 1.96. If the quartet (x_1, y_2, z_3, z_4) fulfills all three conditions, there is enough evidence to consider x_1, y_2 paralogs. The algorithm was implemented in the programming environment *Darwin* (Gonnet et al., 2000).

A note about parameter choice. As mentioned previously, the classification of protein pairs in orthologs and non-orthologs can be very difficult or even impossible, especially when a speciation event immediately follows a duplication event, or in the situation of frequent gene gain and gene loss, as it is observed in certain groups of proteins such as metabolic enzymes. Here, the choice of $k = 1.96$ standard deviations was established empirically such that the false-positive rate (orthologs misclassified as non-orthologs) is much smaller than the false-negatives rate (missed non-orthologs). In other words, we expect that our algorithm reports only clear-cut cases of paralogy.

5.1.4 Phylogenetic Analysis

To verify individual cases reported by the algorithm, phylogenetic trees were constructed using independent, common software packages, as follows: sequences were aligned using Muscle (Edgar, 2004) and ClustalW (Chenna et al.,

2003). Whenever they differed, the one that seemed more likely to our expert curator was selected. Short sequences, suspicious regions and most gap-containing columns removed. Distance matrices (JTT, gamma) generated with protdist (Felsenstein, 1993) were used to construct phylogenetic trees using neighbor (Felsenstein, 1993). Clusters of interest were selected for detailed analysis. Alignments of the selected data were performed using Tcoffee (Poirot et al., 2003) and the result subsequently modified as described above, and considering the Tcoffee CORE (Consistency of Overall Residue Evaluation) values for the alignment. Information on the stability of the tree topology was assessed building an extended majority rule consensus tree using consense (Felsenstein, 1993) from BIONJ (Gascuel, 1997) searches performed on 1,000 bootstrap replicates, which were constructed with seqboot (Felsenstein, 1993). Protein trees of the data subset were constructed using the Bayesian tree-building method MrBayes (Ronquist and Huelsenbeck, 2003) (JTT; invgamma-4; 1,000,000 generations). The trees were rooted using an outgroup whenever a suitable ancient paralog could be found. Note that since the analysis attempts at clustering homologs into clans, and not at predicting their hierarchical order, placement of the root is not critical here.

5.1.5 Validation

The performances of the algorithm were evaluated using the HAMAP database (Gattiker et al., 2003), a collection of orthologous microbial protein families generated manually by expert curators in the Swiss-Prot group. The database was retrieved on Nov 23th, 2005. Proteins from the 99 most represented species also present in our OMA project were used in the analysis: of all 29,245 proteins, there were 21,831 proteins (75,6%), grouped in 1,189 orthologous families. That yielded 309,829 pairwise relations to be verified by our procedure.

The algorithm classified 279,568 (90.2%) relations as orthologous and 9,420 (3.0%) as paralogous. The remaining 20,841 (6.7%) relations had alignments below our significance threshold and could therefore not be processed. The accuracy of the algorithm, in particular its very low false-positive rate was confirmed by following observations:

First, paralogy is often reflected by different Swiss-Prot ID names (e.g. GREA/GREB) (Boeckmann et al., 2003). From the 9,420 predicted paralogs, only 2,728 (29.0%) of them have identical ID names. Second, the distribution of the paralogs among HAMAP families was investigated: all 9,420 cases of paralogy found by the algorithm are concentrated in only 150 (12.6%) of the 1,189 HAMAP families. This is consistent with the fact that the inclusion of just one paralogous protein into an orthologous family is likely to result in several paralogous relations inside that family. And indeed, in all except 8 of these 150 families, more than one paralogous pair was detected. Third, these 8 improbable cases were inspected individually using phylogenetic analysis, which

confirmed that they are *bona fide* paralogs (possibly xenologs). Fourth, the predicted cases of paralogy were compared to the gene trees over HAMAP families built by the group of Laurent Duret [2], in a similar way as HOBACGEN (Perriere et al., 2000). 7,217 predicted cases could be mapped to those trees. In 6,418 (88.9%) instances, paralogy was confirmed by the trees, a remarkably high level of consistency considering that the two methods are very different. As for the conflicting 799 cases, which are distributed among 51 families, we believe that most of them are caused by inaccuracies on the gene trees, which are constructed using a variant of Neighbor Joining on observed divergence, a rather crude measure of evolutionary distance.

5.2 Results and Discussion

The algorithm was run on the current release of the COGs database (Tatusov et al., 2003). We used the precomputed all-against-all results from 107 complete genomes, of which 52 are represented in COGs, whereas the remaining 55 genomes were only used as potential witnesses of non-orthology[3]. From all 4,654 COGs, there is a total of 5,537,713 pairwise relations. 484,043 of those involve pairs of proteins within the same species and were therefore not considered further. Additionally, 2,733,371 relations involve at least one protein from a species outside our set of 107 genomes. Consequently, the following results were obtained through the verification of 2,320,199 relations, 45.9% of all potential orthologous relations.

The results are presented in Table 5.1. Surprisingly, 44% of the relations had alignment scores below our significance threshold of 130, which corresponds to an E-value of about 2e-6, and could therefore not be verified. This implies that an important fraction of relations within COGs cannot be, on the basis of pairwise alignments, reliably considered *homologous*.

Table 5.1: Results of the algorithm of the COGs database.

	#	%
Pairs with score below threshold, not tested	1,021,764	44.0
Pairs with score above threshold	1,298,435	66.0
Non-orthologous pairs	360,856	27.8
Orthologous pairs	937,579	72.2
COG groups with non-orthology	1,604	34.5
COG groups without non-orthology	3,050	65.5

[2] http://pbil.univ-lyon1.fr/help/HAMAP.html

[3] the complete list is available in the supplementary materials of Dessimoz et al. (2006a)

The other result is the significant proportion of non-orthologous relations found by the algorithm, more than a quarter of the pairs that could be verified. They are distributed among about a third of all COGs. The list of such groups, along with all detected non-orthology cases are available in the supplementary materials.

If we require the presence of at least *two* witnesses of non-orthology for a pair to be considered non-orthologous, the algorithm still finds 251,391 (19.4%) such pairs within 1146 (24.6%) COGs. When removing the sequence with the most non-orthologous relations from each COG group, the total number of non-orthologous pairs decreases by only 24,868 (1.9%).

The majority (70%) of the groups predominantly non-orthologs are involved in metabolic processes, according to the functional description of the COGs database, although they only constitute a minority of all COGs. In contrast, groups involved in information storage and processing (8%) or cellular processing and signaling (11%) include less frequently non-orthologs. The remainder 11% are poorly characterized proteins. This result is in agreement with previous studies, which state that in prokaryotes, metabolic functions are under high evolutionary pressure from changing environments (Pal et al., 2005).

5.2.1 Phylogenetic analysis of selected COG groups

The presence of non-orthology in some COG groups is hardly a surprise and was in fact recently acknowledged by Koonin, coauthor of COG, in a review article (Koonin, 2005). What is surprising here is rather the *extent* of non-orthology detected by the algorithm. That prompted us to verify, in addition to the validation work reported in Sect. 5.1.5, a number of our predictions using detailed phylogenetic analysis. In this section, we report the conclusion of such analysis on three COGs, for which we could build Bayesian likelihood trees of high confidence, confirmed by consensus neighbor-joining trees with high bootstrap values. Clan assignments were made based on those trees, and considering lineage and function, whenever reliable annotations could be found. We strongly expect that pairs of proteins across clans be non-orthologous, and use these results to evaluate the accuracy of the predictions made by the algorithm.

COG0508 consists of complex-forming acyltransferases that are composed of an N-terminal biotin or lipoic acid attachment domain, a central protein-protein interaction domain, followed by the catalytic 2-oxoacid dehydrogenases acyltransferase domain. The phylogenetic analysis of roughly half of the proteobacterial sequence data from COG0508 suggests the existence of at least 4 distinct subgroups (see Fig. 5.4): clan 1 is formed by sequences from gammaproteobacteria, including the dihydrolipoyllysine-residue acetyltransferase component of the pyruvate dehydrogenase complex (EC 2.3.1.12) (AceF) from *Escherichia coli*. Clan 2 consists of proteins highly similar to the *Bacillus subtilis* lipoamide acyltransferase component of the branched-chain alpha-keto acid de-

hydrogenase complex (EC 2.3.1.168). All sequences in clan 2 are alphaproteobacterial, except for *Pseudomonas aeruginosa* proteins, which are found in both clan 1 and clan 2. As mentioned in Sect. 5.1, such situation could arise through lateral gene transfer from an alphaproteobacteria to *P. aeruginosa*. If that was the case, there would be strong evidence that clans 1 and 2 should be merged. However, in the present case, it is possible to populate both clans with additional sequences from more distant species (not shown here), legitimating the separation in 2 clans. Additionally, the long distance between the two clans and the distinct function of at least one family member of each subgroup also supports this conclusion. Clan 3 includes the dihydrolipoyllysine-residue succinyltransferase component of 2-oxoglutarate dehydrogenase complex (EC 2.3.1.61) (SucB) of *E. coli*. Note that clan 3 includes two protein sequences of *Rhizobium meliloti*, but those are clearly ancient duplicates, and thus sequence 3b is likely to form yet a separate clan on its own. Finally clan 4 is formed by a presumably further dehydrogenase component from alphaproteobacteria. The algorithm predicted 382 cases of non-orthologous relations within the sequences considered here. An extract of the result list is given in Table 5.2 (the full list of paralogy is available in the supplementary material). 379 predictions are consistent with the clan assignment, while the remaining 3 predictions support the exclusion of *R. meliloti* 3b from clan 3. Furthermore, comparison with the clan assignment reveals that the algorithm missed 24 non-orthologous relations, which implies a false-negative rate of 6.0%.

COG0513 includes various DEAD-box containing RNA helicases. The phylogenetic analysis of the proteobacterial data from this group suggests the existence of 6 clans (see Fig. 5.5), of which 5 are formed around the following proteins from *Escherichia coli*: 1) the ATP-dependent RNA helicase SrmB, which is involved in an early assembly step of 50S ribosomal subunits (Charollais et al., 2003); 2) the cold-shock DEAD-box protein A (DeaD), required for cell division and normal cell growth at low temperature (Jones et al., 1996); 3) the DEAD-box RNA helicase B (RhlB), a component of the RNA degradosome, which seems to have little activity unless being activated by the endoribonuclease RNase E (Carpousis, 2002); 4) the putative RNA helicase RhlE, which has been shown to be nonessential for normal cell growth (Ohmori, 1994); 5) the ATP-independent RNA 3'→5' helicase DbpA (Diges and Uhlenbeck, 2005). The sixth subgroup includes RNA helicases that are conserved in some alphaproteobacteria. The algorithm predicted 408 cases of non-orthology, 88.9% of the 459 non-orthologous relations that can be deduced from the clan assignment. In this case, there was no false-positive prediction.

COG1113 consists of members of the amino acid-polyamine-organo-cation (APC) superfamily from bacteria, specifically those integral membrane proteins that are involved in the transport of amino acids in prokaryotes. The phylogenetic analysis of this group suggests the existence of various clans (see Fig. 5.6), including those formed around the 7 proteins found in *Escherichia coli*: 1)

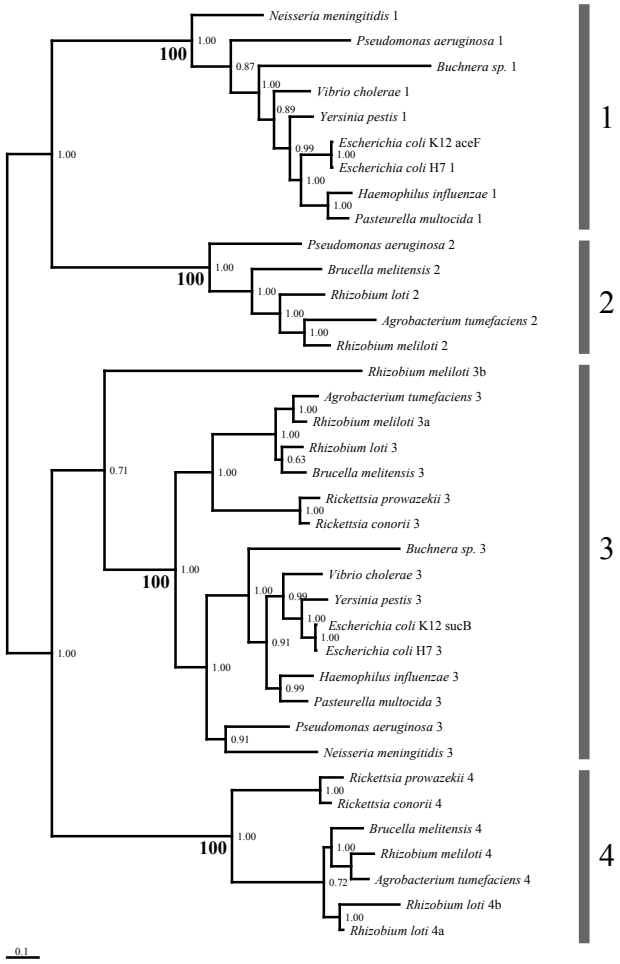


Figure 5.4: Unrooted phylogenetic consensus tree constructed from a Bayesian analysis of a subgroup from COG0508. Posterior probabilities are indicated to the right of the nodes and clan-supporting bootstrap values are indicated below the probability value. Predicted clans are indicated by the vertical bars on the right side. The leaf labels correspond to the following COG identifiers: *A. tumefaciens* (2: AGl2719, 3: AGc4775, 4: AGc2641), *B. melitensis* (2: BMEII0746, 3: BMEI0141, 4: BMEI0856), *Buchnera sp.* (1: BU206, 3: BU303), *E. coli* K12 (COG identifier corresponds to the gene name: aceF, sucB), *E. coli* H7 (1: ECs0119, 3: ECs0752), *H. influenzae* (1: HI1232, 3: HI1661), *N. meningitidis* (1: NMB1342, 3: NMB0956), *P. multocida* (1: PM0894, 3: PM0278), *P. aeruginosa* (1: PA5016, 2: PA2249, 3: PA1586), *R. loti* (2: mll4471, 3: mll4300, 4a: mlr0385, 4b: mll3627), *R. meliloti* (2: SMc03203, 3a: SMc02483, 3b: SMb20019, 4: SMc01032), *R. conorii* (3: RC0226, 4: RC0764), *R. prowazekii* (3: RP179, 4: RP530), *V. cholerae* (1: VC2413, 3: VC2086), *Y. pestis* (1: YPO3418, 3: YPO1114).

Table 5.2: Predicted non-orthologous relations for the data shown in Fig. 5.4. The sequences in the first two columns are predicted to be non-orhtologous by the pair of witnesses in the third column.

Predicted non-orthologs		Pair of witnesses
A. tumefaciens 2	*Buchnera sp.* 1	*P. aeruginosa* 2 + 1
A. tumefaciens 2	*E. coli* H7 1	*P. aeruginosa* 2 + 1
A. tumefaciens 2	*E. coli* K12 acef	*P. aeruginosa* 2 + 1
A. tumefaciens 2	*H. influencae* 1	*P. aeruginosa* 2 + 1
A. tumefaciens 2	*N. menininginitidis* 1	*P. aeruginosa* 2 + 1
A. tumefaciens 2	*P. multocida* 1	*P. aeruginosa* 2 + 1
A. tumefaciens 2	*R. loti* 4a	*B. melitensis* 2 + 4
A. tumefaciens 2	*R. meliloti* 4	*B. melitensis* 2 + 4
A. tumefaciens 2	*V. cholerae* 1	*P. aeruginosa* 2 + 1
A. tumefaciens 2	*Y. pestis* 1	*P. aeruginosa* 2 + 1
A. tumefaciens 3	*B. melitensis* 2	*Buchnera sp.* 3 + 1
A. tumefaciens 3	*Buchnera sp.* 1	*P. aeruginosa* 3 + 1
A. tumefaciens 3	*P. aeruginosa* 2	*B. melitensis* 3 + 2
A. tumefaciens 3	*P. aeruginosa* 1	*Buchnera sp.* 3 + 1
A. tumefaciens 3	*P. multocida* 1	*Buchnera sp.* 3 + 1
A. tumefaciens 3	*R. conorii* 4	*B. melitensis* 3 + 4
A. tumefaciens 3	*R. loti* 2	*B. melitensis* 3 + 2
A. tumefaciens 3	*R. loti* 4a	*B. melitensis* 3 + 4
A. tumefaciens 3	*R. meliloti* 2	*B. melitensis* 3 + 2
A. tumefaciens 3	*R. meliloti* 4	*B. melitensis* 3 + 4
A. tumefaciens 3	*R. prowazekii* 4	*B. melitensis* 3 + 4
A. tumefaciens 4	*B. melitensis* 2	*R. loti* 4a + 2
A. tumefaciens 4	*Buchnera sp.* 1	*B. melitensis* 4 + 2
A. tumefaciens 4	*E. coli* H7 3	*B. melitensis* 4 + 3
A. tumefaciens 4	*E. coli* K12 sucB	*B. melitensis* 4 + 3
A. tumefaciens 4	*H. influencae* 1	*R. loti* 4a + 2
A. tumefaciens 4	*H. influencae* 3	*B. melitensis* 4 + 3
A. tumefaciens 4	*N. menininginitidis* 1	*B. melitensis* 4 + 2
A. tumefaciens 4	*N. menininginitidis* 3	*B. melitensis* 4 + 3
A. tumefaciens 4	*P. aeruginosa* 2	*B. melitensis* 4 + 2
A. tumefaciens 4	*P. aeruginosa* 3	*B. melitensis* 4 + 3

Table 5.3: Continuation of table for data in Fig. 5.4.

A. tumefaciens 4	*P. multocida* 1	*B. melitensis* 4 + 2
A. tumefaciens 4	*P. multocida* 3	*B. melitensis* 4 + 3
A. tumefaciens 4	*R. conorii* 3	*B. melitensis* 4 + 3
A. tumefaciens 4	*R. loti* 2	*B. melitensis* 4 + 2
A. tumefaciens 4	*R. loti* 3	*B. melitensis* 4 + 3
A. tumefaciens 4	*R. meliloti* 2	*B. melitensis* 4 + 2
A. tumefaciens 4	*R. meliloti* 3a	*B. melitensis* 4 + 3
A. tumefaciens 4	*R. prowazekii* 3	*B. melitensis* 4 + 3
A. tumefaciens 4	*V. cholerae* 3	*B. melitensis* 4 + 3
A. tumefaciens 4	*Y. pestis* 3	*B. melitensis* 4 + 3
B. melitensis 2	*E. coli* H7 1	*P. aeruginosa* 2 + 1
B. melitensis 2	*E. coli* H7 3	*A. tumefaciens* 2 + 3
B. melitensis 2	*E. coli* K12 acef	*P. aeruginosa* 2 + 1
B. melitensis 2	*E. coli* K12 sucB	*A. tumefaciens* 2 + 3
B. melitensis 2	*H. influencae* 1	*P. aeruginosa* 2 + 1
B. melitensis 2	*H. influencae* 3	*A. tumefaciens* 2 + 3
B. melitensis 2	*N. menininingitidis* 1	*P. aeruginosa* 2 + 1
B. melitensis 2	*N. menininingitidis* 3	*A. tumefaciens* 2 + 3
B. melitensis 2	*P. aeruginosa* 3	*A. tumefaciens* 2 + 3
B. melitensis 2	*P. multocida* 1	*P. aeruginosa* 2 + 1
B. melitensis 2	*P. multocida* 3	*A. tumefaciens* 2 + 3
B. melitensis 2	*R. conorii* 3	*A. tumefaciens* 2 + 3
B. melitensis 2	*R. conorii* 4	*A. tumefaciens* 2 + 4
B. melitensis 2	*R. loti* 3	*A. tumefaciens* 2 + 3

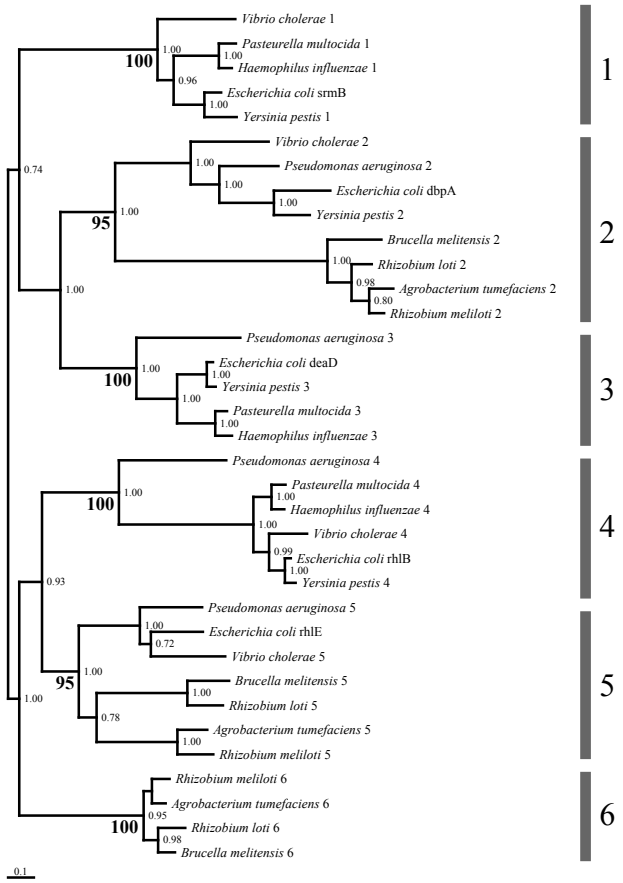


Figure 5.5: Unrooted phylogenetic consensus tree for COG0513, constructed from a Bayesian analysis. Posterior probabilities are drawn to the right of the nodes and clan-supporting bootstrap values are below the relevant nodes. The vertical bars bars to the right indicate the prediced clans. The leaf labels correspond to the COG identifiers: *A. tumefaciens* (2: AGl1362, 5: AGc4238, 6: AGc3366), *B. melitensis* (2: BMEI1824, 5: BMEI0934, 6: BMEI1035), *E. coli* K12 (COG identifier corresponds to the gene name: dbpA, deaD, rhlB, rhlE, srmB), *H. influenzae* (1: HI0422, 3: HI0231, 4: HI0892), *P. multocida* (1: PM1840, 3: PM1112, 4: PM1921), *P. aeruginosa* (2: PA0455, 3: PA2840, 4: PA3861, 5: PA0428), *R. loti* (2: mlr4393, 5: mlr0349, 6: mll0224), *R. meliloti* (2: SMc01090, 5: SMb20880, 6: SMc00522), *V. cholerae* (1: VC0660, 2: VC2564, 4: VC0305, 5: VCA0204), *Y. pestis* (1: YPO2708, 2: YPO1776, 3: YPO3488, 4: YPO3869).

phenylalanine-specific permease (PheP), 2) aromatic amino acid transport protein (AroP), 3) probable transport protein YifK, 4) proline-specific permease (ProY), 5) D-serine/D-alanine/glycine transporter (CycA), 6) L-asparagine permease (AnsP), 7) GABA (4-aminobutyrate) permease (GabP). The 7 clans were predicted with high probability and their clusterings confirmed by significant bootstrap values (99-100%) except for one (92%). The analyzed data set includes members of quite related organisms, but most clans can already be populated with further members from other species of COG1113. The algorithm predicted 257 pairs of non-orthologs, of which 254 are consistent with the phylogenetic analysis. That represents 97.7% of the 260 non-orthologous relations that can be deduced from the clan assignment. The conflicting 3 predictions suggest that *P. aeruginosa* 4a is non-orthologous to *E. coli K12* ProY and to *E. coli H7 EDL933* 4, and that *P. aeruginosa* 4b is non-orthologous to *Y. pestis* 4b. But here too, the extension of the phylogenetic analysis using additional sequences from the UniProtKB database supports the division of clan 4 into further subgroups (not shown here).

5.3 Conclusions

We present here a new algorithm for the detection of non-orthologous relations caused by the limitations of genome-specific best hit methods such as the COGs database. The algorithm, rather than building gene trees, a process both computationally expensive and error-prone, works with pairwise distance estimates. The accuracy of the algorithm was evaluated through verification of the distribution of predicted cases, case-by-case phylogenetic analysis and comparisons with prediction from other projects using independent methods. Using conservative parameters, the algorithm detected non-orthology in a third of the COG groups. Methods sensitive to correct orthology assignments, such as function prediction, phylogenetic trees or genome rearrangement analysis, will profit from both the algorithm and the results presented here.

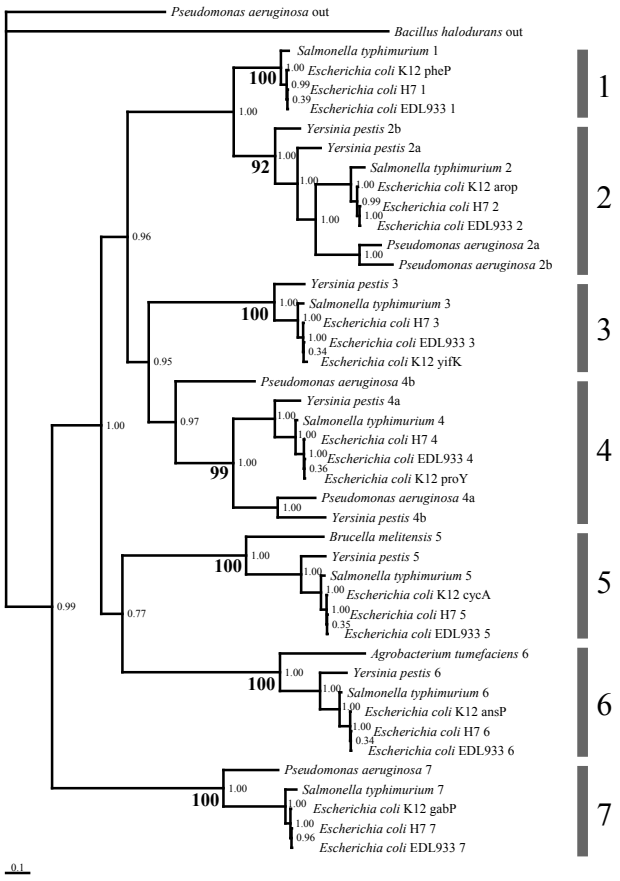


Figure 5.6: Phylogenetic consensus tree rooted by outgroups for COG1113, constructed from a Bayesian analysis of a data subgroup from COG1113. Posterior probabilities of the Bayesian analysis are drawn to the right of the nodes and clan-supporting bootstrap values below relevant nodes. Predicted clans are indicated by vertical bars to the right. The leaf labels correspond to the COG identifiers: *A. tumefaciens* C58 (6: AGl2082), *B. halodurans* (out: BH2171), *B. melitensis* (5: BMEII0038), *E. coli* K12 (COG identifier corresponds to the gene name: ansP, aroP, cycA, gabP, pheP, proY, yifK), *E. coli* H7 EDL933 (1: ZpheP, 2: ZaroP, 3: ZyifK, 4: ZproY, 5: ZcycA, 6: ZansP, 7: ZgabP), *E. coli* H7 (1: ECs0614, 2: ECs0116, 3: ECs4729, 4: ECs0452, 5: ECs5186, 6: ECs2057, 7: ECs3524), *P. aeruginosa* (2a: PA3000, 2b: PA0866, 4a: PA5097, 4b: PA0789, 7: PA0129, out: PA2079), *S. typhimurium* LT2 (1: STM0568, 2: STM0150, 3: STM3930, 4: STM0400, 5: STM4398, 6: STM1584, 7: STM2793), *Y. pestis* (2a: YPO3421, 2b: YPO1743, 3: YPO3854, 4a: YPO3201, 4b: YPO4015, 5: YPO1859, 6: YPO1937).

6

OMA: Large Scale Orthology Inference

This chapter describes OMA, a large-scale project for the identification of orthologs in complete genomes initiated in our research group in 2004. We first provide a brief overview, then describe the algorithm. The third section is an extensive comparison of OMA with other methods, and evaluates the predictions in the context of both phylogenetic and functional studies. Finally, the last section describes the web browser, which allows interactive exploration of the data.

Overview The identification of orthologs, pairs of homologous genes in different species that started diverging through speciation events, is a central problem in genomics with applications in various research areas, including comparative genomics, phylogenetics, protein function annotation, genome rearrangement, or transcription binding site prediction. Evidence for the importance of these tasks can be found in the growing number of orthology assignment projects developed in recent years.

The COG method (Tatusov et al., 1997) was the first one to extend the systematic orthology search beyond the relatively simple reciprocally best matches approach. This algorithm, originally applied mostly on bacterial genomes, was later extended and applied to eukaryotic genomes (the KOG database of Tatusov et al. 2003 and the EGO/TOGA project of Lee et al. 2002). Following these approaches, more sophisticated orthology determination algorithms were proposed in recent years, most notably Inparanoid/MultiParanoid (Remm et al., 2001; Alexeyenko et al., 2006), OrthoMCL (Li et al., 2003), KEGG Orthol-

ogy (Kanehisa et al., 2004) and Roundup (DeLuca et al., 2006). Other projects, such as IMG (Markowitz et al., 2006) and MicrobesOnline (Alm et al., 2005) employ more basic algorithms, but distinguish themselves by a large number of analyzed sequences.

The OMA project (Dessimoz et al., 2005) is a massive cross-comparison of complete genomes to identify the evolutionary relation between any pair of proteins. The main features of OMA are the large number of genomes from all kingdoms of life, the strict verification of orthology assignments and the determination of the phylogenetic relation between any two proteins. The distincting features of OMA include the use of distances instead of scores and statistically sounder measures for establishing the set of potential orthologs in the early stages of the algorithm. Dealing with such large amounts of data requires a high degree of automation and integrated quality checks. Unlike comparable projects such as the COGs database or KEGG Orthology, OMA does not rely on human intervention. Once a genome database is integrated into the local databases, the process is fully automated.

Strictness Orthology inference can be a difficult problem, for instance when gene families went through intense expansion and reduction or when duplication and speciation events occurred at close time intervals. In OMA, we have a strict approach across the entire procedure, such as full dynamic programming alignments instead of Blast or the systematic use of evolutionary distances with confidence intervals. When lacking discriminating information, we favor false-negative (missing orthologous relations) over false-positive (erroneous orthology assignment). A notable and in this extent unique feature is the systematic verification of every putative pair of orthologs by an exhaustive search for "witnesses of non-orthology" in third-party genomes (Chapter 5).

Orthology inference at the level of pairs Orthology is not necessarily a one-to-one relation and also not always transitive, therefore any clustering approach will have its limits. Although OMA initially focused on groups of orthologs, it also reports information about pairwise orthologous relations, which can be categorized as follows:

- One-to-one orthologs: in both species, there is only one corresponding ortholog.
- One-to-many or many-to-many orthologs: in at least one of the two species, the gene duplicated after speciation.
- Paralogs: the two proteins arose through gene duplication, not speciation.

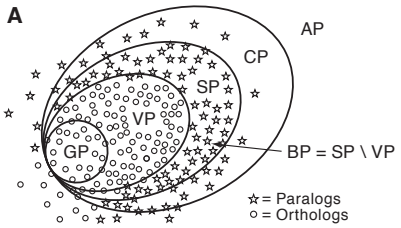
As far as we know, the only other project providing orthology inference at the level of pairs is Ensembl (Hubbard et al., 2005).

Nevertheless, for many analyses (particularly for phylogenetics), it is convenient to have groups of orthologs with at most one protein from each species and every pair inside the group being a pair of orthologs. Therefore the OMA Browser also provides a group-centric view. OMA groups are cliques of orthologs chosen in a way such that the alignment scores are maximized. The clique requirement ensures that the above stated properties of an orthologous group are fulfilled.

6.1 Algorithm

This section is joint-work with Alexander Roth and Gaston Gonnet. It was published in Roth et al. (2008).

The goal of the OMA project is to detect all orthologous sequence relations among sequenced genomes. Considering that orthology is a pairwise relation, the starting point is all $\binom{n}{2}$ pairs of sequences which are filtered successively in several steps to yield pairs of orthologous (Fig 6.1A). Figure 6.1B lists the names of these shrinking subsets and their meaning in terms of their evolutionary relation.



B

Pairs	Evolutionary Relation
All Pairs (AP)	Any
Candidate Pairs (CP)	Homologs
Stable Pairs (SP)	Orthologs, Pseudo-Orthologs
Broken Pairs (BP)	Paralogs
Verified Pairs (VP)	Orthologs
Group Pairs (GP)	Close Orthologs

Figure 6.1: **A** Diagram showing the hierarchy of pairs classifying evolutionary relations. **B** The connection between pairs of sequences and their corresponding evolutionary relation. The verified pairs (VP) cover all orthologs and the group pairs (GP) are a subset of these. The broken pairs (BP) are cases were paralogy have been explicitly classified.

The algorithm goes through four steps (see Figure 6.2), in each of which the pairs are filtered:

1. To find homology we compute pairwise alignments between *all pairs* of sequences for all genes in all genomes. Pairs that have significant alignments are kept as *candidate pairs*.

2. Orthologs are often the closest genes, because they start diverging only after speciation whereas paralogs already diverge before speciation. Genes in different genomes that mutually have each other as the most closely related sequences are upgraded to *stable pairs*.

3. In cases where the ortholog is missing, we try to avoid erroneous classification of paralogs as orthologs (pseudo-orthologs) by verifying with a third genome. Pairs that pass the verification step are upgraded to *verified pairs*. Pairs that do not pass are referred to as *broken pairs*.

4. For some applications (e.g. functional annotation and species tree reconstruction) it is advantageous to cluster orthologs into *groups of orthologs*. Pairs of sequences in such groups are called *group pairs*.

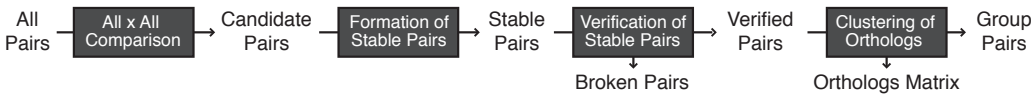


Figure 6.2: Flow chart of the algorithm. Boxes represent the steps of the algorithm. The arrows are the input and output data for the steps.

These four steps will now be examined in detail.

6.1.1 All Against All Alignments

The first step of the process aims at detecting homology. To do this, all pairwise protein sequences from annotated complete genomes are aligned using full dynamic programming.

It should be stressed that the comparison of protein sequences to assess orthology is not an obvious choice, since the evolutionary relations discussed here also apply to genes. There are several advantages in using protein sequences. Very distant homologies are difficult to find at the DNA level and protein sequences suffer less from convergence due to mutational biases. Also, their length is one third of that of the DNA sequence, a considerable advantage given that the time complexity of aligning sequences is quadratic in their lengths. The downside to using protein sequences instead of DNA is that complications arising from multiple gene products need to be handled explicitly.

The sequences used are available from public databases (mainly GenBank (Benson et al., 2005) for Prokaryotes and Ensembl (Hubbard et al., 2002) for Eukaryotes). All data are checked for consistency and quality, as errors from poor gene sequencing or annotation are not only difficult to detect and correct, but can also corrupt other good data. The problem of multiple splice variants is handled by selecting the longest splice variant, as well as isoforms with at least 10% non-redundant positions.

Homology is established in two sub-steps: First, alignments using full local dynamic programming are performed between all sequences using a fixed PAM matrix to find all homologous sequences (Smith and Waterman, 1981a; Dayhoff et al., 1978). Empirically, we found that the PAM-224 matrix is a good choice.

Secondly, all significant alignments (score > 85) are refined using the Dayhoff matrix that maximizes the score. Since scores are log of odd ratios, the PAM number of this matrix corresponds to the maximum likelihood estimator of the evolutionary distance. Refined alignments with score above 200 (which roughly corresponds to an E-values of 10^{-16}) are considered significant. Below this value, the amount of candidate pairs are increasing rapidly, but they only contribute little to the number of verified pairs (not shown).

The all-against-all step is computationally expensive and the running time increases quadratically with the total number of amino acids. Using a heuristic based algorithm such Blast (Altschul et al., 1990) would increase the speed, at the cost of reduced sensitivity (Chen, 2003).

We consider genes, not domains, to be the basic evolutionary unit. Why then not use global alignments? There is good reason to ignore the ends of protein sequences, which are often variable. In order to guarantee that a significant fraction of the sequences are aligned and avoid matching of individual domains, we use a *length tolerance* criterion. The length of the shorter matching part of the sequences needs to be at least the fraction ℓ times of the longest sequence. That is

$$\min(|a_1|, |a_2|) > \ell \cdot \max(|s_1|, |s_2|)$$

where a_1 and a_2 are the lengths of the aligned subsequences of s_1 and s_2. Alignments that pass the length and the score criteria are upgraded to *candidate pairs* (CP).

Parameter Choice and Validation

The parameter l is determined by two tests. The first test, the *triangle inequality test*, is performed over all candidate pairs. Under a time-reversible Markovian model, the evolutionary distances between homologous sequences should obey the triangle inequality condition: in a triplet of sequences any distance between two sequences must be less than or equal to the sum of the other two distances. Since these distances are estimates, this property is only expected to hold within a confidence interval.

$$d_{xz} \leq d_{xy} + d_{xz}$$

For example, sequences x and y share one domain, and sequences y and z share another domain, but sequences x and z are not related. The triangle inequality test detects such violations, likely to arise when inconsistent parts of the sequences are matched. With increasing (i.e. stricter) length criteria, a larger fraction of candidate pairs pass the triangle inequality test (Fig. 6.3).

In the second test, the candidate pairs are verified by assuming that the number of domains for both sequences should agree. The domain information is taken from the Pfam database, consisting of conserved protein regions and domains (Bateman et al., 2004). The fractions of proteins with the same number of

domains increases with stricter length tolerance.

Figure 6.3 also shows that, in addition to the results of the two validation tests, the number of orthologous relations (i.e. VP) decrease with increasing length criteria. There is a trade-off between sensitivity and selectivity. In general, the values are chosen to include all alignments that may correspond to orthology. We want to choose the best score criteria and length tolerance value ℓ, such that we do not drastically reduce the number of proteins in the matrix while removing violating pairs. Choosing $\ell = 0.61$ was found to be a good compromise in term of minimizing triangle inequality violations and numbers of different domains while still including enough ambiguous alignments.

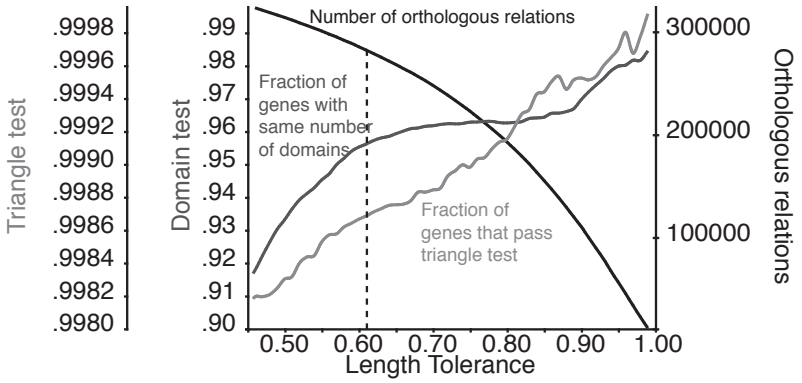


Figure 6.3: The fraction of candidate pairs that pass the triangle inequality test and the fraction that have same number of domains increases with stricter (higher) length tolerance. The number of predicted orthologous relations decrease with stricter length tolerance. A length criterion, $\ell = 0.61$, was used in this study.

6.1.2 Formation of Stable Pairs

In the second step of the algorithm, potential orthologs are detected by finding sequences more closely related to each other than either of them is to any other sequence. We call these sequences *stable pairs* [1] (SP)

To measure the relatedness of sequences, one can use either similarity scores or evolutionary distances. Most methods use the similarity score ("best hit") because it is directly obtained by the alignment process and the highest scoring sequence is usually the closest sequence. However, scores is not a direct measure of relatedness. In particular, they depend on protein lengths. Evolutionary distances such as PAM/JTT units, although more expensive to compute, constitute a sounder measure of relatedness, not least because they are additive (in their expected value) and have statistical properties that have been studied extensively.

[1] This name is used due to its close association with the stable marriage problem in computer science.

In this section, a tolerance criterion is used to allow the inclusion of more than one potential ortholog, this becomes necessary when a gene duplication occurred after speciation, a case ignored by most other methods based on pairwise sequence comparison. This tolerance criterion can be defined by including scores in an interval below the top score when using similarity scores or by using the variance of the distance estimates to compute a confidence interval when distances are used.

Consequently, we can classify orthology assignment methods based on pairwise sequence comparison in four categories (Fig. 6.4). Bidirectional best hits (BBH) is the most common approach and uses scores with no tolerance (e.g. Tatusov et al. 2003). Reciprocal best Blast hits (RBH) is based on Blast scores and uses a tolerance by including all hits within a P-value (Fulton et al., 2006). The reciprocal smallest distance (RSD) use evolutionary distances, but without a tolerance (Wall et al., 2003; DeLuca et al., 2006). OMA's stable pairs use distances to measure the relatedness between the genes and their variances as the tolerance criterion.

	Score	Distance
No Tolerance	Bidirectional best hits	Reciprocal smallest distances
Tolerance	Reciprocal best BLAST hits	Stable pairs

Figure 6.4: Different methods to find potential orthologs.

As mentioned above, the use of confidence intervals is necessary to take into account many-to-many orthologous relations, which arise when duplications have taken place after speciation. Additionally, no distance estimation is exact and thus it is possible that true orthologs do not have the shortest estimated distance.

Formally, a pair of sequences (x, y) from genomes X and Y is a stable pair if and only if, for all $x_i \in X, x_i \neq x$, and for all $y_j \in Y, y_j \neq y$:

$$\hat{d}_{xy_j} - \hat{d}_{xy} > k\sqrt{\sigma^2(\hat{d}_{xy_j}) + \sigma^2(\hat{d}_{xy}) - 2 \cdot \mathrm{cov}(\hat{d}_{xy_j}, \hat{d}_{xy})}$$

and

$$\hat{d}_{x_i y} - \hat{d}_{xy} > k\sqrt{\sigma^2(\hat{d}_{x_i y}) + \sigma^2(\hat{d}_{xy}) - 2 \cdot \mathrm{cov}(\hat{d}_{x_i y}, \hat{d}_{xy})}$$

where $\hat{d}$ is a pairwise ML distance estimate and k the tolerance parameter of the square root of the variance of the two distances (e.g. $\sigma^2(\hat{d}_{xy_j} - \hat{d}_{xy})$). An estimate of the variance can be obtained by the ML distance estimation procedure, while efficient estimation of the covariance for this case has been previously shown in Chapter 4.

Parameter Choice and Validation

The tolerance parameter k controls the trade-off between sensitivity (more true orthologs as stable pairs) and selectivity (few out-paralogs as stable pairs). The optimal value of k for our purpose is determined by using the *out-paralog test*.

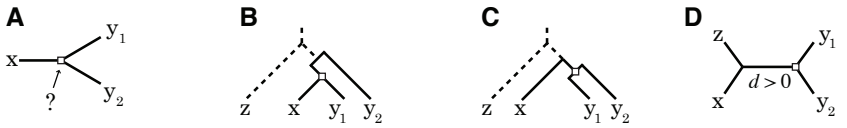


Figure 6.5: **A** What is the relation of sequences y_1 and y_2 with regards to x? Find on which branch to place the root by finding an out-group sequence z. **B** y_1 and y_2 are out-paralogs. **C** Duplication taking place after speciation, y_1 and y_2 are in-paralogs **D** y_1 and y_2 are in-paralogs if the distance d of the internal branch is larger than zero.

The out-paralog test should discriminate cases of one-to-many orthology from cases of out-paralogy. More precisely, it should determine whether the divergence of the sequences x, y_1 and y_2, illustrated in Figure 6.5A, is due to a speciation or a duplication event. This can be evaluated by finding on which branch to place the root. If the root is located on the branches leading to y_1 or y_2, this suggests that the divergence is a speciation and the sequence y_2 is an out-paralog (Fig. 6.5B). On the other hand if the root is on the branch leading to x, the divergence is a duplication and both the sequences in Y are orthologous to x. To find an suitable out-group z to place root, the information of a trusted phylogenetic topology is used. The sequence z is selected to be the gene closest to x in the out-group genome Z that is closest to the divergence of X and Y. Figure 6.5D shows the quartet that is the result of y_1 and y_2 being in-paralogs. If the length of the internal branch d for the given topology (i.e. the least square fit) is larger than zero, the sequences are accepted.

$$d = \frac{d_{zy_1} + d_{zy_2} + d_{xy_1} + d_{xy_2} - 2d_{zx} - 2d_{y_1y_2}}{4} > 0$$

To evaluate the parameter k, the fraction of SP passing the test is measured. Figure 6.6A depicts the decreasing fraction of passing stable pairs with increasing stable pair tolerance. Again, the problem is to reduce the amount of conflicting out-paralogs while not discarding interesting many-to-many relations. In this implementation, the distance for a more distant stable pair must be with $k = 1.81$ of the closest stable pair. At this value there is a locally increased amount of stable pairs that are passing the out-paralog test.

6.1.3 Verification of Stable Pairs

Although the construction of stable pairs is likely to identify the corresponding ortholog of each sequence, there is at least one special case in which it will sys-

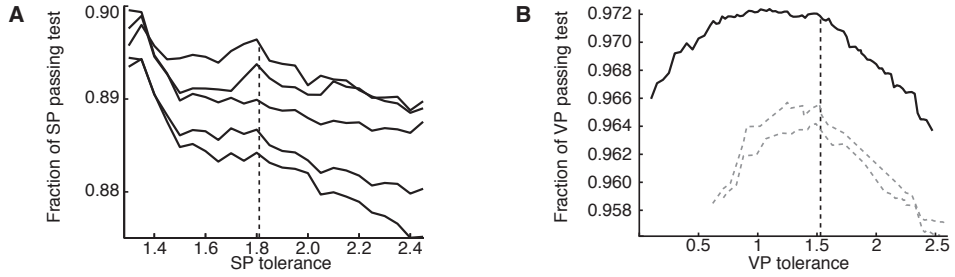


Figure 6.6: **A** The fraction of passing stable pairs with increasing SP-tolerance using 5 different length criteria. Increasing the tolerance results in a larger fraction of stable pairs that are suspected out-paralogs. **B** The fraction verified pairs passing the out-paralog test. The top curve is using the optimal previous parameters and the lower curves at other parameter setting also have locally optimal values.

tematically fail. This problem affects all pairwise approaches, and is shown in Figure 6.7A. An ancient duplication event is followed by two speciation events resulting in three species, X, Y and Z. In two of these species, a different one of the two duplicates was lost with the result that when comparing species X and Y, x_1 and y_2 have each other as the highest scoring match. In such a case, (x_1, y_2), although paralogs, will form a stable pair.

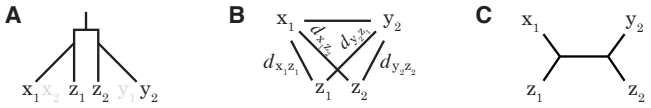


Figure 6.7: **A** An evolutionary scenario where genes have been lost asymmetrically. An ancestral gene is duplicated, followed by two speciation events, followed by the loss of genes x_2 and y_1. The remaining paralogous genes x_1 and y_2 could be mistaken for orthologs. **B** Scheme for verifying a stable pair between x_1 and y_2. If (x_1, z_1) and (y_2, z_2) form stable pairs and are the closest relatives then x_1 and y_2 are paralogs and are not verified. **C** The only possible quartet that can be formed when (x_1, z_1) and (y_2, z_2) are the closest related genes.

The purpose of the third step is to detect such stable pairs corresponding to non-orthology. The presence of a third genome Z, which has retained both copies z_1 and z_2 of the duplication event, can act as *witnesses of non-orthology*. The details of this procedure were treated in Chapter 5. Following the procedure we described there, the quartet in Figure 6.7C is the only possible quartet that is in agreement with the data provided in Figure 6.7A.

Each stable pair is verified by comparison to all other genomes. Stable pairs for which no witness of non-orthology could be found, are called *verified pairs* (VP) and are very likely to be orthologs. Furthermore, stable pairs that are not verified are defined as *broken pairs* (BP) and are likely to correspond to paralogs.

Parameter Choice and Validation

As an example of the validity of the step obvious examples from nature corresponding to the scenario in Figure 6.7 can be examined. It is easy to find many such examples of differential gene loss. For example in yeasts, an average of 5% of the stable pairs are broken (non-orthologous) using other yeasts as witnesses.

To find the confidence interval k, the out-paralog test (from section 6.1.2) is used to measure the fraction of passing verified pairs, which should now be further increased, since the verification ideally should identify all out-paralogs. In figure 6.6B the fraction of passing VP as a function of tolerance is drawn. There is a dependence between the VP- and the SP-tolerance. Increasing the VP-tolerance has little effect if the SP-tolerance is low (i.e. only the closest stable pairs were chosen). The decrease of the number of VP with stricter VP-tolerance is much less than with stricter SP-tolerance. Hence, it is reasonable to have a stricter VP-tolerance than SP-tolerance in order to maximize the coverage. A VP-tolerance of $k = 1.53$ was chosen, which is a good value to minimize the inclusion of errors.

6.1.4 Clustering of Orthologs

The final step of the algorithm creates groups of orthologs. A sequence in one genome may have several verified pairs to sequences in other genomes that correspond to several orthologous relation (co-orthologs). In certain cases, it is desirable to have the subset of the closest orthologs with a one-to-one relation. These are useful for inferring function and constructing phylogenetic trees.

A clique algorithm is used to search for maximal completely connected subgraphs in a graph, where the vertices are genes and the edges are verified pairs. To compute cliques, algorithms exit to maximize either the size of the clique (number of vertices) or the total weight of cliques (sum of edge weights). Figure 6.8A shows a graph with edges between all vertices except (z_1, z_2) and (z_1, y_2), which are paralogous relations. The highest scoring partition is $\{w_1, x_1, z_1\}$, $\{y_2, z_2\}$, which has the total sum of edge weights of $2 \cdot 800 + 1100 + 900 = 3600$. The score is higher than the highest scoring maximum size clique $\{w_1, x_1, y_2, z_2\}$, $\{z_1\}$, where the sum of the scores is $4 \cdot 100 + 800 + 900 = 2100$. Hence, a smaller clique is chosen due to higher edge weights, which correctly assigns orthologs according to the evolutionary scenario in Figure 6.8B, assuming that the subscripts correspond to functionality.

Finding cliques is a NP-complete problem. The implementations used here are based on a reasonable effective approximation of the vertex cover problem (Balasubramanian et al., 1998). Employing cliques for groups is a strict requirement, because if a edge is missing (i.e. a broken pair) we assume that the sequences are not orthologous. But together with the output from the previous steps (i.e. CP, SP and VP), the information necessary to deduce the evolution-

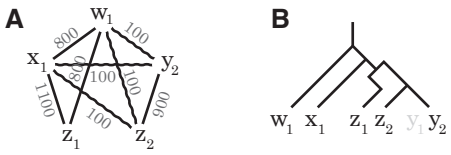


Figure 6.8: **A** Example of a graph containing one 4-clique, three 3-cliques, and eight 2-cliques. The highest edge scoring partition of the graph is $\{w_1, x_1, z_1\}, \{y_2, z_2\}$. **B** Possible evolutionary scenario that corresponds to the graph.

ary relations is complete. Each of the cliques constitutes an *orthologous group*, where the pairs of sequences in an orthologous group are denoted *group pairs* (GP), corresponding to close orthologs. The members of a group have at most one close ortholog in each other genome.

Parameter Choice and Validation

To validate our methods and to compare the different algorithms for building the cliques, a species tree is built from the orthologous groups for each algorithm and the fit of the data to the tree is measured using the dimensionless index (Gil et al., 2005). This technique assumes that better data will have the best fit of the tree.

To verify, 100 trials using a various number of genomes and different taxa were completed using four different versions of clique. *Maximum size clique* chooses the largest clique in the graph starting with the highest scoring edge (but does not use any other edge information). *Maximum size score clique* is an extension that uses the sum of the edge weights and chooses the higher scoring clique from several maximum cliques of same size. The above described algorithm, *maximum edge weight clique*, is used twice, first using the *scores* and then the *distances* as edge weights.

The versions of the maximum edge weight clique algorithm using edge weights perform better in general than versions of maximum size clique (in 82 % of the trials). This supports the argument of the hypothetical example in Figure 6.8. We have chosen scores as edge weights, over distances, which twice as often give better fit for the build trees.

6.2 Validation and Comparison

> This section is joint-work with Adrian Altenhoff. It was published in Altenhoff and Dessimoz (2009)

A large variety of methods for predicting orthologs and the resulting databases

have appeared in recent years (Tatusov et al., 1997; Remm et al., 2001; Li et al., 2003; Dessimoz et al., 2005; DeLuca et al., 2006; Wheeler et al., 2007; Hubbard et al., 2007; Jensen et al., 2008). But although the accuracy of the predictions highly impacts any downstream analyses, there are only few comparative studies of the quality of the different prediction algorithms (Hulsen et al., 2006; Chen et al., 2007). This paucity can be attributed to at least three major challenges. The first challenge resides in the multiple and sometimes intrinsically conflicting definitions of orthology (Ouzounis, 1999; Fitch, 2000; Jensen, 2001). The original definition of Fitch (1970) is based on the evolutionary history of genes: two genes are orthologs if they diverged through a speciation event. On the other hand, given that orthologs often have similar function, many people uses the term orthologs to refer to genes with conserved function. Yet another definition is used in some studies of genome rearrangement, in which the ortholog refers, in the event of a duplication, to the "original" sequence (Marron et al., 2004), which remains in its genomic context.

The second challenge resides in the difficulty of validating the predictions. Take the case of phylogenetic orthology. Gene tree inference can be a notoriously difficult task, but it is usually precisely in difficult cases that the performances of methods can be differentiated. Indeed, in simple cases, most methods perform equally well. Validation of the definition based on function is not easier: orthology is in this context arguably *impossible* to verify because there is no universally applicable, unequivocal definition of conserved function, that is, the required similarity in terms of regulation, chemical activity, interaction partners, etc. for two genes to qualify as orthologs often varies across studies. For instance, in some wet lab experiments (Azevedo et al., 2002; Elliott et al., 2002), two genes are only considered orthologs if they have the ability to complement each other's function.

The third challenge is of practical nature: to compare the different orthology inference projects, their methods must either be replicated on a common set of data, or the results produced by the different databases must be mapped to each other for comparison. Replication is not always possible, because some projects depend on human curation, or are not documented in detail. Mapping data is complicated by the lack of homogeneity in the sources of genomic data used by the different projects. The resulting intersection sets are often relatively small and may not be representative.

In this section, we provide an in-depth comparison of the prediction from 11 major projects, including OMA (Dessimoz et al., 2005), our own orthology inference effort. We try to address the aforementioned challenges by testing phylogenetic and functional definitions of orthologs, using a variety of tests. We took the approach of comparing the inferred orthologs available from the different projects, which required mapping the data between projects. The rest of this introduction provides a description of the projects retained here, a review on the representation of orthology in those projects so to provide a common

basis for comparison, and finally, some words on our sequence mapping strategy.

Projects under Scrutiny

In this study, we consider publicly available databases of orthologs that distinguish themselves by popularity, size, quality, or methodology. One of the oldest large-scale orthology database is COG (Tatusov et al., 1997, 2003) and its eukaryotic equivalent KOG (Tatusov et al., 2003), which despite no recent update are still considered by many authors as the standard orthologs databases. Their reliance on manual curation make them not scalable to all complete genomes. Unsupervised orthology assignment requires more sophisticated algorithms, such as those of Inparanoid (Remm et al., 2001; Berglund et al., 2008), OrthoMCL (Li et al., 2003) or EggNog (Jensen et al., 2008). We also investigated the results of RoundUp (DeLuca et al., 2006), interesting for its relatively large size and its use of pairwise evolutionary distances between genes to detect orthology. OMA (Dessimoz et al., 2005; Roth et al., 2008), our own orthology assignment project, is also based on evolutionary distances but takes into account the variance of the distance estimates and try to exclude pseudo-orthologs arising from differential gene losses using third-party species. A very different approach is taken in the orthology prediction of Ensembl Compara(Hubbard et al., 2007), which is based on inference and reconciliation of gene and species trees. Homologene (Wheeler et al., 2007) uses a pairwise gene comparison approach combined with a guide tree and gene neighborhood conservation to group orthologs, but the details of their methodology are unpublished. Finally, we also compare the results to the standard approaches of bidirectional best-hits (BBH) (Overbeek et al., 1999), common in ad-hoc analyses, and reciprocal smallest distance (RSD) (Wall et al., 2003). The size of the different projects is depicted in Fig. 6.9.

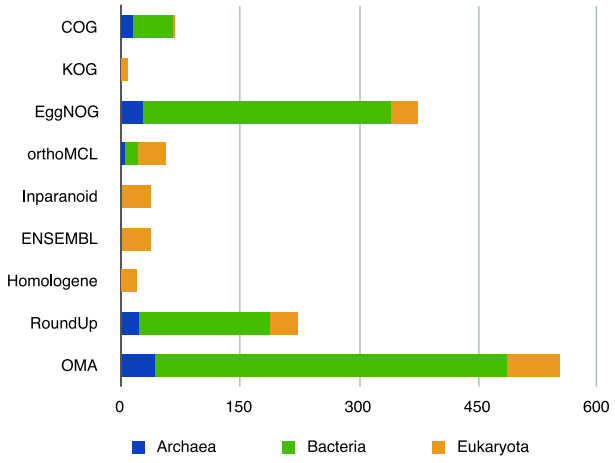


Figure 6.9: Number of complete genomes analyzed by the different projects

Grouping of Orthologs

Orthology is a relation over pairs of genes. However, few projects (namely Ensembl Compara, OMA and RoundUp) explicitly provide output of all pairs of predicted orthologs. This representation, although precise, has practical drawbacks: on one hand, it scales poorly (quadratically with the number of genes analyzed), and on the other hand, it does not present the predictions in a particularly insightful way. To solve these issues, many projects cluster pairs of orthologs into groups. This grouping process is not trivial, because orthology, at least when the phylogeny-based definition applies, is a non-transitive relation.

The most common approach (taken by all other projects) is to form groups of orthologs and "in-paralogs". The relations in- and out-paralogs were defined by Remm et al. (2001), and are used to distinguish between paralogs from recent and old duplication events respectively. Formally, these two relations are not defined over a pair, but over a triplet: two genes and a speciation event of reference. Two genes are in-paralogs with respect to a particular speciation event if they are paralogs *and* their duplication event occurred after that speciation event of reference. They are out-paralogs if they are paralogs *and* their duplication event occurred before the speciation event of reference. See Fig. S1 (cap. a) in Altenhoff and Dessimoz (2009) for an example. Unfortunately, the fact that in- and out-paralogy are ill-defined in the absence of a clear speciation event of reference is underappreciated in the literature. We now come back to the description of groups of orthologs and in-paralogs: such groups are constructed such that every pair of genes in the group is either orthologous or in-paralogous with respect to the last speciation event in their clade, that is, such in-paralogs are genes inside the same species resulting from a duplication event that occurred *after* all speciation. Consequently, in such groups, the implication is that gene pairs are orthologs if they belong to the different species, else they are paralogs. Note that this grouping approach shows its limits when one or several duplication events have occurred after the first, but before the last speciation events. In such cases, not uncommon in Eukaryotes, the non-transitive nature of orthology makes it impossible to partition all genes in such groups without losing orthologous relations (see Fig. S1 (cap. b) for an example). In OMA for instance, groups of orthologs include less than half of all predicted pairwise orthologous relations (Table S1). This problem does not affect Inparanoid, because it provides predictions for each pair of species separately, and so in every case, there is only one speciation event.

Mapping Strategy

To perform a fair comparison of the different predictions, a common set of sequences must be established. Unfortunately, the different projects vary considerably in their sizes, the type of genome analyzed and the origin of the protein

sequences used. In fact, some projects have no overlap at all, and therefore comparison on a common set of sequences for all projects is not possible. Instead, we performed pairwise project comparisons with OMA (which includes the largest amount of sequences), and then we repeated the tests on an intersection set with only the most competitive projects.

First, sequences from the different projects were mapped to OMA's only if they were identical, between consistent genomes. This strict requirement avoids reliance on IDs, which may refer to different sequences depending on the genome version, and also the problem of different splicing variants. Tables S1 and S2 in Altenhoff and Dessimoz (2009) present some statistics on the mapping procedure of the sequences and the predictions.

In pairwise tests, we compared the pairs of mappable proteins identified as orthologs by the different methods with those identified by OMA. In joint tests, we computed the intersection of the mappable sequences of each project under consideration, and compared pairs in this intersection set identified as orthologs by the different methods.

6.2.1 Results and Discussion

In this section, we present all results, first in pairwise comparisons between each project and OMA, then in joint comparisons of the most competitive projects. We group the tests according to the definition of orthology that they should verify: the first two tests verify orthology based on phylogeny, while the four following tests verify orthology based on on function. At the end of the section, we justify the absence of tests that were not included here, and compare our results with the previous study of Hulsen et al. (2006).

Phylogeny-based definition

According to the phylogenetic definition, two homologous genes are orthologs if they diverged through a speciation event. Therefore, the phylogenetic tree of a set of orthologs (a set of genes in which any pair is orthologous) has by definition the same topology as the corresponding species tree.

Gene Tree Reconstruction We reconstructed gene trees from species with an accepted phylogeny and predicted orthologs from the different projects using two independent methods and software packages (distance-trees from Smith–Waterman pairwise alignments and ML trees from multiple sequence alignments), and compared the congruence of the resulting trees with the species trees using the fraction of correct splits, which is defined as one minus the Robinson-Foulds (RF) split distance measure (Robinson and Foulds, 1981). The RF distance is defined as the normalized count of the bipartitions induced by one tree, but not by the other. The experiment was performed on sets of

bacteria, of eukaryotes and of fungi. Note that this test can only verify the correctness of the reported orthologs (the specificity) for each project, but not the false-negative rate (the sensitivity).

Though some level of incongruence is expected from errors in the input data or in the tree reconstruction, these perturbations affect, on average, all methods equally. Results for ML trees are presented in Figure 6.10 while distance trees are presented in Figure S2 in Altenhoff and Dessimoz (2009). As a first observation, it is comforting to see that the choice of tree reconstruction method does not affect the ranking or the significance of the results. It appears that COG, EggNog and OrthoMCL suffer from comparatively high false-positive rates, which is reflected in the significantly reduced amount of correctly reconstructed gene trees. The high-level of non-orthology in the COGs database is consistent with previous reports (Dessimoz et al., 2006a; van der Heijden et al., 2007). The differences among the better performing projects are small. The predictions of Ensembl Compara, being made on the basis of tree reconciliation, could have been expected to perform better than pairwise gene comparison methods, but their predictions are in fact slightly worse than OMA in this test. The generic BBH and RSD methods are also dominated by OMA in the pairwise comparison. Note that the intersection set is not large enough to allow the ranking of the best performing projects (OMA, RoundUp, Homologene, Inparanoid). Finally, KOG covers too few genomes for inclusion in this test.

Benchmarks from literature The accuracy of the different projects in terms of the phylogeny-based definition of orthology was also assessed from manually curated gene trees or reference orthology sets from four studies (Engelhardt et al., 2006; Dessimoz et al., 2006a; Hughes, 1998; Hulsen et al., 2006). In addition, this method allows us also to estimate the true positive rate (sensitivity) of the different projects, that is, the fraction of reported orthologs over all bona fide orthologs. Figure 6.11 summarizes the performance of the projects on those difficult phylogenies. In the pairwise project comparison (Fig. 6.11a), the relative difference between the true positive rate of OMA and the comparative project versus their relative difference of the false-positive rate is shown. Strictly speaking, only pairwise comparisons with OMA should be made, since the underlying protein sets are not the same across different projects and thus, the difficulties of prediction may differ. On the other hand, Figure 6.11b) compares a selection of the projects on a common set of sequences. The results for projects analyzed in both contexts have good agreement, which suggest that pairwise comparisons (which are based on more data) also provide a global picture across projects. The confidence interval around the points are relatively large, due to the limited data used in this test.

First, COG/KOG/EggNog show higher sensitivity (true positive rate), but at the cost of very low specificity (high false-positive rate). This is a clear sign

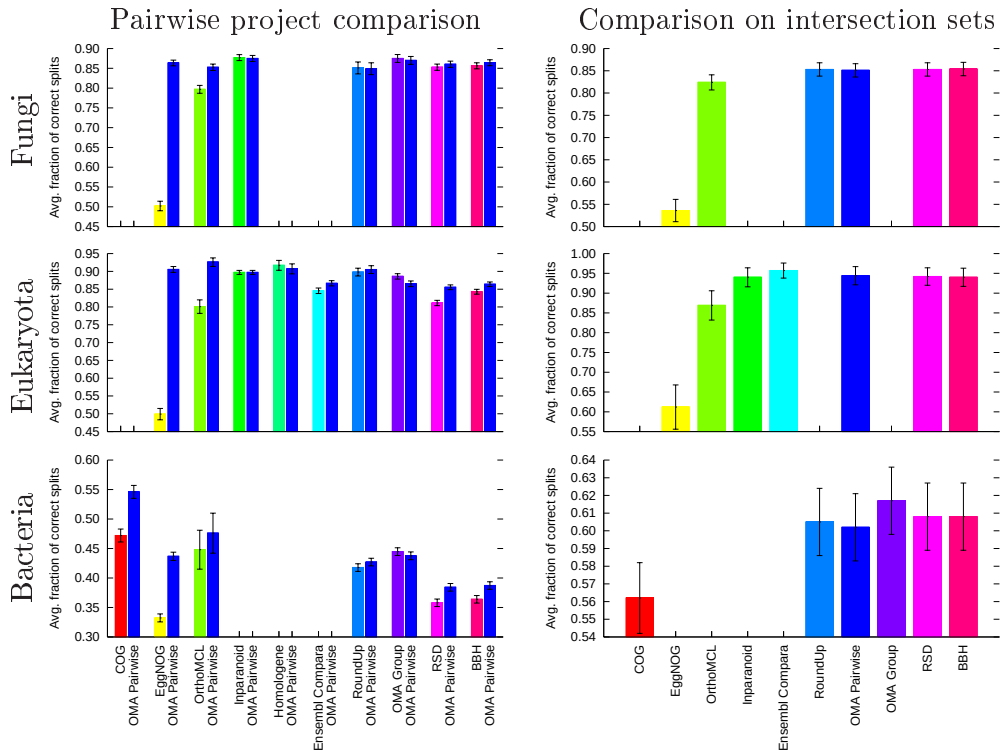


Figure 6.10: Results of the phylogenetic tree test. The mean percentage of correct split of ML trees for gene trees from three different kingdoms are shown. The higher the values, the better the gene trees agree with the species tree. On the left, the pairwise results between every project and OMA are shown, whereas on the right, the result for the comparison on the common set of proteins of a larger number of projects is shown. Note that the pairwise project comparisons are made based on varying protein sets, and thus can not be compared to each other. Error bars indicate the 95 % confidence intervals of the estimated means.

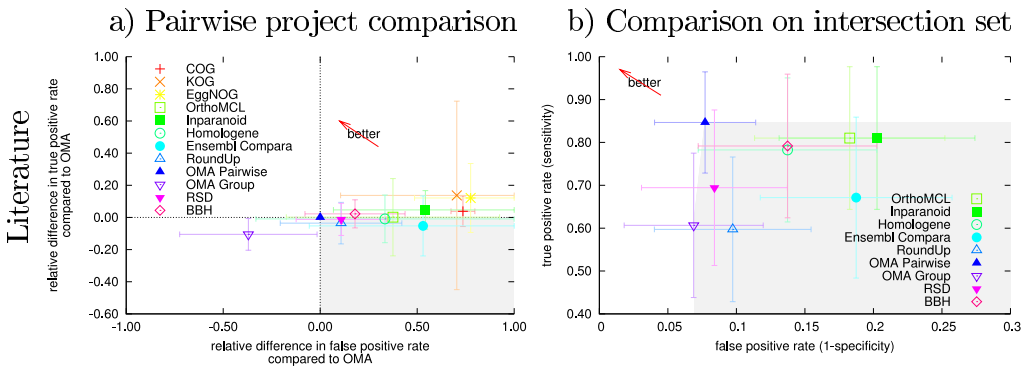


Figure 6.11: Results of benchmarks from literature. Performance on manually curated gene trees from 4 published studies (Engelhardt et al., 2006; Dessimoz et al., 2006a; Hughes, 1998; Hulsen et al., 2006). a) The pairwise outcome of every project against OMA are shown, indicated with the relative difference of the true positive rate between OMA and its counter project versus their relative difference of the false-positive rate. b) Performance for the protein intersection data set. Shown are the true positive rate (sensitivity) versus the false-positive rate (1 - specificity). In both plots, the error bars indicate the 95 % confidence interval and the "better arrow" points into the direction of higher specificity and sensitivity. Projects lying in the gray area are dominated, in a) by "OMA Pairwise" and in b) by at least one other project.

of excessive clustering. It also appears that the relatively higher false-positive rate of OrthoMCL is not compensated by a significantly higher true-positive rate. Ensembl and RoundUp report fewer orthologs, but the accuracy of their predictions is not significantly higher than OMA or even BBH. Inparanoid, with its relatively low specificity, is doing worse than in the previous test. But overall, the agreement with the previous test in terms of false-positive rate is good, even though the testing methodology is here very different.

Function-based definition

One of the main application of orthology is the propagation of functional annotation, because orthologs often have a similar function. In fact, this application is so prominent that many authors use the term "orthologs" to refer to genes with conserved function in different species. As mentioned in the introduction, this definition is ambiguous. Therefore, we could only test specific aspects of what can be implied by "conserved function".

The four tests presented here evaluate the similarity of predicted orthologs in terms of Gene Ontology annotations, enzyme classification numbers, expression level, and gene neighborhood conservation. In the following, we present and discuss their results.

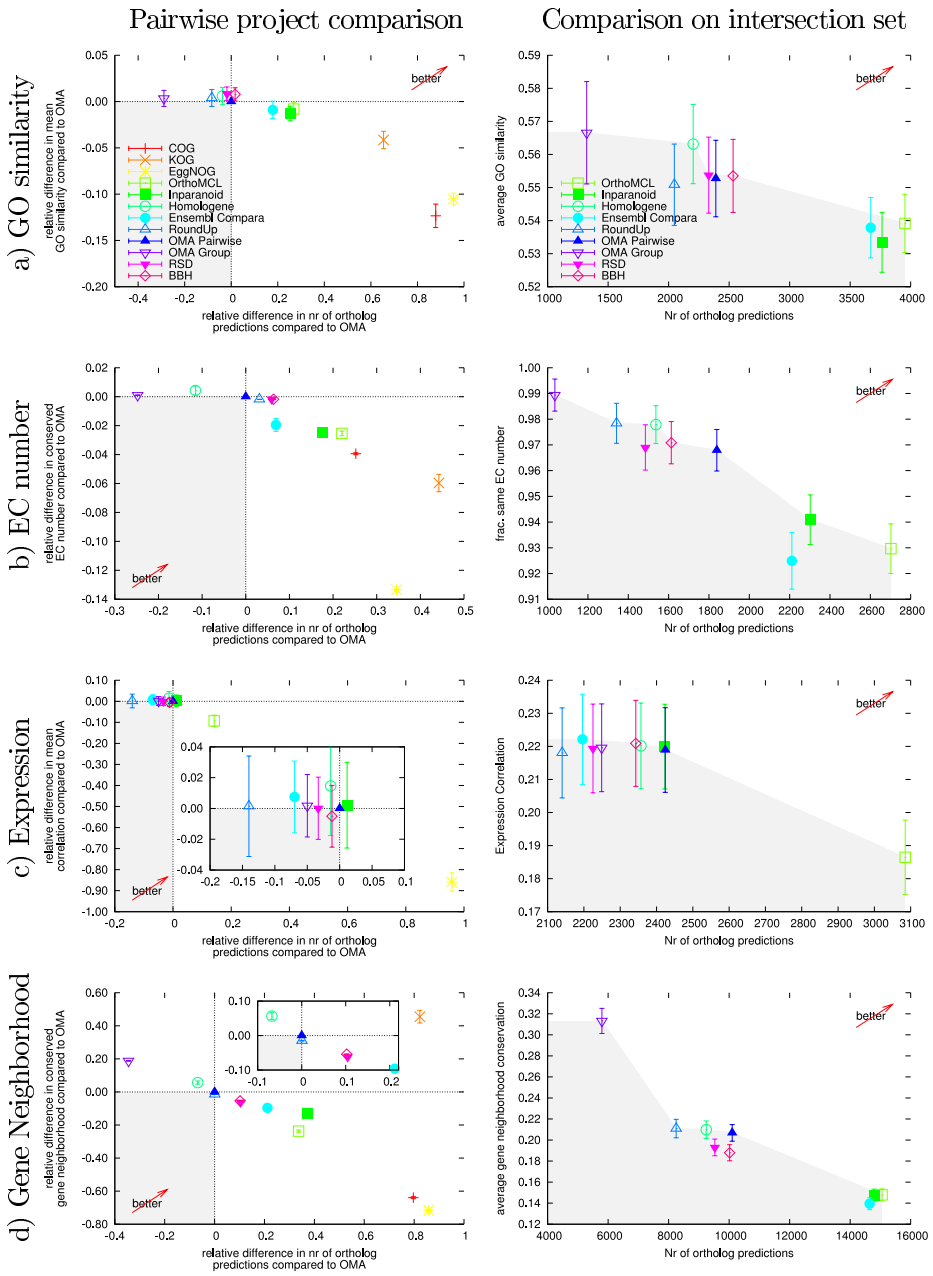


Figure 6.12: Results of functional conservation tests for GO similarity, EC number expression correlation and gene neighborhood conservation. In the pairwise project comparisons (left) the relative difference of functional similarity between OMA and its counter project versus the relative difference of the number of predicted orthologs are shown. In the comparison on the intersection set (right), the mean functional similarity versus the number of predicted orthologs on the common set of sequences are shown. The vertical error bars in all the results state the 95 % confidence interval of the means. The "better arrow" indicates the direction towards higher specificity and sensitivity. Projects lying in the gray area are dominated by "OMA Pairwise" in the pairwise comparison (left) and by at least one other project in the intersection comparison (right).

Gene Ontology In the first test, we assessed the agreement in gene ontology (GO) annotations (Harris et al., 2004) between predicted orthologs, only considering annotations with experimental support (Evidence codes IDA, IEP, IGI, IMP and IPI). Indeed, annotation obtained automatically are for the most part done using the methods that we are testing here: inclusion of this information would cause a serious circular dependency. We measure the level of conservation in terms of GO annotation using the similarity measure developed by Lin (1998) which computes for a pair of terms a score between 0 (unrelated) and 1 (identical terms) using the hierarchical structure of the GO terms and their frequencies.

Figure 6.12a) shows the average similarity of GO annotations in pairs of orthologs from the different projects. The mean similarity of all projects falls in a relatively small range, and is quite low. COG/KOG/EggNog do comparatively many predictions, but the average similarity score is significantly lower. Hence, the results of COG/KOG/EggNog are particularly suited for coarse-grained functional classification. On the other hand, if a high functional similarity is desired, the relatively simple BBH approach dominates more sophisticated algorithms such as RoundUp and Homologene (which does fewer predictions at same degree of similarity) or OMA (which does only few more predictions, but significantly lower degree of similarity). This result suggests that sequence similarity is a stronger predictor of functional relatedness than the evolutionary history of the genes. At mid specificity level, OrthoMCL outperforms Ensembl Compara and Inparanoid, yielding many more predictions at roughly the same similarity level.

Enzyme Classification A second measure for the quality of the orthologous assignments with respect to function can be obtained from the enzyme classification numbers (EC), which strictly depend on the chemical reaction they catalyze. Thus, we could expect in general that orthologous enzymes have identical EC number. Obviously, this test can only be applied to the small and rather specific fraction of genes that are enzymes. The results must be interpreted accordingly. As reference, we use the EC database curated by the Swiss-Prot group (Bairoch, 2000).

Figure 6.12b) shows the difference between the projects. The results are very similar to the GO annotations test, but BBH is not as good, and Inparanoid has now moved to the Pareto frontier, i.e. it is not dominated by OrthoMCL here.

Correlation in Expression Profiles In this third test, conserved function is assessed using protein expression profiles from large-throughput experiments. In such data, proteins with similar function are expected to have similar expression profiles. We measured this similarity by computing the average correlation between the expression profiles of putative orthologs between the human and mouse genomes as presented by Liao and Zhang (2006). Some projects, such as

COG and KOG did not have sufficient mappable proteins in those genomes to be considered here. Although certainly relevant for many researchers, Human–Mouse orthologs hardly constitute a representative sample of all orthologs, and thus here too their assessment should be extrapolated to all predictions with prudence.

The results are shown in figure 6.12c). In general, the correlations found are relatively low and within a narrow band. This range is however consistent with the results of Liao and Zhang. Most projects perform very similarly, with average correlation mostly within 2 standard deviations and number of predicted orthologs differing by less than 10%. Predictions by OrthoMCL have significantly lower average expression correlation, but in absolute terms, the difference is modest, and they have a significantly higher number of predictions. Finally, with 40 times more predictions but almost no correlation in terms of expression, EggNog does not appear to provide useful information to propagate expression levels.

Gene Neighborhood Conservation To assess the quality of the ortholog assignments on the basis of genome structure, conservation of the gene arrangement on the chromosomes has been used to validate functional orthology in previous studies (Notebaart et al., 2005; Hulsen et al., 2006; van der Heijden et al., 2007). Conservation of the genomic context is indeed a strong indicator of function conservation. Note that gene neighborhood conservation is not a reliable indicator of phylogenetic orthology: not only speciation, but also duplication of DNA segments stretching over more than a single gene, such as operons, preserve the immediate neighborhood.

In this test, we measure the fraction of orthologs that have at least one pair of flanking orthologs (see Appendix A2). The results are presented in figure 6.12d). The pairwise project comparison shows results consistent with previous tests, with the exception of KOG, which appears to perform extremely well in the pairwise test with OMA. However, the results are based on relatively few and distant genomes that have low absolute conservation values (see raw data in Altenhoff and Dessimoz 2009). In such a context, the much larger number of ortholog predictions of KOG significantly increases the probability of having adjacent pairs of orthologs due to chance only.

In terms of methodology, Homologene is the only project that uses gene neighborhood conservation as part of their methodology. The details of how precisely such information is exploited in their inference process remain unpublished, but the present test does not show significant improvement over other approaches in terms of neighborhood conservation.

About Absent Tests

We now justify the absence of three other tests that have been previously reported in the literature. We did not verify orthology based on common keywords in the annotation because those are often assigned on the basis of sequence similarity or using the methods that are tested here: this would introduce circularity in the testing strategy. Nor do we test orthology based on conservation in protein-protein interaction (PPI). Though there are studies such as Bandyopadhyay et al. (2006) reporting modest but measurably higher PPI between some orthologs, it remains unclear to us how current PPI data can be turn into a test of orthology for the following two reasons: first, PPI data show large variations in reliability and completeness across experiments and species, but more importantly, the general problem of matching (or "aligning") networks is computationally hard (Dost et al., 2007). To reduce complexity, most approaches, including Bandyopadhyay et al. (2006), strongly constrain the network alignment using heuristics based on sequence similarity. In the present context, this too would introduce circularity in the validation. Finally, we do not use the latent class analysis approach of Chen et al. (2007). This approach computes ML estimates of false-positive and false-negative rates for all the projects directly from the various ortholog predictions (the data) and a parameterized multivariate distribution of the errors (the model). This looks very attractive, because the assessment does not require any of the external information used in the tests described here. Our critique with this approach is that their results are conditional on their error model, which is not verified (at least not in the context of evaluating orthology inference projects). In a sense, the issue of validation is shifted to their error model, but remains open.

Comparison with Results of Hulsen *et al.*

The main other systematic evaluation of orthology prediction projects is from Hulsen et al. (2006). Smaller in scope, their study tested functional orthologs predictions in Human–Mouse and Human–C. elegans, using a manually curated reference set of orthologs, expression correlation and conservation of gene neighborhood. They compared BBH, Inparanoid, OrthoMCL, KOG, as well as two other methods not under analysis here ("PhyloGenetic Tree" and "Z 1 Hundred").

On the tests and data common to both studies, the results are largely consistent (data not shown). However, we observed that considering only two pairs of species can introduce significant biases in the assessment: as it turns out, the overwhelming majority (89.1%) of all orthologous pairs predicted by Inparanoid on Human–Worm data arise from one large cluster of olfactory-type receptor proteins (cluster number 4604). This very atypical distribution explains why the results are so different from those for the HUMAN-MOUSE genome pair (see

Fig. 3 and 4 from Hulsen et al. 2006).

They concluded that in terms of functional orthology, Inparanoid performed best overall, while also noting that the appropriate method depends on the user's requirements in terms of sensitivity and specificity. As our results show, this trade-off remains true today, but Inparanoid is no longer the overall best performer: besides being one of the projects with fewest genomes under analysis, there are other projects with either higher specificity, or with higher sensitivity; this reduces the scope of applications in which it constitutes an appropriate choice.

6.2.2 Conclusions of Comparison Study

Accurate ortholog prediction is crucial for many applications ranging from protein annotation to phylogenetic analysis. There is a number of publicly available orthology databases but little is known about their performances. In this study we compared 11 different projects and methods by submitting them to a variety of tests with respect to both phylogenetic and functional definitions of orthology.

The results obtained in the tests for both definitions are consistent, and allow us comparison of the different projects on an objective basis.

In phylogenetic tests, OMA and Homologene showed the best performances. The same two projects do also best in functional tests if a high level of specificity is required. At a somewhat lower degree of specificity, but at a higher coverage, function-based tests suggest that OrthoMCL outperforms Ensembl Compara, and to a less extent Inparanoid. Finally, for applications that only require coarse-grained functional categories, EggNog provides the largest coverage.

In terms of methodology, the one project based on gene and species tree reconciliation, Ensembl, had overall decent performances, but was overperformed by some of the best pairwise approaches. This suggests that tree reconciliation, although more powerful a method in theory, is still limited in practice by its high computational complexity. Another surprise is the good overall performance of the simple BBH approach. Although the method is restricted to 1:1 orthologs, the derived relations show good comparative accuracy in terms of Fitch's definition. Orthologs predicted by BBH also show close functional relatedness. This result probably explains why many people use ad-hoc BBH implementations for their analyses rather than a more sophisticated orthology method.

Beyond the accuracy aspects discussed in the present work, other factors will also affect the choice of orthologs database, such as the number of genomes analyzed, the state of maintenance, the availability of the predictions, or the usability of the web-interface.

There is still improvement potential in orthology inference, and we expect

much development in the coming years. We hope that the present work helps setting performances standards. But it is also the responsibility of upcoming orthology assignment projects or releases to clearly state the definition of orthology they pursue, to explain their grouping strategy, and in the very least to demonstrate the improvement of their methods over basic methods such as BBH or RSD.

6.3 OMA Browser

This section is joint-work with Adrian Schneider and Gaston Gonnet. It was published in Schneider et al. (2007)

6.3.1 Implementation

The OMA browser is a web application using as basis *Darwin* (Gonnet et al., 2000), a software package for bioinformatics developed within our group. Benefits include efficient data-structures for biological sequences, and a large library of functions for bioinformatics analyses. While most data is precomputed, some computations are performed in real time or on user request.

6.3.2 Protein-centric view

Proteins of interest can be accessed through a search interface. Searches can be conducted on identifiers, accession numbers or descriptions. Furthermore, we provide a fast sequence search on the 1.6 million protein sequences that takes as input any sequence substring.

The protein view provides cross-references, mainly to GenBank, Ensembl or Swiss-Prot/UniProt (for more than 92 percent of the proteins a link to another databases can be provided), annotations as found in the source database, chromosome/locus information as well as links to the different types of orthologs and the corresponding OMA group.

6.3.3 OMA group-centric view

The OMA group detail view contains several 'tabs'. The main view is a list of all proteins in the group with an identifier and a description, providing the complete information of the protein family. Since the OMA project itself is automated, no additional annotation by hand is performed. A short description of the OMA group is inferred from the available sequence descriptions.

A multiple sequence alignment of all proteins of a given group can be requested and will be displayed after computation is completed.

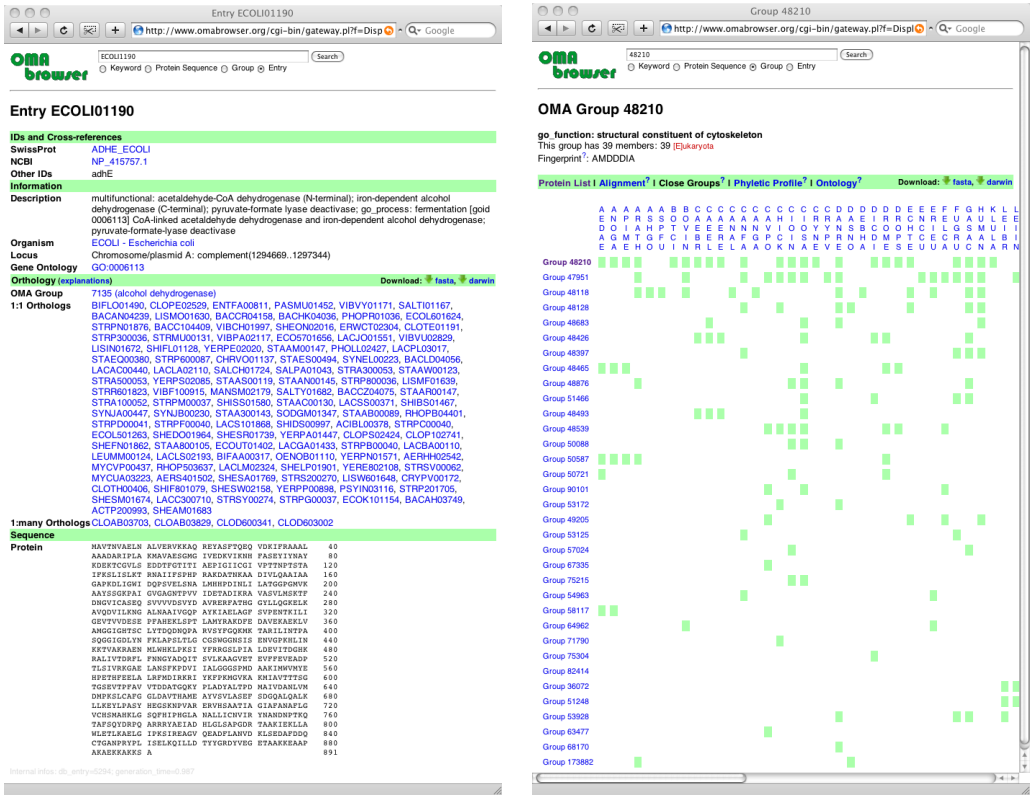


Figure 6.13: Protein-centric (l.) and group-centric view (r.)

Related groups can be explored by two options: 'Close groups' are OMA groups in which at least one protein is orthologous to a protein in the current data-set, while 'Phyletic profile' lists groups having similar patterns of presence/absence across species. This is a possible way of identifying interacting protein families, where either all members must be present in a genome or none of them are required. Whenever available, Gene Ontology (Harris et al., 2004) annotations of the different proteins of the group can also be compared and provide additional indication about the functionality of the proteins.

6.3.4 Data export and integration

All the data can also be downloaded from the browser web page in numerous formats: Fasta, text (list of IDs), *Darwin* database (SGML format) or in a COG-compatible format. These files are available for all OMA groups in one file or for each group individually.

The OMA Browser offers also a SOAP-based application programming interface (API), allowing for the integration of the OMA data into applications or web services.

7

Lateral Gene Transfer Detection

This chapter is joint work with Daniel Margadant and Gaston Gonnet. It was published in Dessimoz et al. (2008)

7.1 Introduction

Lateral gene transfer (LGT), or horizontal gene transfer (HGT), is widely recognized as a major force in prokaryotic genome evolution, but the study of its nature and extent is constrained by the limitations of current methods for LGT detection (Philippe and Douady, 2003; Lawrence and Ochman, 2002). These methods can be divided in two broad categories: parametric methods and phylogenetic methods. In parametric methods, sequence properties such as nucleotide composition (Lawrence and Ochman, 1997, 1998), dinucleotide frequencies (Karlin, 1998), codon usage biases (Moszer et al., 1999; Mrazek and Karlin, 1999; Medigue et al., 1991), or, more recently, nucleotide substitution matrices (Hamady et al., 2006) are calculated for a specific gene and compared with the rest of the genome. A transferred gene has parameter values typical for its donor genome, which makes it distinguishable from the recipient genome. For this reason, the method can only detect LGT events taking place between organisms with significantly different patterns of evolution. Furthermore, parametric methods are limited to recent LGT transfers because the transfered sequences adapt to their new host relatively rapidly (Lawrence and Ochman, 1997). Lastly, some native genes may have atypical nucleotide composition for reasons other than LGT.

Phylogenetic methods identify LGT events by analyzing the discrepancy between the phylogeny of laterally transfered genes and their host genomes. Therefore, most phylogenetic methods consist of inference of gene and species trees, and their reconciliation (Beiko et al., 2005; Gophna et al., 2003). Other methods, such as Lawrence's rank correlation test (Lawrence and Hartl, 1992) or Clarke's phylogenetic discordance test (Clarke et al., 2002) use unexpected sequence similarity scores to detect LGT, and do not require the inference of gene trees. To distinguish between the two types, we refer to the former by *explicit*, the latter by *implicit* phylogenetic methods. Explicit methods have the potential of describing in detail LGT events (involved species, direction of transfer, time of the transfer), but suffer from the difficulties associated with the inference of gene trees, a task both computationally expensive and error-prone. On the other hand, the two implicit phylogenetic methods mentioned here are fast and robust, though limited by their reliance on similarity scores, which do not always reflect phylogeny (Koski and Golding, 2001) in the first place, and by the relative coarseness of their underlying models, which limits their detection power.

In this chapter, we introduce a new phylogenetic method for LGT detection, which we call DLIGHT (Distance Likelihood based Inference of Genes Horizontally Transfered). Based on evolutionary distances and applied in a probabilistic framework, it combines the speed, the lack of gene tree requirement, and the robustness of implicit methods with the high level of details obtained by explicit methods. The next section presents the algorithm, and is followed by validation using simulation and biological data.

7.2 Method

7.2.1 Preliminaries

Definition (family of orthologs)**.** A set of sequences (genes or proteins[1]) $f = \{x_1, x_2, ...\}$ is a family of orthologs if all pairs of sequences (x_i, x_j) in f are either orthologs or xenologs through orthologous replacement. We denote the set of all such families by F.

DLIGHT's objective is to detect LGT in such families of orthologs. In the above definition, we require that the families have no paralogs (to detect paralogs, we can use the method introduce in Chapter 5). This also ensures that there is at most one sequence per species in any family of orthologs. Thus, a sequence is also uniquely referenced by the pair (f, g), where f is a family that contains the sequence and g the species it belongs to (or the genome – the two terms are used here interchangeably). We denote by $G(f)$ the set of species of sequences of f. We denote the evolutionary distance between sequences of species

[1] In this work, we consider at most one protein sequence per gene.

i and j in family f by $d_f(i,j)$.

Assumption 1 (interspecies distance, family-specific rates)**.** We assume that, in the absence of LGT, all distances between orthologs of species i and j are proportional to an *interspecies distance* $d(i,j)$, with a family-specific proportionality constant τ_f. Formally, $d_f(i,j) = \tau_f \cdot d(i,j)$. Furthermore, we require that on average, the proportionality constant be one ($\frac{1}{|F|}\sum_{f\in F}\tau_f = 1$). This model is referred to as *proportional branch lengths* by Pupko et al. (2002).

Estimator $\hat{d}_f(i,j)$

The evolutionary distance $d_f(i,j)$ can be estimated from a pairwise alignment by maximum likelihood (ML) under a model of amino-acid substitution. We call this estimator $\hat{d}_f(i,j)$. The ML estimator is asymptotically unbiased and asymptotically normally distributed. The ML procedure also provides an estimate of its variance $\sigma^2(\hat{d}_f(i,j))$. Furthermore, we have seen in Chapter 4 how to estimate the covariances.

Estimator $\hat{d}(i,j)$

We estimate the interspecies distance $d(i,j)$ using the unweighted sample average over all $|F|$ families of orthologs:

$$\hat{d}(i,j) = \frac{1}{|F|}\sum_{f\in F}\hat{d}_f(i,j)$$

The estimator is unbiased, because:

$$\mathbb{E}(\hat{d}(i,j)) = \frac{1}{|F|}\sum_{f\in F}\mathbb{E}(\hat{d}_f(i,j)) = \frac{1}{|F|}\sum_{f\in F}\tau_f\cdot d(i,j) = d(i,j)\underbrace{\frac{1}{|F|}\sum_{f\in F}\tau_f}_{=1} = d(i,j)$$

Assumption 2. In the following, we will consider $\hat{d}(i,j)$ to be a point estimate, that is, we assume that $\sigma^2(\hat{d}(i,j)) = 0$.

This assumption may appear to be quite strong, especially if the number of families under consideration is small. In most cases, however, the number of families is relatively large (larger than the size of a typical family), and the variances of interspecies distances are much smaller than those of the other estimators under consideration here. In terms of computation, the assumption considerably reduces the time complexity of our approach.

Estimator $\hat{\tau}_f$

We estimate the rate τ_f of family f using the following estimator:

$$\hat{\tau}_f = \frac{\frac{1}{n_f(n_f-1)} \sum_{i,j \in G(f), i \neq j} \hat{d}_f(i,j)}{\frac{1}{n_f(n_f-1)} \sum_{i,j \in G(f), i \neq j} \hat{d}(i,j)} = \frac{\sum_{i,j \in G(f), i \neq j} \hat{d}_f(i,j)}{\sum_{i,j \in G(f), i \neq j} \hat{d}(i,j)}$$

where $n_f = |G(f)|$. Due to assumption 2, the denominator is constant, and thus $\hat{\tau}_f$ follows a normal distribution with variance

$$\sigma^2(\hat{\tau}_f) = \frac{\sum_{i,j,k,l \in f, i \neq j, k \neq l} cov(\hat{d}_f(i,j), \hat{d}_f(k,l))}{(\sum_{i,j \in f, i \neq j} \hat{d}(i,j))^2}$$

Lateral gene transfer

Definition (lateral gene transfer)**.** In the present work, a lateral gene transfer (LGT) event is the transfer of a gene from a donor species d (or an ancestor thereof) to a recipient species r (or an ancestor thereof).

Assumption 3. Since the divergence of d and r, at most one LGT event per family of orthologs took place between the two lineages.

Assumption 4. The rate of evolution (the branch length on the phylogenetic tree) of a sequence after LGT is homogeneous among all donor and recipient lineages.

Definition (δ)**.** Given a LGT event in family f between lineages of d and r, the evolutionary distance between the transfered sequence and the current sequences in r or d is expressed by δ (Fig. 7.1). The distance since LGT is the same for both species due to assumption 4.

Consequently, the expected distance between sequences in f of d and r is 2δ. For instance, if $\delta = 0$, the two proteins have not diverged since the LGT event, and thus the LGT is very recent.

7.2.2 Algorithm

DLIGHT identifies LGT events by considering, in all families of orthologs, all potential pairs of donor and recipient species. For each configuration, a likelihood ratio test is performed between the hypothesis of a LGT (alternative hypothesis) and the hypothesis of no LGT (null hypothesis). Formally, the set of significant LGT events is given by:

$$LGT = \left\{ (f,d,r) \,\middle|\, f \in F; d, r \in G(f); 2\ln \frac{l(f,d,r,\delta_{ML})}{l(f,d,r,\delta=\infty)} > \chi^2(\alpha,1) \right\}$$

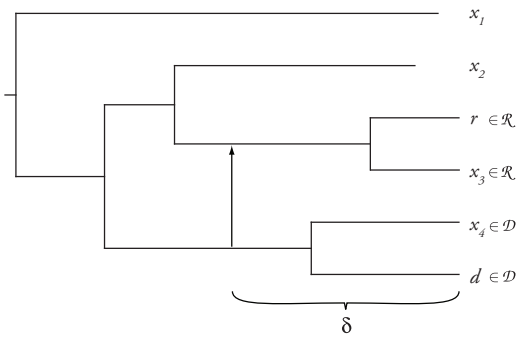


Figure 7.1: Distance to LGT event as captured by the parameter δ. The LGT event is represented by the arrow.

where F is the set of all families of orthologs, d a potential donor species, r a potential recipient species, and $l(f,d,r,\delta)$ is the likelihood of an LGT in f from lineages of d and r at distance δ in the past. δ_{ML} is the value that maximizes this likelihood under the alternative hypothesis. $l(f,d,r,\delta=\infty)$ is the likelihood under the null hypothesis (in which δ is fixed to ∞, see below), and $\chi^2(\alpha,1)$ is the critical value of the chi-square distribution with significance level α and one degree of freedom. This test is known as the likelihood ratio test (see e.g. Felsenstein 2004a). The ratio follows a chi-square distribution if the two models are nested, which is the case here, as we shall see below.

Below, we show how the likelihood of a LGT event $l(f,d,r,\delta)$ can be computed. The process can be split in three parts: first, given (f,d,r,δ), we infer which species of $G(f)$ belong to the set of donor species $\mathscr{D}$ and of recipient species $\mathscr{R}$. From these sets, we show how to compute the expected values of all $2|f|-3$ evolutionary distances of pairs in f that involve r and/or d, as well as their variances and covariances. Finally, we compute the likelihood of the event, which is based on the deviation of the observed distances from the modeled distances.

Step 1 – Assignment of species to sets of donors ($\mathscr{D}$) and recipients ($\mathscr{R}$)

First, given a quartet (f,d,r,δ), we infer members of $G(f)$ belonging to the donor and recipient lineages, that is, the set of species that directly descend from the donor (set $\mathscr{D}$) and recipient species (set $\mathscr{R}$). These subsets of $G(f)$ can be defined as follows:

$$\mathscr{D} = \{j \in G(f) \mid \tau_f \cdot d(j,d) \leq 2\delta\}$$

$$\mathscr{R} = \{j \in G(f) \mid \tau_f \cdot d(j,r) \leq 2\delta\}$$

We shall now justify these definitions (illustrated in Fig. 7.1). First, note that as could be reasonably expected, $d \in \mathscr{D}, r \in \mathscr{R}$, because in both cases the distance to themselves is 0, and δ being a distance is non-negative. As for the other species

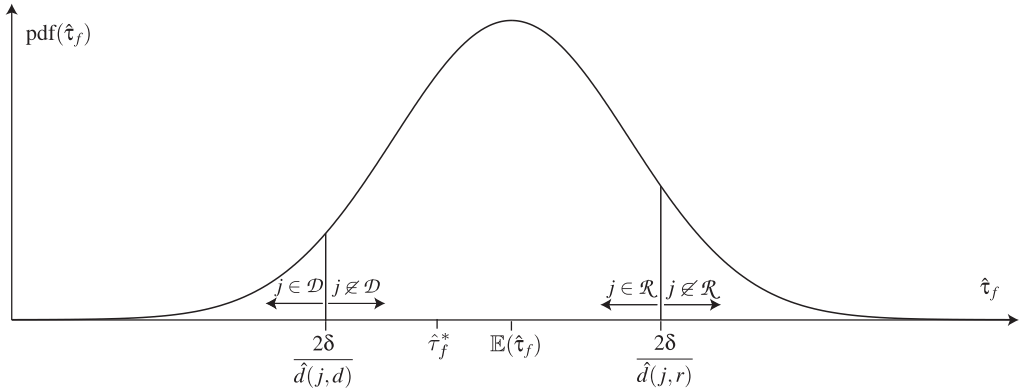


Figure 7.2: The assignment of sequence j to sets $\mathcal{D},\mathcal{R}$ depends on $\hat{\tau}_f$. For instance, at the point $\hat{\tau}_f^*$, j is in $\mathcal{R}$, but not in $\mathcal{D}$.

of $G(f)$, the definitions use assumption 4 (we focus on the definition of $\mathcal{D}$; the rational for $\mathcal{R}$ is similar): if all sequences from the donor lineage in f evolve at the same rate, they will all be δ away from the LGT. Further, by definition, members of the donor lineage have speciated after the LGT event, and therefore, their sequences in f are separated by a distance of at most 2δ.

To build these sets, we must rely on the estimators $\hat{\tau}_f$ and $\hat{d}(j,d)$ (or $\hat{d}(j,r)$ in the case of $\mathcal{R}$). Since the interspecies distances are point estimates (assumption 2), we only need to consider the distribution of $\hat{\tau}_f$ (see Sect. 7.2.1): the sets of donors and recipients differ depending on the value of the estimator $\hat{\tau}_f$. Fig. 7.2 depicts the distribution with the critical values of $\hat{\tau}_f$ for the assignment of a species j to $\mathcal{D}$ and $\mathcal{R}$.

Thus, if we consider the two critical values for all species j in $G(f)$, the distribution of $\hat{\tau}_f$ will be partitioned into $2|f|+1$ ranges. Each of these ranges map to particular $\mathcal{D}_i$ and $\mathcal{R}_i$, whose probability is the area of the density function pdf($\hat{\tau}_f$) in that particular range. We refer to the probability of the ith range as p_i. We will compute for each of these sets of donors and recipients the corresponding likelihood, and then average them according to their probability. The next step is therefore repeated for all $2|f|+1$ possible assignments of $\mathcal{D},\mathcal{R}$.

Step 2 – Pairwise distance statistics

Given a sextet $(f,d,r,\delta,\mathcal{D}_i,\mathcal{R}_i)$, the computation of the likelihood of a particular LGT event is based on the $2|f|-3$ pairwise distances in f that involve d or r. These distances are of interest because they are particularly altered by the LGT event, but the procedure could trivially be extended to all $\binom{|f|}{2}$ pairs in f.

The *observed* distances are simply the ML estimators for the relevant pairs of

sequences of f. Estimators for the *modeled* distances are provided in Table 7.1. Most distances involving the donor species d are unaffected by the LGT event, i.e. they are expected to follow the interspecies distances scaled by the family rate. Distances to the recipient species r however are mostly expected to follow the scaled interspecies distance to the donor d, because the sequence originated from the donor lineage, and after the LGT event, they evolved at the same rate as in the donor lineage (assumption 4). The special cases are: (i) distances between two recipients: they are unaffected by the LGT because the transfer happened before they speciated; (ii) distances between recipient and donor species: they are expected to be 2δ per definition; (iii) distances involving *inconsistent* species: the estimators and parameters can be such that a species is in both $\mathscr{D}_i$ and $\mathscr{R}_i$, for instance if δ is particularly large. In those cases, we treat the distance the same way as under the null hypothesis (no LGT transfer) and assign it an expected value that corresponds to the scaled interspecies distance. In terms of the model, this also has the advantage that the null hypothesis of no LGT is equivalent to the special case of a LGT with parameter $\delta = \infty$. This means that the models are nested, and therefore that the likelihood ratio follows a chi-square distribution with number of degree of freedom given by the difference in free parameters (one in our case).

label	$\in \mathscr{D}_i$	$\in \mathscr{R}_i$	$M(\hat{d}_f(j,d))$	$M(\hat{d}_f(j,r))$
outgroup	no	no	$\hat{\tau}_f \cdot \hat{d}(j,d)$	$\hat{\tau}_f \cdot \hat{d}(j,d)$
donor	yes	no	$\hat{\tau}_f \cdot \hat{d}(j,d)$	2δ
recipient	no	yes	2δ	$\hat{\tau}_f \cdot \hat{d}(j,r)$
inconsistent	yes	yes	$\hat{\tau}_f \cdot \hat{d}(j,d)$	$\hat{\tau}_f \cdot \hat{d}(j,r)$

Table 7.1: LGT event: modeled distances to r and d in f. Note that the last row (*inconsistent*) can occur in our model if δ is large; the adverse impact of such inherently inconsistent case is limited by using the same modeled distances as under the null hypothesis (no LGT event).

Note that in our model, both observed and modeled pairwise distances are normally distributed random variables, which can be expressed using two $2|f|-3$ dimensional vectors $\boldsymbol{x}$ and $\boldsymbol{y}$. In both cases, we have estimators for their variance-covariance matrices $\Sigma_{\boldsymbol{x}}$ and $\Sigma_{\boldsymbol{y}}$: for observed distances, the diagonal entries can be obtained by ML theory, and the covariances can be computed as described in Susko (2003). As for the modeled distances, the variance is either that of $\hat{\tau}_f$ scaled appropriately, or else null when $\hat{\tau}_f$ does not appear in the expression. The modeled distances do not covary, and thus all off-diagonal entries are null.

Let $\boldsymbol{z} = \boldsymbol{x} - \boldsymbol{y}$. The vector $\boldsymbol{z}$ is normally distributed, with expected value $\mathbb{E}(\boldsymbol{z}) = \boldsymbol{0}$. If we now assume that $\boldsymbol{x}, \boldsymbol{y}$ are independent, $\Sigma_{\boldsymbol{z}} = \Sigma_{\boldsymbol{x}} + \Sigma_{\boldsymbol{y}}$. In reality, they are not strictly independent, because $\boldsymbol{x}$ is a component (albeit a minor one)

of $\hat{\tau}_f$, which itself is used in the computation of $\boldsymbol{y}$.

Step 3 – Computation of the likelihoods, and estimation of δ

The likelihood of the LGT event $(f,d,r,\delta,\mathscr{D}_i,\mathscr{R}_i)$ can be computed from the multivariate normal probability density function of the vector z and covariance matrix Σ_z from the previous section:

$$l(f,d,r,\delta,\mathscr{D}_i,\mathscr{R}_i)=\frac{exp(-\frac{1}{2}z^T\Sigma_z^{-1}z)}{\sqrt{(2\pi)^{2|f|-3}|\Sigma_z|}}$$

We can now marginalize over the $2|f|+1$ different sets of donors and recipients (see step 1) to compute the likelihood of the LGT event (f,d,r,δ):

$$l(f,d,r,\delta)=\sum_i p_i \cdot l(f,d,r,\delta,\mathscr{D}_i,\mathscr{R}_i)$$

Furthermore, the parameter δ can be estimated by maximizing the likelihood. As mentioned above, the likelihood for the null hypothesis of no LGT event is obtained by the special case with parameter $\delta=\infty$.

7.2.3 Model Violations and Test of Multivariate Normality

DLIGHT is based on assumptions that do not always hold, in particular when dealing with biological sequences whose evolution strongly deviates from the Markovian model. To limit the adverse effect of such model violations, we test the multivariate normality of the data by computing a P-value based on the squared Mahalanobis distance $z^T\Sigma_z^{-1}z$, which is known to be chi-square distributed if z is multivariate normal. Data falling in extreme quantiles are considered dubious. In experiments reported here, predictions with data falling in the $(1-10^{-10})$ quantile were considered artifacts due to model violation, and were disregarded.

Furthermore, in case of poorly estimated variances or covariances, the matrix Σ_z may not be positive definite, or it may be singular if the sequences of two species are identical. In our implementation, we still try to identify LGT events by working with a subset of the family that constitute a well-posed problem (the problematic sequences are excluded on the basis of a simple greedy approach).

7.2.4 Combination of Results and Correction for Multiple Testing

As we presented above, DLIGHT computes a likelihood ratio test in all families of orthologs, for all different possible pairs of potential donor and recipient species. This raises the issues of combining results and correcting for multiple

testing. Currently, we take the conservative approach of combining results that are consistent, for instance when a LGT event happened before speciation of the recipient species into two species g_1 and g_2: the algorithm may detect a transfer when run with both species as recipient, but if in both cases the estimated δ suggests a transfer prior to their speciation, the prediction is consistent and can be combined. Another common case for combination are pairs of results that report LGT between consistent sets of donor and recipient genomes, but with reverse direction. The direction of some LGT events, such as transfers between close species, is inherently difficult to assess. Nevertheless, if one direction has a significantly higher probability, and provided that the estimated parameter δ is consistent, the direction of the LGT can be inferred.

We address the issue of multiple testing by using the Bonferroni adjustment, a common approach that discounts the significance by a factor corresponding to the total number of tests. If the tests are not independent from each other, which is the case here, the correction is excessive and some sensitivity is wasted.

7.3 Validation and Results

DLIGHT was tested using four different approaches: simulation, artificial LGT events, real biological data and comparison with previous results from the literature. The results of simulation are also reported for three simple LGT detection methods that serve as benchmark: methods based on GC-content, best-hits, and perturbed-distances. They are described in the *Appendix* A3.

7.3.1 *In Silico* Evolution Scenarios

Although a simulation will never fully capture the complexity and diversity of natural evolutionary processes, it allows the evaluation of algorithms with knowledge of the history of events, and therefore constitutes a traceable baseline. Synthetic genomes were generated using the software *ALF* (manuscript in preparation). ALF starts from a single organism and simulates the following evolutionary mechanisms: codon mutation based on empirical substitution probabilities (Schneider et al., 2005), with biased genome-specific GC contents and gene-specific mutation rates, codon indel, gene duplication, gene loss, LGT (both orthologous replacement and novel gene acquisition), and speciation. The probabilities of LGTs, gene duplications and gene loss were set to a proportion of 1:2:3, thereby keeping the expected number of genes constant (as suggested in Kunin and Ouzounis 2003). The two types of LGT events, novel gene acquisition and orthologous replacement, were set to have an equal probability of occurrence. Table 7.2 details the remaining parameters of the two different evolutionary scenarios investigated here. Genes from the resulting genomes were grouped in orthologous families using the OMA algorithm (Chapter 6).

Name	# of species	Avg. # of genes	Avg. genome distance (expect. identity)	# LGT	# of families
simulation 1	9	197	16 PAM (85.4%)	50	241
simulation 2	9	202	74 PAM (50.7%)	42	295

Table 7.2: Overview of the simulation parameters. In *simulation 1* closely related organisms are used while in *simulation 2* more distantly related organisms are analyzed.

The different algorithms were run on the two datasets and the performances were analyzed in terms of both sensitivity and specificity, at three levels of precision: first, the ability to report families of orthologs that contain at least one laterally transfered gene; second, the ability to identify the protein involved in a LGT event, that is, either report a *donor* or a *recipient* species; and third, the ability to correctly identify the direction of the LGT, in addition to the species involved. The six resulting ROC curves are presented in Fig. 7.3. Overall, DLIGHT showed significantly higher sensitivity and specificity than the other methods. It also performed more consistently than the other methods, with curves of similar shape across all experiments. The significance threshold is rather conservative (a consequence of the stringent Bonferroni correction) and led to 100% specificity in most cases. In the case in which the direction of LGTs was required, in distantly related species, the GC content and the perturbed distance approach outperformed DLIGHT. This may be due to the difficulties in estimating distances and variances when organisms are so far apart. In those cases, simpler methods may prove to be more robust.

7.3.2 Artificial LGT Events in Real Data

LGT events between real biological genomes can be simulated by introducing a gene from one species into another, either as substitute for its ortholog ("orthologous replacement") or as additional sequence. Such *artificially introduced* LGT event allows the testing of the algorithm on real biological data while having a positive control. However, only the specific case of very recently introduced genes can be simulated. Furthermore, real occurrences of LGTs may already be present in the dataset and their signals may conflict with the artificially introduced ones.

The biological data consisted of 15 archaea with 2273 gene families, of which 727 families had at least 6 genes. 200 cases of LGT events from random donors to random recipients were introduced, as orthologous replacement, in families with at least 6 genes. Fig. 7.4 presents the results of the tests. The 200 top scoring predictions were compared to the set of artificially introduced LGTs. Of all four methods, our performed best. Given the relatively good results obtained with the perturbed-distance approach in the previous test, its performance here is

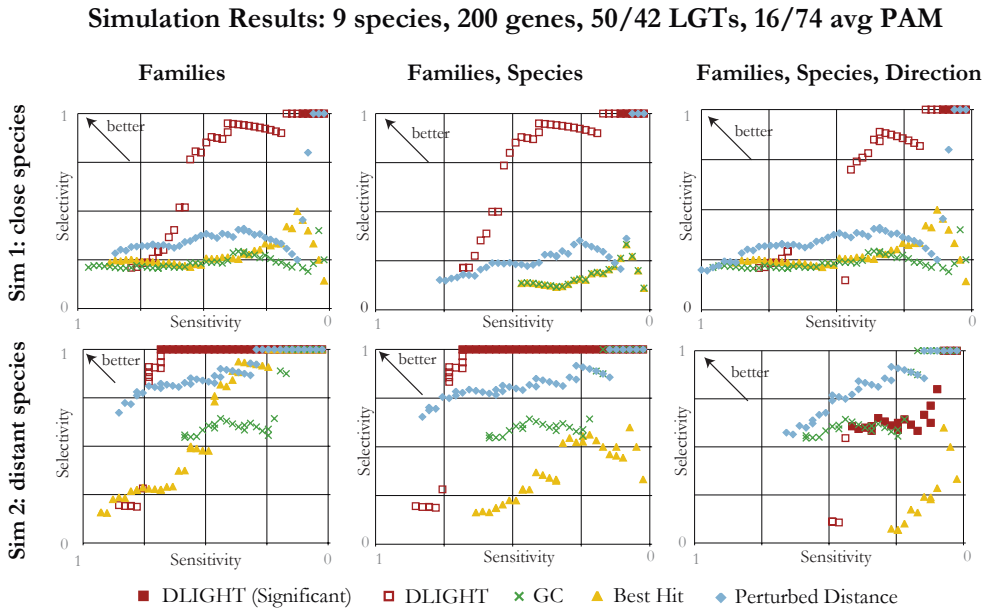


Figure 7.3: ROC analyses. Sensitivity is plotted along the X axis, specificity along the Y axis. Plots on the first line were obtained from a simulation with closer species, plots on the second line from more distantly related ones. The left column shows results of identifying families with LGT events. The middle column shows results of identifying families with LGT events and the involved species. The right column shows results of identifying families with LGT events, the involved species and the direction of the transfers.

surprisingly poor, with only 7 artificial LGTs recovered. Note also that being recent, transfers introduced here constitute ideal conditions for both the GC method (the composition has not had time to adapt to the new host) and the best-hit approach (transfer after all speciations).

7.3.3 Real Biological Data

LGT events are believed to happen throughout the prokaryotes, but not uniformly so. Some organisms are considered to be little affected by LGT while others are thought to have acquired many genes from distant species. Endosymbionts and endoparasites are micro-organisms that spend most of their life inside a host cell. As a consequence, for an LGT event to happen, foreign DNA would need to cross the membrane and defensive system of both the organism and its host. Therefore, such organisms are expected to have very few genes acquired through LGT compared to free living micro-organisms (Lawrence and Hendrickson, 2003).

Our algorithm was verified against these observations by comparing predic-

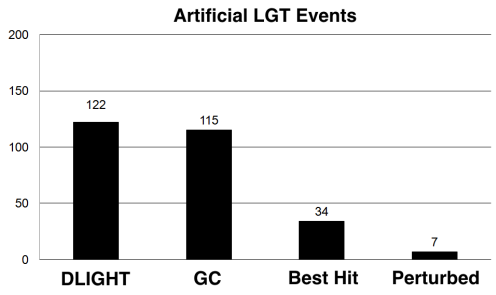


Figure 7.4: Artificially introduced LGT. The number of such LGTs among the top 200 predictions is given.

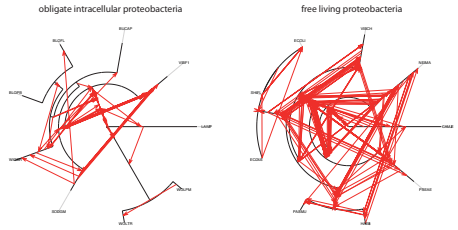


Figure 7.5: LGT flow among proteobacteria. LGTs are drawn with arrows indicating the direction of the transfer. DLIGHT was run with the same parameters on both datasets individually.

tions of two different datasets. We inferred LGTs for 9 endosymbionts[2] and for 9 free living pathogenic proteobacteria[3]. The organisms were classified according to HAMAP (Boeckmann et al., 2003).

DLIGHT detected between 1 and 22 foreign genes (6.3 in average) in endosymbionts, and between 2 and 70 genes (40.7 in average) in free living bacteria. Normalized with the genome sizes, this gives between 0.15% and 0.89% percent of foreign genes in endosymbionts, versus 0.12% to 2.43% in free living. Thus, endosymbionts appear indeed to have lower LGT rates than their free living counterparts. In figure 7.5 the LGT events are indicated in both trees as thin lines and there too, the difference in LGT occurrences is clearly visible.

The detected percentages of foreign genes is much lower than the values of 2% to 60% found in previous reports (Lerat et al., 2005; Ge et al., 2005; Dagan and Martin, 2007). However, these higher numbers represent all genes received by any organism outside the vertical genealogy, while our data reflect only gene transfer among 9 bacteria.

A larger set with 15 archaea[4] consisting of 2273 orthologous families was analyzed in a similar way. The average LGTs per gene was at 1.07%, with 292 detected LGT events in all 15 archaea. The number of acquired genes varies

[2]Candidatus Blochmannia floridanus, Blochmannia pennsylvanicus (strain BPEN), Buchnera aphidicola (subsp. Schizaphis graminum), Lawsonia intracellularis (strain PHE/MN1-00), Sodalis glossinidius (strain morsitans), Vibrio fischeri (strain ATCC 700601 / ES114), Wigglesworthia glossinidia brevipalpis, Wolbachia pipientis wMel, Wolbachia sp. (subsp. Brugia malayi) (strain TRS)

[3]Campylobacter jejuni, Escherichia coli O6, Escherichia coli, Haemophilus influenzae (strain 86-028NP), Neisseria meningitidis serogroup A, Pasteurella multocida, Pseudomonas aeruginosa, Shigella flexneri, Vibrio cholerae

[4]Methanocaldococcus jannaschii, Methanosarcina mazei, Pyrobaculum aerophilum, Sulfolobus solfataricus, Methanosarcina acetivorans, Aeropyrum pernix, Archaeoglobus fulgidus, Halobacterium salinarium, Methanobacterium thermoautotrophicum, Methanopyrus kandleri, Pyrococcus horikoshii, Thermoplasma volcanium, Nanoarchaeum equitans, Thermoplasma acidophilum, Methanococcus maripaludis

from 1 for *Nanoarchaeum equitans* to 37 for *Methanosarcina mazei* . Looking at the relative gene uptake with regard to the genome size, *Nanoarchaeum equitans* still recieved the fewest genes with 0.19%. *Thermoplasma volcanium* received the most genes with 2.4%. It has been proposed previously that LGT is common between Thermoplasmatales and Sulfolobales (Philippe and Douady, 2003). In our dataset, *Thermoplasma volcanium* exchanged 14 genes with *Sulfolobus solfataricus* and *Thermoplasma acidophilum* also 14 genes with *Sulfolobus solfataricus*. This is significantly more than the 3.6 average LGTs between archaea.

In addition to these tests, DLIGHT was applied to a dataset of 10 mammals[5]. Although LGT between higher eukaryotes and bacteria are found by some authors, we are not aware of any case of LGT between two mammals. Mammals serve therefore as negative control for our LGT detection method. Indeed, DLIGHT did not detect any LGT among the 10 mammals.

7.3.4 Comparison with Previous Results

Results from different LGT inference approaches can be very inconsistent, with overlaps at times smaller than expected by random (Ragan, 2001). This is particularily true when comparing the results of parametric and phylogenetic methods. Thus, the results of DLIGHT were compared with two studies based on phylogenetic approaches.

Comparison with Zhaxybayeva *et al.* (2006)

In Zhaxybayeva et al. (2006), the authors used an embedded quartet decomposition analysis to search events of LGT in 11 completely sequenced cyanobacteria. Orthologs were grouped via reciprocal top-scoring blast hits, resulting in families with few paralogs. A set of 1128 orthologous genes was found to be present in at least nine of the 11 cyanobacterial genomes and taken as input for the LGT search. Within the group of cyanobacteria, 135 LGTs were detected, mostly between *Gloeobacter violaceus* and *Synechococcus elongatus* (45) and *Prochlorococcus marinus* SS120 and *Prochlorococcus marinus* (strain MIT 9313) (28).

We tried to confirm the predictions of LGT in these 135 families using DLIGHT. In 54 families (40%), significant LGTs were reported. In 32 of them, the species predicted to be involved were either the same, or in agreement with the trees constructed by Zhaxybayeva et al. (2006). The 22 other predictions were conflicting with their trees. Additionally, it should be noted that the interspecies distances estimated by DLIGHT were computed on the basis of these 135 families, none of which is congruent to the species tree according to Zhaxybayeva et al. (2006); this suggests that DLIGHT is relatively robust with respect

[5]Homo sapiens, Mus musculus, Canis familiaris, Rattus norvegicus, Bos taurus, Pan troglodytes, Monodelphis domestica, Macaca mulatta, Loxodonta africana, Oryctolagus cuniculus

to perturbations in the data.

Comparison with Beiko *et al.* (2005)

DLIGHT was compared with results from Beiko et al. (2005), a large scale LGT inference study using an explicit phylogenetic method. For 22,437 families of proteins in 144 genomes, they constructed gene trees and compared in each tree all bifurcations to a reference species tree. They reported bifurcations with significant posterior probability (PP), classified in either consistent or conflicting with the species tree.

A subset of their 8,315 protein families of size up to 15 sequences was randomly selected. Based on their bifurcation analysis, these families were partitioned in four categories: *i*. 28.5% families with strong support of no LGT (all bifurcations consistent with species tree with $PP \geq 0.95$), *ii*. 38.4% families with mild support of no LGT (no conflicting bifurcation with $PP \geq 0.5$), *iii*. 15.2% families with mild support of LGT (at least one conflicting bifurcation with $PP \geq 0.5$, none with $PP \geq 0.95$), and *iv*. 17.8% families with strong support of LGT ($PP \geq 0.95$).

DLIGHT was run on this dataset, with, as sole input, the protein sequences labeled with family and species identifiers. The computation of all pairwise evolutionary distances within families required about 2 days on a single AMD Opteron 1.8 GHz. DLIGHT used another day to predict significant LGT events, which were found in 634 families. The distribution of inferred LGT events among the four categories defined from their predictions was as follows: *i*. 7.1%, *ii*. 13.1%, *iii*. 19.2%, and *iv*. 60.6%. As almost 80% of the predictions are the same, the level of agreement between the two methods is quite high, especially considering the large differences in methodologies.

7.4 Conclusion

In this chapter, we introduce a new implicit phylogenetic method for LGT detection, based on pairwise evolutionary distances in a probabilistic framework. Validation shows that it compares favorably with existing parametric and implicit phylogenetic methods. Furthermore, its advantages over explicit phylogenetic methods include speed and lack of reliance on multiple sequence alignments and gene tree inference.

There are, though, a number of aspects that could be the object of further improvement: the sensitivity could be increased by the computation of the likelihoods using all pairwise distances within gene families, and not only the distances to the transfered genes; confidence intervals in the estimation of the interspecies distances. Instead of the approximation of multivariate normality, and at expense of increased time complexity, the distribution of the distances could

possibly be estimated in an MCMC framework.

8 Conclusions

In the first part of this thesis, we reviewed and extended methods to estimate and handle evolutionary distances between pairs of sequences: we compared pairwise and multiple sequences alignment approaches, and observed that MSAs are faster and more accurate than pairwise alignments, but this does not result in a significant improvement in evolutionary distance estimation. We presented an estimator for the variance of the difference of two distances from sequences aligned pairwise, and illustrated its usefulness in improving the detection of asymmetric evolution, and in identifying the closest relative of a given sequence in a group of homologs. Our final contribution in part one was an estimator for the covariance of pairwise distances. We showed that it performs similarly to the ML variance estimator. In particular, it shows no sign of bias when sequence divergence is below 150 PAM units (i.e. above $\sim 29\%$ expected sequence identity). Above that distance, the covariances tend to be underestimated, but then ML variances are also underestimated.

The second part applies the tools developed in the first part to large-scale comparative genomics studies. The first application as an algorithm based on pairwise distances for the detection of non-orthologs that arise by mistake in current orthology classification methods based on genome-specific best hits, and in particular in the COGs database. Our results show that a very significant fraction of the COG groups include non-orthologs: using conservative parameters, the algorithm detects non-orthology in a third of all COG groups. This results motivated the development of our own orthology inference effort, OMA, the second application of part two. Besides the verification of non-orthology, OMA has several other distinctive features: it computes evolutionary distances from pairwise maximum likelihood alignments, it uses confidence intervals to

account for statistical inference uncertainty and to allow for many-to-many orthology, and clusters groups of orthologs using an edge-weighted clique algorithm. We validated our results and compare them with those of 11 other projects and methods. The study tested orthologs predictions on the basis of phylogeny (through reconstruction of orthologous gene trees and through comparison with phylogenetic analyses from the literature) and on the basis of function conservation (in terms of GO annotation, EC number classification, expression level, and gene neighborhood conservation). The results of OMA are among the best in the phylogenetic tests. In functional tests, it also performs well where high functional specificity is required, at the expense of a lower recall than projects such as OrthoMCL or EggNog. In terms of size, OMA is by a wide margin the largest orthology inference effort; as of July 2008, we have performed all-against-all alignments of 633 complete genomes and have built orthologous matrices for the major clades of genomes. The matrices are regularly updated as new species are included, the largest now including 550 genomes. The final application of part two was an algorithm to detect lateral gene transfer based on pairwise distances. At its core, it computes a likelihood ratio between hypotheses of LGT and no LGT. As opposed to explicit phylogenetic LGT detection approaches, it avoids the high computational cost and pitfalls associated with multiple sequence alignments and gene tree inference, while maintaining the high level of characterization (species involved in LGT, direction, distance to the LGT event in the past) associated with such methods. The results and validation of the algorithm, both through simulated and real data, showed that the method outperforms common LGT detection approaches.

At the end of the day (or the thesis), a method should be judged by its capacity to answer specific biological questions. Certainly, pairwise distance approaches are not a panacea; in particular, when computationally affordable, methods that optimize simultaneously over all sequences can be more powerful. But when a large amount of data is at hand, we hope to have shown in this thesis that pairwise distance approaches can be a practical, and indeed competitive, alternative.

Appendix

A1 Complexity of the analytical solution of k-states model for triplets

To motivate the numerical approximation form chapter 4.2, we show here that the analytical solution of the ML estimator for the distances of a triplet is very complex, even for a simplified model of mutation. The k-state model (Cannarozzi and Gonnet, 2005) is an idealized situation where each position has k possible states and the transition probabilities are all identical and only depend on the time t. For $k = 4$ this is equivalent to the Jukes-Cantor model (Jukes and Cantor, 1969). Whatever is the initial state, the probability of a mutation after time t is given by

$$p(t) = \frac{k-1}{k}\left(1 - r^t\right)$$

where r is

$$r = 1 - \frac{k}{100(k-1)}$$

so that t is measured in PAM units. (Measuring in PAM units is proportional to any other measure, and it means that at $t = 1$ one percent of the characters are changed, i.e. $p(1) = 1/100$.) and that all transitions are equally likely, and only depend on the PAM distance. Under this model, the log–likelihood can be expressed in terms of the counts of matches/mismatches of the triplet (X, Y, Z), i.e. N_{xxx} is the number of positions where all the characters are identical, N_{xxz} is the number of positions where X and Y coincide but Z differs, etc.

$$l(A \mid t) = N_{xxx}\log(P_{xxx}) + N_{xxz}\log(P_{xxz}) + N_{xyx}\log(P_{xyx}) +$$

$$N_{xyy}\log(P_{xyy}) + N_{xyz}\log(P_{xyz})$$

$$P_{xxx} = (1-p_x)\left(1-p_y\right)(1-p_z) + \frac{p_x p_y p_z}{(k-1)^2}$$

$$P_{xxz} = (1-p_x)\left(1-p_y\right)p_z + \frac{p_x p_y (1-p_z)}{k-1} + \frac{(k-2)\,p_x p_y p_z}{(k-1)^2}$$

$$P_{xyx} = (1-p_x)\,p_y\,(1-p_z) + \frac{p_x\left(1-p_y\right)p_z}{k-1} + \frac{(k-2)\,p_x p_y p_z}{(k-1)^2}$$

$$P_{xyy} = p_x\left(1-p_y\right)(1-p_z) + \frac{(1-p_x)\,p_y p_z}{k-1} + \frac{(k-2)\,p_x p_y p_z}{(k-1)^2}$$

$$P_{xyz} = \frac{k-2}{k-1}\left((1-p_x)\,p_y p_z + p_x\left(1-p_y\right)p_z + p_x p_y\,(1-p_z) + \frac{(k-3)\,p_x p_y p_z}{k-1}\right)$$

where p_x is the probability of mutating from the origin to X and similarly for p_y and p_z. Taking partial derivatives of the likelihood with respect to p_x, p_y and p_z gives a system of 3 rational polynomial equations (all the logarithms disappear) in 3 unknowns and 6 parameters. Such a system of equations has a solution that will be an algebraic function of the parameters (a root of a polynomial, where the coefficients of the polynomial involve the parameters). Despite its simple appearance, this system of equations is beyond the capabilities of current computer algebra systems to resolve. And this is not a complete surprise, as the algebraic numbers/functions involved are at least of degree 23. The special case where two of the branches have the same length, has been solved exactly in (Chor et al., 2006), they find that their solution is an algebraic function of degree 11. This unfortunately is not applicable as we are interested in the cases where the branches away from the origin are of different lengths.

We have computed the exact solution for concrete values of the parameters, in particular $N_{xxx} = 10, N_{xxz} = 5, N_{xyx} = 4, N_{xyy} = 3, N_{xyz} = 2, k = 3$ using Maple and the value of p_x is a root of the irreducible polynomial

$$-6582435840000 + 189590785228800\,z - 2438333515038720\,z^2 + \ldots$$

$$\ldots + 10304020514917800\,z^{21} - 1635488137841976\,z^{22} + 99990709180560\,z^{23}$$

This means that the general solution will be an algebraic function of degree 23 or higher, it cannot be lower. If an instantiation of the polynomial with values gives this irreducible polynomial, then the general polynomial must be irreducible of degree 23 or higher (some terms could have simplified in the instantiation). This makes the usefulness of an exact solution inexistent, it is more difficult to solve the polynomial and select the right root than to maximize the likelihood and/or solve the system of equations by numerical methods.

A2 Comparison of Orthology Projects: Materials and Methods

A2.1 Input data

All the projects included in this study are publicly available. A short description of the chosen configurations and references are given in the following. We used the default parameters unless mentioned otherwise.

RoundUp: RoundUp can be downloaded from https://rodeo.med.harvard.edu/tools/roundup/. It is available with different parameter settings to tune for the desired sensitivity. In this comparison we included the strictest parameter set (also default settings), i.e. Blast E-value cutoff 10^{-20} and divergence cutoff 0.2.

Inparanoid: Inparanoid is available from http://inparanoid.sbc.su.se. We used the release 6.0 from Aug 2007 including 35 species.

Ensembl Compara: The orthology predictions from Ensembl were obtained from the Compara database version 47, which is available from http://oct2007.archive.ensembl.org/.

COG,KOG: Cluster of Orthologous Groups and its eukaryotic equivalent are available from http://www.ncbi.nlm.nih.gov/COG/. We used the versions from Mar 2003 and Jul 2003 respectively.

OrthoMCL: We obtained the version from Sep 2006 of OrthoMCL from http://orthomcl.cbil.upenn.edu/.

Homologene: Homologene is available from the NCBI webpage www.ncbi.nlm.nih.gov/HomoloGene/. For this comparison, we used built 58 from Nov 2007.

EggNog: EggNog is available from http://eggnog.embl.de/. We used the data from Oct 2007 including 373 species.

OMA: OMA is available in various formats on http://www.omabrowser.org. We used the the data from Nov 2007 including 550 species. OMA infer orthology at the level of pairs of sequences ("OMA Pairwise"), from which it also computes groups of orthologs ("OMA Group"). Both type of predictions are included in the comparisons.

BBH: The typical Bidirectional Best Hit implementation uses Blast for aligning the protein sequences. We used the more accurate algorithm from Smith and Waterman (1981b) for the alignment with the same scoring threshold as used by the OMA algorithm for the all-against-all step.

RSD: Reciprocal Smallest Distance orthology relations are computed using ML distance estimates from pairwise alignments having significant alignment scores (Dayhoff score > 217, the cut-off used by OMA as well)

A2.2 Phylogenetic Reconstruction Test

A consequence of Fitch's definition is that trees of orthologs are congruent to the species tree (i.e. the topology, or branching order, is the same). The phylogenetic reconstruction test uses this property to test the predicted orthologs. It uses three reference species trees (see *Supplementary Materials*) whose branching order is well-accepted, and whose topology follows a "comb" structure, that is, completely unbalanced. Each leaf consists of one or several species. The phylogeny of species that share the same leaf is not necessarily well resolved, but this fact is irrelevant here, because, as we shall see below, the test includes at most one sequence per leaf in each tree reconstructed. Including more than one species per leaf is merely a way to include more data in the test.

In each trial, a starting sequence from a random species in the innermost leaf is randomly chosen. Then, for each project under scrutiny, we try to build a set of sequences consisting of one ortholog per leaf. If a project predicts more than one sequence orthologous to the starting sequence in a leaf, one of them is picked randomly. If a project predicts no ortholog in a particular leaf, sequence from that leaf are excluded from other projects as well, such that the resulting sets of sequences are of the same size for all projects. If the orthologous groups have less than 5 sequences, the procedure restarts with another starting sequence. Else, based on each orthologous set, we build a tree (as described below) and assess its agreement with the reference species tree by computing the percentage of correct split derived from the Robinson-Foulds metric (Robinson and Foulds, 1981).

The "comb" structure of the topology is necessary to ensure that a set of sequences orthologous to a starting sequence indeed constitutes an orthologous groups (that is, a set of sequences in which every pair is orthologous): recall that two sequences are orthologs if they split through speciation. Thus, if all bifurcations in the gene trees are speciation events, the set of sequences constitute an orthologous group. Due to the particular topology, each bifurcation is the split of the innermost sequence from another sequence. Since the innermost sequence is orthologous to all other sequences, all bifurcations are speciation events, and the conclusion follows.

Darwin Least-Squares distance trees

The sequences are aligned pairwise using Smith and Waterman (1981b), with joint ML estimation of all pairwise distances using the *Align* function of Darwin (Gonnet et al., 2000). The estimated distance and variances are used to compute a least-squares distance tree using Darwin's *LeastSquaresTree* function.

Muscle and RaxML

As a second method for computing the gene tree, we use Muscle (Edgar, 2004) as multiple sequence alignment tool in combination with RaxML-VI-HPC version 2.2.3 (Stamatakis, 2006) as tree building package. RaxML builds ML trees. Muscle was run with default parameters, while RaxML was run with *JTT* with 4 gamma categories as amino acid substitution model. The method is repeated from ten random start topologies. The tree with the highest likelihood is taken as the resulting tree of this method.

A2.3 Benchmarks from literature

We used four different sources of manually curated orthology reference sets from the literature: (1) A reconciled tree of Pfam adenosine/AMP deaminase family (PF00962) produced by Engelhardt et al. (2006, 2005). This tree contains 251 proteins from which we could map 146. (2) Results from detailed phylogenetic analysis on three different COGs presented in Dessimoz et al. (2006a). From the originally 116 proteins, 82 were mappable, again restricting on identical sequences. (3) Resulting trees from the phylogenetic analysis by Hughes (1998) of 10 gene families. 33 of 165 proteins could be mapped. (4) The ortholog reference set proposed by Hulsen et al. (2006). From there 102 of the 167 proteins could be mapped.

For every of those difficult phylogenies, we extracted the orthologous and paralogous relations. For the purpose of this study, those assignments are considered to be error free and are taken as a reference set. For every possible protein pair where both proteins are present in the common set of sequences, we determined whether the project made a true positive, a true negative, a false-positive or a false-negative prediction. Those measurements are then used to infer the true positive and the false-positive rate respectively by taking a Bayesian approach with a uniform prior. Finally, the results of the performance on the four phylogenies have been averaged.

A2.4 Functional based definition

Gene Ontology

GO terms and their evidence codes are obtained from EBI and Ensembl for all available species. 255,806 proteins had at least one annotation. Since most annotations are automatically obtained from sequence similarity and all the orthology projects base their predictions on sequence similarity, we only keep the annotations inferred experimentally (Evidence codes *EXP,IDA,IEP,IGI,IMP,IPI*). We end up with 26,676 proteins having 78,912 annotations in total. The similarity between two annotated proteins i and j having GO terms c_i and c_j is

computed as proposed by Lin (1998)

$$sim(c_i, c_j) = \frac{2lnP_{ms}(c_i, c_j)}{lnP(c_i) + lnP(c_j)},$$

where $P(c)$ is the probability of encountering the term c and

$$P_{ms}(c_i, c_j) = \min_{c \in S(c_i, c_j)} P(c)$$

is the probability of the minimum subsummer (or most specific parent) between term c_i and c_j. The similarity score obviously varies between 0 (unrelated) and 1 (identical terms). The occurrence probability of GO term c is estimated from the occurrence frequency of GO term c or a child term of c for any instance of a protein intersection set independently.

Proteins are often annotated with multiple GO terms. In such situations, the similarities need to be combined. We follow the rationale of Lord et al. (2003) and average all the possible similarity values between putative orthologs i and j, since in general a protein has all the attributed roles. Thus the overall similarity between proteins i and j each having its set of GO terms GO_i and GO_j is

$$\overline{sim_{i,j}} = \frac{1}{|GO_i||GO_j|} \sum_{c_k \in GO_i} \sum_{c_l \in GO_j} sim(c_k, c_l).$$

The mean similarity of a project given a (intersection) set of proteins that we show in figure 6.12a) is the mean similarity between all the putative orthologs stated by the project in the given set of proteins.

Enzyme Classification

The Swiss Institute of Bioinformatics operates a database on Enzyme nomenclature (Bairoch, 2000). In this study we use the release from Nov. 13 2007 of the database. As a first step, we remove all the proteins that are assigned to more than one EC number (3.83%). Then, the proteins from the EC database are mapped to OMA (61,518 proteins or 71.16%). For those proteins, we computed the ratio of putative orthologs that map to the same EC class.

Correlation in Expression Profiles

MAS 5.0 processed tissue expression data from human and mouse Affymetric microarray chips (human:U133A/GNF1H; mouse:GNF1M) and the gene mappings as used by Liao and Zhang (2006) have been provided by the authors. A total of 25,854 probe sets could be mapped to 16,295 proteins in the human genome and 17,872 probe sets to 15,522 mouse proteins. As a measure for the

accuracy of the orthology predictions, we computed the average Pearson correlation coefficient of the relative abundance level RA between the putative human and mouse orthologs with respect to the projects' common sequences sets. The relative abundance level of gene i and tissue t is defined as the relative expression signal intensity in tissue t, thus

$$RA(i,t) = \frac{S(i,t)}{\sum_t S(i,t)},$$

and the correlation between two putative orthologs i and j having n tissues in common

$$\rho_{i,j} = \frac{n\sum_t RA(i,t)RA(j,t) - \sum_t RA(i,t)\sum_t RA(j,t)}{\sqrt{n\sum_t RA(i,t)^2 - \left(\sum_t RA(i,t)\right)^2}\sqrt{n\sum_t RA(j,t)^2 - \left(\sum_t RA(j,t)\right)^2}}$$

Gene Neighborhood Conservation

The conservation of gene order is measured in the following way. We use the coding sequence features (CDS) from OMA's genome sources (mainly Ensembl, Genome Reviews and EMBL) to determine the order of the genes in the genome. Overlapping genes are excluded, as the order is not resolved. For every predicted orthologous protein pair, we check whether their directly adjacent neighbors (if present) are orthologous too. The verification is performed using the union of all predictions. This ensures that projects with many ortholog predictions are not advantaged over more stringent ones. Whenever we find at least one of the four possible neighbor configurations in the union, we conclude that the neighborhood is conserved.

Formally, the average conservation is

$$\overline{X} = \frac{1}{|orth|} \sum_{\substack{(g_1,g_2)\in orth \\ N(g_1)\neq\emptyset, N(g_2)\neq\emptyset}} min\left(1, \sum_{\substack{n_1\in N(g_1) \\ n_2\in N(g_2)}} \begin{cases} 1, & \text{if } (n_1,n_2)\in \cup_{orth} \\ 0, & \text{else} \end{cases}\right)$$

where $N(g)$ are the neighbors of gene g in the projects' common set of proteins, $orth$ is the set of orthologous pairs and $\cup_{orth}$ the union of the ortholog predictions.

A3 Alternative LGT scoring functions

In the following, we give the mathematical formulation of the three common approaches to identify LGT events which were used in the validation part of chapter 7. All three consist of a scoring function to rank all genes as potentially laterally transfered candidates.

A3.1 GC Content

The GC method used here is a basic implementation of this common parametric approach. A more advanced implementation can be found in Lawrence and Ochman (1997). The version used here considers the GC content on the first and third codon position, without performing a codon usage analysis. The score for a gene x in a species X is computed as follows:

$$S_{GC}(x) = \frac{(GC(x,1) - \mu_{GC}(X,1))^2}{\sigma^2_{GC}(X,1)} + \frac{(GC(x,3) - \mu_{GC}(X,3))^2}{\sigma^2_{GC}(X,3)}$$

where $GC(x,i)$ is the average GC content of the gene x at its ith codon position, and $\mu_{GC}(X,i), \sigma^2_{GC}(X,i)$ the average and variance of GC content among all ith codon position of genes in species X.

A3.2 Best Hit Approach

The best hit method is a variant of a common approach based on blast that infers LGT when a sequence has its highest scoring match is in a distant species (Clarke et al., 2002). Our implementation improves this idea by considering the shortest evolutionary distance rather than the top similarity score. More precisely, the score of a gene x from a species X and family of orthologs f is computed as follows:

$$S_{BH}(x) = \frac{Rank_f(T)}{|f|}$$

where T is the organism in which x has its closest homolog, $Rank_f(T)$ the rank of T among the species represented in f ordered by increasing average interspecies distance to X.

A3.3 Perturbed-Distances Approach

The third method detects LGT using the same underlying idea as our algorithm – the discrepancy between gene and interspecies pairwise distances that results from an LGT event – but in a much cruder way: the score of a gene x from an species X, in family f is

$$S_{PD}(x) = \frac{1}{|f|-1} \sum_{y \in f, y \neq x} (\Delta(x,y) - \Delta(X,Y))$$

where $d_(x,y)$ denotes the evolutionary distance between genes x and y, $\Delta(X,Y)$ the interspecies distance between X and Y.

Bibliography

Alexeyenko, A., I. Tamas, G. Liu, and E. L. L. Sonnhammer. 2006. Automatic Clustering of Orthologs an Inparalogs Shared by Multiple Genomes. Bioinformatics **22**:e9–e15.

Alm, E. J., K. H. Huang, M. N. Price, R. P. Koche, K. Keller, I. L. Dubchak, and A. P. Arkin. 2005. The MicrobesOnline Web site for comparative genomics. Genome Res **15**:1015–22.

Altenhoff, A. M., and C. Dessimoz. 2009. Phylogenetic and Functional Assessment of Orthologs Inference Projects and Methods. PLoS Computational Biology **5**:e1000262.

Altschul, S. F., W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. 1990. Basic Local Alignment Search Tool. Journal of Molecular Biology **215**:403–410.

Azevedo, C., A. Sadanandom, K. Kitagawa, A. Freialdenhoven, K. Shirasu, and P. Schulze-Lefert. 2002. The RAR1 interactor SGT1, an essential component of R gene-triggered disease resistance. Science **295**:2073–2076.

Bairoch, A. 2000. The ENZYME database in 2000. Nucleic Acids Res. **28**:304–305.

Balasubramanian, R., M. R. Fellows, and V. Raman. 1998. An improved fixed-parameter algorithm for vertex cover. Inf. Process. Lett. **65**:163–168.

Bandyopadhyay, S., R. Sharan, and T. Ideker. 2006. Systematic identification of functional orthologs based on protein network comparison. Genome Res. **16**:428–435.

Bateman, A., L. Coin, R. Durbin, et al. 2004. The Pfam Protein Families Database. Nucleic Acids Res **32**:D138–41.

Beiko, R. G., T. J. Harlow, and M. A. Ragan. 2005. Highways of gene sharing in prokaryotes. Proc Natl Acad Sci U S A **102**:14332–14337. Comparative Study.

Benner, S. A., M. A. Cohen, and G. H. Gonnet. 1993. Empirical and Structural Models for Insertions and Deletions in the Divergent Evolution of Proteins. J Mol Biol **229**:1065–1082.

Benson, D. A., I. Karsch-Mizrachi, D. J. Lipman, J. Ostell, and D. L. Wheeler. 2005. GenBank. Nucleic Acids Res. **33 Database Issue**:34–38.

Berglund, A., E. Sjölund, G. Ostlund, and E. Sonnhammer. 2008. InParanoid 6: eukaryotic ortholog clusters with inparalogs. Nucleic Acids Res **36**:D263–D266.

Blackshields, G., I. M. Wallace, M. Larkin, and D. G. Higgins. 2006. Analysis and comparison of benchmarks for multiple sequence alignment. In Silico Biol **6**:321–339.

Blanc, G., A. Barakat, R. Guyot, R. Cooke, and M. Delseny. 2000. Extensive Duplication and Reshuffling in the Arabidopsis Genome. Plant Cell **12**:1093–1102.

Boeckmann, B., A. Bairoch, R. Apweiler, et al. 2003. The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. Nucleic Acids Res **31**:365–370.

Bulmer, M. 1991. Use of the method of generalized least-squares in reconstructing phylogenies from sequence data. Mol Biol Evol **8**:868–883.

Cannarozzi, G. M., and G. H. Gonnet. 2005. Idealized Mutational Clocks. Technical report, Informatik, ETH, Zurich.

Carillo, H., and D. Lipman. 1988. The Multiple Sequence Alignment Problem in Biology. SIAM J. Appl. Math. **48**:1073–1082.

Carpousis, A. J. 2002. The Escherichia coli RNA degradosome: structure, function and relationship in other ribonucleolytic multienzyme complexes. Biochem Soc Trans **30**:150–155.

Cavalli-Sforza, L. L., and A. W. F. Edwards. 1967. Phylogenetic analysis: models and estimation procedures. Evolution **21**:550–570.

Charollais, J., D. Pflieger, J. Vinh, M. Dreyfus, and I. Iost. 2003. The DEAD-box RNA helicase SrmB is involved in the assembly of 50S ribosomal subunits in Escherichia coli. Mol Microbiol **48**:1253–1265.

Chen, F., A. J. Mackey, J. K. Vermunt, and D. S. Roos. 2007. Assessing performance of orthology detection strategies applied to eukaryotic genomes. PLoS ONE **2**:e383.

Chen, Z. 2003. Assessing Sequence Comparison Methods with the Average Precision Criterion. Bioinformatics **19**:2456–2460.

Chenna, R., H. Sugawara, T. Koike, R. Lopez, T. J. Gibson, D. G. Higgins, and J. D. Thompson. 2003. Multiple sequence alignment with the Clustal series of programs. Nucleic Acids Res. **31**:3497–3500.

Chor, B., M. D. Hendy, and S. Snir. 2006. Maximum Likelihood Jukes-Cantor Triplets: Analytic Solutions. Mol Biol Evol **23**:626–632.

Chotia, C., and A. Lesk. 1986. The relation between the Divergence of Sequence and Structure in Proteins. EMBO J. **5**:823–826.

Clarke, G. D. P., R. G. Beiko, M. A. Ragan, and R. L. Charlebois. 2002. Inferring genome trees by using a filter to eliminate phylogenetically discordant sequences and a distance matrix based on mean normalized BLASTP scores. J Bacteriol **184**:2072–2080.

Conant, G. C., and A. Wagner. 2003. Asymmetric Sequence Divergence of Duplicate Genes. Genome Res. **13**:2052–2058.

Dagan, T., and W. Martin. 2007. Ancestral genome sizes specify the minimum rate of lateral gene transfer during prokaryote evolution. Proc Natl Acad Sci U S A **104**:870–875.

Dayhoff, M. O., R. M. Schwartz, and B. C. Orcutt. 1978. A model for evolutionary change in proteins. In: Dayhoff, M. O., editor, Atlas of Protein Sequence and Structure, volume 5. National Biomedical Research Foundation, 345–352.

DeLuca, T. F., I.-H. Wu, J. Pu, T. Monaghan, L. Peshkin, S. Singh, and D. P. Wall. 2006. Roundup: a multi-genome repository of orthologs and evolutionary distances. Bioinformatics **22**:2044–2046.

Dermitzakis, E. T., and A. G. Clark. 2001. Differential Selection After Duplication in Mammalian Developmental Genes. Mol Biol Evol **18**:557–562.

Dessimoz, C., B. Boeckmann, A. Roth, and G. H. Gonnet. 2006a. Detecting Non-Orthology in the COG Database and Other Approaches Grouping Orthologs Using Genome-Specific Best Hits. Nucleic Acids Res **34**:3309–3316.

Dessimoz, C., G. Cannarozzi, M. Gil, D. Margadant, A. Roth, A. Schneider, and G. Gonnet. 2005. OMA, A Comprehensive, Automated Project for the Identification of Orthologs from Complete Genome Data: Introduction and First Achievements. In: McLysath, A., and D. H. Huson, editors, RECOMB 2005 Workshop on Comparative Genomics, volume LNBI 3678 of *Lecture Notes in Bioinformatics*. Springer-Verlag, 61 – 72.

Dessimoz, C., and M. Gil. 2008. Covariance of maximum likelihood evolutionary distances between sequences aligned pairwise. BMC Evol. Biol. **8**.

Dessimoz, C., M. Gil, A. Schneider, and G. H. Gonnet. 2006b. Fast estimation of the difference between two PAM/JTT evolutionary distances in triplets of homologous sequences. BMC Bioinformatics **7**.

Dessimoz, C., D. Margadant, and G. H. Gonnet. 2008. DLIGHT - Lateral Gene Transfer Detection Using Pairwise Evolutionary Distances in a Statistical Framework. In: RECOMB 08: Research in Computational Molecular Biology, 12th Annual International Conference, Singapore, 2008, Proceedings, volume 4955 of *Lecture Notes in Computer Science*. Springer, 315–330.

Diges, C. M., and O. C. Uhlenbeck. 2005. Escherichia coli DbpA is a 3' → 5' RNA helicase. Biochemistry **44**:7903–7911.

Doolittle, R. F. 1994. Convergent evolution: the need to be explicit. Trends Biochem Sci **19**:15–18.

Dost, B., T. Shlomi, N. G. 0002, E. Ruppin, V. Bafna, and R. Sharan. 2007. QNet: A Tool for Querying Protein Interaction Networks. In: Speed, T. P., and H. Huang, editors, RECOMB, volume 4453 of *Lecture Notes in Computer Science*. Springer, 1–15.

Edgar, R. C. 2004. MUSCLE: a multiple sequence alignment method with reduced time and space complexity. BMC Bioinformatics **5**:113.

Edgar, R. C., and S. Batzoglou. 2006. Multiple sequence alignment. Curr Opin Struct Biol **16**:368–373.

Efron, B., and R. J. Tibshirani. 1993. An introduction to the bootstrap. Chapman & Hall, New York.

Elliott, C., F. Zhou, W. Spielmeyer, R. Panstruga, and P. Schulze-Lefert. 2002. Functional conservation of wheat and rice Mlo orthologs in defense modulation to the powdery mildew fungus. Mol. Plant Microbe Interact. **15**:1069–1077.

Engelhardt, B. E., M. I. Jordan, and S. E. Brenner. 2006. A Graphical Model for Predicting Protein Molecular Function. In: Cohen, W. W., and A. Moore, editors, ICML 2006: Proceedings of the 23th International Conference on Machine Learning. ACM, 297 – 304.

Engelhardt, B. E., M. I. Jordan, K. E. Muratore, and S. E. Brenner. 2005. Protein molecular function prediction by Bayesian phylogenomics. PLOS Computational Biology **1**:432 – 445.

Felsenstein, J. 1981. Evolutionary trees from gene frequencies and quantitative characters: finding maximum likelihood estimates. Evolution **35**:1229–1242.

———. 1993. PHYLIP (Phylogeny Inference Package) version 3.5c. Distributed by the author.

———. 2004a. Inferring Phylogenies. Sinauer Associates, Inc., Sunderland, MA.

———. 2004b. PHYLIP (Phylogeny Inference Package) version 3.6. Distributed by the author.

Fitch, W. 2000. Homology a personal view on some of the problems. Trends Genet. **16**:227–231.

Fitch, W. M. 1970. Distinguishing Homologous from Analogous Proteins. Syst Zool **19**:99–113.

Fleischmann, R., M. Adams, O. White, R. Clayton, E. Kirkness, A. Kerlavage, C. Bult, J. Tomb, B. Dougherty, and J. Merrick. 1995. Whole-genome random sequencing and assembly of Haemophilus influenzae Rd. Science **269**:496–512.

Fujibuchi, W., H. Ogata, H. Matsuda, and M. Kanehisa. 2000. Automatic detection of conserved gene clusters in multiple genomes by graph comparison and P-quasi grouping. Nucleic Acids Res. **28**:4029–4036.

Fulton, D., Y. Li, M. Laird, B. Horsman, F. Roche, and F. Brinkman. 2006. Improving the Specificity of High-throughput Ortholog Prediction. BMC Bioinformatics **28**:270.

Gascuel, O. 1997. BIONJ: an improved version of the NJ algorithm based on a simple model of sequence data. Mol Biol Evol **14**:685–695.

Gattiker, A., K. Michoud, C. Rivoire, et al. 2003. Automated annotation of microbial proteomes in SWISS-PROT. Comput Biol Chem **27**:49–58.

Ge, F., L.-S. Wang, and J. Kim. 2005. The cobweb of life revealed by genome-scale estimates of horizontal gene transfer. PLoS Biol **3**:e316.

Gherardini, P. F., M. N. Wass, M. Helmer-Citterich, and M. J. E. Sternberg. 2007. Convergent evolution of enzyme active sites is not a rare phenomenon. J Mol Biol **372**:817–845.

Gil, M., C. Dessimoz, and G. H. Gonnet. 2005. A Dimensionless Fit Measure for Phylogenetic Distance Trees. J Bioinform Comput Biol **3**:1429–1440.

Goldman, N., and Z. Yang. 1994. A Codon-based Model of Nucleotide Substitution for Protein-coding DNA Sequences. Mol. Biol. Evol. **11**:725–736.

Gonnet, G. H. 1994. A Tutorial Introduction to Computational Biochemistry Using Darwin. Technical report, Informatik, ETH Zurich, Switzerland.

Gonnet, G. H., M. A. Cohen, and S. A. Benner. 1992. Exhaustive matching of the entire protein sequence database. Science **256**:1443–1445.

Gonnet, G. H., M. T. Hallett, C. Korostensky, and L. Bernardin. 2000. Darwin v. 2.0: An Interpreted Computer Language for the Biosciences. Bioinformatics **16**:101–103.

Goodman, M., J. Czelusniak, G. W. Moore, and A. E. Romero-Herrara. 1979. Fitting the gene lineage into its species lineage: a parsimony strategy illustrated by cladograms constructed from globin sequences. Syst Zool **28**:132–168.

Gophna, U., E. Z. Ron, and D. Graur. 2003. Bacterial type III secretion systems are ancient and evolved by multiple horizontal-transfer events. Gene **312**:151–163.

Gough, J. 2005. Convergent evolution of domain architectures (is rare). Bioinformatics **21**:1464–1471.

Graham, R. L., and L. R. Foulds. 1982. Unlikelihood that minimal phylogenies for a realistic biological study can be constructed in reasonable computational time. Mathematical Biosciences **60**:133–142.

Hamady, M., M. D. Betterton, and R. Knight. 2006. Using the nucleotide substitution rate matrix to detect horizontal gene transfer. BMC Bioinformatics **7**:476.

Harris, M. A., J. Clark, A. Ireland, et al. 2004. The Gene Ontology (GO) database and informatics resource. Nucleic Acids Res **32**:258–261.

Hasegawa, M., H. Kishino, and T. Yano. 1985. Dating of the human-ape splitting by a molecular clock of mitochondrial DNA. J. Mol. Evol. **22**:160–174.

Hubbard, T., D. Andrews, M. Caccamo, et al. 2005. Ensembl 2005. Nucl Acids Res **33**:D447–D453.

Hubbard, T., D. Barker, E. Birney, et al. 2002. The Ensembl genome database project. Nucl Acids Res **30**:38–41.

Hubbard, T. J. P., B. L. Aken, K. Beal, et al. 2007. Ensembl 2007. Nucl. Acids Res. **35**:D610–617.

Huelsenbeck, J. P., D. M. Hillis, and N. Rasmus. 1996. A likelihood-ratio test of monophyly. Syst. Biol. **45**:546–558.

Hughes, A. L. 1998. Phylogenetic tests of the hypothesis of block duplication of homologous genes on human chromosomes 6, 9, and 1. Mol. Biol. Evol. **15**:854 – 870.

Hulsen, T., M. A. Huynen, J. de Vlieg, and P. M. Groenen. 2006. Benchmarking ortholog identification methods using functional genomics data. Genome Biol **7**:R31.

Iliopoulos, I., S. Tsoka, M. Andrade, P. Janssen, B. Audit, A. Tramontano, A. Valencia, C. Leroy, C. Sander, and C. Ouzounis. 2001. Genome sequences and great expectations. Genome Biol. **2**:INTERACTIONS0001.

Jensen, L. J., P. Julien, M. Kuhn, C. von Mering, J. Muller, T. Doerks, and P. Bork. 2008. eggNOG: automated construction and annotation of orthologous groups of genes. Nucl. Acids Res. **36**:D250––D254.

Jensen, R. A. 2001. Orthologs and paralogs - we need to get it right. Genome Biol **2**:INTERACTIONS1002.

Jones, D. T., W. R. Taylor, and J. M. Thornton. 1992. The Rapid Generation of Mutation Data Matrices from Protein Sequences. Comput. Applic. Biosci. **8**:275–282.

Jones, P. G., M. Mitta, Y. Kim, W. Jiang, and M. Inouye. 1996. Cold shock induces a major ribosomal-associated protein that unwinds double-stranded RNA in Escherichia coli. Proc Natl Acad Sci U S A **93**:76–80.

Jukes, T., and C. Cantor. 1969. Evolution of protein molecules. In: Munro, H., editor, Mammalian protein metabolism III. New York: Academic Press, 21–132.

Kanehisa, M., S. Goto, S. Kawashima, Y. Okuno, and M. Hattori. 2004. The KEGG resource for deciphering the genome. Nucleic Acids Res. **32**:277–280.

Karlin, S. 1998. Global dinucleotide signatures and analysis of genomic heterogeneity. Curr Opin Microbiol **1**:598–610.

Katoh, K., K. Kuma, H. Toh, and T. Miyata. 2005. MAFFT version 5: improvement in accuracy of multiple sequence alignment. Nucleic Acids Res **33**:511–518.

Kellis, M., B. W. Birren, and E. S. Lander. 2004. Proof and evolutionary analysis of ancient genome duplication in the yeast Saccharomyces cerevisiae. Nature **428**:617–624.

Koonin, E., L. Aravind, and A. Kondrashov. 2000. The impact of comparative genomics on our understanding of evolution. Cell **101**:573–576.

Koonin, E. V. 2005. Orthologs, paralogs, and evolutionary genomics. Annu Rev Genet **39**:309–38.

Koski, L. B., and G. B. Golding. 2001. The Closest BLAST Hit Is Often Not the Nearest Neighbor. J. Mol. Evol. **52**:540 – 542.

Kunin, V., and C. A. Ouzounis. 2003. The balance of driving forces during genome evolution in prokaryotes. Genome Res **13**:1589–1594.

Lawrence, J. G., and D. L. Hartl. 1992. Inference of horizontal genetic transfer from molecular data: an approach using the bootstrap. Genetics **131**:753–760.

Lawrence, J. G., and H. Hendrickson. 2003. Lateral gene transfer: when will adolescence end? Mol Microbiol **50**:739–749.

Lawrence, J. G., and H. Ochman. 1997. Amelioration of bacterial genomes: rates of change and exchange. J Mol Evol **44**:383–397.

———. 1998. Molecular archaeology of the Escherichia coli genome. Proc Natl Acad Sci U S A **95**:9413–9417.

———. 2002. Reconciling the many faces of lateral gene transfer. Trends Microbiol **10**:1–4. News.

Lee, Y., R. Sultana, G. Pertea, et al. 2002. Cross-referencing eukaryotic genomes: TIGR Orthologous Gene Alignments (TOGA). Genome Res **12**:493–502.

Lerat, E., V. Daubin, H. Ochman, and N. A. Moran. 2005. Evolutionary origins of genomic repertoires in bacteria. PLoS Biol **3**:e130.

Li, L., C. J. J. Stoeckert, and D. S. Roos. 2003. OrthoMCL: identification of ortholog groups for eukaryotic genomes. Genome Res **13**:2178–2189.

Li, Y.-J., and S. C.-M. Tsoi. 2002. Phylogenetic analysis of vertebrate lactate dehydrogenase (LDH) multigene families. J Mol Evol **54**:614–24.

Liao, B.-Y., and J. Zhang. 2006. Evolutionary Conservation of Expression Profiles Between Human and Mouse Orthologous Genes. Mol. Biol. Evol. **23**:530–540.

Lin, D. 1998. An information-theoretic definition of similarity. In: Proc. 15th International Conf. on Machine Learning. Morgan Kaufmann, San Francisco, CA, 296–304.

Lord, P. W., R. D. Stevens, A. Brass, and C. A. Goble. 2003. Investigating semantic similarity measures across the Gene Ontology: the relationship between sequence and annotation. Bioinformatics **19**:1275–1283.

Margulies, M., M. Egholm, W. Altman, et al. 2005. Genome sequencing in microfabricated high-density picolitre reactors. Nature **437**:376–380.

Markowitz, V. M., F. Korzeniewski, K. Palaniappan, et al. 2006. The integrated microbial genomes (IMG) system. Nucl Acids Res **34**:D334–8.

Marron, M., K. M. Swenson, and B. M. E. Moret. 2004. Genomic distances under deletions and insertions. Theor. Comput. Sci. **325**:347–360.

McClure, M., T. Vasi, and W. Fitch. 1994. Comparative analysis of multiple protein sequence alignment methods **11**:571 – 92.

Medigue, C., T. Rouxel, P. Vigier, A. Henaut, and A. Danchin. 1991. Evidence for horizontal gene transfer in Escherichia coli speciation. J Mol Biol **222**:851–856. Comparative Study.

Mewes, H. W., C. Amid, R. Arnold, et al. 2004. MIPS: analysis and annotation of proteins from whole genomes. Nucl. Acids Res. **32**:D41–44.

Moszer, I., E. P. Rocha, and A. Danchin. 1999. Codon usage and lateral gene transfer in Bacillus subtilis. Curr Opin Microbiol **2**:524–528.

Mrazek, J., and S. Karlin. 1999. Detecting alien genes in bacterial genomes. Ann N Y Acad Sci **870**:314–329.

Muller, T., and M. Vingron. 2000. Modeling amino acid replacement. J Comput Biol **7**:761–776.

Needleman, S. B., and C. D. Wunsch. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. J. Mol. Biol. **48**:443–453.

Nei, M., J. C. Stephens, and N. Saitou. 1985. Methods for computing the standard errors of branching points in an evolutionary tree and their application to molecular data from humans and apes. Mol Biol Evol **2**:66–85.

Notebaart, R. A., M. A. Huynen, B. Teusink, R. J. Siezen, and B. Snel. 2005. Correlation between sequence conservation and the genomic context after gene duplication. Nucleic Acids Res **33**:6164–6171.

Notredame, C. 2007. Recent Evolutions of Multiple Sequence Alignment Algorithms. PLoS Comput Biol **3**:e123.

O'Brien, S., M. Menotti-Raymond, W. Murphy, W. Nash, J. Wienberg, R. Stanyon, N. Copeland, N. Jenkins, J. Womack, and J. Marshall Graves. 1999. The promise of comparative genomics in mammals. Science **286**:458–462.

Ogden, T. H., and M. S. Rosenberg. 2006. Multiple sequence alignment accuracy and phylogenetic inference. Syst Biol **55**:314–328.

Ohmori, H. 1994. Structural analysis of the rhlE gene of Escherichia coli. Jpn J Genet **69**:1–12.

Ohno, S. 1970. Evolution by Gene Duplication. Springer-Verlag, New York.

Ouzounis, C. 1999. Orthology: another terminology muddle. Trends in Genetics **15**:445.

Overbeek, R., M. Fonstein, M. D'Souza, G. D. Pusch, and N. Maltsev. 1999. The use of gene clusters to infer functional coupling. Proc. Natl. Acad. Sci. U.S.A. **96**:2896 – 2901.

Pal, C., B. Papp, and M. J. Lercher. 2005. Adaptive evolution of bacterial metabolic networks by horizontal gene transfer. Nat Genet **37**:1372–1375.

Pearson, W. R., and M. L. Sierk. 2005. The limits of protein sequence comparison? Curr Opin Struct Biol **15**:254–260.

Perriere, G., L. Duret, and M. Gouy. 2000. HOBACGEN: database system for comparative genomics in bacteria. Genome Res **10**:379–385.

Philippe, H., and C. J. Douady. 2003. Horizontal gene transfer and phylogenetics. Curr Opin Microbiol **6**:498–505.

Poirot, O., E. O'Toole, and C. Notredame. 2003. Tcoffee@igs: A web server for computing, evaluating and combining multiple sequence alignments. Nucleic Acids Res **31**:3503–3506.

Pupko, T., D. Huchon, Y. Cao, N. Okada, and M. Hasegawa. 2002. Combining multiple data sets in a likelihood analysis: which models are the best? Mol Biol Evol **19**:2294–2307.

Ragan, M. A. 2001. On surrogate methods for detecting lateral gene transfer. FEMS Microbiol Lett **201**:187–191.

Remm, M., C. Storm, and E. Sonnhammer. 2001. Automatic clustering of orthologs and in-paralogs from pairwise species comparisons. J Mol Biol **314**:1041–52.

Rice, J. A. 2001. Mathematical Statistics and Data Analysis. Duxbury Press.

Robinson, D. F., and L. R. Foulds. 1981. Comparison of Phylogenetic Trees. Mathematical Biosciences **53**:131 – 147.

Ronquist, F., and J. P. Huelsenbeck. 2003. MrBayes 3: Bayesian phylogenetic inference under mixed models. Bioinformatics **19**:1572–1574.

Rosenberg, M. 2005a. Evolutionary distance estimation and fidelity of pair wise sequence alignment. BMC Bioinformatics **6**:102.

———. 2005b. Multiple sequence alignment accuracy and evolutionary distance estimation. BMC Bioinformatics **6**:278.

Roth, A. C., G. H. Gonnet, and C. Dessimoz. 2008. The algorithm of OMA, large-scale orthology inference. BMC Bioinformatics **9**:518.

Schneider, A., G. M. Cannarozzi, and G. H. Gonnet. 2005. Empirical codon substitution matrix. BMC Bioinformatics **6**.

Schneider, A., C. Dessimoz, and G. H. Gonnet. 2007. OMA Browser – Exploring orthologous relations across 352 complete genomes. Bioinformatics **23**:2180–2182.

Seoighe, C., and K. Scheffler. 2005. Very Low Power to Detect Asymmetric Divergence of Duplicated Genes. In: McLysath, A., and D. H. Huson, editors, RECOMB 2005 Workshop on Comparative Genomics, volume LNBI 3678 of *Lecture Notes in Bioinformatics*. Springer-Verlag, 142 – 152.

Sierk, M. L., and W. R. Pearson. 2004. Sensitivity and selectivity in protein structure comparison. Protein Sci **13**:773–785.

Smith, T. F., and M. S. Waterman. 1981a. Identification of common molecular subsequences. J. Mol. Biol. **147**:195–197.

———. 1981b. Identification of Common Molecular Subsequences. Journal of Molecular Biology **147**:195–197.

Stamatakis, A. 2006. RAxML-VI-HPC: maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models. Bioinformatics **22**:2688–2690.

Stebbings, L. A., and K. Mizuguchi. 2004. HOMSTRAD: recent developments of the Homologous Protein Structure Alignment Database. Nucleic Acids Res **32**:D203–7.

Susko, E. 2003. Confidence regions and hypothesis tests for topologies using generalized least squares. Mol. Biol. Evol. **20**:862 – 868.

Swofford, D. L., G. L. Olsen, P. J. Waddell, and D. M. Hillis. 1996. Phylogenetic inference. Sunderland, Massachusetts: Sinauer Associates, 2nd edition, 407–514.

Tatusov, R. L., N. D. Fedorova, J. D. Jackson, et al. 2003. The COG database: an updated version includes eukaryotes. BMC Bioinformatics **4**:http://www.biomedcentral.com/1471–2105/4/41.

Tatusov, R. L., E. V. Koonin, and D. J. Lipman. 1997. A Genomic Perspective on Protein Families. Science **278**:631–7.

Thompson, J., P. Koehl, R. Ripp, and O. Poch. 2005. BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark. Proteins **61**:127–136.

Thompson, J., F. Plewniak, and O. Poch. 1999. A comprehensive comparison of multiple sequence alignment programs. Nucl. Acids Res. **27**:2682–2690.

Thorne, J., H. Kishino, and J. Felsenstein. 1991. An evolutionary model for maximum likelihood alignment of DNA sequences. J. Mol. Evol. **33**:114–124.

Van de Peer, Y., J. S. Taylor, I. Braasch, and A. Meyer. 2001. The Ghost of Selection Past: Rates of Evolution and Functional Divergence of Anciently Duplicated Genes. J Mol Evol **53**:436–446.

van der Heijden, R. T., B. Snel, V. van Noort, and M. A. Huynen. 2007. Orthology prediction at Scalable Resolution by Phylogenetic Tree analysis. BMC Bioinformatics **8**:83.

Wagner, A. 2002. Asymmetric Functional Divergence of Duplicate Genes in Yeast. Mol Biol Evol **19**:1760–1768.

Wall, D. P., H. B. Fraser, and A. E. Hirsh. 2003. Detecting putative orthologs. Bioinformatics **19**:1710 – 1711.

Wheeler, D. L., T. Barrett, D. A. Benson, et al. 2007. Database resources of the National Center for Biotechnology Information. Nucl. Acids Res. **35**:D5 – D12.

Yang, Z. 2006. Computational Molecular Evolution. Oxford University Press, London.

Zhaxybayeva, O., J. P. Gogarten, R. L. Charlebois, W. F. Doolittle, and R. T. Papke. 2006. Phylogenetic analyses of cyanobacterial genomes: quantification of horizontal gene transfer events. Genome Res **16**:1099–1108.

Index

2-oxoacid, 57
2-oxoglutarate, 58
4-aminobutyrate, 63

AA, 5, 9, 21, 26, 27, 93
acyltransferase, 57
adenosine/AMP deaminase, 113
Aeropyrum pernix, 102
algorithm, 7, 19, 42, 50, 51, 53–58, 63, 65, 67, 69, 70, 74–77, 84, 92, 94, 99–101, 107, 111, 116
alignment, *see* sequence alignment
all pairs, *see* AP
all-against-all, 24, 53, 56, 69, 108, 111
alphaproteobacteria, 58
amino-acid, *see* AA
anchor, 37, 38, 40, 44, 46
approximation, 21, 24, 26–30, 32, 33, 35, 36, 40, 42, 45, 74, 104, 109
archaea, 100, 102
Archaeoglobus fulgidus, 102
artificial LGT, 99, 100
assumption, 5, 26, 29, 30, 36, 39, 42, 51, 93, 94, 96–98
asymmetric divergence, 32
asymmetric evolution, 24, 29, 31, 32, 107
asymptotics, 10, 12, 26

Balibase, 7, 14
Bayesian analysis, 64
Bayesian statistics, 57
BBH, 50, 51, 71, 77, 80, 84, 86–88, 111
best bi-directional hits, *see* BBH
bias, 10, 12, 14, 36, 42, 44, 46, 99, 107
bifurcation, 104, 112
BIONJ, 55
bipartition, 79
Blast, 33, 66, 69, 71, 103, 111, 116
Bonferroni correction, 99, 100
bootstrap, 28, 30, 55, 57, 62
Bos taurus, 103
BP, 67, 73
branch, *see* tree, branch
broken pairs, *see* BP

Canis familiaris, 103
CDS, 115
Chapman-Kolmogorov, 6
character, 1, 2, 5, 7, 12, 17, 19, 22, 23, 37, 109
clan, 55, 57, 58, 62
clique, 37, 67, 74, 75, 108
cluster, 55, 63, 66, 68, 74, 78, 86, 108, 111
coding sequence features, *see* CDS
COG, 49–51, 53, 54, 56, 57, 62, 63, 65, 66, 77, 80, 84, 85, 90, 107, 111, 113
cold-shock, 58
confidence interval, 21, 24, 28, 33, 71, 74, 80–83, 104, 107
conservation, 49, 50, 77, 83–86, 108, 115
correlation coefficient, 115
covariance, 2, 21, 24, 26, 27, 36, 38, 40–42, 45, 53, 71, 93, 97, 98, 107
coverage, 87
curation, 50, 76, 77
cut-off, 35, 111
cyanobacteria, 103

Darwin, 8
database, 14, 49, 50, 53, 55–57, 63, 65, 66, 68, 69, 75–77, 80, 84, 87, 88, 90, 111, 114
dataset, 8, 100, 102–104
Dayhoff, 8, 27, 28, 111
DEAD-box, 58
deaminase, 113
degradosome, 58
dehydrogenase, 57
diagonal, 45, 97
dinucleotide, 91
divergence, 10, 44, 56, 72, 94, 107, 111
DLIGHT, 92, 98, 100, 102–104
domain, 53, 57, 69, 70
donors, 95, 96, 98, 100
duplication, 31, 33, 49–51, 54, 58, 66, 71–73, 76, 78, 85, 99
dynamic programming, 7, 68

E-value, 54, 56, 69, 111
EC, 57, 82–84, 108, 114
edge, 74, 75, 108
EggNog, 77, 80, 84, 85, 87, 111
EGO/TOGA, 65
empirical, 5, 21, 29, 36, 99
endoparasite, 101
endoribonuclease, 58
endosymbiont, 101, 102
Ensembl, 66, 78, 80, 82, 84, 87, 88, 111, 113, 115
enzyme, 82, 84, 114
enzyme classification, *see* EC
Escherichia coli, 57, 58
estimator, 7, 8, 10, 12, 14–17, 19–31, 35–38, 40–46, 69, 71, 81, 93, 94, 96–99, 103, 104, 107, 109, 112, 114
  unbiased, 25, 26
eukaryote, 1, 29, 65, 68, 77, 78, 80, 103, 111
evolutionary history, 41, 49, 76, 84, 99
exponential, 1, 22

false-negative, 54, 58, 66, 80, 86, 113
false-positive, 54, 55, 58, 66, 80, 82, 86, 113
Fasta, 90
fission, 53
frequency, 9, 12, 22, 23, 53, 91, 114
fusion, 51

GABA, 63
gammaproteobacteria, 57
GC-content, 99
Gene Ontology, *see* GO
generalized least-squares, *see* GLS
GLS, 21, 46
GO, 82–84, 108, 113, 114
GP, 67, 75
group pairs, *see* GP

Halobacterium salinarium, 102
HAMAP, 55, 102
helicase, 58
HGT, *see* LGT
HOBACGEN, 56
Homo sapiens, *see* human
Homologene, 77, 80, 84, 85, 87, 111
homology, 7–9, 20, 22, 24, 25, 33, 35, 37, 55, 67
horizontal gene transfer, *see* LGT
human, 50, 66, 76, 84, 86, 114

identity, 27, 100, 107
in-paralogs, 50, 51, 72, 78
indel, 9, 99
independence, 28–30, 36, 41–46
induced pairwise alignment, *see* IPA
Inparanoid, 65, 77, 78, 80, 82, 84, 86, 87, 111
insertion-deletion, *see* indel
intersection, 76, 79, 80, 82, 83, 114
interspecies distance, 93, 96, 97, 103, 116, 117
IPA, 7, 10, 17, 19

JTT, 27, 28, 55, 70, 113

Jukes-Cantor, 26, 109

KEGG, 65, 66
KOG, 65, 77, 80, 84–86, 111

L-asparagine, 63
large-scale analysis, 2, 19, 21, 24, 35, 51, 65, 77, 107
lateral gene transfer, *see* LGT
Lawsonia intracellularis, 102
leaf, *see* tree, leaf
least-squares method, 9, 40, 112
  generalized, *see* GLS
  ordinary, *see* OLS
  weighted, *see* WLS
LGT, 2, 21, 49, 51, 58, 91–104, 108, 116
lineage, 23, 57, 94, 95, 97
lipoamide acyltransferase, 57
Loxodonta africana, 103

Macaca mulatta, 103
Mafft, 9, 12, 14, 17
Mahalanobis distance, 98
mammal, 103
Maple, 110
mapping function, 56, 76, 78, 79, 85, 96, 113, 114
Markov process (or model), 2, 5, 6, 12, 21, 29, 38, 41, 69, 98
Markov-chain Monte Carlo, *see* MCMC
match, 7, 116
matrix, 5, 6, 8–10, 22, 23, 27, 28, 36, 40, 41, 45, 46, 55, 68, 70, 91, 97, 98, 108
maximization, 8, 69, 74, 110
maximum likelihood, *see* ML
MCMC, 105
metabolism, 54, 57
Methanobacterium thermoautotrophicum, 102
Methanocaldococcus jannaschii, 102
Methanococcus maripaludis, 102
Methanosarcina acetivorans, 102
Methanosarcina mazei, 102
microarray, 114
misalignment, 17, 44, 46
misclassification, 51
ML, 8, 10, 12, 14, 17, 19, 22, 24–26, 28, 29, 35, 38, 43–46, 53, 79–81, 93, 96, 97, 107, 109, 112
  distance estimation, 21–23, 25, 36, 37, 71, 86, 111
model, 1, 2, 5, 6, 9, 12, 19, 21–23, 26, 27, 29, 36, 38, 41, 42, 44, 46, 69, 86, 92, 93, 95, 97, 98, 109
Monodelphis domestica, 103
monophyly, 21
mouse, 84, 86, 114
MrBayes, 55
MSA, 7–10, 12, 14, 17, 19, 22, 23, 37, 40, 45, 79, 88, 107, 113
MultiParanoid, 65
multiple sequence alignment, *see* MSA
multivariate, 24, 86, 98, 104
Mus musculus, 103
Muscle, 9
mutation, 5, 27, 33, 39, 68, 99, 109, 110

N-terminal, 57
Nanoarchaeum equitans, 103
NCBI, 49, 111
Needleman–Wunsch, 8, 10, 12, 14, 16
neighborhood, 77, 82, 83, 85, 108, 115
Newton-Raphson, 22
non-anchor, 37, 38
non-transitivity, 49, 78
NP-complete, 1, 74
nucleotide, 5, 91

OMA, 55, 65–67, 71, 78–80, 82, 84, 87, 88, 90, 107, 111, 114, 115
  Browser, 67, 88, 90
OPA, 7, 19, 36–38, 41, 44, 46
optimal sequence alignment, *see* OPA
optimization, 19, 22, 108
ordinary least-squares, *see* OLS
organism, 1, 63, 91, 99–102, 116

orthology, 2, 9, 21, 29, 31, 44, 49–51, 53–58, 63, 65–68, 70–72, 74–76, 78–80, 82–88, 90, 92–95, 98–100, 102, 103, 107, 111–116
  co-orthologs, 74
  many-to-many, 66, 71, 72, 108
  one-to-many, 66, 72
  one-to-one, 66, 74
  pseudo-orthologs, 68, 77
OrthoMCL, 65, 77, 80, 82, 84–87, 108, 111
Oryctolagus cuniculus, 103
out-group, 31, 32, 72

P-value, 71, 98
pairwise distance, 2, 7, 8, 15–17, 19, 21, 25, 27, 35, 41, 63, 96, 97, 104, 107, 108, 112, 116
PAM, 6, 8, 9, 12, 22, 26, 27, 33, 34, 36, 40–42, 46, 53, 68, 70, 100, 107, 109
Pan troglodytes, 103
paralogy, 29, 49–51, 55, 58, 66, 67, 72, 73, 78, 92
  out-paralogy, 78
parametric, 5, 21, 26, 91, 103, 104, 116
Pareto frontier, 84
parsimony, 1
partition, 74, 75, 78
permease, 63
Perturbed-Distances, 116
Pfam, 69, 113
phenylalanine, 63
Phyletic Profile, 90
phylogeny, 1, 8, 17, 19, 20, 22, 33, 35, 45, 49, 51, 54, 55, 57, 58, 62, 63, 65–67, 72, 74, 76, 78–81, 85–87, 91, 92, 94, 103, 104, 108, 112, 113
point accepted mutation, *see* PAM
polynomial, 27, 28, 110
Prochlorococcus, 103
prokaryote, 50, 57, 68, 91, 101
proline, 63
proteobacteria, 102
pseudocode, 38
Pseudomonas aeruginosa, 58, 102
Pyrobaculum aerophilum, 102
Pyrococcus horikoshii, 102

quartet, 2, 36, 40–42, 44, 45, 54, 72, 73, 95, 103

random, 9, 14, 27, 33, 39, 40, 42, 97, 100, 103, 112, 113
range, 6, 28, 84, 85, 96
rank, 14, 92, 116
rate, 6, 33, 42, 46, 49, 51, 54, 55, 58, 80, 82, 86, 93, 94, 96, 97, 99, 102, 113
ratio, 8, 21, 69, 94, 95, 97, 98, 108, 114
RaxML, 113
re-sampling, *see* sampling
rearrangement, 63, 65, 76
reciprocal shortest distance, *see* RSD
replacement, 5, 99, 100
Replication, 76
RNA, 58
Robinson-Foulds, 17, 79, 112
robust, 14, 17, 92, 103
root, 28, 55, 71, 72, 110
RoundUp, 19, 66, 77, 78, 80, 82, 84, 111
RSD, 71, 77, 80, 111

sampling, 12, 21, 24, 27, 28, 40–42, 85, 93
score, 7, 8, 10, 27, 28, 35, 36, 38, 54, 56, 66, 67, 69–71, 73–75, 84, 92, 100, 103, 111, 114, 116
sensitivity, 69, 70, 72, 80, 82, 87, 99–101, 104
sequence alignment, 7, 8, 12, 19, 22, 26, 53, 79, 88, 108, 113
sequence similarity, 70, 71, 76, 82–84, 86, 92, 113, 114, 116
sequencing, 1, 68

SGML, 90
Shigella flexneri, 102
similarity, *see* sequence similarity
simulation, 8–10, 12, 14, 24, 26–28, 34, 40, 92, 99–101
Smith–Waterman, 79
SOAP, 90
Sodalis glossinidius, 102
SP, 60, 67, 70–74
SP-tolerance, 73, 74
speciation, 49, 50, 52, 54, 65–67, 71–73, 76, 78, 79, 85, 99, 101, 112
specificity, 80, 82–84, 87, 100, 101, 108
stable pairs, *see* SP
statistical distribution
  chi-square, 95, 97, 98
  normal, 25, 53, 94, 97, 98, 104
  uniform, 33, 40, 42
  Zipfian, 9, 41
statistical significance, 8
storage, 57
subset, 35, 55, 67, 74, 95, 98, 104
substitution, 19, 21, 22, 36, 99, 100, 113
substitution matrix, 5, 6, 23, 91
Sulfolobus solfataricus, 102
Swiss-Prot, 55, 84, 88

taxon, 75
Tcoffee, 55
Thermoplasma acidophilum, 102
Thermoplasma volcanium, 102
threshold, 55, 56, 100, 111
time-reversible, 12, 22, 69
tolerance, 69–73
topology, 2, 14, 17, 19, 22, 25, 42, 55, 72, 79
  "comb" structure, 112
  unrooted, 22, 23, 59, 62
trade-off, 2, 70, 72, 87
transcription, 65
tree, 1, 2, 9, 14, 19, 22–25, 33, 39–42, 45, 51, 53, 54, 56, 57, 62, 63, 68, 75–77, 79–82, 86, 87, 92, 94, 102–104, 108, 112, 113
  branch, 2, 20, 22, 23, 32, 33, 40–42, 46, 72, 94
  leaf, 33, 59, 62, 112
  reconciliation, 51, 77, 80, 87, 92, 113
triangle inequality, 69, 70
triplet, 20, 22–29, 32, 35, 36, 42–46, 69, 78, 109

UniProt, 88

validation, 55, 70, 75, 76, 86, 92, 99, 104, 108, 116
variance, 10, 12, 20–22, 24–33, 35, 36, 38, 40, 43, 44, 46, 53, 71, 77, 93–95, 98, 100, 107, 112, 116
variance-covariance matrix, 23, 25, 45, 46, 97
verified pairs, *see* VP
vertex cover, 74
Vibrio cholerae, 102
Vibrio fischeri, 102
violation, 44, 69, 70, 98
VP, 67, 68, 70, 73, 74
VP-tolerance, 74

webpage, 111
weight, 74, 75
weighted least-squares, *see* WLS
Wigglesworthia glossinidia, 102
witness of non-orthology, 51, 56, 57, 66, 73
WLS, 40
Wolbachia pipientis, 102

xenology, 56, 92

yeast, 74

Curriculum Vitae

Born on Nov 21st, 1980
Dual citizen of Switzerland and United States

Education

- ETH Zurich — PhD in Science, 2004-2009
- ETH Zurich — Master of Science, Biotechnology Major, 1999-2003
- Gymnase de la Cité, Lausanne — Federal Maturity type XB (latin-mathematics)

Research Experience

- 2004-present, PhD Student, G. Gonnet's Lab, Computer Science, ETH Zurich
  Large-Scale Comparative Genomics Using Pairwise Evolutionary Distances
- 2003-2004, Guest Researcher, C. Lursinsap's Lab, Chulalongkorn University, Thailand
  Disease Gene Mapping through Linkage Disequilibrium Analysis
- 2003, Master thesis, GQ Chen's Lab, Tsinghua University, Beijing, China
  Study of Polyhydroxyalkanoates Biocompatibility Based on Protein Adsorption Assay
- 2002, Semester project, V. Hatzimanikatis's Lab, Northwestern University, USA
  Modelling of the Translation's Initiation in Prokaryotes.
- 2002, Semester project, M. Parsek's Lab, Northwestern University, USA
  Study on the Quorum Sensing Controlled Genes in P. aeruginosa

Teaching Experience

- Guest lecturer in *Bioinformatik: Vertiefung*, ETH and Uni Zurich, 2008
- Guest lecturer in *Probabilistic Modeling in Molecular Evolution*, ETH Zurich, 2008
- Guest lecturer in *Introduction to Bioinformatics I*, Universität Basel, 2006, 2007, and 2008
- Head TA, *Foundations of Computer Science: Computational Science,* ETH Zurich, 2008
- TA for courses *Informatik II, Linear Algebra, Datenstrukturen und Algorithmen, Introduction to Computational Science, Scientific Computing,* ETH Zurich, 2005-2007

Awards, Honors, Grants

- Travel Grant from US Department of Energy, 2008
- Swiss-Prot 20th Anniversary Conference, Best Poster Award (3rd Place), 2006
- Travel Grant from Swiss Proteomics Society, 2006
- 4 Awards with ETH Team at iGEM (synthetic biology), MIT 2005
- Dean's list, Northwestern University, 2002
- Fellow of the Swiss Study Foundation, 2002-2008

Professional Organizations and Services

- Performed peer-reviews for Systematic Biology, Genome Biology, Bioinformatics, and Information Science.
- CALZONE (Colloquium for Ambitious Life-Scientist in Zurich: Opportunity for Networking and Exchanges), initiator and coordinator, 2005-2006
- Young European Biotech Network, founding member
- STARTbiotech conference, co-organizer, 2001

Peer-Reviewed Publications

- A Altenhoff and C Dessimoz
  Phylogenetic and Functional Assessment of Orthologs Inference Projects and Methods
  PLoS Comp Biol 2009, 5:1, e1000262

- A Roth, GH Gonnet, C Dessimoz
  Algorithm of OMA for large-scale orthology inference
  BMC Bioinformatics 2008, 9:518

- A Szalkowski, C Ledergerber, P Krähenbühl, C Dessimoz
  SWPS3 – A multi-threaded vectorized implementation of Smith-Waterman for the IBM Cell/B.E.
  BMC Research Notes 2008, 1:107

- MT Holder, DJ Zwickl, C Dessimoz
  Evaluating the Robustness of Phylogenetic Methods to Among-Site Variability in Substitution Processes
  Phil. Trans. R. Soc. B., 2008, 363:1512, pp. 4013-4021

- C Dessimoz and M Gil (equal contribution)
  Covariance of Evolutionary Distances Estimated from Optimal Pairwise Alignments
  BMC Evolutionary Biology, 2008, 8:179

- C Dessimoz, D Margadant, GH Gonnet
  DLIGHT - Lateral Gene Transfer Detection Using Pairwise Evolutionary Distances in a Statistical Framework
  RECOMB 2008, Singapore, LNCS 4955, pp. 315-330

- C Ledergerber and C Dessimoz
  Alignments with Non-overlapping Moves, Inversions and Tandem Duplications in O(n4) time
  COCOON 2007, LNCS 4598, pp. 151-164
  also selected for publication in Journal of Combinatorial Optimization, 2008, 16:3, pp. 263-278

- A Schneider, C Dessimoz, GH Gonnet
  OMA Browser - Exploring orthologous relations across 352 complete genomes
  Bioinformatics 2007, 23(16), pp. 2180-2182

- C Dessimoz, M Gil, A Schneider, GH Gonnet
  Fast Estimation of the Difference between two PAM/JTT Evolutionary Distances in Triplets of Homologous Sequences
  BMC Bioinformatics 2006, 7:529

- C Dessimoz, B Boeckmann, A Roth, GH Gonnet
  Detecting Non-Orthology in the COGs Database and Other Approaches Grouping Orthologs Using Genome-Specific Best Hits
  Nucleic Acids Res 2006, Jul 11;34(11), pp. 3309-3316

- C. Dessimoz, G Cannarozzi, M Gil, D Margadant, A Roth, A Schneider, GH Gonnet
  OMA, a Comprehensive, Automated Project for the Identification of Orthologs from Complete Genome Data: Introduction and First Achievements
  RECOMB CG 2005, Dublin, LNBI 3687, pp. 61-72

- M Gil, C Dessimoz, GH Gonnet
  A Dimensionless Fit Measure for Phylogenetic Distance Trees
  J Bioinform Comput Biol, Vol. 3, No. 6 (2005), pp. 1429-1440