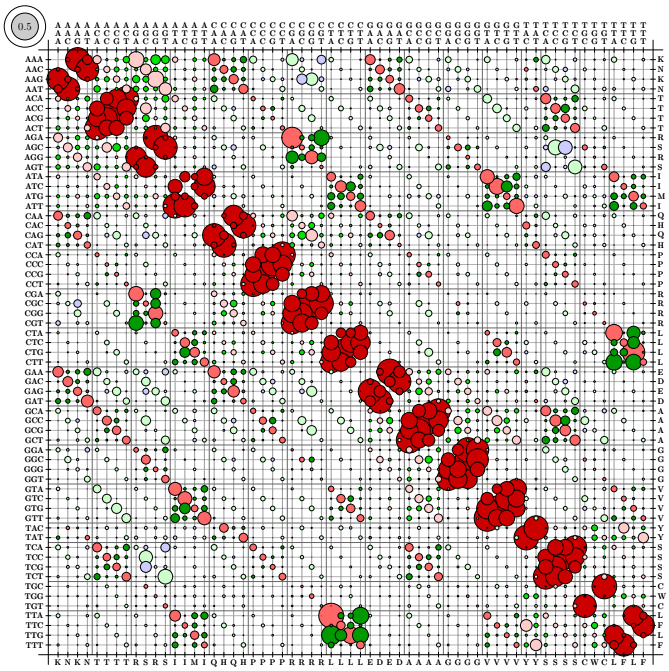
Markov Models for Protein Sequence Evolution

Carolin Kosiol

Wolfson College

A dissertation submitted to the University of Cambridge
for the degree of Doctor of Philosophy



European Molecular Biology Laboratory,
European Bioinformatics Institute, Wellcome Trust Genome Campus, Hinxton,
Cambridge, CB10 1SD, United Kingdom.

Email: kosiol@ebi.ac.uk

March 3, 2006

To My Family

This thesis is the result of my own work and includes nothing which is the outcome of work done in collaboration except where specifically indicated in the text.

This dissertation is not substantially the same as any I have submitted for a degree, diploma or other qualification at any other university, and no part has already been, or is currently being submitted for any degree, diploma or other qualification.

This thesis does not exceed the specified length limit of 300 single-sided pages of double-spaced text as defined by the Biology Degree Committee.

Carolin Kosiol

Markov Models for Protein Sequence Evolution

Summary

Carolin Kosiol
Wolfson College

March 3, 2006

Markov models for protein sequence evolution were introduced 30 years ago. However, although Markov models are widely used in database searches, alignment programs, phylogeny and recently in comparative genomics, their properties are seldom investigated. This thesis presents new tools to to analyse Markov models as well as new models.

I develop a method called almost invariant sets (AIS) which summarises the information given in Markov models by identifying subgroups of residues with high probability of change amongst residues of the same group but small probability of change between groups.

Protein evolution can be modelled on the amino acid and the codon level. I present a model called an aggregated Markov process which combines both levels and allows the study of their relationship. The aggregated Markov process can explain some of the non-Markovian behaviour observed in the literature, and it suggests that protein evolution should be modelled with codon-level rather than amino acid substitution models.

So far empirical models have only been formulated on the amino acid level. I have estimated empirical codon models using maximum likelihood techniques. Results indicate that modelling of the evolutionary process is improved by allowing for of single, double and triple nucleotide changes. A grouping of the 61 sense codons into subsets using the AIS algorithm shows that the affiliation between a triplet of DNA and the amino acid it encodes is a main factor driving the process of codon evolution. Both factors, multiple nucleotide changes as well as the strong influence of the genetic code and physico-chemical properties, are not part of standard mechanistic models, and this new view of how codon evolution proceeds leads to consequences for selection studies. An assessment of the accuracy of empirical codon models using likelihood ratio tests and a performance analysis in phylogenetic inference are accomplished.

Acknowledgements

Firstly, I would like to thank my supervisor Nick Goldman for teaching me thoroughness in science, for his endless patience looking at bubble plots and being unable to say no to all the conference related travel I came up with. His guidance and help have been invaluable. Thank you to Ari Löytynoja, Simon Whelan, Lee Bofkin, Fabio Pardi and the 'visiting' members of the Goldman group for many science and non-science discussions. I have very much enjoyed and benefited from the meetings with my thesis advisory committee consisting of internal members, Ewan Birney and François Nedélec, and the external member, Tim Hubbard. All this together with EMBL-predocs made EBI the right place to be.

Ian Holmes has been a great collaborator who sends wine bottles for 'destructive testing' of his code. His program DART has proven very valuable for the estimation of empirical codon models. Ziheng Yang's PAML package was helpful in many ways throughout this work, from calculating evolutionary distances to hi-jacking the source code to test the utility of empirical codon models. I am also thankful to Ziheng for days out in London: one day for assistance with PAML-hacking, several others for PHYLO-group meetings. Thanks to Gerton Lunter and Leo Goodstadt for providing me with their carefully constructed datasets. Jessica Severin, Abel Ureta-Vidal and Michael Hoffman were always helpful when I was pestering them with questions about Ensembl database.

I am indebted to the Wellcome Trust and EMBL for funding my Ph.D. and making this thesis possible.

My family and friends have been a source of distraction and counter-balance to lonely academic work. Still, some managed to contribute towards the completion of this Ph.D. Thanks to the members of 15 Glisson Road, Alistair Robinson, Rachel Jackson, Josh Robinson, Nathan Chong and Nicola Petty for feeding me with cake & curries through the last phase of writing-up.

I am very grateful to my mother Brigitte and uncle Hans-Viktor for encouragement, optimism and their sense of humour about the whole Ph.D project. My brother Stephan has improved aesthetics of probably every powerpoint presentation I have given during my Ph.D. My father Hans has taught me how to swim in many ways.

Finally, I would like to thank my school teacher Dr. Schön at Schloßgymnasium where I first thought that applied maths problems are fun.

Contents

Chapter 1

Introduction

For an indefinite time I clung to the [time] machine as it swayed and vibrated, quite unheeding how I went, and when I brought myself to look at the dials again I was amazed to find where I had arrived. One dial records days, and another thousands of days, another millions of days, and another thousands of millions [...]

(H.G. Wells in The Time Machine, 1895 [Wel05])

1.1 Evolutionary models for protein sequences

Zuckerkandl and Pauling [ZP62] showed that molecular sequences are 'documents of evolutionary history'. The composition of the molecular sequences of organisms living today has been passed on from ancestors from long ago. During this evolutionary history, the molecular sequences undergo modifications. Unused segments get lost, new parts are added and small changes occur because replication is not error free. Whatever modifications have happened, they will have left traces in the molecular sequences. Research in molecular evolution tries to reconstruct this process and to work out the mechanisms which lead to today's variety of organisms. This thesis deals with the study of protein sequences which have evolved from a common ancestor and whose evolution can be modelled with

probabilistic models and phylogenetic trees.

The first molecular sequences available in larger numbers were amino acid sequences. Indeed, in their investigations Zuckerkandl and Pauling sequenced the amino acid sequences of hemoglobin of different species, counted the differences, and speculated how many million years ago two species diverged from a common ancestor. Accordingly, early models for DNA were of theoretical nature, while amino acid models could be based on real sequences. Jukes and Cantor [JC69] proposed a simple model of instantaneous rates of change between nucleotides and Neyman [Ney71] showed how this could be turned into a Markov process [DEKM98] describing how DNA sequences evolve along an evolutionary tree. Neither models were built from real data nor applied to any significant data at the time. In contrast, Dayhoff and colleagues [DEP72] introduced a Markov model of protein evolution which was based on (by their standards) large amounts of data. This was the first major attempt to derive an empirically-based probabilistic model of amino acid substitution, and it resulted in the development of the widely used amino acid replacement matrix known as the PAM matrix based on 1572 changes inferred by parsimony in 71 sets of closely related proteins [DSO78].

Probabilistic models have the advantage that they can provide more accurate estimates of the evolutionary distance than simple counting methods: when comparing reasonably divergent sequences, counting the raw sequence identity (percentage of identical sites) underestimates the the amount of evolution that has occurred because, by chance alone, some sites will have undergone multiple substitutions.

Substitution models are also fundamental for the estimation of phylogenetic trees with distance matrix, Bayesian and maximum likelihood (ML) methods [Fel03]. This thesis will focus on applications of evolutionary models in ML phylogenetics.

ML, in common with Baysian methods, offers effective and robust ways to obtain a tree topology estimate and to measure our confidence in that estimate [Fel03]. The optimal tree topology is that with the highest likelihood. Furthermore, substitution matrices have proved very useful for other topics, including homology detection (e.g., the PAM matrices used by BLAST [AGM$^+$90]), and sequence alignment (e.g., CLUSTAL [THG94]; see also Durbin *et al.* [DEKM98] for a introduction to methods used).

In 1992, Jones *et al.* [JTT92] employed much the same methods as Dayhoff and colleagues, but based the estimation of their JTT matrix on a larger sequence database. In 2001 Whelan and Goldman [WG01] used a maximum likelihood estimation method to generate the WAG matrix. The PAM, JTT and WAG models give increasingly good descriptions of the average patterns and processes of evolution of large collections of sequences [WG01], but such 'average' models can fail to describe proteins with particular functions and structures [GTJ98]. For example, in functional domains where hydrophobicity is important, glycine (G) is rarely replaced by arginine (R). However, the replacement between the two occurs more frequently when hydrophobicity is not important. Likewise many seemingly different amino acids are good helix-formers (A, L) and many seemingly similar amino acids may differ much in their contribution to forming helices (L, M) [GTJ98]. Thus, when evolution 'sees' a helix structure as important, rates of amino acid replacements will depend on the propensity for helix-formation of the amino acids. In such cases the amino acid replacement models have been improved by incorporating functional and structural properties of the proteins. For example, Overington *et al.* [ODJ$^+$92] observed different patterns of amino acid replacements for different structural environments, Jones *et al.* [JTT94] demonstrated that transmembrane proteins have markedly different replacement

dynamics and Goldman *et al.* [GTJ98] considered different protein secondary structure categories, e.g. α-helices, β-sheets, turns and loops, with each category further classified by whether it is exposed to solvent or buried in the protein core, and inferred an amino acid replacement matrix for each of the categories. Furthermore, Adachi and Hasegawa [AH96] and Yang *et al.* [YNH98] have implemented models of amino acid replacement derived from mitochondrially-encoded proteins, Adachi *et al.* [AWMH00] compiled a chloroplast-derived amino acid replacement matrix, and Dimmic *et al.* [DRMG02] derived a matrix for retroviral polymerase proteins.

In real proteins, chemical interactions between neighbouring sites or the protein structure affects how other sites in the sequence change. Steps have been made towards context dependent models, where the specific characters at neighbouring sites affect the sites evolution. For example, sites in areas with differing secondary structure are assigned different substitution matrices by Goldman *et al.* [GTJ98]. Those matrices are then combined as a mixture model in the form of a hidden Markov model (HMM). The HMM model exploits the dependency of amino acid evolution on its solvent accessibility and its secondary structure environment. HMMs were first introduced to phylogenetic inference by Felsenstein and Churchill [FC96], who used the approach to describe local similarities in evolutionary rate. Koshi and Goldstein [KG95] have presented further context-dependent models and Dimmic, Mindell and Goldstein [DMG00] developed a model using estimated 'fitness' values for different amino acids that affected their probabilities of substitution. Furthermore, Robinson *et al.* [RJK$^+$03] developed a model that included the dependency between codons resulting from the secondary and tertiary structure of the protein.

In the meantime experimental techniques to determine DNA sequences im-

proved during the 1980s. The invention of the polymerase chain reaction (PCR) [MFS+86, MF87] allowed segments of DNA to be gathered quickly from very small amounts of starting material. Sanger and colleagues, in another revolutionary discovery, invented the method of 'shotgun' sequencing, a strategy based on the isolation of random pieces of DNA from the host genome to be used as primers for the PCR amplification of the entire genome [SNC77]. The amplified portions of DNA are then assembled by the identification of their overlapping regions, to form contiguous transcripts. The introduction of computers to reassemble the fragments dramatically increased the speed of sequencing and allowed the tackling of large genomes. The International Human Genome Consortium finished the human draft sequence in June 2000 [Con01], and finally completed the human genome sequence in April 2003 [Con04a], the 50th anniversary of the discovery of DNA.

Together with a variety of molecular sequences of non-human species (e.g. mouse [Con02], rat [Con04b], dog [LTWM+05], chimpanzee [SC05]) this very large quantity of data is now publicly available for evolutionary studies. At the DNA level, gene finders have been developed which exploit variation in the evolutionary process of exons, introns, splice sites, and intergenic regions to improve their accuracy (e.g. [MD02], [FKH+03]). Furthermore, phylogenetic HMMs are becoming increasingly popular and starting to find mainstream use, with their results now routinely used to examine the degree of conservation shared between genomic sequences [SH05].

In ML phylogeny, amino acid sequences are popular, because they evolve more slowly than DNA and are easier to align. Amino acid sequences are also less prone to 'saturation' effects that some phylogenetic inference methods handle poorly. Amino acid frequency biases are often less marked than DNA nucleotide

frequency biases. However, DNA sequences contain more information and to study protein evolution by modelling the evolutionary process on coding DNA is appealing, because it allows us to take the genetic code into account. There are 20 amino acids, but 64 possible codons. Three amino acids – arginine, leucine and serine – are encoded by six different codons, while another five can each be produced by four codons which only differ in the third position. A further nine amino acids are specified by a pair of codons which differ by a transition substitution at the third position, while isoleucine is produced by three different codons and methionine and tryptophan by only a single codon. Codon-level models insist that these differences matter, and more importantly they are able to distinguish between codons which produce the same amino acid and those that do not. Codon models allow the study of whether there is a tendency of substitutions maintaining the encoded amino acid (synonymous changes) to be accepted by selection less, equally, or more frequently than those that change the amino acid (nonsynonymous changes). Thus, by introducing a parameter describing the rate ratio of nonsynonymous to synonymous changes, it is possible to measure the effect of natural selection on the sequence. Markov models of codon substitution were first proposed by Goldman and Yang [GY94], and Muse and Gaut [MG94] and since then they have been modified in various ways to improve the detection of selection (e.g. [YNGP00], [MG05]).

Finally, substitution models can be distinguished into two types of models. Mechanistic models are often formulated on the codon level and take into account factors such as transition-transversion bias, codon frequencies and nonsynonymous-synonymous substitution bias. Empirical models do not explicitly consider factors that shape protein evolution, but attempt to summarise the substitution patterns from large quantities of data. This distinction is very useful

and I will classify all the evolutionary models above in a later chapter of this thesis.

To date, empirical models have only been formulated on the amino acid level, but not on the codon level. The availability of genome scale data – there are 14 chordate genomes released [BAC+06] now – means that now for the first time there is sufficient data available to estimate and study various empirically-based probabilistic models of codon evolution. This is one of the goals of this thesis.

1.2 Outline of the thesis

This thesis is largely concerned with probabilistic mathematical models describing the evolution of protein sequences.

Chapter 2 introduces the Markov process as a mathematical framework to study protein evolution and points out its assumptions and properties. Particular parameterisations of the existing Markov process models for amino acid and codon sequence evolution are discussed. As the parameters have been chosen to be biologically informative, the actual values are of biological interest. ML techniques can be applied for the estimation of the parameters, and likelihood ratio tests can be used to decide whether a given parameter is significant.

Dayhoff and colleagues [DEP72, DSO78] were the first to apply Markov process models to protein evolution. When I re-visited these roots at the start of my Ph.D., I was surprised to find that the implementations of the Dayhoff model differed among phylogenetic software packages. Markov models are usually expressed in terms of an instantaneous rate matrix, but the Dayhoff model was originally only published in terms of time dependent probability matrices. Since different methods were used to calculate the instantaneous rate matrix, this led to the different implementations of supposedly identical methods. I have followed

the problem up in a detailed study which identifies the methods used to derive an instantaneous rate matrix from the probability matrices published by Dayhoff, performs a comparison and suggests a standard version to facilitate communication among researchers. This study has been published [KG05], and is included in the Appendix A as a part of my studies which is related to the main thesis topics, but not part of the connected argument developed here.

Since Dayhoff, numerous substitution models for the average process of protein evolution have been devised, as well as very specialised models considering functional and structural properties of proteins. However, little work has been done to abstract important and intuitive features from the models and compare them across the different models. Chapter 3 presents a grouping method, Almost Invariant Sets (AIS), to classify amino acids, starting from amino acid replacement matrices. The grouping method is inspired by the conductance, a quantity that reflects the strength of mixing in a Markov process [Beh00]. I present groupings resulting from two standard amino acid models. The AIS method is derived for amino acid models, but is extended and applied to codon models at the end of the chapter.

Further investigating the behaviour of Markov processes, I study the relationship between codon models and amino acid models in Chapter 4. The analysis is based on a model called the aggregated Markov process (AMP) which combines rate heterogeneity among different codon sites of the protein and the properties of the amino acids encoded by the sequences. AMPs behave in a non-Markovian manner, as the Chapman-Kolmogorov equation shows, and they can explain some of the non-Markovian behaviour of experimental data described in the literature. The latter results suggests that protein evolution should be modelled with codon-level rather than with amino acid substitution models.

In Chapter 5 I have estimated an empirical codon model using data taken from the alignment database Pandit [WdBQ$^+$06] and the estimation program Dart [HR02]. Results indicate that modelling the evolutionary process is improved by allowing for single, double and triple nucleotide changes and that the affiliation to one amino acid is a main factor driving the process of codon evolution, as shown by an AIS grouping into codon sets. Taking advantage of the recent dramatic increase in genomic data I extended the estimation of empirical codon models to matrices estimated from human-mouse, mouse-rat and human-dog coding pairs of orthologous genes. The multiple data sets allow us to distinguish biological factors from estimation errors and alignment artifacts.

In Chapter 6 the empirical codon model estimated from the Pandit database is tested for utility in phylogenetic inference. An assessment of the accuracy of empirical codon models using ML performance analysis is accomplished. Mechanistic parameters are re-introduced into the empirical codon model to establish a foundation for its future use in studies of natural selection.

Specific conclusions are made at the end of each chapter, with possible suggestions for future research. These themes are unified in Chapter 7.

Chapter 2

Mathematical background

2.1 Markov process models

The aim of this chapter is to highlight the assumptions made to model the process of evolution and to introduce the mathematical terminology that will be used throughout the thesis.

One of the primary assumptions made in defining evolutionary models is that future evolution is only dependent on its current state and not on previous (ancestral) states. Statistical processes with this lack of memory are called Markov processes. The assumption itself is reasonable because during evolution, mutation and natural selection can only act upon the molecules present in an organism and have no knowledge of what came previously. Large-scale mutational events occur, including recombination [PC02], gene conversion [Saw89, SW99], or horizontal transfer [Doo99] that might not satisfy this condition. However, provided the above events are rare and their effect is small, they can be ignored.

To reduce the complexity of evolutionary models, it is often further assumed that each site in a sequence evolves independently from all other sites. There is evidence that the independence of sites assumption is violated (e.g. [OJSB90]). In real proteins, chemical interactions between neighbouring sites or the protein

structure affects how other sites in the sequence change. Steps have been made towards context dependent models, where the specific characters at neighbouring sites affect the sites evolution (see discussion of context dependent models in section 1.1 and [KG95, GTJ98, DMG00, RJK$^+$03]).

However, the majority of current models describing context dependency are experimental. At present there are no studies comparing performance of competing methodology and the independence of sites assumption is still commonly accepted because of its computational benefits. For example, even if a change at one site is only dependent on the state of its immediate neighbours, these neighbours are affected by their neighbours, and so on. This 'contagious dependence' [LH04] makes computation very difficult because the evolution of a sequence has to be described as a whole leading to an enormous number of character states. The effort required to calculate the likelihood of a tree grows with N^3, where N is the number of states (see section 2.7). In this thesis, the assumption of independence will be made.

2.2 Instantaneous rate and probability matrices

A substitution model is a description of how one character state changes into another at a given site in a molecular sequence. We are looking at substitution models, where the evolution at each site in a sequences is modelled as a continuous-time Markov process [Nor97].

The Markov model asserts that one protein sequence is derived from another protein sequence by a series of independent substitutions, each changing one character in the first sequence to another character in the second during evolution. Thereby we assume independence of evolution at different sites. A continuous-time Markov process is defined by its instantaneous rate matrix $Q = (q_{ij})_{i,j=1,\ldots N}$,

an $N \times N$ matrix, where N is the number of character states. Three types of character alphabets will be considered for protein evolution in this thesis: nucleotides, amino acids and codons ($N = 4$, 20 and 64, respectively). The matrix entry q_{ij}, $j \neq i$, represents the instantaneous rate of change from state i to state j, independently at each site.

Waiting in a particular state, changes at each site occur as a Poisson process with these given rates. The rate at which any change from that state occurs is the sum of all the rates of changes from that state, and this determines the waiting time in a given state before moving to another. The q_{ii} entry of the matrix is set to be the minus the sum of all other entries in that row, representing (-1 times) the rate at which changes leave state i:

$$q_{ii} = -\sum_{j,\, j \neq i}^{N} q_{ij} \quad .$$

Molecular sequence data consist of actual characters at some given time, not the rate at which they are evolving. The quantity needed for calculations is the probability of observing a given character after time t has elapsed. Let $P_{ij}(t)$ be the probability of a site being in state j after time t given that the process started in state i at that site at time 0. Since there are N character states, i and j take the values $1, 2, \ldots, N$ and we can write the probability $P_{ij}(t)$ as an $N \times N$ matrix that we denote $P(t)$. In evolutionary studies it is necessary to be able to compute the probability matrix $P(t)$ for any real evolutionary time (distance) $t \geq 0$. This is achieved using the instantaneous rate matrix Q, which is related to $P(t)$ via

$$P(t) = e^{tQ} \quad , \tag{2.1}$$

where the exponential of a matrix is defined by the following power series [Nor97],

with I being the appropriate identity matrix:

$$e^{tQ} = \quad I + tQ + \frac{(tQ)^2}{2!} + \frac{(tQ)^3}{3!} + \cdots \tag{2.2}$$

In practise, this power series is calculated numerically using standard linear algebra techniques (see [MVL03] for 19 ways to calculate the exponential of a matrix). The most popular method in molecular phylogeny uses eigendecomposition [LG98].

If a reasonable Markov process is left evolving for a long time, the probability of finding it in a given state converges to a value independent of the starting state; this distribution is known as the equilibrium distribution $\pi = (\pi_1, \ldots, \pi_N)$. The equilibrium distribution π can be found by solving

$$\pi P(t) = \pi$$

for any $t > 0$, or equivalently [Nor97]:

$$\pi Q = 0 \quad .$$

Time and rate are confounded, and only their product can be inferred without extrinsic information [Fel81]. Consequently, we can normalise the instantaneous rate matrix with any factor. Typically in phylogenetic applications, Q is normalised so that the mean rate of replacement at equilibrium ($\sum_i \sum_{j \neq i} \pi_i q_{ij}$, where π_i is the equilibrium frequency of state i) is 1, meaning that times (evolutionary distances) are measured in units of expected substitutions per site. Sometimes, especially in protein studies, PAM distance are used. The PAM distance effectively corresponds to 100 $\times$ the normal distance (see Appendix A for details).

2.3 Reversibility and connectedness

Markov processes for amino acid sequence evolution can have two important properties: connectedness and reversibility. In a connected process, for all $i, j \in 1, \ldots, N$ there exists a time $t > 0$ such that

$$P_{ij}(t) > 0$$

and this is equivalent to $P_{ij}(t) > 0$ for all $t > 0$ [Nor97]. Connected Markov processes have a unique equilibrium distribution [Nor97].

Reversibility means that

$$\pi_i P_{ij}(t) = \pi_j P_{ji}(t) \text{ for all } i, j \in \{1, \ldots, N\} \text{ and } t > 0.$$

A consequence of reversibility is that the process of protein sequence evolution is statistically indistinguishable from the same process observed in reverse.

All reasonable Markov models of sequence evolution will be connected and thus will have a unique equilibrium distribution. In contrast, reversibility is just a mathematical/computational convenience [Fel81], but not a biological requirement (e.g., see [JKA+05] and [McD06] for further discussion).

2.4 Amino acid and codon models

This section gives an introduction to definitions of standard amino acid and codon models as well as the biological motivation which has led to the choice of parameters used in these models. A more detailed view of protein evolution, although restricted to experimental results, can be found in Lercher *et al.* [PPL06].

For the amino acid models in this thesis we assume that amino acid sites in an alignment evolve independently and according the same reversible Markov process, which is defined by the 20×20 instantaneous rate matrix $Q = (q_{ij})_{i,j=1,\ldots,20}$.

Under the reversibility assumption, instantaneous rates q_{ij} can be expressed as:

$$q_{ij} = \pi_j s_{ij} \qquad \text{for all } i \neq j \quad , \tag{2.3}$$

where $s_{ij} = s_{ji}$ is a symmetric term often denoted exchangeabilities, and frequency parameters π_j that describe the equilibrium frequencies of the amino acids (see [WG01] and [GGJM05]). The 380 non diagonal entries of Q can therefore be described by just 208 terms, namely 189 exchangeabilities s_{ij} (remember the Q matrix is normalised to mean rate 1) and 19 frequency parameters π_i (remember that the π_i sum to 1).

In general, the number of independent parameters for a reversible substitution model with N character states can be calculated as

$$\underbrace{\frac{N^2 - N}{2} - 1}_{\text{exchangeabilities allowing for rate normalization}} + \underbrace{(N-1)}_{\text{frequencies of characters}} = \frac{N(N+1)}{2} - 2. \tag{2.4}$$

The +F method [CAJ+94] proposes that during the analysis of a specific dataset the fundamental properties described by the s_{ij} of a database remain unaltered, but that the π_j describing evolutionary pressures acting on the sequences be replaced with a set of frequencies more appropriate to the data analysed. Although the distinction between the exchageabilties and the amino acid frequencies has been introduced because of statistical convenience, there is some experimental evidence for the validity of the assumption that exchangeability parameters s_{ij} do represent characteristic patterns of amino acid substitutions applicable across a wide range of proteins. Grantham [Gra74] introduced an amino acid matrix whose entries are defined by a distance formula considering the chemical composition of the side chain, the polarity and the molecular volume. Grantham's approach was extended by Xia and Li [XL98] in a study of the relevance of 10 amino acid properties to protein evolution and by Atchley *et al.* [AZFD05] who

summarise 500 amino acid attributes. Yampolsky and Stolzfus [YS05] derive a distance matrix from amino acid exchangeability (selection) experiments.

There is also clear evidence for biases in amino acid composition among different organisms, proteins as well as within protein parts. Proteins of thermophilically organisms are thermophilic more stable than orthologous proteins of relatives that live at moderate temperatures [SH03]. For example, for thermophilic bacteria a well known observation is a biased amino acid composition rich in charged residues such as Glu, Asp, Lys and Arg and poor in polar residues such as Ser, Thr, Asn and Gln. These effects operate at the level of selection, not mutation, and they are not restricted to differences between organisms. Systematic biases in amino acid composition between different proteins caused by varying structural and functional constraints of the proteins are addressed by using different equilibrium distributions π. In fact, structural constraints can further enhance variation in amino acid composition such that it plays a role within proteins. For example, a study by Nishikawa *et al.* [NFN03] shows that the differences in amino acid composition bias of thermophilic and mesophilic proteins are most obvious in the composition of the protein surface. This fits well with the view that exchangeabilities as well as amino acid frequencies are significantly different for different structural elements of proteins (see [GTJ98] and section 1.1).

Summarising the patterns of character composition variation on the level of protein-coding DNA is more difficult than on the amino acid level, because of the variety of the different biological factors influencing base composition which act at different scales (e.g., selective constraints, DNA repair, mutational influences, translation, DNA strand etc.). A simple and useful way of measuring composition variation within or among genomes is through their relative usage of the bases

GC and AT (note that these are the bases that pair on the two strands of genomic DNA, and hence skew to G or A will equally apply to C or T). Table 2.4 illustrates the huge variation among genomes, with bacteria in the range 25%-75% GC, unicellular eukaryotes showing similar variation, and vertebrates relatively little variation.

Nucleotide composition can also vary considerably within genomes. In particular, eukaryotes present a diverse, and little understood, spectrum of composition variation. For example, in *Drosophila melangaster*, noncoding regions vary in GC content, but the most marked base composition variation seen between coding regions in terms of the GC content at synonymous sites [SSHW88]. Even within genes there is variation in GC content, with a trend for higher GC at the 5' end of genes [KEW98]. This could relate to translation as I will later discuss in this section.

Group	*Species*	*GC content (%)*
Bacteria	*Closteridium tetani*	29
	Escherichia coli	51
	Streptomyces coelicolor	72
Eukaryotes	*Plasmodium falciparum*	23
	Caenorhabditis elegans	35
	Saccharomyces cerevisae	35
	Arabidopsis thaliana	35
	Drosophila melangogaster	42
Vertebrates	*Mus musculus*	40
	Homo sapiens	41
	Fugu rubrioes	44
Organelles	*H.sapiens* mitochondrion	44
	A. thaliana chloroplast	49

Table 2.1: Table of GC content (taken from McVean [McV05]).

Simple GC content as a measure of base composition bias does not take into account the underlying genome structure. However, the coding nature of genes imposes constraints on base composition through the structure of the genetic code. The physicochemical properties of an amino acid are most strongly correlated with the second position of the codon, while the third codon position often has considerable freedom to vary due to the degeneracy of the genetic code. Hence, proteins with identical functions can potentially be encoded for by genes of considerably different nucleotide base composition. The nonrandom usage of different codons is referred to as codon bias.

Whole-genome analysis obscures any variation in codon bias that might occur between genes within a genome. Within *Escherichia coli* there is significant variation among genes [Ike81]. What could be the reason for codon bias among the genes in the *E. coli* genome? An important clue is the finding that the more commonly used codons typically correspond to the more abundant tRNA species. Organisms typically do not have tRNAs corresponding to every codon. Furthermore, different tRNAs are present in different copy numbers in the genome and are expressed at different levels. Variation in the cellular concentration of different tRNA species has consequences for the translational process. Codons which are translated rapidly, because they are recognised by an abundant tRNA, are used preferentially, while rare tRNAs tend to be avoided. It seems likely that the preferential use of rapidly translated codons is determined by selection upon the rate of elongation during translation, selection acting to reduce the cost of proof reading and to increase translational accuracy (see Akashi [Aka94] for a study on *Drosophila* and Akashi [Aka03] for a study on yeast). In many unicellular organisms, a strong correlation between codon biases and the level of gene expression has been found [RD04]. Selection for structural robustness

against mistranslation to avoid protein misfolding has been put forward as a possible explanation (Drummond et al. [DBA+05]). It has also been shown that for *E. coli* additional codon bias at the start of the gene (first 100 base pairs) can be observed [EWB93] and that the codon bias is related to gene length [EW96].

Many biological effects related to GC content and codon bias have been to complex to model. The only widely used approach so far has been the +F-models introduced by Cao *et al.* [CAJ+94] mentioned earlier in this section, which can handle variation from one dataset to another (i.e. one gene or protein to another). The extra complexity that the presence of different biases within a specific data set can occur, has often been cited as reason why phylogenetic inference can make errors [LPH+92]. Some work has been done to address this problem on the DNA level with more complex models [GG95, GG98]. However, I do not know of any approach on the codon level and further consideration of this issue is beyond the scope of this thesis.

Markov models of codon substitution were first proposed by Goldman and Yang [GY94] and Muse and Gaut [MG94]. This thesis will mainly refer to a simplified model called M0 from Yang *et al.* [YNGP00]. The M0 model assumes that codons evolve through a series of single nucleotide substitutions. Changes between codons differing in more than one position cannot occur instantaneously. A stop codon signals the end of a coding sequence. Substitution to or from a stop codon is extremely likely to be deleterious and will rarely be fixed within the population. They can therefore reasonably be considered to occur at rate 0. Since stop codons do not occur at random, it is thus best to remove them from the sequence data, reducing the number of required states for the model (e.g., to 61 states for protein coding sequences following the universal genetic code).

A (single nucleotide) substitution may either be a transition or a transversion

depending on the underlying nucleotide substitution which caused it. Transitions are changes between A and G (purines), or between C and T (pyrimidines). Transversions are changes between purine and pyrimidine. It is widely observed that transitions occur more often than transversions [BPWW82].

In the same year as their proposal of the structure of DNA Watson and Crick suggested a mechanism of point mutation that immediately favours transitions. Their idea was that while the maintenance of the double helix demands that purines always paired pyrimindines and vice versa, mutations could occur if disfavoured tautomeric forms of the four bases were misincorporated during DNA replication to form A-C and G-T translation mispairs [Wak96].

The incorporation of disfavoured tautomers of bases into growing DNA strand has, however, not subsequently been observed to play a significant role in mutation. Instead it appears that the standard forms of the bases are misincorporated directly, and in many different pairings [Wak96]. These include purine-purine and pyrimidine-pyrimidine pairs, and the orientation of mispairs can differ greatly from the standard helical geometry. However, the free energy of pairings between the template base and the incoming deoxynucleoside triphoshates as well as pressure to maintain the Watson-Crick structure do influence the rate of misincorporation; consequently, G–T, G–A, and A–C are generally the most frequent mispairs. Thus the bias towards transitions starts at the level of mutations.

However, transition and transversion events have very different probabilities of generating synonymous/nonsynonymous changes, and nonsynonymous transversions tend to involve more dissimilar amino acids than nonsynonymous transitions [HH91]. In turn, this tends to lead to greater selective pressure against transversions than against transitions. Thus selection-level processes can also influence transition-transversion bias in protein-coding DNA.

Although the transition-transversion bias, denoted by the parameter κ, is known to be a general property of DNA sequence evolution it is more pronounced in animal mitochondrial DNAs than in nuclear or chloroplast DNAs (see [Wak96], [EY99]). Using ML techniques, Yoder and Yang [YY99] showed that this bias κ is variable among evolutionary lineages.

An important distinction between different codon substitutions is whether or not they are synonymous: does the new codon code for the same amino acid? Natural selection acts upon the protein for which the sequence codes. A synonymous change can be assumed to be essentially neutral as it does not affect the resulting protein. On the other hand, a nonsynonymous mutation resulting in a different amino acid may be fixed in the population with a greater or smaller rate depending on the level of selection the protein is undergoing. A parameter ω is often used to give a dataset-specific measure of the selective pressure on a protein's evolution. In the M0 model, the parameter ω relates to when an amino acid change occurs, but not to what the amino acid interchange is.

By explicitly incorporating the codon equilibrium frequencies as parameters π_j the M0 model and related models can consider codon bias on a per protein level. However, these standard codon models do not considered selection on synonymous sites, as for example, required to adequately model the effects of selection on translation efficiency and accuracy. Furthermore, variation within a gene (e.g., treating the first 100 nucleotides of a gene differently) is not covered in the standard view.

Combining all the features mentioned above, the instantaneous rate matrix

Q of M0 is defined by:

$$q_{ij,i\neq j} = \begin{cases} 0 & \text{if } i \text{ or } j \text{ is a stop codon or requires} > 1 \text{ nucleotide substitution,} \\ \pi_j & \text{if } i \to j \text{ synonymous transversion,} \\ \pi_j\kappa & \text{if } i \to j \text{ synonymous transition,} \\ \pi_j\omega & \text{if } i \to j \text{ nonsynonymous transversion,} \\ \pi_j\kappa\omega & \text{if } i \to j \text{ nonsynonymous transition,} \end{cases} \tag{2.5}$$

where ω is the nonsynonymous-synonymous rate ratio, κ is the transition-transversion rate ratio, and π_j is the equilibrium frequency of codon j. The nonsynonymous-synonymous rate ratio ω is an indicator of selective pressures at the protein level, with $\omega < 1$ meaning purifying selection, $\omega = 1$ neutral selection and $\omega > 1$ positive selection.

The selective forces acting upon a protein are highly informative about its biological function and evolutionary history [YB00]. For example, the interaction of proteins through their regulatory and metabolic network are also reflected in the selection acting upon them. Recently, it was demonstrated that the more interactions a protein has with other molecules, the slower it evolves, and that proteins operating in complexes (e.g., involved in translation and DNA repair) are on average, more restricted than those with simple house-keeping functions [AB05]. In viruses, the sites on envelope proteins interact with host molecules and are targets for the immune system, leading a host-viral 'arms race', and the amino acids at the interacting sites evolve under continuous positive selection [NBC$^+$05]. Another reason for positive selection is the compensation of deleterious mutations [BKO$^+$04].

Studies examining the effect of natural selection in proteins estimated an average ω across all the sites in an alignment and usually drew similar conclusions: adaptive evolution is rare and most proteins are under strong purifying selection [EIG96].

The lack of positive selection in these studies suggested by these studies is most probably an underestimate by the true amount. The contents of a genome have been evolving for million of years and are highly adapted to the function they perform. Consequently, purifying selection will have been acting on the majority of sites in a protein to maintain this function and the average value of ω would be expected to be low. Positive selection would normally expected to affect only a few key residues in a protein, to successfully find its footprint during molecular evolution requires a more sophisticated approach.

The most advanced codon models do not assume a single fixed ω for all sites, but permit consideration of a distribution of ω values over sites. Yang *et al.* [YNGP00] summarised the standard selection models, and called them M0 to M13. The models M0, M7, M8 are most commonly used. M7 describes variation in ω between 0 and 1 with a β-distribution allowing for purifying selection and neutral evolution only. M8 contains the β-distribution of M7, but also includes a single variable category of ω to describe positive selection. When a statistical test shows that M8 explains the data significantly better than M7 and the additional category is greater than 1, positive selection is inferred. Recently there have been also updates of the M8 models, called M8a and M8*, which have favourable statistical properties when applied in statistical tests [WYGN04]. Typical values inferred with these models have most sites under strong purifying selection (e.g., $\omega < 0.2$) and some times a few sites evolving neutrally ($\omega \approx 1$) or with positive selection ($\omega > 1$)

Models which allow for a distribution of ω values over sites are not applied to data in this thesis. However, they are eventually referred to when considering future directions in some topics.

Finally I would like to point out that the mixture models described above

assume selective pressures acting on a protein remain constant through evolutionary time. This is a suitable assumption when sites are under consistent selective pressure to change, but inappropriate when molecular adaptation occurs during relatively short periods of evolutionary time. to detect episodic adaptive evolution, models have been updated to allow ω to vary during evolution. These function either by allowing ω to vary in all branches in a phylogeny (across-branch models) [YN02] or across a prespecified group of branches in a phylogeny (branch-sites models). In summary, among the parameters discussed here ω is allowed to vary the most: between proteins, between sites and between lineages.

2.5 Gamma distribution

In the post-genomic era, the sequences used for phylogenetic and comparative genomic analyses are increasingly long and therefore more likely to contain regions evolving under different mutational processes and selective constraints [Bof06]. Many of the complexities of molecular evolution are primarily manifested as a difference in the rate that sites change. For example, the active units in a genome evolve under different and complex selective constraints, but techniques for identifying them from non-functional areas usually rely on the observation that they evolve more slowly [MBHG03]. Similarly, the degeneracy of the genetic code results in substantially different rates of evolution at the three codon positions [Yan96, Mas02, Bof06].

The standard approach to characterising this variation is to describe each site's rate as a random draw from a statistical distribution, whilst maintaining all other aspects of the evolutionary process. In other words, each site has a defined probability of evolving at a given rate, independent of its neighbours. It is also assumed that this rate is constant throughout evolution; a fast changing site

has an elevated evolutionary rate throughout the phylogenetic tree. Uzzel and Corbin [UC71] first suggested that in DNA the rate variation observed in coding sequences could be described using a Γ-distribution. Yang implemented this description as a probabilistic model, using a continuous Γ-distribution containing a single, biologically interpretable, shape parameter that can accommodate varying degrees of rate heterogeneity [Yan93]. The shape parameter is normally denoted α. The value of the shape parameter is inversely related to the degree of rate variation it describes: for values below 1.0 it describes the extensive rate variation characteristic of functional regions (e.g., protein coding sequences), with numerous sites evolving at a low rate and a few faster evolving sites; values greater than 1.0 convey limited rate variation, which occurs often in non-functional regions (e.g., pseudogenes). Later, Yang proposed breaking the distribution into a pre-specified number of categories to make the model computationally more efficient [Yan94b]. This approach has been successfully employed in many studies and on DNA as well as amino acid data. The inclusion of Γ-distributed rates has been demonstrated to affect, and usually improve, the estimation of other evolutionary parameters, including the tree topology [Yan96].

2.6 Maximum likelihood (ML)

ML is a long established method for statistical inference [Fis25, Edw72], extensively tested for many years and successfully applied to a wide variety of problems. The likelihood value, L, used in phylogenetic inference is the probability of observing the data (e.g., a set of aligned amino acid sequences) under a given phylogenetic tree and a specified model of evolution:

$$L = \text{Prob}(\text{data}|\text{tree, model}) \quad . \tag{2.6}$$

ML is used in phylogenetics to find the optimal set of parameters contained within the tree and model that best describes the observed data [Fel03]. The tree describes the topology of the evolutionary relationships between the sequences and a set of branch lengths describing how much evolution has occurred in different regions of the tree.

Given a branch length, the substitution model is used to calculate the probabilities of characters either remaining the same or replacing each other, and, using the pruning algorithm of Felsenstein [Fel81] the likelihood L can be calculated. The parameters comprising the tree and model are estimated using numerical optimisation procedures [Fel03] to find the highest likelihood, which represents the combination of parameter values that best describes the observed data.

Good software for phylogenetic maximum likelihood calculations is available. I will use PAML [Yan94b] and DART [HR02] in this thesis.

2.7 Calculating the likelihood of a tree: Felsenstein's pruning algorithm

Suppose we have a set of aligned sequences of length l and, for simplicity, consider DNA sequences with $N = 4$ character states A,C,G,T. We are given a phylogeny with b branches and a substitution model allowing us to calculate probabilities $P_{ij}(t)$, the probability that state j will exists at the end of a branch of length t, if the state of the start of the sequence is i. We make two assumptions: firstly, evolution in different sites on the given tree is independent, and secondly, evolution in different lineages is independent.

The first of these allows us to take the likelihood and decompose it into a

product

$$L = \text{Prob(data|tree, model)} = \prod_{i=1}^{l} \text{Prob}(D^{(i)}|\text{tree, model}) \tag{2.7}$$

where $D^{(i)}$ is the data at the ith site of the alignment. This means that we only need to know how to calculate the likelihood at a single site. Suppose now that we have a tree and data at site i. An example with five sequences (tips of the tree) is shown in Fig. 2.1.

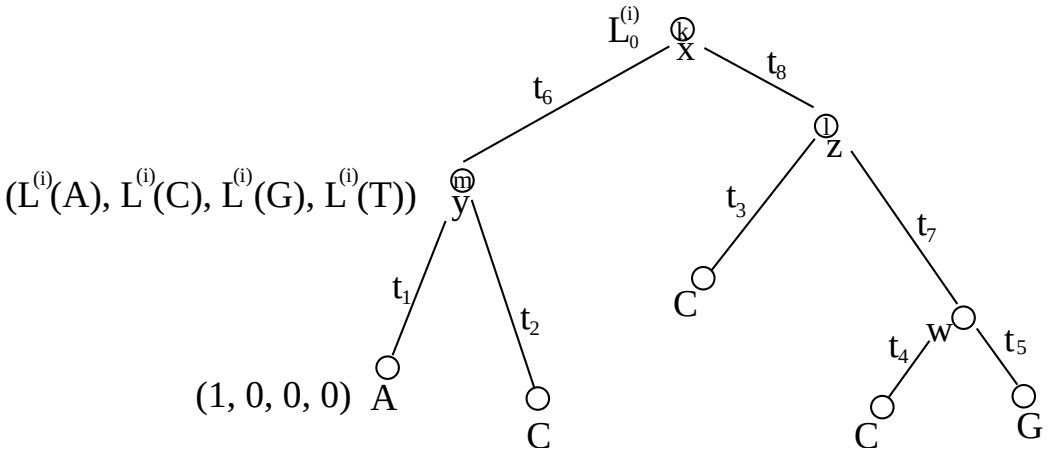


Figure 2.1: Illustration of Felsensteins's pruning algorithm. Character states at the nodes are indicated x, y, z, w (possible states at internal nodes) and A, C, C, C, G (observed states at tips), and three nodes (k, and its descendents l, m) are labelled. See text for details.

The likelihood of the observed data for site i is the sum, over all possible nucleotides x, y, z, w that may have existed at the interior nodes of the tree, of the joint probability of the observed data (here: A,C,C,C,G as shown in Fig. 2.1) and the unobserved ancestral nucleotides:

$$\text{Prob}(D^{(i)}|\text{tree, model}) =$$
$$\sum_x \sum_y \sum_z \sum_w \text{Prob}(A, C, C, C, G, x, y, z, w|\text{tree, model}), \tag{2.8}$$

with each summation running over all four nucleotides.

The second assumption about the independence of lineages allows us to decompose the probability into a product

$$\text{Prob}(D^{(i)}|\text{tree, model}) =$$

$$\sum_x \sum_y \sum_z \sum_w \text{Prob}(x) P_{xy}(t_6) P_{yA}(t_1) P_{yC}(t_2) P_{xz}(t_8) P_{zC}(t_3) P_{zw}(t_7) P_{wC}(t_4) P_{wG}(t_5),$$

where $\text{Prob}(x)$ is the probability of observing nucleotide x at this point in the tree. Standard procedure is to assume this equal to π_x, from the equilibrium distribution of the substitution model. Equation 2.8 can be written as:

$$\text{Prob}(D^{(i)}|\text{tree, model}) = \sum_x \pi_x \left(\sum_y P_{xy}(t_6) P_{yA}(t_1) P_{yC}(t_2) \right)$$
$$\times \left(\sum_z P_{xz}(t_8) P_{zC}(t_3) \times \left(\sum_w P_{zw}(t_7) P_{wC}(t_4) P_{wG}(t_5) \right) \right)$$

As indicated by Felsenstein [Fel81], it is possible to express the calculation of $\text{Prob}(D^{(i)}|$ tree, model) for any topology in this form, i.e., the product of terms containing only one summation. He suggested an interpretation of the last equation as a flow of information down the tree, and proposed the following algorithm to compute Eq. 2.8.

For a given a node k, let $L_k^{(i)}(x)$ be the probability of everything that is observed from node k down, at site i, conditional on node k having state x (conditional likelihood of the subtree). The algorithm is most easily expressed as a recursion that computes $L_k^{(i)}(x)$ for each node of the tree from the same quantities at the immediate descendant nodes. Suppose that node k has immediate descendants l and m, which are the bottom ends of branches of length t_l and t_m. Then we can compute

$$L_k^{(i)}(x) = \left(\sum_z P_{xz}(t_l) L_l^{(i)}(z) \right) \left(\sum_y P_{xy}(t_m) L_m^{(i)}(y) \right) \tag{2.9}$$

This simply means that the probability of everything at or below node k is the product of the probability of events of both the descendant lineages. The extension to multifurcating trees simply involves more factors on the right side of Eq. 2.9.

To start the process, we need values of $L^{(i)}(x)$ at the tips of the tree. Considering the definition of $L^{(i)}(x)$, if state A is found at a tip, the values will be

$$\left(L^{(i)}(A), L^{(i)}(C), L^{(i)}(G), L^{(i)}(T)\right) = (1, 0, 0, 0) \quad ,$$

and similarly for states C, G, T: whichever base is seen at the tip has the corresponding value $L^{(i)}$ set to 1, and all others to 0. Were we to fail to observe a nucleotide, so that a base was encoded as N, the $L^{(i)}$ would be $(1, 1, 1, 1)$ because the observation of total ambiguity would have probability 1, no matter what the true nucleotide, given that we did not observe any base.

The recursion is applied starting at any node that has all of its immediate descendants being tips. Then it is applied successively to nodes further up the tree until the topmost node in the tree. The same procedure can be carried out on all sides of the topmost node until there are $L^{(i)}(x)$ values for each branch leading to it. The result is $L_0^{(i)}$ for the topmost subtree, i.e. the entire tree. We then complete the evaluation of the likelihood for this site by making an average of these over all four bases, weighted by their prior probabilities under the probabilistic model:

$$\text{Prob}(D^{(i)}|\text{tree}, \text{model}) = \sum_x \pi_x L_0^{(i)}(x) \quad . \tag{2.10}$$

Once the likelihood for each site is computed, the overall likelihood of the tree is the product of these as noted in Eq. 2.7.

The computational effort needed is as follows: for each site the recursion is carried out b times (once for each branch); each time it comprises N calculations, each one of these the product of two terms each involving a sum over N characters. We perform $\mathrm{O}(b \cdot N^2)$ work on $\mathrm{O}(l)$ sites, so the computational effort is $O(l \cdot b \cdot N^2)$ [Fel03]. However, the calculation of the transition probabilities requires matrix multiplication with $\mathrm{O}(N^3)$ (see [LG98] or Appendix A for how to calculate the exponential of a matrix with eigen-decomposition). Thus the total complexity of the likelihood calculation of a tree with respect to the number of states is $\mathrm{O}(N^3)$. However, in practice the pruning algorithm takes up more running time than calculation of the transition probabilities when calculating a tree for a considerable amount of species.

2.8 Statistical testing

Likelihood inference can be used to address many biologically important questions by examining parameters estimated from the data or by comparing how well related models explain sequence evolution. For example, parameter values may be used to identify those data that contain the most extreme transition-transversion mutation bias amongst a set of alignments, those that are the most conserved, and those that have evolved under the greatest selective constraints. One of the most appealing features of ML estimation is that it provides a long-established method for statistical inference [Fis25, Edw72, Fel03]. In addition to providing point estimates of parameters, it also gives information about the uncertainty of our estimates, for example through the calculation of confidence intervals (CIs), which allows the rigorous comparison of competing hypotheses. CIs are a simple measure of how much we trust parameter estimates from the data. A large CI suggests a parameter that is difficult to estimate, whilst a small CI is indicative

of an accurate parameter estimate. The range of the CI can be used as a simple measure for testing hypotheses; for example, to test if a parameter is not significantly different from 1, the 95% CI estimate can be examined and if it does not include 1, the hypothesis is rejected.

Likelihood also offers another very powerful way of comparing hypotheses, the likelihood ratio test (LRT) [Fel03, Fis25, Edw72]. This requires the formation of two competing hypotheses, represented by models with different restraints on their parameters. For example, the relative frequency of transitions and transversions in DNA evolution can be investigated through two competing hypotheses. The null hypothesis (H0, with likelihood L0) describes the rates of transitions and transversions as equal, and the alternate hypothesis (H1, with likelihood L1) has transitions occurring at a different rate than transversions. Note that in this example H0 and H1 only differ in this way. Subsequently L1 and L0 are optimised: for L1 all parameters are optimised; for L0 all parameters are optimised but the transition-transversion ratio, which is constrained to equal 1. The ML values ($\hat{L}$) for the competing hypotheses are compared using the LRT statistic

$$2\Delta = 2\ln(\frac{\hat{L}_1}{\hat{L}_0}) = 2(\ln(\hat{L}_1) - \ln(\hat{L}_0)) \quad . \tag{2.11}$$

This statistic has very useful properties for significance testing when certain conditions are met. In straightforward cases, when H0 can be formed by placing restrictions on the parameters in H1, the hypotheses are said to be nested and for significance testing 2Δ can be compared to the 95% point of a χ^2_n distribution [Fel03], where n is the number of parameters by which H0 and H1 differ (see [WG99] and [GW00] for more complex cases).

Many complex biological problems about the evolutionary process have been investigated using carefully constructing nested hypotheses, and the approach

now plays a crucial role in many phylogenetic studies [YNGP00, WLG01]. It is not possible to use χ^2 distributions for assessing the significance of LRTs under certain conditions, the most common occurring when comparing non-nested models. The rigorous comparison of hypotheses in this situation necessitates simulation methods for obtaining the required distribution for significance testing [ET93, Gol93].

Chapter 3

How to compare models using Almost Invariant Sets (AIS)

3.1 Introduction to amino acid groupings

In this chapter I develop a criterion and method to group character states of substitution models, starting from replacement matrices. The criterion and grouping method identify disjoint sets of character states with a high probability of change amongst the elements of each set, but small probabilities of change between elements of different sets. Since we are looking at protein evolution, the character states can be amino acids as well as codons. However the primary example in this chapter will be amino acid models. The sets of groupings are easy to understand and may be readily compared for the numerous replacement matrices that have arisen in the last 30 years from studies of protein evolution.

To date, little progress has been made in comparing amino acid replacement matrices such as the PAM, JTT and WAG matrices. Although the entries of these matrices are numerically quite different, it has been hard to express differences in an intuitive form. Groupings of the 20 amino acids derived directly from the matrices are a suitable tool to analyse and compare these different models, and are much more comprehensible than graphical or tabular representations of

an instantaneous rate matrix. A reliable and biologically meaningful method to summarise the information they contain, and that could lead to comparisons and contrasts of replacement patterns in different parts of proteins and in proteins of different types, would assist in a better understanding of protein sequence evolution. In addition to uses in studying models of protein sequence evolution, other important applications of amino acid groupings have already been established; for example, by Wang and Wang [WW99] in protein design and modelling and by Coghlan *et al.* [CMB01] to develop filtering algorithms for protein databases.

Several methods have been proposed to classify amino acids, although rarely using evolutionary information. Grantham [Gra74] introduced an amino acid distance formula that considers the chemical composition of the side chain, the polarity and the molecular volume. This approach was extended by Xia and Li [XL98] in a study of the relevance of 10 amino acid properties to protein evolution. Grantham and Xia and Li presented their results in the form of distance matrices, whereas French and Robson [FR83] arranged their results in two-dimensional diagrams using multidimensional scaling. Taylor [Tay86] also adopted a graphical approach and developed Venn diagrams of amino acids sets. The unions and intersections of a Venn diagram allow determination of (potentially hierarchical) sets of amino acids that might be evolutionary conserved. The number of possible subsets is large, however, and includes many that have little physical meaning. The interpretation of these Venn diagrams requires detailed expert knowledge.

Accordingly, a recent approach from Cannata *et al.* [CTRV02] is interesting since it automates the group-finding process. These authors propose a branch and bound analysis based on amino acid replacement probability matrices, but their method suffers from two drawbacks. First, the classification criterion used has no

clear evolutionary meaning. Second, the approach leads to different groupings for different time periods of the same matrix (e.g., PAM120 and PAM250), whereas it is desirable to have a criterion that is dependent on the replacement patterns of evolution but independent of the time scale on which evolution may be observed.

The method proposed in this chapter has its origin in the convergence diagnosis of Markov chain Monte Carlo methods [Beh00]. I introduce the conductance, a measure for the grouping of amino acids into non-empty sets, whose value will indicate the quality of the classification. Unfortunately the measure itself does not suggest a method that would determine optimal groupings. I explain the relationship between the eigenvalues and eigenvectors and the structure of the amino acid replacement matrix. Mathematically speaking, we are looking for a structure of the Markov matrix that is almost of block diagonal type. Markov matrices that show an almost block diagonal structure also show a low conductance value. The identification of the block structure leads to an algorithm that produces groupings for a given amino acid matrix. I apply the conductance measure and the grouping algorithm to standard amino acid and codon replacement matrices in section.

This chapter has largely been published [KGB04]. Significant changes to the published version are found in the introduction and discussion sections. Additionally, this chapter augments the published version with a section on how to apply the AIS method to codon models. Where necessary, notation has been adapted in order to be consistent with the rest of the thesis.

3.2 The almost invariant sets algorithm

3.2.1 The conductance measure

The goal is to identify sets of amino acids with a high probability of change amongst the elements of each set but small probability of change between elements of different sets. Such sets represent groups of amino acids which replace each other regularly but do not interchange across groups much, and so the groupings identify common evolutionary changes.

The starting point is to consider amino acid replacement matrices $P(t)$, for example the PAM series ([DSO78], see also Appendix A). In order for groupings to be interpretable in terms of the processes of evolutionary replacement of amino acids, and not levels of divergence between protein sequences, we expect that groupings should perform equally under measures based on (e.g.) PAM120 or PAM250 (the PAM matrices $P(t)$ for t approximately equal to 1.20 and 2.50, respectively [DSO78]), and that optimal groupings derived from these matrices should be the same. The measure presented here has been inspired by the conductance [Sin92, Beh00], a measure of the strength of mixing of a Markov process that is used in the convergence diagnosis of Markov chain Monte Carlo methods [Beh00]. Below, I redefine the conductance in terms of the instantaneous rate matrix Q instead of the probability matrix $P(t)$, to fulfil the requirement for independence of the measure and particular times t.

Let Q define a Markov process that is connected and reversible with equilibrium distribution π, and is normalised so that the mean rate of replacement at equilibrium is 1. (The mean rate of replacement is given by $\sum_i \sum_{j \neq i} \pi_i q_{ij}$. Dividing any Q by this mean rate of replacement provides a matrix with a mean rate of unity, so that evolutionary distances t are measured in units of expected

numbers of changes per site [LG98].) Now consider an amino acid sequence of N sites. The expected number of changes of i to j per unit time is $N\pi_i q_{ij}$, or $\pi_i q_{ij}$ per site. Similar analysis can be carried out for *sets* of amino acids. Let $A_1, \ldots, A_K$ be K proper subsets of the set of amino acids $A = \{1, \ldots, 20\}$, where $A_k \cap A_l = \emptyset$ for $k, l = 1, \ldots, K$, $k \neq l$, and $\bigcup_k A_k = A$. If π_i is the ith component of the equilibrium distribution π, we expect to observe

$$N \cdot \sum_{i \in A_k,\, j \in A_l} \pi_i q_{ij}$$

changes per unit time from subset A_k to subset A_l, $k \neq l$, in the whole sequence, or

$$f_{kl} = \sum_{i \in A_k,\, j \in A_l} \pi_i q_{ij}$$

changes per site. The quantity f_{kl} is called the *flow* from subset A_k to subset A_l.

When the Markov process is close to equilibrium, the frequencies of the amino acids remain more or less the same. The frequency of amino acids of subset A_k, called the *capacity* c_k of A_k, is then

$$c_k = \sum_{i \in A_k} \pi_i \ .$$

The ratio

$$\phi_{kl} = \frac{f_{kl}}{c_k}, \quad k \neq l, \tag{3.1}$$

is called the *conductance* [Beh00]. This is the expected number of changes from subset A_k to subset A_l per site per unit time when commencing at subset A_k.

Using the above definition we can define a new matrix $\Phi = (\phi_{kl})_{k,l=1,\ldots,K}$:

$$\Phi = \begin{pmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1K} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{K1} & \phi_{K2} & \cdots & \phi_{KK} \end{pmatrix}$$

where the off-diagonal entries are defined as above, and the diagonal entries ϕ_{kk} are defined by the mathematical requirement that each row sums to zero. The matrix Φ is itself then an instantaneous rate matrix. If we had "perfect" subsets, no changes between the subsets could be observed and $f_{kl} = 0$ for all $k \neq l$. Thus Φ would be a null matrix. In the more general "imperfect" case, the expression

$$\varphi = \sum_{k} \sum_{l \neq k} \phi_{kl} \tag{3.2}$$

measures the difference between Φ and the null matrix. I therefore use φ as a measure of the quality of the partition of the set A of 20 amino acids into K groups $A_1, \ldots, A_K$. A grouping with $\varphi = 0$ is said to be a groupings into invariant subsets.

Example 1

To set ideas, we consider a simple illustrative system of seven amino acids with rate matrix Q having the following block diagonal form:

$$\begin{pmatrix} -.35 & .35 & 0 & 0 & 0 & 0 & 0 \\ .35 & -.35 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -2.1 & 2.1 & 0 & 0 & 0 \\ 0 & 0 & 2.1 & -2.1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -.7 & .35 & .35 \\ 0 & 0 & 0 & 0 & .35 & -.7 & .35 \\ 0 & 0 & 0 & 0 & .35 & .35 & -.7 \end{pmatrix}.$$

The block diagonal structure of the rate matrix suggests the partition into $A_1 = \{1, 2\}$, $A_2 = \{3, 4\}$ and $A_3 = \{5, 6, 7\}$. Since this Markov process is reversible, the flow from set A_k to A_l is same as the flow from set A_l to set A_k:

$$\begin{aligned} f_{A_1 \to A_2} &= f_{12} = f_{21} = 0 \\ f_{A_1 \to A_3} &= f_{13} = f_{31} = 0 \\ f_{A_2 \to A_3} &= f_{23} = f_{32} = 0. \end{aligned}$$

The equilibrium distribution in this example is not unique, since the corresponding Markov process is not connected. The rates ϕ_{kl}, however, are independent of any choice of equilibrium distribution. Since the f_{kl} are all zero, we get

$$\begin{pmatrix} \phi_{11} & \phi_{12} & \phi_{13} \\ \phi_{21} & \phi_{22} & \phi_{23} \\ \phi_{31} & \phi_{32} & \phi_{33} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad .$$

Finally, the conductance measure is given by

$$\varphi = \sum_{k} \sum_{l \neq k} \phi_{kl} = 0.$$

In the general case, however, choosing a partition into sets is often not obvious. One may wish to consider all possible partitions. The total number of partitions of a set of n elements into non-empty subsets is the nth Bell number, B_n [Wei99]. The Bell numbers are given by the recurrence

$$B_{n+1} = \sum_{i=0}^{n} \binom{n}{i} B_i$$

where B_0 is defined to equal 1.

Example 2

To determine the number of possible partitions of the set of four letters $\{ATGC\}$, the fourth Bell number is computed as follows:

$$\begin{aligned} B_1 &= \binom{0}{0} B_0 = 1 \\ B_2 &= \binom{1}{0} B_0 + \binom{1}{1} B_1 = 2 \\ B_3 &= \binom{2}{0} B_0 + \binom{2}{1} B_1 + \binom{2}{2} B_2 \\ &= 1 + 2 + 2 = 5 \\ B_4 &= \binom{3}{0} B_0 + \binom{3}{1} B_1 + \binom{3}{2} B_2 + \binom{3}{3} B_3 \\ &= 1 + 3 + 6 + 5 = 15 \end{aligned}$$

The 15 possible partitions into non-empty subsets are

$$\begin{array}{lll} \{ATGC\}, & \{AT\}\{GC\}, & \{A\}\{TC\}\{G\}, \\ \{A\}\{TGC\}, & \{AC\}\{GT\}, & \{T\}\{AG\}\{C\}, \\ \{ATG\}\{C\}, & \{AG\}\{TC\}, & \{G\}\{AT\}\{C\}, \\ \{AGC\}\{T\}, & \{A\}\{GT\}\{C\}, & \{G\}\{AC\}\{T\}, \\ \{ATC\}\{G\}, & \{A\}\{GC\}\{T\}, & \{A\}\{G\}\{C\}\{T\}. \end{array}$$

Cannata *et al.* [CTRV02] have pointed out that for 20 amino acids there exist 51,724,158,235,372 (roughly 51×10^{12}) possible partitions. Furthermore, they list how these partitions are distributed among the partitions into particular numbers ($K = 1, \ldots, 20$) of sets. For example, under the restriction of partitioning only into exactly 8 sets, as many as 15×10^{12} partitions still have to be considered. This means that exhaustive enumeration of the groupings and calculation of the conductance measure to find the optimal grouping of 20 amino acids is out of the question. In sections 3.2.2 and 3.2.3 I describe a heuristic algorithm that seeks optimal or near optimal groupings of amino acids. One advantage of the algorithm presented is that the computational cost of searching for a high quality partition of the 20 amino acids into K subsets is independent of the value of K and the algorithm can easily be run for all non-trivial values of K ($2, \ldots, 19$, for amino acids) given any matrix Q. Once partitions of amino acids have been determined algorithmically one may calculate the conductance measure φ in order to exhibit the quality of the groupings.

3.2.2 Block Structure of Matrices

The aim of this and the next section is not to give a mathematical proof of the method used, but to motivate it by means of examples. For proofs I will refer to Deuflhard *et al.* [DHFS00].

Example 1 has indicated that blocks within matrices can act as "traps" for the flow between the sets and that choosing a partition accordingly results in a low

conductance score φ. In this section I state results that link certain properties of the eigenvalues and eigenvectors of an amino acid replacement matrix to a block diagonal or perturbed block diagonal structure of the matrix. The main idea is to identify an almost block diagonal structure of the replacement matrix in order to find good candidates with low conductance score φ among all possible partitions. The eigenvectors are especially suitable to identify time-independent groupings, since the eigenvalues for different time distances t of the probability matrix $P(t) = e^{tQ}$ (for example, PAM120 and PAM250) are different, but the eigenvectors remain the same [LG98].

Suppose the eigenvalues λ_i of $P(t)$, where $1 \leq i \leq 20$, are ordered according to

$$|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_{20}| \quad .$$

By the Perron-Frobenius theorem [Beh00], all eigenvalues are real and are contained in $[-1, 1]$. Since $P(t)$ is reversible it is known that for every right eigenvector there is a corresponding left eigenvector that corresponds to the same eigenvalue. The greatest eigenvalue λ_1 is unity and is called the Perron root. The right eigenvector corresponding to λ_1 is $e = (1, \ldots, 1)^T$, and the corresponding left eigenvector $\pi = (\pi_1, \ldots, \pi_{20})^T$ represents the equilibrium distribution under the assumption that it is normalised so that $\pi^T e = 1$. In matrix notation we have:

$$\pi^T P(t) = \pi^T \quad \text{and} \quad P(t)\, e = e \quad \text{for } t > 0 \quad .$$

The above results are true for a general Markov matrix. Moreover, the underlying Markov chains are known to be reversible (see section 2.3). Therefore,

$$\pi_i P_{ij}(t) = \pi_j P_{ji}(t) \text{ for all } i, j \in \{1, \ldots, N\} \text{ and } t > 0 \quad ,$$

or equivalently, in terms of some weighting matrix $\mathcal{D} = diag(\sqrt{\pi_i})$,

$$\mathcal{D}^2 P = P^T \mathcal{D}^2 \quad .$$

Throughout the subsequent analysis, we assume that the discrete states have been selected that all elements of π are strictly positive, or equivalently, that the weighting matrix $\mathcal{D}$ is nonsingular. Once $\pi > 0$, we may introduce the inner product $< \cdot, \cdot >_\pi$ as

$$< x, y >_\pi = x^T \mathcal{D}^2 y \quad .$$

Vectors x, y satisfying $< x, y >_\pi = 0$ will be called π-orthogonal.

We will now focus on matrices where we can decompose the 20 amino acids into invariant subsets $A_1, \ldots A_K$ of amino acids. This means that whenever the Markov process is in one of the invariant sets, e.g. A_1, it will remain in A_1 thereafter. If we use an appropriate ordering of the amino acid residues, the amino acid replacement matrix $P(t)$ appears in block diagonal form

$$B = \begin{pmatrix} D_{11} & 0 & \ldots & 0 \\ 0 & D_{22} & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & D_{KK} \end{pmatrix}$$

where each block D_{kk} $(k = 1, \ldots, K)$ is a Markov matrix, reversible with respect to some corresponding equilibrium sub-distribution. Assume again that each of these matrices is connected. Then, due to the Perron–Frobenius theorem, each *block* possesses a unique right eigenvector $e_k = (1, \ldots, 1)^T$ of length $\dim(D_{kk})$ corresponding to its Perron root.

In terms of the complete amino acid replacement matrix $P(t)$, the eigenvalue $\lambda = 1$ is K-fold degenerate and all K corresponding right eigenvectors can be written as linear combinations of the K vectors of the form

$$(0, \ldots, 0, e_k^T, 0, \ldots, 0)^T, \quad k = 1, \ldots, K. \tag{3.3}$$

As a consequence, right eigenvectors corresponding to $\lambda = 1$ are constant on each invariant set of states. This this means that the right eigenvectors can be written in the above form, which would reveal the block structure immediately. However, in general they will not appear in this 'ideal' form if found from some $P(t)$ (or Q) matrix. In the example below, I show that we can identify important features in the right eigenvector which will help us to recover the 'ideal' form.

Example 3

To obtain a block diagonal probability matrix we calculate $B = P(t) = e^{Qt}$, where Q is the block diagonal rate matrix of *Example 1* and $t = 1$:

$$P(1) = \begin{pmatrix} .75 & .25 & 0 & 0 & 0 & 0 & 0 \\ .25 & .75 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & .51 & .49 & 0 & 0 & 0 \\ 0 & 0 & .49 & .51 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & .56 & .22 & .22 \\ 0 & 0 & 0 & 0 & .22 & .56 & .22 \\ 0 & 0 & 0 & 0 & .22 & .22 & .56 \end{pmatrix} .$$

The eigenvalues of $P(1)$ are

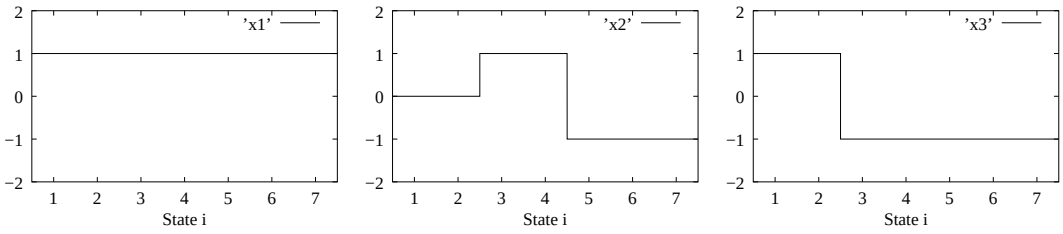
$$\lambda_1 = 1 \quad \lambda_2 = 1 \quad \lambda_3 = 1$$

$$\lambda_4 = 0.5 \quad \lambda_5 = 0.34 \quad \lambda_6 = 0.34 \quad \lambda_7 = 0.02$$

and the right eigenvectors corresponding to $\lambda = 1$ are

$$\begin{aligned} x_1 &= (1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1 \;\; 1)^T \\ x_2 &= (0 \;\; 0 \;\; 1 \;\; 1 \;\; -1 \;\; -1 \;\; -1)^T \\ x_3 &= (1 \;\; 1 \;\; -1 \;\; -1 \;\; -1 \;\; -1 \;\; -1)^T \end{aligned}$$

which are indeed linear combination of eigenvectors with the 'ideal' form (Eq. 3.3) and which form a π-orthogonal basis of the eigenspace corresponding to λ.



(a) (b) (c)

Figure 3.1: The eigenvectors x_1, x_2, x_3 of *Example 3*, corresponding to $\lambda = 1$ as functions of the states.

Figure 3.1 shows the eigenvectors x_1, x_2 and x_3 corresponding to $\lambda = 1$, as function of the seven states (corresponding to amino acids in the real application) s_i, $i \in \{1, \ldots, 7\}$. A constant level can be observed for each of the invariant sets {1,2} {3,4} and {5,6,7}. Moreover, the same pattern can be observed if we restrict the investigation to the sign structure σ_i, $i \in \{1, \ldots, 7\}$, of the states instead of the actual values. For example, the sign of state 1 is positive for eigenvectors x_1 and x_3 and is zero for eigenvector x_2. Thus the sign structure σ_1 for state 1 can be written $(+, 0, +)$. Analogously, we determine the sign structure of all states:

$$\begin{array}{cc} \sigma_1 = (+,0,+) & \sigma_2 = (+,0,+) \\ \sigma_3 = (+,+,-) & \sigma_4 = (+,+,-) \\ \sigma_5 = (+,-,-) & \sigma_6 = (+,-,-) \end{array}$$
$$\sigma_7 = (+,-,-).$$

The sign structure is the same for states of the same invariant set $\{1, 2\}$, $\{3, 4\}$ or $\{5, 6, 7\}$.

Stated more formally, and reverting to consideration of 20-state (amino acid) matrices: given a block diagonal substitution matrix P consisting of reversible, connected blocks, an equilibrium distribution $\pi > 0$ and a π-orthogonal basis $\{X_i\}_{i=1,\ldots,k}$ of its eigenspace corresponding to $\lambda = 1$, and we associating with

every state its particular sign structure

$$s_i \mapsto (\text{sign}(x_1)_i, \ldots, \text{sign}(x_K)_i) \quad i = 1, \ldots, 20 \quad ,$$

then the following statements hold:

- invariant sets are collections of states with common sign structure
- different invariant sets exhibit different sign structures.

A proof is given by Deuflhard *et al.* [DHFS00]. This indicates that the set of K right eigenvectors of the amino acid replacement matrix can be used to identify K invariant sets of amino acid residues via the sign structure.

3.2.3 Perturbation Theory

The previous section 3.2.2 related to matrices with perfect block diagonal structure. The standard amino acid replacement matrices like PAM [DSO78] and WAG [WG01] do not exhibit block diagonal structure. As mentioned in section 2.3, most amino acid replacement matrices are connected. This means that for any time $t > 0$ all the entries of their probability matrices $P(t)$ are non-zero. Therefore it is impossible to identify perfect invariant sets of amino acids. However, it is still possible to identify *almost* invariant sets (i.e. conductance measure $\varphi \approx 0$) via the sign structures σ_i, as the following example illustrates:

Example 4

We add a perbutation matrix E to the block diagonal matrix B of *Example 3*:

$$P := 0.8B + 0.2E$$

where the perturbation matrix E is given below:

$$\begin{pmatrix} .01 & .09 & .10 & .25 & .08 & .30 & .17 \\ .09 & .10 & .25 & .08 & .30 & .17 & .01 \\ .10 & .25 & .08 & .30 & .17 & .01 & .09 \\ .25 & .08 & .30 & .17 & .01 & .09 & .10 \\ .08 & .30 & .17 & .01 & .09 & .10 & .25 \\ .30 & .17 & .01 & .09 & .10 & .25 & .08 \\ .17 & .01 & .09 & .10 & .25 & .08 & .30 \end{pmatrix}.$$

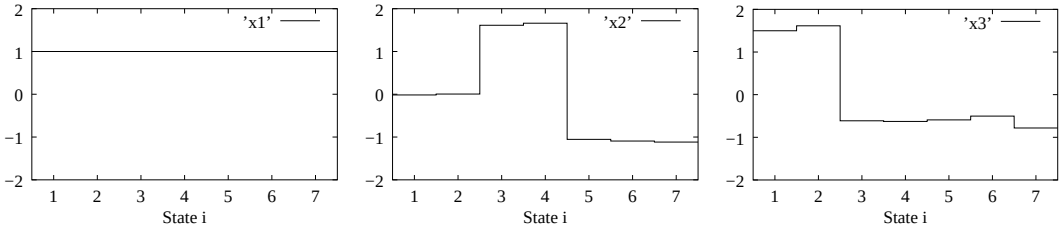
The eigenvalues of P are now calculated as

$$\lambda_1 = 1 \quad \lambda_2 = 0.85 \quad \lambda_3 = 0.76$$

$$\lambda_4 = 0.41 \quad \lambda_5 = 0.31 \quad \lambda_6 = 0.24 \quad \lambda_7 = -0.02 \ .$$

The eigenvalue spectrum of the perturbed block diagonal amino acid replacement matrix can then be divided into three parts: the Perron root $\lambda_1 = 1$, a cluster of two eigenvalues $\lambda_2 = 0.85$, $\lambda_3 = 0.76$ close to 1, and the remaining part of the spectrum, which is bounded away from 1. The right eigenvectors x_1, x_2, x_3 corresponding to $\lambda = 1, 0.85, 0.76$ are:

$$\begin{matrix} (1, & 1, & 1, & 1, & 1, & 1, & 1)^T \\ (-0.02, & 0.01, & 1.61, & 1.66, & -1.05, & -1.09, & -1.12)^T \\ (1.50, & 1.61, & -0.61, & -0.63, & -0.59, & -0.50, & -0.78)^T \end{matrix} \quad .$$



(a) (b) (c)

Figure 3.2: The eigenvectors x_1, x_2, x_3 of *Example 4*, corresponding to $\lambda = 1, 0.85, 0.76$, as functions of the states.

Figure 3.2 shows that for the perturbed block diagonal Markov matrix, nearly constant level patterns can be observed on the three almost invariant sets $\{1, 2\}$, $\{3, 4\}$ and $\{5, 6, 7\}$.

In order to have an automated procedure for determining the sign structure, we need to define a threshold value θ that will separate components with clear sign information from those that might have been perturbed to such an extent that the sign information has been lost. Elements $x_k(s)$ of x_k satisfying $|x_k(s)| > \theta$ are taken to have clear sign information, $\sigma_s(k) = +$ or $-$, whereas $\sigma_s(k) = 0$ if $|x_k(s)| < \theta$. For example, by choosing $\theta = 0.25$ in the above example, we ensure that all states $\{1, \ldots, 7\}$ still have clear defined sign structure and that at least one of the eigenvectors, apart from x_1, has a sufficiently large component $|x_k(s)| > \theta$. In this example, the small components of the eigenvectors $x_2(1) = -0.02$ and $x_2(2) = 0.01$ are neglected and we obtain the following sign structure:

$$\sigma_1 = (+, 0, +) \quad \sigma_2 = (+, 0, +) \quad \sigma_3 = (+, +, -) \quad \sigma_4 = (+, +, -)$$
$$\sigma_5 = (+, -, -) \quad \sigma_6 = (+, -, -) \quad \sigma_7 = (+, -, -)$$

This sign structure is identical to the sign structure of the unperturbed Markov matrix, leading to the same grouping of the states $\{1, 2\}$, $\{3, 4\}$ and $\{5, 6, 7\}$. *Example 4* indicates that the sign structure of eigenvectors corresponding to eigenvalues in the cluster around the Perron root λ_1 can be used to identify sets of amino acids that are almost invariant. An exact formulation and proof of the behaviour of the eigenvectors under the influence of perturbation is given in Deuflhard *et al.* [DHFS00].

3.2.4 Program

This section transforms the results of sections 3.2.2 and 3.2.3 above to an algorithm that has three steps:

1. Find states with stable sign structure.

2. Define equivalence classes.

3. Sort states to seek almost invariant sets.

Step 1: Find states with stable sign structure

First the π-orthogonal right eigenvectors forming the basis of the eigenspace corresponding to $\lambda = 1$ need to be found. I have used Mathematica which will do this automatically, and will also order the right eigenvectors according to the absolute value of their eigenvalues. Then we start from the heuristic that the sign of an eigenvector component is "more likely" to remain stable under perturbation the "larger" this component is. In order to make the positive and negative parts of the eigenvectors comparable in size, we scale them as follows:

For $k = 1, \ldots, K$, we split $x_k = x_k^+ + x_k^-$ component-wise, where $x_k^+(s) = \max(0, x_k(s))$ and $x_k^-(s) = \min(0, x_k(s))$, and we set $\tilde{x}_k = x_k^+/\|x_k^+\|_\infty + x_k^-/\|x_k^-\|_\infty$ (where $\|v\|_\infty$ is the maximum norm of vector v, defined as $\max_i |v(i)|$).

By means of a heuristic threshold value $0 < \delta < 1$, which is common for all eigenvectors, we then select those states that exhibit a "stable" sign structure according to

$$\mathcal{S} = \{s \in \{1, \ldots, N\} : \max_{k=1,\ldots,K} |\tilde{x}_k(s)| > \delta\}.$$

Only those states in $\mathcal{S}$ can be assigned to groups using the following procedure; states $s \notin \mathcal{S}$ are unclassifiable. *Step 1* is a check that all of the states (i.e., amino acids) have at least one of the eigenvectors x_k, $k > 1$, with a "significantly" large component $x_k(s)$. I have chosen $\delta = 0.5$ for the amino acid replacement matrices I have investigated. In the case of the occurrence of unclassifiable states the algorithm aborts. However, one could then lower the value of δ at the expense of a higher risk of a false assignment of the states into subsets. This situation never arose in the tested examples of residue matrices.

Step 2: Define sign equivalence classes

Based on the sign structures of the states in $\mathcal{S}$, we proceed to define K equivalence classes with respect to sign structures. As already indicated, the underlying idea is that only "significantly" large entries in the scaled vectors $\tilde{x}_k$ are permitted to contribute to a sign structure $\sigma_{(s,\theta)}$ for a state s with respect to some heuristic threshold value θ (with $0 < \theta < 1$) by virtue of

$$\begin{aligned} \sigma_{(s,\theta)} &= (\sigma_1, \ldots, \sigma_K) \\ \text{with } \sigma_k &= \begin{cases} \text{sign}(\tilde{x}_k(s)) & \text{if } |\tilde{x}_k(s)| > \theta, \\ 0 & \text{otherwise.} \end{cases} \end{aligned}$$

Two sign structures are defined to be equivalent if, and only if, their pointwise multiplication yields only non-negative entries. This has the effect that any entry 0 – as determined via the threshold – may be interpreted either as +1 or -1 [DHFS00]. Sign structures of states that are not equivalent are said to be inequivalent. (Note that this definition of equivalence class does not correspond to a formal equivalence relationship). The goal is to find the smallest threshold $\tilde{\theta}$, for which we can find an assignment into exactly k classes.

Step 3: Sort states to seek almost invariant sets

In step 2 we have found states with stable sign structures. It is now necessary to sort the states with respect to their sign structure, compute the number of almost invariant sets and finally determine the invariant sets. Various methods can be applied to this challenge, and I have decided to transform the problem to a graph colouring problem. Therefore I construct a graph where every state with stable sign structure is represented by a vertex and in which inequivalent states are connected by edges. Colouring this graph determines K colour sets $\mathcal{S}_1, \ldots, \mathcal{S}_K$ and I assign each of the states in $\mathcal{S}$ to one sign structure class. An introduction

to graph colouring and code that performs this task is given by Trick [Tri94].

By combining the three steps above we arrive at the following procedure to compute a partition into a particular number K of almost invariant sets:

Specify desired number of sets K

Read in the K eigenvectors with largest eigenvalues

Step 1: Find states with stable sign structure:

Set $\theta^- = 0$ and $\theta^+ = 1$
Step 2:
Set $\tilde{\theta} = \frac{\theta^- + \theta^+}{2}$ (bisection search to find θ

giving required number of subsets)

Determine the sign structures $\sigma_{(s,\theta)}$ with respect to $\tilde{\theta}$

Step 3: Calculate almost invariant sets and the number

of almost invariant sets, $\mathcal{K}(\tilde{\theta})$. Then:

if $(\mathcal{K}(\tilde{\theta}) = K)$ write out invariant sets

else if $(\mathcal{K}(\tilde{\theta}) > K)$ $\theta^+ = \tilde{\theta}$ and **goto Step2**

else $\theta^- = \tilde{\theta}$ and **goto Step2**

I have found that the AIS algorithm works well on replacement matrices occurring in evolutionary modelling. However, because of the heuristics involved, the AIS algorithm is not guaranteed to find the optimal solution. In some cases, better conductance scores have been found by additionally editing the groupings by hand.

3.3 Results: application of AIS to Dayhoff and WAG matrices

The above algorithm has been implemented in a program called Almost Invariant Sets (AIS). The C code for finding amino acid groupings and for calculating the conductance measure is available at

`http://www.ebi.ac.uk/goldman-srv/AIS`.

I now apply the code to standard amino acid replacement matrices as they are widely used in practice. I start with the PAM1 matrix [DSO78]. The eigenvalues of the PAM1 matrix are given in the Figure 3.3. The spectrum of the PAM1 matrix (Fig. 3.3) does not exhibit a clearly identifiable cluster around the Perron root $\lambda_1 = 1$. Rather all 20 eigenvalues of the PAM1 matrix are close to 1.

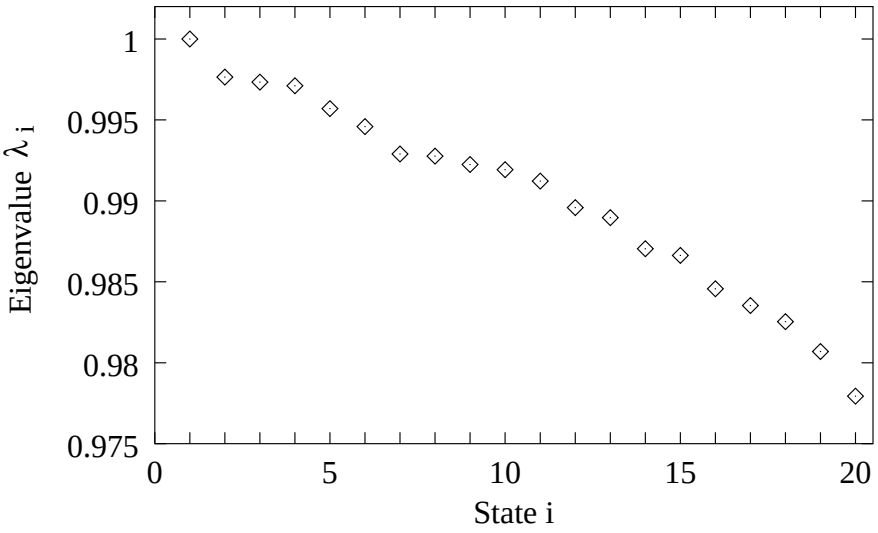


Figure 3.3: Eigenvalues of the PAM1 matrix.

I decided firstly to calculate a grouping into five amino acid sets since I could compare this grouping to one derived from the physicochemical properties of the

amino acids by Taylor [Tay86] (see also [FR83]), illustrated in Figure 3.4 and summarised as follows:

Hydrophilic:	A P G T S D E Q N
Basic:	K R H
Aromatic:	W Y F
Aliphatic:	M L I V
Sulphydryl:	C

The colours used in Figures 3.4, 3.5 and 3.7 follow the scheme proposed by Taylor [Tay97]. The use of this scheme may itself lead to new insights into protein sequence evolution [Tay97], but this approach has not been further pursued here.

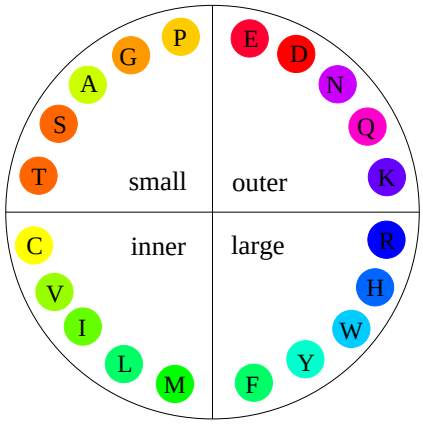
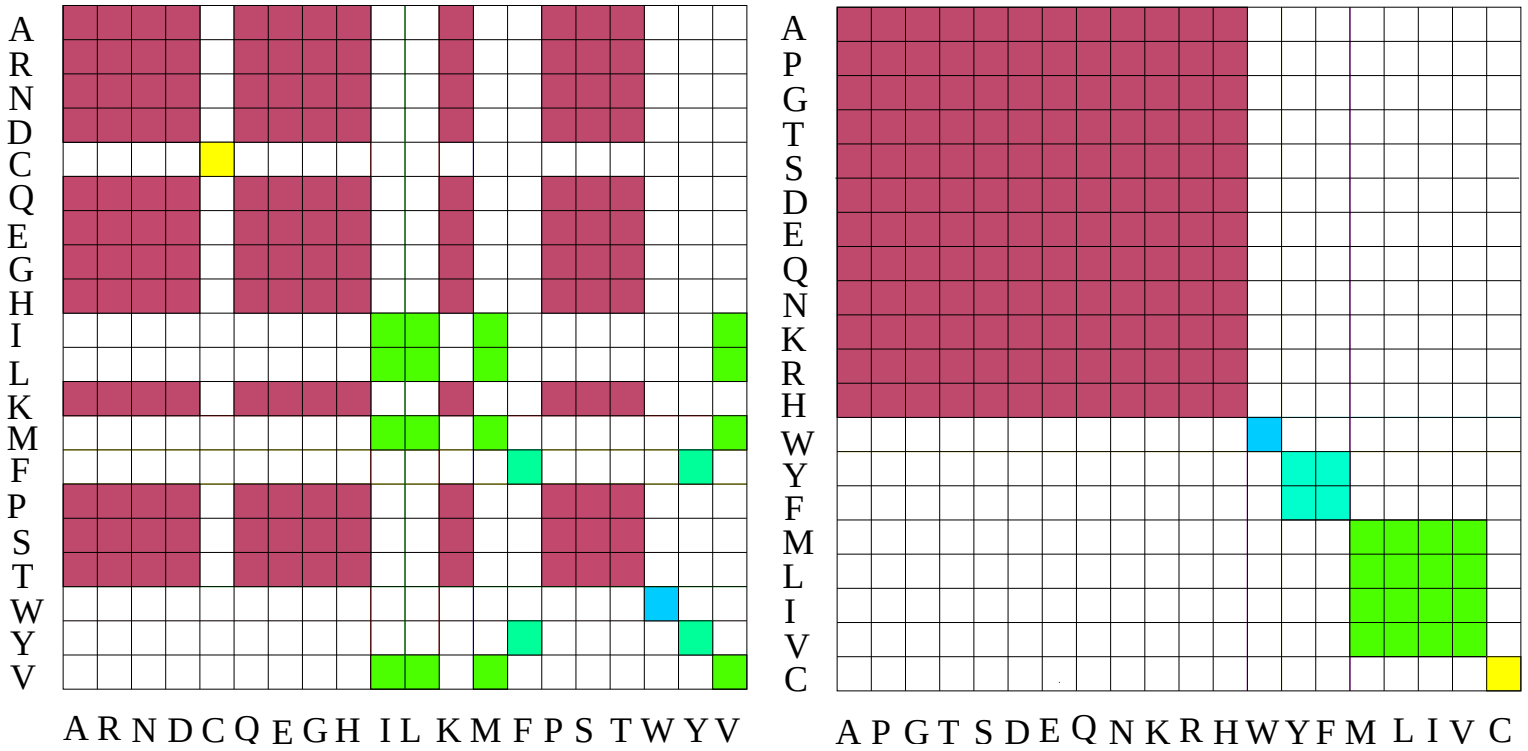


Figure 3.4: Representation of the PAM matrix. This projection of the matrix by multidimensional scaling is an idealisation adapted from Robson and French [FR83] by Taylor [Tay86]. The vertical axis of the circle corresponds to hydrophobicity, and consequently to whether the amino acid is mostly found in the inner or outer parts of proteins, and the horizontal axis corresponds to the molecular volume (small or large) of the amino acid. Amino acids that are close together exchange frequently. Colours used are those proposed by Taylor [Tay97].



(a) Hidden Block Structure of the PAM matrix.

(b) Sorted PAM matrix.

Figure 3.5: Application of the AIS algorithm to PAM. Amino acid colours are combined for each group according to the scheme of Taylor [Tay97].

The algorithm identified the five blocks for the PAM matrix shown in Figure 3.5(a). After reordering of the amino acids the inferred almost block diagonal structure of the PAM matrix is clearly visible. I read out the grouping from the ordered PAM matrix (Fig. 3.5(b)) as follows:

$$\{\text{A P G T S D E Q N K R H}\}\ \{\text{W}\}\ \{\text{Y F}\}\ \{\text{M L I V}\}\ \{\text{C}\}$$

The groupings derived from the physicochemical properties and using the AIS algorithm show similarities: only direct neighbours in the circle of amino acids of Figure 3.4 constitute groups. The sets {M L I V} and {C} are identical. The hydrophilic group {A P G T S D E Q N} and the basic group {K R H} are merged by the AIS algorithm into one set. Phenylalanine (F) and tyrosine (Y) remain members of the same set, but according to the AIS algorithm tryptophan (W) is not a member of this aromatic group and forms its own group {W}. Tryptophan is known to show unique behaviour. To compare these groupings quantitatively I calculate the conductance measure for both:

$$\varphi_{\text{AIS algorithm}} = 1.306 < \varphi_{\text{physicochem}} = 1.738$$

and thus the grouping that was found by the algorithm outperforms the grouping suggested by physical and chemical properties of the amino acids.

By definition (Eq. 3.1 and Eq. 3.2) the conductance score is non-decreasing as the number of subsets increases. Moving on from the division into five subsets, the best partitions between 1 and 20 subsets have been calculated and are given in Figure 3.7(a). Figure 3.6 shows how the conductance measure φ increases with the number of sets. The conductance measure grows moderately for a grouping into n=1–4 sets. The growth then changes to a rapid rise for divisions into n=5–15 groups, slows down for n=16–17 groups and finally grows rapidly again for

n=18–20. Overall the conductance measure increases strictly monotonically and no local extrema or plateaus can be observed.

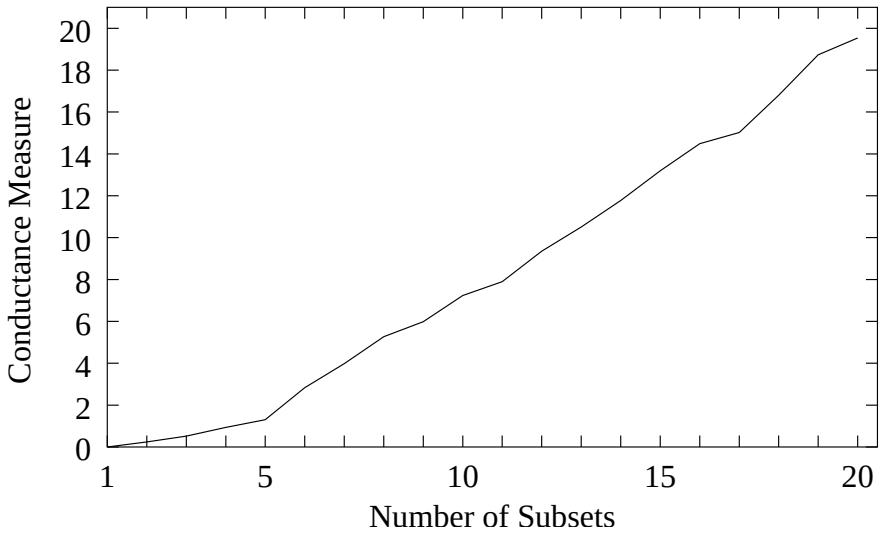
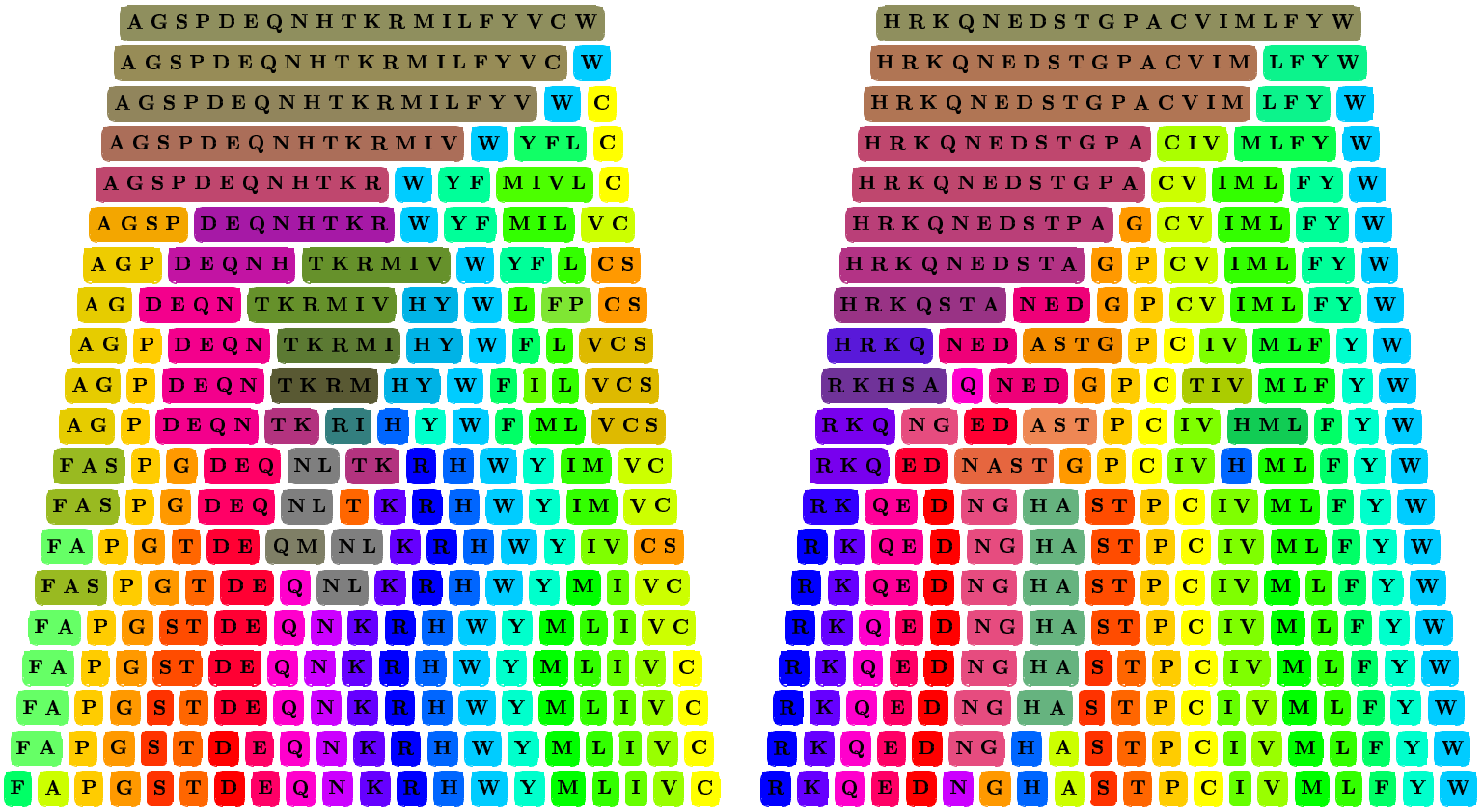


Figure 3.6: The conductance measure for 20 best groupings according to the PAM matrix.

I have also applied the AIS algorithm to the WAG matrix of Whelan and Goldman [WG01]. In Figure 3.7(b) I present the partitions of between 1–20 subsets found by the AIS algorithm. The 20 best groupings of the PAM and the WAG matrix are clearly distinguishable. For example, the most conserved group of WAG is {L M F Y} (along with its subgroups {L M F} and {L M}). In contrast, the set {C S V} (and {C V}) is the most stable among the groupings of the PAM matrix. Generally in the case of the PAM matrix new sets evolve by splitting up the previous sets. Among the groupings according to the WAG matrix swaps between sets can frequently be observed in addition to simple splits.



(a) The 20 best groupings according to the PAM matrix.

(b) The 20 best groupings according to the WAG matrix.

Figure 3.7: Complete results of the AIS algorithm for the PAM and the WAG matrices.

3.4 Extension of AIS to codon models

The AIS method is not restricted to amino acid models with a 20 character alphabet, but can be extended to any alphabet. In particular, it is interesting to apply AIS to codon models with a 64 alphabet (61 if stop codons are discarded).

The mathematical derivation of the AIS method in previous sections can be directly transferred by choosing 61 as the size of the state space. Furthermore, the program only needs minor adaption of the output routines. I have applied the AIS method to the standard codon model M0 (see section 2.4 and [YNGP00]). Parameter values are chose realistically, $\omega = 0.2$ (purifying selection) and $\kappa = 2.5$, as they can be found in a typical protein family of the Pandit database [WdBQ+06]. The frequency of codon i (triplet $i_1 i_2 i_3$) is calculated by the product $\pi_{i_1}^{(1)} \pi_{i_2}^{(2)} \pi_{i_3}^{(3)}$, where $\pi_{i_k}^{(k)}$ is the observed frequency of nucleotide i_k at codon position k:

$$\begin{aligned} &\pi_A^{(1)} = 0.221477, \pi_C^{(1)} = 0.187919, \pi_G^{(1)} = 0.392617, \pi_T^{(1)} = 0.197987, \\ &\pi_A^{(2)} = 0.285235, \pi_C^{(2)} = 0.276846, \pi_G^{(2)} = 0.145973, \pi_T^{(2)} = 0.291946, \\ &\pi_A^{(3)} = 0.216443, \pi_C^{(3)} = 0.266779, \pi_G^{(3)} = 0.182886, \pi_T^{(3)} = 0.333893. \end{aligned} \tag{3.4}$$

First I investigate a division into 20 groups; the codons as well as their amino acid translation are given:

{TGG(W)} {TAC(Y) TAT(Y)}
{TTC(F) TTT(F) CTC(L) CTT(L)} {CTA(L) CTG(L) TTA(L) TTG(L)}
{ATG(M)} {ATA(I) ATC(I) ATT(I)} {GTA(V) GTC (V) GTG(V) GTT(V)}
{TGC(C) TGT(C)} {ACA(T) ACC(T) ACG(T) ACT(T)}
{TCA(S) TCC(S) TCG(S) TCT(S)} {AGC(S) AGT(S) AGA(R) AGG(R)}
{GCA(A) GCG(A) GCC(A) GCT(A)} {GGA(G) GGC(G) GGG(G) GGT(G)}
{CCA(P) CCC(P) CCG(P) CCT(P)} {GAA(E) GAC(D) GAG(E) GAT(D)}
{AAC(N) AAT(N) } {CAA(Q) CAG(Q)} {AAA(K) AAG(K)}
{CGA(R) CGG(R) CGC(R) CGT(R)} {CAC(H) CAT(H)}

Often synonymous codons are grouped together (e.g., {TAC(Y) TAT(Y)}, {TGC(C) TGT(C)} and {ACA(T) ACC(T) ACG(T) ACT(T)}). However, sometimes the genetic code is not recovered (e.g.,{TTC(F) TTT(F) CTC(L) CTT(L)} {GAA(E) GAC(D) GAG(E) GAT(D)}).

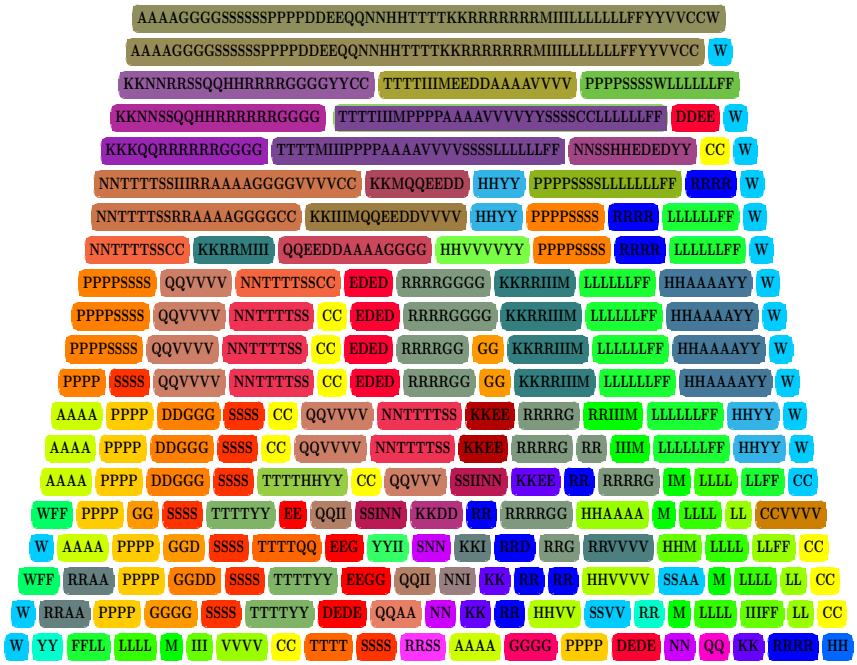


Figure 3.8: Results of the AIS algorithm for 20 groupings for the codon model M0. The results for 20 subsets was inspected by hand, and a better conductance score was found than was achieved by the algorithm. Codons are given by the amino acid they encode.

Codons can be grouped into 1–61 groups. For space reasons I only show groupings for M0 for 1–20 subsets (Fig. 3.8). Instead of listing codons I have listed amino acids, which also makes the pyramid of groupings directly comparable to results from the PAM and WAG groupings in Fig. 3.7 (a) and (b). Apart from the affiliation to an amino acid the codon model does not seem to reflect any of the physicochemical properties detected in the PAM and WAG models. Although the genetic code is somewhat represented here, the results will be seen later to be quite different from other, superior codon models (see section 5.3). This puts into question whether M0 is a good model for codon evolution.

3.5 Discussions and conclusions: AIS method

The conductance measure and the grouping algorithm have been proven useful in finding disjoint sets of amino acids. However, the criterion and the method only enable us to find the best grouping for a particular given number of subsets. No decision on the best number of subsets can be made, since neither the clustering of the eigenvalues around the Perron root $\lambda_1 = 1$ nor the graph of conductance measure as function of the number of subsets allow us to choose in a sensible way. To make progress here it might be necessary to modify the definition of the conductance measure (e.g., making it also a function of the number of subsets).

The actual groupings found from PAM for a particular number of subsets also appear reasonable from a biochemical point of view, as the comparison with the grouping of Taylor [Tay86] into five subsets shows. The advantage of the AIS approach is that the algorithm automates the process of finding groupings and that the conductance allows a quantitative assessment of the partition in a biologically meaningful way. The grouping algorithm identifies sets of amino acids with a high probability of substitution between amino acids of the same

set but small probabilities of change between different sets. The conductance measure quantifies the evolutionary changes between subsets that are of most interest. Furthermore, if the analysis is based on the normalised rate matrix of a Markov model, it is possible to directly compare the results of different models.

The analysis of the WAG matrix and the PAM matrix indicates that different amino acid replacement matrices lead like fingerprints to different groupings. In the future we will therefore use groupings and their scores according to the conductance measure as a tool to analyse and compare various Markov models of protein sequence evolution. I have also extended the AIS method to larger 61×61 matrices of codon models. This will be useful in a later chapter of this thesis.

Chapter 4

Analysing the relationship between amino acid and codon models: aggregated Markov processes

4.1 Non-Markovian behaviour

In 1972, Dayhoff and colleagues [DEP72] introduced a first Markov model of protein sequence evolution that resulted in the development of the widely used amino acid replacement matrices known as the PAM matrices. Since Dayhoff's PAM matrix, there have been increasingly good descriptions of the average patterns and processes of evolution of large collections of protein sequences (e.g., Jones *et al.* [JTT92]; Whelan and Goldman [WG01]), as well as more and more specialised matrices considering functional and structural properties of proteins (e.g., Adachi and Hasegawa [AH96]; Goldman *et al.* [GTJ98]).

However, whilst most work in phylogenetic modelling and statistics is aimed at devising improved Markov models, some criticisms have been directed at the models' Markov nature itself. A Markov model is defined by the assumption that each amino acid evolves independently of its past history. The instantaneous rate

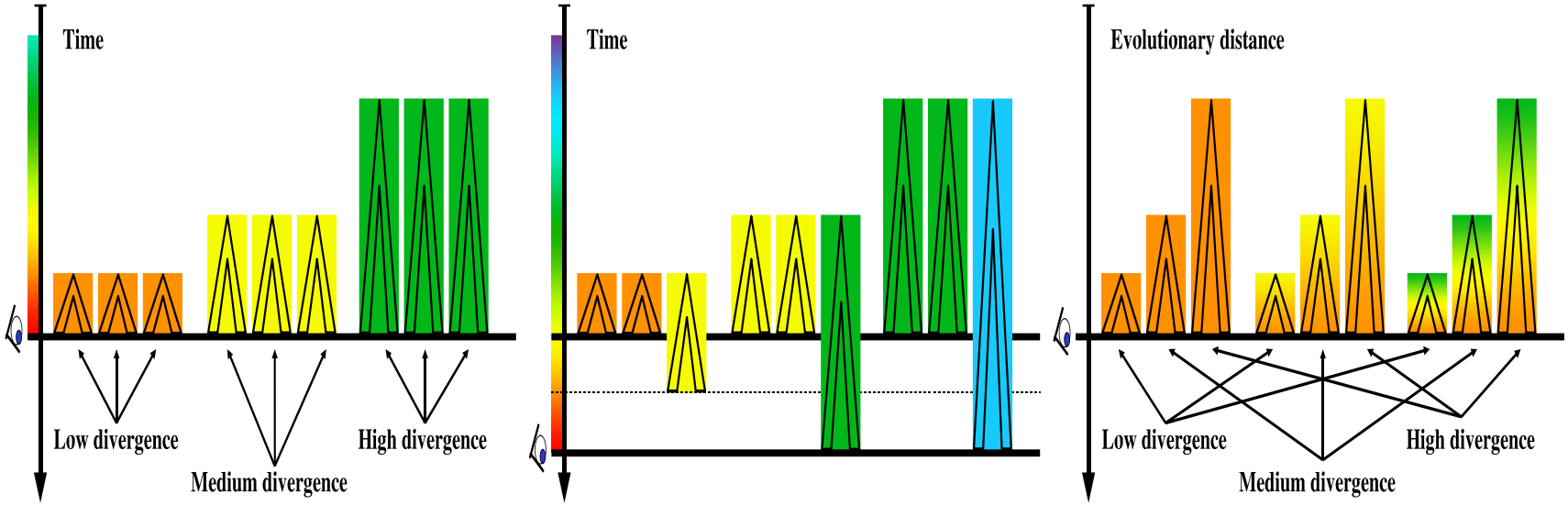
matrix, which represents the patterns of the substitution process and specifies a Markov model completely, is the same at any time.

Suppose after a speciation event a pair of sequences evolve under a Markov model from a common ancestor. After some time we can measure the divergence level between the two sequences, and because the model is Markov, the sequences will continue to evolve according to the same process leading to a higher divergence level. Thus, the Markov model implies that the patterns and dynamics of substitutions are similar at low and high sequence divergence.

Henikoff and Henikoff [HH92] derived a series of BLOSUM matrices, which are still probability matrices but are not based on a Markov model. Instead, every possible amino acid replacement is counted between conserved sub-blocks of aligned protein sequences from many different protein families. Sub-blocks were made by single linkage clustering about percentage identity threshold, and different matrices were obtained by varying the clustering threshold. The matrices of the BLOSUM series are identified by a number after the matrix (e.g., BLOSUM62) which refers to the percentage identity of the sub-blocks of multiple aligned amino acids used to construct the matrix. Because the BLOSUM matrices are not based on an evolutionary model, it is not possible to calculate BLOSUM matrices at different evolutionary distances simply by extrapolating. BLOSUM matrices often perform better than PAM matrices for the purpose of amino acid sequence alignment or database searching. It is unclear why this should be, but it may be because protein sequences behave in a non-Markov manner. Mitchison and Durbin [MD95] tried to find an exponential family which would generate a series of matrices they had estimated empirically from sequences that diverged at different times, but failed. Furthermore, Benner *et al.* [BCG94] inferred replacement matrices from different sets of sequences separated by simi-

lar divergent levels and found qualitative differences in the substitution patterns (see also section 4.4, especially Figs. 4.6 below). They concluded that the evolutionary process changed as function of sequence divergence and that in reality, the assumption that high divergence can be modelled by extrapolating the patterns of low sequence divergence does not hold.

Benner and colleagues perhaps formulated the most detailed criticism on the Markov model. Fig. 4.1(a) illustrates their approach. Pairs of protein sequences are symbolised by a triangle-shaped 'tree'. Sequence pairs are grouped together if their proteins are separated by the same divergence level. Benner *et al.*'s conclusion that the evolutionary process is different for different divergence levels is indicated by colouring the sequences pairs orange for low, yellow for medium and green for high divergence levels and by linking the time axis with a colour scale distinguishing the different types of evolutionary processes. The link between the time and the character of the process is surprising, since one would expect that the time of observation has no influence on amino acid substitutions. Fig. 4.1(b) illustrates this problem with Benner and colleagues' approach. Suppose that a pair of sequences is separated by a low divergence level, that it evolves according to the orange process and that it will be observed later in the future. According to Benner, at some time point in the future these sequences will have evolved under a different process, the yellow process, because they are then separated by medium divergence. Likewise, a pair of medium diverged sequences (yellow process), observed at some later time in the future, will look like a pair of sequences evolved according to the green process and a pair evolved under the green process will look like a pair evolving under the blue process. But when does the sequence pair switch process? And how can an amino acid in a sequence know what time/process applies, or when it will be observed?



(a) Benner and colleagues' approach I.

(b) Benner and colleagues' approach II.

(c) Modelling the 'average' process with a time-homogenous approach.

Figure 4.1: Observing evolutionary processes at different time points.

In contrast, the 'time-homogenous' approach (Fig. 4.1(c)) assumes that individual sequences can evolve either according to one process (orange process) or according to a combination of several different processes (illustrated as the yellow and the orange or the green, yellow and orange processes, etc.). The change of evolutionary process can be caused, for example, if the environment of a species changes resulting in changed selective pressures on specific sites of the protein. Clearly, the combination of processes is specific to each protein and is not linked to a specific time or a specific level of divergence from any other protein. However, protein sequences might evolve with a similar combination of processes, illustrated for example with a combination of yellow and orange or a combination of green, yellow and orange processes. The combined processes can happen at a variety of different evolutionary rates leading to different branch lengths measured in evolutionary distance. By taking lots of proteins, a whole database of proteins, the one hopes to find an 'average' combined process covering the substitution patterns common to all the proteins. A mixture of Markov processes is not Markov, and the 'average' combined process would be not Markov even if the component processes were. However, this thought experiment indicates that, contrary to Benner approach the 'average' process can be assumed to be time-homogeneous and should be the same if estimated from enough sequences with low divergence, medium divergence or high divergence levels, and irrespective of whether it is estimated in 1994, in 2006, or in one million or one hundred million years' time.

The 'time-homogeneous' approach is logically consistent, but it is unable to explain the experimental results of Benner *et al.*, and Mitchison and Durbin, and the success of the BLOSUM matrices. Accepting Benner *et al.*'s observations but not necessarily their conclusion, I investigate if other factors could have caused the

differences in inferred evolutionary dynamics depending on observed divergence levels.

In this chapter I use aggregated Markov processes [Lar98] to model evolution as Markovian at the codon level but observed (via the genetic code) only at the amino acid level. This approach is interesting, because all previous studies of non-Markovian behaviour were done on the amino acid level only and often with inference techniques which are less advanced than the ones now available.

Evolution, however, occurs on the DNA level (or codon triplets if looking at coding DNA). Furthermore, codon-level models have been tested over years [GY94, YNGP00], such that we can hope that we have adequate codon models to base this study on.

In the following sections I simulate data from aggregated Markov processes and I study the consequences of their non-Markovian, but time-homogenous behaviour using the Chapman-Kolmogorov equation [Gil92]. Finally I investigate how far the aggregated Markov process can explain the experimental results observed by Benner *et al.* [BCG94] and Mitchison and Durbin [MD95].

4.2 What is an aggregated Markov process?

Protein sequence evolution can be considered both on an amino acid and on a codon level. Suppose, for example, that between times t_k and t_{k+1} the following substitution has occurred in a coding region:

Amino acids $Y(t_k)$	I	C	I	K	V	L	L	T
Codons $X(t_k)$	ATA	TGT	ATA	AAG	GTC	CTT	TTA	ACA
						↓		
Codons $X(t_{k+1})$	ATA	TGT	ATA	AAG	GTC	CCT	TTA	ACA
Amino acids $Y(t_{k+1})$	I	C	I	K	V	P	L	T

Now assume that substitutions, like the above, are described by a continuous Markov process $\{X(t), t \geq 0\}$ on the codon level with state space $\mathcal{S} = \{AAA, AAC, \ldots, TTG, TTT\}$, equilibrium distribution π and probability matrix $P(t) = e^{Qt}$. For the universal code, which has 61 sense codons, Q is a 61×61 instantaneous rate matrix.

Further, suppose that the codons are not directly observable, but that a deterministic function of the underlying Markov process, $Y(t) = f(X(t))$, which maps the state space to the aggregate set $\mathcal{A} = \{A, R, N, \ldots, V\}$, can be observed. Clearly, we are considering observing the amino acids encoded by the codons, with this function f defined by the universal genetic code. The observable process of amino acids $\{Y(t), t \geq 0\}$ is then called an aggregated Markov process (AMP) [Lar98]. The dependence structure for the highlighted site in the example above is represented by the following graph:

$$\begin{array}{ccccccc} \ldots & \rightarrow & X(t_k) = CTT & \rightarrow & X(t_{k+1}) = CCT & \rightarrow & \ldots \\ & & \downarrow f & & \downarrow f & & \downarrow \text{Genetic code} \\ \ldots & & Y(t_k) = L & & Y(t_{k+1}) = P & \ldots & \text{Non-Markov Process} \end{array} \quad \begin{array}{c} \\ \text{Markov process} \\ \\ \end{array}$$

Given only the amino acid level $Y(t)$, it is impossible to tell whether the substitution of leucine (L) with proline (P) was caused by a substitution from CTT to CCT, from CTC to CCC, from CTA to CCA or from CTG to CCG. (We assume only single nucleotide changes; a more general model might permit double and triple nucleotide changes instantaneously (e.g., CTT → CCA) which results in an even more varied dependence structure.) Consequently, the probability of a change to proline (P) does not only depend on the present amino acid leucine (L), but also on the hidden state $X(t_k)$. Therfore the stochastic process $Y(t)$ describing the amino acid evolution is non-Markov. This can be understood more generally: aggregated Markov processes are a subclass of hidden Markov models (HMM); more general HMMs allow the observed $Y(t)$ to be probabilis-

tically determined given $X(t)$. The theory of HMMs says that the stochastic process $X(t)$ on the state space is Markov, but the observable process Y(t) is non-Markov [Kue01]. Hence, the theory of HMMs can explain why the observable process acting on the amino acids is non-Markov. Can it also explain some of the observations from experimental data?

4.3 Understanding the behaviour of AMPs: Chapman-Kolmogorov

In this section I will show that assuming Markovian behaviour for observations (e.g., amino acid changes) generated with an AMP can lead to substantial error in the estimation of substitution probabilities.

Suppose we want to calculate the probability of a change from amino acid cysteine (C) at time t_0 to methionine (M) at time t_1, but the data we have only allows us to determine the amino acid substitution probabilities for the time between t_0 and some intermediate time τ, and between τ and the final time t_1. If the process is Markov we can still calculate this probability by applying the Chapman-Kolmogorov equation:

$$P(\mathrm{M}, t_1|\mathrm{C}, t_0) = \sum_{i=1}^{20} P(\mathrm{M}, t_1|A_i, \tau) \times P(A_i, \tau|\mathrm{C}, t_0) \quad \text{for any } \tau \in [t_0, t_1], \quad (4.1)$$

where A_i represents amino acid i (see also Fig. 4.2). Conversely, if the Chapman-Kolmogorov equation does not hold we know that the examined process behaves in a non-Markovian manner [Gil92].

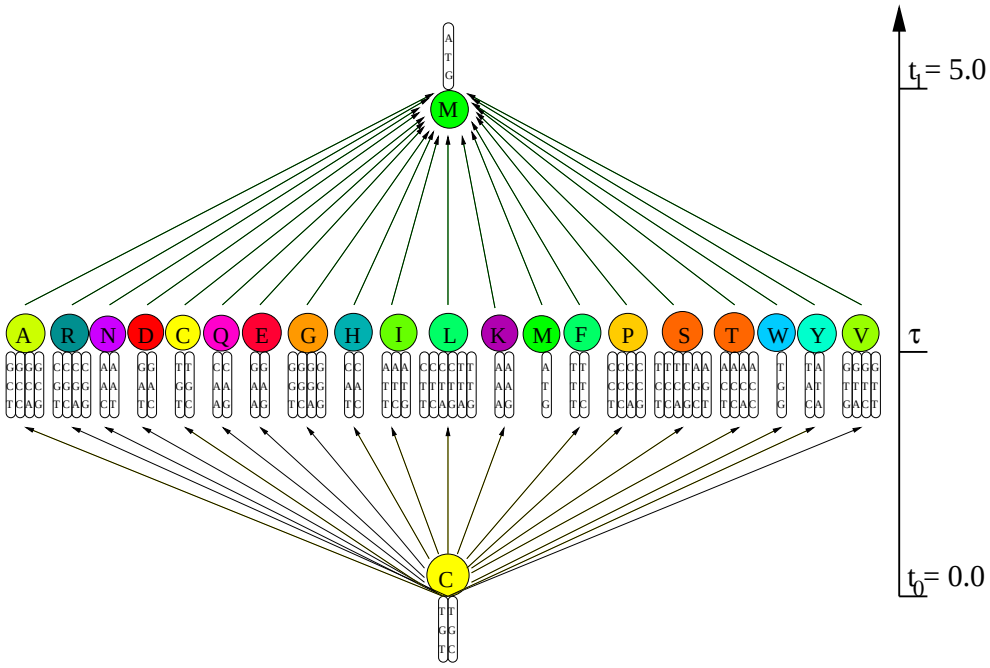


Figure 4.2: Illustration of the Chapman-Kolmogorov equation: If the Markov property holds, the probability $P(\mathrm{M}, t_1 = 5.0|\mathrm{C}, t_0 = 0.0)$ of a change from cysteine (C) at time $t_0 = 0.0$ to methionine (M) at time $t_1 = 5.0$ can be calculated as the sum of the probabilities of this occurring via all possible amino acids at any intermediate fixed time τ.

In order to investigate violations of the Chapman-Kolmogorov equation for protein-coding sequence AMP, I have written a C program to calculate the probabilities given in equation 4.1.

As codon model I choose the M0 model, which is often used to detect selection in a protein [GY94] and whose equation (2.5) was already introduced in the introductory chapter 2. The elements of Q are defined as:

$$q_{ij,i\neq j} = \begin{cases} 0 & \text{if } i \text{ or } j \text{ is a stop codon or requires} > 1 \text{ nucleotide substitution} \\ \pi_j & \text{if } i \rightarrow j \text{ synonymous transversion} \\ \pi_j\kappa & \text{if } i \rightarrow j \text{ synonymous transition} \\ \pi_j\omega & \text{if } i \rightarrow j \text{ nonsynonymous transversion} \\ \pi_j\kappa\omega & \text{if } i \rightarrow j \text{ nonsynonymous transition} \end{cases}$$

where κ is the transition-transversion rate ratio, ω is the nonsynonymous-

synonymous rate ratio, and π_j is the equilibrium frequency of codon j.

On the codon level the process is Markov by construction (see section 4.2) and the Chapman-Kolmogorov equation formulated on codons holds. Let A_C be the set of codons coding for C and A_M be the set of codons coding for M. To be able to compare codon results with amino acid results we look at $t_0 = 0.0$ at starting distribution for the codons of cysteine (C) defined by

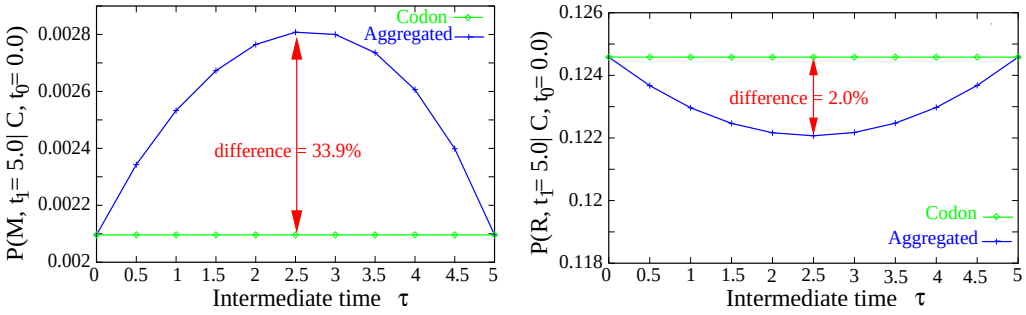
$$[v_0]_i = \begin{cases} \frac{\pi_i}{\sum_{j=1}^{k} \pi_j} & i \in A_C \\ 0 & \text{otherwise.} \end{cases}$$

This starting vector is then evolved with $P = e^{tQ}$ over some time t. For example, at time $t_1 = 5.0$ the state vector is $v_0 P(t_1 - t_0)$ and the probability to be in state M at time $t_1 = 5.0$ is given as:

$$P(\mathrm{M}, t_1 | \mathrm{C}, t_0) = \sum_{k \in A_M} (v_0 P(t_1 - t_0))_k \quad . \tag{4.2}$$

I have calculated the righthand-side of the Chapman-Kolmogorov equation (4.1) for intermediate times, $\tau \in [0.0, 5.0]$, for AMPs ('Aggregated') and the pure Markov codon model ('Codon'). The results for a change from cysteine to methionine (C $\rightarrow$ M) and for a change from cysteine to arginine (C $\rightarrow$ R) are shown in Figure 4.3. Unlike the results of the Markovian codon process, the graphs of the AMP are not constant. This means that the probabilities of amino acid substitution depend on the intermediate time τ when the amino acid sequences are observed. The effect is particularly strong if the amino acids are distant in the genetic code (e.g., they are 2 or 3 nucleotide changes apart).

This brings us back to Benner's claim that the time at which the evolutionary process is observed, is relevant for the estimation of the substitution process. The Chapman-Kolmogorov equation illustrates that for AMPs observation times do matter.



(a) Probability of a change from cysteine (C) to methionine (M) via an intermediate step at time τ. Cysteine and methionine are distant in the genetic code (3 base changes).

(b) Probability of a change from cysteine (C) to arginine (R) via an intermediate step at time τ. Cysteine and arginine are close in the genetic code (1 base change).

Figure 4.3: Dependency of the probability on the time τ of the intermediate step.

4.4 Comparison to experimental results

The important role of the genetic code has been suggested before: Benner and colleagues' experiments [BCG94] indicated that the genetic code influences accepted point mutations strongly at early stages, while the chemical properties of the side chains dominate at later stages after divergence. They concluded that the patterns of substitutions are not similar at low and high sequences divergence and that high divergence can not be modelled by extrapolating the patterns of low sequence divergence. Accepting Benner *et al.*'s observations but not their conclusion, I investigate if the *aggregation* of the codon sequence to an amino acid sequence according to the genetic code could have caused the behaviour observed.

Benner *et al.* choose to compare log-odds matrices L as these are often used as scoring matrices in alignment programs. Positive scores in a log-odds matrix designate a pair of residues that replace each other more often than expected by chance. Negative scores in a log-odds matrix designate pairs of residues that

replace each other less than would be expected by chance.

Below, I will adopt Benner's procedure to estimate log-odds matrices. I use the same symbols Q, π and P for both amino acid and codon. Their subscripts, however, indicate whether they refer to codons (i and j) or to amino acids (x and y). Benner and colleagues split their aligned amino acid sequence pairs into 10 sets based on their divergence level. The divergence level is measured by PAM distance, and the average PAM distance in each set is denoted by t_k, $k = 1, \ldots, 10$. They then compiled a matrix of counts T, where $[T(t_k)]_{xy}$ is the number of substitutions from amino acid x to amino acid y in a given set of sequences, and a diagonal matrix N with $[N(t_k)]_{xx}$ the total number of amino acids. From these they estimated amino acid substitution matrices using the formula

$$P(t_k) \approx T(t_k) \cdot N(t_k)^{-1} \quad .$$

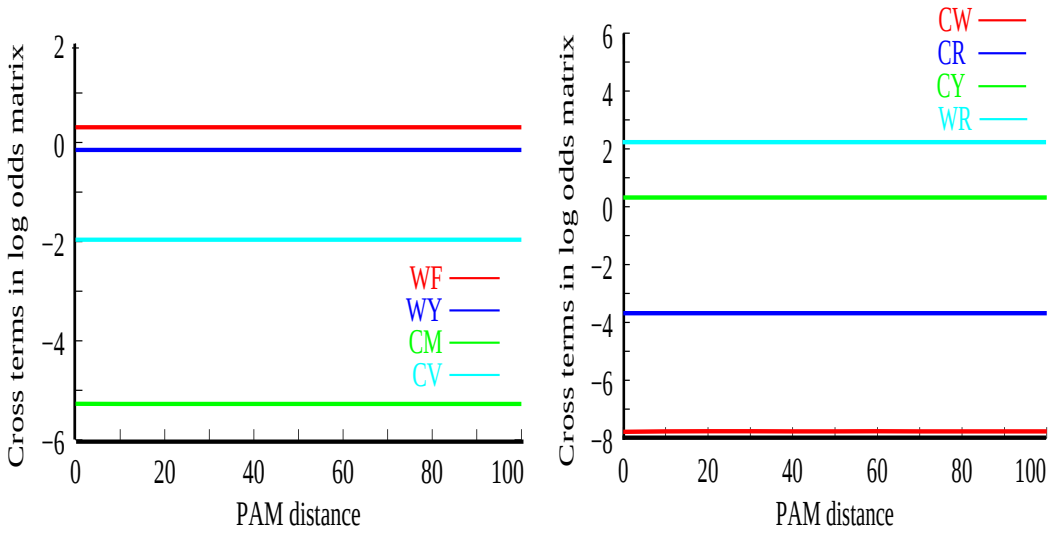
Like Benner, I extrapolate these matrices to a divergence of 1 substitution expected per site:

$$P(1) = P(t = 1) = ((P(t_k))^{\frac{1}{t_k}} = \approx (T(t_k) \cdot N(t_k)^{-1})^{\frac{1}{t_k}}$$

and convert the probability matrix to a 250 log-odds matrix:

$$[L(250)]_{xy} = 10 \log_{10} \frac{[P(250)]_{xy}}{f_x} = 10 \log_{10} \frac{[P(1)^{250}]_{xy}}{f_x} \quad ,$$

where f_x $(= \frac{[N(t_k)]_{xx}}{\sum_y [N(t_k)]_{yy}})$ is the frequency of amino acid x in the each data set. The extrapolated matrices $L(250)$ obtained from the 10 data sets should be the same, if the underlying process of the observed protein evolution was Markov. To check this, I have used Dayhoff's amino acid model to simulate perfectly Markov data in the form of pairwise alignments at different divergent levels t_k. Then I applied the Benner inference procedure described above to calculate the log-odds

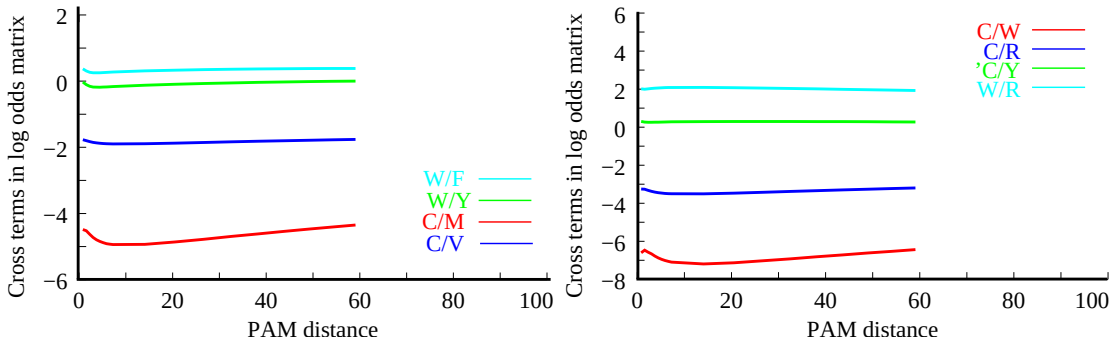


(a) (b)

Figure 4.4: Graphs of some off-diagonal elements of the 250 PAM log-odds matrix versus amino acid divergence times. The matrices are estimated from data simulated under Dayhoff's amino acid substitution model [DSO78, KG05]. For example, the line labelled WF represents the value of $[L(250)]_{WF}$ computed from data simulated at divergence levels from 0 to 100 PAMs.

matrices. Fig. 4.4 confirms that for perfect Markov data the elements of the log-odds matrix are not dependent on the divergence level t_k.

I also generated data from a mixture process acting on amino acids, a methodology was not available to Benner *et al.* [BCG94] at the time of their analysis. As a mixture process I chose a discrete gamma distribution of rates of evolution over amino acid sites which is determined by the parameter α (see section 2.5 and [Yan94a]). To simulate data, 182 experimental values of α representing the distribution of αs for typical globular proteins were used. These were taken from Goldman and Whelan [GW02]. However, although a little curvature could be observed (Fig. 4.5) the effect was not as severe as observed by Benner *et al.*



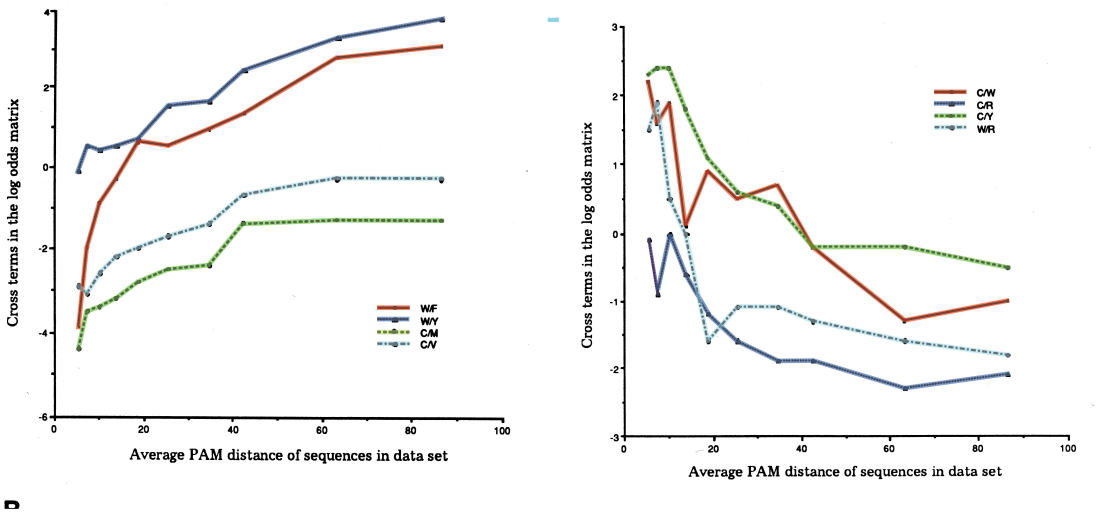
(a) Cross terms of some amino acid pairs similar in chemical properties but distant in the genetic code.

(b) Cross terms of some amino acid pairs different in chemical properties but close in the genetic code.

Figure 4.5: Markov mixture process on the amino acid level. Graphs of some off-diagonal elements of the 250 PAM log-odds matrix versus amino acid divergence times. The elements presented are identical to the the ones proposed by Benner [BCG94].

In comparison, Benner's graphs (Fig. 4.6) derived from experimental data are not flat, but have considerable curvature. Picking out particular elements of the log-odds matrix, the graphs show that the log-odds matrix is a function of the PAM distance. The interpretation given by Benner *et al.* is that the genetic code influences the matrix strongly at early stages of divergence, while physicochemical properties are dominant at later stages. For example, the element (C→W) indicates that substitutions are frequent at small PAM distances, the interpretation being that only a single base change is necessary. At larger PAM distances substitutions are infrequent, perhaps because the side chain of tryptophan (W) is large and hydrophobic while the side chain of cysteine (C) is small and can form disulfide bonds inaccessible to tryptophan (W).

I will now use simulated codon data, to investigate if the effect of aggregation could lead to observations similar to Benner *et al.* I have calculated $P(t_k^*) = e^{Qt_k^*}$ (see section 2.2) using again the instantaneous rate matrix of M0 for 20 different



(a) Cross terms of some amino acid pairs similar in chemical properties but distant in the genetic code.

(b) Cross terms of some amino acid pairs different in chemical properties but close in the genetic code.

Figure 4.6: Graphs of some off-diagonal elements of the 250 PAM log-odds matrix versus amino acid divergence times taken from Benner [BCG94]. The graphs have been coloured to better distinguish them.

divergence codon times $0.1 = t_1^* < \cdots < t_k^* < \cdots < t_9^* = 0.9$ and $1.0 = t_{10}^* < \cdots < t_{20}^* = 10.0$. The frequency of observing codon i in one sequence and j in another is given by

$$\pi_i P_{ij}(t_k^*) \quad .$$

Letting X and Y represent the sets of codons that code for amino acids x and y respectively, then the frequency of observing amino acid x in one sequence and amino acid y in the other sequence is

$$\sum_{i \in X} \sum_{j \in Y} \pi_i P_{ij}(t_k^*) \quad . \tag{4.3}$$

To estimate substitution matrices I set

$$[T(t_k^*)]_{xy} = \sum_{i \in X} \sum_{j \in Y} \pi_i [P(t_k^*)]_{ij} \quad \text{and} \quad [N(t_k^*)]_{xx} = \sum_x [T(t_k^*)]_{xy} \quad ,$$

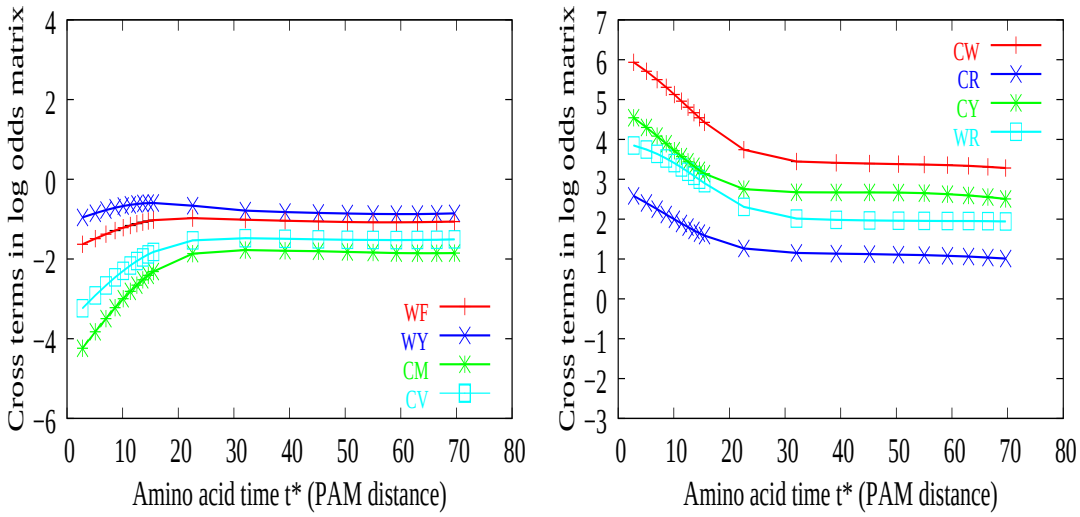
so that I can repeat Benner's analysis, but using 'perfect' simulation data. Furthermore, I do not use the codon times t_k^*, $k = 1, \ldots, 20$, to extrapolate the matrices to time $t = 1$, but amino acid times given as PAM distances. Benner and colleagues had to estimate PAM distances t_k, because they only had amino acid sequences, and it is important to mimic this because the amino acid time estimates t_k may be systematically and non-linearly biased relative to the codon times t_k^*. The phylogenetic software package PAML [Yan94b] can read site pattern frequencies like ones given in equation 4.3 instead of sequence data. I have estimated the divergence levels of the amino acid data $t_1 < \cdots < t_k < \cdots < t_{20}$ as PAM distances, as did Benner *et al.* Then I have calculated separate $P(1)$ matrices for one unit amino acid time and hence the corresponding $L(250)$.

Initially, I used a simple M0 model choosing realistic values for $\omega = 0.2$ (moderate purifying selection), $\kappa = 2.5$ and codon frequencies as specified in equation (3.5). The log-odds matrices showed some dependency on the PAM distance, but the magnitude of the effect on the log-odds elements did not reflect the strong variation of the log-odds of Benner's experimental data (see Fig. 4.6). Further trial studies choosing different values for ω and κ did not lead to qualitatively different results (not shown).

However, introducing a more realistic model allowing for among site rate variation by using a Γ model (see section 2.5) clearly compared better with plots of experimental data in Fig. 4.6. Finally, instead of determining the rate categories by a Γ distribution I have specified rate categories by hand. Best results were achieved by aggregation of simulated data from a mixture codon model with 12 rate categories:

$$\begin{array}{llll} r_1 = 0.00001, & r_2 = 0.0001, & r_3 = 0.0001, & r_4 = 0.001, \\ r_5 = 0.01, & r_6 = 0.1 & r_7 = 0.15, & r_8 = 0.2, \\ r_9 = 0.3, & r_{10} = 0.5, & r_{11} = 2.0, & r_{12} = 8.73889. \end{array} \tag{4.4}$$

All rate 12 categories have the same weight. Results are shown in Fig. 4.7 and should be compared with Benner's results on experimental data Fig. 4.6.



(a) Cross terms of some amino acid pairs similar in chemical properties but distant in the genetic code.

(b) Cross terms of some amino acid pairs different in chemical properties but close in the genetic code.

Figure 4.7: Graphs of some off-diagonal elements of the 250 PAM log-odds matrix versus amino acid divergence times. The elements presented are identical to the the ones proposed by Benner [BCG94]. The nonsynonymous-synonymous rate ratio ω is 0.2 (purifying selection).

Similar to the graphs of the experimental data, the graphs of the simulated data have some curvature. For the specific elements of the log-odds matrix plotted, the order and the trends of the graphs agree for experimental and simulated data. However, the ranges of the experimental graphs are different. For high PAM distances the simulated graphs converge to zero, while the experimental graphs actually often cross the zero line.

This shows that relatively complex, but not unrealistic, codon models can generate behaviour similar to what Benner *et al.* observed. However, the rate variation across sites needs to be large to capture the right behaviour. Since

gamma distributions and modified gamma distributions have not been widely used to model rate variation in codon models, it is hard to decide whether the values used in the simulation compare to values observed in nature. Instead, it would also be valuable to consider rate variation caused by variation in selection pressure. Models like M7 have been widely studied [YNGP00] and it would be very interesting to test whether a simulation using M7 parameters with parameter values taken from real data, could lead to 'Benner-shape' graphs.

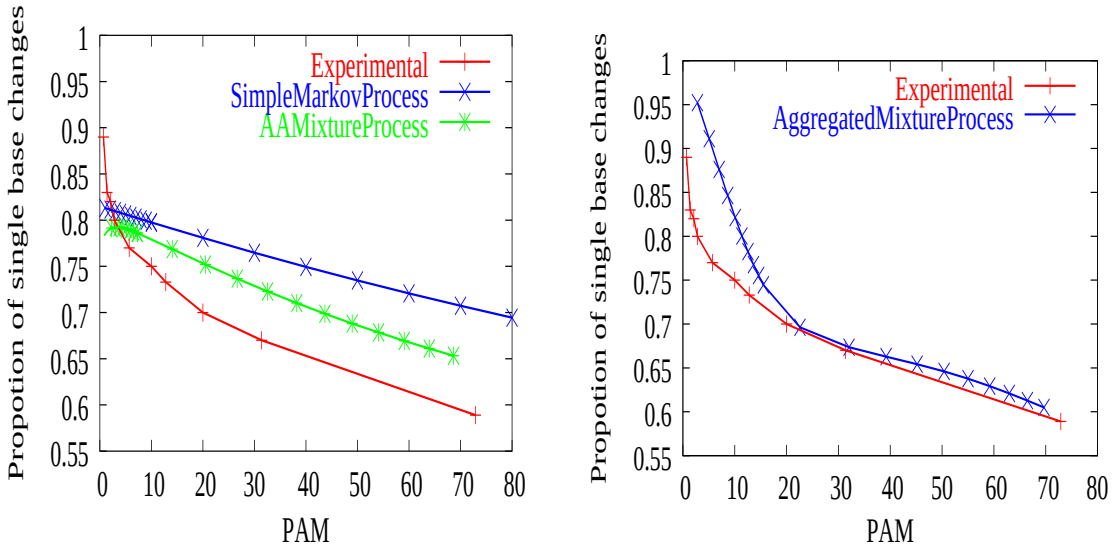
Mitchison and Durbin [MD95] estimate amino acid substitution matrices from experimental data using maximum likelihood methods. They then sum the matrix entries over all one-base-change substitutions and take the ratio of this to the sum of all non-diagonal entries:

$$\frac{\sum_{(i\to j)\in S} P_{ij}(t)}{\sum_i \sum_{j\neq i} P_{ij}(t)} \quad , \tag{4.5}$$

where S is the set of single base changes. This too is diagnostic of non-Markovian behaviour and since only single base changes are investigated, it also presumes relevance of the genetic code.

I repeated Mitchison and Durbin's analysis using again three different simulations. First I simulated data from Dayhoff's amino acid model as a simple Markovian amino acid model. This simulation indicates what could be observed if amino acid sequence evolution was Markov. Secondly, I use an amino acid mixture model as described before using 182 α-values taken from Goldman and Whelan [GW02]. Thirdly, I simulated data from an aggregated Markov process created from a codon mixture model with 12 rate categories as described in equation (4.4).

Finally, I compared the results of these three models to experimental data of Mitchison and Durbin. Figure 4.8(a) shows that a mixture process is closer



(a) Markov mixture process on the amino acid level.

(b) Aggregated process of a Markov mixture process on the codon level. We choose a discrete gamma distribution with 12 categories.

Figure 4.8: The influence of the genetic code.

to experimental data than a simple Markov process, but even for the mixture process the proportion of single base changes decreases fairly linearly. Thus a mixture of processes alone cannot explain the observations of Mitchison and Durbin. Figure 4.8(b) shows that although the combination of mixture process and *aggregation* does not reflect well the behaviour the experimental results for small times (PAM distances), it captures the rapid and non-linear decline of the proportion of single base changes.

4.5 Discussion: aggregated Markov processes

Since Dayhoff there have been increasingly good empirical models of the average patterns and processes of evolution of large collections of amino acid sequence, as well as more and more specialised matrices considering functional and structural

properties of proteins. However, whilst most work in phylogenetic modelling is aimed at devising improved empirical Markov models, some criticisms have been directed at the Markov nature of the models itself.

Several studies on experimental data (e.g., by Benner *et al.* [BCG94] and Mitchison and Durbin [MD95]) observe different substitution patterns at different levels of sequence divergence. These observation indicate that amino acid sequence evolution behaves in a non-Markovian manner.

In a thought experiment I show that Benner *et al.*'s explanation that the process of evolution is different for different divergent times is irrational, because the time of observation cannot have any influence on amino acid substitutions. However, Markov models are fundamental to many applications in evolutionary studies (see section 1.1) and we need to find some explanation for the observations.

I show that some of the non-Markovian behaviour described in the literature can be explained by an aggregated Markov process (AMP) which combines rate heterogeneity among different codon sites of the protein and the properties of the amino acids encoded by the sequence. I focus on a study of the consequences of non-Markovian behaviour using the Chapman-Kolmogorov equation [Gil92] and a comparison to studies on experimental data by Benner *et al.* [BCG94] and Mitchison and Durbin [MD95]. Results show that Benner's results cannot be explained by a pure Markov model or a realistic mixture Markov model on the amino acid level; in contrast, an aggregated Markov model combined with rate variation on the codon level leads to good results, although it does not incorporate any of the physicochemical properties considered by Benner *et al.* to be responsible for the shape of the graph. To check how realistic the rate variation on the codon level is, the simulation studies could in future be extended from M0 to M7-type models which consider variation in rate caused by variation in

selective pressure across sites.

Furthermore, the comparisons were limited since the original data of the above studies is not available anymore. A full study would require the re-creation of a suitable experimental data set. However, the results of the my simulation study using AMPs already strongly suggest that protein evolution should be modelled with codon rather than amino acid substitution models. Modelling protein sequence evolution on the amino acid level introduces a systematic error. The nature of protein-coding sequence evolution is such that modelling on the codon level seems reasonable, but leads to non-Markov behaviour on the amino acid level.

Recently, there have been suggestions to use mechanistic codon models not only for the detection of selection, but also for phylogenetic inference. Ren *et al.* [RTY05] study the utility of codon models for phylogenetic reconstruction and molecular dating. Codon models are reported to have good performance in both recent and deep divergences. Although their computational burden makes codon models currently infeasible for tree searching, Ren and colleagues recommend them for comparing pre-determined candidate trees. This illustrates that it is increasingly feasible to use codon models where amino acid models have been used in the past, and give further reasons why codon models are preferable.

Convinced by the necessity for accurate codon models, the next chapters are devoted to the development of an empirical codon model. We will also see that observations on the nature of codon sequence evolution supplement the results of studying the aggregated Markov process in an unexpected way.

Chapter 5

Derivation of empirical codon models

5.1 Mechanistic and empirical codon models

Two kinds of Markov models have been considered to describe protein sequence evolution in the past. Mechanistic models take into account features of protein evolution such as selective pressures and the frequency of character states in the data (e.g., relative occurrence of amino acids). Empirical models do not explicitly consider biological factors that shape protein evolution, but simply attempt to summarise the substitution patterns observed from large quantities of data. In empirical models, the substitution patterns are described by parameters, the rates of replacements between amino acids or codons, that have more statistical convenience than physical meaning. These rates of replacement are an aggregated measure of the physicochemical properties of the amino acids or codons and of their interaction with their local environment. Often empirical codon models have many such parameters and they are typically estimated once from a large dataset and subsequently re-used with the assumption that these parameters are applicable to a wide range of sequence data sets (see also section 2.4. On the other hand, mechanistic models explicitly describe the factors of the evolutionary

process, allowing the testing of hypothesis related to these factors for each data set of interest. Often there is only a relatively small number of parameters used and these are not assumed too be 'constants', but are estimated afresh for each data set. Many evolutionary models contain a combination of mechanistic and empirical parameters.

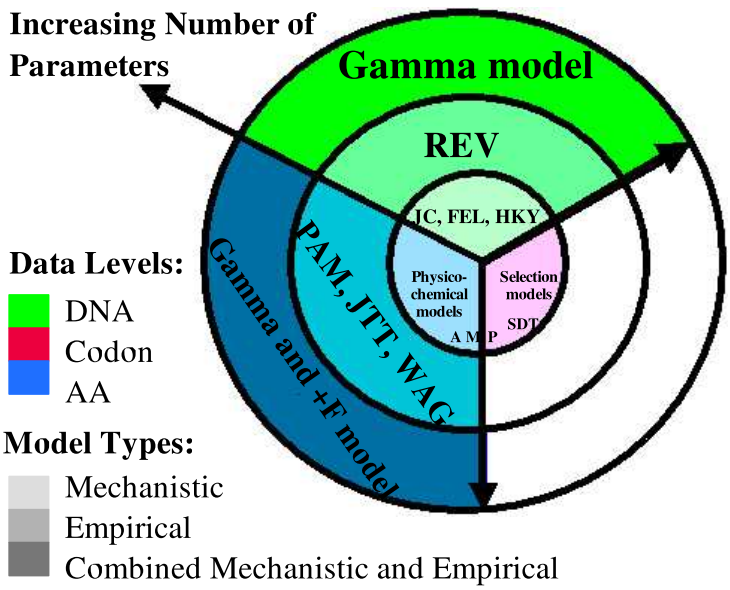


Figure 5.1: A diagram demonstrating the relationship between mechanistic and empirical evolutionary models.

Given the above definitions of empirical and mechanistic, the models presented in the introductory chapter 1 can be reordered. Fig. 5.1 summarises the relationship between the two approaches of modelling. The shaded concentric

circles describe the different types of parameters: mechanistic, empirical and combined mechanistic and empirical parameters. The coloured wedges show that protein evolution can be investigated at three levels: the green wedge represents the raw coding DNA, the red wedge codon triplets and the blue wedge translated amino acids. Mutation occurs at the DNA level and it is therefore natural to model the process of evolution on the DNA level. However, when looking at protein coding regions it is reasonable to model evolution as a process which acts on codon triplets. The triplets can be translated to amino acid sequences according to the genetic code. For more distantly related species, amino acid sequences might be more useful than DNA: the translation of DNA to amino acids may act as a filter in which some stochastic noise is removed. Information is lost too, since the genetic code is degenerate (i.e. there are 61 sense codons, but only 20 amino acids). Obviously, codon sequence studies should be preferable, because they can use simultaneously the nucleotide-level information in DNA sequences and knowledge of the genetic code and hence amino acid-level information. Only codon models capture effects of mutation acting on the nucleotide level and effects of selection acting on the amino acid level. White circle segments indicate if a model type is unexplored to date.

Following the DNA wedge in Fig. 5.1 we see that the majority of parameters in DNA models are mechanistic. The Jukes-Cantor (JC) model assumes that each nucleotide is substituted by any other at equal rate [JC69]. Felsenstein introduced a mechanistic model (FEL) in which the rate of substitution to a nucleotide depends on the equilibrium frequency of that nucleotide [Fel81]. Hasegawa and colleagues' model (HKY) added transition-transversion bias into Felsenstein's model [HKY85].

The next ring in Fig. 5.1 describes empirical models. On the DNA level the

REV model [Yan94a] can be seen as an empirical model, because it does not explicitly consider factors that shape protein evolution, but attempts to summarise the substitution patterns observed from large quantities of data. The only constraint of the REV model is that it is a reversible model. However, since the DNA alphabet only consists of four letters, the REV models are not used with fixed parameter values across different data sets, but estimated from scratch for every data set. REV can be combined with additional mechanistic parameters by allowing for rate variation between different structural and functional sites in a protein. The inclusion of a Γ distribution [Yan94b] containing a single biologically interpretable shape parameter that can accommodate varying degrees of rate heterogeneity has been proven to affect and usually improve the description of DNA sequence evolution.

Codon models are traditionally mechanistic and, in addition to the parameters contained in DNA models, describe the tendency of mutations maintaining the encoded amino acid (synonymous changes) to be accepted by selection more frequently than those changes that change the amino acid (nonsynonymous changes). A single parameter ω, the synonymous-nonsynonymous amino acid substitution rate ratio, is widely used to detect selection in proteins [GY94]. Whelan and Goldman [WG04] introduced a model including the same evolutionary factors as the standard mechanistic codon models, but allowed for single, double and triple nucleotide changes (the SDT model).

The aggregated Markov model (AMP) presented in Chapter 5 embraces the codon as well as the amino acid level and is placed between both data levels in Fig. 5.1.

In the codon 'wedge' there has been very little work on empirical models. Empirical codon models are harder to estimate; they have a high number of

parameters since they work on a 64 letter alphabet (or 61 if stop codons are discarded). I know of only one work, by Schneider *et al.* [SCG05], in which a log-odds matrix is derived from codon sequences separated by a small evolutionary distance (time) and applied in an alignment program. However, although Schneider and colleagues' codon matrix goes in this direction their approach is not a fully evolutionary model because it only gives probabilities and log-odds for a particular range of time distances. I hope to fill the white circle segments of Fig. 5.1 by deriving empirical codon models in this chapter and by analysing the performance of a combination of empirical codon models with mechanistic parameters in the next chapter.

At the amino acid level mechanistic models are rare; they came historically later than empirical amino acid models and were introduced to find explanations for the observed amino acid substitution patterns. Koshi, Mindell and Goldstein [KMG97] have developed a mechanistic amino acid model which incorporates the fitness of each of the amino acids as a function of the physico-chemical properties of that amino acid. Their model, based on Boltzmann statistics and Metropolis kinetics [MRR$^+$53], greatly reduces the 380 adjustable parameters of an amino acid model such that is possible to optimise the model for each specific data set of protein sequences. Yang and colleagues collapsed the mechanistic codon models M0 to a mechanistic amino acid model enforcing the Markov property and reversibility [YNH98]. These 'collapsed' AA models performed significantly better when they also incorporated mechanistic parameters describing physico-chemical properties.

On the other hand, there is a long tradition of empirical amino acid models. Just to recall a few: Dayhoff and colleagues [DEP72, DSO78] estimated an amino acid model which resulted in the widely used PAM matrices; Jones

et al. [JTT92] employed a much the same methods but based the estimation of JTT on a larger sequence database; and Whelan and Goldman [WG01] used a maximum likelihood estimation technique to generate the WAG matrix.

In the same manner as on the DNA level, empirical amino acid models have been combined with mechanistic parameters by introducing a Γ distribution to model rate variation [Yan94a]. Additionally the '+F' method of Cao *et al.* [CAJ+94] allows the replacement of the frequencies of the amino acids of the database the substitution matrix was estimated from with the amino acid frequencies from the specific data set analysed. Goldman and Whelan generalise the the +F approach [GW02]. Pure empirical models and/or combined mechanistic and empirical models on the amino acid level have been very successfully used in database searches, alignment, phylogeny and whole genome analysis. The success of empirical amino acid models indicates that empirical codon models could potentially be very useful.

Apart from a wide range of possible applications, the development of codon models is of value in itself to study general evolutionary pressures and processes. Jordan *et al.* [JKA+05] analyse substitutions with a counting method and observe universal trends in gain and loss for codons which code for specific amino acids. Averof *et al.* [ARWS00] find evidence for simultaneous doublet nucleotide changes when studying multiple nucleotide changes between serine-encoding codons. Bazykin *et al.* [BKO+04] and Friedman and Hughes [FH05] discuss whether the high frequencies of multiple nucleotide changes in mouse-rat sequences comparisons can be explained by positive selection resulting in rapid successive replacement. A study by Keightley *et al.* [KLEW05] indicates a reduction of the effectiveness of natural selection against deleterious mutations because of low effective population size of humans in comparison to mice. Using maximum

likelihood (ML) techniques I show that the estimation of a codon model can shed new light on some of these recently discussed questions on codon substitution patterns.

5.2 Estimation of an empirical codon model

The estimation of empirical models started with Dayhoff *et al.* [DEP72, DSO78]. Jones *et al.* [JTT92] employed much the same methods, basing the estimation of the JTT matrix on a larger sequence database. In contrast, Whelan and Goldman [WG01] used a maximum likelihood estimation method to generate the WAG matrix. In this section we will use the best of the above estimation methods to derive an empirical codon model.

Dayhoff and colleagues used a parsimony based counting method to generate the 'point accepted mutation' (PAM) matrices from the limited amount of protein sequence available at the time. Phylogenetic trees were estimated for multiple protein families, along with the ancestral sequences within those trees. The sequences within a tree were chosen to be >85% identical; the ancestral sequences were therefore even closer. This information was then used to estimate probability matrices of all amino acid replacements by simply counting inferred numbers of different amino acid substitutions. Jones, Taylor, and Thornton applied a faster, automated procedure based on Dayhoff and colleagues' approach, and used it produce a substitution matrix from a much larger database. After estimating a phylogenetic tree for each protein family in the database, their method selected a pair of sequences from a phylogeny that were nearest-neighbours and were >85% identical and counted the differences between them. The pair of sequences was then discarded to avoid recounting changes on any given branch of a phylogeny. Jones and colleagues' counting method assumes that for any given

site in the aligned pair of sequences, only one change takes place. This can lead to a serious underestimation of the true number of replacements that have occurred in branches where multiple changes have occurred and may lead to systematic error. The '>85%-rule' of the JTT method attempts to reduce this effect. Counting methods are still used on closely related sequences [KG05], although often now instantaneous rate matrices are estimated directly and not via probability or log-odd matrices as pointed out in Appendix A.

However, as well as failing to completely solve the problem, the reduction to very closely related sequences is often also quite wasteful. It discards the sequences with <85% identity and only uses pairs instead of taking advantage of the full phylogeny. A maximum liklihood estimation tries to find the amino acid substitution matrix that gives the highest likelihood simultaneously with the optimal tree topology and branch lengths for multiple alignments related by a single phylogenetic tree. The ML approach avoids the problems associated with the counting methods by using all information available in the sequences across all levels of homology and by having a model that explicitely allows for multiple changes occurring at any site in the alignment. Unfortunately ML, while providing a more reliable estimate of a substitution matrix than counting methods, is much more computationally intensive.

For the estimation of the WAG substitution matrix, Whelan and Goldman developed an approximate likelihood method which is based on the observation that the parameters describing the evolutionary process remain relatively constant across near optimal tree topologies. This means that so long as tree topologies and the branch lengths are close enough to optimal when estimating a new model, any changes when re-estimated under the model would not influence the model estimated to any great extend [SHS96, AMJS05, SAJS05]. To estimate

WAG the tree topologies were fixed and the branch length were specified within one common multiplicative constant, allowing all branch lengths to increase or decrease together.

For the estimation of a reversible empirical amino acid model 208 independent parameters need to be calculated. In contrast, to estimate a reversible empirical codon model 1889 independent parameters have to be determined (see Eq. 2.4 in section 2.4 for the calculation of the number of independent parameters). Even using Whelan and Goldman's approximation, an ML estimation of an empirical codon model has seemed infeasible for a long time because of the computational burden and the lack of a suitable data set.

The introduction of an expectation-maximisation (EM) algorithm to maximum likelihood training of substitution rate matrices by Holmes and Rubin [HR02] has greatly speeded up the computations, now making it reasonable to attempt the estimation of an empirical codon model. In statistical computing, an EM algorithm is an algorithm for finding maximum likelihood estimates of parameters in probabilistic models, where the model depends on unobserved variables. Holmes and Rubin derived an EM algorithm which is suitable to estimate matrices from a database of multiple alignments and their phylogenetic trees. However, below I only explain the principle idea of the EM algorithm with a single pairwise alignment. For multiple alignments, EM involves a dynamic programming calculation on a tree, Pearl's belief propagation algorithm [Pea88], as described in Holmes and Rubin's article.

For simplicity, consider a tree which consists of two residues a and b at the ends of a single branch of length T. The precise substitution history $\mathbf{h}$, i.e. the path from a to b, is of course unknown to us. We have a substitution model M, and are interested in finding a better model M' (i.e. a better parameterisation

of the instantaneous rates q_{ij} leading to higher likelihoods). This corresponds to first fixing the posterior distribution $P(\mathbf{h}|a, b, T, M)$ of the substitution history given the current substitution model M, then choosing a new substitution model M' that maximises the the expected log-likelihood of the data over the posterior distribution of histories. This is iterated over many steps until M and M' converge. In other words, the EM algorithm for pairwise sequences consists of maximising

$$\mathcal{Q}(M, M') = \sum_{\mathbf{h}} P(\mathbf{h}|a, b, T, M) \log P(\mathbf{h}|T, M') \quad .$$

Note that $h_0 = a$ and $h_T = b$. Holmes and Rubin solve the EM optimisation by a change of variables. As new variables they introduce the expected numbers of $i \rightarrow j$ substitutions $\hat{u}_{ij}$ and the expected waiting times $\hat{w}_i$ spent in each state (residue) i and show that the optimal instantaneous rate of residue $i \rightarrow j$ can be calculated as:

$$q'_{ij} = \hat{u}_{ij}/\hat{w}_i \quad .$$

The authors of the EM algorithm also provide an implementation within a C^{++} program called DART. DART is constructed so that it estimates reversible substitution matrices from multiple alignments and their phylogenetic trees. Although by default DART is used to estimate DNA or amino acid matrices it can be applied to any user-defined alphabet and therefore allows us to use a 61-letter codon alphabet. Robustness tests have been performed to check the suitability of DART for the estimation of an empirical codon model and suggest that DART does well (see specific details in the next sections). Small problems with convergence to ML solution were noted, and with assistance from Ian Holmes the code was improved and strategies were devised and used to ensure that good solutions were obtained. This involved, for example, adapting the granularity

on discretised branch lengths such that DART could be applied to small branch lengths which, for example, occur in pairwise alignments of genes of closely related species. Furthermore DART's possibility to restrict the instantaneous rate matrix to particular changes (e.g. single nucleotide changes only) was improved by ensuring that rates initialised with zero remain exactly zero despite the occurrence of rounding errors during long DART runs. The next sections shows how these changes to the original DART code became valuable.

5.3 Results and Discussion: Pandit

The Goldman group at the European Bioinformatics Institute (EBI) develops and maintains the Pandit database [WdBG03, WdBQ$^+$06] of aligned protein and nucleotide domains with their corresponding inferred trees. The Pandit database is based on the Pfam database of amino acid alignments of families of homologous protein domains [BCD$^+$04]. Each family in Pandit includes an alignment of amino acid sequences that matches the corresponding Pfam-A seed alignment, an alignment of DNA sequences containing the coding sequence of the Pfam-A alignment when they could be recovered, and the alignment of amino acid sequences restricted to those sequences for which a DNA sequence could be recovered. Each of the alignments has an estimate of the phylogenetic tree associated with it.

For the estimation of an empirical codon model only the DNA alignment and the DNA tree will be utilised; both are now described in detail. The DNA sequence alignment is formed using the corresponding amino acid alignment. This means the DNA alignment relies on back-translation of the manually curated Pfam protein sequences alignment. It is not realigned using DNA sequence alignment software. The Pfam-A seed alignments vary in the quality of their reconstruction of homology, both within and between alignments. This is a con-

sequence of the varying difficulty of the alignment problems encountered and the different alignment methods used in the construction of Pfam (e.g., allowing for evolutionarily distinct insertion elements in different sequences to be aligned to one another). To introduce a quality control, in Pandit each amino acid alignment is analysed using the *hmmbuild* program of the HMMER package [Edd98]. *Hmmbuild* generates a profile hidden Markov model (pHMM) classifying columns in the alignment. The alignment columns inferred to be derived from a 'match' state are taken to be reliable homologues and are labelled as such in the newest version, Pandit 17.0. In the best case, regions of 'non-match' sites might introduce random noise into the substitution patterns represented in the inferred empirical codon model. In the worst case, they could lead to systematically biased substitution patterns. For example, alignment programs often gather gaps at the same sites even when this is inconsistent with the phylogeny, with residues being distributed symmetrically around gap regions [LG05].

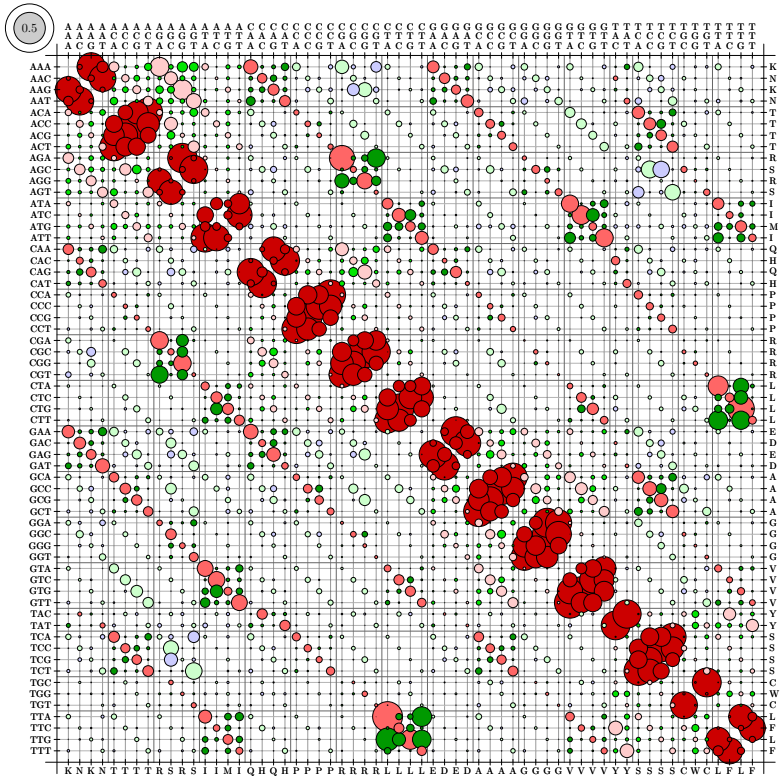
All matrices in this section are estimated using only reliable columns. Further data cleaning (e.g., discarding some residues right and left of unreliable regions; removing very short alignment fragments) did not noticeably change the substitution patterns of the empirical codon models estimated.

Pandit contains only trees based on DNA or amino acid data, not on codon data. We assume that the topologies of the codon trees and the DNA trees are the same and that the branch lengths differ by just one common scaling factor. This scaling factor is expected to be around 3, because there are three nucleotides in a codon and the branch lengths in the DNA trees are measured in expected number of substitutions per nucleotide site. However, the exact value of the scaling factor is irrelevant, since the instantaneous rate matrix will be normalised to mean rate 1. Although perhaps some of the topologies will not be ML topologies under the

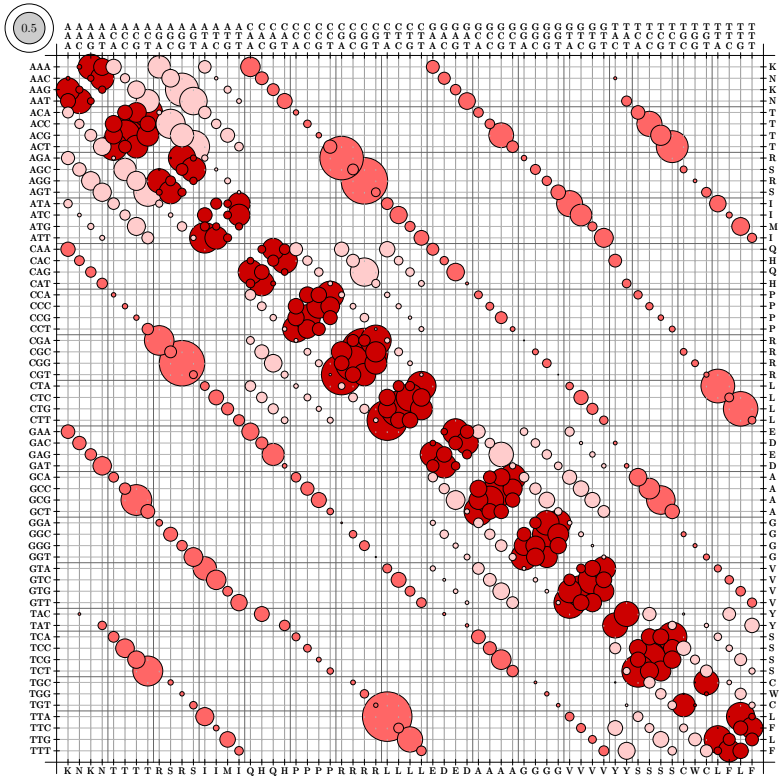
new model ultimately estimated, we adopt Whelan and Goldman's approximation that they are close enough to permit reasonably accurate estimation of this new model. The branch lengths of the Pandit DNA trees were calculated using the HKY model and the *baseml* application in the PAML [Yan97] package.

Having decided on the data set, the idea is now that we take DNA alignments out of Pandit, 'translate' them to codon alignments and run DART with a 61 character alphabet. The Pandit 17.0 database has 7738 protein families given as homologous sequence alignments and phylogenetic trees. We removed all families which cannot be confidently classified as using the universal genetic code, and all families that included any sequences with internal stop codons, leaving us with 7332 protein families. I have estimated instantaneous rate matrices from this entire database of 7332 protein families taken from Pandit. Figure 5.2 illustrates the results in form of 'bubble plots'. The bubble plots were drawn with a METAPOST-program by Nick Goldman which was specifically written for that purpose. The areas of the bubbles represent rates of instantaneous change q_{ij}. The grey bubble in the upper left corner shows the area representing an instantaneous rate of value 0.5. The rate matrices are not symmetric, because the codons have different frequencies. The codons are listed on left and top, and amino acid translations are given on the bottom and right.

Figure 5.2(a) shows an unconstrained instantaneous rate matrix permitting single, double and triple nucleotide changes. For this matrix 1889 parameters had to be estimated. The best likelihood calculated for unrestricted estimation is $\ln L_{SDT}$ = -9.157731e+07. Figure 5.2(b) shows the bubble plot of an instantaneous rate matrix restricted to single nucleotide changes. For this matrix 322 parameters had to be estimated. The best likelihood calculated for restricted estimation is $\ln L_S$ = -9.343274e+07.



(a) Best instantaneous rate matrix found by DART permitting single, double and triple changes. Single nucleotide changes are represented by red bubbles, double changes by green bubbles and triple changes by blue bubbles. Different shades of the colours red and green indicate the different positions of changes in the codon triplet.



(b) Best instantaneous rate matrix found by DART permitting single nucleotide changes only. Single changes in the 1st codon position are represented by medium red bubbles, in the 2nd position by light red and in the 3rd by dark red. Note that the dark red bubbles along the main diagonal often also represent synonymous changes.

Figure 5.2: Bubble plots for the Pandit data set.

If we accept the unrestricted matrix as reasonable, there are some double and triple changes occurring and these cannot appear in the restricted matrix. Therefore those changes must be accommodated in some other way, for example by succession of permitted single nucleotide changes. Where certain changes appear as such intermediates, we expect to see elevated rates in the restricted matrix. In fact, we note that some of the bubbles in Fig. 5.2(b) are surprisingly big. For example, the change TTG(L)→CTA(L) can be replaced by TTG(L)→TTA(L)→CTA(L), and indeed the instantaneous rates of TTG→TTA and TTA→CTA are higher in the restricted matrix than in the unrestricted matrix.

The likelihood value is the probability of observing the data given the trees and the model (i.e. the instantaneous rate matrix). The log-likelihood for the restricted rate matrix is significantly worse than for the unrestricted matrix. Significance is tested using a standard likelihood ratio test between the two models, comparing twice the difference in log-likelihood with a χ^2_{1567} distribution, where 1567 is the degrees of freedom by which the two models differ. Using the normal approximation for $\chi^2_{(1567,0.01)}$ we compare $(2(\ln L_{SDT} - \ln L_S) - 1567)/\sqrt{2 \times 1567} = 66258.42$ with the relevant 99% critical value of 2.33 taken from a standard normal $\mathcal{N}(0,1)$. The difference is seen to be significant; the P-value is too small to be calculated reliably.

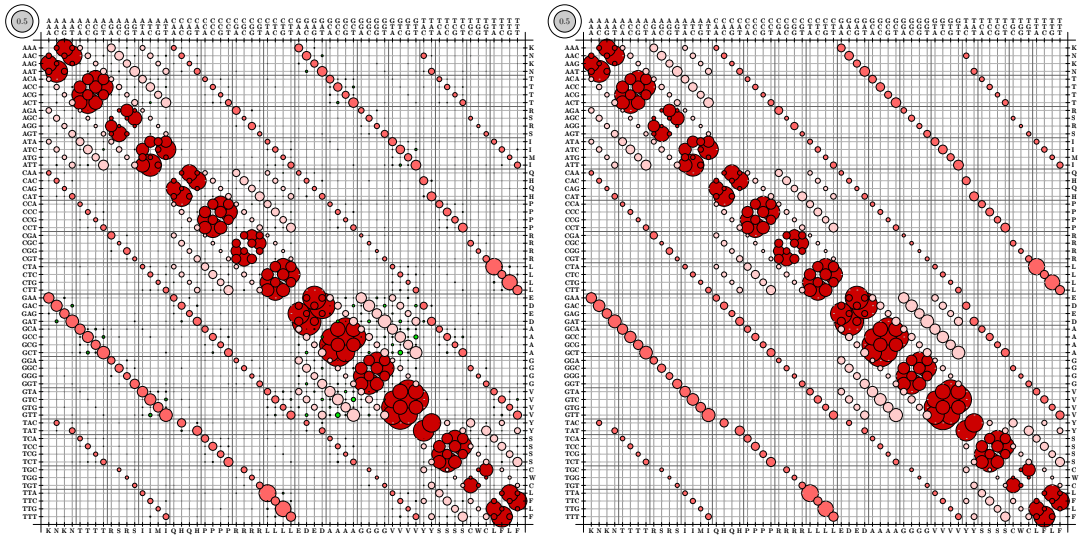
This means substitution patterns in the data set are better explained by multiple nucleotide changes than successive single changes. The instantaneous rate matrices are normalised to mean rate one (i.e. $\sum_i \sum_{j \neq i} \pi_i q_{ij} = 1$) allowing us to calculate the proportions of single, double and triple changes. We define S to be the set of all single nucleotide changes, D the set of double nucleotide

changes and T the set of all triple nucleotide changes and calculate

$$\sum_{(i\to j)\in S} \pi_i q_{ij} = 0.753, \quad \sum_{(i\to j)\in D} \pi_i q_{ij} = 0.212, \quad \sum_{(i\to j)\in T} \pi_i q_{ij} = 0.035 \quad . \qquad (5.1)$$

In words, we observe 75.3% single, 21.2% double and 3.5% triple changes.

We can also restrict the rate matrices to single and double or to single and triple changes only. The best likelihood calculated for an instantaneous rate matrix restricted to single and double changes is $\ln L_{SD}$ = 9.167463e+07 and we observe 75.3% single and 24.7% double changes. Analogously, the best likelihood calculated for a matrix restricted to single and triple changes is $\ln L_{ST}$ = - 9.195009e+07 and we observe 88.3% single and 11.7% triple changes. The introduction of double changes leads to a major improvement in log-likelihood and further introduction of triple changes is still significant. The single, double and triple model differs from a single and double model in 783 degrees of freedom. Using the normal approximation for $\chi^2_{(783,0.01)}$ we compare $(2(\ln L_{SD} - \ln L_{SDT}) - 783)/\sqrt{2 \times 783} = 4896.48$ with 2.33 from a standard normal $\mathcal{N}(0,1)$.



(a) Estimation from simulated data (b) True instantaneous rate matrix.

Figure 5.3: Bubble plots for codon model M0 with ω= 0.2 and κ = 2.5.

To check the robustness of the estimation and the results I have simulated a data set using all phylogenies of the Pandit data set, but using M0 with ω= 0.2 and κ = 2.5 and codon frequencies according to equation (3.5) as substitution model. DART is able to recover M0 well from this 'artifical' Pandit database (see Fig. 5.3).

The following scatter plot allows a more detailed analysis. Fig. 5.4 shows the true instantaneous rates $q_{ij}^{(true)}$ of M0 plotted versus the instantaneous rates $q_{ij}^{(est)}$ estimated from data simulated from M0. If $q_{ij}^{(true)} = q_{ij}^{(est)}$ the points would lie on the bisection line $y = x$. Thus the deviation of the points from the bisection line indicates how different the rates are.

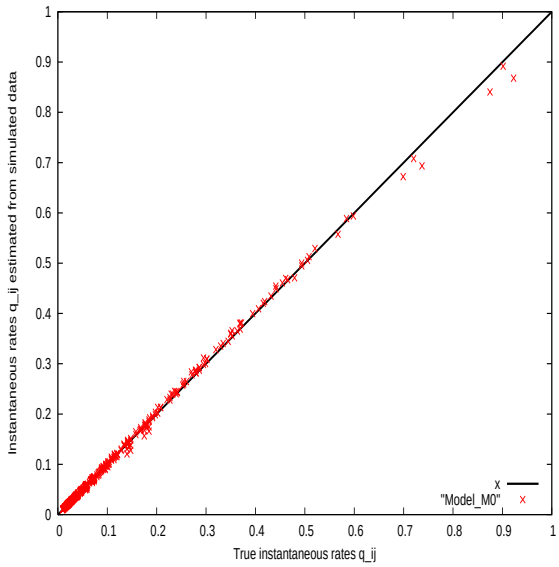


Figure 5.4: Scatter plot comparing true instantaneous rates of M0 with estimated rates from simulated data.

Especially interesting is the question how well DART can estimate instantaneous rates of multiple nucleotide changes which are zero in the mechanistic M0 model. DART sometimes inferred erroneously very small non-zero values for the instantaneous rates of double and triple changes from the data set simulated from

M0. The histogram (Fig. 5.5) shows the distribution of double (green) and triple (blue) rates changes estimated from Pandit data and from simulated data (light green and light blue, respectively).

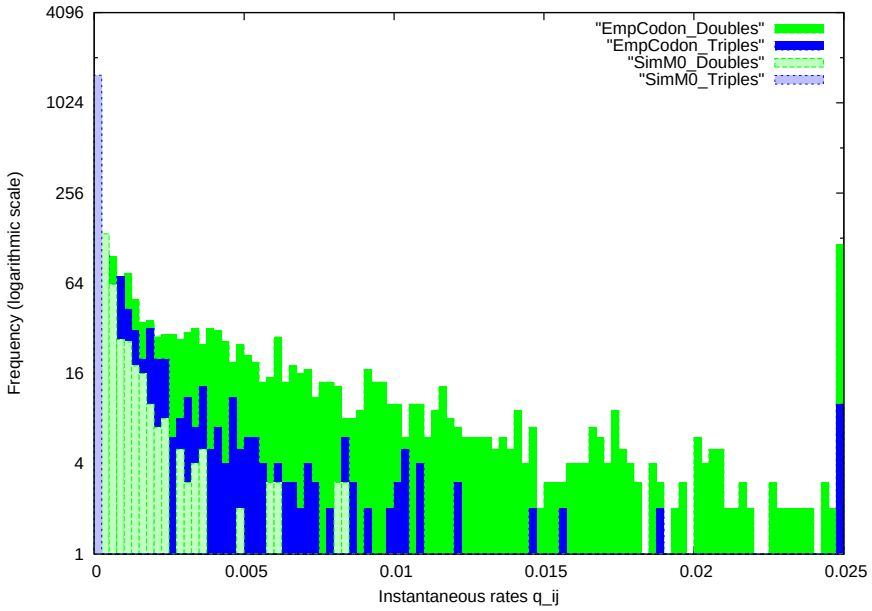


Figure 5.5: Histogram comparing instantaneous rates estimated from real data and from simulated M0 data. Note logarithmic scale on y-axis.

There is a small overlap of the distributions of rates from Pandit and erroneous rates from simulated data. However, the bulk part of the triples and especially the double rates estimated from the real Pandit database are well above these estimation errors. In summary, these robustness tests show that our methodology and DART can accurately recover zero and near-zero rates, so we can trust the small but non-zero rates observed for multiple nucleotide changes in real data (e.g. Fig. 5.2(a)).

Apart from the observation of multiple nucleotide changes, it is quite difficult to extract biologically relevant information from all matrix elements, 61×61 values, at once.

Empirical Codon Model		**Mechanistic Codon Model M0**		**Empirical AA Model (WAG)**
20 subsets	*7 subsets*	*20 subsets*	*7 subsets*	*7 subsets*
{W}	{W}	{W}	{W}	{W}
{YY}		{YY}		
{FF}	{YY FF}	{FF(TTY) LL(CTY)}	{FF LLLLLL}	{Y F}
{LLLLLL}	{LLLLLL	{LL(CTR) LL(TTR)}	{M III VVVV	
{M}	M II VVVV}	{M}	KK QQ EE DD}	{L M I}
{III}		{III}		
{VVVV}		{VVVV}		
{CC}	{CC}	{CC}	{CC SS(AGY)}	{V C}
{TTTT}	{TTTT SSSSSS	{TTTT}	TTTT	{T S
{SSSSSS}	AAAA EE	{SSSS(TCN)}	NN AAAA	A E
{AAAA}	DD NN QQ KK	{SS (AGY) RR(AGR)}	GGGG	D N Q K
{GGGG}	RRRRRR HH}	{AAAA}	RR}	R H}
{PPPP}	{GGGG}	{GGGG}	{ SSSS(TCN) PPPP}	{G}
{EE}	{PPPP}	{PPPP}		{P}
{DD}		{EE(GAY) DD(GAR)}		
{NN}		{NN}		
{QQ}		{QQ}		
{KK}		{KK}		
{RRRRRR}		{RRRR(CGN)}	{RRRRRR}	
{HH}		{HH}	{HH YY}	

Table 5.1: Application of the AIS algorithm to qthe empirical codon model. For clarity the codons are generally represented by the amino acid they encode. If relevant, the codon triplet is given as well, with R≡purine, Y≡pyrimidines, N≡any base. Colours are those proposed by Taylor [Tay97].

The Almost Invariant Sets (AIS) [KGB04] algorithm, which I have presented in Chapter 3, is a method to summarise the information of Markov models for substitutions by looking at their instantaneous rate matrix. It is a grouping method that identifies disjoint sets with a high rates of change between elements of each set but small rates of change between elements of different sets. Table 5.1 shows the results of applying AIS to the empirical codon model as well as to the mechanistic codon model M0. For the division into 20 subsets the grouping algorithm perfectly separated the 61 codons of the empirical codon model according to the amino acids they are coding for, i.e. in perfect agreement with the genetic code. For grouping into 5 subsets some of the physico-chemical properties of Taylor's grouping (see Chapter 3 and [Tay86]) are recovered (results not shown). For a division into 7 subsets the grouping algorithm leads to a very similar result as the grouping of the amino acid replacement matrix WAG (see section 3.3). The similarity between these groupings is particularly striking as the two models compared were estimated from very different data sets (see [WG01], [WdBG03]) and with one data set interpreted at the amino acid level and the other at the codon level.

Codons coding for hydrophilic and basic amino acids are grouped together. The codons of the aromatics phenylalanine (F) and tyrosine (Y) constitute one group. Tryptophan (W), proline (P) and glycine (G) form their own group. Leucine (L), methionine (M) and isoleucine (I) share the aliphatic group. The only difference to WAG grouping is that valine (V) is assigned to the aliphatic group and cysteine (C) stands alone. This recovery of the genetic code is in itself an interesting result since it shows that amino acid properties are relevant to codon substitution patterns, although they are not incorporated in standard models like M0 yet. In fact, the AIS algorithm applied to M0 reveals a quite

different grouping into 20 subsets. In particular transversion-transition biases seem to play an overly important role for the M0 groupings. For example, codons coding for leucine (L) and phenylalanine (F) share a group, while other codons coding for leucine form their own group. Likewise the codons of serine (S) are split over two groups. For a grouping of M0 into 7 subsets the groups contain codons coding for mixture of amino acids with very different physico-chemical properties (e.g., hydrophilic, aromatic and basic). The codons coding for serine are still separated.

Both observations, multiple nucleotide change and the strong influence of the genetic code and physico-chemical properties, are not reflected in M-series of standard codon models of [YNGP00]. The existence of simultaneous multiple nucleotide changes is controversial: Averof *et al.* [ARWS00] find evidence for simultaneous multiple changes in residues coding for serines. The SDT model of Whelan and Goldman [WG04] implies that multiple nucleotides changes occur. However, Bazykin *et al.* [BKO$^+$04] argue for successive single compensatory changes instead. They choose murid sequences for their study because they expect non-parsimony (i.e. normal multiple hits in a codon) to be rare for mouse and rat. So far, my results support the existence of multiple nucleotide changes. But are they really instantaneous changes, or are they compensatory changes and so fast that they just appear instantaneous in this analysis?

Physico-chemical properties were introduced in early codon models [GY94] by using a Grantham matrix [Gra74], but subsequently have been dropped for lack of success (Nick Goldman, pers. comm.). Massingham [Mas02] has recently used large quantities of data to estimate exchangeability parameters representing the bias given by physico-chemical properties. The estimation of an empirical codon matrix tries to settle these controversies as well as proposing a new model of codon

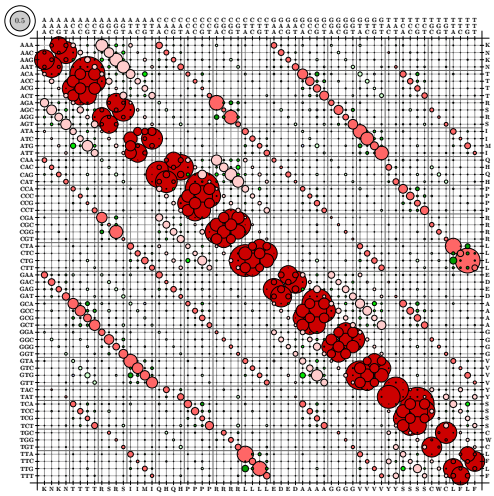
evolution at the same time. A major application of codon models is the detection of selection and it is likely that these results will also have consequences for selection studies. Therefore, the next sections will treat the following questions: What is causing the observed substitution patterns? Are they observed in other data sets than Pandit? What are the consequences of applying empirical codon models?

5.4 Results and discussion: genomic data

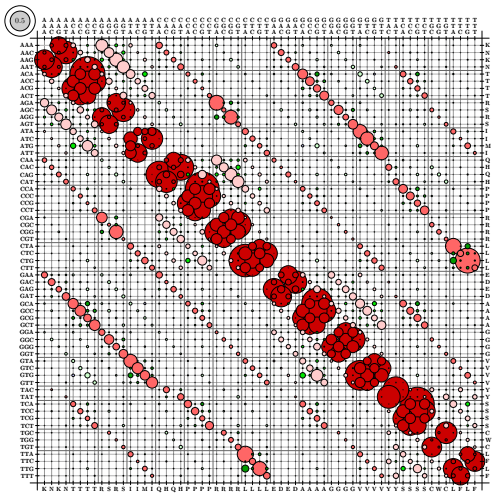
The completion of sequencing projects of vertebrates such as human [Con01], mouse [Con02], rat [Con04b], dog [LTWM+05], chimpanzee [SC05] and many others has recently lead to dramatic increase in genomic data. I have estimated instantaneous rate matrices from each of 14028 human-mouse, 14482 mouse-rat and 10299 human-dog coding sequence pairs of orthologous genes. Matrices estimated from human-chimp alignments are not considered in this analysis because of the small number of differences between the sequences, problems with the quality of the sequencing and effects caused by the small population size of human and chimp that are specific to this data set. If not mentioned otherwise these sequence data were taken from the Ensembl database [BAC+06]. The Ensembl Compara software allows retrieval of the orthologue predictions from the results of a genome-wide comparison calculated for each data release. The comparisons are pairwise, making the evolutionary relationship between the orthologous genes particularly simple to calculate: the tree consists of a single branch of evolutionary distance l. To make the results comparable to the Pandit results I have estimated the pairwise evolutionary distances for each gene in PAML [Yan97] using the HKY model.

In Ensembl, orthologues are identified on a large scale and in an automated

way. Only pairwise comparisons are taken into account and only 'Best Reciprocal Hits' are considered. However, no standardised method to identify orthologues has yet been established. To test the adequacy of the Ensembl orthologue predictions, matrices were estimated for orthologues found with two different methods. For human-dog orthologues we compared the instantaneous rate matrices estimated from Ensembl data with a more carefully chosen subset of orthologues based on a phylogenetic approach, kindly provided by Leo Goodstadt. There are no noticeable differences between the bubble plots of the matrices (Figs. 5.6(a) and 5.6(b)) and the scatter plot (Fig. 5.7) shows that the values of corresponding instantaneous rates are very close to each other. Thus for the purpose of estimating instantaneous rate matrices the choice of orthologues in Ensembl seems to be sufficiently reliable.



(a) Rate matrix from human-dog sequences. 'Best reciprocal hit' is used to find the orthologues by Ensembl-compara.



(b) Rate matrix from human-dog sequences. Goodstadt uses a phylogenetic approach to identify the orthologues.

Figure 5.6: Different sets of orthologues.

Analogously to the robustness validation on the Pandit data set, I have tested

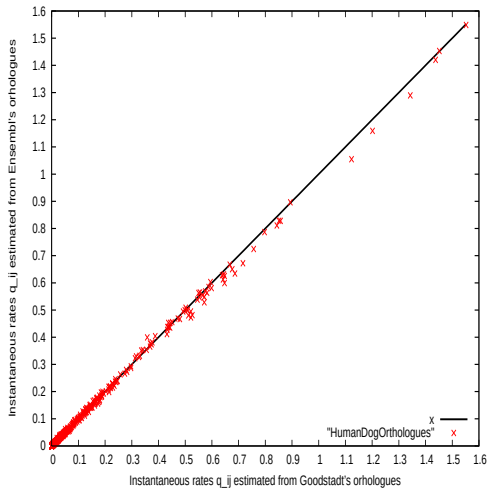


Figure 5.7: Scatter plot comparing instantaneous rates of Ensembl's and Goodstadt's human-dog orthologues.

that the genome data sets are suitable for the derivation of an empirical codon model by estimating a model from data simulated under the model M0 and human-dog pairwise distances. Fig. 5.8 shows a comparison between the instantaneous rates of multiple nucleotide changes estimated from simulated data and the real human-dog data. The bulk of instantaneous rates for double and triple changes estimated from real data exceeds the wrongly inferred rate values of double and triple changes for M0 data indicating that most of the observed rates are real and well above estimation errors.

However, although many double and triple changes can be observed, the proportions of multiple changes are clearly smaller in human-dog than in Pandit, as comparison of Fig 5.2(a) and Fig 5.6(a) shows. This observation was confirmed for the other genomic data sets, human-mouse and mouse-rat (not shown). The coding DNA of human and mouse (and human-dog and mouse-rat) orthologous genes is less diverged from each other, and easier to align, than Pandit sequences. This raises the question whether an increasingly difficult alignment procedure or

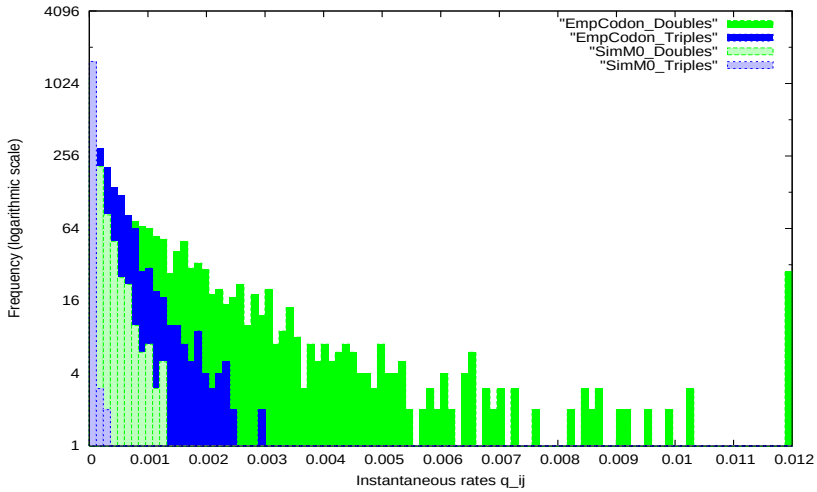
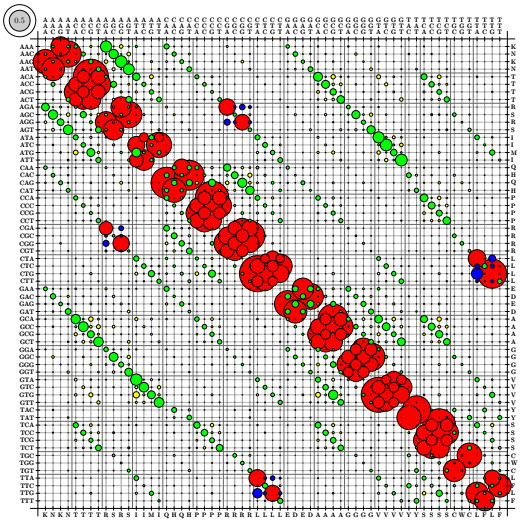


Figure 5.8: Robustness tests for genomic data: Instantaneous rates of multiple nucleotide changes in real and simulated human-dog data sets. Note logarithmic scale on y-axis.

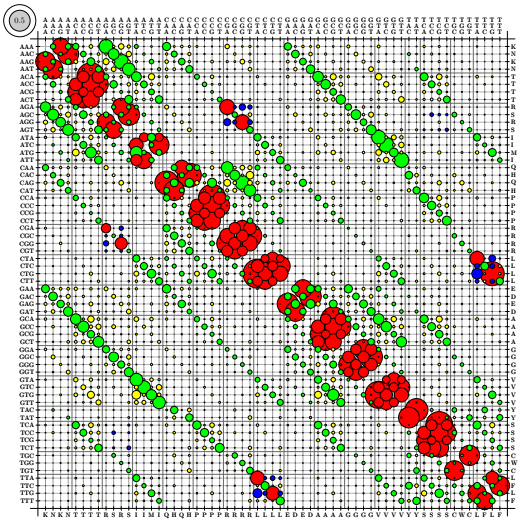
increasing evolutionary distances (time) could influence the inferred substitution patterns. Protein coding DNA alignments in Pandit and Ensembl are formed by aligning amino acid sequences and back-translating to DNA. This procedure obviously 'knows about' the genetic code. Amino acid alignment programs could be inclined to put identical amino acids together in error despite the absence of any indication on the codon level. Amino acid alignment programs often also use knowledge about physico-chemical properties of amino acids, preferring to align similar residues. This too might bias subsequent model estimates.

The UCSC genome browser [KBD+03] provides coding DNA from the same genes as Ensembl, but the alignments are performed directly on the DNA level. Although instantaneous rate matrices estimated from DNA alignments might suffer from different, they do not suffer from the same alignment artifacts as matrices estimated from AA alignments. I have worked with a data set of UCSC human-mouse alignments which was extracted by Gerton Lunter to trace possible alignment artifacts. After looking at the distribution of evolutionary distances l mea-

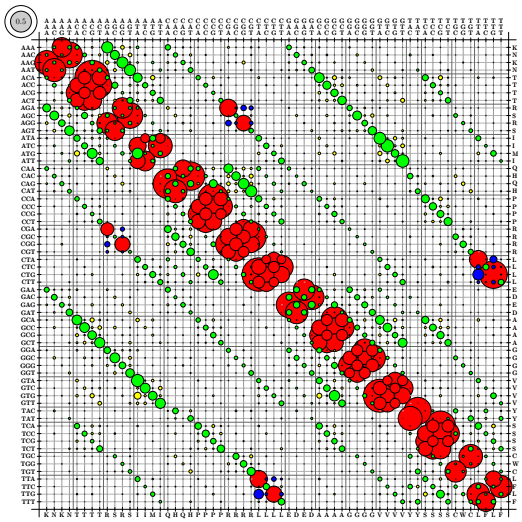
sured in HKY, the Ensembl as well as the UCSC human-mouse alignments were split into two subsets according to the median evolutionary distance (l =0.107) of the Ensembl data set. Subsampling guaranteed that the same amount of changes were occurring in all the data subsets, (although that a later analysis showed that results without subsampling were not noticeably different). Rate matrices were estimated using DART (Fig. 5.9). To more clearly distinguish effects of multiple nucleotide changes and the genetic code, the colour coding of the bubble plots is as follows: red represents synonymous single changes, green nonsynonymous single changes, blue synonymous multiple changes and yellow nonsynonymous multiple changes. We observe different patterns for slowly ($l < 0.107$) and fast evolving ($l \geq 0.107$) genes independently, whether the sequences were aligned on the DNA or AA level. In comparison, the instantaneous rate matrices derived from human-mouse sequences aligned on the AA and DNA levels are very similar.



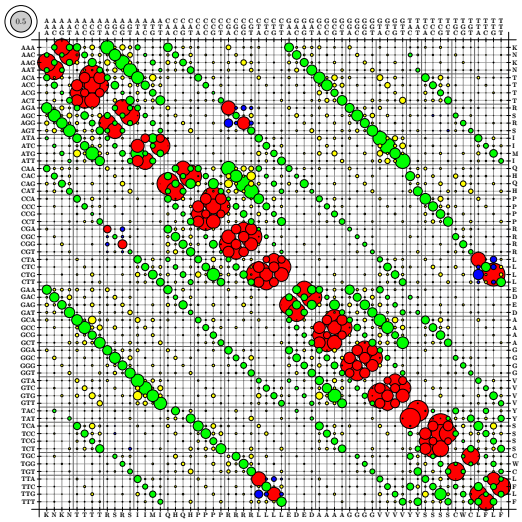
(a) AA level. Rate matrix from slowly evolving human-mouse genes using Ensembl-compara alignments.



(b) AA level. Rate matrix from fast evolving human-mouse genes using Ensembl-compara alignments.



(c) DNA level. Rate matrix from slowly evolving human-mouse genes using UCSC whole genome alignments.

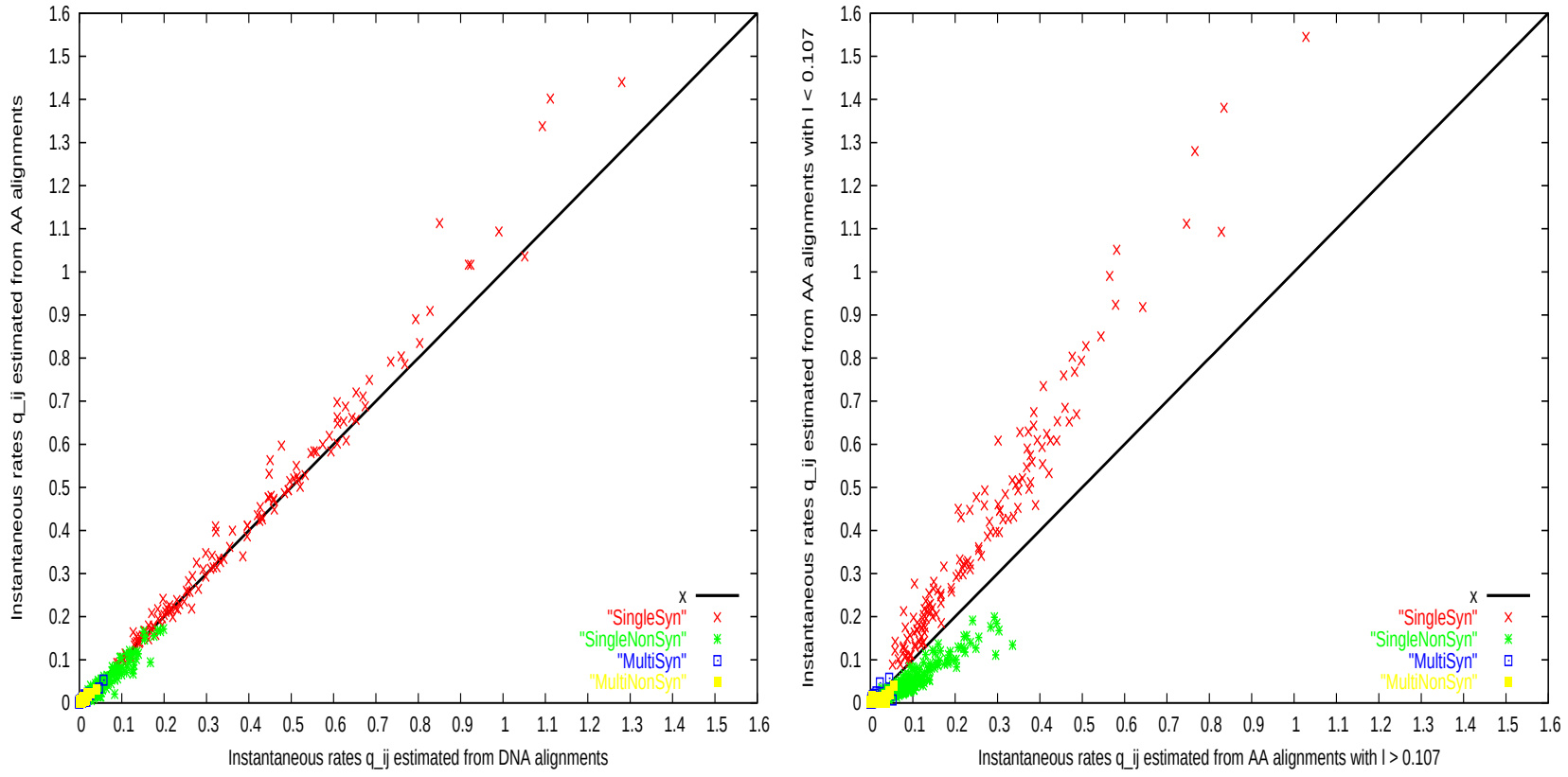


(d) DNA level. Rate matrix from fast evolving human-mouse genes using UCSC whole genome alignments.

Figure 5.9: Bubble plots of matrices estimated from DNA and amino acid alignments of fast and slowly evolving genes.

The following scatter plots allow a more detailed analysis. Fig 5.10(a) shows instantaneous rates $q_{ij}^{(DNA)}$ estimated from DNA aligned genes plotted versus the instantaneous rates $q_{ij}^{(AA)}$ estimated from AA aligned genes from the same range of evolutionary distances $l \leq 0.107$. If $q_{ij}^{(DNA)} = q_{ij}^{(AA)}$ the points would lie on the bisection line $y = x$. The colour scheme of the points is analogous to the bubble plot colouring, distinguishing synonymous and nonsynonymous and single and multiple changes. Figure 5.10(a) shows that there is a slight bias for red points to lie above the bisection line, and green points to be found below. This indicates a minor tendency that the sequences aligned on the amino acid level have been overaligned. Overalignment can be caused if the alignement program matches identical amino acids in gapped regions, although on the codon level these residues would not match. We counted that only 3.03% of the multiple changes happened next to a gap in the human-mouse data set. From a simulation study of 100 samples we know that by chance 0.37% (median; 5th and 95th percentile: 0.34%, 0.40%) of the changes would happen next to a gaps. Thus 3.03% is an excess, but still only a very small proportion of the changes observed in the human-mouse data set, and this should not bias the overall results. The effects caused by overalignment slightly increase for sequences separated by $l > 0.107$ (results not shown).

Fig. 5.10(b) reveals a far greater effect on the divide between rates of synonymous and nonsynonymous changes estimated from DNA alignments when plotting $q_{ij}^{(l \leq 0.107)}$ against $q_{ij}^{(l > 0.107)}$. There is a bias for red and blue points to lie above the bisection line, and green and yellow points to be found below. This can be observed on the AA level as well (not shown). It seems that it is not the alignment method but the biology which extensively influences the substitution patterns.

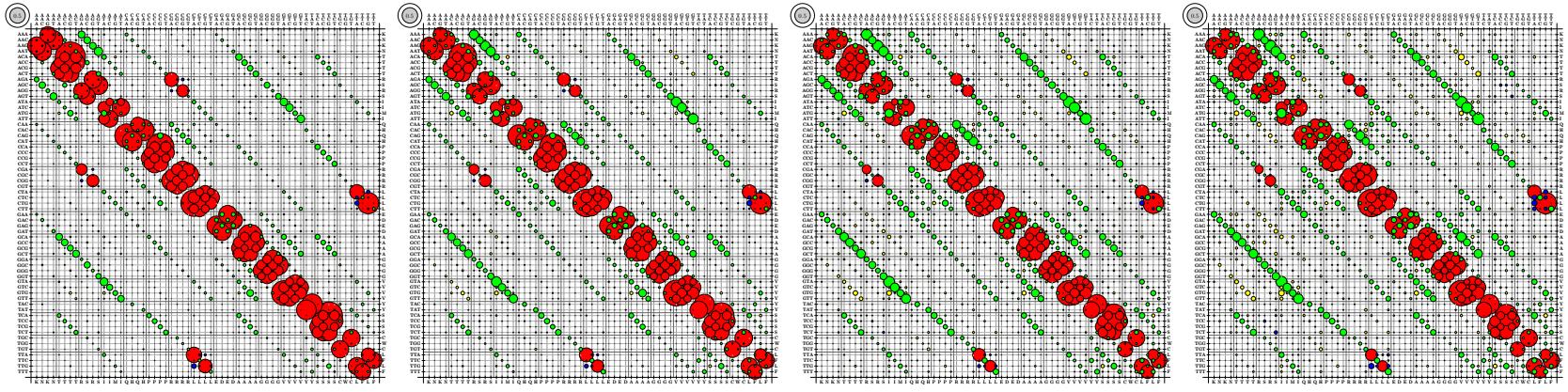


(a) (b)

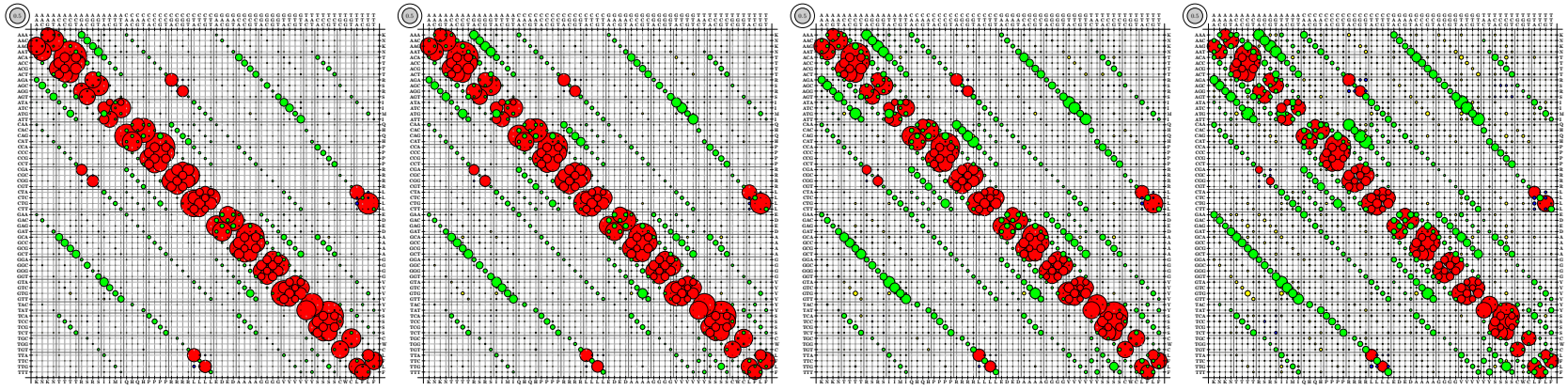
Figure 5.10: Scatter plots comparing matrices estimated from AA and DNA alignments of fast and slowly evolving genes.

In fact this leads back to an earlier research topic of this Ph.D. thesis. In chapter 4 the phenomenon of time-dependent evolution observed in amino acid substitution matrices was explained by an aggregated Markov process. Here again, this looks like time-dependent evolution, but on the codon level.

To investigate if there is an effect due to evolutionary distance I further subdivided the human-mouse data set and split the mouse-rate data set in to the same number of subsets. The quartile evolutionary distances for human-mouse are 0.2541, 0.3594, 0.5156, and for mouse-rat 0.0854, 0.1162, 0.1615, respectively. If split according to the quartile values, the human-mouse as well as the mouse-rat data set show a series of different substitution patterns. For the smallest evolutionary distances ($l < 0.2541$ in human-mouse and $l < 0.0854$ in mouse-rat) the matrices are sparse (Fig. 5.11 (a) and (e)) . With increasing l, more and more non-zero rates for multiple nucleotide changes can be observed (Fig. 5.11). However, according to evolutionary distance l the human-mouse and the mouse-rat series barely overlap, and even the matrices with overlapping time distances in Fig. 5.11(a) and Fig. 5.11(h) do not show similar substitution patterns; indeed, they are at opposite extremes of what is observed. Thus the effects are not due to distance. This also rules out alignment artifacts again. Alignment should be equally difficult at equivalent time distances, but matrices Fig. 5.11(a) and Fig. 5.11(h) are not similar.



(a) $l < 0.2541$ (b) $0.2541 \leq l < 0.3594$ (c) $0.3594 \leq l < 0.5156$ (d) $0.5156 \leq l$



(e) $l < 0.0854$ (f) $0.0854 \leq l < 0.1162$ (g) $0.1162 \leq l < 0.1615$ (h) $0.1615 \leq l$

Figure 5.11: Split of human-mouse (top) and mouse-rat (bottom) data sets into four equal size subsets sets according to evolutionary distance l measured with M0.

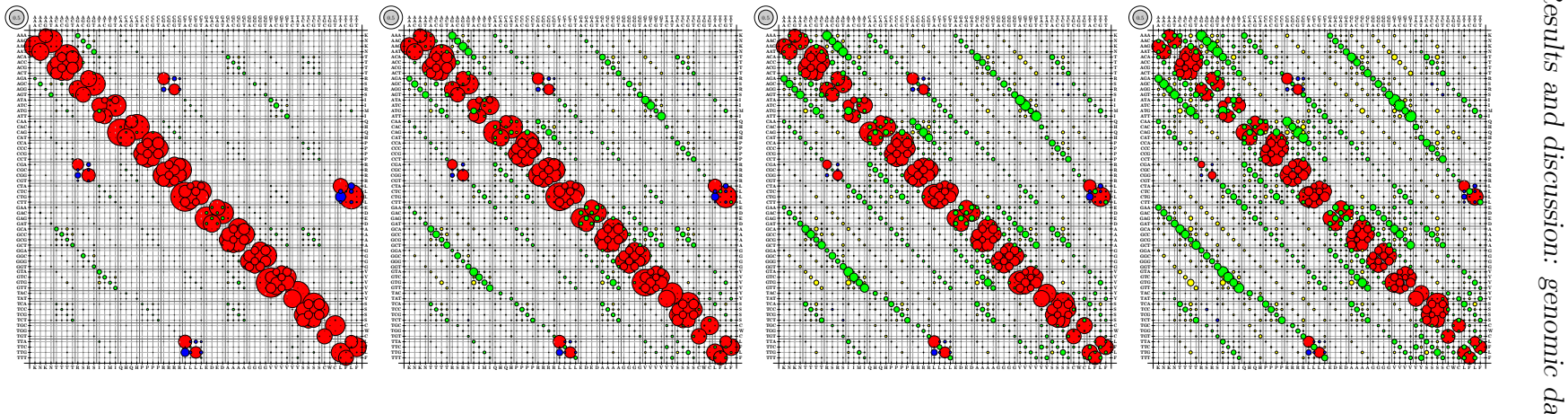
Another biological effect attributed to evolutionary distance would be that the faster evolving genes are less constrained. This means selection acts less strongly on fast evolving genes, and it might be that double and triple changes can be fixed in the population more easily. However, the evolutionary distance l is only a very limited measure for evolutionary constraint by selection. A better measure for selection is the synonymous-nonsynonymous rate ratio ω. For each pairwise alignment, ω was estimated under the M0 model in PAML. Fig. 5.12 shows instantaneous rate matrices where the human-mouse and the mouse-rat data set are split into subsets according to quartile ω values.

In Fig. 5.12, observe that substitution patterns gradually change with increasing ω. For low ω values the instantaneous rate matrices are sparse; for high ω the matrices have gained plenty of additional non-zero elements. This mirrors the behaviour observed for increasing l, but in contrast to l the gradual variation in substitution patterns is consistent over different datasets. Instantaneous rate matrices estimated from corresponding human-mouse and mouse-rat datasets can be matched with each other (e.g., Fig. 5.12(a) and (e), (b) and (f), (c) and (g), (d) and (h)) because they have similar substitution patterns and similar ω values. It is even possible to explain slight differences in substitution patterns with slight difference in the ranges of ω values: Fig. 5.12 therefore offsets the mouse-rat matrices slightly against the human-mouse matrices. The gradual change of the instantaneous rates q_{ij} as ω varies makes the parameter ω a valuable candidate to be re-introduced to an empirical codon model. The next chapter will combine mechanistic parameters with the empirical codon models in this way.

Fig. 5.12 also allows investigation of the occurrence of double and triple nucleotide changes in more detail. For strong purifying selection, $\omega < 0.0349$ (Fig. 5.12 (a)), the only multiple nucleotide changes inferred to occur noticeable

on the bubble plot are among codons which code for leucine (L) and arginine (R). These amino acids are each encoded by two groups of codons and changes between the two types of codons could occur directly with a multiple change or indirectly via two successive single changes. For leucine (and arginine) the intermediate step can be achieved via a residue coding for leucine (arginine, respectively). Likewise serine (S) is encoded by two groups of codons, AGY and TCN, where Y stands for purines and N stands for any base. But serine is unique among the amino acids, because the interchange between AGY and TCN cannot result from two separate single nucleotide substitutions without involving an intermediate step whereby the triplet would not encode for serine (S), but for threonine (T) or cysteine (C), residues that are physico-chemically very different from serine. Thus such changes are unlikely to be tolerated in functional or structural important sites.

For strong purifying selection, $\omega < 0.0349$ (Fig. 5.12 (a) and (e)), multiple nucleotide changes for leucines and arginines are observed, but not serine switches. Also, the double changes occurring in leucine and arginine are changes at the first and third codon position. There is no biological process known which causes simultaneous double changes with this pattern. Furthermore, although non-synonymous single changes are observed if the codons code for physico-chemically similar amino acids, no significant rates are observed for double or triple changes conserving the physico-chemical properties of the amino acids.



(a) $\omega < 0.0349$ (b) $0.0349 \leq \omega < 0.0873$ (c) $0.0873 \leq \omega < 0.1781$ (d) $0.1781 \leq \omega$

(e) $\omega < 0.0484$ (f) $0.0484 \leq \omega < 0.1267$ (g) $0.1267 \leq \omega < 0.2575$ (h) $0.2575 \leq \omega$

Figure 5.12: Split of human-mouse (top) and mouse-rat (bottom) data sets into four equal size subsets sets according to synonymous- nonsynonymous rate ratio ω.

The above findings strongly suggest that on the mutation level only single nucleotide changes occur. In particular, the occurrence of double changes in the 1st and 3rd position (e.g., ACG(S)→TCC(S)) suggests the occurrence of compensatory changes: I do not know of any biological mechanism affecting the 1st and 3rd codon, but not the 2nd codon position. Apart from compensatory changes, the only other explanation is that triple mutations occur 'changing' the 2nd position to the same nucleotide by chance, which highly unlikely.

On the population level, could the multiple changes observed amongst arginines and leucines be successive single changes in one individual? For example in humans, the probability of a mutation is 10^{-8} per nucleotide per generation [Neu03]. Therefore the probability of two independent mutations in one individual is $10^{-8} \times 10^{-8}$ and the ratio of double changes to single changes would be 10^{-8}. This clearly cannot explain the high ratio of double and triple changes observed in the empirical codon models. Likewise, recombination events [Nor03] are not a plausible explanation for the effect observed: the probability of one individual having one mutation at one site, and another individual a mutation at neighbouring site, and those two mating and the cross-over placing the two mutations onto one genome is highly unlikely as well, since the cross-overs would require a break right between the two neighbouring sites.

Positive selection caused by the compensation of a deleterious mutation by a mutation at another, espistatically interacting, site in the genome, seems to be the most likely mechanism to explain the multiple changes observed. The latter mutation is then positively selected even if it was deleterious in the wild-type background [KSK02]. The commonness of compensatory mutations is indicated by the observation that for each deleterious amino acid substitution, there might be as many as 10-12 potential mutations at other sites that can compensate for

the loss in fitness [PDC05]. Note these compensatory changes are not necessarily found at neighbouring sites. Unsurprisingly, organisms with high mutation rates (e.g., viruses) provide the clearest examples of compensatory evolution. However even for human, a remarkably high fraction of pathogenic mutations are fixed in other closely related species, probably reflecting compensatory substitutions [KSK02].

Because individual mutations can either decrease or increase the stability of protein structures depending on other sites, it has been suggested that compensatory mutations are particularly important for maintaining the structure of in evolution [DWH05]. As expected, the probability of compensatory substitutions increases with the severity of the deleterious fitness effect [PDC05].

Positive selection causing compensatory changes is also dependent on often unknown population genetic factors such as population size, allowing for various scenarios. Multiple nucleotide changes could be the result of neutral mutations spreading in a population by genetic drift [Neu03] and then an advantageous mutation occuring which is positively selected for. In very large populations a mildly deleterious can be sustained in a subpopulation[Exc03]; if a compensatory mutation then happens it will be positively selected, and may spread through the whole population and be fixed. On the other hand very small populations are more susceptible to mutations[Neu03], such that even deleterious mutations can be fixed in the population. These mutations may then be followed by compensatory mutation which becomes fixed too. This mechanism could give a plausible mechanism for serine switches, where the intermediate amino acid is very deleterious [ARWS00]; in particular, when changing populations sizes are also considered. Assume a very deleterious change occurs in a small population and gets fixed: if a compensatory changes then happens as well as a population expansion, the

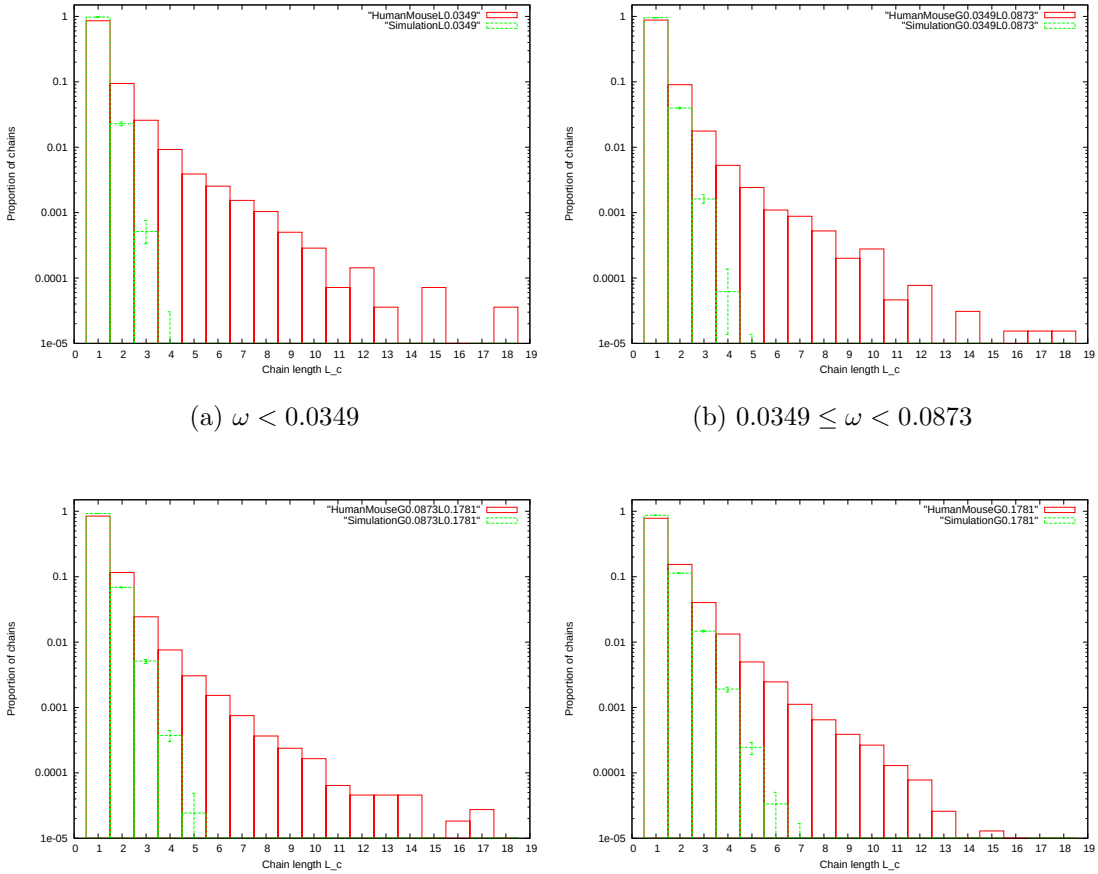
resulting double change may be fixed very quickly in the population.

The last two explanations using subpopulations or expanding populations are most plausible to me. However, this topic requires further study, for example by combining comparative analysis with large-scale polymorphism data (e.g., HapMap [Hap03] and Trace Archive [Tra]).

Compensatory change has similar been argued to explain pairs of changes in sequences with RNA structure. Observing a mutation that changes a single base in a stem region of a RNA molecule is rare because they are strongly selected to maintain complementary base pairing. Regular changes, however do occur at a low rate when both complementary bases mutated in rapid sucession. This problem has been addressed by recording the data in stem regions as pair of nucleotides and allowing these rapid mutations to occur as simultaneous events in the model [Hig98, SHH01].

Returning to Figure 5.12 we have seen that compensation could lead successive single mutations to be fixed at a very high rate which makes them indistinguishable from multiple mutations when observed only on a 'phylogenetic timescale'. For the data sets containing less constrained genes with a larger ω, mutations are more easily fixed in the population, explaining why we observe continuously higher rates for multiple nucleotide changes in these data sets. Also, slightly deleterious mutations can be sustained in a population for longer, if purifying selection is weak (high ω), meaning there is more opportunity for a compensatory change to occur.

However, less constrained genes are also more likely to have undergone large scale effects like gene conversion [Saw89, SW99]. For example, once a gene duplication event has generated two 'daughter sequences' nearby on one chromosome, gene conversion and consequent multiple changes more likely to happen. The



(a) $\omega < 0.0349$

(b) $0.0349 \leq \omega < 0.0873$

(c) $0.0873 \leq \omega < 0.1781$

(d) $0.1781 \leq \omega$

Figure 5.13: Comparison of chains of multiple changes for the human-mouse data set and a simulated data. Note the logarithmic scale on the y axes.

existence of another copy means that each copy is less constrained. Thus, duplication leads to higher rates of conversion occurring and to higher rates of it being accepted by selection.

To detect larger scale events we investigate if multiple changes tend to happen at neighbouring codon. Starting at a codon site with a multiple nucleotide change we count how many neighbouring codon have undergone a multiple changes, building up a chain of positions of length L_c in the genome. Fig 5.13 shows the proportions of the different chain lengths in the human-mouse data sets split

with respect to ω (red) and what these proportions are if multiple changes were distributed randomly (green). Results are calculated from 100 samples of a simulation assuming that multiple changes are uniformly distributed on the genome. Therefore an observation (from real data) greater than the biggest simulated chain length has estimated probability of the occurrence < 0.01. We observe chain lengths of up to 18 codon in the human-mouse data sets. For the most weakly constrained genes ($\omega > 0.1781$) chain lengths no greater than 7 occur by chance, and for very constrained genes ($w < 0.0349$) no chains longer than 4 occur by chance. This indicates that additionally to the single nucleotide changes, large scale effects do occur which can not be explained by random accumulation. However, since chains of length $L_c \geq 3$ only constitute a very small proportion of all events (chains), they should not bias the overall substitution patterns observed in instantaneous rate matrices.

It should be noted that it is known that the assumption used in the null model (that mutations are randomly distributed over the genome) is somewhat very simplistic. Poon *et al.* [PDC05] have shown that mutations are not uniformly distributed over the genome. However, genome-scale studies of polymorphism data have only recently begun and it is not clear yet which distribution would act as a better null models for the above test.

5.5 Conclusions and future work: empirical codon models

In this chapter empirical codon models were estimated from Pandit and from various genomic data sets taken from Ensembl and the UCSC database. Comparing the variety of estimated matrices allows us to draw conclusions about the biological pressures and processes acting during codon sequence evolution. We

also show that multiple data sets can be used to distinguish biological factors from estimation errors and alignment artifacts.

Schneider *et al.* [SCG05] calculated a first codon matrix from an Ensembl-derived data set which was constructed by concatenating multiple pairwise alignments. Because they used a counting method, orthologous genes were restricted to a specific evolutionary (time) distance. Primarily focusing on alignment improvement, Schneider and colleagues do not present a biological interpretation of substitution patterns. They simply report that their codon model is successful in the alignment of sequence of a particular time distance and that log-odds scores for multiple changes are generally smaller than for log-odds of single changes.

Using ML techniques, we can estimate matrices from Ensembl data of various evolutionary distances without risking the underestimation by multiple changes. We observe different substitution patterns, i.e. instantaneous rates, for different evolutionary distances. With increasing evolutionary distance higher instantaneous rates for multiple nucleotide changes can be observed. This could be caused by alignment artifacts, as well as different biology in fast or slowly evolving genes.

However, the comparison of matrices estimated from DNA and AA alignments show that although there is some bias introduced alignment artifacts, this effect is minor in comparison to the effect caused by selection acting differently on slow- and fast-evolving genes. It is shown that the synonymous-nonsynonymous rate ratio ω is a suitable measure to classify different substitution patterns represented by the matrices. This emphasises the importance of re-introducing an ω-parameter into the empirical codon model, which will allow modelling of the effect seen on a per-protein basis.

The dependency on ω also sheds new light on a previous chapter of this thesis. In Chapter 4 I have shown that some of the time-dependent evolution observed by

Benner *et al.* [BCG94] can be explained by aggregated Markov processes which combined DNA mutational biases, rate heterogeneity among different codon sites, the properties of the amino acid encoded by the sequence and effects of selection operating on these amino acids. However, all above factors were modelled with mechanistic models in which selection was only considered by a single parameter ω. It would be interesting to return to this research problem with an empirical model and the knowledge about the importance of variation in ω for the effect.

The dependency of the substitution patterns on ω also has consequences for the understanding of substitution patterns observed. Double and triple nucleotide changes do appear to occur. They seem instantaneous but in fact are probably mostly not, as the study of very constrained genes has shown. Instead, if selection pressure is low enough then compensatory changes on a fast time scale (population level) can be fixed, even if via a deleterious intermediate codon. On the other hand, if purifying selection is strong then only compensatory changes via synonymous intermediates occur.

The Pandit (and genomic) empirical models have double and triple changes, but the study of the patterns in genomic data suggests that only single changes occur instantaneously and yet the Pandit model had a better likelihood than a matrix restricted to single nucleotide changes and we will see in the next chapter that the Pandit model is good for ML phylogenies. The explanation for this 'discrepancy' is that the multiple changes are in fact successive single changes occurring on a (much faster) timescale. If this is the case, the ML phylogeny would still work well because those sorts of data are over long timescales and cannot indicate short timescale over which compensatory changes occur. Additionally, this could probably also explain the conflicting results on multiple nucleotide changes in the literature (e.g., Bazyakin [BKO$^+$04]; Averof [ARWS00] and Whelan and

Goldman [WG04]).

Table 5.2 summarises the proportion of single, double and triple nucleotide changes of the matrices presented in this chapter.

Data set	*single*	*double*	*triple*
Pandit	75.3%	21.2%	3.5%
human-dog	94.9%	4.2%	0.9%
human-mouse ($\omega < 0.0349$)	99.0%	0.9%	0.1%
human-mouse ($0.0349 \leq \omega < 0.0873$)	96.7%	2.8%	0.5%
human-mouse ($0.0873 \leq \omega < 0.1781$)	94.5%	4.8%	0.7%
human-mouse ($0.1781 \leq \omega$)	91.6%	7.4%	1.0%
mouse-rat ($\omega < 0.0484$)	98.2%	1.8%	0.0%
mouse-rat ($0.0484 \leq \omega < 0.1267$)	97.6%	1.9%	0.6%
mouse-rat ($0.1267 \leq \omega < 0.2575$)	94.7%	3.9%	1.4%
mouse-rat ($0.2575 \leq \omega$)	91.3%	6.7%	2.0%

Table 5.2: Proportion of single, double, triple nucleotide changes in the data sets analysed.

Pandit has a large proportion of diverged sequences separated by long branch lengths (i.e. large evolutionary distances). Consequently, more multiple hits resulting in substitutions in the same codon are observed in Pandit data than in genomic data. Using a maximum likelihood method can correct for these multiple hits. Indeed, DART's results on simulated data from M0 but using trees from Pandit have shown that the method can correct for multiple hits efficiently.

However, Pandit has by far the highest percentage of double changes in the data set. One explanation is that Pandit contains the DNA of a mixture of slow and fast evolving genes, and constrained and relaxed genes. But because fast evolving genes contribute more changes, this could lead to the high proportions of multiple nucleotide changes observed in the Pandit data set. Also Pandit might contain more sequences showing high rate variation within individual genes. This relates to the type of model misspecification caused by site-specific variation in evolutionary rates and selection which we have encountered before. As in Chapter 4 site-specific variation of evolutionary rates can be investigated using a gamma distribution and in particular focussing on its shape parameter α. It would be interesting to split the Pandit data into subsets with small/large α and compare the matrices estimated from these subsets. Furthermore, the Pandit dataset could in principle be split into subsets with small/large ω to investigate effects specifically caused by selective constraints. However, the later cannot be done as easily as on genomic data, since the Pandit data is bound to a tree and not just pairwise distances. Although codon models like M0 have recently been considered for tree estimation, they are still not robust and fast enough to run on a whole database like Pandit (see [RTY05]). However, estimating matrices from simulated data from Pandit trees under a model allowing for site-specific variation of omega like M7, should give a rough idea of the effects of this model misspecification. This experiment could be carried out analogous to the robustness test performed in section 5.3 using M0.

In the next chapter I will test the performance of the empirical codon model estimated from Pandit for ML phylogenetics. The Pandit model is the most general model estimated in this study and may be useful to a broad range of applications. However, it would be interesting to investigate more specialised

models like the Ensembl models. For example, non-overlapping Ensembl data sets such as human-dog and mouse-rat could be combined into one bigger data set or even specific models based on viral sequences could be estimated, once the consequences of applying an empirical codon model are better understood.

Furthermore I hope that this chapter also has demonstrated that the estimation of empirical codon model is a suitable tool to investigate questions about codon evolution. I have primarily here focused on the occurrence of multiple nucleotide changes, but other questions can be asked. Ideas include the usage of data sets of the studies of Keithley *et al.* [KLEW05] and Jordan *et al.* [JKA$^+$05]. An analysis of a instantaneous rate matrices from mouse-rat and human-chimp could shed new light on Keightley's hypothesis that natural selection is reduced in hominids compared to murids because of the low effective population size of hominids. By using a new feature in Dart which enables the estimation of irreversible Markov models (Ian Holmes, pers. communication), it will be possible to check Jordan's hypothesis of a universal trend in amino acid replacement but now on a codon level (see [JKA$^+$05] and [McD06] for a discussion).

Chapter 6

Empirical codon models in practice

6.1 Applications of empirical codon models in phylogenetics

In this chapter I will implement the empirical codon model of the previous chapter, estimated from the Pandit database, in phylogenetic software, to see if it is useful in the analysis of particular protein-coding sequences. In particular, different modifications of the empirical codon model estimated from Pandit will be compared to the standard mechanistic model M0.

Probabilistic modelling always aims to seek better models, and this study evaluates how well the empirical codon model describes the evolution of protein-coding DNA sequences. The empirical codon model could be an important contribution, because it is qualitatively different: it allows for single, double and triple nucleotide changes and acknowledges the importance of the genetic code. The ability of existing models to detect selection clearly relies on them correctly inferring evolutionary events, especially synonymous and nonsynonymous changes. The new empirical codon model could have an important influence on this since it permits these changes to be multiple nucleotide changes.

In ML phylogeny the empirical codon model which was estimated from Pandit data could simply be used in the same way that an unmodified Dayhoff, JTT or WAG model can be used for amino acid sequences. However, already for amino acid substitution models past experience showed that the performance of empirical models can be significantly improved by combining them with mechanistic parameters. For example, the amino acid frequencies can be used as additional free parameters, calculated from the data set analysed and used in place of the equilibrium frequencies of the database the matrix was estimated from (+F method; Cao *et al.* [CAJ$^+$94]).

Existing mechanistic codon models are based upon on parameters for codon frequencies π_i, for transition-transversion bias κ and for nonsynonmous-synonymous bias ω. The latter parameter plays a central role in selection studies. Additionally, we have seen in the previous chapter that the substitution patterns vary strongly for sequences with different ω values.

All this suggests that it will be beneficial to consider re-introducing mechanistic parameters π_i, κ and ω and the next section(s) discusses how this can be accomplished.

6.2 Re-introducing mechanistic parameters to empirical codon models

The selective forces acting upon a protein are highly informative about its biological function and evolutionary history [YB00]. For example, the interactions of proteins through their regulatory and metabolic networks are also reflected in the selection acting upon them. Recently, it was demonstrated that the more interactions a protein has with other molecules, the slower it evolves, and that proteins operating in complexes (e.g., involved in translation or DNA repair) are, on av-

erage, more restricted than those with simple house-keeping functions [AB05].

Models describing evolution at the codon level allow the estimation of selective forces acting on sequence alignment. The ratio of rates between nonsynonymous and synonymous substitutions, referred to as ω, is widely used as a direct measure of these forces. It can be used to detect when coding DNA is evolving neutrally, under negative (purifying) selection and under positive (adaptive) selection. When there are few selective pressures acting, sequences are said to be evolving neutrally and the relative rates of fixation of synonymous and nonsynonymous mutations are roughly equal (ω is approximately 1). When a protein has an important function its sequence is highly conserved through evolution and ω takes a value substantially less than 1. Conversely, when proteins are under pressure to adapt quickly to their environment, nonsynonymous changes are strongly selected for and ω will take a value greater then 1.

In the section 2.4 I described 'state-of-the-art' codon models allowing for variation of ω over sites, such as the mechanistic models M7, M8 or M8a. My research is eventually aiming at comparable empirical codon models which allow for variation in selective constraints over sites. However, as a first step towards this I investigate the estimation of the average selective forces across all sites of an alignment.

In the Chapter 5 it was shown that the strength of selection measured by ω with an M0 model has a strong relationship to observed substitution patterns. To capture this effect and to make empirical codon models suitable for selection studies, it is necessary to re-introduce a parameter measuring selective strength to the empirical codon model. However, in an empirical codon model ω can not be simply defined as a rate ratio anymore. Empirical codon models already reflect the synonymous-nonsynonymous bias which was present in the proteins

composing the database. In contrast, I use ω to represent the 'relative strength of selection' with respect to the 'average' ω given in the database. For a particular data set values greater than 1 should be interpreted as above the average and values less than 1 below average. All changes including multiple nucleotide changes will either result in one single synonymous or in one nonsynonymous change. This is reflected by using a single parameter ω.

The transition-tranversion bias, traditionally modelled by a single parameter κ, can be re-introduced in an empirical codon model as well. Analogous to ω, the parameter κ in empirical codon models represents a measure of 'the relative strength of the transition-transversion bias', with respect to the 'average' κ implicit in the database. The permitting of double and triple nucleotide changes in the empirical codon model, however, leads to new scenarios, which are different from the one transition or one transversion inherent in single nucleotide changes. Below is a list of the 9 possible ways to combine transitions (ts) and transversions (tv) in multiple nucleotide changes within one codon:

$$\text{1 nt change:} \quad (1\text{ts}, 0\text{tv}), (0\text{ts}, 1\text{tv}) \tag{6.1}$$

$$\text{2 nt change:} \quad (1\text{ts}, 1\text{tv}), (2\text{ts}, 0\text{tv}), (0\text{ts}, 2\text{tv}), \tag{6.2}$$

$$\text{3 nt change:} \quad (2\text{ts}, 1\text{tv}), (1\text{ts}, 2\text{tv}), (3\text{ts}, 0\text{tv}), (0\text{ts}, 3\text{tv}) \quad . \tag{6.3}$$

As a consequence, the parameter κ should rather be modelled as a function $\kappa(i, j)$ depending on the number of transitions n_{ts} and/or transversions n_{tv} of the change from codon i to codon j. Analogous to the definition of the mechanistic codon model M0 (Eq. 2.5) I define the instantaneous rate matrix of the empirical codon model with mechanistic parameters as:

$$q_{ij} = \begin{cases} 0 & \text{if } i \text{ or } j \text{ is a stop codon} \\ s^*_{ij}\pi_j\kappa(i,j) & \text{if } i \to j \text{ synonymous change} \\ s^*_{ij}\pi_j\kappa(i,j)\omega & \text{if } i \to j \text{ nonsynonymous change} \end{cases} \tag{6.4}$$

where s^*_{ij} are the exchangeabilities estimated from the Pandit database and π_j is the equilibrium frequency of codon j estimated from the particular data set analysed. Note that there is no longer any condition that i, j differ by exactly one nucleotide position, as required by the definition of the standard model M0.

Ten different models for $\kappa = \kappa(i,j)$ were devised and studied for this project:

1. **The 0κ-Model**: The factor κ is set to 1 for all changes:

$$\kappa(i,j) \equiv 1 \qquad .$$

   This model assumes that the transition-transversion bias is fully accounted for by the Pandit exchangeabilities s^*_{ij}, and does not vary significantly from one protein to another.

2. **The 1$\kappa-$Models** (1κ(a)-1κ(d)):

   (a) This approach models the occurrence of transitions only, and does not take account their multiplicity:

$$\kappa(i,j) = \begin{cases} \kappa & \text{if } n_{ts} \neq 0 \\ 1 & \text{if } n_{ts} = 0 \end{cases} \qquad .$$

   (b) This approach models the occurrence of transversions, but does not take account their multiplicity:

$$\kappa(i,j) = \begin{cases} \kappa & \text{if } n_{tv} \neq 0 \\ 1 & \text{if } n_{tv} = 0 \end{cases} \qquad .$$

   (c) In this model the multiplicity of transitions is incorporated by describing the effect in terms of a multiplicative κ:

$$\kappa(i,j) = \kappa^{n_{ts}} \qquad .$$

(d) In this model the multiplicity of transversions is incorporated by describing the effect in terms of a multiplicative κ:

$$\kappa(i,j) = \kappa^{n_{tv}} \quad .$$

The 1κ-Model (a) is similar to existing mechanistic codon models, where a maximum number of one transition is permitted. However, in standard mechanistic codon models we expect $\kappa > 1$. Here this is not longer the case, since κ is a measure relative to s^*_{ij}. The 1κ-Model (c) considers that the biasing effect introduced by multiple transitions may be multiplicative.

The 1κ-models (a) and (c) are similar to 1κ-models (b) and (d) except that they focus on transversions. This is unusual, but perhaps more natural in the same way that the standard ω parameter is generally considered a 'rate reducing' effect.

3. **The 2κ-Models** (2κ(a) and 2κ(b)):

   (a) This approach incorporates separate parameters, κ_1 and κ_2, for transitions as well as for transversions. The multiplicity of transitions and transversions is not considered:

$$\kappa(i,j) = \begin{cases} \kappa_1 & \text{if } n_{ts} \neq 0, n_{tv} = 0 \\ \kappa_2 & \text{if } n_{ts} = 0, n_{tv} \neq 0 \\ \kappa_1\kappa_2 & \text{if } n_{ts} \neq 0, n_{tv} \neq 0 \end{cases} \quad .$$

   (b) In this model transition and transversions are modelled with an individual parameter (κ_1 for transitions, κ_2 for transversions) and the effect is seen as multiplicative in terms of the relative rates:

$$\kappa(i,j) = \kappa_1^{n_{ts}}\kappa_2^{n_{tv}} \quad .$$

4. **The 3κ-Models** ((3κ(a) and 3κ(b)):

   (a) This model distinguishes between 1, 2 and 3 transitions:

$$\kappa(i,j) = \begin{cases} \kappa_1 & \text{if } n_{ts} = 1 \\ \kappa_2 & \text{if } n_{ts} = 2 \\ \kappa_3 & \text{if } n_{ts} = 3 \\ 1 & \text{otherwise} \end{cases} .$$

   (b) This model distinguishes between 1, 2, and 3 transversions:

$$\kappa(i,j) = \begin{cases} \kappa_1 & \text{if } n_{tv} = 1 \\ \kappa_2 & \text{if } n_{tv} = 2 \\ \kappa_3 & \text{if } n_{tv} = 3 \\ 1 & \text{otherwise} \end{cases} .$$

5. **The 8κ/9κ-Model**: All nine possible cases (listed in equations (6.1)-(6.3)) are modelled by an individual κ. This model is equivalent to a model with 8 κ's, since the rates are relative and can always be divided by one common factor.

Note that the 0κ-model is nested in all the other models. The 1κ-models (a) and (b) are nested in the 2κ-model (a), and the 1κ-models (c) and (d) are nested in the 2κ-model (b). Furthermore, the 1κ-model (a) is nested in 3κ-model (a), 1κ-model (c) is nested in 3κ-model (a), 1κ-model (b) is nested in 3κ-model (b) and 1κ-model (d) is nested in 3κ-model (b). Also, all the 1, 2 or 3κ-models are nested in the 8κ-model. The next section explains how all the above models were implemented in a standard phylogenetic software package.

6.3 Implementation in PAML

PAML is a package of programs for phylogenetic analyses of DNA and protein sequences using maximum likelihood, written and maintained by Ziheng

Yang [Yan94b]. The empirical codon models introduced in the previous section were implemented in the program *codeml* from release 3.14b of the package. The C-program *codeml* includes a routine *EigenQc* to calculate the instantaneous rate matrix Q. Previously, *EigenQc* only calculated instantaneous rate matrices from mechanistic rate matrices, and only permitted single nucleotide changes. I extended the *EigenQc* routine such that it allows for single, double and triple changes and reads in the empirically estimated exchangeabilities s^*_{ij} from an input file.

The part of the *EigenQc* routine which defines the κ parameter gets replaced by code implementing the different κ-models defined in section 6.2. Because one ω is sufficient to model nonsynonsymous changes in single, double and triple changes, the parts of the *EigenQc* routine treating the parameter ω stay the same. Likewise, the treatment of the codon frequencies remains unchanged.

PAML's optimisation routine conveniently requires only a vector containing all the parameters. This vector is constructed and initialised in the routine *GetInitials*. I have amended the *GetInitials* routine such that for each of the different κ-models it calculates the size of the parameter vector correctly and initialises all the parameters with a value 1.0 + a random number in (0,1]. Furthermore, I have restricted the values for κ and ω to a range of [0.0001,999] in the routine *SetxBound* and adapted the size of the κ-vector in the the routine *SetParameters*, since some models now require up to 8 κ-parameters instead of the single one used in all existing deterministic codon models.

Finally, the code was recompiled and run using the M0 options in the control file *codeml.ctl*.

6.4 ML Analysis

I have presented different versions of an empirical codon model combined with mechanistic parameters, and implemented all of these versions in the *codeml* program of the PAML package. The suite of 10 models was designed to cover a variety of possible relationships between parameters and 'real' sequence data, without preassumptions about what might fit the data best.

A preliminary small study confirmed that the likelihood score of the $8/9\kappa$-model was always best. However, the improvement it gave in likelihood values over any of the less parameter-rich models was never significant, clearly indicating that the $8/9\kappa$-model is 'over-parameterised'. Similarly, both of the 3κ-models generally were not significantly better than the 1κ- and 2κ-models (results not shown). Consequently, the ML analyses presented will focus on the 0κ-, 1κ- and 2κ-models and will compare them among each other and to the selection model M0 and the mechanistic single, double and triple (SDT) model of Whelan and Goldman [WG04].

I calculated the maximum likelihoods for 200 protein families and their associated tree topologies taken from release 17.0 of the Pandit data base using the models described above. A complete list of Pandit ids and log-likelihood values for all 200 protein families can be found in Appendix B. Equilibrium frequencies were estimated for each protein family using the '+F' method from Cao *et al.* [CAJ$^+$94]. For all 200 protein families the increase in likelihood when using an +F-model and a ω parameter instead of using an +F-model only is significant ($P < 0.05$). Table 6.1 shows likelihood results for the κ-models for five representative families from the PANDIT database. Table 6.2 summarises results of the bigger test set of 200 families from the PANDIT database.

PANDIT ID	**PF01056**	**PF01229**	**PF01481**	**PF02122**	**PF04163**
M0	-5335.15	-6718.81	-1533.40	-11145.31	-2499.28
0κ-model	-5244.12	-6610.25	-1458.36	-10861.97	-2410.96
(improvement over M0)	91.03	108.56	75.04	283.34	88.32
1κ-model (a)	-5234.25	-6604.24	-1458.11	-10860.23	-2410.12
(improvement over M0	100.90	114.57	75.29	285.08	89.16
and 0κ-model)	9.87*	6.01*	0.25	1.74	0.84
1κ**-model (b)**	-5232.94	-6601.98	-1458.13	-10859.32	-2410.13
(improvement over M0	102.21	116.83	75.27	285.99	89.15
and 0κ-model)	11.18*	8.27*	0.23	2.65*	0.83
1κ**-model (c)**	-5240.09	-6596.51	-1458.05	-10857.33	-2410.31
(improvement over M0	95.06	122.30	75.35	287.98	88.97
and 0κ-model)	4.03*	13.74*	0.31	4.64*	0.65
1κ**-model (d)**	-5220.72	-6596.29	-1457.25	-10857.88	-2410.66
(improvement over M0	114.43	122.52	76.15	287.43	88.62
and 0κ-model)	23.40*	13.96*	1.11	4.09*	0.30
2κ**(a)-model**	-5232.50	-6595.96	-1458.09	-10859.32	-2410.04
(improvement over M0,	102.65	122.85	75.31	285.99	89.24
0κ-model,	11.62*	14.29*	0.27	2.65	0.92
1κ-model (a)	1.75	8.28*	0.02	0.91	0.08
and 1κ-model (b))	0.44	6.02*	0.04	0.0	0.09
2κ**(b)-model**	-5214.75	-6595.48	-1454.77	-10856.38	-2410.30
(improvement over M0,	120.40	123.33	78.63	288.93	88.98
0κ-model,	29.37*	14.77*	3.59*	5.59*	0.66
1κ-model (c)	25.34*	1.03	3.28*	0.95	0.01
and 1κ-model (d))	5.97*	0.81	2.48*	1.5	0.36

Table 6.1: Log-likelihood values for protein families from Pandit under different mechanistic and empirical codon models. For nested models, an asterisk indicates that the increase in likelihood is statistically significant (P<0.05; $\chi^2_{1,0.05}$=3.84, $\chi^2_{2,0.05}$ = 6.84).

Tables 6.1 and 6.2 illustrate that the new empirical codon model clearly outperforms $M0$ for all κ models and for all five protein families shown, although the number of parameters of these models are comparable. Approaches relating transition-transversion bias to transversions only (1κ-models (b) and (d)) generally do better than approaches modelling transitions only (1κ-models (a) and

(c)). Models which consider the multiplicity of transition and transversions (1κ-models (c), (d) and 2κ-model (b)) generally perform better than models which only model the occurrence (or non-occurrence) of transitions and transversions (1κ-model (a) (b) and 2κ-model (a)).

Implementation	*Median rank*	*Interquartile range of ranks*
M0	8	[8, 8]
0κ model	7	[7, 7]
1κ-model (a)	5	[4, 5]
1κ-model (b)	4	[4, 5]
1κ-model (c)	5	[4, 6]
1κ-model (d)	3	[2, 5]
2κ-model (a)	3	[1, 3]
2κ-model (b)	1	[1, 2]

Table 6.2: Relative success of eight implementations of combined mechanistic and empirical codon models over a test set of 200 protein sequence alignments.

For each of the 200 families, the 8 models were ranked according to their maximum likelihood. This leads to each model version being assigned 200 ranks – potentially from 1 to 8 – according to its relative performance for the 200 protein families. Table 6.2 reports the medians and the interquartile rank of these ranks. Note that this table takes no account of the varying numbers of parameters estimated.

Among the 2κ-models, the model (b) does best. However, the interquartile ranges of the 2κ-models (b) and (a) overlap, such that in practice both should still be considered. Also the improvement in likelihood of the 2κ-model (b) in compar-

ison to the 0κ-model is only significant for 115 out of 200 families ($P<0.05$). In relation to the 1κ-models, this reduces to 102 significant cases for the comparison of the 2κ-model with the 1κ model (c) and 95 with 1κ-model (d).

Among the 1κ-models, the model (d) does best. But again only in 98 cases out of 200 is the improvement in log-likelihood from the 0κ-model to the 1κ-model significant.

Not all the κ models listed above are nested, so we cannot perform likelihood ratio tests to choose between certain ones of them. Choosing the model with the highest likelihood may lead to one that is unnecessarily complex, however, as it will often be the case that a more general model will have a higher likelihood than a more restricted model. The Akaike information criterion (AIC) is a method that tries to reach a compromise between goodness of fit and complexity of the model [Fel03]. We take the negative of twice the maximum log-likelihood for a model and penalise it by subtracting twice the number of parameters. So for model i with p_i parameters,

$$AIC_i = -2 \ln \hat{L}_i + 2p_i$$

we get quantities that can be compared among models i. One prefers the model that has the lowest value of AIC.

Table 6.3 lists the number of significant tests. Significance is determined by Akaike information criterion or by likelihood ratio test if possible and the models are nested.

	0κ-model 1 param.	1κ-model(a) 2 param.	1κ-model(b) 2 param.	1κ-model(c) 2 param.	1κ-model(d) 2 param.	2κ-model(a) 3 param.	2κ-model(b) 3 param.
0κ-model 1 param.		68 (54)	91 (67)	117 (91)	135 (98)	116 (95)	138 (115)
1κ-model(a) 2 param.	132 (–)		200 (n/a)	200 (n/a)	200 (n/a)	143 (106)	158 (n/a)
1κ-model(b) 2 param.	109 (–)	0 (n/a)		200 (n/a)	200 (n/a)	128 (95)	146 (n/a)
1κ-model(c) 2 param.	83 (–)	0 (n/a)	0 (n/a)		200 (n/a)	102 (n/a)	135 (102)
1κ-model(d) 2 param.	65 (–)	0 (n/a)	0 (n/a)	0 (n/a)		81 (n/a)	125 (95)
2κ-model(a) 3 param.	84 (–)	57 (–)	72 (–)	98 (n/a)	109 (n/a)		200 (n/a)
2κ-model(b) 3 param.	62 (–)	42 (n/a)	54 (n/a)	65 (–)	75 (–)	0 (n/a)	

Table 6.3: Number of protein families for which the likelihood was significantly improved out of a total of 200 protein families using the Akaike information criterion. For nested models the results of a LRTs are given in brackets; otherwise the LRT was not applicable (n/a).

In a nutshell, this suggests that if the new empirical codon model, as implemented here, is to be used as a replacement for models of the type M0, it seems best to consider all seven κ-models (including the 0κ-model) and choose among them using log-likelihood ratio tests on a per-dataset basis.

Overall, the likelihood improvements gained by using different κ-models were not convincing. This suggests that much of the transition-transversion bias is common to many proteins studied and it it quite well-modelled by the transition-transversion bias which implicitly already modelled in the empirical codon model without re-introducing a mechanistic κ. the small observed residual effect (i.e. some variation over data sets suggests that may be some slight extra effect transition-transversion effect was covered, which is varying between datasets and is probably not very well modelled by the κ-models tried. On the other hand the

little effect measured by the k-models could also be capturing something else.

Problems of model misspecification caused by rate heterogeneity over protein sites were already throughly discussed in this thesis (see Chapter 4 and 5) and here again some of the likelihood improvements may be related to this rather than an adequate description of the transition-transversion bias. Transition-transversion mutation bias varies both at the level of organisms and, to some extent, genes (e.g., mitochondrially encoded proteins are known to have elevated levels of bias [BPWW82]). I have checked the Pfam documentation files for any features in the families which had significantly improved families likelihoods. No relation between the organisms and likelihood performance could be identified. Furthermore I have checked the significant protein families for common structural and functional features (e.g., transmembrane proteins would be an obvious case). The dataset of 200 protein families is enriched for ribosomal proteins (15 out of 200) and the likelihood for the trees of most of these (12 out of 15) were significantly improved by using one or other (typically many) κ models. However, I could not identify why this should be the case. For a more systematic analysis of the relation between significant improvement in likelihood and protein function information from the GO database [CMB$^+$03] could be considered in the future.

Finally, I like to compare the empirical codon model to the mechanistic model SDT model of Whelan and Goldman [WG04]. The SDT model is a mechanistic codon model which uses the parameters σ, δ and τ to try to model protein-coding sequence evolution at the codon level, allowing for double and triple substitutions in proportions estimated on a per-dataset basis. Furthermore, the SDT model considers one κ parameter for transition-transversion bias on the nucleotide level, two parameters ω_1 and ω_2 for nonsynoymous-synonymous substitution bias in simple changes and changes spanning over codon boundaries, and parameters for

codon frequencies. To do a fair comparison with SDT, we need to change the method used to calculate the equilibrium codon frequencies within the empirical codon model. Instead of treating the frequencies as 61 free parameters, they can be calculated from the average nucleotide frequencies (F1x4) or from the average nucleotide frequencies at the three codon positions (F3x4) [Yan97]. However, the SDT model, in common with the model in Muse and Gaut [MG94], assumes that the instantaneous rates of change are proportional to the frequency of the altered nucleotide(s), and not proportional to the frequency of the codon observed. This method to determine the codon frequencies is referred to as F1x4MG in the PAML documentation [Yan97].

The comparison with the SDT model was restricted to a total of 15 families, because only the log-likelihood values published in the SDT paper were available. Additionally, the SDT study was performed on an older version of Pandit, so we only could use protein families whose DNA sequences were identical in both versions.

Out of the 15 protein families studied, PF01056 is the only protein family for which SDT is better than the empirical codon models. For all other families the empirical codon models perform better. Log-likelihoods for κ-models using F1x4MG model are worse than the +F model (compare Tables 6.1 and 6.4). However, the κ-models still perform better than M0, and the overall picture amongst the κ-models remains unchanged. Table 6.4 shows the log-likelihoods calculated combining M0, SDT and all κ-models with F1x4MG for five representative families. In the cases studied, the performance of SDT lies between M0 and the κ-models. This suggests that SDT was a good attempt at modelling a real effect. However, SDT could be improved as the performance of the empirical model shows.

PANDIT ID	**PF01056**	**PF01229**	**PF01481**	**PF02122**	**PF04163**
M0	-5483.54	-6865.90	-1593.52	-11256.83	-2630.13
SDT	-5360.42	-6818.44	-1549.32	-11119.16	-2596.90
0κ-model	-5397.32	-6770.95	-1521.86	-11007.45	-2539.89
(improvement over M0	86.22	94.95	71.66	249.38	90.24
and SDT)	*-36.9*	47.49	27.46	111.71	57.01
1κ-model (a)	-5387.47	-6765.32	-1521.78	-11005.36	-2539.88
(improvement over M0,	96.07	100.58	71.74	251.47	90.25
SDT and	*-27.05*	53.12	27.54	113.8	57.02
0κ-model)	9.85*	5.63*	0.08	2.09*	0.01
1κ**-model (b)**	-5382.26	-6763.03	-1521.83	-11001.88	-2539.55
(improvement over M0,	101.28	102.87	71.69	254.95	90.58
SDT and	*-21.84*	55.41	27.49	117.28	57.35
0κ-model)	15.06*	7.92*	0.03	5.57*	0.34
1κ**-model (c)**	-5392.67	-6756.74	-1521.75	-11006.57	-2539.80
(improvement over M0,	90.87	109.16	71.77	250.26	90.33
SDT and	*-32.25*	61.70	27.57	112.59	57.1
0κ-model)	4.65*	14.21*	0.11	0.88	0.09
1κ**-model (d)**	-5373.78	-6756.16	-1521.04	-11000.08	-2539.03
(improvement over M0,	109.76	109.74	72.48	81.04	91.1
SDT and	*-13.36*	62.28	28.28	119.08	57.87
0κ-model)	23.54*	14.79*	0.82	1.11	0.86
2κ**(a)-model**	-5382.20	-6753.29	-1521.77	-11000.97	-2538.34
(improvement over M0	101.34	112.61	71.75	255.86	91.79
SDT,	*-21.78*	65.15	27.55	118.19	58.56
0κ-model,	15.12*	17.66*	0.09	6.48	1.55
1κ-model (a)	5.27*	12.03*	0.01	4.39	1.54
and 1κ-model (c))	10.47*	3.45*	0.06	5.60	1.46
2κ**(b)-model**	-5367.29	-6750.14	-1518.76	-10996.44	-2537.15
(improvement over M0	116.25	115.76	74.76	260.39	92.98
SDT,	*-6.87*	68.30	30.56	122.72	59.75
0κ-model,	30.03*	20.81*	3.10	11.01*	2.74
1κ-model (b)	14.57*	12.89*	3.07*	5.44*	2.4*
and 1κ-model (d))	6.49*	6.02*	2.28*	3.64*	1.88

Table 6.4: Log-likelihood values for protein families from Pandit under different mechanistic and empirical codon models using F1x4MG to determine codon frequencies. Negative italic numbers represent decreases in likelihood.

6.5 Conclusion and future work: ML analysis

In this chapter the empirical codon model estimated from Pandit data was tested for utility in phylogenetic analysis. Past experience and results in previous chapters suggested that it would be beneficial to consider combining some mechanistic parameters with the pure empirical codon model. The choice of parameters was oriented towards those used in existing mechanistic codon models used for the detection of selection: the codon frequencies π_i, κ parameters to model transition-transversion bias, and ω modelling nonsynonymous-synonymous bias were re-introduced to empirical codon models and various combined models successfully implemented in PAML. The various versions of κ-models differ in their treatment of the transition and transversions, resulting from new scenarios which occurred because instantaneous single, double and triple changes are permitted in the empirical codon model.

In general, the empirical models outperform mechanistic models M0 and the SDT. Considering 1κ- and 2κ-models can further improve log-likelihoods significantly. This proves that the empirical codon is suitable for usage in phylogenetic analysis, and since recently codon models have become an option in phylogenetic reconstruction despite their computational burden [RTY05], I hope that the empirical codon models will be used for this purpose.

Furthermore this ML study counts as major progress towards an empirical selection model. Two further steps will be necessary: firstly, the empirical codon model needs to cover the modelling of variation of ω amongst sites. This can be achieved by amending PAML code such that the M7 and M8 models can also be used in combination with the empirical codon model.

Secondly, the meaning of selection parameter ω needs to be re-defined. Es-

timates obtained from mechanistic codon models, denoted ω_M, and estimates from empirical codon models, ω_E, cannot be compared directly. The parameter ω_M represents the nonsynonymous-synonymous rate ratio, while ω_E measures the 'relative strength of selection' with respect of an 'average' ω implicit in the Pandit database. To compare both we need to untangle the value of ω_E from its value for neutral evolution.

To do this, we take an approach which was pursued in the very early mechanistic codon model of Yang and Goldman [GY94]. There, the ratio of the instantaneous rates per codon of synonymous and nonsynonymous nucleotide substitutions is calculated. The synonymous substitution rate is given by

$$\rho_s = \sum_{i=1}^{61} \sum_{\substack{j=1 \\ j \neq i \\ aa_i = aa_j}}^{61} \pi_i q_{ij} \quad .$$

The nonsynonymous rate per codon ρ_a can be calculated by $\rho_a = 1 - \rho_s$, since the overall rate is normalised to 1.

We can also take the values $\rho_s^{neutral} = 0.21$ and $\rho_a^{neutral} = 0.79$ from Goldman and Yang [GY94], first derived by Nei and Gojobori [NG86] as typical values for neutrally evolving proteins. Thus the 'corrected' nonsynonymous-synonymous rate ratio $\bar{\omega}_E$ is given by

$$\bar{\omega}_E = \frac{\rho_a \rho_s^{neutral}}{\rho_s \rho_a^{neutral}} \quad .$$

Note that $\bar{\omega}_E$ implicitly depends on ω_E since ρ_s and ρ_a are functions of ω_E.

Table 6.5 lists $\bar{\omega}_E$ values calculated in this way from the 2κ-model(b) and ω_M values estimated with the M0 model, for the five representative protein families used previously.

PANDIT ID	**PF01056**	**PF01229**	**PF01481**	**PF02122**	**PF04163**
$\bar{\omega}_E$	0.0698	0.1049	0.0365	0.1977	0.0172
ω_M	0.0578	0.1011	0.0761	0.2997	0.0176

Table 6.5: Nonsynonymous-synonymous rate ratios for five Pandit families, estimated using the empirical 2κ-model(b) ($\bar{\omega}_E$) and the mechanistic model M0 (ω_M; see text for details).

As expected the ω_M values for M0 and $\bar{\omega}_E$ for 2κ-model(b) are not the same. However, the comparison indicates that the rate ratios measured for the empirical codon model appear broadly comparable with those calculated for M0. All five families indicate strong purifying selection.

The pioneering studies examining the effect of natural selection in proteins estimated an average ω (or equivalent value) across all the sites in an alignment, and they usually found that adaptive evolution is rare and most proteins are under strong purifying selection (e.g., [EIG96]). The lack of positive selection suggested by these studies is most probably an underestimate of the true amount. The contents of a genome have been evolving for millions of years and are highly adapted to the functions they perform. Consequently, purifying selection will have been acting on the majority of sites in a protein to maintain this function and the average value of ω would be expected to be below 1. Positive selection would normally be expected to affect only a few key residues in a protein, to successfully find its footprint during molecular evolution requires a more sophisticated approach. In the future, I therefore plan to investigate the consequences of empirical codon models in selection studies using models comparable to M7 and M8, which make allowance for ω varying across sequence positions.

The transition from M0 to M7- and M8- type models is not only important for applications of the empirical codon model in selection studies. Once we understand empirical codon models within this framework it should also be possible to go back to the matrix estimation itself and re-estimate the exchangeabilties considering site-specific rate variation. This could potentially solve the problems of model misspecification which preoccupied a good share of this thesis.

Chapter 7

Conclusion

This thesis has contributed to the understanding of the processes of evolution acting on protein sequences, and their mathematical modelling.

In probabilistic mathematical modelling, the evolutionary process is often described with a Markov model. In this thesis I have investigated some of the fundamental methods to infer historically important amino acid models. In the past, various amino acid and codon models have been developed. However, even though there were multiple existing models, there were no good ways of abstracting important and intuitive features and comparing them across models. I have developed the AIS method, which is a tool to compare instantaneous rate matrices of different Markov models in a biologically meaningful way. The AIS method was successfully applied to existing amino acid and codon models. While I do not expect to undertake further development of the AIS method, I expect to use it in future to compare newly devised models.

The work on the AMP model has improved understanding of the relationship between the amino acid and codon models. It showed that that the nature of protein-coding sequence evolution is such that modelling on the codon level seems reasonable, but leads to non-Markovian behaviour on the amino acid level. The AMP was able to give a rational explanation of the time-dependent behaviour of

amino acid sequences described in the literature.

Consequently, the rest of the thesis focused on codon sequence evolution. I have estimated an empirical codon model from the Pandit database using ML techniques. Results indicated that the modelling of codon sequences evolution is improved by allowing for single, double and triple nucleotide changes. Applying the AIS method to the empirical codon model revealed that the genetic code and the physico-chemical properties of the amino acids drive the process of codon sequence evolution. Multiple nucleotide changes, as well as the influence of the physico-chemical properties of the amino acids encoded by the codons, are not part of standard mechanistic codon models.

To deepen the understanding of this finding I extended the estimation of empirical codon matrices to genomic data taken from the Ensembl database. I observed different substitution patterns for different evolutionary distances; with increasing evolutionary distance, instantaneous rates for multiple nucleotide changes were higher. This effect cannot be explained by alignment artifacts, but by selection acting differently on slow- and fast-evolving genes. I have shown that the synonymous-nonsynonymous rate ratio ω is a suitable measure to classify different substitution patterns represented by the matrices.

Interestingly, this also sheds new light on work on the AMP model. The AMP model was based on the mechanistic codon model M0 in which selection was only considered by a single parameter ω. It would be very interesting to re-do the analysis using an empirical codon model allowing for variation in ω.

The dependency of the substitution patterns on ω also has consequences for the understanding of observed substitution patterns. Double and triple nucleotide changes do appear to occur. They seem instantaneous but in fact are probably mostly not, as the study of very constrained genes has shown. Instead, if puri-

fying selection is weak enough then compensatory changes on a fast time scale (population level rather than species level) can be fixed, even if via a deleterious intermediate codon. On the other hand, if purifying selection is strong then only compensatory changes via synonymous intermediates occur. A more complete understanding could be achieved by estimating matrices from even more closely related sequences such as human-chimpanzee or human intraspecies data, when these are available in sufficient quantity and quality in the future. It would also be interesting to clarify the effects of larger scale events such as gene conversion. A simulation study in this thesis suggested that effects that are compatible with gene conversion (or other large scale events) are small but significant.

Substitution models are widely used in sequence alignment, phylogenetics, database searches and comparative genomics, and the empirical codon model could be potentially useful in all those fields. In this thesis, I have focused on testing the utility of the empirical codon model estimated from Pandit data in ML phylogenetics. This model was preferred to the empirical codon models estimated from genomic data because it was derived from a broad range of different species, with individual data sets more like those encountered in other phylogenetic studies. Past experience and the results described above suggested that it would be beneficial to consider combining some mechanistic parameters with the pure empirical codon model. An ω-parameter modelling nonsynonymous-synonymous substitution bias and κ-parameter(s) modelling transition-transversion bias were re-introduced to the empirical codon model estimated from Pandit. The different versions of the empirical codon model outperformed previous mechanistic models.

Motivated by the good performance of the empirical codon models, I have started to explore whether empirical codon models are suitable for selection studies. The ω-parameter which was re-introduced into the empirical codon model,

represents the 'relative strength of selection' with respect of an 'average' ω implicit in the Pandit database. However, when this ω was corrected using its value under neutral evolution, it became comparable with values estimated with the mechanistic model M0.

Empirical codon models could have important implications for selection studies, because they are qualitatively different to previously-used mechanistic models: they permit multiple nucleotide changes and incorporate the influence of the physico-chemical properties of the amino acids encoded by the codons. The next step is to extend the empirical codon models to models allowing for variation in selective constraints over sites and then investigate their behaviour on large datasets which have been well studied with mechanistic models before.

From the point of view of selection studies this thesis is the foundation and only a starting point of using empirical codon models in many future studies. Once the consequences of using codon models for selection studies are better understood it will be interesting to devise more specialised models, ranging, e.g., from models suitable for mammalian sequences estimated from the genomic data sets, to models suitable for viral sequences.

Appendix A

Different versions of the Dayhoff rate matrix

In 1972 Dayhoff and colleagues introduced a Markov model to protein evolution. The Dayhoff model was a historical step towards the probabilistic modelling of protein sequences (see section 1.1). Dayhoff et al.*'s methodology influenced estimation of generation of matrices to follow (see section 5.2). It only recently was replaced by modern ML approaches (see section 5.2 and [AH96, WG01, MV00, HR02]). Today, the Dayhoff model remains important because it still widely used as a standard to test and compare newly devised code. In this context I used instantaneous rate matrices of the Dayhoff model which I accidently copied from two different sources at an early stage of my Ph.D. Surprisingly, I received slightly different answers for the different copies. Furthermore, the instantaneous rate matrix did not only differ for these two copies, but I found numerous slightly different versions of the Dayhoff's rate matrix throughout widely used standard phylogenetic software packages. Nick Goldman encouraged me to follow the problem up in a detailed study. Since this study is not part of the connected argument of this thesis, it appears in this appendix section. The contents of Appendix A was published under the title "Different versions of the Dayhoff rate matrix" in*

Molecular Biology and Evolution 22: 193–199, 2005, by C. Kosiol and N. Goldman.

Author contributions: *C.K. and N.G. reconstructed Dayhoff's estimation method for probability matrices from the original papers. C.K. calculated instantaneous rate matrices from probabilities using eigen-decomposition as well as various direct methods and compared the resulting different rate matrices with an ML performance analysis. N.G. illustrated the relationship between the different methods in Fig.1. C.K. and N.G. wrote the paper.*

RESEARCH ARTICLES

Different Versions of the Dayhoff Rate Matrix

Carolin Kosiol and Nick Goldman

EMBL-European Bioinformatics Institute, Hinxton, United Kingdom

Many phylogenetic inference methods are based on Markov models of sequence evolution. These are usually expressed in terms of a matrix (Q) of instantaneous rates of change but some models of amino acid replacement, most notably the PAM model of Dayhoff and colleagues, were originally published only in terms of time-dependent probability matrices ($P(t)$). Previously published methods for deriving Q have used eigen-decomposition of an approximation to $P(t)$. We show that the commonly used value of t is too large to ensure convergence of the estimates of elements of Q. We describe two simpler alternative methods for deriving Q from information such as that published by Dayhoff and colleagues. Neither of these methods requires approximation or eigen-decomposition. We identify the methods used to derive various different versions of the Dayhoff model in current software, perform a comparison of existing and new implementations, and, to facilitate agreement among scientists using supposedly identical models, recommend that one of the new methods be used as a standard.

Introduction

Dayhoff and colleagues (Dayhoff, Eck, and Park 1972; Dayhoff, Schwartz, and Orcutt 1978[1]) introduced a Markov model of protein evolution that resulted in the development of the widely used amino acid replacement matrices known as the PAM matrices. In these articles, the protein evolution model is only given in the form of probability matrices relating to specific levels of sequence divergence (e.g., PAM1 and PAM250 in DSO78). In many phylogenetic inference methods (including pairwise distance estimation, maximum likelihood phylogeny estimation and Bayesian analysis of phylogeny), however, it is necessary to be able to compute this probability matrix (which we denote $P(t)$) for any real evolutionary time (distance) $t \geq 0$ (Felsenstein 2003). This is achieved using an instantaneous rate matrix (IRM), often denoted $Q = (q_{ij})_{i,j=1,\ldots,20}$, which is related to $P(t)$ via

$$P(t) = \exp(tQ) = I + tQ + \frac{(tQ)^2}{2!} + \frac{(tQ)^3}{3!} + \cdots \quad (1)$$

(Liò and Goldman 1998). The probability matrix $P(t)$ for any time $t > 0$ is fully determined by the IRM Q, which is itself independent of t.

In this article we summarize different methods that have been used to construct IRMs from the probability matrices of DSO78, identify their implementations in standard phylogenetic software packages, and indicate ways in which they lead to different and potentially inaccurate IRMs. We describe two simpler methods for deriving an IRM from information such as that published by Dayhoff and collaborators. We then compare the performance of all of these implementations using a test set of 200 protein domain families.

[1] For brevity, we generally refer to these publications as the work of "Dayhoff and colleagues." The latter, abbreviated to "DSO78," is better known and contains the data used for all implementations of the Dayhoff model, but we note that all important methodology was introduced in the former.

Key words: amino acid replacement, Dayhoff matrix, Markov models, phylogenetic inference, protein evolution.

E-mail: goldman@ebi.ac.uk.

Mol. Biol. Evol. 22(2):193–199. 2005
doi:10.1093/molbev/msi005
Advance Access publication October 13, 2004

The Dayhoff model and other similar models (e.g., the JTT model of Jones, Taylor, and Thornton 1992) have been very influential in molecular phylogenetics, database searching, and other fields, and they continue to be widely and regularly used. It is important to have a complete understanding of how these models can be accurately derived from raw data. In the interests of (1) remaining faithful to the information collected and published by Dayhoff and colleagues and others, (2) keeping computations as simple and accurate as possible, and (3) facilitating agreement among scientists using different implementations of supposedly identical methods, we propose a standardization of these IRMs in phylogenetic software.

Methods

Markov Models for Protein Evolution

Proteins are sequences of amino acids. The Markov model asserts that one protein sequence is derived from another during evolution by a series of independent substitutions, each changing one amino acid in the ancestral sequence to another in its descendant. We assume independence of evolution at different sites.

The continuous-time Markov process is a stochastic model in which $(P(t))_{ij}$ gives the probability that amino acid i will change to amino acid j at any single site after any time $t \geq 0$. Since there are 20 amino acids, i and j take the values 1, ... ,20. The 20×20 probability matrix is calculated as $P(t) = \exp(tQ)$ (eq. 1), where the matrix Q is independent of time in the Markov processes typically used in molecular phylogenetics. Q has off-diagonal entries $q_{ij,i\neq j}$ equal to the instantaneous rates of replacement of i by j, and diagonal entries defined by the mathematical requirement that each row sum is 0.

Associated with a matrix Q are equilibrium frequencies of the 20 amino acids, denoted f_i, and mutabilities, m_i, defined as the rate at which amino acid i changes to any other amino acid: $m_i = \sum_{j\neq i} q_{ij}$. Typically in phylogenetic applications, Q is normalized so that the mean rate of replacement at equilibrium ($\sum_i \sum_{j\neq i} f_i q_{ij}$ or $\sum_i f_i m_i$) is 1, meaning that times t are measured in units of expected numbers of changes per site. Here, we simplify our notation by omitting the trivial multiplicative constants

that achieve this normalization. More detailed descriptions of Markov models in molecular phylogenetics are given by Liò and Goldman (1998), Felsenstein (2003), and Thorne and Goldman (2003).

Dayhoff and colleagues devised a method to estimate $P(t)$ that relied on the comparison of closely related pairs of aligned sequences. They selected pairs of sequences that were 85% identical and counted the differences between them. In general these differences underestimate the actual numbers of changes because when multiple changes occur at a single site they are observed as at most single replacements. However, if sufficiently closely related sequences are used, then the probability of these "multiple hits" is reduced. Dayhoff and colleagues assumed that at 85% sequence identity there are no multiple hits (see also Jones, Taylor, and Thornton 1992; Goldman, Thorne, and Jones 1996, 1998). The cost of this 85% rule is some loss in accuracy and inefficient use of data, because divergent sequence pairs have to be discarded.

Advances in methodology and computer speed have made it possible to estimate Markov models of amino acid replacement without any constraints on the levels of divergence between the sequences (e.g., Adachi and Hasegawa 1996; Yang, Nielsen, and Hasegawa 1998; Adachi et al. 2000; Müller and Vingron 2000; Whelan and Goldman 2001; Devauchelle et al. 2001; Dimmic et al. 2002; Veerassamy, Smith, and Tillier 2003). While these methods may now give superior results, less sophisticated methods based on simple counts of changes under the 85% rule are still used (e.g., Goldman, Thorne, and Jones 1996, 1998) and were the basis for two of the most widely used evolutionary models in molecular phylogenetics (DSO78; Jones, Taylor, and Thornton 1992). The rest of our analysis concentrates solely on cases such as these, where differences between sequence pairs can be counted and assumed to accurately represent actual evolutionary events.

We write the observed number of occurrences of sites in all aligned sequence pairs with amino acids i in one sequence and j in the other as n_{ij}. The data collection technique of Dayhoff and colleagues leads to the relationships $n_{ij} = n_{ji}$, because the direction of an evolutionary change cannot be determined from the observation of two contemporary sequences. (As a consequence, equilibrium is assumed and resulting models must be reversible; see Liò and Goldman 1998.) Because we assume the n_{ij} accurately represent all evolutionary events, an intuitive estimate for the rate of change of i to j ($j \neq i$) is given by the number of such events as a proportion of all observations of i:

$$q_{ij} = \frac{n_{ij}}{\sum_k n_{ik}}. \tag{2}$$

Corresponding estimators for the m_i and f_i are given by

$$m_i = \frac{\sum_{j \neq i} n_{ij}}{\sum_k n_{ik}} \tag{3}$$

and

$$f_i = \frac{\sum_j n_{ij}}{\sum_k \sum_l n_{kl}} \tag{4}$$

(see also Goldman, Thorne, and Jones 1996).

Equation (2) represents a simple way to estimate the IRM Q directly from counts of observed differences between closely related sequence pairs and was used by Goldman, Thorne, and Jones (1996, 1998). Applications of the models of Dayhoff and colleagues and Jones, Taylor, and Thornton (1992) in molecular phylogentics have, however, used a different approach, estimating Q not directly but via corresponding probability matrices $P(t)$. We proceed by describing these methods.

The Dayhoff Model

Dayhoff and colleagues published probability matrices based on counts of sequence differences, frequencies, and mutabilities. These articles include the values of the counts n_{ij}, but in an incomplete manner: the numbers of positions containing amino acid i in both sequences, n_{ii}, are omitted.

The mutabilities m_i were calculated slightly differently from equation (3) above. The data from individual sequence pairs $s = 1, \ldots, S$ were considered separately, and the estimator

$$\begin{aligned} m_i &= \frac{\sum_{s=1}^{S} \left(\sum_{j \neq i} n_{ij}^{(s)} \right)}{\sum_{s=1}^{S} \left(\sum_g \sum_{h \neq g} n_{gh}^{(s)} / N_s \right) \left(\sum_k n_{ik}^{(s)} \right)} \\ &= \frac{\sum_{j \neq i} n_{ij}}{\sum_{s=1}^{S} \left(\sum_g \sum_{h \neq g} n_{gh}^{(s)} / N_s \right) \left(\sum_k n_{ik}^{(s)} \right)} \end{aligned} \tag{5}$$

used. Here, the observed counts from each sequence pair s are denoted $n_{ij}^{(s)}$ and N_s is the number of sequence positions in pair s (so $\sum_s n_{ij}^{(s)} = n_{ij}$ and $\sum_{i,j} n_{ij}^{(s)} = N_s$). Note that the numerators of the estimators given by equations (3) and (5) are the same. The denominators differ only by the use of the weighting factor $\sum_g \sum_{h \neq g} n_{gh}^{(s)} / N_s$ for each sequence pair s. Dayhoff and colleagues described this as an estimator of the "exposure to evolution" or evolutionary separation of each sequence pair. There is no reason why this factor is necessary, as the evolutionary separation of pair s does not make the changes $n_{ij}^{(s)}$ any more or less indicative of typical protein evolution than those observed in any other pair. Nevertheless, the weighting factor introduces no systematic bias (it will not be correlated with the relative values of the $\sum_k n_{ik}^{(s)}$ term it is applied to) and analysis (not shown) of the expectations of the numerators and denominators of equations (3) and (5) indicates that both ratios give reasonable estimators of mutability.

Dayhoff and colleagues used much the same idea to estimate the frequencies f_i of the amino acids, combining the data for each comparison s to derive the following estimator:

$$\begin{aligned} f_i &= \frac{\sum_{s=1}^{S} \left(\sum_g \sum_{h \neq g} n_{gh}^{(s)} / N_s \right) \left(\sum_j n_{ij}^{(s)} \right)}{\sum_{s=1}^{S} \left(\sum_k \sum_l n_{kl}^{(s)} \right)} \\ &= \frac{\sum_{s=1}^{S} \left(\sum_g \sum_{h \neq g} n_{gh}^{(s)} / N_s \right) \left(\sum_j n_{ij}^{(s)} \right)}{\sum_k \sum_l n_{kl}}. \end{aligned} \tag{6}$$

This agrees with equation (4) above, again apart from the unneeded weighting factor $\sum_g \sum_{h \neq g} n_{gh}^{(s)} / N_s$. Dayhoff and colleagues published the values they computed for the

m_i and f_i. It is not possible to work out Dayhoff and colleagues' missing n_{ii} values from their published $n_{ij,i\neq j}$, m_i and f_i.

Dayhoff and colleagues described an estimator $P_D = (p_D)_{ij}$ for probability matrices as follows:

$$(p_D)_{ij} = \begin{cases} \dfrac{cm_i n_{ij}}{\sum_{k\neq i} n_{ik}} & \text{if } i \neq j \\ 1 - cm_i & \text{if } i = j \end{cases} \quad (7)$$

where c is a constant. The quantity $\delta = 1 - \sum_i f_i (p_D)_{ii}$ then gives the proportion of amino acids that are observed to differ after the evolutionary interval represented by the matrix P_D and depends on the choice of c:

$$\delta = 1 - \sum_i f_i (p_D)_{ii} = 1 - \sum_i f_i (1 - cm_i)$$
$$= 1 - \sum_i f_i + c \sum_i f_i m_i = c \sum_i f_i m_i. \quad (8)$$

To create the PAM1 matrix, Dayhoff and collaborators chose c so that δ is 1 observed mutation per 100 sites. Thus

$$c = \frac{\delta}{\sum_i f_i m_i} = \frac{0.01}{\sum_i f_i m_i}, \quad (9)$$

and we may write PAM1 $= P_D(\delta = 0.01)$. Dayhoff and colleagues also defined matrices PAMn, equal to $(\text{PAM1})^n$ (*not* $P_D(\delta = n/100)$).

The "time" at PAM1 is defined by $\delta = 1$ observed amino acid change per 100 amino acids, i.e., excluding changes that are unobservable because of multiple hits. Therefore, PAM1 does not correspond to a probability matrix $P(0.01)$ calculated according to equation (1), but to some $P(0.01 + \epsilon)$, where ϵ is the extra time needed to account for unobserved substitutions. For small times t, the difference is negligible (e.g., PAM1 approximates $P(0.01 + 6.85 \times 10^{-5})$ when using the IRM derived with the DCMut method described below); even for larger times, the difference is still small (PAM250 is close to $P(2.52)$ under the DCMut model).

Dayhoff and colleagues' estimator embodied in equations (7)–(9) makes two assumptions: that the data from which the observed counts n_{ij} are derived are taken to be sufficiently closely related to exclude multiple hits (the 85% rule), and that there will be no multiple hits in the probability model defined by P_D. The latter assumption is approximately true for $\delta = 0.01$, and becomes more accurate as δ decreases. For larger values of δ it is less accurate; if δ is large enough, the resulting matrix P_D will not even be a valid probability matrix (see fig. 1).

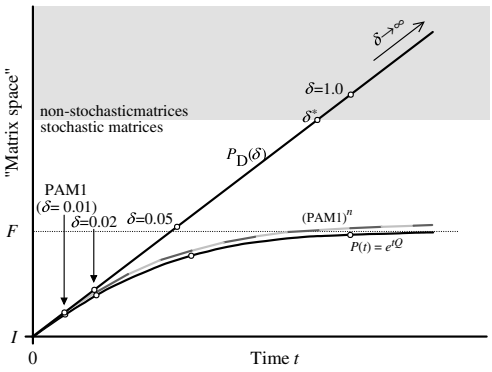


FIG. 1.—Schematic diagram showing relationships of matrices described in the text. The y-axis represents the unbounded 400-dimensional space of 20×20 matrices, with the region of valid stochastic matrices (all elements $\in [0,1]$ and all row sums $= 1$; a bounded 380-dimensional subspace) and its complement (matrices that are not stochastic) indicated. Time t measures evolutionary distance. The lowest (smooth) curve represents matrices $P(t)$ describing amino acid replacement probabilities generated from a particular instantaneous rate matrix Q according to $P(t) = \exp(tQ)$. When $t = 0$, no replacements have occurred and $P(t)$ equals the identity matrix I. As $t \to \infty$, $P(t)$ converges to F, the matrix with each row equal to the equilibrium distribution of Q (i.e., $F_{ij} = f_j$). Q can be recovered from $P(t)$ for any time t, using the procedure described by Kishino, Miyata, and Hasegawa (1990; equation (10) in this article). The uppermost (straight) line indicates the matrices $P_D(\delta)$ generated by the procedure of Dayhoff and colleagues, embodied in equations (7)–(9), which is "linear" in the sense that it assumes that no multiple hits occur. For sufficiently small values of δ, $P_D(\delta)$ is a reasonable approximation to $P(t)$ and the method of Kishino, Miyata, and Hasegawa (1990) applied to $P_D(\delta)$ can give a close approximation to Q (see *From Probability to Rate Matrix via Eigen-Decomposition*). Alternatively, *any* matrix on the uppermost line (i.e., $P_D(\delta)$ for any δ) can be used to recover Q using the method embodied in equation (14), including matrices $P_D(\delta)$, which are not stochastic matrices (there is a value $\delta^* < 1$ such that at least one diagonal element of $P_D(\delta)$ is < 0 for all $\delta \geq \delta^*$). Dayhoff and colleagues' PAM1 matrix lies on the uppermost line and corresponds to $\delta = 0.01$. Matrices denoted PAMn ($n > 1$) by Dayhoff and colleagues are defined to equal $(\text{PAM1})^n$; consequently, each remains close to $P(n/100)$. This is illustrated by the middle line, which shows $(\text{PAM1})^n$ for increasing powers of n (note the piecewise linearity of this line, shown in alternating lighter and darker shades, which represents n increasing in integer steps). A color version of this graph is available in the Supplementary Material online.

From Probability to Rate Matrix via Eigen-Decomposition

Kishino, Miyata, and Hasegawa (1990) described a method to obtain an IRM matrix from the probability matrix PAM1. The key idea of this method is the relationship of the eigenvalues and eigenvectors of matrices $P(t)$ and Q related by equation (1). If P has eigenvalues ρ_i and eigenvectors u_i ($i = 1, \ldots, 20$), and defining also $\lambda_i = \log \rho_i$, $U = (u_1, \ldots, u_{20})$ and $\Lambda = \text{diag}\ (\lambda_1, \ldots, \lambda_{20})$ (the diagonal matrix with entries λ_i), then the IRM Q can be calculated as

$$Q = U \Lambda U^{-1}. \quad (10)$$

This method is appropriate for recovering Q if P is generated according to $P(t) = \exp(tQ)$ for any specific t. However, Dayhoff and colleagues' approach (equations 7–9) generates a matrix $P_D(\delta)$ which is only an approximation to $P(t)$ because multiple hits occurring over the time period corresponding to δ are neglected. (Indeed, the PAM1 matrix cannot be generated as $\exp(t^*Q^*)$ for any valid IRM Q^* and time $t^* \geq 0$—proof not shown.) This approximation is increasingly poor as δ increases (fig. 1). Kishino, Miyata, and Hasegawa (1990) adopted $\delta = 0.01$, to generate PAM1 as in DSO78. It turns out that this value of δ is not small enough. Using this eigen-decomposition method, we have calculated the IRMs Q for δ in the range $[3 \times 10^{-7}, 3 \times 10^{-1}]$, and figure 2 shows that these can

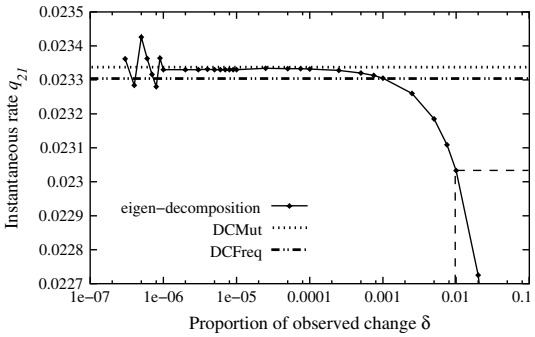


FIG. 2.—Element q_{21} (Arg → Asn) of the IRM computed by the eigen-decomposition method of Kishino, Miyata, and Hasegawa (1990) applied to the PAM1 matrix, and by the DCMut and the DCFreq methods. For the eigen-decomposition method, the value of q_{21} depends on δ; note the numerical instability we encountered for $\delta < 10^{-6}$. Also indicated is the value of q_{21} derived using $\delta = 0.01$; clearly, this does not attain the convergent behavior observed for $10^{-6} < \delta < 10^{-4}$. Note that in general the values of q_{ij} from the converged eigen-decomposition and DCMut methods do not agree as closely as observed in this case. A color version of this graph is available in the Supplementary Material online.

suffer from two convergence problems. On the one-hand, if δ is too large, elements q_{ij} may not yet be converged. On the other hand, choosing δ too small may cause the numerical eigen-analysis that finds the ρ_i and u_i to become unstable. Thus it is necessary to check convergence for each of the $19 \times 20 = 380$ values of $q_{ij,i\neq j}$. We find that many of the q_{ij} computed by the method of Kishino, Miyata, and Hasegawa (1990) using $\delta = 0.01$ do not attain converged values. Therefore, the IRM derived this way does not give a fully accurate representation of the amino acid replacement data collected by DSO78.

Direct Derivation of the Rate Matrix Using Mutabilities or Frequencies

It would be preferable to compute the elements of the IRM Q directly, and so avoid both relatively complex eigen-decomposition and lengthy convergence analysis to check that a suitable value (or values) of δ were used. The method of equation (2), as used by Goldman, Thorne, and Jones (1996, 1998), would be suitable, but this requires knowledge of the n_{ii} and these were not published by Dayhoff and colleagues. We now present two direct ways to estimate Q using only the information given by Dayhoff and colleagues, i.e., the observed changes $n_{ij,i\neq j}$, the mutabilities m_i and the frequencies f_i.

We call the first method *Direct Computation with Mutabilities* (DCMut), because it uses only the observed changes and the mutabilities. Rearranging equations (2) and (3):

$$q_{ij} = \frac{m_i n_{ij}}{\sum_{k \neq i} n_{ik}} \tag{11}$$

for $i \neq j$ (as usual, $q_{ii} = -\sum_j q_{ij}$).

The second method, *Direct Computation with Frequencies* (DCFreq), relies only on the observed changes and the frequencies and is defined by rearranging Equations (2) and (4):

$$q_{ij} = \frac{n_{ij}}{f_i \sum_k \sum_l n_{kl}}. \tag{12}$$

Since $\sum_k \sum_l n_{kl}$, the total number of amino acid sites in the sequence data, is a constant, it may be ignored because of the subsequent normalization of Q to have mean rate 1 (see above). Therefore, we may simply write:

$$q_{ij} = \frac{n_{ij}}{f_i}. \tag{13}$$

Note that both the DCMut and DCFreq derivations of IRMs require neither the consideration of any limit $\delta \to 0$ nor matrix eigen-analysis. They do, however, still require that the sequence pairs from which the n_{ij} are derived should be closely related (e.g., satisfying the 85% rule). Although there are many ways to calculate $P(t) = \exp(tQ)$ (Moler and Van Loan 2003), the most popular method in molecular phylogenetics uses eigen-decomposition (Liò and Goldman 1998) and thus even with the DCMut and DCFreq approaches to deriving Q we do not avoid eigen-analysis in its use.

Had Dayhoff and colleagues calculated mutabilities and frequencies according to equations (3) and (4), the IRMs given by equations (2), (11), and (13) would be identical. Likewise, if δ approaches zero then the Kishino, Miyata, and Hasegawa (1990) method applied to P_D (eq. 10) would also converge to this rate matrix. But because Dayhoff and colleagues estimated the mutabilities and frequencies incorporating the weighting factors described above (eq. 5 and eq. 6), we expect slightly different IRMs. Figure 2 shows the element q_{21} (Arg → Asn) calculated according to equations (10), (11), and (13). Although very close, the values are not identical. To facilitate agreement among scientists wishing to use supposedly identical models, we would like to be able to suggest one standard implementation of the model of DSO78 for molecular phylogenetics. In the *Results and Discussion* we perform a comparison of the different IRMs' performance in practice and propose a new standard.

Results and Discussion

We have identified different versions of the Dayhoff model described in the literature and used in current phylogenetic software packages, and implemented all of these variants in the *codeml* program of the PAML package (Yang 1997).[2] We distinguish between six different implementations, KMH, Paml, Proml, Molphy, DCMut, and DCFreq, and have assessed their impact on phylogenetic analysis by maximum likelihood.

KMH refers to the IRM calculated according to the eigen-decomposition method described by Kishino, Miyata, and Hasegawa (1990) applied to PAM1 of DSO78. No widely used software packages currently use exactly this implementation. We find that a number of off-diagonal

[2] Files prepared following the pattern of the distributed files (e.g.) dayhoff.dat and jones.dat are available from http://www.ebi.ac.uk/goldman/dayhoff.

elements of Q are < 0, meaning that this is not strictly a valid IRM (see the earlier section *From Probability to Rate Matrix via Eigen-Decomposition*). However, this implementation ran without problems in *codeml*, and so we made no further alterations.

The IRM for the Dayhoff model distributed with the PAML package, version 3.13d, (Yang 1997) is calculated using the Kishino, Miyata, and Hasegawa (1990) eigen-decomposition method applied to PAM1 (as for KMH above). The rate matrix $Q = (q_{ij})$ is then uniquely decomposed into $q_{ij}=f_j \times s_{ij}$ (see, e.g., Whelan and Goldman 2001), where the f_j are the equilibrium frequencies derived from this Q (and which thus differ minutely from those published by DSO78). The exchangeabilities s_{ij} are multiplied by approximately 100, rounded, and stored as integers. All except one of the $s_{ij} < 0$ become equal to 0 after this rounding. Following Yang (1997), we replace the rounded value of -1 for $s_{\mathrm{Glu} \rightarrow \mathrm{Arg}}$ with $+1$. We refer to this version of the Dayhoff model as the Paml implemention; it is also used in the *MrBayes* v. 3.0 (Ronquist and Huelsenbeck 2003) and *Phyml* v. 2.4 (Guindon and Gascuel 2003) programs.

The *proml* program distributed with version 3.6 of Felsenstein's PHYLIP package (Felsenstein 2002) includes an implementation of the Dayhoff model based on an IRM Q_{F} defined by

$$Q_{\mathrm{F}} = U \cdot \mathrm{diag}(\rho_{\mathrm{i}} - 1) \cdot \mathrm{U}^{-1}, \qquad (14)$$

where the ρ_i and u_i are the eigenvalues and eigenvectors, respectively, of a probability matrix P_{F} (Joe Felsenstein, pers. comm.). This means that $Q_{\mathrm{F}} = P_{\mathrm{F}} - I$, where I is the identity matrix. Felsenstein (2002) uses a matrix P_{F} based on the published DSO78 PAM1 matrix (not the original counts n_{ij}), modified to ensure reversibility (Felsenstein 1996; reversibility is violated by the originally published PAM1 matrix because of rounding errors). This implementation is thus an approximation to the DCMut method since $Q_{\mathrm{F}} = P_{\mathrm{F}} - I \approx P_{\mathrm{D}} - I$, which is proportional to the matrix defined by equation (11).

Early versions of the *protml* program of the MOLPHY package (Adachi and Hasegawa 1992) used the Kishino, Miyata, and Hasegawa (1990) eigen-decomposition of PAM1; later versions (e.g., v. 2.3b3) use the DCFreq approach, suggested to the authors of MOLPHY by Korbinian Strimmer and Arndt von Haeseler (Korbinian Strimmer, pers. comm.). However, the counts n_{ij} used were changed from those in DSO78: where zeros occurred, they were substituted by scaled values taken from Jones, Taylor, and Thornton (1992). This makes the later MOLPHY implementation, also used in the TREE-PUZZLE v. 5.2 package (Schmidt et al. 2002) and in the sequence simulation program *PSeq-Gen* v. 1.1 (Grassly, Adachi, and Rambaut 1997), a hybrid between the Dayhoff and JTT models.

The DCMut method has never been used before. We implemented the Dayhoff model according to this method, using data from DSO78. We have also calculated a DCFreq implementation, again using exactly the data from DSO78.

To get an idea of the impact of the different versions of the Dayhoff model on phylogenetic analysis, we performed a small test. We calculated the maximum likelihoods (Felsenstein 2003) for 200 protein families and their associated tree topologies taken from release 7.6 of the PANDIT database (Whelan, de Bakker, and Goldman 2003) using the implementations described above. Equilibrium frequencies were either taken from the appropriate implementation of the Dayhoff model (analyses labeled KMH, Paml, Proml, Molphy, DCMut, DCFreq), or were estimated separately for each protein family and incorporated using the "+F" method of Cao et al. (1994; see also Thorne and Goldman 2003; analyses KMH+F, Paml+F, Proml+F, Molphy+F, DCMut+F, DCFreq+F). (Note that it is not currently possible to use the Proml+F model in the *proml* program of Felsenstein's PHYLIP package. Note also the importance of re-normalizing Q so that the mean rate of change at equilibrium equals 1 when using the +F method, particularly if different software is used for simulating sequence data and then analyzing that simulated data.) Table 1 summarizes the results.

Table 1
Relative Success of 12 Implementations of the Dayhoff Matrix over a Test Set of 200 Protein Domain Families

Implementation	Median Rank	Interquartile Range of Ranks
Molphy+F	1	[1, 2]
DCMut+F	3	[3, 4]
DCFreq+F	3	[2, 5]
Paml+F	4	[4, 5]
Proml+F	6	[4, 6]
KMH+F	6	[5, 6]
Molphy	7	[7, 7]
DCMut	8	[8, 9]
DCFreq	8	[8, 10]
Paml	10	[9, 10]
KMH	11	[10, 11]
Proml	12	[8, 12]

NOTE.—The 200 amino acid sequence alignments used and log-likelihoods attained under all 12 models are listed in the Supplementary Material online.

For each of the 200 protein families, the 12 model versions were ranked according to their maximum likelihood values. This led to each model version being assigned 200 ranks—potentially from 1 to 12—according to its relative performance for the 200 protein families, and we report the medians and interquartile ranges of these ranks. First, as would be expected, we note that every implementation performs better in its "+F" version. Second, irrespective of whether we consider +F versions or not, we find the six basic implementations are ordered (best to worst): Molphy, DCMut, DCFreq, Paml, KMH $\approx$ Proml (the KMH and Proml models fared approximately equally poorly). Differences in actual maximum likelihoods among the six +F implementations were up to 102.82 log-likelihood units (median 2.28; inter-quartile range 0.77–6.36) over the 200 alignments analyzed; these differences may be of the order of the differences between competing topologies. For the six implementations without the +F modification, the corresponding differences were up to 87.67 log-likelihood units (median 2.28; inter-quartile range 0.79–6.82). Full results of this experiment are available in the Supplementary Material online.

By modern standards not much data was available to Dayhoff and colleagues, and their published values of n_{ij}

include a number of zeros. It seems likely that these give significant underestimates of the frequencies of replacement between certain amino acids, and we attribute the Molphy implementation's success to its hybrid nature, incorporating information from the counts published by Jones, Taylor, and Thornton (1992) to redress this underestimation. Among versions based solely on the information published by DSO78, the DCMut and DCFreq versions have advantages in computational ease and seem to give a small advantage in terms of maximum likelihood scores over the test data sets studied. The versions based on eigen-decomposition of PAM1 (i.e., with $\delta = 0.01$, which does not guarantee convergence) give the worst performance and seem to have no advantages. It is interesting to note that the Paml implementation fares better than KMH; evidently the variations introduced by the rounding procedure in the Paml version, in combination with the usage of a non-converged eigen-decomposition method and the sparse data published by DSO78, give a model that is no worse, and may even be better, than the version without rounding.

Conclusions

Methodological advances, increased database sizes, and faster computers now permit the inference of potentially superior Markov process models for protein sequence evolution. Even so, it is still valuable to consider simpler methods which are appropriate for making inferences from pairwise comparisons of closely related proteins. These methods may become more important as sequencing projects complete the proteomes of closely related species such as human and chimpanzee (International Human Genome Sequencing Consortium 2001; International Chimpanzee Chromosome 22 Consortium 2004), mouse and rat (Mouse Genome Sequencing Consortium 2002; Rat Genome Sequencing Project Consortium 2004), and many others. Indeed, even within-species studies will become possible, for example using SNPs (International SNP Map Working Group 2001).

We have summarized different approaches (including two new ones) to calculating an instantaneous rate matrix from the (incomplete) information given by Dayhoff and colleagues. In practice, the differences are small but may be non-trivial. All the implementations studied are valid models of protein sequence evolution, and they may be applied to any protein sequence alignments. Nevertheless, while this and similar models remain of interest in molecular phylogenetics and other fields and for the sake of consistency, particularly among investigators developing models and software implementations, we suggest that it is of value to identify a "standard" implementation for the model of Dayhoff, Schwarz, and Orcutt (1978).

Although the Molphy implementation (Adachi and Hasegawa 1992) performs well, it uses a hybrid data set based on both Dayhoff, Schwarz, and Orcutt (1978) and Jones, Taylor, and Thornton (1992). The Proml implementation also uses the Dayhoff, Schwarz, and Orcutt (1978) data in a modified form. The Proml, Paml, and KMH implementations perform least well in our maximum likelihood implementation experiment, and the generation of their instantaneous rate matrices is based on relatively complex eigen-decomposition and, in the cases of Paml and KMH, exhibits convergence problems. Therefore, looking for standardization based only on the data published by Dayhoff, Schwarz, and Orcutt (1978), we propose the adoption of the DCMut matrix, which is straightforward to calculate and performed well in our test. Adopting DCMut also means that no decision has to be made regarding the fact that, because of rounding errors, the equilibrium frequencies published by Dayhoff, Schwarz, and Orcutt (1978) do not sum to 1. We do not suggest that DCMut is a better model than any other implementation described here, but only that it is a reasonable choice when a single model is required for comparing results across different analyses, computer programs, or data sets.

Jones, Taylor, and Thornton (1992) used much the same methodology as Dayhoff and colleagues, but based on a larger sequence database. Again, only a probability matrix, mutabilities, frequencies and incomplete counts are provided in their original article. The JTT probability matrix has also been used to compute an IRM for use in phylogenetics and has been implemented (again, in a variety of ways) in the MOLPHY, *MrBayes*, PAML, PHYLIP, *Phyml, PSeq-Gen*, and TREE-PUZZLE software. We again suggest that an implementation based on the DCMut method be adopted as standard.[3]

We hope to persuade software developers to agree to include our DCMut implementations of the Dayhoff and JTT models into their programs, and we are happy to discuss providing the necessary data in whatever forms required.[4]

Acknowledgments

Thanks go to Jeff Thorne, for assistance with understanding Dayhoff and colleagues' calculation of relative mutabilities, and to Joe Felsenstein, Korbinian Strimmer, and Elisabeth Tillier for helpful discussions of their implementations of the Dayhoff and JTT models. C.K. is supported by a Wellcome Trust Prize Studentship and is a member of Wolfson College, University of Cambridge. N.G. is supported by a Wellcome Trust Fellowship in Basic Biomedical Research.

Literature Cited

Adachi, J., and M. Hasegawa. 1992. MOLPHY version 2.3: Programs for Molecular Phylogenetics Based on Maximum Likelihood. Computer Science Monographs 28, Institute of Statistical Mathematics, Tokyo. http://www.ism.ac.jp/software/ismlib/softother.e.html#molphy

———. 1996. Model of amino acid substitution in proteins encoded by mitochondrial DNA. J. Mol. Evol. **42**:459–468.

[3] A file containing this matrix, suitable for use in the *codeml* program of the PAML package, is available from http://www.ebi.ac.uk/goldman/dayhoff.

[4] The authors of *MrBayes* (Ronquist and Huelsenbeck 2003), PAML (Yang 1997), PAUP* (Swofford 2002), *Phyml* (Guindon and Gascuel 2003), *Seq-Gen* (Rambaut and Grassly 1997; the successor to *PSeqGen*) and TREE-PUZZLE (Schmidt et al. 2002) have agreed to incorporate the DCMut versions of the Dayhoff and JTT IRMs in future releases of their software.

Adachi, J., P. J. Waddell, W. Martin, and M. Hasegawa. 2000. Plastid genome phylogeny and a model of amino acid substitution for proteins encoded by chloroplast DNA. J. Mol. Evol. **50**:348–358.

Cao, Y., J. Adachi, A. Janke, S. Pääbo, and M. Hasegawa. 1994. Phylogenetic relationships among eutherian orders estimated from inferred sequences of mitochondrial proteins: instability of a tree based on a single gene. J. Mol. Evol. **39**: 519–527.

Dayhoff, M. O., R. V. Eck, and C. M. Park. 1972. A model of evolutionary change in proteins. Pp. 89–99 *in* M. O. Dayhoff, ed., Atlas of Protein Sequence and Structure Vol. 5. National Biomedical Research Foundation, Washington, D.C.

Dayhoff, M. O., R. M. Schwartz, and B. C. Orcutt. 1978. A model of evolutionary change in proteins. Pp. 345–352 *in* M. O. Dayhoff, ed., Atlas of Protein Sequence and Structure Vol. 5, suppl. 3. National Biomedical Research Foundation, Washington, D.C.

Devauchelle, C., A. Grossmann, A. Hénaut, M. Holschneider, M. Monnerot, J. L. Riesler, and B. Torrésani. 2001. Rate matrices for analyzing large families of protein sequences. J. Comp. Biol. **8**:381–399.

Dimmic, M. W., J. S. Rest, D. P. Mindell, and R. A. Goldstein. 2002. rtREV: an amino acid substitution matrix for inference of retrovirus and reverse transcriptase phylogeny. J. Mol. Evol. **55**:65–73.

Felsenstein, J. 1996. Inferring phylogenies from protein sequences by parsimony, distance, and likelihood methods. Methods Enzymol. **266**:418–427.

———. 2002. PHYLIP (Phylogeny Inference Package) Version 3.6a. Department of Genome Sciences, University of Washington, Seattle, Wash. http://evolution.genetics.washington.edu/phylip.html

———. 2003. Inferring phylogenies. Sinauer Associates, Sunderland, Mass.

Goldman, N., J. L. Thorne, and D. T. Jones. 1996. Using evolutionary trees in protein secondary structure prediction and other comparative sequence analysis. J. Mol. Biol. **263**:196–208.

———. 1998. Assessing the impact of secondary structure and solvent accessibility on protein evolution. Genetics **149**: 445–458.

Grassly, N. C., J. Adachi, and A. Rambaut. 1997. PSeq-Gen: an application for the Monte Carlo simulation of protein sequence evolution along phylogenetic trees. CABIOS **13**:559–560. http://evolve.zoo.ox.ac.uk/software.html?id=pseqgen

Guindon, S., and O. Gascuel. 2003. A simple, fast, and accurate algorithm to estimate large phylogenies by maximum likelihood. Syst. Biol. **52**:696–704. http://atgc.lirmm.fr/phyml

International Chimpanzee Chromosome 22 Consortium. 2004. DNA sequence and comparative analysis of chimpanzee chromosome 22. Nature **429**:382–388.

International Human Genome Sequencing Consortium. 2001. Initial sequencing and analysis of the human genome. Nature **409**:860–921.

International SNP Map Working Group. 2001. A map of human genome sequence variation containing 1.42 million single nucleotide polymorphisms. Nature **409**:928–933.

Jones, D. T., W. R. Taylor, and J. M. Thornton. 1992. The rapid generation of mutation data matrices from protein sequences. CABIOS **8**:275–282.

Kishino, H., T. Miyata, and M. Hasegawa. 1990. Maximum likelihood inference of protein phylogeny and the origin of chloroplasts. J. Mol. Evol. **31**:151–160.

Liò, P., and N. Goldman. 1998. Models of molecular evolution and phylogeny. Genome Res. **8**:1233–1244.

Moler, C., and C. Van Loan. 2003. Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. SIAM Rev. **45**:3–49.

Mouse Genome Sequencing Consortium. 2002. Initial sequencing and comparative analysis of the mouse genome. Nature **420**:520–562.

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

Rambaut, A., and N. C. Grassly. 1997. Seq-Gen: an application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. CABIOS **13**:235–238. http://evolve.zoo.ox.ac.uk/software.html?id=seqgen

Rat Genome Sequencing Project Consortium. 2004. Genome sequence of the Brown Norway rat yields insights into mammalian evolution. Nature **428**:493–521.

Ronquist, F., and J. P. Huelsenbeck. 2003. MrBayes 3: Bayesian phylogenetic inference under mixed models. Bioinformatics **19**:1572–1574. http://morphbank.ebc.uu.se/mrbayes3

Schmidt, H. A., K. Strimmer, M. Vingron, and A. von Haeseler. 2002. Tree-Puzzle: maximum likelihood phylogenetic analysis using quartets and parallel computing. Bioinformatics **18**:502–504. http://www.tree-puzzle.de

Swofford, D. L. 2002. PAUP*. *Phylogenetic analysis using parsimony (and other methods) version 4. Sinauer Associates, Sunderland, Mass. http://paup.csit.fsu.edu

Thorne, J. L., and N. Goldman. 2003. Probabilistic models for the study of protein evolution. Pp. 209–226 *in* D. J. Balding, M. Bishop, and C. Cannings, eds. Handbook of Statistical Genetics, 2nd Ed. Wiley, Chichester.

Veerassamy, S., A. Smith, and E. R. M. Tillier. 2003. A transition probability model for amino acid substitutions from Blocks. J. Comp. Biol. **10**:997–1010.

Whelan, S., P. I. W. de Bakker, and N. Goldman. 2003. Pandit: a database of protein and associated nucleotide domains with inferred trees. Bioinformatics **19**:1556–1563. http://www.ebi.ac.uk/goldman-srv/pandit

Whelan, S., and N. Goldman. 2001. A general empirical model of protein evolution derived from multiple protein families using a maximum-likelihood approach. Mol. Biol. Evol. **18**:691–699.

Yang, Z. 1997. PAML: a program package for phylogenetic analysis by maximum likelihood. CABIOS **13**:555–556. http://abacus.gene.ucl.ac.uk/software/paml.html

Yang, Z., R. Nielsen, and M. Hasegawa. 1998. Models of amino acid substitution and applications to mitochondrial protein evolution. Mol. Biol. Evol. **15**:1600–1611.

Arndt von Haeseler, Associate Editor

Accepted October 4, 2004

Appendix B

List of Log-likelihoods

This appendix includes the complete results of the ML analysis on 200 protein families for the M0 model and the seven different κ-models (see section 6.4). Each row corresponds to the analysis of the Pfam-A seed alignment of a family and its corresponding phylogeny, taken from release 17.0 of the Pandit database Log-likelihoods are reported for each of the 8 models specified. Phylogenies' branch lengths were re-estimated in each analysis.

PFAM id	M0 model	0κ-model	1κ(a)-model	1κ(b)-model	1κ(c)-model	1κ(d)-model	2κ(a)-model	2κ(b)-model
PF01201	-15654.97	-14851.71	-14844.44	-14842.87	-14841.65	-14840.72	-14840.61	-14840.60
PF01202	-48816.80	-45685.71	-45667.50	-45666.97	-45660.90	-45660.85	-45660.21	-45659.84
PF01203	-4373.91	-4210.53	-4210.49	-4209.69	-4209.61	-4209.48	-4209.44	-4209.03
PF01204	-37214.27	-35046.25	-35045.32	-35045.11	-35044.97	-35044.92	-35043.55	-35043.35
PF01205	-4465.13	-4209.68	-4209.68	-4209.66	-4209.45	-4209.37	-4209.32	-4208.25
PF01206	-1717.22	-1652.36	-1652.19	-1651.99	-1651.57	-1651.44	-1651.34	-1651.31
PF01207	-19095.40	-18363.28	-18360.37	-18360.23	-18354.37	-18352.72	-18352.50	-18350.77
PF01208	-16528.70	-15747.71	-15747.03	-15746.52	-15745.22	-15745.04	-15744.98	-15744.42
PF01209	-4052.36	-3853.20	-3853.20	-3853.14	-3852.67	-3851.65	-3850.51	-3849.59
PF01210	-24019.42	-22889.77	-22886.56	-22885.74	-22885.69	-22885.23	-22881.90	-22881.89
PF01212	-47225.13	-43891.82	-43862.93	-43862.17	-43847.50	-43847.46	-43816.73	-43812.64
PF01213	-9692.27	-9328.87	-9328.86	-9328.83	-9328.80	-9328.79	-9328.06	-9326.77
PF01214	-3799.77	-3568.21	-3568.14	-3568.11	-3566.88	-3566.24	-3563.23	-3562.61
PF01215	-983.62	-954.37	-954.17	-953.92	-953.72	-953.72	-953.48	-953.21
PF01216	-3120.69	-3020.64	-3019.96	-3019.85	-3019.73	-3019.72	-3019.07	-3019.06
PF01217	-3359.57	-3209.45	-3209.39	-3208.20	-3208.12	-3208.06	-3207.81	-3206.56
PF01218	-5737.63	-5397.38	-5397.37	-5397.36	-5397.35	-5397.21	-5395.47	-5393.79
PF01219	-2300.72	-2230.32	-2230.29	-2230.26	-2230.21	-2230.13	-2229.59	-2229.31
PF01220	-3076.42	-2879.23	-2876.48	-2875.91	-2875.19	-2875.08	-2873.62	-2873.60
PF01221	-1590.65	-1505.55	-1503.61	-1502.88	-1502.82	-1502.69	-1502.63	-1502.52
PF01222	-7795.75	-7517.59	-7517.47	-7517.19	-7517.09	-7516.81	-7516.80	-7511.30
PF01223	-40153.00	-38356.26	-38356.26	-38356.19	-38356.02	-38355.63	-38355.51	-38353.55
PF01226	-5659.72	-5499.90	-5499.78	-5499.78	-5499.77	-5499.58	-5499.56	-5499.53
PF01227	-4619.41	-4370.58	-4370.57	-4370.37	-4369.87	-4368.96	-4366.69	-4364.24
PF01228	-2978.93	-2863.92	-2863.89	-2863.64	-2863.61	-2863.42	-2863.42	-2863.42

PFAM id	M0 model	0κ-model	1κ(a)-model	1κ(b)-model	1κ(c)-model	1κ(d)-model	2κ(a)-model	2κ(b)-model
PF01229	-6718.82	-6610.25	-6604.24	-6601.98	-6596.51	-6596.29	-6595.96	-6595.48
PF01230	-7376.82	-7031.01	-7030.87	-7030.82	-7028.57	-7025.95	-7025.84	-7021.24
PF01231	-5430.65	-5291.25	-5289.47	-5289.41	-5288.03	-5287.85	-5287.64	-5287.55
PF01233	-2400.04	-2285.63	-2285.54	-2285.35	-2285.24	-2285.24	-2285.23	-2285.13
PF01234	-5228.38	-5022.01	-5021.99	-5021.94	-5021.92	-5021.92	-5021.73	-5021.05
PF01235	-8664.61	-8352.70	-8352.57	-8352.53	-8352.53	-8352.50	-8352.43	-8352.43
PF01237	-9734.83	-9488.76	-9488.41	-9488.29	-9487.70	-9487.58	-9487.58	-9487.38
PF01238	-9740.61	-9312.21	-9312.11	-9312.04	-9311.82	-9311.66	-9308.48	-9305.91
PF01239	-12891.35	-12357.18	-12356.55	-12356.36	-12356.20	-12354.07	-12350.51	-12339.54
PF01241	-2324.87	-2253.35	-2253.29	-2253.23	-2253.05	-2253.00	-2253.00	-2252.75
PF01242	-966.45	-937.53	-936.98	-935.93	-935.79	-935.78	-934.38	-934.37
PF01243	-40524.58	-38122.75	-38097.03	-38096.87	-38095.98	-38095.65	-38093.76	-38093.53
PF01244	-48646.66	-45815.90	-45815.86	-45815.69	-45815.41	-45814.24	-45811.85	-45810.38
PF01245	-2902.63	-2765.46	-2764.43	-2764.09	-2763.77	-2763.74	-2761.87	-2761.51
PF01246	-1272.53	-1195.94	-1195.94	-1195.94	-1195.94	-1195.93	-1195.71	-1195.43
PF01247	-897.14	-865.85	-865.61	-864.51	-863.74	-863.73	-863.70	-862.77
PF01248	-13676.99	-12895.38	-12894.10	-12894.04	-12893.58	-12893.54	-12893.53	-12893.53
PF01249	-953.83	-883.71	-881.49	-880.78	-880.60	-880.27	-878.74	-878.68
PF01250	-3480.91	-3358.28	-3358.26	-3357.60	-3357.39	-3356.97	-3356.93	-3356.41
PF01251	-2608.58	-2459.09	-2459.06	-2459.05	-2459.02	-2459.01	-2458.91	-2458.42
PF01253	-5485.69	-5239.05	-5236.30	-5235.73	-5235.51	-5235.36	-5235.35	-5235.21
PF01254	-1151.62	-1147.06	-1145.24	-1135.67	-1135.27	-1135.02	-1134.80	-1133.69
PF01255	-5966.97	-5687.17	-5687.15	-5686.24	-5685.47	-5685.42	-5684.69	-5682.22
PF01256	-8213.21	-7967.68	-7964.98	-7964.90	-7963.93	-7963.92	-7957.58	-7955.63
PF01257	-2542.39	-2443.90	-2443.90	-2443.89	-2443.81	-2443.78	-2443.56	-2443.47

PFAM id	M0 model	0κ-model	1κ(a)-model	1κ(b)-model	1κ(c)-model	1κ(d)-model	2κ(a)-model	2κ(b)-model
PF01258	-1574.85	-1531.53	-1531.13	-1530.43	-1530.37	-1530.20	-1530.06	-1530.03
PF01259	-7272.30	-6983.94	-6982.71	-6982.59	-6982.18	-6982.18	-6979.99	-6979.59
PF01263	-70492.61	-66927.24	-66920.75	-66919.29	-66911.94	-66911.13	-66904.24	-66900.23
PF01264	-13848.55	-12980.26	-12979.19	-12978.98	-12975.68	-12974.21	-12971.28	-12968.07
PF01265	-4722.65	-4539.81	-4539.74	-4539.67	-4539.56	-4538.60	-4538.55	-4534.65
PF01267	-4534.37	-4357.51	-4357.37	-4357.35	-4357.33	-4357.28	-4356.87	-4356.72
PF01268	-12621.13	-12021.81	-12020.68	-12019.69	-12019.24	-12019.19	-12018.60	-12018.34
PF01270	-9719.87	-9311.00	-9310.33	-9310.22	-9309.90	-9309.90	-9308.94	-9308.68
PF01271	-14326.44	-14282.27	-14259.38	-14250.91	-14210.86	-14203.68	-14192.61	-14176.68
PF01272	-2006.13	-1925.50	-1925.32	-1925.10	-1922.45	-1922.36	-1920.17	-1920.16
PF01273	-20188.31	-19708.79	-19641.68	-19624.92	-19595.37	-19595.28	-19573.95	-19566.30
PF01274	-25679.26	-24177.16	-24177.16	-24177.14	-24176.53	-24175.91	-24175.33	-24173.13
PF01275	-4858.39	-4699.55	-4697.27	-4696.32	-4695.02	-4695.01	-4686.78	-4681.99
PF01276	-12869.78	-12259.51	-12259.50	-12259.41	-12258.67	-12258.51	-12258.28	-12256.96
PF01277	-5138.32	-4967.01	-4961.74	-4958.21	-4953.01	-4952.92	-4951.92	-4949.64
PF01278	-6597.98	-6476.39	-6468.36	-6465.24	-6450.36	-6447.99	-6432.38	-6414.72
PF01279	-1782.92	-1769.84	-1769.57	-1768.28	-1765.68	-1765.30	-1762.17	-1756.17
PF01280	-2971.24	-2852.14	-2852.05	-2852.04	-2851.29	-2851.25	-2850.92	-2850.67
PF01281	-1658.86	-1563.09	-1562.98	-1562.76	-1562.73	-1562.71	-1561.96	-1561.81
PF01282	-1590.76	-1539.76	-1539.07	-1538.51	-1538.49	-1538.42	-1538.36	-1538.33
PF01283	-1805.14	-1723.06	-1723.04	-1722.88	-1722.83	-1722.40	-1721.76	-1721.75
PF01284	-21371.30	-20811.08	-20795.95	-20795.77	-20792.45	-20791.90	-20788.05	-20788.00
PF01285	-7953.97	-7808.99	-7808.99	-7808.99	-7808.96	-7808.81	-7808.58	-7806.76
PF01286	-584.64	-558.53	-558.53	-558.24	-557.36	-557.13	-557.02	-556.32
PF01287	-1259.58	-1190.29	-1190.26	-1190.18	-1190.06	-1189.85	-1188.94	-1188.40

PFAM id	M0 model	0κ-model	1κ(a)-model	1κ(b)-model	1κ(c)-model	1κ(d)-model	2κ(a)-model	2κ(b)-model
PF01288	-7041.41	-6688.19	-6688.18	-6687.90	-6687.85	-6687.84	-6687.81	-6686.98
PF01289	-7697.10	-7333.70	-7332.78	-7332.15	-7331.65	-7331.64	-7323.40	-7316.58
PF01290	-292.93	-277.37	-277.36	-277.29	-277.29	-277.21	-277.20	-277.15
PF01291	-3004.44	-2992.90	-2975.56	-2972.71	-2955.57	-2955.25	-2954.22	-2954.21
PF01292	-17306.79	-16609.91	-16609.86	-16609.85	-16604.03	-16604.02	-16596.18	-16590.67
PF01293	-14157.29	-13228.56	-13228.54	-13228.49	-13227.17	-13226.04	-13224.10	-13219.37
PF01294	-3173.53	-3033.64	-3029.46	-3028.99	-3024.53	-3024.50	-3022.77	-3021.91
PF01295	-12733.15	-12177.61	-12175.11	-12175.09	-12174.10	-12174.02	-12171.84	-12171.78
PF01296	-133.19	-131.52	-131.50	-131.33	-131.32	-131.30	-131.22	-131.06
PF01297	-21088.95	-20122.03	-20121.37	-20120.03	-20114.48	-20112.93	-20110.53	-20107.68
PF01299	-9858.28	-9695.88	-9671.14	-9663.99	-9655.69	-9655.43	-9638.65	-9634.09
PF01300	-12004.35	-11507.87	-11507.39	-11506.56	-11506.47	-11506.06	-11505.55	-11505.54
PF01301	-32093.97	-30303.66	-30303.62	-30303.56	-30303.28	-30303.17	-30302.76	-30302.10
PF01302	-9060.36	-8459.42	-8453.97	-8453.77	-8453.33	-8453.11	-8451.23	-8451.20
PF01303	-3912.34	-3754.56	-3754.48	-3754.14	-3746.76	-3740.82	-3737.91	-3714.75
PF01304	-549.59	-549.58	-549.58	-547.86	-547.41	-545.33	-544.84	-542.60
PF01305	-3069.87	-2953.55	-2945.29	-2944.51	-2942.78	-2941.97	-2939.30	-2939.30
PF01306	-5314.76	-5092.40	-5091.89	-5091.59	-5091.58	-5091.56	-5090.98	-5090.98
PF01307	-9223.69	-8839.94	-8839.94	-8839.65	-8838.91	-8838.66	-8837.83	-8832.11
PF01308	-3365.25	-3253.88	-3252.93	-3252.07	-3251.63	-3251.56	-3250.71	-3250.39
PF01309	-1683.56	-1666.75	-1648.36	-1625.38	-1622.22	-1621.76	-1612.64	-1607.46
PF01310	-5442.17	-5279.30	-5278.93	-5278.61	-5278.59	-5277.96	-5277.78	-5277.70
PF01311	-13677.47	-13255.47	-13255.46	-13255.41	-13255.14	-13255.02	-13254.91	-13254.47
PF01312	-18491.69	-17788.60	-17787.64	-17786.79	-17786.71	-17786.61	-17785.65	-17785.65
PF01313	-3459.13	-3292.09	-3291.65	-3291.18	-3290.67	-3290.43	-3290.41	-3290.35

PFAM id	M0 model	0κ-model	1κ(a)-model	1κ(b)-model	1κ(c)-model	1κ(d)-model	2κ(a)-model	2κ(b)-model
PF01314	-12200.29	-11746.14	-11746.14	-11745.57	-11745.15	-11745.08	-11742.66	-11738.70
PF01315	-10659.25	-10046.18	-10045.17	-10044.81	-10040.98	-10038.67	-10032.55	-10022.02
PF01316	-2418.71	-2353.57	-2353.54	-2353.53	-2353.53	-2353.43	-2353.20	-2353.19
PF01318	-1823.31	-1769.65	-1769.65	-1769.57	-1769.49	-1769.48	-1769.42	-1769.17
PF01320	-1419.43	-1351.33	-1351.28	-1351.21	-1351.12	-1351.12	-1350.93	-1349.88
PF01321	-1084.07	-1039.08	-1038.39	-1038.11	-1038.10	-1038.08	-1037.91	-1037.84
PF01322	-1865.37	-1812.96	-1812.96	-1812.95	-1812.90	-1812.89	-1812.80	-1812.65
PF01323	-17264.74	-16512.21	-16511.60	-16510.92	-16509.69	-16509.15	-16508.33	-16507.63
PF01324	-609.39	-608.90	-608.71	-608.41	-608.30	-608.28	-608.27	-607.91
PF01325	-1996.81	-1910.02	-1909.51	-1909.18	-1909.17	-1909.08	-1908.70	-1908.70
PF01326	-26405.86	-25164.11	-25163.96	-25163.85	-25163.85	-25163.79	-25162.58	-25162.26
PF01328	-2309.38	-2282.40	-2282.11	-2282.10	-2281.20	-2281.14	-2280.81	-2280.48
PF01329	-1804.22	-1751.17	-1749.01	-1748.57	-1748.41	-1748.21	-1747.60	-1747.59
PF01330	-6177.42	-5840.07	-5838.80	-5838.03	-5836.58	-5836.52	-5835.69	-5835.25
PF01331	-7582.10	-7365.11	-7365.10	-7365.07	-7365.04	-7364.85	-7364.79	-7363.69
PF01333	-3501.10	-3376.36	-3376.02	-3375.34	-3375.14	-3375.12	-3374.62	-3373.93
PF01335	-10259.36	-9889.75	-9884.80	-9884.50	-9883.18	-9882.96	-9876.80	-9874.51
PF01337	-603.30	-589.37	-588.53	-588.46	-588.14	-588.14	-587.65	-587.57
PF01338	-2506.44	-2408.91	-2408.22	-2407.90	-2406.58	-2406.24	-2405.53	-2404.99
PF01339	-4765.14	-4521.74	-4521.69	-4521.65	-4519.78	-4519.00	-4518.52	-4514.13
PF01340	-635.44	-632.57	-630.54	-630.49	-628.14	-627.66	-627.51	-627.36
PF01341	-13041.42	-11986.49	-11984.35	-11983.02	-11975.74	-11973.44	-11971.53	-11965.80
PF01342	-2725.30	-2629.89	-2629.88	-2629.84	-2627.90	-2626.47	-2623.79	-2622.54
PF01343	-10344.12	-9846.36	-9846.34	-9846.31	-9845.43	-9845.33	-9844.67	-9844.26
PF01345	-8922.15	-8605.20	-8598.69	-8597.27	-8595.55	-8595.42	-8592.24	-8592.08

PFAM id	M0 model	0κ-model	1κ(a)-model	1κ(b)-model	1κ(c)-model	1κ(d)-model	2κ(a)-model	2κ(b)-model
PF01346	-3583.51	-3448.49	-3448.39	-3448.37	-3445.55	-3443.08	-3442.63	-3437.76
PF01347	-83192.70	-80132.32	-80116.53	-80115.75	-80082.15	-80070.43	-80044.70	-80018.83
PF01349	-4885.34	-4732.49	-4732.48	-4732.42	-4732.39	-4732.34	-4732.17	-4731.71
PF01350	-2648.51	-2546.18	-2546.17	-2546.17	-2546.16	-2546.16	-2546.15	-2546.08
PF01351	-9244.36	-8826.78	-8822.72	-8821.71	-8818.29	-8818.13	-8808.10	-8804.79
PF01352	-6430.18	-6415.37	-6385.26	-6376.31	-6319.92	-6313.59	-6293.01	-6274.24
PF01353	-11415.46	-11113.79	-11111.45	-11107.56	-11091.26	-11082.77	-11078.69	-11053.54
PF01354	-3369.19	-3183.53	-3182.90	-3182.57	-3181.34	-3181.26	-3181.18	-3180.88
PF01357	-10331.25	-9638.85	-9638.85	-9638.22	-9637.56	-9637.50	-9636.58	-9626.11
PF01358	-5305.82	-5146.87	-5146.14	-5145.31	-5145.27	-5145.23	-5145.11	-5145.02
PF01359	-10364.56	-9979.87	-9978.56	-9977.79	-9968.97	-9964.53	-9961.71	-9951.11
PF01361	-5749.19	-5471.32	-5470.52	-5470.50	-5470.27	-5470.24	-5469.72	-5469.63
PF01363	-4521.94	-4361.86	-4361.02	-4360.87	-4360.42	-4360.42	-4359.00	-4358.47
PF01364	-2683.05	-2640.66	-2640.62	-2640.61	-2640.61	-2640.56	-2640.48	-2640.46
PF01365	-8915.61	-8582.72	-8580.77	-8579.97	-8578.75	-8578.75	-8575.83	-8575.05
PF01366	-30120.64	-28754.62	-28748.80	-28747.86	-28746.69	-28746.48	-28739.35	-28738.15
PF01367	-8500.73	-7961.90	-7957.37	-7956.20	-7949.17	-7948.97	-7947.19	-7943.74
PF01369	-6966.86	-6678.10	-6678.02	-6677.76	-6677.64	-6677.43	-6675.74	-6673.74
PF01370	-73936.57	-69791.69	-69770.67	-69770.29	-69743.21	-69741.29	-69741.00	-69735.16
PF01371	-4506.49	-4321.43	-4321.34	-4321.32	-4320.71	-4320.23	-4319.15	-4318.43
PF01373	-6039.13	-5784.27	-5784.26	-5784.12	-5783.92	-5783.89	-5783.14	-5781.96
PF01374	-3911.44	-3750.47	-3750.47	-3750.46	-3750.46	-3750.44	-3750.34	-3749.88
PF01375	-2396.07	-2338.26	-2336.30	-2335.88	-2334.17	-2334.12	-2333.57	-2333.37
PF01376	-629.76	-629.49	-629.07	-627.53	-626.25	-625.30	-621.11	-620.72
PF01378	-629.64	-595.62	-595.59	-595.51	-595.40	-595.28	-593.33	-591.71

PFAM id	M0 model	0κ-model	1κ(a)-model	1κ(b)-model	1κ(c)-model	1κ(d)-model	2κ(a)-model	2κ(b)-model
PF01379	-12035.24	-11471.47	-11470.80	-11469.57	-11468.76	-11468.76	-11467.06	-11465.64
PF01380	-18497.86	-17794.56	-17794.30	-17794.06	-17793.86	-17793.78	-17793.75	-17793.73
PF01381	-34142.44	-32522.44	-32512.27	-32508.85	-32507.42	-32507.15	-32505.92	-32505.68
PF01382	-1544.96	-1504.87	-1504.80	-1504.74	-1504.72	-1504.51	-1503.29	-1502.40
PF01383	-3438.61	-3325.62	-3325.22	-3325.03	-3324.93	-3323.31	-3321.49	-3314.54
PF01384	-30411.32	-29450.25	-29443.64	-29442.93	-29441.96	-29441.64	-29436.24	-29434.51
PF01385	-13672.34	-13099.22	-13099.22	-13098.88	-13098.61	-13098.60	-13097.37	-13094.76
PF01386	-1824.40	-1763.89	-1763.84	-1763.80	-1763.67	-1763.65	-1763.51	-1763.36
PF01387	-1114.74	-1070.72	-1070.62	-1070.49	-1070.24	-1069.97	-1069.96	-1069.95
PF01388	-3805.32	-3680.01	-3680.01	-3680.01	-3679.63	-3679.39	-3678.68	-3678.41
PF01389	-6308.02	-5921.51	-5917.78	-5917.26	-5912.32	-5911.82	-5911.62	-5910.90
PF01390	-6840.03	-6727.95	-6714.16	-6710.91	-6698.80	-6697.75	-6696.59	-6694.21
PF01391	-6485.76	-6288.09	-6288.04	-6287.88	-6287.41	-6287.17	-6286.92	-6284.69
PF01392	-11919.35	-11430.98	-11423.73	-11422.68	-11422.10	-11421.89	-11410.06	-11408.85
PF01393	-1480.17	-1422.49	-1420.42	-1420.31	-1420.13	-1418.79	-1418.02	-1417.97
PF01394	-7468.14	-7036.70	-7035.27	-7033.97	-7032.45	-7032.07	-7031.25	-7030.90
PF01395	-34039.49	-32805.84	-32801.17	-32800.64	-32800.41	-32800.05	-32778.95	-32772.18
PF01396	-3065.02	-2860.01	-2857.69	-2857.42	-2856.08	-2856.05	-2854.95	-2854.78
PF01397	-39090.44	-38561.43	-38526.04	-38509.25	-38391.67	-38348.01	-38305.43	-38187.79
PF01398	-12561.89	-11939.48	-11938.68	-11938.67	-11936.86	-11936.00	-11932.56	-11929.86
PF01399	-28603.11	-27308.86	-27308.86	-27308.84	-27308.83	-27308.79	-27308.74	-27308.34
PF01400	-13686.88	-12980.96	-12980.95	-12980.86	-12979.99	-12978.50	-12977.07	-12973.41
PF01401	-6303.85	-6208.91	-6174.21	-6167.58	-6141.93	-6140.63	-6137.95	-6135.45
PF01402	-6351.17	-6094.66	-6093.79	-6092.27	-6087.73	-6087.65	-6086.19	-6084.72
PF01403	-33866.69	-32617.17	-32612.68	-32608.86	-32604.08	-32603.98	-32591.87	-32582.37

PFAM id	M0 model	0κ-model	1κ(a)-model	1κ(b)-model	1κ(c)-model	1κ(d)-model	2κ(a)-model	2κ(b)-model
PF01404	-6285.25	-6001.93	-6001.68	-6001.68	-6000.40	-5999.49	-5997.28	-5993.56
PF01405	-221.75	-215.52	-213.94	-213.47	-213.45	-213.37	-213.14	-213.01
PF01407	-5122.52	-4931.09	-4931.05	-4930.91	-4930.77	-4930.50	-4929.74	-4926.88
PF01408	-18767.55	-17899.51	-17898.02	-17897.09	-17893.61	-17893.28	-17892.87	-17889.50
PF01410	-5787.23	-5477.43	-5477.42	-5477.32	-5477.27	-5476.77	-5476.40	-5473.24
PF01412	-8516.89	-8033.17	-8033.14	-8032.99	-8032.21	-8031.78	-8031.77	-8030.84
PF01413	-7129.76	-6700.72	-6698.60	-6698.50	-6697.85	-6697.83	-6687.26	-6683.33
PF01414	-2699.21	-2572.22	-2572.17	-2572.00	-2571.93	-2571.92	-2571.27	-2568.71
PF01415	-1661.63	-1661.03	-1661.02	-1658.76	-1657.47	-1656.15	-1654.35	-1650.33
PF01417	-7822.58	-7405.51	-7405.51	-7404.82	-7403.30	-7402.57	-7401.31	-7400.65
PF01418	-3062.70	-2952.32	-2952.06	-2952.00	-2951.78	-2951.78	-2951.70	-2951.67
PF01419	-16881.75	-16317.53	-16311.43	-16303.07	-16284.19	-16279.57	-16279.39	-16262.42
PF01421	-17449.90	-17014.97	-16996.35	-16982.02	-16945.30	-16939.54	-16917.15	-16884.35
PF01422	-1167.44	-1152.58	-1152.30	-1150.16	-1148.86	-1148.22	-1148.20	-1143.62
PF01423	-24284.33	-22920.00	-22920.00	-22920.00	-22919.98	-22919.92	-22919.72	-22919.12
PF01424	-6548.18	-6312.74	-6312.68	-6312.59	-6311.83	-6311.23	-6311.12	-6309.67
PF01426	-17821.15	-17210.15	-17210.15	-17210.09	-17210.08	-17210.00	-17209.97	-17209.95
PF01427	-3468.21	-3395.71	-3395.71	-3395.68	-3395.64	-3395.63	-3395.62	-3395.49
PF01428	-1170.44	-1128.44	-1128.23	-1127.73	-1127.61	-1127.48	-1127.47	-1127.25
PF01429	-5051.70	-4892.27	-4892.26	-4892.12	-4892.08	-4892.08	-4891.73	-4891.18
PF01431	-12300.42	-11803.81	-11802.78	-11801.36	-11800.64	-11800.62	-11799.91	-11799.31
PF01433	-44661.68	-42667.44	-42667.30	-42666.66	-42664.63	-42663.72	-42661.05	-42655.60
PF01434	-18870.37	-17790.12	-17789.47	-17789.24	-17789.24	-17789.06	-17788.19	-17788.19
PF01435	-13317.99	-12835.11	-12835.11	-12835.02	-12831.80	-12828.96	-12826.78	-12819.85
PF01436	-9164.41	-8690.55	-8690.46	-8690.21	-8690.21	-8690.18	-8689.82	-8686.76

Bibliography

[AB05] S. Aris-Brosou. Determinants of adaptive evolution at the molecular level: the extended complexity hypothesis. *Mol Biol Evol*, 22(2):200–209, 2005.

[AGM+90] S.F. Altschul, W. Gish, W. Miller, E.W. Myers, and D.J. Lipman. Basic local alignment search tool. *J Mol Biol*, 215(3):403–410, 1990.

[AH96] J. Adachi and M. Hasegawa. Model of amino acid substitution in proteins encoded by mitochondrial DNA. *J Mol Evol*, 42(4):459–468, 1996.

[Aka94] H. Akashi. Synonymous codon usage in Drosophila melanogaster: natural selection and translational accuracy. *Genetics*, 136(3):927–35, 1994.

[Aka03] H. Akashi. Translational selection and yeast proteome evolution. *Genetics*, 164(4):1291–303, 2003.

[AMJS05] Z. Abdo, V.N. Minin, P. Joyce, and J. Sullivan. Accounting for uncertainty in the tree topology has little effect on the decision-theoretic approach to model selection in phylogeny estimation. *Mol Biol Evol*, 22(3):691–703, 2005.

[ARWS00] M. Averof, A. Rokas, K.H. Wolfe, and P.M. Sharp. Evidence for a high frequency of simultaneous double-nucleotide substitutions. *Science*, 287(5456):1283–1286, 2000.

[AWMH00] J. Adachi, P.J. Waddell, W. Martin, and M. Hasegawa. Plastid genome phylogeny and a model of amino acid substitution for proteins encoded by chloroplast DNA. *J Mol Evol*, 50(4):348–358, 2000.

[AZFD05] W.R. Atchley, J. Zhao, A.D. Fernandes, and T. Druke. Solving the protein sequence metric problem. *Proc Natl Acad Sci U S A*, 102(18):6395–400, 2005.

[BAC+06] E. Birney, D. Andrews, M. Caccamo, Y. Chen, L. Clarke, G. Coates, T. Cox, F. Cunningham, V. Curwen, T. Cutts, T. Down, R. Durbin, X.M. Fernandez-Suarez, P. Flicek, S. Graf, M. Hammond, J. Herrero, K. Howe, V. Iyer, K. Jekosch, A. Kahari, A. Kasprzyk, D. Keefe, F. Kokocinski, E. Kulesha, D. London, I. Longden, C. Melsopp, P. Meidl, B. Overduin, A. Parker, G. Proctor, A. Prlic, M. Rae, D. Rios, S. Redmond, M. Schuster, I. Sealy, S. Searle, J. Severin, G. Slater, D. Smedley, J. Smith, A. Stabenau, J. Stalker, S. Trevanion, A. Ureta-Vidal, J. Vogel, S. White, C. Woodwark, and T.J. Hubbard. Ensembl 2006. *Nucleic Acids Res*, 34(Database issue):D556–61, 2006.

[BCD+04] A. Bateman, L. Coin, R. Durbin, R.D. Finn, V. Hollich, S. Griffiths-Jones, A. Khanna, M. Marshall, S. Moxon, E.L. Sonnhammer, D.J. Studholme, C. Yeats, and S.R. Eddy. The Pfam protein families database. *Nucleic Acids Res*, 32(Database issue):D138–141, 2004.

[BCG94] S.A. Benner, M.A. Cohen, and G.H. Gonnet. Amino acid substitution during functionally constrained divergent evolution of protein sequences. *Protein Eng*, 7(11):1323–1332, 1994.

[Beh00] E. Behrends. *Introduction to Markov Chains with Special Emphesis on Rapid Mixing.* Viehweg, Wiesbaden, 2000.

[BKO+04] G.A. Bazykin, F.A. Kondrashov, A.Y. Ogurtsov, S. Sunyaev, and A.S. Kondrashov. Positive selection at sites of multiple amino acid replacements since rat-mouse divergence. *Nature*, 429(6991):558–562, 2004.

[Bof06] L. Bofkin. *The Causes and Consequences of Variation in Evolutionary Processes Acting on DNA Sequences.* PhD thesis, Cambridge University, Cambridge, 2006. (Submitted).

[BPWW82] W.M. Brown, E.M. Prager, A. Wang, and A.C. Wilson. Mitochondrial DNA sequences of primates: tempo and mode of evolution. *J Mol Evol*, 18(4):225–39, 1982.

[CAJ+94] Y. Cao, J. Adachi, A. Janke, S. Pääbo, and M. Hasegawa. Phylogenetic relationships among eutherian orders estimated from inferred sequences of mitochondrial proteins: instability of a tree based on a single gene. *J Mol Evol*, 39(5):519–527, 1994.

[CMB01] A. Coghlan, D.A. MacDónaill, and N.H. Buttimore. Representation of amino acids as five-bit or three-bit patterns for filtering protein databases. *Bioinformatics*, 17(8):676–685, 2001.

[CMB+03] E. Camon, M. Magrane, D. Barrell, D. Binns, W. Fleischmann, P. Kersey, N. Mulder, T. Oinn, J. Maslen, A. Cox, and R. Apweiler.

The Gene Ontology Annotation (GOA) project: implementation of GO in SWISS-PROT, TrEMBL, and InterPro. *Genome Res*, 13(4):662–72, 2003.

[Con01] International Human Genome Sequencing Consortium. Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921, 2001.

[Con02] Mouse Genome Sequencing Consortium. Initial sequencing and comparative analysis of the mouse genome. *Nature*, 420(6915):520–562, 2002.

[Con04a] International Human Sequencing Consortium. Finishing the euchromatic sequence of the human genome. *Nature*, 431(7011):931–45, 2004.

[Con04b] Rat Genome Sequencing Project Consortium. Genome sequence of the Brown Norway rat yields insights into mammalian evolution. *Nature*, 428(6982):493–521, 2004.

[CTRV02] N. Cannata, S. Toppo, C. Romualdi, and G. Valle. Simplifying amino acid alphabets by means of a branch and bound algorithm and substitution matrices. *Bioinformatics*, 18(8):1102–1108, 2002.

[DBA+05] D.A. Drummond, J.D. Bloom, C. Adami, C.O. Wilke, and F.H. Arnold. Why highly expressed proteins evolve slowly. *Proc Natl Acad Sci U S A*, 102(40):14338–43, 2005.

[DEKM98] R. Durbin, S. Eddy, A. Krogh, and G. Mitchison. *Biological Sequence Analysis.* Cambridge University Press, Cambridge, 1998.

[DEP72] M.O. Dayhoff, R.V. Eck, and C.M. Park. A model of evolutionary change in proteins. In M.O. Dayhoff, editor, *Atlas of protein sequence and structure*, volume 5, pages 89–99. Biomedical Research Foundation, Washington,D.C., 1972.

[DHFS00] P. Deuflhard, W. Huisinga, A. Fischer, and Ch. Schütte. Identification of almost invariant aggregates in reversible nearly uncoupled Markov chains. *Linear Algebra Appl*, 315:39–59, 2000.

[DMG00] M.W. Dimmic, D.P. Mindell, and R.A. Goldstein. Modeling evolution at the protein level using an adjustable amino acid fitness model. *Pac Symp Biocomput*, pages 18–29, 2000.

[Doo99] W.F. Doolittle. Phylogenetic classification and the universal tree. *Science*, 284(5423):2124–2129, 1999.

[DRMG02] M.W. Dimmic, J.S. Rest, D.P. Mindell, and R.A. Goldstein. rtREV: an amino acid substitution matrix for inference of retrovirus and reverse transcriptase phylogeny. *J Mol Evol*, 55(1):65–73, 2002.

[DSO78] M.O. Dayhoff, R.M. Schwarz, and B.C. Orcutt. A model of evolutionary change in proteins. In M.O. Dayhoff, editor, *Atlas of protein sequence and structure*, volume 5, suppl. 3, pages 345–352. National Biomedical Research Foundation, Washington,D.C., 1978.

[DWH05] M.A. DePristo, D.M. Weinreich, and D.L. Hartl. Missense meanderings in sequence space: a biophysical view of protein evolution. *Nat Rev Genet*, 6(9):678–87, 2005.

[Edd98] S.R. Eddy. Profile hidden Markov models. *Bioinformatics*, 14(9):755–763, 1998.

[Edw72] A.W.F. Edwards. *Likelihood.* Cambridge University Press., Cambridge, 1972.

[EIG96] T. Endo, K. Ikeo, and T. Gojobori. Large-scale search for genes on which positive selection may operate. *Mol Biol Evol*, 13(5):685–690, 1996.

[ET93] B. Efron and R.J. Tibshirani. *An Introduction to the Bootstrap.* Chapman and Hall, New York, 1993.

[EW96] A. Eyre-Walker. Synonymous codon bias is related to gene length in Escherichia coli: selection for translational accuracy? *Mol Biol Evol*, 13(6):864–72, 1996.

[EWB93] A. Eyre-Walker and M. Bulmer. Reduced synonymous substitution rate at the start of enterobacterial genes. *Nucleic Acids Res*, 21(19):4599–603, 1993.

[Exc03] L. Excoffier. Analysis of Population Subdivison. In D.J. Balding, M. Bishop, and C. Cannings, editors, *Handbook of Statistical Genetics*, volume II, pages 713–745. Wiley, New York, 2003.

[EY99] L. Excoffier and Z. Yang. Substitution rate variation among sites in mitochondrial hypervariable region I of humans and chimpanzees. *Mol Biol Evol*, 16(10):1357–68, 1999.

[FC96] J. Felsenstein and H.A. Churchill. A hidden Markov model approach to variation among sites in rate of evolution. *Mol Biol Evol*, 13:93–104, 1996.

[Fel81] J. Felsenstein. Evolutionary trees from DNA sequences: a maximum likelihood approach. *J Mol Evol*, 17(6):368–376, 1981.

[Fel03] J. Felsenstein. *Inferring Phylogenies.* Sinauer Associates, Massachusetts, 2003.

[FH05] R. Friedman and A.L. Hughes. The pattern of nucleotide difference at individual codons among mouse, rat, and human. *Mol Biol Evol*, 22(5):1285–1289, 2005.

[Fis25] R.A. Fisher. Theory of statistical estimation. *Proc Camb Phil Soc*, 22:700–725, 1925.

[FKH+03] P. Flicek, E. Keibler, P. Hu, I. Korf, and M.R. Brent. Leveraging the mouse genome for gene prediction in human: from whole-genome shotgun reads to a global synteny map. *Genome Res*, 13(1):46–54, 2003.

[FR83] S. French and B. Robson. What is a conservative substitution? *J Mol Evol*, 19:171–175, 1983.

[GG95] N. Galtier and M. Gouy. Inferring phylogenies from DNA sequences of unequal base compositions. *Proc Natl Acad Sci U S A*, 92(24):11317–21, 1995.

[GG98] N. Galtier and M. Gouy. Inferring pattern and process: maximum-likelihood implementation of a nonhomogeneous model of DNA sequence evolution for phylogenetic analysis. *Mol Biol Evol*, 15(7):871–9, 1998.

[GGJM05] N. Galtier, O. Gascuel, and A. Jean-Marie. Markov Models in Molecular Evolution. In R. Nielsen, editor, *Stastistical Methods in Molecular Evolution*, pages 4–24. Springer, New York, 2005.

[Gil92] D.T. Gillespie. *Markov processes: an introduction for physical scientists*, pages 61–64. London Academic Press, Boston, 1992.

[Gol93] N. Goldman. Statistical tests of models of DNA substitution. *J Mol Evol*, 36(2):182–198, 1993.

[Gra74] R. Grantham. Amino acid difference formula to help explain protein evolution. *Science*, 185(4154):862–864, 1974.

[GTJ98] N. Goldman, J.L. Thorne, and D.T. Jones. Assessing the impact of secondary structure and solvent accessibility on protein evolution. *Genetics*, 149(1):445–58, 1998.

[GW00] N. Goldman and S. Whelan. Statistical tests of gamma-distributed rate heterogeneity in models of sequence evolution in phylogenetics. *Mol Biol Evol*, 17(6):975–978, 2000.

[GW02] N. Goldman and S. Whelan. A novel use of equilibrium frequencies in models of sequence evolution. *Mol Biol Evol*, 19(11):1821–1831, 2002.

[GY94] N. Goldman and Z. Yang. A codon-based model of nucleotide substitution for protein-coding DNA sequences. *Mol Biol Evol*, 11(5):725–736, 1994.

[Hap03] The International HapMap Project. *Nature*, 426(6968):789–96, 2003.

[HH91] D. Haig and L.D. Hurst. A quantitative measure of error minimization in the genetic code. *J Mol Evol*, 33(5):412–7, 1991.

[HH92] S. Henikoff and J.G. Henikoff. Amino acid substitution matrices from protein blocks. *Proc Natl Acad Sci U S A*, 89(22):10915–10919, 1992.

[Hig98] P.G. Higgs. Compensatory neutral mutations and the evolution of RNA. *Genetica*, 102-103(1-6):91–101, 1998.

[HKY85] M. Hasegawa, H. Kishino, and T. Yano. Dating of the human-ape splitting by a molecular clock of mitochondrial DNA. *J Mol Evol*, 22(2):160–174, 1985.

[HR02] I. Holmes and G.M. Rubin. An expectation maximization algorithm for training hidden substitution models. *J Mol Biol*, 317(5):753–764, 2002.

[Ike81] T. Ikemura. Correlation between the abundance of Escherichia coli transfer RNAs and the occurrence of the respective codons in its protein genes: a proposal for a synonymous codon choice that is optimal for the E. coli translational system. *J Mol Biol*, 151(3):389–409, 1981.

[JC69] T.H. Jukes and C.R. Cantor. Evolution of protein molecules. In H.N. Munro, editor, *Mammalian Protein Metabolism*, pages 21–132. Academic Press, New York, NY, 1969.

[JKA+05] I.K. Jordan, F.A. Kondrashov, I.A. Adzhubei, Y.I. Wolf, E.V. Koonin, A.S. Kondrashov, and S. Sunyaev. A universal trend

of amino acid gain and loss in protein evolution. *Nature*, 433(7026):633–638, 2005.

[JTT92] D.T. Jones, W.R. Taylor, and J.M. Thornton. The rapid generation of mutation data matrices from protein sequences. *Comput Appl Biosci*, 8(3):275–282, 1992.

[JTT94] D.T. Jones, W.R. Taylor, and J.M. Thornton. A mutation data matrix for transmembrane proteins. *FEBS Lett*, 339(3):269–275, 1994.

[KBD+03] D. Karolchik, R. Baertsch, M. Diekhans, T.S. Furey, A. Hinrichs, Y.T. Lu, K.M. Roskin, M. Schwartz, C.W. Sugnet, D.J. Thomas, R.J. Weber, D. Haussler, and W.J. Kent. The UCSC Genome Browser Database. *Nucleic Acids Res*, 31(1):51–54, 2003.

[KEW98] R.M. Kliman and A. Eyre-Walker. Patterns of base composition within the genes of Drosophila melanogaster. *J Mol Evol*, 46(5):534–41, 1998.

[KG95] J.M. Koshi and R.A. Goldstein. Context-dependent optimal substitution matrices. *Protein Eng*, 8(7):641–645, 1995.

[KG05] C. Kosiol and N. Goldman. Different versions of the Dayhoff rate matrix. *Mol Biol Evol*, 22(2):193–199, 2005.

[KGB04] C. Kosiol, N. Goldman, and N.H. Buttimore. A new criterion and method for amino acid classification. *J Theor Biol*, 228(1):97–106, 2004.

[KLEW05] P.D. Keightley, M.J. Lercher, and A. Eyre-Walker. Evidence for widespread degradation of gene control regions in hominid genomes. *PLoS Biol*, 3(2):e42, 2005.

[KMG97] J.M. Koshi, D.P. Mindell, and R.A. Goldstein. Beyond mutation matrices: physical-chemistry based evolutionary models. *Genome Inform Ser Workshop Genome Inform*, 8:80–89, 1997.

[KSK02] A.S. Kondrashov, S. Sunyaev, and F.A. Kondrashov. Dobzhansky-Muller incompatibilities in protein evolution. *Proc Natl Acad Sci U S A*, 99(23):14878–83, 2002.

[Kue01] H.R. Kuensch. State space and hidden Markov models. In O.E. Barndorff-Nielsen, D.R. Cox, and C. Klüppelberg, editors, *Complex Stochastic Systems*, pages 109–173. CRC Press, New York, 2001.

[Lar98] B. Larget. A canonical representation for aggregated Markov processes. *J Appl Probab*, 32(2):313–324, 1998.

[LG98] P. Lio and N. Goldman. Models of molecular evolution and phylogeny. *Genome Res*, 8(12):1233–1244, 1998.

[LG05] A. Loytynoja and N. Goldman. An algorithm for progressive multiple alignment of sequences with insertions. *Proc Natl Acad Sci U S A*, 102(30):10557–10562, 2005.

[LH04] G.A. Lunter and J. Hein. A nucleotide substitution model with nearest neighbour interactions. *Bioinformatics*, 20 (supp.1):i216–i223, 2004.

[LPH+92] P.J. Lockhart, D. Penny, M.D. Hendy, C.J. Howe, T.J. Beanland, and A.W. Larkum. Controversy on chloroplast origins. *FEBS Lett*, 301(2):127–31, 1992.

[LTWM+05] K. Lindblad-Toh, C.M. Wade, T.S. Mikkelsen, E.K. Karlsson, D.B. Jaffe, M. Kamal, M. Clamp, J.L. Chang, E.J. Kulbokas, M.C. Zody, E. Mauceli, X. Xie, M. Breen, R.K. Wayne, E.A. Ostrander, C.P. Ponting, F. Galibert, D.R. Smith, P.J. DeJong, E. Kirkness, P. Alvarez, T. Biagi, W. Brockman, J. Butler, C.W. Chin, A. Cook, J. Cuff, M.J. Daly, D. DeCaprio, S. Gnerre, M. Grabherr, M. Kellis, M. Kleber, C. Bardeleben, L. Goodstadt, A. Heger, C. Hitte, L. Kim, K.P. Koepfli, H.G. Parker, J.P. Pollinger, S.M. Searle, N.B. Sutter, R. Thomas, C. Webber, Broad Sequencing Platform members, and E.S. Lander. Genome sequence, comparative analysis and haplotype structure of the domestic dog. *Nature*, 438(7069):803–19, 2005.

[Mas02] T. Massingham. *Detecting Postitive Selection in Proteins: Models of Evolution and Statistical Tests.* PhD thesis, Dept. Zoology, University of Cambridge., 2002.

[MBHG03] E.H. Margulies, M. Blanchette, D. Haussler, and E.D. Green. Identification and characterization of multi-species conserved sequences. *Genome Res*, 13(12):2507–2518, 2003.

[McD06] J. H. McDonald. Apparent trends of amino acid gain and loss in protein evolution due to nearly neutral variation. *Mol Biol Evol*, 23(2):240–244, 2006.

[McV05] G.A.T. McVean. Base composition variation. In R. Nielsen, editor, *Statistical Methods in Molecular Evolution*, pages 355–374. Springer, 2005.

[MD95] G. Mitchison and R. Durbin. Tree-based maximal liklihood substitution matrices and hidden markov models. *J Mol Evol*, 41:1139–1151, 1995.

[MD02] I.M. Meyer and R. Durbin. Comparative *ab initio* prediction of gene structures using pair HMMs. *Bioinformatics*, 18(10):1309–1318, 2002.

[MF87] K.B. Mullis and F.A. Faloona. Specific synthesis of DNA in vitro via a polymerase-catalyzed chain reaction. *Methods Enzymol*, 155:335–350, 1987.

[MFS+86] K. Mullis, F. Faloona, S. Scharf, R. Saiki, G. Horn, and H. Erlich. Specific enzymatic amplification of DNA in vitro: the polymerase chain reaction. *Cold Spring Harb Symp Quant Biol*, 51 Pt 1:263–273, 1986.

[MG94] S.V. Muse and B.S. Gaut. A likelihood approach for comparing synonymous and nonsynonymous nucleotide substitution rates, with application to the chloroplast genome. *Mol Biol Evol*, 11(5):715–724, 1994.

[MG05] T. Massingham and N. Goldman. Detecting amino acid sites under positive selection and purifying selection. *Genetics*, 169(3):1753–1762, 2005.

[MRR+53] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller. Equation of state calculation for fast computing machines. *J. Chem. Phys.*, 21:1087–1092, 1953.

[MV00] T. Muller and M. Vingron. Modeling amino acid replacement. *J Comput Biol*, 7(6):761–76, 2000.

[MVL03] C. Moler and C. Van Loan. Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. *SIAM Rev*, 45:3–49, 2003.

[NBC+05] R. Nielsen, C. Bustamante, A.G. Clark, S. Glanowski, T.B. Sackton, M.J. Hubisz, A. Fledel-Alon, D.M. Tanenbaum, D. Civello, T.J. White, J. J Sninsky, M.D. Adams, and M. Cargill. A scan for positively selected genes in the genomes of humans and chimpanzees. *PLoS Biol*, 3(6):e170, 2005.

[Neu03] C. Neuhauser. Mathematical Models in Population Genetics. In D.J. Balding, M. Bishop, and C. Cannings, editors, *Handbook of Statistical Genetics*, volume II, pages 577–599. Wiley, 2003.

[Ney71] J. Neyman. Molecular studies of evolution: a source of novel statistical problems. In J. Gupta, S. and Yackel, editor, *Statistical Theory and Related Topics*, pages 1–27. Academic Press, New York, 1971.

[NFN03] H. Nakashima, S. Fukuchi, and K. Nishikawa. Compositional changes in RNA, DNA and proteins for bacterial adaptation to higher and lower temperatures. *J Biochem (Tokyo)*, 133(4):507–13, 2003.

[NG86] M. Nei and T. Gojobori. Simple methods for estimating the numbers of synonymous and nonsynonymous nucleotide substitutions. *Mol Biol Evol*, 3(5):418–426, 1986.

[Nor97] J.R. Norris. *Markov Chains.* Cambridge University Press, Cambridge, 1997.

[Nor03] M. Nordborg. Coalescent Theory. In D.J. Balding, M. Bishop, and C. Cannings, editors, *Handbook of Statistical Genetics*, volume II, pages 602–631. Wiley, New York, 2003.

[ODJ+92] J. Overington, D. Donnelly, M.S. Johnson, A. Sali, and T.L. Blundell. Environment-specific amino acid substitution tables: tertiary templates and prediction of protein folds. *Protein Sci*, 1(2):216–26, 1992.

[OJSB90] J. Overington, M.S. Johnson, A. Sali, and T.L. Blundell. Tertiary structural constraints on protein evolutionary diversity: templates, key residues and structure prediction. *Proc Biol Sci*, 241(1301):132–145, 1990.

[PC02] D. Posada and K.A. Crandall. The effect of recombination on the accuracy of phylogenetic estimation. *J Mol Evol*, 54:396–402, 2002.

[PDC05] A. Poon, B.H. Davis, and L. Chao. The coupon collector and the suppressor mutation: estimating the number of compensatory mutations by maximum likelihood. *Genetics*, 170(3):1323–32, 2005.

[Pea88] J. Pearl. *Probabilistic Reasoning in Intelligent Systems.* Morgan Kaufmann Publishers, San Mateo, CA, 1988.

[PPL06] C. Pal, B. Papp, and M.J. Lercher. An integrated view of protein evolution. *Nat Rev Genet*, 7(5):337–48, 2006.

[RD04] E.P. Rocha and A. Danchin. An analysis of determinants of amino acids substitution rates in bacterial proteins. *Mol Biol Evol*, 21(1):108–16, 2004.

[RJK+03] D.M. Robinson, D.T. Jones, H. Kishino, N. Goldman, and J.L. Thorne. Protein evolution with dependence among codons due to tertiary structure. *Mol Biol Evol*, 20(10):1692–1704, 2003.

[RTY05] F. Ren, H. Tanaka, and Z. Yang. An empirical examination of the utility of codon substitution models in phylogenetic reconstruction. *Syst. Biol.*, 54:808–818, 2005.

[SAJS05] J. Sullivan, Z. Abdo, P. Joyce, and D.L. Swofford. Evaluating the performance of a successive-approximations approach to parameter optimization in maximum-likelihood phylogeny estimation. *Mol Biol Evol*, 22(6):1386–1392, 2005.

[Saw89] S. Sawyer. Statistical tests for detecting gene conversion. *Mol Biol Evol*, 6(5):526–538, 1989.

[SC05] Chimpanzee Sequencing and Analysis Consortium. Initial sequence of the chimpanzee genome and comparison with the human genome. *Nature*, 437(7055):69–87, 2005.

[SCG05] A. Schneider, G.M. Cannarozzi, and G.H. Gonnet. Empirical codon substitution matrix. *BMC Bioinformatics*, 6(1):134, 2005.

[SH03] G.A. Singer and D.A. Hickey. Thermophilic prokaryotes have characteristic patterns of codon usage, amino acid composition and nucleotide content. *Gene*, 317(1-2):39–47, 2003.

[SH05] A. Siepel and D. Haussler. Phylogenetic hidden Markov models. In R. Nielsen, editor, *Statistical Methods in Molecular Evolution.*, pages 325–351. Springer, New York, 2005.

[SHH01] N.J. Savill, D.C. Hoyle, and P.G. Higgs. RNA sequence evolution with secondary structure constraints: comparison of substitution rate models using maximum-likelihood methods. *Genetics*, 157(1):399–411, 2001.

[SHS96] J. Sullivan, K.E. Holsinger, and C. Simon. The effect of topology on estimates of among-site rate variation. *J Mol Evol*, 42(2):308–312, 1996.

[Sin92] A. Sinclair. Improved bounds for mixing rates of Markov chains and multicommodity flow. *Combin Probab Comp*, 1:315–370, 1992.

[SNC77] F. Sanger, S. Nicklen, and A.R. Coulson. DNA sequencing with chain-terminating inhibitors. *Proc Natl Acad Sci U S A*, 74(12):5463–5467, 1977.

[SSHW88] D.C. Shields, P.M. Sharp, D.G. Higgins, and F. Wright. "Silent" sites in Drosophila genes are not neutral: evidence of selection among synonymous codons. *Mol Biol Evol*, 5(6):704–16, 1988.

[SW99] C. Semple and K.H. Wolfe. Gene duplication and gene conversion in the *caenorhabditis elegans* genome. *J Mol Evol*, 48(5):555–564, 1999.

[Tay86] W.R. Taylor. The classification of amino acid conservation. *J Theor Biol*, 119(2):205–218, 1986.

[Tay97] W.R. Taylor. Residual colours: a proposal for aminochromography. *Protein Eng*, 10(7):743–746, 1997.

[THG94] J.D. Thomson, D.G. Higgins, and T.J. Gibson. Clustal W: improving the sensitivity of progressive multiple sequnence alignment through sequence weighting, position specific gap penalities and weight matrix choice. *Nucleic Acids Res*, 22:4673–4680, 1994.

[Tra] Trace Archive v.3. http://www.ncbi.nih.gov/Traces.

[Tri94] M. Trick. http://mat.gsia.cmu.edu/COLOR/color.html, 1994.

[UC71] T. Uzzel and K.W. Corbin. Fitting discrete probability distributions to evolutionary events. *Science*, 172:1089–1096, 1971.

[Wak96] J. Wakeley. The excess of transitions among nucleotide substitutions: New methods of estimating transition bias underscore its significance. *TREE*, 11:436–442, 1996.

[WdBG03] S. Whelan, P.I. de Bakker, and N. Goldman. Pandit: a database of protein and associated nucleotide domains with inferred trees. *Bioinformatics*, 19(12):1556–1563, 2003.

[WdBQ+06] S. Whelan, P.I. de Bakker, E. Quevillon, N. Rodriguez, and N. Goldman. PANDIT: an evolution-centric database of protein and associated nucleotide domains with inferred trees. *Nucleic Acids Res*, 34(Database issue):D327–331, 2006.

[Wei99] E.W. Weissenstein. http://mathworld.wolfram.com/BellNumber.html, 1999.

[Wel05] G.H. Wells. *The Time Machine.* Penguin Books Ltd., London, 2005.

[WG99] S. Whelan and N. Goldman. Distributions of statistics used for the comparison of models of sequence evolution in phylogenetics. *Mol Biol Evol*, 16(9):1292–1299, 1999.

[WG01] S. Whelan and N. Goldman. A general empirical model of protein evolution derived from multiple protein families using a maximum-likelihood approach. *Mol Biol Evol*, 18(5):691–699, 2001.

[WG04] S. Whelan and N. Goldman. Estimating the frequency of events that cause multiple-nucleotide changes. *Genetics*, 167(4):2027–2043, 2004.

[WLG01] S. Whelan, P. Lio, and N. Goldman. Molecular phylogenetics: state-of-the-art methods for looking into the past. *Trends Genet*, 17(5):262–272, 2001.

[WW99] J. Wang and W. Wang. A computational approach to simplifying the protein folding alphabet. *Nat Struct Biol*, 6(11):1033–1038, 1999.

[WYGN04] W.S. Wong, Z. Yang, N. Goldman, and R. Nielsen. Accuracy and power of statistical methods for detecting adaptive evolution in protein coding sequences and for identifying positively selected sites. *Genetics*, 168(2):1041–1051, 2004.

[XL98] X. Xia and W.H. Li. What amino acid properties affect protein evolution? *J Mol Evol*, 47(5):557–564, 1998.

[Yan93] Z. Yang. Maximum likelihood estimation of phylogeny from DNA sequences when substitution rates differ over sites. *Mol Biol Evol*, 19:1396–1401, 1993.

[Yan94a] Z. Yang. Estimating the pattern of nucleotide substitution. *J Mol Evol*, 39(1):105–111, 1994.

[Yan94b] Z. Yang. Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: approximate methods. *J Mol Evol*, 39(3):306–314, 1994.

[Yan96] Z. Yang. Among-site rate variation and its impact on phylogenetic analysis. *Trends Ecol Evol*, 11:367–372, 1996.

[Yan97] Z. Yang. PAML: a program package for phylogenetic analysis by maximum likelihood. *Comput Appl Biosci*, 13(5):555–556, 1997.

[YB00] Z. Yang and J.P. Bielawski. Statistical methods for detecting molecular adaptation. *Trends Ecol Evol*, 15(12):496–503, 2000.

[YN02] Z. Yang and R. Nielsen. Codon-substitution models for detecting molecular adaptation at individual sites along specific lineages. *Mol Biol Evol*, 19(6):908–17, 2002.

[YNGP00] Z. Yang, R. Nielsen, N. Goldman, and A.-M. Pedersen. Codon-substitution models for heterogeneous selection pressure at amino acid sites. *Genetics*, 155(1):431–449, 2000.

[YNH98] Z. Yang, R. Nielsen, and M. Hasegawa. Models of amino acid substitution and applications to mitochondrial protein evolution. *Mol Biol Evol*, 15(12):1600–1611, 1998.

[YS05] L.Y. Yampolsky and A. Stoltzfus. The exchangeability of amino acids in proteins. *Genetics*, 170(4):1459–72, 2005.

[YY99] Z. Yang and A.D. Yoder. Estimation of the transition/transversion rate bias and species sampling. *J Mol Evol*, 48(3):274–83, 1999.

[ZP62] E. Zuckerkandl and L. Pauling. Molecular disease, evolution and genetic heterogeneity. In M. Marsha and B. Pullman, editors, *Horizons in Biochemistry*, pages 189–225. Academic Press, New York, 1962.