Statistical Analysis of Coevolution in Protein Structure and in Ecology

Colleen Nooney

Submitted in accordance with the requirements for the degree of

Doctor of Philosophy

The University of Leeds

Department of Statistics

September 2016

The candidate confirms that the work submitted is her own, except where work which has formed part of jointly authored publications has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated below. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others.

Jointly authored publications:

Nooney, C., Gusnanto, A., Gilks, W.R. & Barber, S. (2015). Do protein structures evolve around 'anchor' residues? In I.L. Dryden & J.T. Kent, eds., *Geometry Driven Statistics*, chap. 16, 311–336, John Wiley & Sons

Chapter 2 of the thesis is based on this publication. The contribution of authors is as follows:

Colleen Nooney implemented the methodology and Arief Gusnanto noted the anomaly (low range divergences). All other results were drafted by Colleen Nooney, and include contributions and edits from all four authors.

I would like to dedicate this thesis to my Supervisors for never doubting my abilities, even when I doubt myself.

Acknowledgements

I am most thankful to my supervisors Prof. Walter Gilks, Dr Stuart Barber and Dr Arief Gusnanto. Over the past four years their experience and guidance has been invaluable. They have always given me the freedom and confidence to explore my own ideas, and their unwavering enthusiasm for the topic has always motivated me.

I am also grateful to the Engineering and Physical Sciences Research Council for funding my Masters and PhD studies. Without this support I never would have discovered my passion for research.

I am thankful to the following people for their help and support during the organisation of RSC at Leeds: Jeanne Shuttleworth, Margaret Jones, Paula Talbot, Nebahat Bozkus, Wafa Almohri and Keith Newman. Every bit of help was a huge weight off of my shoulders. Additional thanks go to Keith for always being there for me, in all aspects of life.

Finally, thank you to all of my family and friends, for the support, understanding, happiness and sanity. Particularly to my parents and James for their unwavering belief and encouragement.

Abstract

In this thesis we explore the theory of coevolution. Yip *et al.* (2008) define coevolution to be the change in one biological object as a result of the change in one or more associated objects. The process of coevolution has been observed at many biological levels; from microscopic to macroscopic. We explore coevolution at the molecular level by studying protein sequences and their corresponding structures to determine how correlated areas of multiple sequence alignments and structures have coevolved. At the species level, we assess how coevolution drives ecological systems of interacting phylogenetic trees.

Determining the three-dimensional structure of proteins is of interest because the structure of a protein is constrained by its function. Proteins carry out vital functions in every cell and are arguably the most important biological molecule found in organisms. Multiple sequence alignments of protein families contain evolutionary information on these functional constraints. In the first part of this thesis, we aim to develop a method to identify correlated mutations within multiple sequence alignments. These correlated positions are used to predict residues that are in close proximity in three-dimensional space. In turn these structural constraints can be used in *ab initio* protein structure prediction. Currently the most accurate way to determine protein structure is using experimental techniques such as Nuclear Magnetic Resonance (NMR) and X-ray Crystallography. These techniques are expensive and take time. As a result, the proteins that are chosen to have their structures determined may be subject to selection bias. Initially, we focus on a preliminary analysis of the trypsin protein family. We align trypsin structures from a variety of species using a multiple structural alignment algorithm, to determine how the structure of the family has evolved. Basic summary statistics of the aligned distance matrices reveal a set of residues where the distance between these specific residues and every other residue in

the structure is highly conserved across all of the structures in the protein family. We label these residues as 'anchor residues' because they appear to hold the structure of the trypsin protein family in place like anchors.

Following this, we develop a regularised logistic regression model to detect correlated mutations in multiple sequence alignments. We successfully apply our method to a number of small artificial test alignments. When applied to real Pfam datasets, our method has varying success at identifying coevolving columns that are close in physical proximity.

In the second part of this thesis we develop a new method to test efficiently for cospeciation in multitrophic ecological systems. Our method can be applied to bitrophic and tritrophic systems, with the potential to generalise to higher order systems and networks. We utilise methods from electrical circuit theory to reduce higher order systems into two vectors of electrically equivalent patristic distances that can be compared using Spearman's rank correlation coefficient. Compared to existing methods, our method has equal or higher performance at both trophic levels.

To test our method, interacting systems of phylogenetic trees were simulated by generating random trees, and separately, their interaction matrices. Simulating the systems in this way does not take into account how the systems might have evolved. We propose a more realistic simulation method that evolves over time. The algorithm starts with one species per lineage, that are assumed to have an ecological interaction. The joint evolution of these species is simulated by sampling the time at which evolutionary events occur from an exponential distribution. We explore speciation events, and gaining and losing ecological interactions. Each of these events are controlled by rate parameters. By experimenting with these parameters, a wide range of systems with different cospeciation properties can be simulated. We show that a wide range of systems that can be produced using our method.

Contents

List of Figures

Chapter 1

Introduction

The process of coevolution drives many biological systems. Identifying where coevolution occurs within these systems provides insight into how they function. We focus on two biological systems. At the molecular level, we explore coevolution as an indicator of residue contacts in three-dimensional protein structures. At the species level we investigate coevolution in ecological systems consisting of interacting phylogenetic trees.

We provide a brief biological background for each level in this chapter. More detail unique to each of the chapters is given within their introductory sections. In Section 1.5 we give a chapter by chapter overview of the research covered in this thesis.

1.1 What is Coevolution?

Yip *et al.* (2008) define coevolution to be the change in one biological object as a result of the change in one or more associated objects. This evolutionary process occurs at many biological levels; from microscopic to macroscopic. For example, at the molecular level positions in multiple sequence alignments coevolve to conserve protein structure (Marks *et al.*, 2011). Coevolution can also occur between genes, and proteins, that physically interact or have functional relationships (Lovell & Robertson, 2010).

At the species level, coevolution, or cospeciation, is where two or more lineages that are ecologically associated jointly evolve to form new species (Page, 2003). The terms cospeciation and coevolution are often used interchangeably. However, there are differing opinions on how to define cospeciation and coevolution. Page (2003)

defines coevolution as the evolution of reciprocal adaptations in hosts and parasites, that is, the evolution of new traits or characteristics as opposed to new species. Thus, coevolution does not imply cospeciation.

1.2 Biological Background: Protein Structure

1.2.1 Protein Sequences

Proteins are biological macromolecules comprised of polypeptide chains; these in turn are made up of amino acid residues. There are 20 standard amino acids; their names and abbreviations are displayed in Table 1.1. Figure 1.1 displays the chemical structure common to all amino acids. Every amino acid is comprised of a central, or alpha, carbon atom (C_α), an amine group (NH_2), a carboxyl group (COOH), and an R group. The R group is connected to the C_α atom and represents the unique side chain that differentiates the 20 standard amino acids. To form the polypeptide chain, the amino-acid residues are combined by peptide bonds, resulting in the loss

Full Name	Abbreviation	Single Letter Code
Alanine	Ala	A
Arginine	Arg	R
Asparagine	Asn	N
Aspartic Acid	Asp	D
Cysteine	Cys	C
Glutamine	Gln	Q
Glutamic Acid	Glu	E
Glycine	Gly	G
Histidine	His	H
Isoleucine	Ile	I
Leucine	Leu	L
Lysine	Lys	K
Methionine	Met	M
Phenylalanine	Phe	F
Proline	Pro	P
Serine	Ser	S
Threonine	Thr	T
Tryptophan	Trp	W
Tyrosine	Tyr	Y
Valine	Val	V

Table 1.1: Full name, three letter abbreviated name and single letter code for each of the 20 standard amino acids.

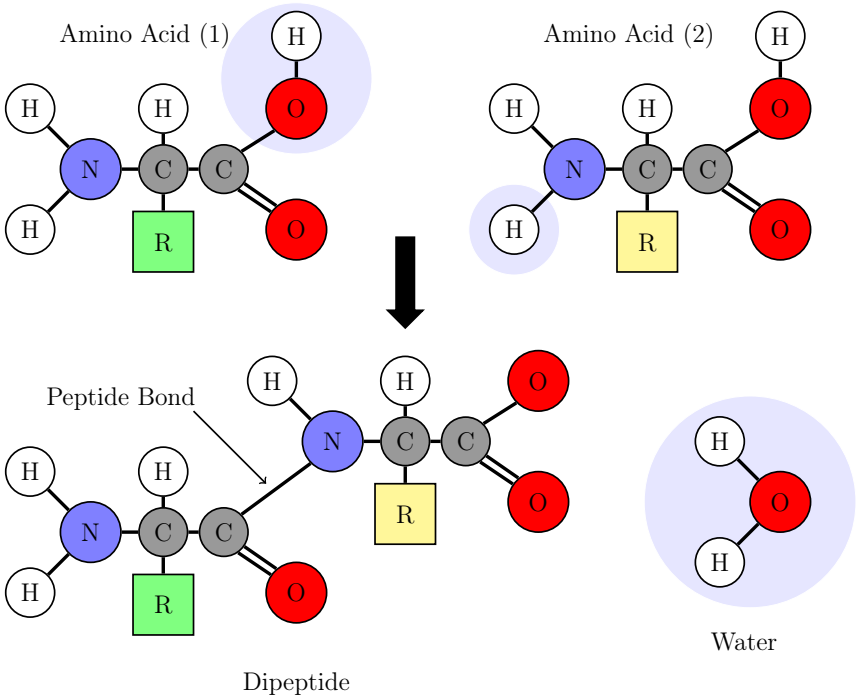


Figure 1.1: A two-dimensional ball-and-stick model of peptide bond formation between two amino acids. Atoms are represented by circles and bonds are lines between them, where double bonds are indicated by two parallel lines. Nitrogen, Carbon, Oxygen and Hydrogen are represented by 'N' (blue), 'C' (grey), 'O' (red) and 'H' (white) respectively. The unique side-chains or 'R' groups of the two amino acids are represented by a square. Peptide bonds are formed when the carboxyl group of one amino acid reacts with the amino group of another resulting in the loss of a water molecule, as shown in the lower panel.

of a water molecule for each link.

Amino acids can be categorised according to their physiochemical properties. These properties include; size, charge, functional group, hydrophobicity and hydrophilicity. Polar uncharged amino acids (S, T, Q, N, Y, C) are hydrophilic, and can therefore form hydrogen bonds. Non-polar amino acids (G, A, V, L, I, M, F, W, P) are hydrophobic, and therefore usually found in the centre of globular proteins, with hydrophilic amino acids on the outside. Electrically charged amino acids (D, E, K, R, H) have electrical properties and are thus influenced by pH levels. Cysteine and Proline have unique properties. Cysteine residues can covalentently bond with other cysteine residues to form disulphide bonds. Disulphide bonds are important in the folding and structure of proteins. Proline is the only amino acid whose R chain connects to the protein backbone twice. This results in a cyclic structure that is conformationally more rigid than the other amino acids. Proline's unique structure can produce kinks in the protein chain.

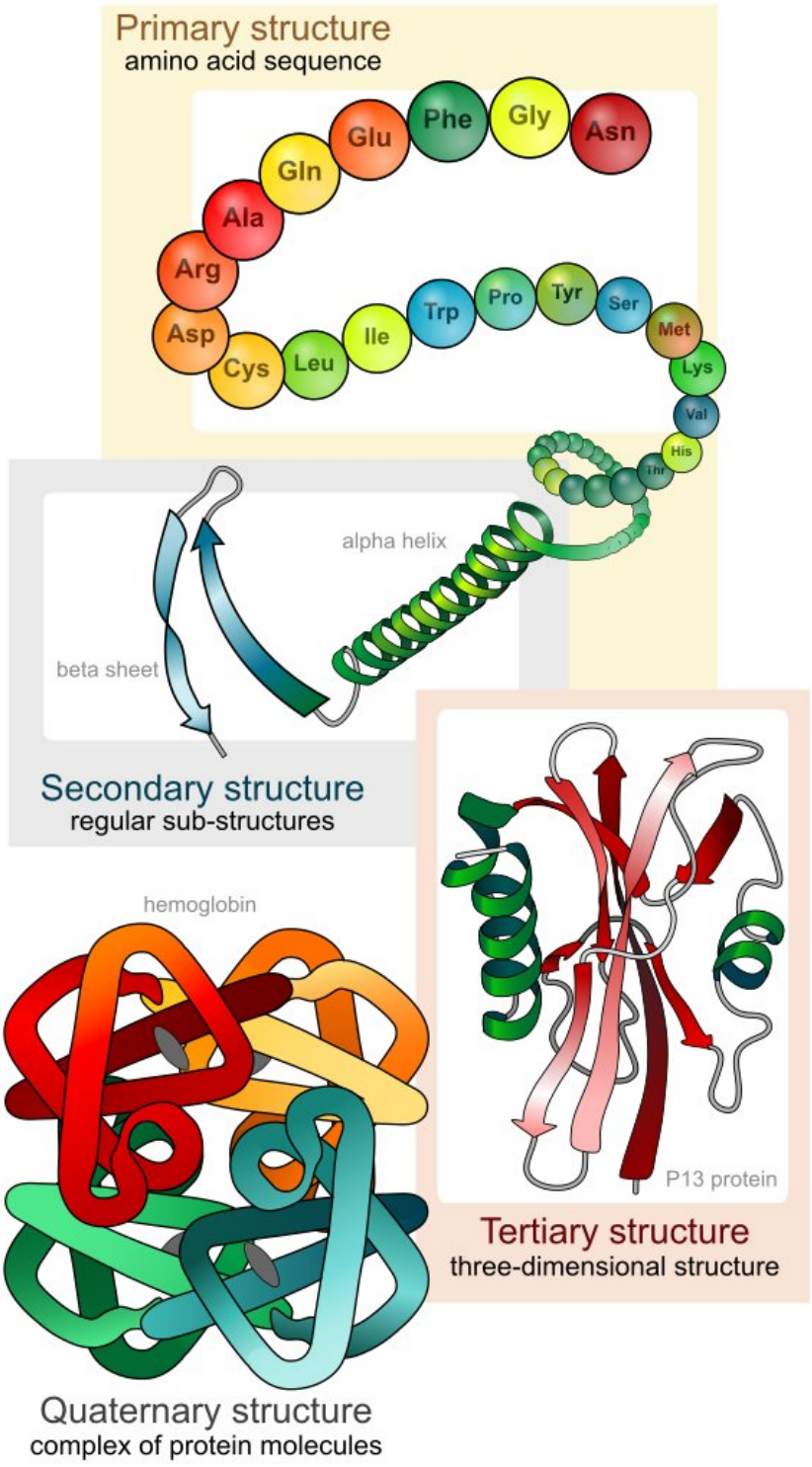
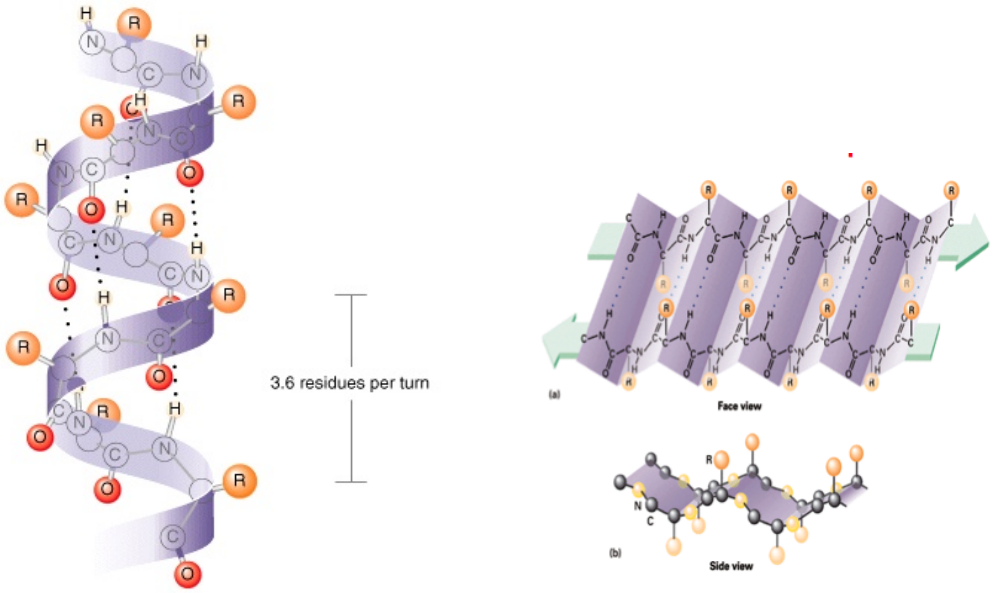


Figure 1.2: Four different levels of protein structure; primary, secondary, tertiary and quaternary structure. Source: `http://en.wikipedia.org/wiki/File:Main_protein_structure_levels_en.svg`

1.2.2 Levels of Protein Structure

The complex structure of a protein is determined by four different levels of folding, known as the primary, secondary, tertiary and quaternary structures. A simple overview is given in Figure 1.2. The primary structure is the sequence of amino-acid residues of each polypeptide chain.

The secondary structures of a protein are the regions of the polypeptide chain that are organised into regular structures identified as alpha helices, and beta-pleated sheets (Figure 1.3). Alpha helices are the most common type of secondary structure. The protein chain twists into a coil held together by hydrogen bonds where the side-chains of the amino acids point outwards. The helix is orientated in an anti-clockwise direction, with approximately 3.6 amino-acid residues per turn. Beta sheets are rigid planar surfaces formed when two or more strands of the protein chain lie side by side. This structure is also held together by hydrogen bonds. The side chains lie alternately above and below the plane of the surface of the beta sheet. Between the organised secondary structure regions are less structured loops and turns, which are less rigid and able to move more freely.



(a) (b)

Figure 1.3: Two types of secondary structure; alpha helix (a) and beta pleated sheet (b). Dotted lines indicate where hydrogen bonds form, stabilising the structures. Sources: `http://www.mun.ca/biology/scarr/F09-05.jpg`, `http://classconnection.s3.amazonaws.com/804/flashcards/1343804/png/screen_shot_2012-04-19_at_105145_pm1334901100443.png`

The tertiary structure of a protein describes the folding of the polypeptide chain to form its final three-dimensional shape. Interactions between the side chains of the amino-acid residues hold this structure in place.

The quaternary structure of a protein is the combination of more than one polypeptide chain. For example, dimers are proteins comprised of two polypeptide chains. The quaternary structure is held together by the same interactions as the tertiary structure. Not all proteins have a quaternary structure. Those that do not have a quaternary structure, consist of one polypeptide chain and are known as monomers (Branden *et al.*, 1991).

1.3 Protein Structure Determination and Prediction

The three-dimensional structure of proteins is of great interest to biologists because the structure of a protein is related to its function. Proteins carry out vital functions in every cell and are arguably the most important biological molecule found in organisms.

Currently the most accurate way to determine protein structure is using experimental techniques such as Nuclear Magnetic Resonance (NMR) and X-ray Crystallography. However, these techniques are expensive and take time. As a result, the proteins that are chosen to have their structures determined may be subject to selection bias. Many computer-based prediction methods have been developed to predict protein structure.

1.3.1 Template Based Modelling

Template based, or homology, modelling is currently the most accurate method for predicting protein structure. It is is a non-experimental method that attempts to predict the structure of a protein sequence by finding closely related sequences which have high sequence similarity, that already have their structures experimentally determined. A suitable structure is then chosen as a template. The motivation behind this approach is the fact that the structure of closely related proteins is highly conserved to conserve their function. Sequences that are very similar are likely to have similar structures. However, the converse is not always true.

The accuracy of the predicted structure depends on the quality of the sequence comparison and the quality of the alignment between the target sequence and the template structure. If the sequence identity is greater than 40%, then approximately 90% of the predicted backbone atoms usually have an RMSD of around 1Å. When the sequence identity is between 30 and 40%, 80% of the predicted backbone atoms tend to have an increased RMSD of around 1.5Å. If the sequence identity is less than 30%, then approximately 20% of the residues may be misaligned, and the predicted backbone atoms usually have an RMSD greater than 3Å (Fiser, 2010).

Fold Recognition

Fold recognition, or threading, is used to predict the structure of proteins that do not have a homologous template available. To do this, the target sequence is compared to structural templates to find a compatible fold. Fold recognition is motivated by the theory that there are a limited number of distinct protein folds, around 2000, and thus many sequences with low similarity still have the same fold (Fiser, 2010).

1.3.2 *De novo* Prediction Methods

De novo, or *ab initio*, protein structure prediction describes the process of predicting the tertiary three-dimensional structure of a protein from its amino acid sequence, or primary structure. Scientists have been trying to solve this problem for decades. However, despite advancements, the problem still remains unsolved. As of 2016, there were 120 057 structures in the Protein Data Bank (PDB) (Berman *et al.*, 2000). This is relatively small compared to the vast amount of sequence data. Correspondingly, from 2016 there were 64 028 668 sequences in the UniProtKB database (Boutet *et al.*, 2016). Experimental methods for determining protein structure are too slow and expensive to close the massive gap between the number of sequences with determined structures.

Correlated Mutation Analysis

The structure of a protein is constrained by its function. Conservation and mutation patterns contained within protein family multiple sequence alignments provide evidence of functional and structural constraints (Göbel *et al.*, 1994). If a mutation occurs within a protein structure, the residue that it is in contact with in three-dimensional space may need to make a compensatory change to preserve the structure, and thus function, of the protein. We can see where this coevolution has

occurred by looking for correlated columns in multiple sequence alignments of protein families. The theory of using evolutionary information to predict residues that are close in three-dimensional space but far apart in sequence was first introduced by Göbel *et al.* (1994). Many methods have been developed to predict residue-residue contacts since, however until recently, predicted contacts consisted of >80% false positive results (Monastyrskyy *et al.*, 2014).

This year documented a significant step forwards for residue-residue contact prediction methods. The structure of a large protein consisting of 256 residues was accurately predicted as a result of improved residue-residue contact prediction (Moult *et al.*, 2016). In the latest round of CASP experiments, one method in particular outperformed all others, including methods in previous CASP experiments, in the residue-residue contact prediction category (Monastyrskyy *et al.*, 2015). The method, CONSIP2, reported an average precision of 27% (ratio of true predicted contacts out of all predicted contacts). CONSIP2 implements the MetaPSICOV method (Jones *et al.*, 2015); a new co-variation technique. The CONSIP2 server uses the target sequence and HHblits (Remmert *et al.*, 2012) to identify homologous sequences and construct an accurate multiple sequence alignment. If less than 2000 homologous sequences are identified, a combination of HHblits and jackHMMer (Finn *et al.*, 2011) is used. The resulting multiple sequence alignment is taken as input by MetaPSICOV (Kosciolek & Jones, 2015). MetaPSICOV combines a classical neural network-based contact prediction method with three different coevolution methods; PSICOV (Jones *et al.*, 2012), CCMpred (Seemayer *et al.*, 2014) and DCA (Marks *et al.*, 2011). The different coevolution methods each predict significant sets of contacts that exhibit minimal overlap. The three methods were selected as each one attempts to solve the statistical decoupling problem in a different way. MetaPSICOV is a two stage neural network predictor; an initial contact map is generated by the first stage network using the three coevolution based contact prediction methods, mutual information measures and classical machine learning-based contact prediction features. These features include amino acid profiles, predicted secondary structures and solvent accessibility, and sequence separation prediction. The second stage network removes outliers and fills in gaps in the contact map supplied by the first stage (Kosciolek & Jones, 2015).

By combining machine learning-based contact prediction and coevolution-based prediction, MetaPSICOV is able to successfully handle a range of alignments. If an alignment is sparse or poor quality, MetaPSICOV downweights coevolution and promotes generic structural features. Conversely, if sufficient homologous sequences exist, coevolution is upweighted (Jones *et al.*, 2015). Monastyrskyy *et al.* (2015)

shows that combining these methods results in a hybrid method that outperforms all others in the residue-residue contact prediction category.

1.4 Biological Background: Phylogenetics in Ecology

1.4.1 Phylogenetic Trees

Different organisms often contain similar DNA sequences. Evolutionary theory suggests that this may be because a common ancestor experienced mutational processes, such as substitution, insertion or deletion events. Thus, any set of species is related, and this relationship is called their phylogeny (Durbin, 1998). This can be represented in a diagram known as a phylogenetic tree. Phylogenetic trees are constructed using protein and nucleic acid sequence alignments (Hall, 2004) and consist of branches and nodes, as shown in Figure 1.4. Edges represent branches. A branch is a line connecting two nodes. Each branch of a phylogenetic tree represents an amount of evolutionary divergence. This defines the length of the branches and is typically a measure of distance between sequences or from a model of substitution of residues over the course of evolution (Durbin, 1998). Vertices represent nodes. A phylogenetic tree has two types of nodes; internal and external. External nodes are the tips, or leaf nodes of the tree, whereas internal nodes are the points representing a common ancestor of two or more other nodes (Hall, 2004). The distance between each pair of external nodes can be calculated by summing the lengths of all the branches between them. This distance is defined to be the patristic distance (Fourment & Gibbs, 2006). Patristic distances describe the amount of genetic change that has occurred in a tree.

Gene duplication and gene speciation are both mechanisms by which two sequences can separate and diverge from a common ancestor. The mechanism of gene duplication results in the problem that the phylogenetic tree of a group of sequences does not always reflect the phylogenetic tree of the species they belong to. Orthologues are genes which diverged because of speciation while paralogues are genes which diverged because of gene duplication (Durbin, 1998). We are interested in the phylogeny of orthologues.

Figure 1.4 also shows the difference between rooted and unrooted trees. Figure 1.4a displays a rooted tree, whereas Figure 1.4b displays an unrooted tree.

(a) (b)

Figure 1.4: Simple example of a rooted and unrooted phylogenetic tree.
(a) Rooted tree where R indicates the root. (b) Unrooted tree.

Often a rooted tree represents the phylogenetic history of species better than an unrooted tree. This is because usually the direction of evolutionary time is known, whereas with unrooted trees the direction of time is undetermined. However, unrooted trees give better correlations between species (Perretto & Lopes, 2005). A true biological phylogeny has a root, that is, all of the sequences have an ultimate ancestor (Durbin, 1998).

All trees in this thesis are rooted and binary or bifurcating. Every branch of the tree splits into two daughter branches and thus every internal node is connected to exactly three branches. Non binary, or multifurcating trees can in fact be approximated by binary trees by simply making some of the branches very short (Durbin, 1998).

The number of nodes and branches of a rooted tree can easily be counted. If a tree has n external nodes, then as we move backwards through evolutionary time the branches merge when another node is reached. Therefore, every time a node is reached, the number of branches decreases by one, thus there are $n-1$ internal nodes. Adding up the internal and external nodes gives $2n-1$ nodes in total. Since there is one fewer branch every time is a node is reached, it follows that there are $2n-2$ branches in total. An unrooted tree with n external nodes has one node fewer than a rooted tree, as it does not have the root node. Thus in total an unrooted tree has $2n-2$ nodes, and therefore $2n-3$ branches in total (Durbin, 1998).

1.4.2 Phylogenetic Networks

Jin *et al.* (2007) define phylogenetic networks to be a special set of directed acyclic graphs that are used when phylogenetic trees are not suitable. The direction of each edge is from the root node to the leaf nodes. Phylogenetic networks can represent interspecies relationships where the usual phylogenetic tree structure cannot.

Reticulate evolution can only be appropriately represented by networks; this includes evolutionary mechanisms such as hybrid speciation, horizontal gene transfer between taxa and genetic recombination (Makarenkov *et al.*, 2006). These processes are represented by a type of phylogenetic network known as a reticulate network, as shown in Figure 1.5. The main differences between reticulate networks and phylogenetic trees is that they contain hybrid nodes. Regular nodes in phylogenetic trees have one parent node. Hybrid nodes have two parent nodes, thereby allowing cross-connections between branches.

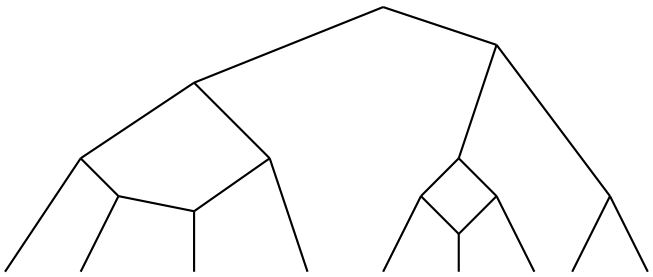


Figure 1.5: A simple example of a phylogenetic network.

1.4.3 Trophisms

Food webs diagrammatically represent the flow of energy through an ecosystem. This involves the feeding relationships and nutrient and energy pathways within an ecosystem (Rau *et al.*, 1983). A simple food web is displayed in Figure 1.6.

Food webs consist of many trophic levels, that is, different feeding levels. The trophic position of an organism is defined by the number of feeding links separating it from the base of production. The base of production has a trophic position of zero and contains the primary producers, such as photosynthesising plants. The next trophic position is occupied by herbivores, that have a trophic position of one and the higher trophic positions contain consumers (Thompson *et al.*, 2007). In Chapters 4 and 5 we look at how species belonging to different trophic layers in a system coevolve.

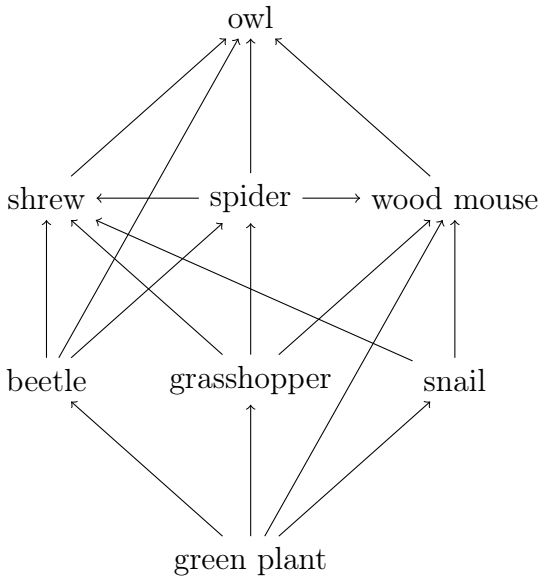


Figure 1.6: A simple example of a food web.

1.5 Thesis Overview

Throughout this thesis we explore the theory of coevolution. In Chapters 2 and 3 we research how coevolution conserves the structure of protein families. In Chapters 4 and 5 we investigate how interacting species coevolve in ecological systems.

We begin with a preliminary analysis of a large protein family, trypsin, in Chapter 2. We initially focus on a small seed sample of structures. The structures were aligned in pairs to determine how the structure of the trypsin family has evolved. By calculating the difference between the pairwise structurally aligned distance matrices we aim to gain insight into the location of conserved areas of structure, and where the main differences in structure may be. Following this, we extend our analysis to a larger sample of 83 structures, and structurally align the entire sample. We align the sample using a structural alignment as opposed to a sequence alignment because the structure of a protein family evolves more gradually than the amino-acid sequence and is therefore more conserved. We aim to identify where conserved regions appear in the structures, and where the regions of main difference are, to gain insight into how the family has evolved. We calculate summary statistics on the resulting aligned distance matrices and discover a set of residues where the distance between these specific residues and every other in the structure is highly conserved across all of the structures in the protein family. These residues appear to hold the structure of the trypsin protein family in place like anchors. We aim to determine

the validity and origin of these 'anchor' residues and the resulting conclusions drawn about the trypsin protein family following their discovery.

In Chapter 3 we explore coevolution between columns of multiple sequence alignments. The structure of a protein is constrained by its function. Sequence alignments from homologous proteins that are from a range of species provide information on these evolutionary constraints. Identifying correlated mutations within multiple sequence alignments can be used to predict residues that are in close proximity in three-dimensional space. We propose a regularised logistic regression model with the aim of successfully identifying these correlated mutations.

In Chapter 4 we investigate cospeciation in interacting systems of phylogenetic trees. We introduce a method to test efficiently for cospeciation in systems consisting of two or more phylogenetic trees. Tritrophic relationships have been observed in many ecological systems (Ahmad *et al.*, 2004; Forister & Feldman, 2011; Micha *et al.*, 2000; Nelson *et al.*, 2014). Mramba *et al.* (2013) developed the only statistical method we are aware of to test cospeciation in tritrophic systems, that does not simply compare the trees at a pairwise level. We propose a method that overcomes the limitations of Mramba *et al.*'s (2013) method; our method has the scope for generalisation to higher order systems and networks, and we do not place constraints on the interaction patterns between the phylogenetic trees in the system. We compare the performance of our method with existing methods at the bitrophic and tritrophic level. We aim to successfully test cospeciation hypotheses in tritrophic datasets and demonstrate that our method outperforms the leading existing methods.

We conduct a series of tests to assess the performance of our method in Chapter 4. To carry out these tests we simulated interacting systems of phylogenies by independently generating random trees, and separately, randomly assigning interactions between the trees. Simulating the systems in this way does not take into account their joint evolution. Chapter 5 introduces a method to simulate bitrophic and tritrophic systems under different evolutionary scenarios. Starting from one interacting species per lineage, their joint evolution is simulated by sampling the times at which evolutionary events occur from an exponential distribution. We focus on three evolutionary events; speciation, gaining an ecological interaction and losing an interaction. These events are controlled by a set of parameters. Experimenting with the intensity of these parameters produces a range of systems with different coevolutionary properties. We aim to simulate systems that extend the full range of the bitrophic and tritrophic cospeciation hypotheses that we test in Chapter 4.

To conclude, in Chapter 6 we summarise our research and findings, and detail the potential for further work.

Chapter 2

Do protein structures evolve around 'anchor' residues?

2.1 Introduction

The exploratory data analysis reported in this chapter focusses on the structural residue-residue distances of the trypsin protein family. By comparing distances across the whole protein family we aim to gain insight into how the structure of the family has evolved. Trypsin is widely used in biotechnological and food industries, as well as for biological and medical research. As a result there are over 2000 trypsin structures that have been experimentally determined, belonging to a large variety of distantly related species. Marks *et al.* (2011) have successful predicted the structure of trypsin; therefore it is an ideal protein family for preliminary analysis.

We initially focus on a small sample of 8 structures taken from the Pfam (Bateman *et al.*, 2004) seed multiple sequence alignment. There are two types of alignment; sequence and structural. The seed structures were aligned in pairs using the TM-align pairwise structural alignment algorithm (Zhang & Skolnick, 2005) to identify regions of similarity throughout evolution. Calculating the difference between the pairwise structurally aligned distance matrices provided an indication of the location of conserved areas of structure, and where the main differences in structure may be. Following this, we extended our analysis to a larger sample of 83 structures.

The structures were aligned using a multiple structural alignment algorithm, MUSTANG (Konagurthu *et al.*, 2006), to determine how the structure of the family has evolved. Calculating basic summary statistics on the resulting aligned distance

matrices revealed an interesting result. We discovered a set of residues where the distance between these specific residues and every other in the structure is highly conserved across all of the structures in the protein family. These residues appear to hold the structure of the trypsin protein family in place like anchors.

We conduct a series of tests to determine the validity and origin of the intriguing concept of 'anchor' residues and the resulting conclusions drawn about the trypsin protein family following their discovery. However, many of these tests proved inconclusive or provided conflicting evidence. Therefore the question is still open; are the anchor residues artefacts?

2.1.1 Trypsin

Before selecting the trypsin protein family, a range of possible families were considered (more details can be found in Appendix A). We required a family with a large number of sequences, and a reasonable number of structures experimentally determined, from a range of species.

There is an overwhelming amount of structural data in the Protein Data Bank (PDB) (Berman *et al.*, 2000), of varying quality. We only considered protein families that satisfy the following criteria:

- Determined using X-ray Crystallography as opposed to NMR. Both techniques tend to produce structures with the same fold, however they often result in different surface-loop regions and side-chain rotational states (Yang *et al.*, 2007). X-ray crystallography has the ability to produce structures with more precise atomic resolution detail than NMR spectroscopy, which is also limited to smaller proteins and protein domains (Krishnan & Rupp, 2012).

- Resolution less than 2.3Å, where $1Å = 10^{-10}$ meters. The resolution of a protein structure measures the quality of the crystal containing the protein. If all of the proteins within the crystal are highly ordered, a greater resolution can be obtained. That is, if the protein atoms are in defined positions throughout the crystal and over time. If the proteins in the crystal are slightly different, due to local flexibility or motion, then less detail can be obtained in the diffraction pattern. The higher the resolution, the greater the detail that can be observed, and less of the atomic structure will need to be inferred (Berman *et al.*, 2000).

- Structures with a genetically manipulated source are excluded because unlike natural source organisms the proteins are not isolated from the organism, but from another genetic origin.

Trypsin is a protein of the serine protease family involved in the digestive processes of most vertebrates. It is produced in the pancreas and breaks proteins down into smaller proteins to be absorbed through the lining of the small intestine. Trypsin has many applications; it is used in many biotechnological processes, the food industry, biological research, as a treatment for inflammation and in microbial form to dissolve blood clots. Due to its multiple varied uses, over 2000 trypsin structures have been experimentally determined over a wide variety of species. A typical trypsin structure is displayed in Figure 2.1, displayed using the molecular visualisation software Jmol (Herraez, 2006). Trypsin is in the all-beta class of proteins because it consists almost entirely of beta sheets, with the exception of two small isolated alpha helices on the peripheral of the structure. Trypsin contains two beta barrels that lie perpendicular to each other in the structure. The beta barrels are a closed structure formed when the beta sheets twist such that the first strand is hydrogen bonded to the last.

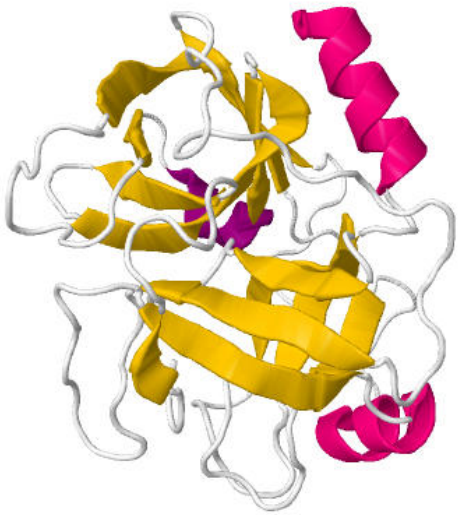


Figure 2.1: Ribbon representation of a trypsin molecule (Protein Data Bank (PDB) accession code: 1S5S) displayed with the molecular visualisation software, Jmol. The secondary structures are coloured; pink indicates an alpha helix, yellow indicates a beta sheet and the purple helix is a 3_{10} helix; a helix with 3 residues per turn rather than 3.6.

2.2 Exploratory Data Analysis

2.2.1 Trypsin Seed Sample

A large number of structures have been experimentally determined for the trypsin protein family; therefore, we initially focus our analysis on the seed multiple sequence alignment structures from the Pfam database of protein families (Bateman *et al.*, 2004). A seed alignment contains a small set of representative structures of a protein family, whose alignment has been manually verified (Sonnhammer *et al.*, 1998). From this sample, we selected one protein structure for each UniProt ID in the seed multiple sequence alignment, resulting in 8 structures in total. Information about these structures is displayed in Table 2.1.

2.2.2 Pairwise Structural Alignment

The protein chains were aligned in order to identify regions of similarity. There are two types of alignment; sequence and structural. Sequence alignments are constructed based on the similarity between amino-acid residues and their physiochemical properties, while structural alignments use shape and three-dimensional conformation to align the atomic coordinates of the structures. Structure alignments are of interest because the structure of a protein family evolves more gradually than the amino-acid sequence and is therefore more conserved. It is particularly useful when the sequence similarity between proteins is low. To analyse where the differences and similarities are between the seed structures, we use a pairwise structural

		Chain ID				
PDB ID	UniProt ID	Domain	Other	Domain Range	Resolution	Species
1MKX	THRB_BOVIN	H,K	L	16-238	2.20Å	Cow
1UTM	TRY1_SALSA	A		16-238	1.50Å	Salmon
2P3U	FA10_HUMAN	B	A	16-238	1.62Å	Human
3Q76	ELNE_HUMAN	A,B		16-238	1.86Å	Human
3ODF	CELA1_PIG	A		16-238	1.10Å	Pig
1QTF	ETB_STAAU	A		28-223	2.40Å	Bacteria
1PQ7	TRYP_FUSOX	A		16-235	1.23Å	Fungi
1S5S	TRYP_PIG	A		16-238	1.40Å	Pig

Table 2.1: UniProt and PDB database identifiers for the seed sample of trypsin structures. Information about the species, the number of chains in the PDB file, and the chain and residue location of the trypsin domain is also provided.

alignment algorithm, the TM-align server (Zhang & Skolnick, 2005). We use these alignments to explore the difference between the aligned distances.

The method of TM-align only takes the C_α atom coordinates of the two protein structures as input. First the secondary structures are initially aligned using dynamic programming. A score matrix is constructed where elements take the value 1 or 0 depending on whether or not the secondary structure elements of aligned residues are the same. A second initial alignment is obtained by threading the smaller of the two proteins against the larger structure, without introducing any gaps. A score matrix is generated using the TM-score rather than RMSD. A third initial alignment is constructed by dynamic programming with a gap-opening penalty of -1, and using the score matrices from the first initial alignment of secondary structures and the second initial alignment, giving each score matrix equal weight. The weighting accelerates the convergence of the dynamic programming. It also corrects for the effects that result from alignment lengths.

A heuristic algorithm takes the initial alignments as input. The structures are rotated based on the alignment in the first initial alignment. The TM-score rotation matrix is used (Zhang & Skolnick, 2004). The score similarity matrix is defined as

$$S(i,j) = \frac{1}{1 + d_{ij}^2/d_0(L_{\min})^2},$$

where d_{ij} is the distance between the i^{th} residue of the first structure and the j^{th} residue of the second structure; $d_0(L_{\min}) = 1.24\sqrt[3]{L_{\min} - 15} - 1.8$ where $L_{\min}$ is the length of the shorter of the two structures. An intermediate alignment is generated by applying dynamic programming to the matrix $S(i,j)$. The intermediate is used to superimpose the structures by the TM-score rotation matrix. An improved alignment is obtained by dynamic programming on the new score matrix. This heuristic procedure is repeated and the alignment with the highest TM-score is returned.

2.2.3 Difference Distance Matrix Analysis

The three-dimensional shape of a protein can be summarised by its residue-residue distances. A distance matrix for a protein structure, k, contains the Euclidean distance, $d_{i,j}^{(k)}$, between the C_α atoms of each amino-acid residue pair, i and j. The alignment produced by TM-align provides us with a set of superposed three-dimensional coordinates for each structure. These superposed coordinates imply a corresponding sequence alignment. We can use this structure-based sequence alignment to

structurally align or superimpose the distance matrices of the aligned structures. This allows us to analyse corresponding distances across the structures. To identify the differences between the corresponding distances of two structures, a difference distance matrix, D^{diff}, is calculated. The $(i,j)^{\text{th}}$ element of D^{diff} is given by

$$(D^{\text{diff}})_{i,j} = d^{(1)}_{i,j} - d^{(2)}_{i,j}, \tag{2.1}$$

where $d^{(1)}_{i,j}$ and $d^{(2)}_{i,j}$ are

1UTM and 1S5S

Structure 1UTM is from species *Salmo salar* (common name: atlantic salmon) while 1S5S is from species *Sus scrofa* (common name: pig). Despite being very different organisms the heat maps generated for the two structures are very similar, almost identical by eye. The difference distance matrix for these structures is plotted as a heat map in Figure 2.2. The heat map is interpreted slightly differently to a typical distance matrix for a structure. The colour scale now indicates how similar or different the distances between the residues of the two aligned structures are; large differences are represented by red and blue, while small differences are given in white. The black regions correspond to gaps in the sequence alignment obtained by aligning the structures.

The heat map confirms that the two structures are very similar as it is largely a pink colour, indicating that the differences are close to zero. Both structures consist of one chain, with the trypsin domain found in the residue range 16-238. Therefore, it may be expected that their structures are very similar. The superimposed structures are displayed in Figure 2.3.

It can clearly be seen that the two structures overlap very closely. However, there are a few small deviations in the loop regions. Closer inspection of these regions revealed that 1UTM has a β-bridge between residues 94 and 95, whereas 1S5S does not. Their sequences at these positions differ by one amino acid; residue 94 is tyrosine in 1UTM and phenylalanine in 1S5S. Tyrosine and phenylalanine are very similar amino acids. Their side chains are similar shapes and sizes and both are hydrophobic and aromatic. Therefore this substitution would be expected to be of little consequence to the structure. 1UTM also has a β-bridge inbetween residues 99-100 where 1S5S does not. Again, the difference is due to one residue; 99. Where 1UTM has isoleucine, 1S5S has leucine. These residues are very similar, and thus the impact on structure is minor.

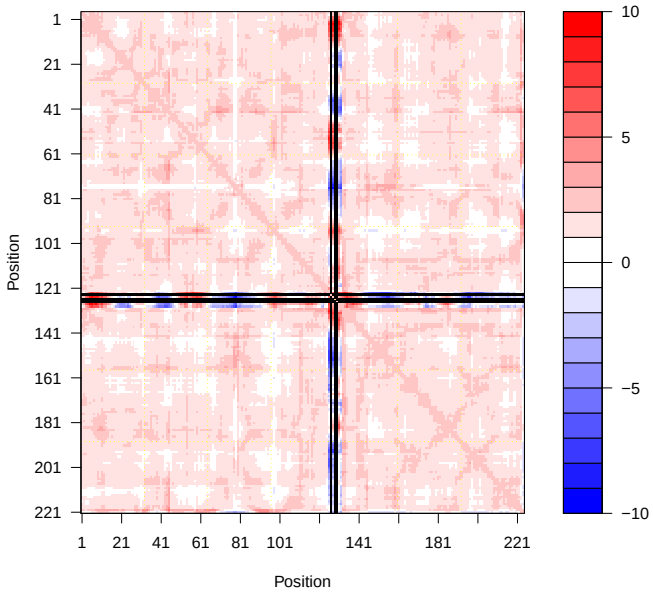


Figure 2.2: Heat map of the difference distance matrix for aligned structures 1S5S and 1UTM. The difference between the residue-residue distances are plotted in red-white-blue colour scale; if there is no difference between the distances they are plotted white, red and blue indicate large differences between the distances. The black regions indicate where gaps occur in the structure alignment.

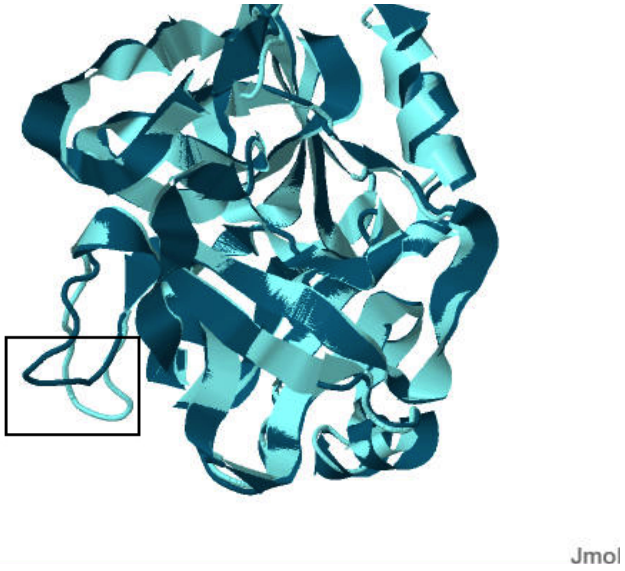


Figure 2.3: PDB file produced by TM-align for the pairwise structural alignment of 1UTM (dark blue) and 1S5S (light blue), displayed using Jmol. The black box highlights a deviation in a loop region of the structural alignment between residues 145 and 150.

The main deviation in the two structures is between residues 145 and 150, a loop region. This region is highlighted by a black box in Figure 2.3. The sequence of aligned amino acids in this region are

MSSTAD

KSSGSS

for 1UTM and 1S5S respectively.

There are many differences in the aligned residues in this region. Methionine and lysine are both hydrophobic amino acids. Threonine and glycine are both small amino acids, however glycine is smaller and threonine is polar. Alanine and serine are similar sized amino acids, however, serine is polar and alanine is aromatic. Aspartic acid and serine are both polar, however serine is much smaller than aspartic acid, which is negatively charged. These differences result in the slightly different loop regions observed.

The final deviation in structure similarity is in the final helix length. The α-helix for 1S5S is 2 residues longer than that of 1UTM.

These relatively minor deviations in the loop regions are not unexpected given that these structures are from very different organisms, but clearly a large amount of the more organised structural regions have been retained.

1S5S and 3ODF

Structures 1S5S and 3ODF are both from species *Sus scrofa* (common name: pig). Both structures consist of a single chain and for both the trypsin domain lies in the residue range 16-238. However, their heat maps are two of the most distinct. The difference distance matrix for the two structures is plotted as a heat map in Figure 2.4. The heat map shows that the distances between the aligned residues are relatively similar for the two structures, however many gaps have been introduced. This is because 3ODF is longer than 1S5S. In order to see how this extra length in 3ODF is accommodated, Figure 2.5 displays the superimposed structures.

The secondary structures and the internal structure of the proteins are closely aligned in structure and thus strongly conserved between these two species. When looking at the structure alignment from different angles it is clear that the additional length of 3ODF is accounted for in the loop regions on the outside of the structure.

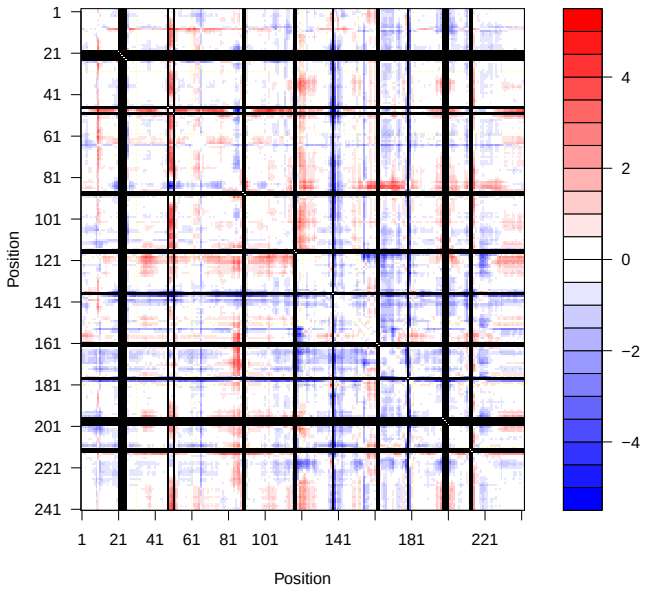


Figure 2.4: Heat map of the difference distance matrix for aligned structures 1S5S and 3ODF. The difference between the residue-residue distances are plotted in red-white-blue colour scale; if there is no difference between the distances they are plotted white, red and blue indicate large differences between the distances. The black regions indicate where gaps occur in the structure alignment.

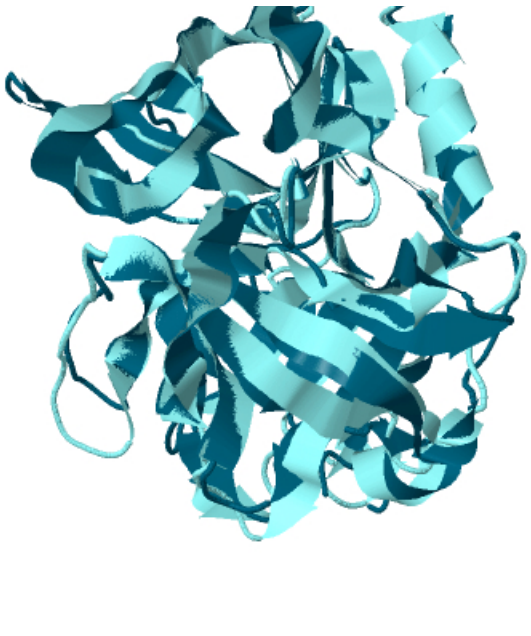


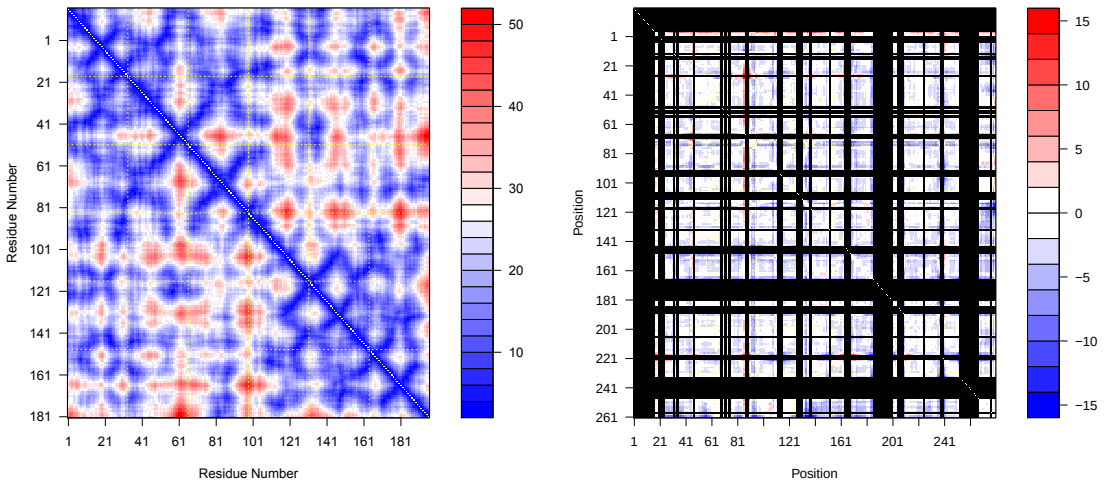
Figure 2.5: PDB file produced by TM-align for the pairwise structural alignment of 3ODF (dark blue) and 1S5S (light blue), displayed using Jmol.

1S5S and 1QTF

All of the seed structures exhibit very similar heat maps. However, the heat maps generated for 1QTF, displayed in Figure 2.6a, are the most distinct. This not surprising as 1QTF is from species *Staphylococcus aureu* (common name: bacteria) and therefore the most evolutionarily diverged from the other species.

Similarly to 1S5S, 1QTF also has only one chain and they are both of similar lengths, meaning 1S5S is ideal to align structurally to determine how 1QTF differs from the structures previously analysed.

The difference distance matrix for the two structures is plotted as a heat map in Figure 2.6b. The heat map shows that the alignment contains many gaps. The superimposed structures are displayed in Figure 2.7. Unsurprisingly, the structures are not as closely aligned as those in Figures 2.3 and 2.5. Visual analysis of the aligned structures in Jmol reveals that the main departures in the alignment are on the outside of the protein, while the secondary structure elements and the hydrophobic core of the protein are conserved.



(a) (b)

Figure 2.6: (a) Heat map of the original distance matrix of structure 1QTF. The distances between residues are plotted in red-white-blue colour scale; small distances are blue and large distances are red. (b) Heat map of the difference distance matrix for aligned structures 1S5S and 1QTF. The difference between the residue-residue distances are plotted in red-white-blue colour scale; if there is no difference between the distances they are plotted white, red and blue indicate large differences between the distances. The black regions indicate where gaps occur in the structure alignment.

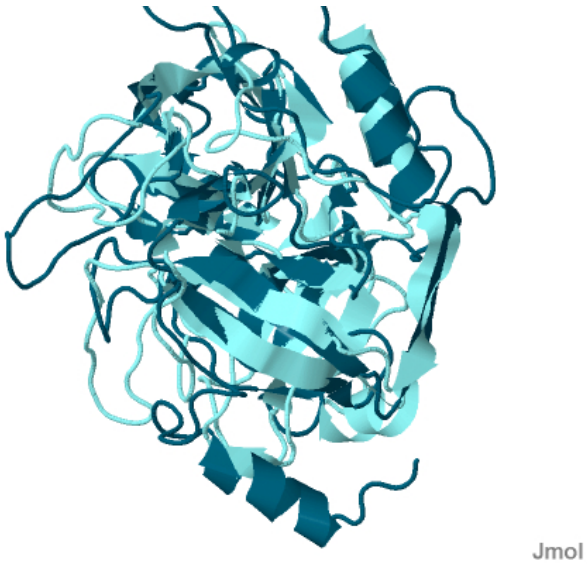


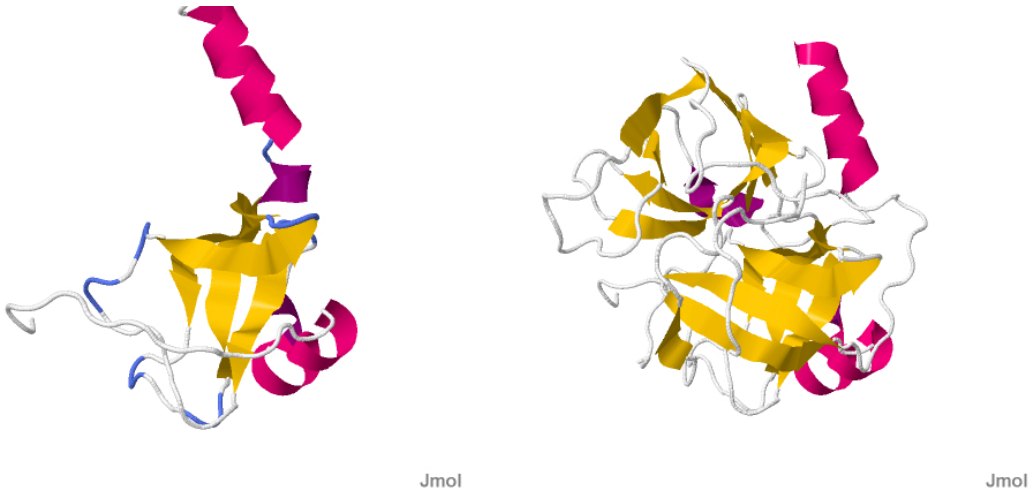
Figure 2.7: PDB file produced by TM-align for the pairwise structural alignment of 1QTF (dark blue) and 1S5S (light blue), displayed using Jmol.

2.2.4 Trypsin Extended Sample

We extended the sample of trypsin structures to conduct our analysis on a larger structural alignment. The structures were chosen according to the criteria in Section 2.1.1. Due to the size of the sample, the MUSTANG multiple structural alignment algorithm (Konagurthu *et al.*, 2006) was used as it is one of the few structural alignment algorithms capable of operating with a large number of structures.

The sequence alignment produced as a result of aligning the structures displayed large gapped regions. These gaps were a result of only two sequences; 1MKX:K and 1QTF. Therefore, these structures were removed and the remaining 89 structures realigned to get the best possible alignment without removing too much of the data.

During the course of the analysis it was found that one of the PDB files, 1JRT:A, appeared to have an unusual entry. Between residues 183 and 184, a residue 983 had been inputted. Due to the unknown cause of this possible error, 1JRT:A was removed to avoid confusion. It was also found that some of the PDB files identify multiple chains which appear to constitute different elements of a single trypsin chain when compared to the other trypsin structures, as illustrated with 2P8O in Figure 2.8. There are 5 chains with this property; 2P8O, 2VGC, 1UHB, 1YM0 and 2QA9. Two of these structures, 2P8O and 2VGC, have an additional chain preceding the trypsin chains suggesting that they are the zymogen trypsinogen. Trypsinogen is the inactive enzyme precursor to trypsin; the additional chain is cleaved in order to produce the active enzyme trypsin.



(a) (b)

Figure 2.8: An example of a "broken" PDB file; PDB ID: 2P80. The single structure is stored in three separate PDB files; 2P80:A, 2P80:B, and 2P80:C. (a) Ribbon representation of the partial structure 2P80:C. (b) Reconstructed full PDB file for 2P80, consisting of the three partial structures.

MUSTANG failed to align these structures once the "broken" chains were altered to produce a single PDB file containing a complete trypsin chain. Therefore these structures were removed from the sample leaving a total of 83 structures in the final sample. See Appendix B for a comprehensive list. Once aligned, the data were prepared for analysis by removing all positions in the alignment where more than 20% of the entries consist of gaps.

2.2.5 Multiple Structure Alignment

An overview of the MUSTANG procedure is given in Figure 2.9. The main steps in the procedure are as follows. The MUSTANG method first tries to find structural similarity in pairwise fragments of structures before building the multiple structure alignment. Each pair of structures are initially scored using root-mean-square deviation (RMSD) in order to find similar substructures. The RMSD is a measure of the average distance between the atoms of superimposed structures. The individual residue alignments are then scored using a similarity measure that is closely based on the elastic similarity function proposed by Holm & Sander (1993). These scores are used to align each pair of structures by a dynamic programming algorithm. The pairwise alignment scores are then recalculated in the context of all of the structures. This is achieved by taking every structure as an intermediate for each pairwise alignment. The more intermediate structures that support the alignment

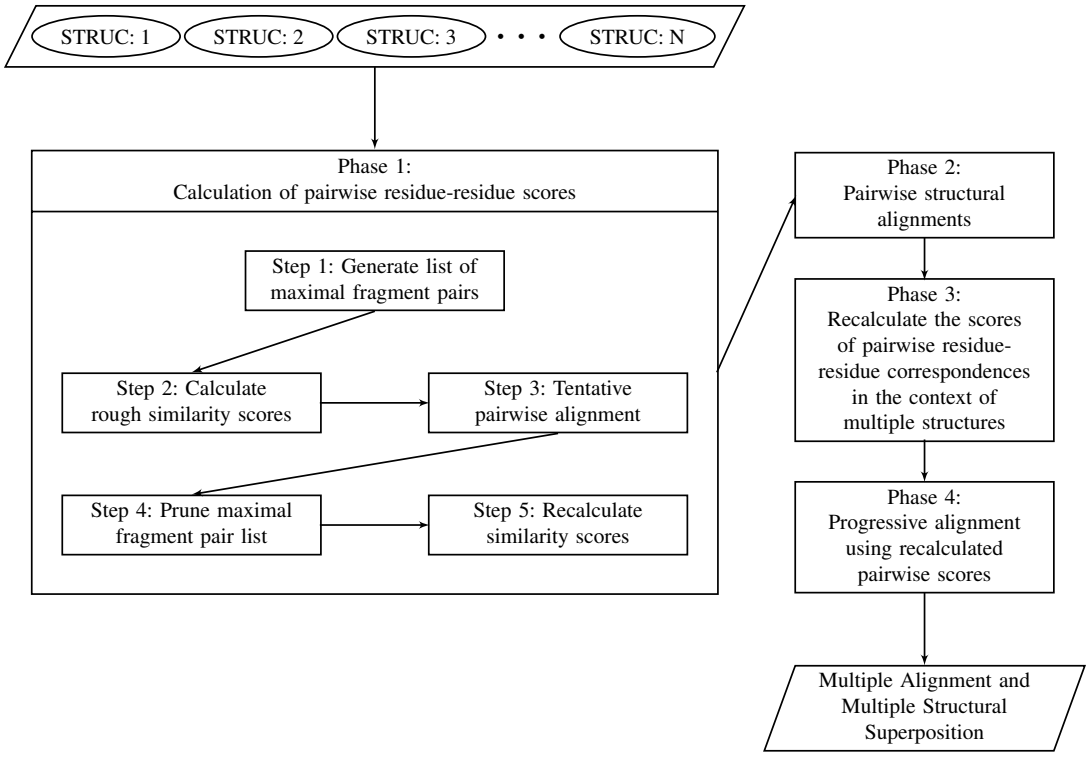


Figure 2.9: An overview of the MUSTANG algorithm (Konagurthu *et al.*, 2006).

of a pair of residues, the higher the score assigned to them. The multiple structure alignment is finally obtained following a binary guide tree constructed using the neighbour-joining method (Saitou & Nei, 1987) applied to the similarity scores.

The method in more detail is as follows:

Phase 1: Step 1: For every pair of structures to be aligned, the three-dimensional coordinates of the C_α atoms of the residues are taken in blocks of at least 6 consecutive residues and superposed using the method of Kearsley (1989). If the RMSD is less than or equal to 1.75 then the superposed fragments are extended until the RMSD is no longer less than 1.75. The fragments are extended by adding successive positions to the C-termini. The fragments are truncated so that they do not end within secondary structures in order to avoid mismatches in terminal regions. This collection of extended superposed fragments are defined to be 'maximal fragment pairs' or MFPs. The values 6 and 1.75 were empirically determined by Konagurthu *et al.* (2006) to give the best results.

Phase 1: Step 2: Rough similarity scores are calculated for residue-residue correspondences between every pair of structures. The scores do not take into account the arrangement of the MFPs in their respective structures; they are simply used to speed up Phase 1: Step 5.

Phase 1: Step 3: Dynamic programming is used to create a tentative pairwise alignment for each pair of structures.

Phase 1: Step 4: The number of MFPs is reduced by ignoring all MFPs that are outside ±30 positions from any of the correspondences produced by the pairwise alignment.

Phase 1: Step 5: Using the reduced set of MFPs the similarity scores from Phase 1: Step 2 are recalculated and this new score is added to the previous score.

Phase 2: Dynamic programming is again used to align each pair of structures using the new scores.

Phase 3: The residue-residue correspondence scores are recalculated again in the context of multiple structures. Each pair of structures is scored taking every other structure as an intermediate in the alignment; the more intermediate structures supporting an alignment of a pair of residues, the higher the increments to it's score.

Phase 4: A distance divergence between each pair of aligned residues is calculated by transforming their normalised alignment score. The normalised alignment score is calculated using the final residue-residue correspondence scores. The distance divergences are then used to construct a guide tree, which in turn is used to progressively align the structures.

Berbalk *et al.* (2009) and Konagurthu *et al.* (2006) compare MUSTANG with other multiple structure alignment algorithms; POSA, CE-MC, MALECON and MultiProt. According to Konagurthu *et al.* (2006), MUSTANG performs as well as the other alignment tools for closely related proteins and outperforms them for more distantly related proteins or proteins that exhibit conformational changes. Berbalk *et al.* (2009) supports the conclusion that MUSTANG performs as well as other alignment tools when the structures have high structural similarity but suggests that there is room for improvement when structures are more distantly related.

When searching for a multiple structure alignment tool, MUSTANG proved to be the easiest to use in terms of data upload and output. MUSTANG also has

no limit on the number of structures it can align whereas other algorithms do, for example CE-MC can align only up to 10 structures and POSA can align only up to 20 structures. The program STRAP was also considered, however it deleted residues with the same residue number, a common occurrence in many PDB files to represent where known insertions occur between aligned residues. However, MUSTANG also has several disadvantages; it can be very temperamental in what can be aligned and as a result wastes a lot of time and is sometimes unreliable. MUSTANG also only uses the information in the C_α coordinates of the structures and the distances between them. The information contained in the amino-acid sequence is ignored completely.

The output of the process is a multiple-sequence alignment constructed using the structural alignment of the chains. We prepared the alignment for subsequent analysis by removing all positions in the alignment where more than 20% of the entries consist of gaps. (Gaps are introduced in alignments where insertions or deletions are predicted to have occurred throughout evolution.) For smaller samples MUSTANG produces a PDB file containing the coordinates for the superimposed structures; this can be visualised using Jmol (Herraez, 2006). Visual analysis is impractical with such a vast number of structures; instead, we considered the distances between the residues in the superimposed structures.

2.2.6 Aligned Distance Matrix Analysis

The positions in the distance matrices can be aligned, or superimposed, using the MUSTANG alignment to analyse corresponding distances across the structures. The alignment produced by MUSTANG respects the sequence order of the amino acids.

There are 219 alignment positions in the MUSTANG alignment of the 83 trypsin structures downloaded from the PDB, resulting in a $219 \times 219 \times 83$ data array. This large data structure can be summarised by calculating a measure of location and divergence for every distance across the aligned structures. We achieved this by calculating a weighted median and a weighted interquartile range, where the weights are calculated using the method of Henikoff & Henikoff (1994) as follows:

- For each position in the alignment, divide a total weight of 1.0 evenly between the unique letter types in that position.
- Divide the weight that has been assigned to each letter type between the number of that letter type in that position.

- For each sequence, sum the weights that have been assigned at each position.
- Normalise the sequence weights to sum to 1.0.

Example

Consider the following partial sequence alignment

ACQKMIVG
RCQLMLVQ
ACQI PLVE
ACEKPLVG

Table 2.2 displays how the Henikoff weights are calculated for each sequence by first weighting the letter types occurring in each position as outlined above. The final weight for each sequence is given in the final column of the table.

Sequences from the same species are likely to be very similar, whereas sequences from more diverged species differ more. If all of the sequences are weighted equally then information may be lost when there are many similar sequences due to independent information from the more diverged sequences being diluted. The sequences are weighted so that very similar sequences are down-weighted and unusual sequences are up-weighted. We constructed a median matrix, $\tilde{D}$, and divergence matrix, D^{div}, using the aligned distance matrices; the $(i,j)^{\text{th}}$ element of each of these matrices is

Position																Total	Normalised
1		2		3		4		5		6		7		8			Weights
$\frac{1}{6}$	A	$\frac{1}{4}$	C	$\frac{1}{6}$	Q	$\frac{1}{6}$	K	$\frac{1}{4}$	M	$\frac{1}{2}$	I	$\frac{1}{4}$	V	$\frac{1}{6}$	G	$\frac{23}{12}$	0.2396
$\frac{1}{2}$	R	$\frac{1}{4}$	C	$\frac{1}{6}$	Q	$\frac{1}{3}$	L	$\frac{1}{4}$	M	$\frac{1}{6}$	L	$\frac{1}{4}$	V	$\frac{1}{3}$	Q	$\frac{9}{4}$	0.2813
$\frac{1}{6}$	A	$\frac{1}{4}$	C	$\frac{1}{6}$	Q	$\frac{1}{3}$	I	$\frac{1}{4}$	P	$\frac{1}{6}$	L	$\frac{1}{4}$	V	$\frac{1}{3}$	E	$\frac{23}{12}$	0.2396
$\frac{1}{6}$	A	$\frac{1}{4}$	C	$\frac{1}{2}$	E	$\frac{1}{6}$	K	$\frac{1}{4}$	P	$\frac{1}{6}$	L	$\frac{1}{4}$	V	$\frac{1}{6}$	G	$\frac{23}{12}$	0.2396

Table 2.2: Calculation of Henikoff weights for a simple sequence alignment.

given by

$$\tilde{d}_{i,j} = Q_2(d_{i,j}^{(1)}, \ldots, d_{i,j}^{(83)}),$$
$$d_{i,j}^{\text{div}} = Q_1(d_{i,j}^{(1)}, \ldots, d_{i,j}^{(83)}) - Q_3(d_{i,j}^{(1)}, \ldots, d_{i,j}^{(83)}),$$

where Q_p is the p^{th} weighted percentile. Recall, $d_{i,j}^{(k)}$ represents the distance between alignment positions i and j for structure k, where $i, j = 1, \ldots, 219$ and $k = 1, \ldots, 83$.

To assess the relationship between the median and divergence matrices they are plotted against each other in Figure 2.10. There are a vast number of data points as a result of the size of the matrices; $\frac{219^2}{2} = 47\,961$ data points, however there does not appear to be an obvious relationship between the divergence and the median. Intuitively it might be assumed that a larger median would correspond to a larger divergence, since the distance between the residues is larger. However only a handful of points exhibit this property, suggesting that for the majority of the sample the overall framework of the structures is very similar. Interestingly, there are a collection of points where the divergence is high while the median is very low. This pattern corresponds to the scenario where the distances between the two residues are small, yet there is a lot of variation in the corresponding distances

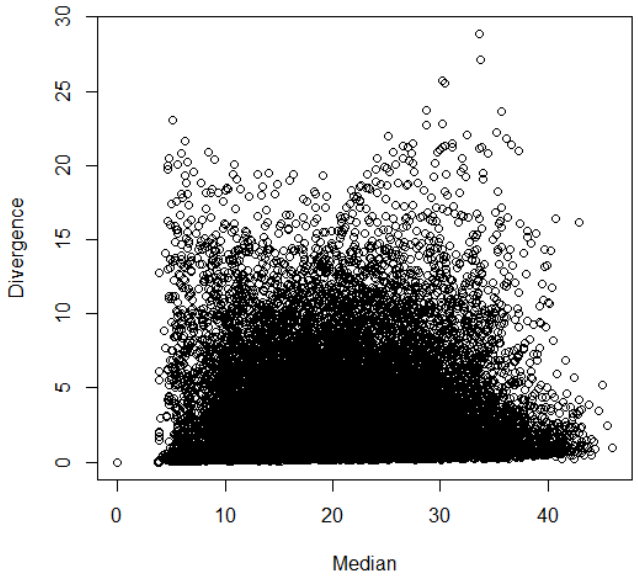
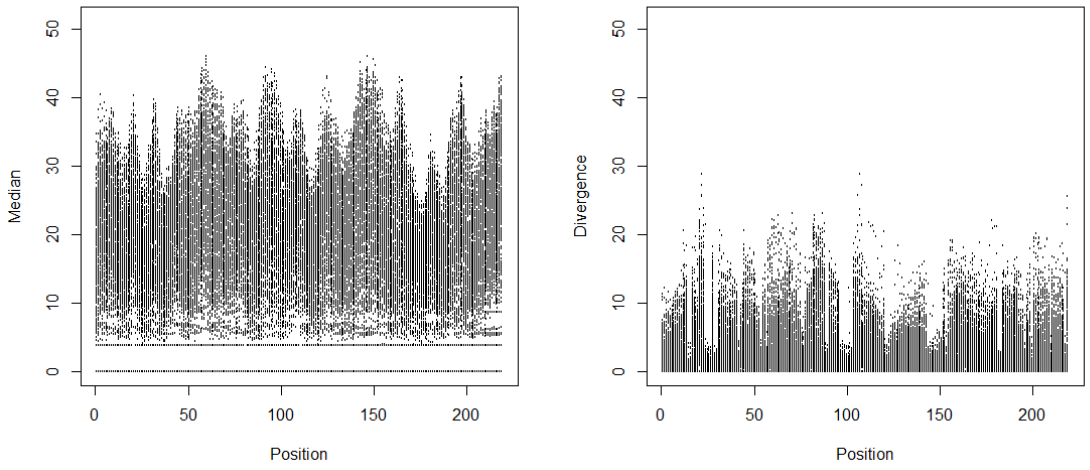


Figure 2.10: Plot of median aligned residue-residue distance against the divergence between the distances for each pair of residues, for the MUSTANG structural alignment of the trypsin sample.



(a) (b)

Figure 2.11: Plots of the rows of the median and divergence matrices calculated from structurally aligned distance matrices of the trypsin sample. The bars appear as a result of many points plotted close together. (a) Median, $\tilde{d}_{i,j}$, of the structurally aligned distances plotted against position, i, in the alignment. There are many columns, j, plotted for each i. (b) Divergence, $d^{\text{div}}_{i,j}$, of the structurally aligned distances plotted against position, i, in the alignment. There are many columns, j, plotted for each i.

across the structures, suggesting a different local structure for some of the samples.

Each row (and column) of the median and divergence matrices corresponds to a position in the structural alignment. This is plotted in Figure 2.11. The bars appear as a result of many points plotted close together. The plot of divergence against position in Figure 2.11b shows that there are positions in the alignment where the range of divergences is low as indicated by distinct troughs between the peaks. This suggests that there are residues, or short subsequences of residues, where the distance between that residue and every other residue in the structure is conserved, across all of the structures. If this result is genuine, these residues could be used to predict the structure of proteins in the trypsin family, and might also provide a basis for predicting structure from multiple sequence alignments of other protein families.

2.2.7 Median Distance Matrix Analysis

The median matrix is plotted as a heat map in Figure 2.12. The heat map is interpreted identically to a typical heat map for a structure; small distances are

represented by blue while large distances are given in red. As a result the heat map is not dissimilar to a typical distance-matrix heat map produced by any of the structures. This is unsurprising given that the median matrix is an average of the aligned distance matrices. This suggests that MUSTANG has produced a reasonable structure alignment and the median distance matrix is a suitable measure to be used to construct a consensus structure to represent the sample, that is, the average structure of the sample.

Multidimensional scaling is a technique used to construct a configuration of data points in Euclidean space using the distances, similarities or dissimilarities between them. The data points are assigned coordinates in n dimensions that aim to preserve the distances between them (see Appendix C for more details). Metric multidimensional scaling can be applied to the median distance matrix in order to obtain a consensus structure. We could also perform multidimensional scaling on the divergence matrix which would allow us to see where the differences from the median structure are.

The R (R Core Team, 2013) function `cmdscale` was used to perform metric multidimensional scaling on the median distance matrix. There are three eigenvalues that are much larger than the remaining eigenvalues. These normalised squared eigenval-

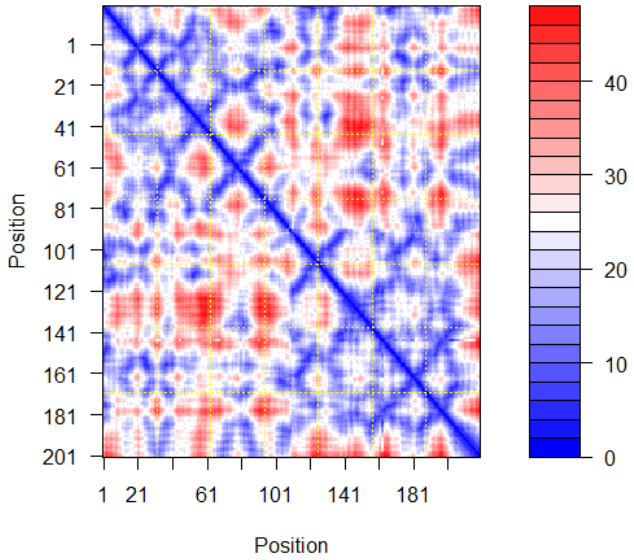


Figure 2.12: Median matrix heat map. The median residue-residue distances are plotted in red-white-blue colour scale; small distances are blue and large distances are red.

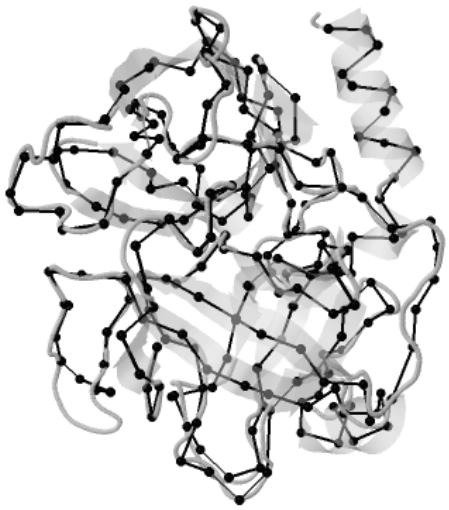


Figure 2.13: Multidimensional scaling structure of the median distance matrix, displayed in black. The C_α atoms of each position in the alignment are given by a black circle. C_α atoms corresponding to adjacent alignment positions are connected by black lines to represent the backbone of the median structure. The trypsin structure in Figure 2.1 is superimposed with the consensus structure and displayed in grey. The structures were superimposed using TM-align pairwise structural alignment algorithm (Zhang & Skolnick, 2005).

ues are 0.61, 0.28, 0.10, while the remaining values are close to zero, suggesting that the first three coordinates are sufficient to reproduce the median distance matrix. This is unsurprising given that we know that the distances are obtained from three dimensional objects. The resulting coordinates are used to produce a PDB file which can be viewed in Jmol. The consensus structure is displayed superimposed over the trypsin structure 1S5S in Figure 2.13.

The consensus structure is comprised only of C_α atoms since the distance matrices used to construct it contain the distances between the C_α atoms of each residue. Despite this, Figure 2.13 shows that the configuration produced using multidimensional scaling is a good approximation of the trypsin structure in Figure 2.1.

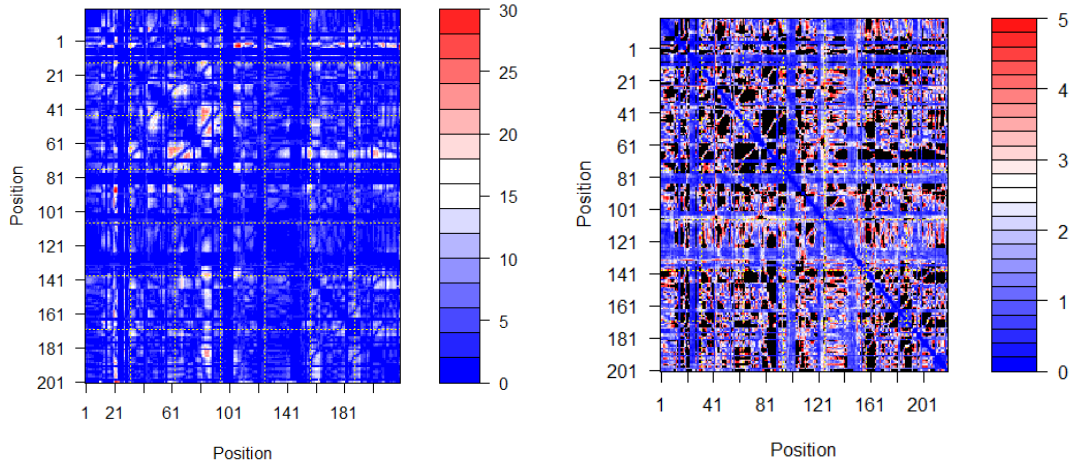
2.2.8 Divergence Distance Matrix Analysis

The divergence matrix is plotted as a heat map in Figure 2.14a. In this case red indicates large divergences implying distances that are less conserved while blue regions represent small divergences or distances that are more conserved. The scale

in Figure 2.14a is inflated by a small area of high divergence. The low-range divergences identified in Figure 2.11 are approximately 5 ångströms (5Å); to analyse alignment positions at this end of the scale, all divergences greater than 5Å are coloured black and the heat map recalculated based on the scale 0 to 5, as displayed in Figure 2.14b.

The pattern of divergence at the lower end of the scale can now be visualised more clearly. There is a clear pattern of horizontal and vertical blue lines running across the heat map. These lines represent where in every structure the distance between one residue and every other residue is highly conserved, in agreement with the conclusions drawn from Figure 2.11. Four distinct groups of alignment positions can be identified as having a low range of divergences. These residues are of interest as they appear to be anchors for each of the structures; conserving their distances and holding them in place.

To accurately determine the positions in the multiple structure alignment corresponding to the low-range divergences, the maximum divergence in each position is plotted in Figure 2.15. The red line indicates a cutoff at 7Å as there is a natural



(a) (b)

Figure 2.14: Divergence matrix heat maps for different colour scales. The divergence between the residue-residue distances are plotted in red-white-blue colour scale; small divergences are blue and large divergences are red. (a) Divergence matrix heat map based on the original scale. The information in blue is diluted by a small amount of red that is pulling up the scale. (b) Divergence matrix heat map recalculated for all of the divergences that are less than 5Å; larger divergences are blacked out.

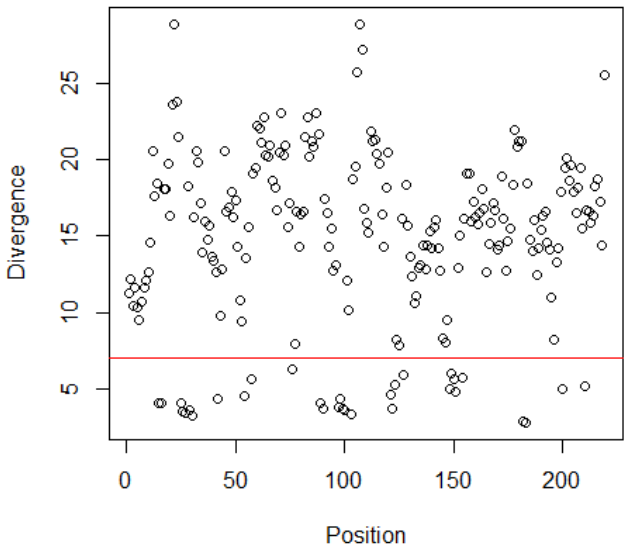
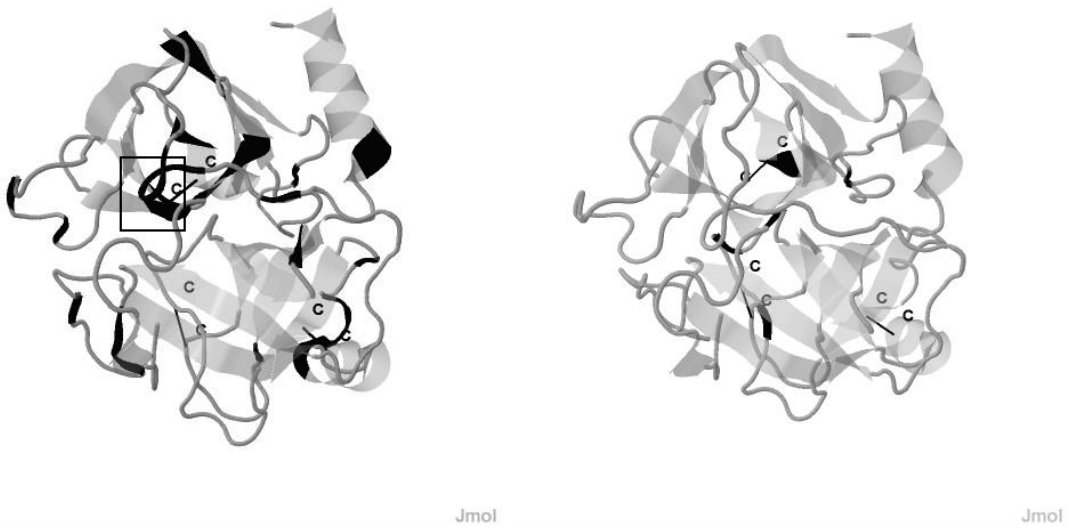


Figure 2.15: Maximum divergence between the distances in each alignment position of the trypsin sample. The dashed line indicates a cutoff of 7Å where there is a natural divide in the maximum divergences.

divide between the maximum divergences at around this threshold. It remains to identify which positions have a maximum divergence of less than 7Å and determine where these lie on each of the structures. We define an anchor residue to be any residue, i, with $\max_j d_{i,j}^{\text{div}} < 7$Å. Figure 2.16a displays the structure of a representative sample structure (PDB identifier 1JIR), in a grey ribbon representation with the anchor residues identified in blocks of black. Consecutive anchor residues are coloured the same, resulting in longer bands of black where anchor residues lie next to each other in sequence. In fact 70 of the structures (84%) in the sample exhibit identical colourings to 1JIR.

The anchor residues are predominantly located on the outside of the protein and in loop regions. One of the beta barrels is the only region that appears to be completely devoid of colour. The beta sheets found on the section of the beta barrels that faces into the centre of the structure form the hydrophobic core that is important in attracting the specific residues that trypsin cleaves.

Protein structure is closely related to its function. The enzymatic mechanism of trypsin involves a catalytic triad of residues: the amino acids histidine-57, aspartic acid-102 and serine-195, where the numbers after the hyphen indicate the sequence position. These three residues form a charge relay that causes the active site serine residue to become nucleophilic by modifying its electrostatic environment (Bate-



(a) (b)

Figure 2.16: (a) Ribbon representation of a trypsin structure (PDB ID: 1JIR) identifying the location of the anchor residues, displayed in blocks of black, and the three disulphide bonds, indicated by black lines and labelled cysteine (C) residues. The black box indicates the cysteine (C) residue that is also an anchor residue. (b) The same structure identifying the location of functional residues; including the catalytic triad of residues and the oxyanion hole, displayed in blocks of black, and the three disulphide bonds, indicated by black lines and labelled cysteine (C) residues.

man *et al.*, 2004). Trypsin also contains an 'oxyanion hole' formed by the backbone amide hydrogen atoms of glycine-193 and serine-195. This hole stabilises the developing negative charge on the carboxyl oxygen atom of the cleaved amides. Another important functional residue is aspartic acid-189 located in the catalytic pocket of trypsin. This residue is responsible for attracting and stabilising positively charged lysine and arginine residues (Bateman *et al.*, 2004).

To determine whether these functional residues coincide with the anchor residues, Figure 2.16b displays the location of the functional residues, coloured in black. The functional residues are generally in the centre of the protein, in contrast to the location of the anchor residues. It can easily be seen that the functional residues and anchor residues do not overlap, that is, none of the anchor residues correspond to a functional residue.

Trypsin has a number of disulphide bonds stabilising its structure. Stroud (1974) claims that trypsin has 6 disulphide bonds, however only 36 of the structures have the required number of cysteine residues; 12. According to Várallyay *et al.* (1997) there are 3 conserved disulphide bonds; C42-C58, C168-C182 and C191-C220. It was

found that 80 of the 83 structures have enough cysteine residues to form at least 3 disulphide bonds. Figure 2.16 indicates by black lines connecting the ribbons in the structure where these three disulphide bonds are found in relation to the anchor residues and functional residues. The bonds appear to be positioned around the substrate-binding pocket; this is unsurprising given that this is the part of the structure vital to the protein's function. Only one of the bonds involves an anchor residue as indicated by the black box in Figure 2.16a.

It is important to check that the positions in the structural alignment that correspond to the anchor residues are not predominantly comprised of gaps. If most of the sequences correspond to gaps in the anchor positions, then the structural conservation in these positions would be the result of a small number of structures in the sample. The percentage of gaps in the anchor columns compared to the other columns in the alignment are represented using boxplots, as displayed in Figure 2.17.

The median percentage of gaps in the anchor columns is 12.05 compared to a median percentage of gaps of 4.22 in the other columns in the alignment. However, because there are fewer anchor columns, the percentage of gaps in the anchor columns is much less variable, with a standard deviation of 1.89 compared to a standard deviation of 37.09 for the percentage of gaps in the other columns. Overall, the anchor columns of the alignment are not excessively gapped compared to the other

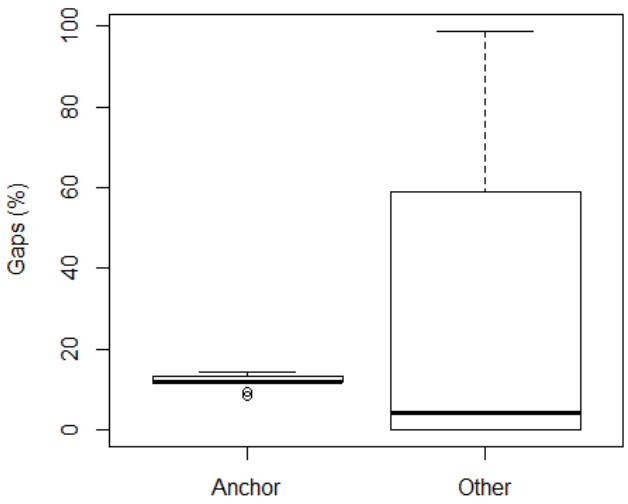


Figure 2.17: Boxplots comparing the percentage of gaps in structural alignment positions corresponding to anchor residues, with the percentage of gaps in the other positions in the alignment.

to the other columns in the alignment; however, the median number of gaps in the anchor columns is larger than that of the other alignment columns. Given that the anchor columns are not disproportionately gapped it remains to determine which residue types are found in each anchor position and how conserved these residues are. Table 2.3 contains the percentage of each residue type in each of the anchor columns. Some of the anchor columns appear to be conserved in sequence; however, overall they do not appear to be more conserved than every other column in the alignment.

Rypniewski *et al.* (1994) propose several conserved residues, in both sequence and structure. Comparing the anchor residues in Table 2.3 to those proposed by Rypniewski *et al.* (1994) results in an overlap for some of the residues; there are 7 anchor columns that correspond to the conserved residues identified in that paper. The residues 42, 43 and 44 correspond to anchor columns 3, 4 and 5 in Table 2.3, respectively. These three residues are strongly conserved in the aligned sequences and they are identified as conserved by Rypniewski *et al.* (1994). These three residues are found close to the active site; glycine-43 forms a hydrogen bond with the carbonyl oxygen of serine-195, one of the catalytic triad residues and cysteine-42 forms a disulphide bond, as displayed in Figure 2.16. Anchor column 11 corresponds to residue 94 which lies in the exposed side of the loop that contains the active site residue aspartic acid-102 and is important in maintaining structure; its side chain is in contact with two residues of the catalytic triad; aspartic acid-102 and histidine-57. In the paper, residue 94 is tyrosine; however, in Table 2.3 the corresponding column shows that the residue is tyrosine in only 39% of the structures. This could be due to the fact that the amino acid at residue 91 that forms a hydrogen bond with residue 94 is variable, and thus residue 94 varies to accommodate this. Conserved residues 171 and 172 are important in the specificity function of trypsin. In particular, residue 172 forms a hydrogen bond with a residue at the bottom of the specificity pocket. These residues correspond to anchor columns 23 and 24. Rypniewski *et al.* (1994) identify residue 172 as tyrosine, but also state that it is substituted in many sequences, explaining why it is not very conserved in Table 2.3. The final residue that is identified as conserved by Rypniewski *et al.* (1994) and is also an anchor residue is residue 225, or anchor column 30. This residue is a conserved proline residue in Table 2.3, and its role is linked to residues 171 and 172. A number of the anchor columns are found next to the residues identified as conserved by Rypniewski *et al.* (1994). Overall, this identifies that some of the anchor columns correspond to known conserved residues, suggesting that MUSTANG has managed to align some of the key conserved residues well.

Table 2.3: Conservation in percentage of each amino-acid residue and gaps in the positions of the structural alignment corresponding to the anchor residues. The most conserved amino acids in each position is given in bold.

Amino Acid	Anchor Column																													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
L	1					**63**	23	24					**86**					1	18		1										23
Q	**84**									**41**												18				6			5		
A		23	2		1										1			23					29		6			18	1	
I		18				23	**54**						2				2	**29**		1					1				18	
V		**45**					8	18		**43**				2			**83**	14			1								**34**	
C			**86**																				6							
G				**86**	**87**													20					1		18				5	
S				2		2								2	**53**				1		**39**	1	**37**	23	23	18				1
T							2				1	2			8					**83**	1	1	1	1		**72**				1
D								1							1						8	8								
N								1	**83**							23			**43**			1							8	
Y								**42**			**39**					28	2						18	**43**			**41**	1	24	
R									2	1		**41**		23		**36**			23	1	20	23		6	1					
P										2															**43**				**61**		
F											28														23			23		
K											1	**45**			1				1		**33**					18		**29**		1
W											18																1			
M															23					1	8									
H																1					23									
E																										1		10		
–	14	14	12	12	12	12	12	13	12	12	13	12	12	12	12	12	12	12	13	13	13	8	8	8	8	8	10	14	14	10	13

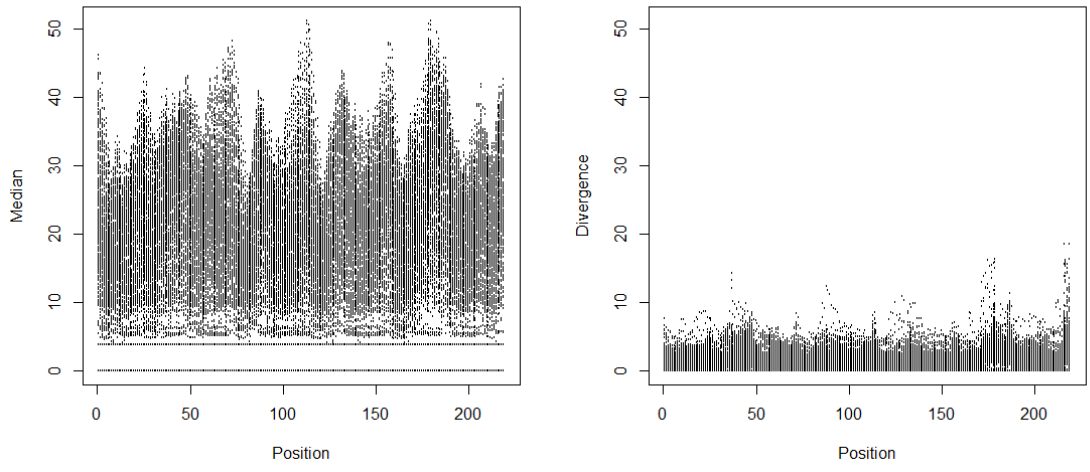
2.3 Are the Anchor Residues Artefacts?

The anchor residues identified by analysing the structure alignment produced by MUSTANG are intriguing. It is necessary to test that these residues are not simply an artefact produced by MUSTANG. There is no common standard for assessing the quality of a structural alignment (Liu *et al.*, 2011), therefore we propose the following tests.

2.3.1 Aligning another protein family

One way to identify whether the anchor residues are an artefact of MUSTANG is to align another protein family and determine whether low-range divergences are apparent. If MUSTANG is reliable and the anchor residues are truly a feature of protein evolution, we expect the anchor residues to be present.

A search of Pfam (Bateman *et al.*, 2004) produced a suitable family from a diverse range of species; short-chain dehydrogenase. A sample of 49 structures were aligned and divergence and median matrices calculated for the aligned distance matrices. Figure 2.18 displays the divergences and medians in each position of the alignment. The plot of divergences in Figure 2.18b does not exhibit the distinct



(a) (b)

Figure 2.18: Plots of the rows of the median and divergence matrices calculated from structurally aligned distance matrices of the short-chain dehydrogenase sample. The bars appear as a result of many points plotted close together. (a) Median, $\tilde{d}_{i,j}$, of the structurally aligned distances plotted against position, i, in the alignment. (b) Divergence, $d_{i,j}^{\text{div}}$, of the structurally aligned distances plotted against position, i, in the alignment.

troughs that were seen for trypsin; however, the majority of the divergences are low at less than 5Å. The distances between the residues in the structures of this protein family are more similar than those in the trypsin family, suggesting that the short-chain dehydrogenase family of proteins is more highly conserved in structure than the trypsin protein family. Therefore, aligning short-chain dehydrogenase does not conclusively determine whether MUSTANG introduces bias. However, it does cast doubt on the significance of the anchor residues, suggesting that they are merely well-aligned regions of the trypsin protein family.

2.3.2 Aligning an artificial sample of trypsin structures

The following method generates a sample of 83 artificial proteins consisting only of C_α atoms by resampling the C_α-atom coordinates of residues from one structure. We expect the anchor residues to be present in the artificial sample if they truly exist, since the structures are created from one structure which exhibits the anchor residue property.

The trypsin structure 1LVY was chosen for the resampling procedure because it is relatively long and exhibits the conserved anchor residue pattern displayed in Figure 2.16. The resampled structures are generated by uniformly selecting a number of residues to remove from 1LVY at random and then closing the resulting gaps in three-dimensional space. The gaps are closed using the following method:

When a gap is produced the adjacent residues are linearly translated such that the Euclidean distance between their C_α atoms is equal to the standard bond length between these atoms in a typical structure.

Consider the example structure displayed in Figure 2.19. The nodes represent the C_α atoms. Let $\mathbf{x_0}$ be the vector of (x, y, z) coordinates of the C_α atom to be removed.

Once the C_α atom corresponding to $\mathbf{x_0}$ is removed, the coordinates of the adjacent C_α atoms, $\mathbf{x_{-1}}$ and $\mathbf{x_1}$, are translated using the following equation

$$\begin{aligned} \mathbf{x'_{-1}} &= \mathbf{x_0} + \lambda_0(\mathbf{x_{-1}} - \mathbf{x_0}) \\ \mathbf{x'_1} &= \mathbf{x_0} + \lambda_0(\mathbf{x_1} - \mathbf{x_0}), \end{aligned} \tag{2.2}$$

where $\mathbf{x'_{-1}}$ and $\mathbf{x'_1}$ are the new coordinates of the adjacent C_α atoms and where $\lambda_0 \in [0, 1]$. When $\lambda_0 = 0$ the new coordinates are $\mathbf{x'_{-1}} = \mathbf{x_0}$ and $\mathbf{x'_1} = \mathbf{x_0}$. At the other extreme, when $\lambda_0 = 1$, the new coordinates are $\mathbf{x'_{-1}} = \mathbf{x_{-1}}$ and $\mathbf{x'_1} = \mathbf{x_1}$. We want to choose λ_0 such that $\mathbf{x'_{-1}}$ and $\mathbf{x'_1}$ lie between these extremes, specifically at

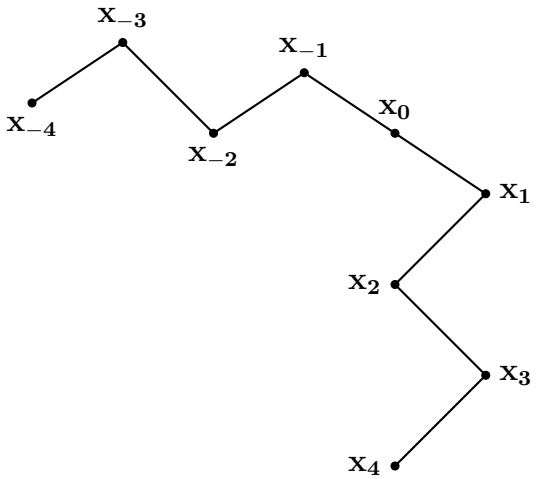


Figure 2.19: Example structure consisting only of C_α atoms, represented by dots; adjacent residues are connected by lines to form the backbone of the structure. The C_α atoms are labelled in accordance with the method for closing gaps in structure; $\mathbf{x_0}$ is a vector containing the (x, y, z)-coordinates of the residue that will be removed to form the gap, and $\mathbf{x_1}, \ldots, \mathbf{x_4}$ and $\mathbf{x_{-1}}, \ldots, \mathbf{x_{-4}}$ are the coordinates of the sequence of residues reading away from the gap on either side. See text for further explanation.

a distance of d_α apart, where d_α is defined to be the standard distance between C_α atoms:

$$d_\alpha^2 = \|\mathbf{x'_1} - \mathbf{x'_{-1}}\|^2. \tag{2.3}$$

The average C_α atom to C_α atom bond distance in the structure 1LVY is calculated to be 3.81Å; therefore d_α is taken to be 3.81Å.

Substituting $\mathbf{x'_{-1}}$ and $\mathbf{x'_1}$ from Equation (2.2) into Equation (2.3) and rearranging in terms of λ_0 gives

$$\lambda_0 = \frac{d_\alpha}{\|\mathbf{x_1} - \mathbf{x_{-1}}\|}.$$

Next we translate the residues adjacent to those either side of the gap. In this case, only one residue is moved in order to preserve the distance between the residues that were translated in the previous step; $\mathbf{x_{-2}}$ is translated to correct for the distance between $\mathbf{x_{-2}}$ and $\mathbf{x'_{-1}}$ as follows

$$\mathbf{x'_{-2}} = \mathbf{x'_{-1}} + \lambda_{-1}(\mathbf{x_{-2}} - \mathbf{x'_{-1}}), \tag{2.4}$$

where the scale constant λ_{-1} is calculated similarly to λ_0 and is thus given by

$$\lambda_{-1} = \frac{d_\alpha}{\|\mathbf{x_{-2}} - \mathbf{x'_{-1}}\|}$$

Equation (2.4) is applied successively to each C_α atom in the structure, substituting for the appropriate coordinates at each iteration.

The number of residues removed was calibrated such that the number of gaps produced by the alignment was close to the average number of gaps in the original sample alignment. The average number of gaps per row in the original alignment was 71.99 and an alignment with 66 gaps per row was produced for the resampled structures when 28 residues are removed at random from 1LVY and aligned using MUSTANG. Removing 30 residues produced too many gaps.

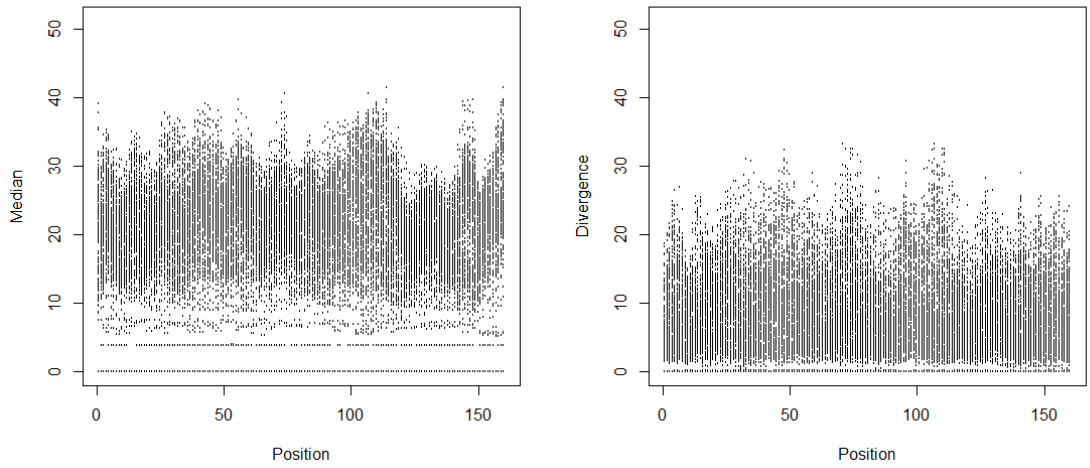
To complete the simulation of artificial proteins it is necessary to add random $N(0, s_\alpha^2)$ noise to the C_α atom coordinates, where s_α^2 is the variance of the C_α atom to C_α atom bond lengths. Noise is added since all of the resampled structures originate from the same structure and because not all C_α atom to C_α atom bond lengths are precisely 3.81Å.

The previous analysis is carried out on the aligned sample of artificial structures to produce the divergence and median plots displayed in Figure 2.20. The plot of divergence against position in Figure 2.20b shows that the range of divergences is very high, certainly none are below 5Å. There is no evidence to suggest the existence of anchor residues. It might be expected that the structures are very similar and would thus align well, producing low divergences; however, the range of divergences is high suggesting the distances are less conserved than in the trypsin sample. There is certainly no evidence of the previously observed anchor residues.

When compared to Figure 2.11a the plot of median against position in Figure 2.20a for the artificial structures does not exhibit similarities with the plot for the real trypsin sample. This difference in median distances suggests that the artificial structures have a different structure to the trypsin sample structures. This is not unusual since the artificial structures are all variations of one structure, 1LVY. However, it is necessary to understand the effect that the gap-closing method has on the shape of a structure, this is explored in Section 2.4.

Similarly to Figure 2.14b the divergence matrix can be displayed as a heat map for the artificial sample, given in Figure 2.21. In this case it is the high divergences that bring up the scale; as a result divergences greater than 10 have been coloured black. There is no longer the pattern of horizontal lines that could be observed in Figure 2.14b, confirming that there are no anchor residues. In fact there are very few areas on the off diagonal that have low divergences at all.

The number of gaps removed was also varied for each resampled structure in the sample; however, the same results were obtained concerning the low range divergences or anchor residues.



(a) (b)

Figure 2.20: Plots of the rows of the median and divergence matrices calculated from structurally aligned distance matrices of the artificial trypsin sample. The bars appear as a result of many points plotted close together. (a) Median, $\tilde{d}_{i,j}$, of the structurally aligned distances plotted against position, i, in the alignment. (b) Divergence, $d^{\text{div}}_{i,j}$, of the structurally aligned distances plotted against position, i, in the alignment.

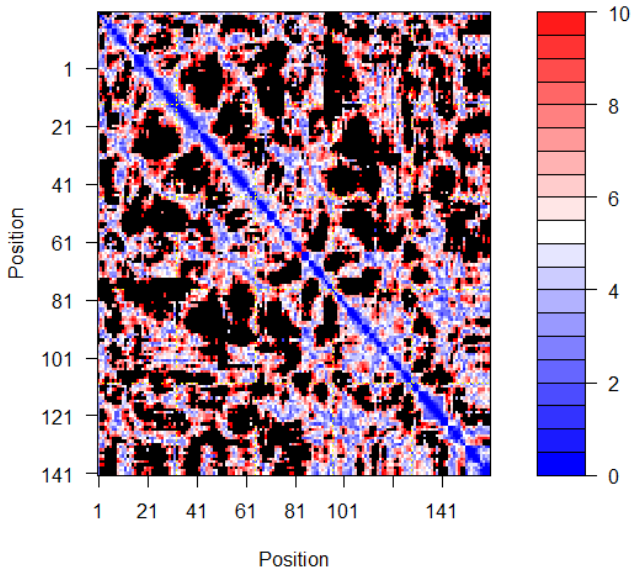


Figure 2.21: Divergence matrix heat map for the artificial trypsin sample, recalculated for all of the divergences that are less than 10Å. Larger divergences are blacked out. The divergence between the residue-residue distances are plotted in red-white-blue colour scale; small distances are blue and large distances are red.

There are a number of ways in which this methodology for producing artificial structures could be improved. In order to reflect true evolutionary processes, insertions and substitutions could be incorporated as well as deletions. The method could also be extended to include all of the atoms in the starting structure, not just the C_α atoms.

Therefore this method provides evidence against MUSTANG; we would expect anchor residues to be apparent in 1LVY if they truly exist. However, they are not apparent in the artificial structures suggesting that the phenomenon is an artefact of MUSTANG. It is also possible that the procedure for closing gaps in the artificial structures is destroying the anchor residue property. We explore the effect the gap-closing method has on the structures in Section 2.4.

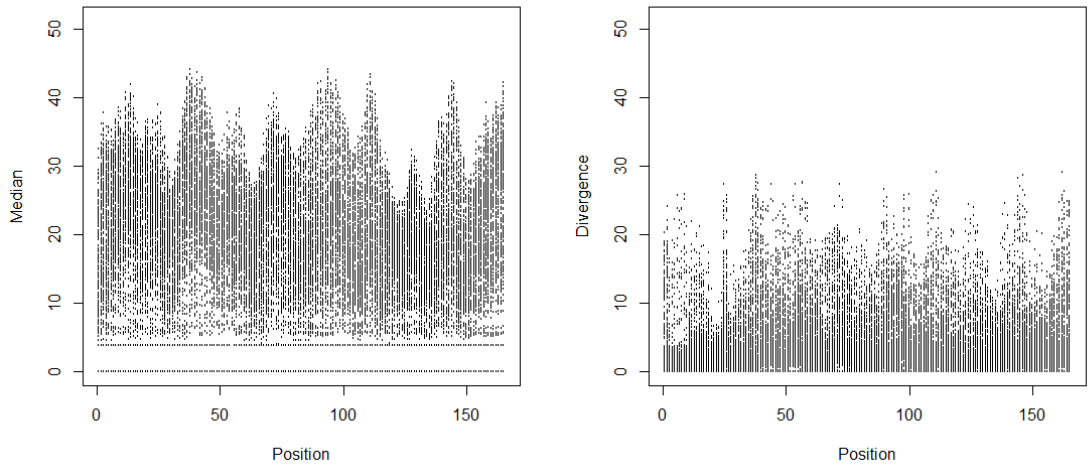
2.3.3 Aligning C_α atoms of the real trypsin sample

Since the method in the previous section uses only the C_α atom coordinates it is necessary to compare the structural alignment of the trypsin sample with the alignment produced when only the C_α atoms of their residues are structurally aligned. MUSTANG appears to use only the C_α atom coordinate of structures when producing an alignment. Therefore we expect the full atom trypsin alignment and the C_α only trypsin alignment to be similar.

In this case the plots of divergence and median against position displayed in Figure 2.22 are produced and compared to the full atom structural alignment of the trypsin sample in Figure 2.11. The distinct troughs in the divergences in Figure 2.11b are not apparent when only the C_α atoms of trypsin are aligned; however, there is a lower range of divergences compared to the artificial structures. There appears to be some correspondence between the peaks of the median distances in Figure 2.11 and Figure 2.22, suggesting the overall shape of the structures is not too different, and thus the two alignments are reasonably similar. However it also suggests that using only C_α atoms is not representative of the full sample.

To understand more about how the full-atom trypsin structural alignment and the corresponding C_α-atom-only structural alignment differ, their gaps are analysed. In this case, a gap is defined to be a consecutive run of insertions where the length of the gap is the number of insertions. Figure 2.23a compares the number of gaps in each sequence for the two alignments from MUSTANG. This information is also represented by boxplots in Figure 2.23b.

The median number of gaps in the C_α-atom alignment is much larger at 41.00, compared to a median number of gaps of 24.00 in the full-atom alignment. The



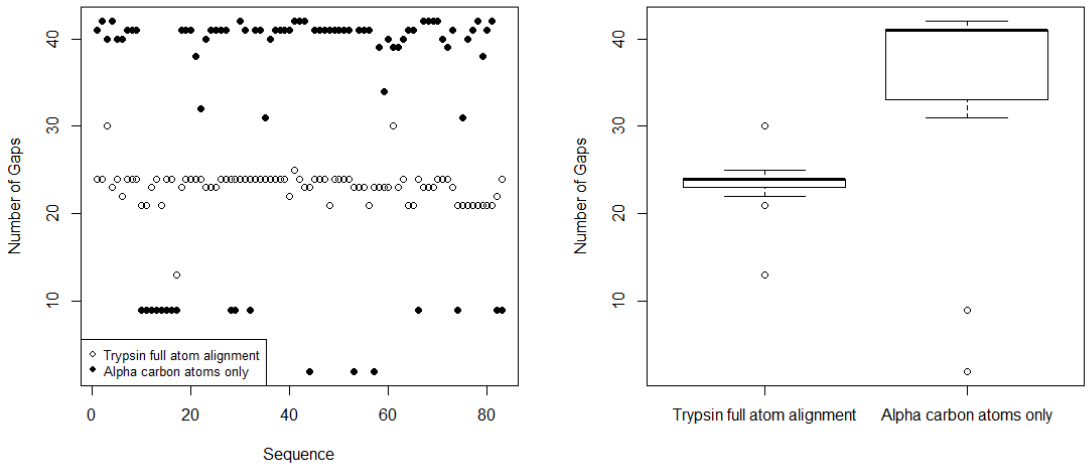
(a) (b)

Figure 2.22: Plots of the rows of the median and divergence matrices calculated from structurally aligned distance matrices of the trypsin sample with only C_α atoms. The bars appear as a result of many points plotted close together. (a) Median, $\tilde{d}_{i,j}$, of the structurally aligned distances plotted against position, i, in the alignment. (b) Divergence, $d^{\text{div}}_{i,j}$, of the structurally aligned distances plotted against position, i, in the alignment.

number of gaps is also much more variable in the C_α-atom alignment with a standard deviation of 13.66 compared to a standard deviation of 1.92 in the full-atom alignment. This can also be seen in Figure 2.23a where many of the points are higher in the C_α-atom alignment, but there are also a smaller number of points that are lower than the full atom case. However, are these gaps shorter than those in the original alignment? The lengths of the gaps for each alignment are displayed in Figure 2.24.

The median length of the gaps in the two alignments is the same at 2.00; however, the range of values is very different. The largest gap in the full-atom alignment is 21.00, compared to an incredibly long gap of 119.00 in the C_α-atom alignment. Unsurprisingly, the standard deviation for the length of the gaps in the C_α-atom alignment is larger at 7.80, compared to a standard deviation of 2.703 for the length of the gaps in the full-atom alignment. Therefore, not only does the C_α-atom alignment appear to have more gaps for most sequences, some of the gaps are also significantly longer compared to the original alignment.

Clearly the full-atom alignment and the C_α-atom alignment are quite different; therefore, the methods for testing bias may not be entirely representative of the full-atom case. This is an interesting result since MUSTANG aligns structures using only



(a) (b)

Figure 2.23: Comparison of the number of gaps in the trypsin structural alignment with those in the structural alignment obtained only for the C_α atoms of the trypsin sample. The number of gaps is defined to be the number of consecutive runs of insertions. (a) Plot of the number of gaps in each position for each alignment. Those corresponding to the full atom trypsin sample are white points. Those corresponding to the C_α atom trypsin sample are coloured in black. (b) Boxplots comparing the number of gaps in the two alignments.

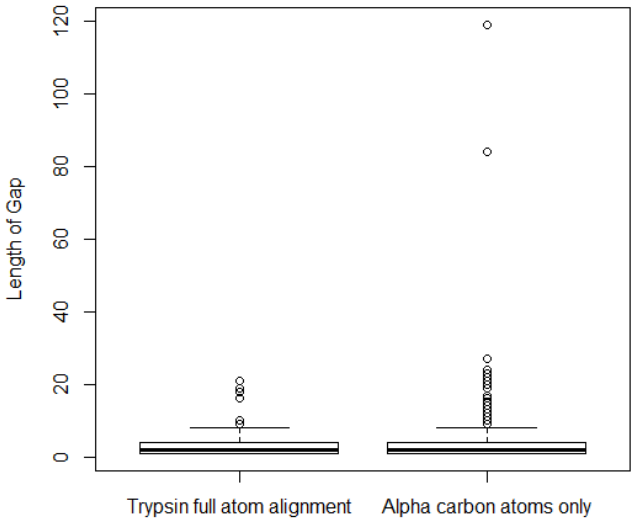


Figure 2.24: Comparison of the length of gaps in the trypsin structural alignment with those in the structural alignment obtained only for the C_α atoms of the trypsin sample.

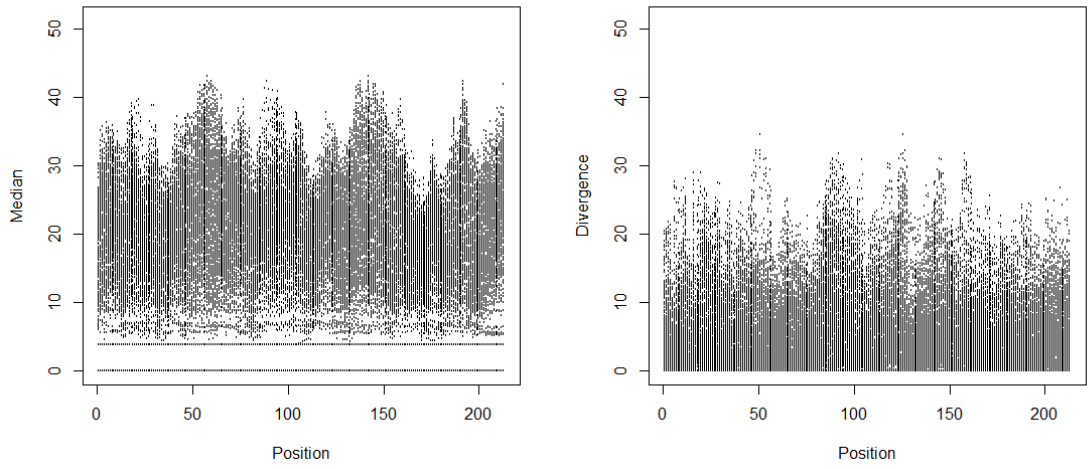
the information from the C_α atoms and the distances between them; therefore, the alignments should be similar.

2.3.4 Aligning the real trypsin sample with anchor residues removed

The following further test was conducted. The anchor residues were removed from the structures in the sample and the resulting structures aligned; if the alignment results in more anchor residues, then MUSTANG is unreliable. The divergence and median were again plotted against position and are displayed in Figure 2.25.

The peaks of the median distance plots in Figure 2.11a and Figure 2.25 are very similar, suggesting that the alignments are similar. However, there is no longer evidence of low-range divergences or anchor residues as the distinct troughs in the divergences in Figure 2.11b are no longer apparent; the divergence between the distances appear to be higher overall.

Therefore, removing the anchor residues produces results in favour of MUSTANG. This suggests that more tests are necessary in order to definitively determine whether the anchor residues are artefacts of MUSTANG.



(a) (b)

Figure 2.25: Plots of the rows of the median and divergence matrices calculated from structurally aligned distance matrices of the trypsin sample with the anchor residues removed. The bars appear as a result of many points plotted close together. (a) Median, $\tilde{d}_{i,j}$, of the structurally aligned distances plotted against position, i, in the alignment. (b) Divergence, $d^{\text{div}}_{i,j}$, of the structurally aligned distances plotted against position, i, in the alignment.

2.4 Effect of gap-closing method on structure shape

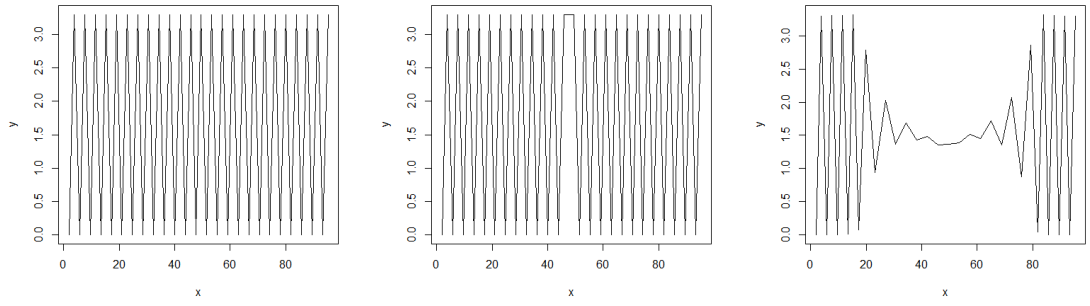
In order to explore the effect of the gap-closing method in Section 2.3.2 on the shape of a structure, we applied it to a selection of shapes typically found in protein secondary structures. The shapes investigated include a straight line, a zigzag and an idealised helix.

2.4.1 Zig-zag

The structure of trypsin has many beta sheets, where the C_α-atoms of residues lie alternately above and below the plane of the beta sheet, not dissimilar to a zigzag. A zigzag structure was generated such that the residues were d_α apart, and such that each set of three consecutive residues formed an equilateral triangle with sides of length d_α. Figure 2.26b shows how the zigzag structure is affected when a gap is closed. The same pattern is observed wherever the gap is placed. However, Figure 2.26c shows the effect on the structure when a gap of size 16 is closed. Clearly, closing large gaps disrupts the structure around the gap significantly.

2.4.2 Idealised helix

The structure of trypsin has two small helices; therefore, it is of interest to analyse how the structure of a helix changes when residues are removed and the gap closed. An idealised helix with 50 residues was generated such that the residues are d_α



(a) (b) (c)

Figure 2.26: Plots displaying the effect of the gap closing method on a zigzag structure. (a) Zigzag structure before a gap is closed. (b) Zigzag structure after closing a gap of size one that is introduced in the middle of the structure. (c) Zigzag structure after closing a gap of size 16 that is introduced in the middle of the structure.

(a) (b) (c)

Figure 2.27: Plots displaying the effect of the gap-closing method on a helix structure. (a) Helix structure before a gap is closed. (b) Helix structure after closing a gap of size one that is introduced in the middle of the structure. (c) Helix structure after closing a gap of size 16 that is introduced in the middle of the structure.

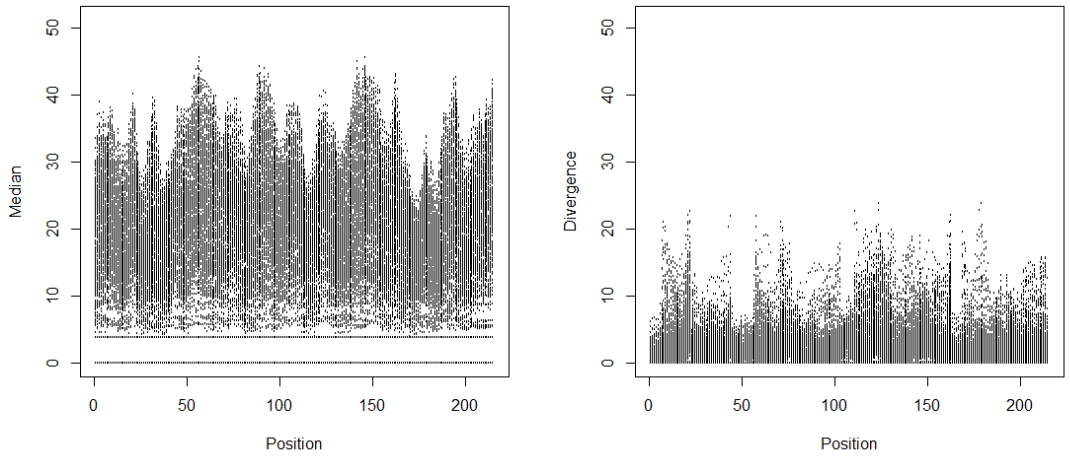
apart and the helix has 3.6 residues per turn. Figure 2.27 displays the effect of the gap-closing method on the helical structure. Figure 2.27b displays the helix structure after one residue is removed. It is difficult to spot, but there is an irregular kink at the end of the helix. This kink occurs regardless of the position of the residue being removed. However, when more residues are removed, the gap is far less subtle. Figure 2.27c displays the result of removing 16 residues and closing the gap; the helical structure is barely recognisable. In fact, the helix structure is almost completely destroyed after only five residues are removed.

2.5 Alternative to Multiple Structure Alignment

One way to be sure that MUSTANG introduces no structural bias is to conduct the analysis using a multiple-sequence alignment of the structures where only sequence and no structural information is used. Distances matrices can be obtained based on the sequence alignment and divergence and median matrices calculated as before. The sequences are aligned using Clustal-W (Thompson *et al.*, 1994), and the divergences and medians plotted against position in Figure 2.28.

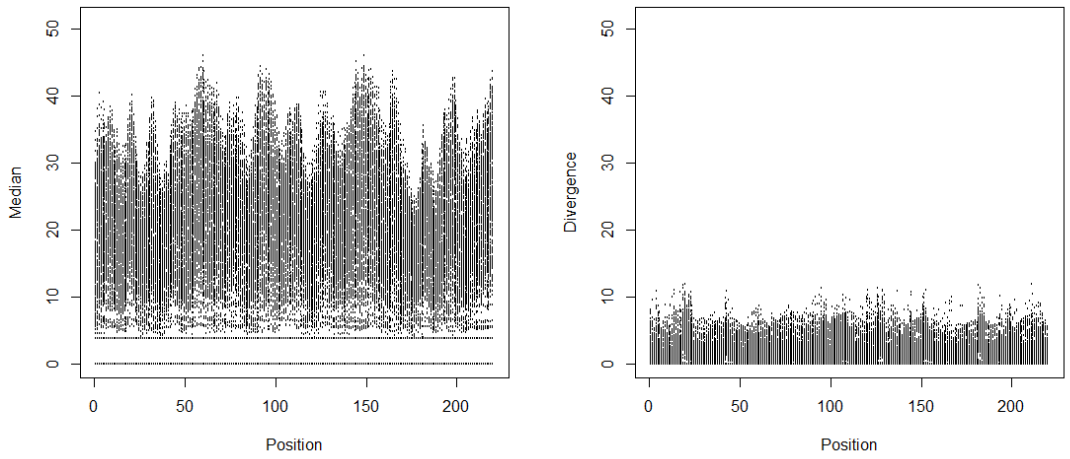
Compared to Figure 2.11b, the divergences in Figure 2.28b are similar in range; however, the divergences in the anchor positions are not as small or distinct. The median plots in Figure 2.11a and Figure 2.28a have a very similar pattern of peaks, further suggesting that the structure alignment is similar to the sequence alignment.

For comparison a second multiple-sequence alignment algorithm is used; MUSCLE (Edgar, 2004). The same plots for this alignment are displayed in Figure 2.29. Compared to Figure 2.28b, the divergences in Figure 2.29b are much smaller overall and there are fewer large divergences. Most of the positions contain divergences



(a) (b)

Figure 2.28: Plots of the rows of the median and divergence matrices calculated from aligned distance matrices of the Clustal-W multiple-sequence alignment of the trypsin sample. The bars appear as a result of many points plotted close together. (a) Median, $\tilde{d}_{i,j}$, of the aligned distances plotted against position, i, in the alignment. (b) Divergence, $d_{i,j}^{\text{div}}$, of the aligned distances plotted against position, i, in the alignment.



(a) (b)

Figure 2.29: Plots of the rows of the median and divergence matrices calculated from aligned distance matrices of the MUSCLE multiple-sequence-alignment of the trypsin sample. The bars appear as a result of many points plotted close together. (a) Median, $\tilde{d}_{i,j}$, of the aligned distances plotted against position, i, in the alignment. (b) Divergence, $d_{i,j}^{\text{div}}$ of the aligned distances plotted against position, i, in the alignment.

small enough to be considered as the anchor residues that were identified previously; however, the divergences are not as low as the troughs in Figure 2.11b. This suggests that the MUSCLE sequence alignment results in more conserved aligned distances compared to the MUSTANG structure alignment, and even the Clustal-W sequence alignment. However, despite producing a better structural alignment than MUSTANG overall, the anchor positions do not appear to be aligned as well. Similarly to Figure 2.28a, the median distances exhibit almost identical peak patterns to Figure 2.11a.

2.6 Discussion

We have presented an investigation into the possibility that the trypsin protein family contains 'anchor' residues. That is, residues where the distance between these residues and every other in the structure is highly conserved across all of the structures in the protein family, compared to the other distances in the structure. These anchor residues were identified from the aligned distance matrices from the structural alignment produced by MUSTANG. We conducted several tests to determine the validity and origin of these anchor residues.

Investigation into the origin of the putative anchor residues did not result in a definitive explanation; while some of the anchor residues appeared to correspond to important conserved residues identified by Rypniewski *et al.* (1994), the evidence was not overwhelming. The anchor residues were not more conserved in sequence compared to the rest of the columns in the structural alignment.

The artefact testing method proposed in Section 2.3.2 proved inconclusive; we would expect anchor residues to be apparent in 1LVY if they truly exist. However, they were not apparent in the artificial structures suggesting that the phenomenon is an artefact of MUSTANG. When the artefact testing method was investigated in Section 2.4 it became clear that the gap-closing method distorts the structures significantly and as a result the distances are also distorted. This method used only the information contained in the C_α atoms of the structures. This was considered reasonable because MUSTANG appears to use this information only. Despite this, aligning only the C_α atoms of the trypsin sample produced a different alignment compared to the trypsin sample; the alignment has more insertions, as well as longer consecutive runs of insertions. This suggests that incorporating only the information contained in the C_α atoms of the structures produces a less desirable alignment, and therefore MUSTANG either incorporates additional information or

is unreliable. Consequently we do not have much confidence in the artefact testing method to accurately determine whether the anchor residues are an artefact of MUSTANG. A simple test of removing the anchor residues in order to test whether MUSTANG would artefactually introduce more anchor residues concluded in favour of MUSTANG, as no new anchor residues were produced. The median distance matrix also provides evidence in favour of the MUSTANG alignment; owing to the fact that the structure produced by multidimensional scaling of the median distance matrix resulted in a homogeneous trypsin structure.

When another protein family was aligned, we expected the anchor residues not to be apparent if MUSTANG is not introducing bias because it is unlikely that this feature would be observed in every protein family. However, the anchor residues may be a feature of protein evolution rather than an artefact. The divergences in each position were all small, suggesting that anchor residues merely identified areas of the alignment where trypsin aligned well. Since this was not a large area it appeared to be an interesting result.

While multiple-sequence alignments do not introduce bias they also do not produce an alignment based on how the structural components are aligned. A reliable structural alignment would be preferred to an alignment based purely on sequence because the protein structure evolves more slowly than sequence.

The Clustal-W sequence alignment results in a similar range of divergences when compared to the MUSTANG alignment. However the MUSCLE sequence alignment is significantly different with a much lower range of divergences overall. We expect differences between the structure and sequence alignments because the structure alignment completely ignores the amino-acid sequence while the sequence alignments only use the amino-acid sequences. MUSTANG ignores the amino-acid sequence in order to align more distantly related proteins; similarly Clustal-W weights sequences based on their similarity. This focus on the evolution of the structures may explain why Clustal-W and MUSCLE produce different alignments.

Out of the tests that were conclusive, many are in favour of MUSTANG. However some tests identify inconsistencies that lead us to believe that MUSTANG may be unreliable. The most convincing result against the existence of anchor residues arose from aligning another protein family; the distances in the short-chain dehydrogenase protein family have smaller divergences than the anchor residues in every position. This strongly suggests that the anchor residues merely indicate well aligned regions of structure in the trypsin family. Combined with the result that the anchor residues do not appear to be strongly conserved in sequence or correspond to important functional residues, we conclude that MUSTANG may be introducing bias, but it is

also possible that the anchor residues are features of the trypsin family. To support this conclusion, a larger range of protein families from diverse organisms would need to be aligned, both in sequence and structure. There is also scope to subject MUSTANG to further testing to determine its reliability.

Chapter 3

Detecting Correlated Mutations in Multiple Sequence Alignments using Regularised Logistic Regression

3.1 Introduction

The structure of a protein is constrained by its function. Sequence alignments from homologous proteins from a range of species provide information on these evolutionary constraints. The analysis of correlated mutations within multiple-sequence alignments can be used to predict residues that are in close proximity in three-dimensional space. We propose a regularised logistic regression model to identify these coevolving positions, and distinguish between the residue correlations that correspond to structural proximity and potential confounding residue correlations, which can occur as a result of noise or other biological evolutionary constraints (Marks *et al.*, 2011). Compared to the method of Sreekumar *et al.* (2011), our method only requires one model to be fitted to the data, with easily interpreted scores as output. Sreekumar *et al.* (2011) fit one model for every pair of columns, and combine the output of thousands of models. We show that our model can successfully identify known coevolving columns in a range of simulated datasets. However, when applied to biological datasets, we obtain mixed results. For 3 of the 7 alignments, at least 80% of the residues identified as coevolving are in close proximity in three-dimensional space. The percentage of predicted coevolving residues that are in contact for the remaining four alignments is between 40-50%.

3.1.1 Critical Assessment of protein Structure Prediction (CASP)

The Critical Assessment of protein Structure Prediction, or CASP, is a worldwide experiment to asses protein structure prediction methods via blind prediction. The CASP experiments started in 1994 and have been held every two years since, resulting in eleven CASP experiments to date. The experiments aim to determine what progress has been made in the field in protein structure prediction (Moult *et al.*, 1995).

The latest CASP experiment; CASP11, started in April 2014. The set of proteins to be predicted are chosen as those that are about to have their structure solved by x-ray crystallography or NMR. Therefore the predictors, and the organisers of CASP, will not know the structure of the proteins, and will only have the sequence information. As the coordinate information for the structures is determined, the prediction models submitted by the predictors are evaluated (Moult *et al.*, 2014).

The prediction of residue-residue contacts was first included in the CASP experiment in CASP2. Monastyrskyy *et al.* (2014) evaluated the performance of residue-residue contact prediction methods in CASP10, and compared these advancements to previous CASPs.

Under CASP regulations, a pair of residues is defined to be in contact when the distance between their C_β atoms (C_α in the case of glycines) is less than 8Å. There are three types of contacts; short, medium and long range. Short range contacts are those separated by 6-11 residues, medium range contacts are separated by 12-23 residues, and long range contacts are at least 24 residues apart in sequence. Any contacts that are separated by less than 6 residues are assumed to correspond to secondary structures.

3.1.2 Regularised Multinomial Regression based Correlated Mutations (RMRCM)

In Section 1.3.2, many correlated mutation methods were introduced. However, there is only one method that utilises generalised linear models. The method of Sreekumar *et al.* (2011), Regularised Multinomial Regression based Correlated Mutations (RMRCM), differs from other methods as it takes into account the network nature of relationships between protein residues to predict correlated mutations be-

tween more than two columns of a multiple sequence alignment. Each column in the alignment is regressed on all other columns, using multinomial regression models.

Each of the sequences in a multiple sequence alignment, A, are mapped to factors with 21 levels (the first 20 levels represent the 20 different amino acids and the last level accounts for gaps). Each of the columns, i, of A is replaced by a binary matrix M_i with 21 columns where 1 represents the occurrence of each particular amino acid. The collection of these binary matrices results in an expanded matrix $M = [M_1, \ldots, M_N]$, where N is the number of alignment columns.

The factor representing column i of A is taken as y, the response variable. This response variable is regressed upon the matrix M_{-i}, which corresponds to the matrix M with the submatrix M_i removed. This is a multinomial regression model because y is a factor with 21 levels. This multinomial model is fitted for each column in A separately. The coefficients β describe the relationships between columns in M_{-i} with the i^{th} column in A. These are then projected back to describe relationships of columns in A with each other. This is achieved by calculating the sum of the absolute values of the regression coefficients to predict links between the columns of the multiple sequence alignment. To reduce the number of parameters, regularisation is used. The elastic-net method is applied when fitting each of the multinomial regression models. We introduce regularisation and define the elastic-net method in Section 3.2.2. The elastic-net mixing parameter, α, for the regularisation was set to 0.99; very close the the Lasso penalty. Each model is fitted for the entire path of solutions of the regularisation parameter λ. The default sequence of 100 values of λ were calculated and the Bayesian Information Criterion (BIC) used to select the best value. The predicted contacts with the minimum BIC are chosen for each column separately. In a different approach, the coefficients β are summed over all values of λ, as this often produced better results (Sreekumar *et al.*, 2011).

The performance of RMRCM was tested on artificial datasets and compared to the corrected Mutual Information (MI). RMRCM out performed MI in every case. When applied to biological data, the performance of RMRCM depends on the number of sequences in the alignment. In particular, when applied to the CASP9 data, RMRCM outperformed other correlated mutation analysis methods when the number of sequences was at least 1000. However, it performed worse than average compared to others methods in CASP9 when the number of sequences was low.

3.2 A New Regularised Logistic Regression Model

We propose a method to identify coevolving columns in a multiple sequence alignment. A regularised logistic regression model is used to score alignment columns based on their interaction coefficients. Large interaction coefficients are indicative of coevolving alignment columns.

Before fitting the model, alignment columns that consist of more than 50% gaps are removed. Alignment columns that largely consist of gaps contain less evidence for coevolution. Alignment columns that are more than 90% conserved are also removed, as they are unable to covary.

3.2.1 Logistic Regression Model Setup

Given a binary outcome variable y_i and p associated predictor variables; $X_i = (x_{i1}, \ldots, x_{ip})^T$, we propose a logistic regression model to discover which predictors are important. Logistic regression is typically applied to data with a binary response variable. To apply this model to sequence alignment data, we view the data as a case control study. Denote the input multiple sequence alignment by an $n \times m$ matrix, A_I, where $i' = 1, \ldots, n$ represents the sequences, and $k = 1, \ldots, m$ represents the alignment columns. Each sequence in A_I is a case. For the controls, we generate pseudo controls by independently shuffling the columns in the true alignment.

Generating Pseudo Controls

We generate the control data from the input alignment, A_I. Each column, l, is permuted independently to produce a new alignment, A_1. Randomising the order of the amino acids in the columns removes the coevolutionary signal by disrupting the correlated mutations.

The input alignment, A_I, is randomised multiple times to produce the control data. We fit the logistic regression model to the following alignment data

$$A^*_{N\times m} = (A_I, A_1, \ldots, A_c)^T,$$

where c is the number of control alignments and $i = 1, \ldots, N$ is the total number of sequences including the pseudo control sequences. That is, $N = n(c + 1)$.

The columns of the input alignment, A_I, are randomised 5 times to produce 5 controls per case. The response variable is thus given by

$$y_i = \begin{cases} 1 & \text{if sequence } i \text{ belongs to the true alignment} \\ 0 & \text{if sequence } i \text{ is a pseudo control} \end{cases}$$

where $i = 1, \ldots, N$ is the total number of sequences including the pseudo control sequences. The values y_i are realisations of a random variable Y_i that take the values one and zero with probabilities π_i and $1 - \pi_i$. Therefore Y_i follows a Bernoulli distribution with parameter π_i.

The matrix X is constructed from the input alignment A^*. We can partition X in terms of the main effects and the second order interaction terms as follows

$$X_{N\times p} = \left(X^{\text{main}}_{N\times 21m}, X^{\text{ints}}_{N\times 441m(m-1)}\right), \tag{3.1}$$

where the number of parameters in the model, p, is given by $p = 21m + 441m(m-1)$. To understand how this number is calculated, we explain in more detail below.

The matrix corresponding to the main effects, $X^{\text{main}}_{N\times 21m}$ in Equation (3.1), can be partitioned in terms of the columns of A^* to produce the following submatrices

$$X^{\text{main}}_{N\times 21m} = \left(X^{(1)}, X^{(2)}, \cdots, X^{(m)}\right), \tag{3.2}$$

where $X^{(k)}_{N\times 21}$, $k = 1, \ldots, m$, is a binary matrix representing the occurrence of the amino acids in column k of the alignment A^*. Each row of $X^{(k)}$ corresponds to the sequences in the alignment. There is a column in $X^{(k)}$ for each of the 20 standard amino acids, and a column for gaps.

For example, consider the simple alignment given in Figure 3.1. The submatrix $X^{(1)}$ in Equation (3.2) is constructed from the first column of the alignment. For each sequence, the corresponding amino acid is coded one in the appropriate column

L H A F D A
R D A F D N
K D A G D A
E H A G D I

Figure 3.1: A simple example alignment consisting of 4 sequences and 6 alignment columns.

of $X^{(1)}$ and all other columns are coded zero, resulting in the following matrix

$$X^{(1)} = \begin{array}{c} \\ \text{Seq 1} \\ \text{Seq 2} \\ \text{Seq 3} \\ \text{Seq 4} \end{array} \begin{array}{c} \begin{array}{cccccccccccccc} A & R & N & D & C & Q & E & G & H & I & L & K & \cdots & - \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \cdots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \end{pmatrix} \end{array}.$$

The matrix corresponding to the second order interaction effects, $X^{\text{ints}}_{N\times 441m(m-1)}$ in Equation (3.1), can be partitioned in terms of each pair of alignment columns to produce the following submatrices

$$X^{\text{ints}}_{N\times 441m(m-1)} = \left(X^{(1,2)}, X^{(1,3)}, \cdots, X^{(1,m)}, X^{(2,3)}, X^{(2,4)}, \cdots, X^{((m-1),m)}\right) \tag{3.3}$$

where $X^{(k,l)}_{N\times 441}$ is a binary matrix representing the occurrence of amino acid pairs in the pair of columns k and l of the alignment A^*. The rows of $X^{(k,l)}$ correspond to the sequences in the alignment. There is a column in $X^{(k,l)}$ for every pairwise combination of the 20 standard amino acids and gaps. Thus, $X^{(k,l)}$ consists of $21^2 = 441$ columns.

Consider again the simple alignment given in Figure 3.1. The submatrix $X^{(1,2)}$ in Equation (3.3) is constructed from columns 1 and 2 of the alignment. For each sequence, the corresponding amino acid pair is coded one in the appropriate column of $X^{(1,2)}$ and all other columns are coded zero, resulting in the following matrix

$$X^{(1,2)} = \begin{array}{c} \\ \text{Seq 1} \\ \text{Seq 2} \\ \text{Seq 3} \\ \text{Seq 4} \end{array} \begin{array}{c} \begin{array}{ccccccccccc} \text{A,A} & \text{A,R} & \cdots & \text{R,D} & \cdots & \text{E,H} & \cdots & \text{L,H} & \text{K,D} & \cdots & \text{–,–} \end{array} \\ \begin{pmatrix} 0 & 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 1 & \cdots & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 & 1 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & \cdots & 1 & \cdots & 0 & 0 & \cdots & 0 \end{pmatrix} \end{array}.$$

For the binary outcome variable y_i and p associated predictor variables; $X_i = (x_{i1}, \ldots, x_{ip})^T$, the logistic regression model is given by

$$\text{logit}(\pi_i) = \log\left(\frac{\pi_i}{1-\pi_i}\right) = \beta_0 + x_i^T\boldsymbol{\beta} \tag{3.4}$$

Solving for π, this gives

$$\pi_i = \frac{e^{\beta_0 + x_i^T \boldsymbol{\beta}}}{1 + e^{\beta_0 + x_i^T \boldsymbol{\beta}}} = \frac{1}{1 + e^{-(\beta_0 + x_i^T \boldsymbol{\beta})}}, \tag{3.5}$$

where $\boldsymbol{\beta}$ can be partitioned similarly to X in Equation (3.1) to produce subvectors corresponding to the main effects and the second order interaction terms as follows

$$\boldsymbol{\beta}_{p\times 1} = \left(\boldsymbol{\beta}^{\text{main}}_{21m\times 1}, \boldsymbol{\beta}^{\text{ints}}_{441m(m-1)\times 1}\right)^T.$$

The vector corresponding to the main effects, $\boldsymbol{\beta}^{\text{main}}_{21m\times 1}$ in Equation (3.2.1), can be partitioned in terms of the columns of A^* to produce the following subvectors

$$\boldsymbol{\beta}^{\text{main}}_{21m\times 1} = \left(\boldsymbol{\beta}^{(1)}, \boldsymbol{\beta}^{(2)}, \cdots, \boldsymbol{\beta}^{(m)}\right)^T \tag{3.6}$$

where $\boldsymbol{\beta}^{(k)}_{21\times 1}$, $k = 1, \ldots, m$, is the vector of regression coefficients corresponding to each amino acid in column k of the alignment A^*. Each element in $\boldsymbol{\beta}^{(k)}$ corresponds to the 20 standard amino acids, plus gaps.

The vector corresponding to the second order interaction terms, $\boldsymbol{\beta}^{\text{ints}}_{441m(m-1)\times 1}$ in Equation (3.2.1), can be partitioned in terms of each pair of alignment columns in A^* to produce the following subvectors

$$\boldsymbol{\beta}^{\text{ints}}_{441m(m-1)\times 1} = \left(\boldsymbol{\beta}^{(1,2)}, \boldsymbol{\beta}^{(1,3)}, \cdots, \boldsymbol{\beta}^{(1,m)}, \boldsymbol{\beta}^{(2,3)}, \boldsymbol{\beta}^{(2,4)}, \cdots, \boldsymbol{\beta}^{((m-1),m)}\right)^T \tag{3.7}$$

where $\boldsymbol{\beta}^{(k,l)}_{441\times 1}$ is the vector of regression coefficients corresponding to each pair of amino acid in the column pair (k, l) of the alignment A^*. Each element in $\boldsymbol{\beta}^{(k,l)}$ corresponds to the 441 pairs of amino acids and gaps.

3.2.2 Fitting the Regularised Model in R

Due to the high-dimensional nature of our data, with $p \gg N$, any linear model is over-parameterised and regularisation is needed to achieve a stable fit (Hastie *et al.*, 2015). There are a large number of predictors, however we are interested in a smaller subset that exhibit the strongest effects.

We fit our regularised logistic regression model using the `glmnet` function of the *glmnet* package (Friedman *et al.*, 2010) in R (R Core Team, 2013). The model is

fitted using penalised maximum likelihood. The likelihood function for the logistic regression model is derived as follows.

Recall, the distribution of Y_i is Bernoulli with parameter p_i, given by

$$\Pr(Y_i = y_i) = \pi_i^{y_i}(1-\pi_i)^{1-y_i},$$

for $y_i = 0, 1$. Therefore, the likelihood function is

$$L(\mathbf{y}, X, \boldsymbol{\beta}) = \prod_{i=1}^{N} \pi_i^{y_i}(1-\pi_i)^{1-y_i},$$

the corresponding log-likelihood is thus

$$\begin{aligned} l(\mathbf{y}, X, \boldsymbol{\beta}) &= \log L(\mathbf{y}, X, \boldsymbol{\beta}) \\ &= \sum_{i=1}^{N} \{y_i \log \pi_i + (1-y_i)\log(1-\pi_i)\} \\ &= \sum_{i=1}^{N} \left\{ y_i \log\left(\frac{\pi_i}{1-\pi_i}\right) + \log(1-\pi_i) \right\}. \end{aligned}$$

Substituting $\log\left(\frac{\pi_i}{1-\pi_i}\right)$ from Equation (3.4) and π_i from Equation (3.5) gives

$$\begin{aligned} l(\mathbf{y}, X, \boldsymbol{\beta}) &= \sum_{i=1}^{N} \left\{ y_i \left(\beta_0 + x_i^T\boldsymbol{\beta}\right) + \log\left(1 - \frac{1}{1+e^{-(\beta_0 + x_i^T\boldsymbol{\beta})}}\right) \right\} \\ &= \sum_{i=1}^{N} \left\{ y_i \left(\beta_0 + x_i^T\boldsymbol{\beta}\right) + \log\left(\frac{1}{1+e^{(\beta_0 + x_i^T\boldsymbol{\beta})}}\right) \right\} \\ &= \sum_{i=1}^{N} \left\{ y_i \left(\beta_0 + x_i^T\boldsymbol{\beta}\right) - \log\left(1+e^{(\beta_0 + x_i^T\boldsymbol{\beta})}\right) \right\}. \end{aligned}$$

Regularisation constrains the coefficients by applying a penalty on the coefficients for each variable. The `glmnet` function solves the following problem:

$$\underset{\beta \in \mathbb{R}^p}{\operatorname{argmax}} \left\{ \frac{1}{N} l(\mathbf{y}, X, \boldsymbol{\beta}) - \lambda P_\alpha(\boldsymbol{\beta}) \right\}, \tag{3.8}$$

where $P_\alpha(\boldsymbol{\beta})$ is the elastic-net penalty

$$P_\alpha(\boldsymbol{\beta}) = \sum_{j=1}^{p} \left[\frac{1}{2}(1-\alpha)\beta_j^2 + \alpha|\beta_j| \right]. \tag{3.9}$$

The elastic-net linearly combines the lasso and ridge regression penalties; $0 \leq \alpha \leq 1$ is the elastic-net mixing parameter. When $\alpha = 1$, Equation (3.9) is the lasso penalty and when $\alpha = 0$ it gives the ridge regression penalty. Setting α between these extremes gives the elastic-net penalty. The tuning parameter, $\lambda \geq 0$, controls the amount of shrinkage that is applied to the estimates. When $\lambda = 0$, no shrinkage is applied. When $\lambda = \infty$ the lasso and ridge estimates are equal to zero. When λ is between these values, the ridge and lasso estimates are shrunk towards zero. In the case of the lasso, some of these estimates may be exactly zero (Hastie *et al.*, 2015).

For each value of λ there is a solution to Equation (3.8). Therefore there is a path of solutions given by the chosen values of λ. The regularisation path is calculated for the elastic-net parameter for an entire path of solutions of the regularisation parameter λ. The default sequence of 100 values of λ supplied by `glmnet` were used.

3.2.3 Scoring Alignment Columns

The regression coefficients of the second order interaction terms provide information about which pairs of columns are covarying. There is a β coefficient for every distinct pair of amino acids in every pair of columns, contained in the vector $\boldsymbol{\beta}^{\text{ints}}$ described in Equation (3.7). Recall that $\boldsymbol{\beta}^{(k,l)}$ contains the regression coefficients for the second order interaction terms corresponding to column pair (k, l). We define $\beta_{(a,b)}^{(k,l)}$ to be the β coefficient corresponding to the amino acid pair (a, b) in columns k and l of the alignment A^*.

The score for alignment columns k and l is given by summing the coefficients over all amino acid pairs (a, b) as follows

$$s_{k,l} = \sum_{(a,b)} \beta_{(a,b)}^{(k,l)},$$

where (a, b) represent each pair of amino acids. A large value of $s_{k,l}$ indicates columns k and l are coevolving.

3.3 Results

We fit our model to a series of small test alignments to explore the ability of our model to identify columns that are set up to covary. We design simulations to determine the optimal value of the elastic-net parameter α and the regularisation parameter λ under various alignment scenarios. We explore the effect of changing the number of columns and sequences, the number of coevolving columns and adding noise to the coevolving columns. We apply our method to a range of Pfam datasets.

3.3.1 Test Alignments

First we explore our proposed model by fitting it to simple test alignments. We start with a 10 column alignment with 100 sequences, before extending our analysis to alignments consisting of 30 and 50 columns. For alignments with 30 columns we look at the effect of having a very small number of sequences, 20, and 100 sequences. For alignments with 50 columns we increase the number of sequences to 50 and 100. We also experiment with the number of coevolving columns and the amount of noise added to these columns.

Simulating Test Alignments

Each alignment is simulated by randomly sampling with replacement the 20 standard amino acids to generate one alignment row. This row is used as the base for the whole alignment. The sampled row is multiplied to produce the desired number of sequences, resulting in a base alignment that is 100% conserved.

Protein family alignments are rarely completely conserved, therefore we replace 60% of the alignment with random amino acids in randomly sampled positions. We also replace 10% of the alignment with gaps in randomly sampled positions.

To generate coevolving columns a pair of columns is randomly sampled. The amino acids in the sampled columns are replaced to reflect coevolution. If the simulated alignment consists of 100 sequences or less, there is one correlated mutation event, as displayed in Figure 3.2a. If the simulated alignment contains more than 100 sequences, there are two correlated mutation events, as displayed in Figure 3.2b. The amino acids that make up the coevolving columns are randomly sampled from the 20 standard amino acids. If more than one pair of coevolving columns is simulated, the sequence order is shuffled between setting up each pair of columns. This ensures that each pair of columns are coevolving independently, and not together.

Figure 3.2: Simple example alignments displaying a pair of columns set up to reflect coevolution. a) An example alignment consisting of one correlated mutation event. The residue pair 'A' and 'D', given in red, are substituted for the pair 'Y' and 'E', in blue. (b) An example alignment consisting of two correlated mutation events. The residue pair 'A' and 'D', given in red, are substituted for the pair 'Y' and 'E' in blue or 'M' and 'G' in green.

Ten Column Alignment

We generate an alignment consisting of 10 columns and 100 sequences. Columns 2 and 8 are coevolving, and separately, columns 3 and 7 are coevolving. The model is fitted to the alignment at a range of values of the elastic-net parameter, $\alpha = (0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1)$. The regularisation path is calculated for the elastic-net parameter at a grid of 100 values for the regularisation parameter, λ, chosen by `glmnet`.

We want to determine which values of α and λ successfully identify the coevolving columns. A standard method for selecting an appropriate value of λ is cross-validation. The value of λ is chosen to maximise some measure of model fit, for example the percentage of null deviance explained. However, cross-validation does not identify the coevolving columns. Figure 3.3 displays the combinations of α and λ that produce non-zero scores for $s_{2,8}$ and $s_{3,7}$, and shrink the scores for all other column pairs to zero.

For the alignment with 10 columns and 100 sequences there are multiple optimal

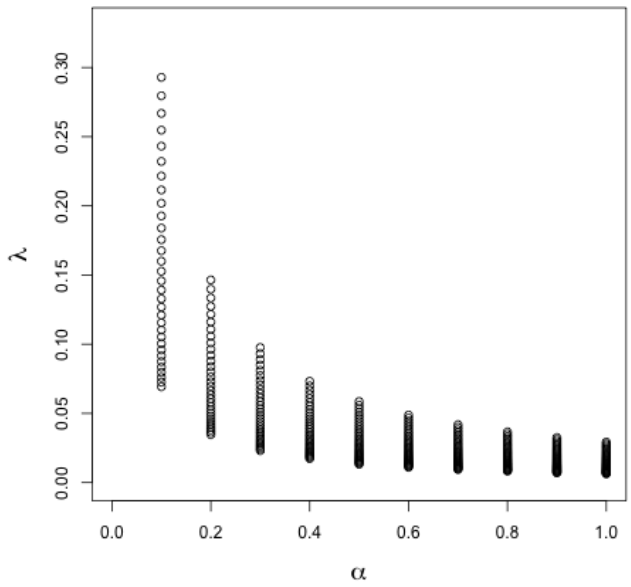


Figure 3.3: Values of the elastic-net parameter, α, and the regularisation parameter, λ, that identify $s_{2,8}$ and $s_{3,7}$ as the only non-zero scores. These scores correspond to the coevolving columns in the 10 column alignment with 100 sequences.

combinations of α and λ, that result in our model successfully identifying only the coevolving columns. However, true alignments are rarely this simple.

Thirty Column Alignments

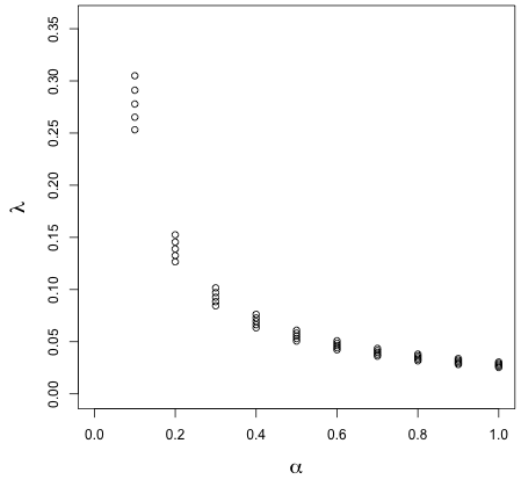
We explore the effect of the number of sequences, the number of coevolving columns, and the effect of adding random noise to these columns. To add noise to the coevolving columns, a percentage of the amino acids in each column are replaced with random amino acids. We generate alignments consisting of 30 columns, and 20 and 100 sequences. We use the following parameter combinations:

Number of coevolving column pairs 1, 2, 3, 4

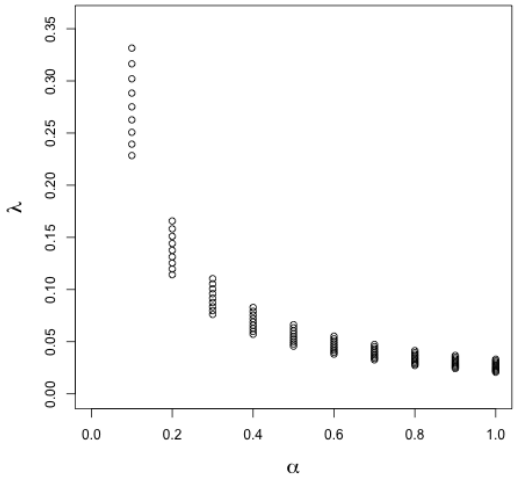
Percentage of random noise added 5%, 10%, 15%, 20%, 25%

We fit our model to each of these 20 alignments at a range of values of the elastic-net parameter, $\alpha = (0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1)$. The regularisation path is calculated for each value of the elastic-net parameter at a grid of 100 values for the regularisation parameter, λ.

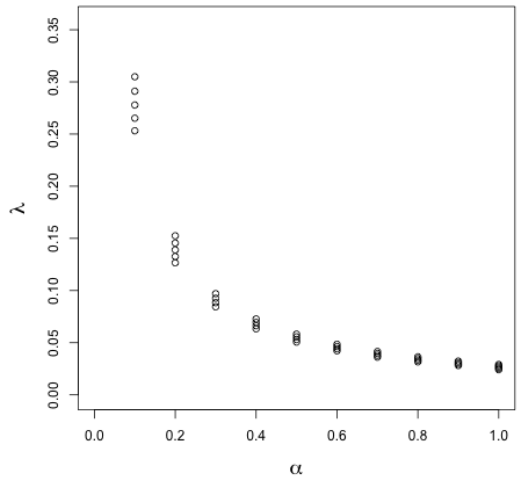
We want to explore which combinations of α and λ are optimal for all 20 alignments. Figure 3.4 displays the combinations of α and λ that identify the coevolving



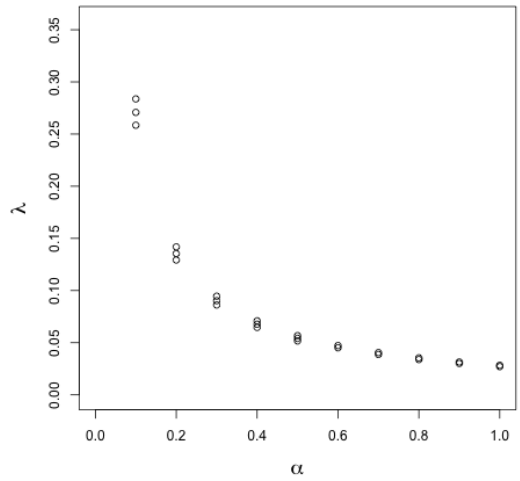
(a) Coevolving pairs=1, noise=0.25



(b) Coevolving pairs=2, noise=0.25



(c) Coevolving pairs=3, noise=0.25



(d) Coevolving pairs=4, noise=0.25

Figure 3.4: Values of the elastic-net parameter, α, and the regularisation parameter, λ, that identify the coevolving column scores as the only non-zero scores for the case where 25% noise is added to the coevolving columns. Each plot corresponds to an alignment, each consisting of 30 columns and 20 sequences. The number of coevolving column pairs differs for each plot. (a) 1 coevolving pair of columns. (b) 2 coevolving pairs of columns. (c) 3 coevolving pairs of columns. (d) 4 coevolving pairs of columns.

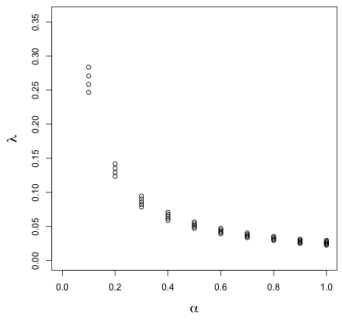
column scores, $s_{k,l}$, as the only non-zero scores. Each plot corresponds to a different number of coevolving columns; each with 25% noise added to the coevolving columns.

The range of λ values appears to decrease slightly for each value of α as the

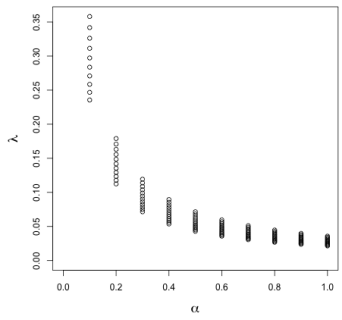
number of coevolving pairs increases. The number of optimal λ values also decreases slightly. However, this pattern is not observed across other noise levels, see Appendix D.1.

Figure 3.5 displays the combinations of α and λ that identify the coevolving column scores as the only non-zero scores. Each plot corresponds to a different percentage of noise added to the coevolving columns; each for alignments with 3 coevolving pairs of columns. There appears to be no obvious pattern as the noise increases.

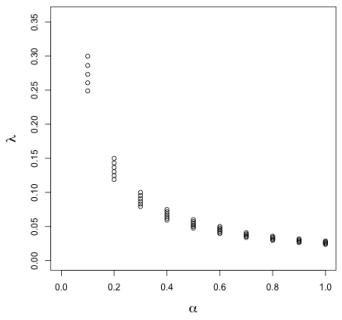
Only one alignment does not have a combination of α and λ that identifies only the coevolving column scores as non-zero. This alignment has 4 coevolving pairs of columns and 10% noise added to these columns. In addition to the scores for the



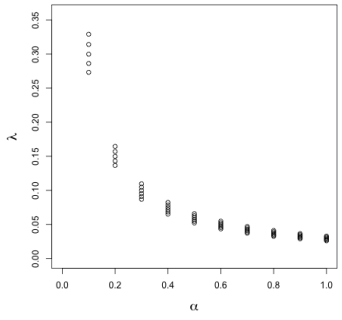
(a) Coevolving pairs=3, noise=0.05



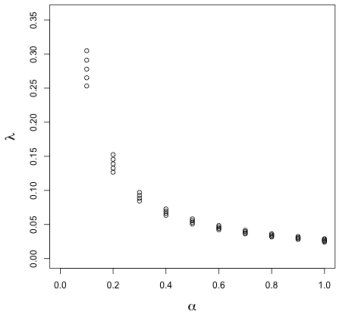
(b) Coevolving pairs=3, noise=0.1



(c) Coevolving pairs=3, noise=0.15



(d) Coevolving pairs=3, noise=0.2



(e) Coevolving pairs=3, noise=0.25

Figure 3.5: Values of the elastic-net parameter, α, and the regularisation parameter, λ, that identify the coevolving column scores as the only non-zero scores for the case where there are 3 coevolving pairs of columns. Each plot corresponds to an alignment, each consisting of 30 columns and 20 sequences. The percentage of noise added differs for each plot. (a) 5% noise added. (b) 10% noise added. (c) 15% noise added. (d) 20% noise added. (e) 25% noise added.

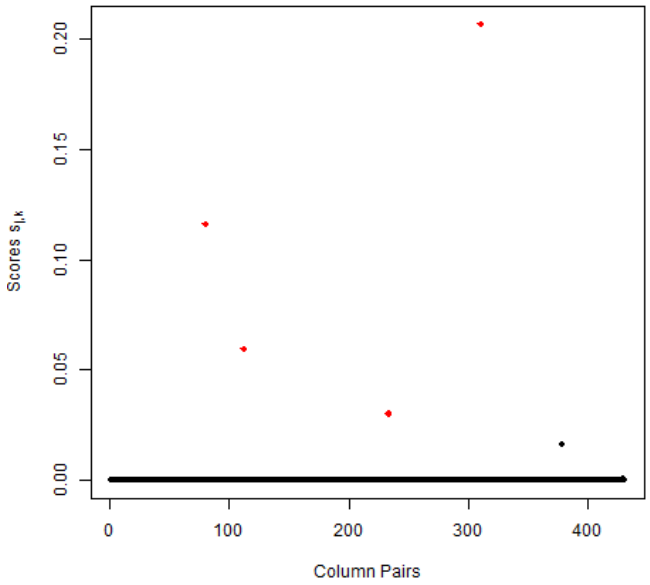


Figure 3.6: Coefficient scores for an alignment with 30 columns and 20 sequences. The alignment has 4 coevolving pairs of columns and 10% noise added to the coevolving columns. The red points correspond to the coevolving column scores, and the black points represent the scores for all other pairs of columns in the alignment. The only non-zero black point corresponds to columns 20 and 23.

coevolving columns, another score, $s_{20,23}$, for columns 20 and 23 is also non-zero for many combinations of α and λ. Figure 3.6 displays the scores for each pair of alignment columns for $\alpha = 0.1, \lambda = 0.2464$. The red points correspond to the true coevolving column scores and the black points represent the scores for the other pairs of columns. All of the black points are zero as expected, except $s_{20,23}$.

For each value of α we observe a range of λ values that are able to identify the known coevolving columns. The range of λ values are consecutive runs of the regularisation path, suggesting that any value of λ in this range would be optimal. There are a small range of optimal λ values for 7 of the α values, that are common to 17 of the 20 alignments. The combination of α and λ values that are optimal are given in Table 3.1.

To test whether the optimal combinations of α and λ in Table 3.1 have the ability to identify coevolving columns in other alignments, we fit our model with these values supplied. The regularisation path for each value of α is calculated for 100 values of λ spanning the range given in the table. Alignments are generated with 30 columns, 20 sequences, and 1-5 coevolving pairs of columns, each of which has 25% noise added. We obtain mixed results:

Table 3.1: Optimal values of α, and the range of optimal λ values for each. Optimal combinations are those that identify the coevolving columns. These combinations are common to 17 of the 20 30×20 alignments.

α	λ range
0.1	[0.2732, 0.2837]
0.2	[0.1366, 0.1418]
0.3	[0.0886, 0.0946]
0.4	[0.0664, 0.0709]
0.5	[0.0532, 0.0567]
0.6	[0.0451, 0.0473]
0.7	[0.0387, 0.0405]

1 pair of coevolving columns Every combination of α and λ identify the known coevolving pair of columns. However, an additional score for another pair of columns is also identified.

2 pairs of coevolving columns Every combination of α and λ successfully identify the coevolving columns in the alignment.

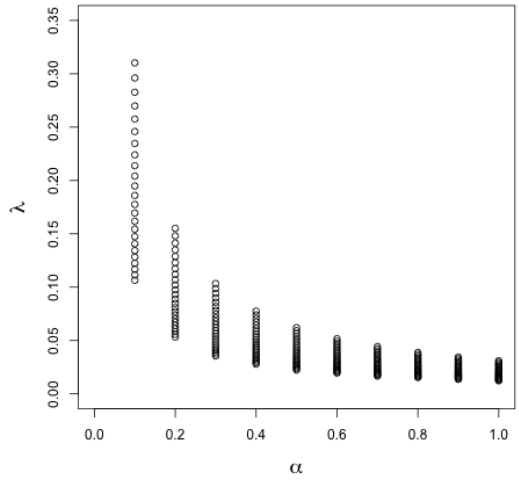
3 pairs of coevolving columns Every combination of α and λ identify the same pairs of columns as coevolving. However, only two of the scores correspond to known coevolving columns.

4 pairs of coevolving columns Every combination of α and λ successfully identify the coevolving columns in the alignment.

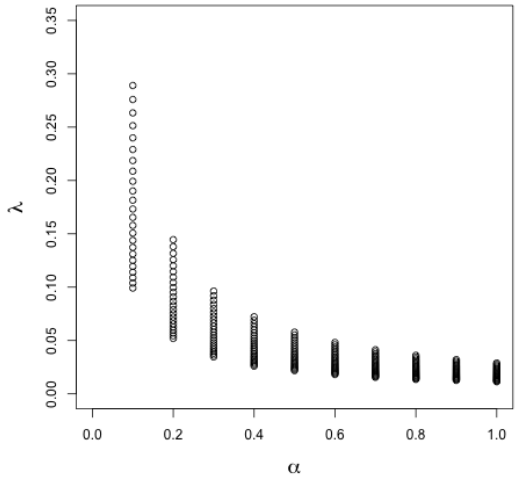
5 pairs of coevolving columns For $\alpha = 0.1$, all values of λ identify the known coevolving columns. However, an additional score, $s_{16,29}$, is also identified. For $\alpha = (0.2, 0.3, 0.4)$, all values of λ identify 4 out of the 5 coevolving columns, and an additional score; $s_{16,29}$. For $\alpha = (0.5, 0.6, 0.7)$, all values of λ identify 3 out of the 5 coevolving columns, and an additional score; $s_{16,29}$.

Using the same parameter combinations for the coevolving columns as the 20-sequence case, the number of sequences is increased to 100. For every alignment there are multiple optimal combinations of α and λ that successfully identify the coevolving column scores as the only non-zero scores. Figure 3.7 displays the combinations of α and λ that identify the coevolving column scores, $s_{k,l}$, as the only non-zero scores. Each plot corresponds to a different number of coevolving columns; each with 25% noise added to the coevolving columns. Plots for the remaining parameter combinations are given in Appendix D.2.

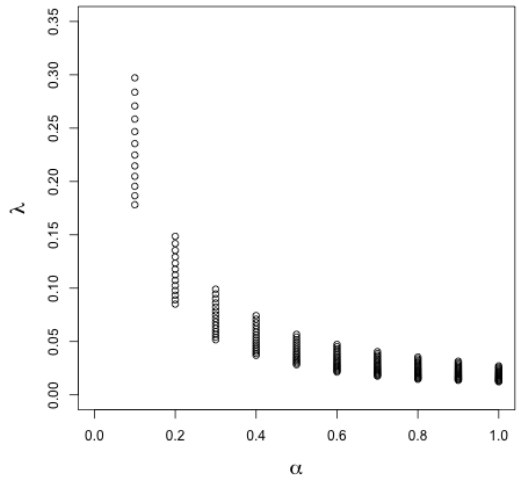
Similarly to the 20-sequence case, there appears to be no pattern in the α/λ combinations as the number of coevolving pairs or percentage of noise increases.



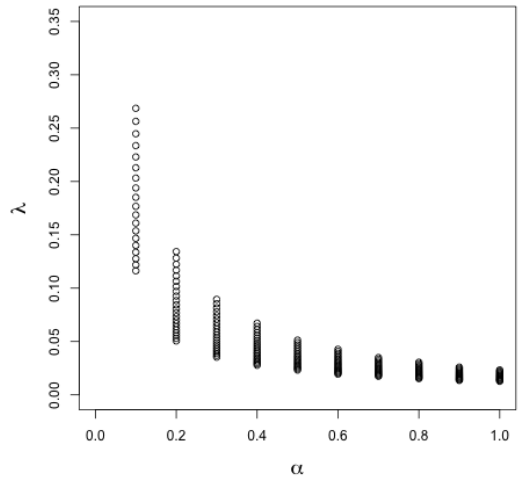
(a) Coevolving pairs=1, noise=0.25



(b) Coevolving pairs=2, noise=0.25



(c) Coevolving pairs=3, noise=0.25



(d) Coevolving pairs=4, noise=0.25

Figure 3.7: Values of the elastic-net parameter, α, and the regularisation parameter, λ, that identify the coevolving column scores as the only non-zero scores for the case where 25% noise is added to the coevolving columns. Each plot corresponds to an alignment, each consisting of 30 columns and 100 sequences. The number of coevolving column pairs differs for each plot. (a) 1 coevolving pair of columns. (b) 2 coevolving pairs of columns. (c) 3 coevolving pairs of columns. (d) 4 coevolving pairs of columns.

However, the number of optimal combinations of α and λ is larger, and the range of optimal λ values is larger for each value of α.

As in the 20-sequence case, the range of λ values are consecutive runs of the regularisation path, suggesting that any value of λ in this range would be optimal.

Table 3.2: Optimal values of α, and the range of optimal λ values for each. Optimal combinations are those that identify the coevolving columns. These combinations are common to all of the 30×100 alignments.

α	λ range
0.1	[0.1781, 0.2686]
0.2	[0.0850, 0.1343]
0.3	[0.0516, 0.0895]
0.4	[0.0370, 0.0671]
0.5	[0.0282, 0.0513]
0.6	[0.0218, 0.0427]
0.7	[0.0178, 0.0350]
0.8	[0.0156, 0.0306]
0.9	[0.0139, 0.0260]
1.0	[0.0128, 0.0234]

There is a range of optimal λ values for 10 of the α values, that are common to all of the alignments. The combination of α and λ values that are optimal are given in Table 3.2. Compared to the 20-sequence case, there are more optimal combinations of α and λ common to each alignment.

To test whether the optimal combinations of α and λ in Table 3.2 have the ability to identify coevolving columns in other alignments, we fit our model with these values supplied. The regularisation path for each values of α is calculated for 100 values of λ spanning the range given in the table. Alignments are generated with 30 columns, 100 sequences, and 1-5 coevolving pairs of columns, each of which has 25% noise added.

The model successfully identifies the known coevolving pairs of columns for all of the 30x100 alignments, including the previously uncalibrated case with 5 pairs of coevolving columns. The difference in success between the 20- and 100-sequence cases illustrates the dependence of our model on the number of sequences in an alignment.

Fifty Column Alignments

We expand our simulations to 50 column alignments. Again, we explore the effect of the number of sequences, the number of coevolving columns, and the effect of adding random noise to these columns. We use the following parameter combinations:

Number of sequences 50, 100

Number of coevolving pairs 1, 2, 3, 4, 5

Percentage of random noise added 5%, 10%, 15%, 20%, 25%

α	λ range
0.1	[0.2093, 0.2627]
0.2	[0.1047, 0.1313]
0.3	[0.0698, 0.0876]
0.4	[0.0500, 0.0627]
0.5	[0.0400, 0.0501]
0.6	[0.0333, 0.0418]
0.7	[0.0286, 0.0358]
0.8	[0.0250, 0.0299]
0.9	[0.0222, 0.0266]
1.0	[0.0200, 0.0228]

Table 3.3: Optimal values of α, and the range of optimal λ values for each. Optimal combinations are those that identify the co-evolving columns. These combinations are common to all of the 50 column alignments, irrespective of the number of sequences.

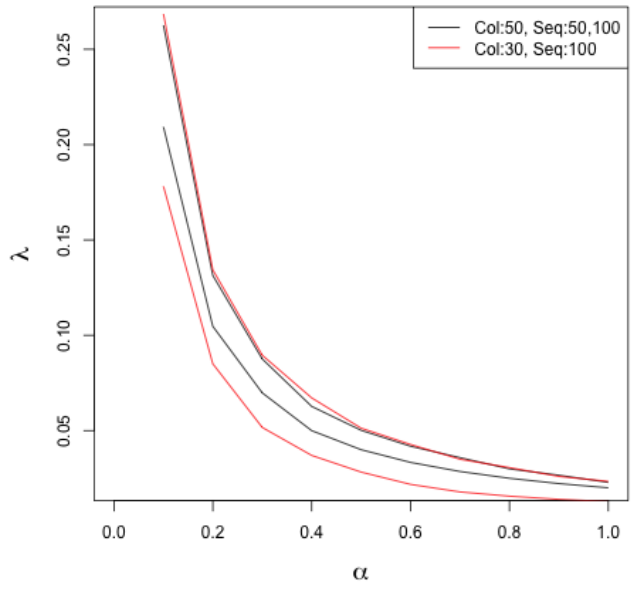


Figure 3.8: Overlap between the optimal range of λ values for each value of α. The black lines correspond to the minimum and maximum optimal values of λ for each α, in the 50-column alignments. The red lines correspond to the 30-column alignment, with 100 sequences.

For the 50- and 100- sequence cases, there are many optimal combinations of α and λ, as given in Table 3.3. Figure 3.8 displays the overlap between the optimal range of α and λ values across the 30×100 alignments (Table 3.2) and the 50×50 and 50×100 alignments (Table 3.3).

3.3.2 Biological Data

We used the same Pfam (Bateman *et al.*, 2004) protein family alignments and reference PDB structures as Sreekumar *et al.* (2011). The 7 selected alignments are

Pfam ID[a]	Number of columns[b]	Number of columns$^{c}_{\mathrm{T}}$	Number of sequences[d]	PDB ID:Chain[e]	UniProtKB ID[f]
PF00029	171	86	424	2ZW3:A	CXB2_HUMAN
PF00193	168	83	432	1POZ:A	CD44_HUMAN
PF00157	117	66	484	1POU:A	PO2F1_HUMAN
PF00243	227	85	2260	1BET:A	NGF_MOUSE
PF00366	144	60	2447	1I94:Q	RS17_THET8
PF00276	304	86	2753	3G6E:S	RL23_HALMA
PF00105	279	57	3702	2NLL:B	THB_HUMAN

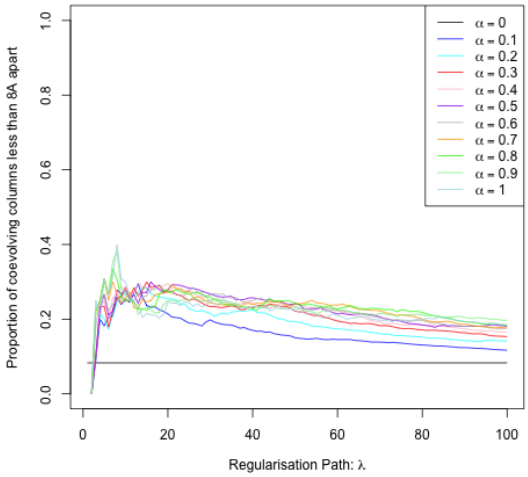
Table 3.4: Pfam alignment data organised by ascending sequence number. [a] Pfam accession code for each alignment. [b,c] The number of columns in the Pfam alignments, and the number of columns after our method has removed highly conserved and gapped columns, respectively. [d] The number of sequences in each alignment. [e] The PBD identifier and chain number of the reference structure used for each alignment. [f] The UniProt ID used to map from the Pfam alignment to the PDB structure

summarised in Table 3.4.

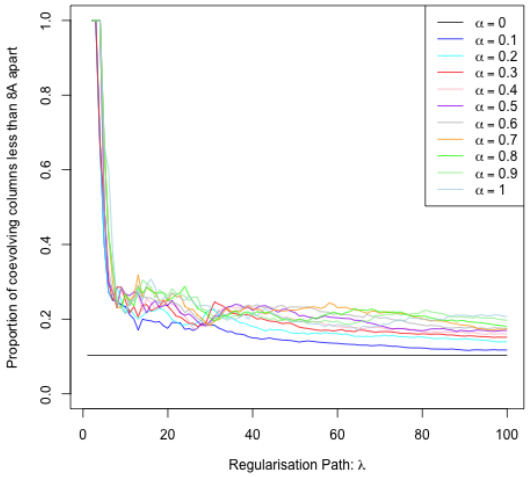
In Section 3.1.1 we report that under CASP regulations, a pair of residues is defined to be in contact when the distance between their C_β atoms (C_α in the case of glycines) is less than 8Å. For each combination of α and λ, the proportion of predicted coevolving pairs that are within 8Å is calculated. Figures 3.9 and 3.10 display these proportions for each alignment. The plots for the alignments are displayed in ascending sequence order.

The proportion of predicted coevolving residues in contact varies between the alignments. However, for all alignments and values of α, the proportion is highest for low values of λ in the regularisation path. Alignments PF00193 and PF00366 have the highest proportion of predicted coevolving residues in contact for some combinations of α and λ. These combinations result in a proportion of 1, suggesting that our method is successful for these alignments as all predicted residue pairs are in contact in three-dimensional space. These alignments do not have similar numbers of sequences or columns.

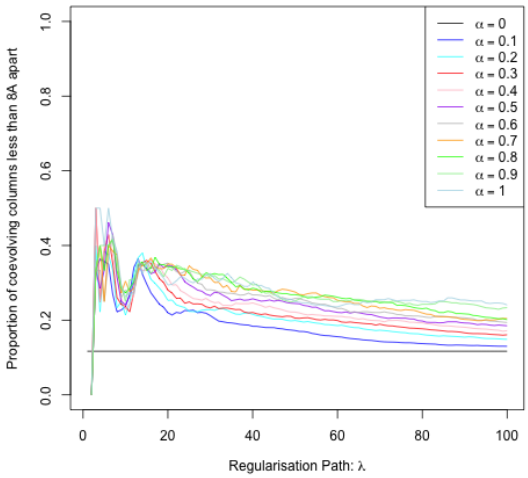
Alignments PF00029 and PF00105 have the lowest proportion of predicted coevolving residues in contact, for all values of α and λ. The largest proportion of successfully predicted contacts is around 40%. PF00029 consists of the fewest sequences, while PF00105 has the most sequences. The two alignments also have different numbers of alignment columns.



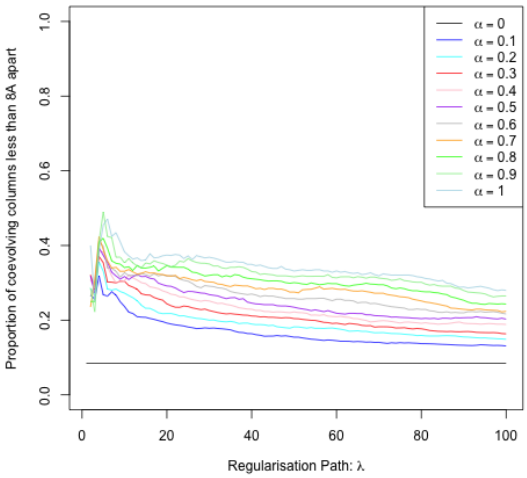
(a) PF00029



(b) PF00193



(c) PF00157



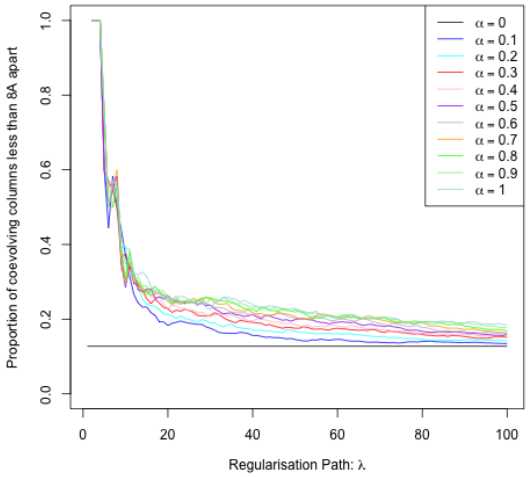
(d) PF00243

Figure 3.9: Proportion of predicted coevolving residue pairs less than 8Å apart in three-dimensional space, for each combination of α and λ. Each plot corresponds to a different Pfam alignment.

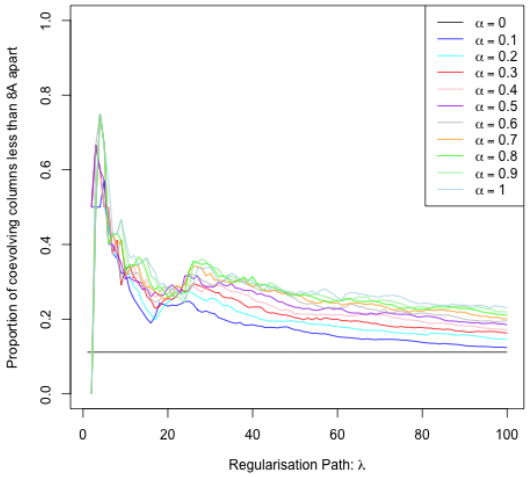
3.4 Discussion

We have introduced a novel method to identify coevolving pairs of columns in protein multiple sequence alignments. Compared to the method of Sreekumar *et al.* (2011) we fit just one model, rather than one model for every column in the alignment.

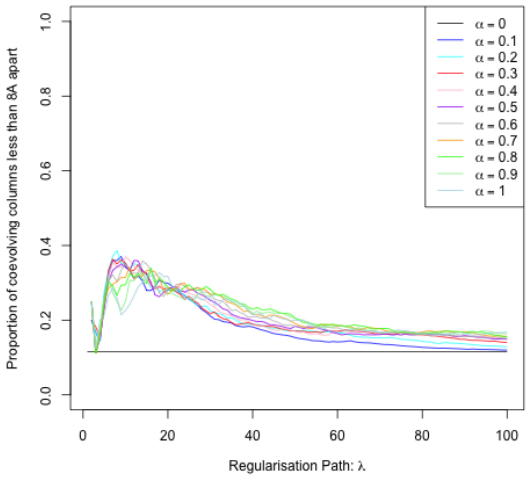
We have shown that our model is successful when applied to small simulated datasets. For alignments with 30 columns and 20 sequences, we are able to success-



(a) PF00366



(b) PF00276



(c) PF00105

Figure 3.10: Proportion of predicted coevolving residue pairs less than 8Å apart in three-dimensional space, for each combination of α and λ. Each plot corresponds to a different Pfam alignment.

fully identify the known coevolving columns in 85% of datasets. When the number of sequences is increased to 100, the model successfully identifies the coevolving columns in all of the datasets. We have shown that our model is capable of dealing with various levels of noise added to the coevolving columns and varying numbers of coevolving pairs.

In addition, our model is successful when applied to datasets with 50 columns and 50–100 sequences. We show that there are many optimal combinations of α and

λ, common to all of the simulated datasets with more than 20 sequences.

When applied to real datasets of Pfam alignments, we obtain mixed results. All of the residues identified as coevolving by our model are found to be in contact for two of the alignments, and a third alignment is successful with 80% of the predictions being in contact. The remaining four alignments report that 40-50% of the predictions are in contact in three-dimensional space.

It would be interesting to determine whether these predicted contacts correspond to short, medium or long range contacts, as defined in Section 3.1.1. This could then be extended to analyse the proportion of predicted short, medium and long range residues that are in contact.

Performance is lower for the biological datasets in comparison to the simulated alignments, suggesting that our simulations may not be accurately representing real biological data. We do not explore here simulated datasets with more than 100 sequences, and multiple coevolutionary mutation events between a pair of columns. Potential future work could extend our simulations to be more representative of biological datasets by introducing more noise to the coevolving columns, and exploring the effect of multiple mutations between coevolving pairs of columns. Other selection procedures for λ could also be explored, for example the sum of coefficients approach used by Sreekumar *et al.* (2011).

Our model may be identifying columns that are truly coevolving, however they are not in close proximity in three-dimensional space for the alignments that had lower contact percentages. To guide the model, or refine the output to predict those residues in contact, additional information about the amino acid physiochemical properties could be used. Further work could also include comparing our predictions to those of other methods. In Section 1.3.2 we see that the leading method has an average precision of 27%.

Chapter 4

Analysing cospeciation in tritrophic ecology using electrical circuit theory

4.1 Introduction

We introduce a new method to test efficiently for cospeciation in tritrophic systems. Our method is a development of the correlation statistic proposed by Hommola *et al.* (2009). We relate Hommola *et al.*'s (2009) method to higher-order systems by applying methods from electrical circuit theory (Curtis *et al.*, 2000). We use these methods to reduce higher order systems into two vectors of electrically equivalent patristic distances that can be compared using Spearman's rank correlation coefficient. The equivalent patristic distances take into account the information contained in the connection to the third phylogenetic tree. We use a sophisticated permutation scheme that weights interactions between two trophic layers based on their connection to the third layer in the system.

As far as we know, Mramba *et al.* (2013) have developed the only method for assessing cospeciation at the tritrophic level. Our method has several advantages compared to the method of Mramba *et al.* (2013). We do not require triangular interactions to connect the three phylogenetic trees and an easily interpreted p-value is obtained in one step. Another advantage of our method is the scope for generalisation to higher order systems and phylogenetic networks.

The performance of our method is compared to the methods of Hommola *et al.* (2009) and Mramba *et al.* (2013) at the bitrophic and tritrophic levels, respectively. This was achieved by evaluating type I error and statistical power. The results in

Section 4.4 show that our method produces unbiased p-values and has greater power overall at both trophic levels. Our method was successfully applied to a dataset of leaf-mining moths, parasitoid wasps and host plants (Lopez-Vaamonde *et al.*, 2005), in Section 4.4.4, at both the bitrophic and tritrophic levels.

4.1.1 Motivation

The study of host-parasite coevolution originated with the work of Von Ihering, who was the first to recognise predictable associations among hosts and their parasites (Klassen, 1992). Parasites and their hosts generally form tight ecological associations and as such it has long been assumed that the speciation of parasites is largely dependent on the speciation of their hosts (Legendre *et al.*, 2002). However, cospeciation is not the only process that occurs, and thus host-parasite phylogenies are rarely exact mirror images. The parasite may switch lineages, speciate independently, go extinct, fail to colonise all descendants of a speciating host lineage, or fail to speciate when the host does (Page, 2003).

Figure 4.1 displays a simple example bitrophic system consisting of Tree X, Tree Y and the interactions between their leaf nodes. We mainly focus on parasitic interactions, however other types of ecological interaction exist. These interactions may have arisen through symbiosis, mutualism, habitat or feeding relationships.

There has been extensive exploration into the bitrophic interactions observed between hosts and their parasites, and between plants and specialised herbivorous insects (Forister & Feldman, 2011). As a result, many statistical tests have been developed to assess cospeciation in these systems (Hommola *et al.*, 2009; Huelsenbeck

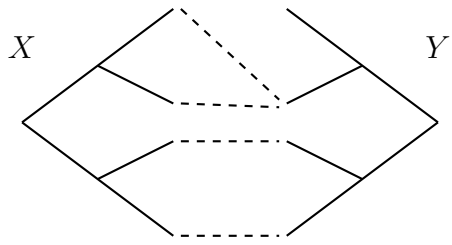


Figure 4.1: An example bitrophic system consisting of two phylogenetic trees and their ecological interactions. The solid lines present the branches of the phylogenetic trees and the dashed lines between the tips of Tree X and Tree Y represent the interactions between them.

et al., 2000; Legendre *et al.*, 2002; Mantel, 1967; Page, 1996). However, shared evolutionary histories have been observed across more than two trophic levels (Forister & Feldman, 2011). For example, tritrophic interactions were observed between hosts, parasites and host plants (Ahmad *et al.*, 2004; Micha *et al.*, 2000). Recently, it was discovered that tritrophic coevolution exists between flies and parasitic nematodes on Mytaceae host plants (Nelson *et al.*, 2014).

Mramba *et al.* (2013) developed the only statistical method we are aware of to test cospeciation in tritrophic systems. However, Mramba *et al.*'s (2013) test requires the interactions between three phylogenies to form triangles to be able to compare patristic distances on the three trees. This is often not the case in naturally occurring tritrophic systems, and thus interactions that do not form triangles are discarded along with the information they provide. We propose an improved method which can accommodate any type of interaction. To draw conclusions about where cospeciation occurs within a tritrophic system, Mramba *et al.*'s (2013) method necessitates the permutation of every pairwise combination of three trees; that is, 7 randomisations and, correspondingly, 7 p-values. By contrast, our more efficient method requires the use of one sophisticated permutation scheme, resulting in one easily interpreted p-value.

Many bitrophic tests (Hommola *et al.*, 2009; Legendre *et al.*, 2002; Mantel, 1967) and Mramba *et al.*'s (2013) tritrophic test are limited to systems consisting of phylogenetic trees. Our method has the scope for generalisation to higher order systems and the application to phylogenetic networks.

4.1.2 Existing Methodology

There are many methods available to test whether bitrophic systems display evidence of cospeciation as reflected by congruent phylogenies and corresponding interactions. The method of Hommola *et al.* (2009) outperforms the Mantel test (Mantel, 1967) and ParaFit (Legendre *et al.*, 2002). Hommola *et al.*'s (2009) statistical test is a development of the Mantel test that overcomes the one-to-one interaction constraint. The Mantel test can only accommodate one-to-one interaction patterns between two phylogenetic trees. It assumes that a species on Tree X can only interact with one species on Tree Y. However one-to-one interaction patterns rarely occur naturally. To manage many-to-one interactions, the generalist species are either discarded or replicated. Both of these solutions introduce bias (Hommola *et al.*, 2009). The method of Hommola *et al.* (2009) calculates the patristic distances on each tree between each pair of interactions in a bitrophic system. A test statistic is calculated

as the Pearson's correlation coefficient between the distances on Tree X and the distances on Tree Y. A high correlation indicates cospeciation between the two trees, while a low correlation indicates that there is no cospeciation. To calculate the significance of this statistic, the tip labels of the trees are permuted many times and the correlation recomputed for each. A p-value is calculated as the proportion of times the permuted correlation is larger than the observed correlation.

Mramba *et al.* (2013) extend the permutation test developed by Hommola *et al.* (2009) to systems with three phylogenies, by computing patristic distances and using three-way interaction matrices. To extend the interaction matrices between each pair of trees into one three-way interaction matrix only interactions that form triangles between the three trees are considered. The patristic distances on each tree between each pair of interaction triangles is calculated, resulting in three vectors of patristic distances. These vectors are combined to form the columns of a distance matrix D. Define λ_{obs} to be the dominant eigenvalue of the covariance matrix of D. Similarly to the bitrophic case, if there is cospeciation somewhere in the system we would expect the columns of D to be correlated. In this case, λ_{obs} would be large relative to the other eigenvalues. Therefore λ_{obs} is used as a statistic to test the following hypothesis:

H_0: Trees X, Y and Z have evolved independently.

H_1: Cospeciation is present somewhere in the X, Y, Z system.

The dominant eigenvalue test statistic is unable to determine where the coevolution in a tritrophic system has occurred. Therefore, additional test statistics are calculated to test the following alternative hypotheses

$H_{YZ.X}$: Cospeciation between Trees Y and Z is not due entirely to their common cospeciation with Tree X.

$H_{XZ.Y}$: Cospeciation between Trees X and Z is not due entirely to their common cospeciation with Tree Y.

$H_{XY.Z}$: Cospeciation between Trees X and Y is not due entirely to their common cospeciation with Tree Z.

Partial correlation test statistics, $r^{\text{obs}}_{yz.x}$, $r^{\text{obs}}_{xz.y}$, and $r^{\text{obs}}_{xy.z}$, are used to distinguish between these hypotheses. For example, $r^{\text{obs}}_{yz.x}$ is the partial correlation between the patristic distances for Tree Y and the patristic distances for Tree Z, when their

Permutation	P_λ significant	$P_{xy.z}$ significant	$P_{xz.y}$ significant	$P_{yz.x}$ significant
X	X involved in cospeciation	X and Y cospeciate	X and Z cospeciate	-
Y	Y involved in cospeciation	X and Y cospeciate	-	Y and Z cospeciate
Z	Z involved in cospeciation	-	X and Z cospeciate	Y and Z cospeciate
XY	Cospeciation occurs somewhere in the system			
XZ				
YZ				
XYZ				

Table 4.1: Basic interpretation of the interaction between the possible permutations of the tritrophic system and the p-values of the method of Mramba *et al.* (2013).

correlations with the patristic distances on Tree X are controlled for. Formally, the partial correlations, $r^{\text{obs}}_{yz.x}$, $r^{\text{obs}}_{xz.y}$, and $r^{\text{obs}}_{xy.z}$, are defined by

$$r^{\text{obs}}_{yz.x} = \frac{r_{yz} - r_{xy}r_{xz}}{\sqrt{\left(1 - r^2_{xy}\right)\left(1 - r^2_{xz}\right)}}$$
$$r^{\text{obs}}_{xz.y} = \frac{r_{xz} - r_{xy}r_{yz}}{\sqrt{\left(1 - r^2_{xy}\right)\left(1 - r^2_{yz}\right)}}$$
$$r^{\text{obs}}_{xy.z} = \frac{r_{xy} - r_{xz}r_{yz}}{\sqrt{\left(1 - r^2_{xz}\right)\left(1 - r^2_{yz}\right)}},$$

where r_{yz} is the Pearson's correlation coefficient between the patristic distances on Tree Y and the patristic distances on Tree Z. Similarly for r_{xz} and r_{xy}. To carry out the test, first assess whether H_0 can be rejected in favour of H_1. Only if H_1 is rejected can the other alternatives be considered. The p-values for each test statistic, denoted P_λ, $P_{xy.z}$,$P_{xz.y}$ and $P_{yz.x}$, are calculated using the same permutation method as Hommola *et al.* (2009), where the tip labels of the trees are randomised. The choice of which tree, or combination of trees, to randomise determines the alternative hypothesis being tested. Randomising the tips of all three trees tests H_0 against H_1. If this p-value is significant, then further permutations are performed to determine where the cospeciation has occurred. If Tree X is randomised, the corresponding p-values test whether Tree X is involved in cospeciation above any cospeciation between Trees Y and Z. If the tips of two trees are randomised, then the resulting p-values investigate their interaction with the third tree in the system. A simple guide to interpreting the relationship between the different p-values and the different permutation methods is given in Table 4.1.

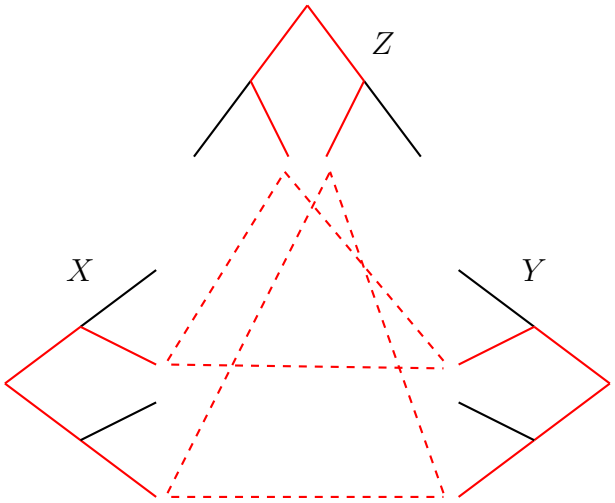


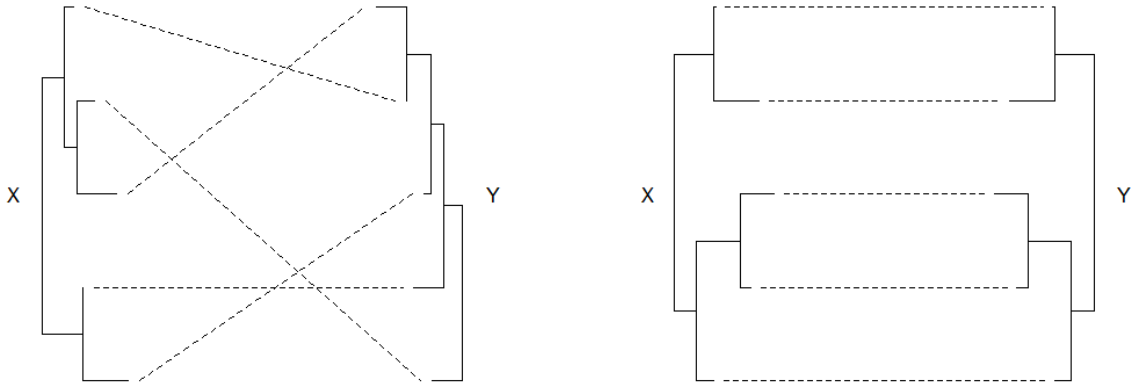
Figure 4.2: An example tritrophic system with interactions between Trees X, Y and Z forming two triangles. The red dashed lines between the tips of the trees represent interactions. The red lines on the trees indicate the branches whose lengths would be summed to produce the patristic distances on Trees X, Y and Z as a result of comparing the two triangles of interactions.

4.2 Methods and Materials

The methods of Hommola *et al.* (2009) and Mramba *et al.* (2013) calculate the patristic distance on each tree between each pair of interactions. In a bitrophic system the calculation of patristic distances is trivial. However, in a tritrophic system, there is no obvious analogue for patristic distances. Patristic distances on the three trees can only be compared by finding pairs of interaction triangles in the system, as displayed in Figure 4.2.

Another situation in which patristic distances are difficult to calculate is when the system involves a phylogenetic network, as there may be more than one path between two leaf nodes.

To overcome these problems we consider electrical networks as an analogy for the network of phylogenetic trees. We utilise electrical circuit theory to develop a method that can be generalised to test cospeciation hypotheses in both bitrophic and tritrophic systems. We apply the forward problem in electrical networks to the system of phylogenetic trees to obtain electrically equivalent distances between a set of carefully placed nodes. Nodes are defined to be points where two or more elements meet. In a circuit the elements are wires and in the case of a phylogenetic tree, the elements are the branches and interactions.



(a) (b)

Figure 4.3: Randomly generated systems consistent with the bitrophic hypotheses. The dashed lines represent the interactions between the leaf nodes of the two phylogenetic trees. (a) System consistent with the null hypothesis. Both trees and the interactions between them have been independently randomly generated. (b) System consistent with the alternative hypothesis. The trees are identical and interactions are placed at corresponding positions on the two trees.

4.2.1 Hypotheses

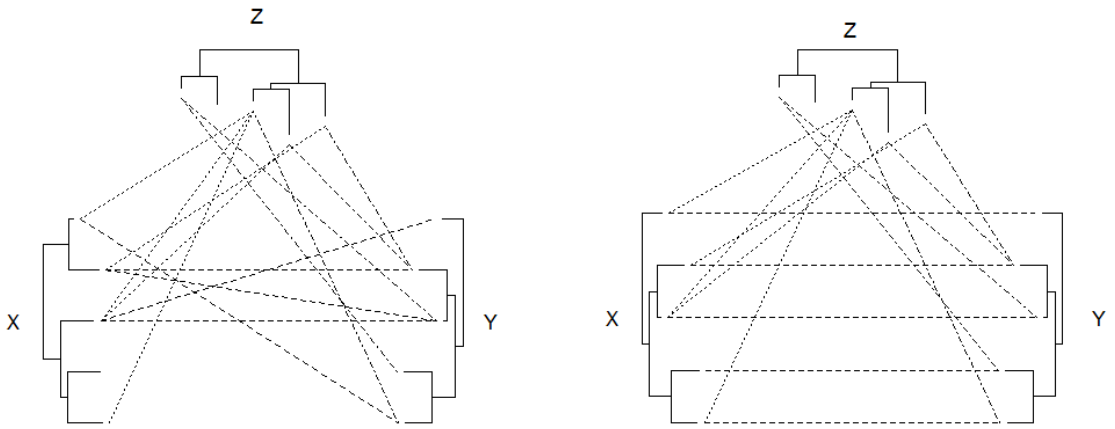
In the bitrophic case we consider two phylogenetic trees, X and Y, and the interactions between their tips. We are interested in the following hypotheses:

H_0: The phylogeny of Tree X and the phylogeny of Tree Y are unrelated, implying no cospeciation between X and Y;

H_1: The phylogeny of Tree X and the phylogeny of Tree Y are related, implying cospeciation between the trees.

Figure 4.3 displays systems generated under the extremes of the above hypothesis. The system in Figure 4.3a is comprised of randomly generated trees with random interactions consistent with the null hypothesis of no cospeciation. In contrast, the system in Figure 4.3b consists of identical trees with corresponding interactions, demonstrating the extreme of perfect cospeciation.

We do not simply want to know whether cospeciation exists somewhere within a tritrophic system. Rather, we are interested in how the cospeciation is driven. In the tritrophic case we consider three phylogenetic trees, X, Y and Z, and the ecological interactions between each pair of trees. We are interested in the following hypotheses:



(a) (b)

Figure 4.4: Randomly generated systems consistent with the tritrophic hypotheses. The dashed lines represent the interactions between the leaf nodes of the three phylogenetic trees. (a) System consistent with the null hypothesis. All three trees and the interactions between them have been independently randomly generated. (b) System consistent with the alternative hypothesis. Two of the trees, X and Y, are identical with interactions placed at corresponding positions on the two trees. The third tree, Z, is independently generated and has random interactions with the other two trees.

H_0: There is no more cospeciation between Trees X and Y than can be explained by the cospeciation between Trees X and Z, and between Trees Y and Z, suggesting that Tree Z is driving the cospeciation in the system;

H_1: There is more cospeciation between Trees X and Y than is due to the cospeciation between Trees X and Z, and Trees Y and Z.

Figure 4.4 displays systems generated under the extremes of the tritrophic hypotheses. The system in Figure 4.4a is comprised of three randomly generated trees with random interactions between them. Clearly, there is no cospeciation between Trees X and Y; none of the trees appear to be cospeciating on a pairwise level. Systems where Tree Z is driving the cospeciation between Trees X and Y would also be consistent with the null hypothesis. The system in Figure 4.4b consists of identical Trees X and Y with corresponding interactions. There is no cospeciation between these trees and Tree Z, so Tree Z does not drive the cospeciation between Trees X and Y.

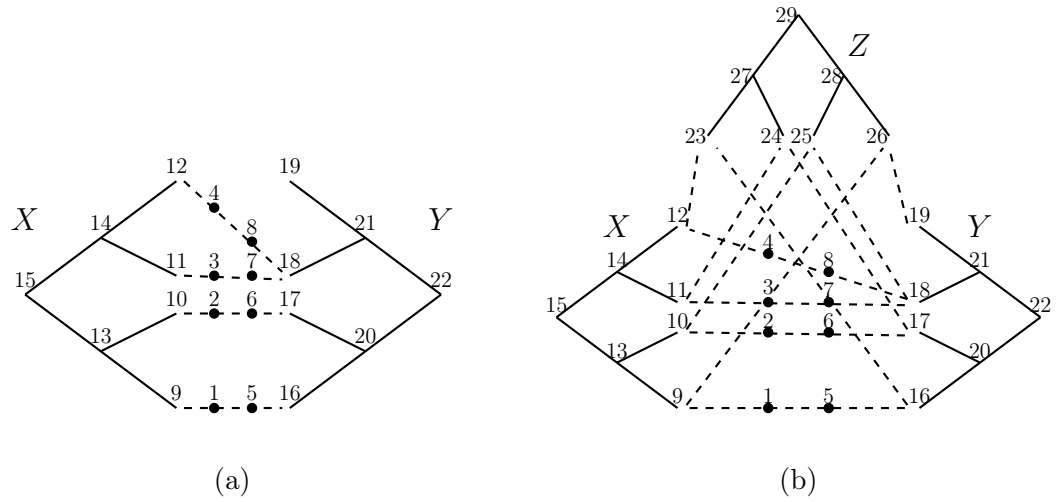
(a) (b)

Figure 4.5: Schematic diagram of the forward problem in electrical networks. (a) Example electrical circuit with nodes displayed as black circles that are connected by wires. (b) Black box containing the circuit in (a) with four nodes exposed, the wiring of the circuit inside the black box is unknown.

4.2.2 Correlation statistic calculated from resolved distances

Suppose we have an electrical circuit where the conductance and topology of the connections is known, as displayed in Figure 4.5a. Suppose we take a subset of nodes, nodes 1 to 4 in Figure 4.5a, and define these nodes to be external; all other nodes are internal. If we impose a voltage on the external nodes, we can calculate the resulting current at these nodes. We now suppose that the circuit, excluding the external nodes, is inside a black box, as displayed in Figure 4.5b. We no longer know how the internal nodes inside the box are connected, or the conductance on the original connections; we only have the conductances on direct connections between the external nodes. The *forward problem* assumes that we know how the circuit is connected, and the conductance on each connection. The conductance on the direct connections between the external nodes is then calculated using this information. The *inverse problem* is to obtain the full circuit from the circuit in the black box where only conductances on direct connections between the external nodes are known. The conductances of each connection in the full circuit is calculated from measurements of voltages and currents at the external nodes in the black box circuit (Curtis *et al.*, 2000).

We use the *forward problem* in electrical networks to calculate the conductance on direct connections between each pair of interactions between Tree X and Tree Y. In the tritrophic case these conductances will take into account how Tree X and Tree Y are connected to Tree Z. These conductances can then be used to calculate distances. In the bitrophic case, for each pair of interactions we will have a distance that corresponds to Tree X, and a distance that corresponds to Tree Y. In the tritrophic case these distances will take into account the connections between Trees X and



(a) (b)

Figure 4.6: External node placement in bitrophic and tritrophic systems. External nodes are represented by black dots. Each node in the system has been numbered. (a) External node placement in a bitrophic system. (b) External node placement in a tritrophic system. The Trees X, Y and Z correspond to Trees X, Y and Z as described in the tritrophic hypotheses.

Y with Tree Z. We calculate Spearman's rank correlation coefficient between the resulting vectors of Tree X and Tree Y distances. For a cospeciated system we expect there to be a correlation between the distance on Tree X and the distance on Tree Y associated with each pair of interactions.

To obtain direct connections between the interactions for Tree X and Tree Y, we need an external node at each end of every interaction. We introduce two artificial nodes on each interaction, dividing the interactions into three connections as displayed in Figure 4.6a. We later show in Section 4.3 that the statistic does not depend on the middle connection between the external nodes. The artificial nodes are the external nodes and every other node in the system is internal. In a tritrophic system the artificial external nodes are introduced on the interactions between Trees X and Y, as shown in Figure 4.6b.

Our test statistic is derived by converting the phylogenetic distances on the branches and interactions of the phylogenetic trees into conductances and calculating a response matrix for the system. The conductance between two nodes i and j, which are directly connected by a single branch, is calculated as

$$\gamma_{i,j} = \frac{1}{d_{i,j}}, \tag{4.1}$$

where $d_{i,j}$ is the phylogenetic distance between nodes i and j and $\gamma_{i,j} = 0$ if nodes i

and j are not directly connected by a single branch. The interactions between the phylogenetic trees do not typically have distances, therefore we assign each of the three connections that make up an interaction a constant distance, ϵ. In our analysis we chose ϵ such that the branches of the phylogenetic trees and the interactions are weighted equally. However, it may be of interest to give the branches more or less weight than the interactions. For example, the interactions may be given different weights based on how strong the association is between the species in nature. The interactions can also be weighted differently to represent how likely they are to exist.

Given an interacting system of phylogenetic trees consisting of m nodes in total, the Kirchhoff matrix, K, is an $m \times m$ Laplacian matrix, assembled using the conductances between nodes connected by a single branch. The non diagonal elements of K are given by the negative conductance between each pair of nodes. The diagonal elements of K are calculated such that the rows and columns of the matrix sum to zero. The $(i,j)^{\text{th}}$ element of K is thus given by

$$k_{i,j} = \begin{cases} -\gamma_{i,j} & \text{if } i \neq j \\ \sum_{j \neq i} \gamma_{i,j} & \text{if } i = j. \end{cases} \tag{4.2}$$

The Kirchhoff matrix has the following interpretation. If u is defined to be a vector of voltages applied to each node of the network, then $\phi = Ku$ is the resulting vector of current flowing into the network at each node. If a voltage of one unit is applied to node j and a voltage of zero is applied to every other node, then $k_{i,j}$ is the current into the network at each node i. Thus column j of K gives the values of the currents into the network at nodes $i = 1, \ldots, m$.

Rearranging the Kirchhoff matrix in terms of the internal and external nodes of the system, where the external nodes are the nodes on the interactions and all of the tree nodes are internal, partitions the matrix into four submatrices.

$$K = \begin{array}{c} \\ \text{E} \\ \text{I} \end{array} \begin{array}{c} \begin{array}{cc} \text{E} & \text{I} \end{array} \\ \begin{pmatrix} A & B \\ B^T & D \end{pmatrix} \end{array}, \tag{4.3}$$

where E and I correspond to the external and internal nodes respectively, and T denotes transposition.

A response matrix, Λ_γ, is obtained by calculating the Schur complement in K of the square submatrix corresponding to the internal nodes of the network, D:

$$\Lambda_\gamma = A - BD^{-1}B^T.$$

This response matrix is simply a Kirchhoff matrix calculated for an electrically equivalent system without internal nodes, and only direct connections between the external nodes. The system is electrically equivalent because if the same voltages are applied to the external nodes in both systems, then the same current will be induced as a result.

The response matrix contains the negative conductance on each pairwise connection between the external nodes. The distances between the external nodes in the collapsed system are obtained by reversing Equations (4.1) and (4.2). We define D^* to be the resulting distance matrix, with $(i,j)^{th}$ element given by

$$d^*_{i,j} = \begin{cases} -\frac{1}{(\Lambda_\gamma)_{i,j}} & \text{if } i \neq j \\ 0 & \text{if } i = j, \end{cases}$$

where $(\Lambda_\gamma)_{i,j}$ is the $(i,j)^{th}$ element of Λ_γ.

The distance matrix can be partitioned in terms of the external nodes corresponding to Tree X; E_X, and the external nodes corresponding to Tree Y; E_Y, as follows:

$$D^* = \begin{array}{c} \\ E_X \\ E_Y \end{array} \begin{array}{c} \begin{array}{cc} E_X & E_Y \end{array} \\ \begin{pmatrix} D_X & D_{XY} \\ D_{XY}^T & D_Y \end{pmatrix} \end{array}, \tag{4.4}$$

where D_X and D_Y are submatrices containing the distances between each pair of external nodes corresponding to Tree X and Tree Y respectively. D_{XY} is a submatrix containing the distances between Tree X and Tree Y. In the tritrophic case, these distances will also take into account the connection with Tree Z. Figure 4.7 displays the connections corresponding to the distances contained in D^* for the systems in Figure 4.6.

Our statistic is obtained by calculating Spearman's correlation coefficient, r_{obs}, between the upper triangle of D_X and D_Y. We use a rank correlation because the response matrix calculations produce large distances when there are extreme interactions between the trees.

4.2.3 Permutations

To determine whether our value of r_{obs} is statistically significant we propose a permutation scheme that simulates under our null hypotheses.

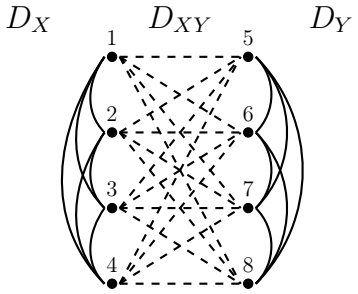


Figure 4.7: Connections contained in D^* for the systems displayed in Figure 4.6. The external nodes are represented by black dots and numbered consistently with Figure 4.6. The internal nodes have been integrated out by the response matrix calculations.

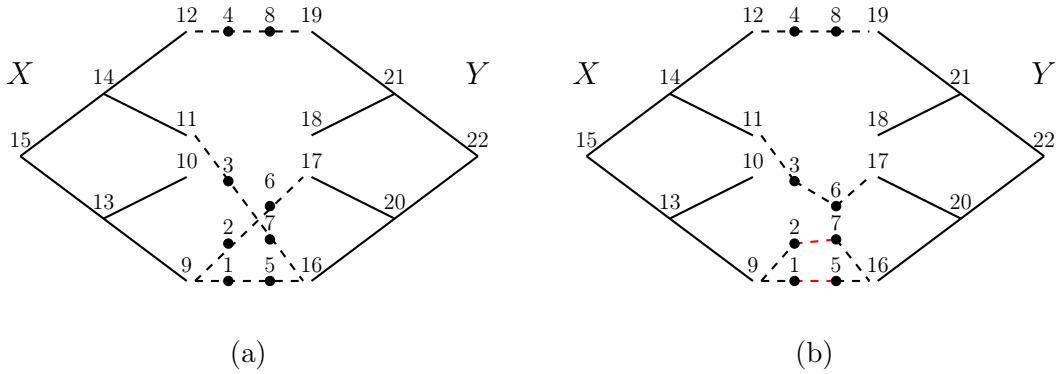
Bitrophic Randomisation Scheme

In a bitrophic system the connections between the external nodes are sampled with equal probability. Consider the simple example system in Figure 4.8. The system in Figure 4.8b displays one example of a possible randomisation of the connections in Figure 4.8a. Permutations of the connection between the external nodes that result in overlapping interactions, such as those displayed in red in Figure 4.8b, are rejected. This is equivalent to simply randomising the existing connections between the external nodes. Randomising in this way preserves the many to one nature of the interactions, however not all of the interactions between the two trees are possible due to the placement of the external nodes on the interactions. For example, in Figure 4.8a, node 9 will always have 2 interactions and node 18 will have none. That is, nodes on the trees without interactions are essentially removed.

Tritrophic Randomisation Scheme

To determine whether our observed statistic is statistically significant, we propose a randomisation scheme that simulates interactions consistent with the null hypothesis. To do this we use a weighted randomisation scheme that samples the connections between the external nodes that connect Trees X and Y.

The response matrix for the system of phylogenetic trees is simply a Kirchhoff matrix calculated only for the external nodes of the electrically equivalent system with the internal nodes integrated out. Therefore the response matrix infers a connection between each pair of external nodes with different conductivities based on the original connections between the trees. These conductivities are obtained from the same partitions of the response matrix as the connections in D_{XY} in Equation (4.4).



(a) (b)

Figure 4.8: A simple example system illustrating a possible permutation arrangement in a bitrophic system. (a) Observed system. (b) The system in (a) where the connections between the external nodes have been randomised to produce new interactions. The red connections display where overlapping interactions have been produced between nodes 9 and 16.

The conductances (analogous to evolutionary similarity) on these connections are used as weights to sample the connections between the external nodes that connect Trees X and Y. Connections consistent with H_0 have a greater probability of being sampled. To obtain these weights we recalculate the response matrix for the system with the direct connections between the external nodes removed, as displayed in Figure 4.9. To do this, these connections are simply not entered into the submatrix A in the Kirchhoff matrix in Equation (4.3). The external nodes are still indirectly connected via Tree Z, representing the joint cospeciation of Trees X and Y with Tree Z. To randomise the tritrophic system consistent with the null hypothesis, we sample the connections between the external nodes with probability proportional to their conductance in the recalculated response matrix.

There are two practical considerations that must be taken into account when sampling the connections. Firstly, the connections must be sampled such that many to one interactions between two external nodes are avoided, as this would correspond to a system where there are interactions between the interactions. Secondly, permutations involving overlapping interactions are rejected, as in the bitrophic case. Any randomisations that do not satisfy these criteria are rejected.

Calculating p-values

Similarly to the methods of Legendre *et al.* (2002), Hommola *et al.* (2009) and Mramba *et al.* (2013), we propose a permutation test to determine whether the value of r_{obs} is statistically significant. A p-value, p, is obtained for r_{obs} by simulating N

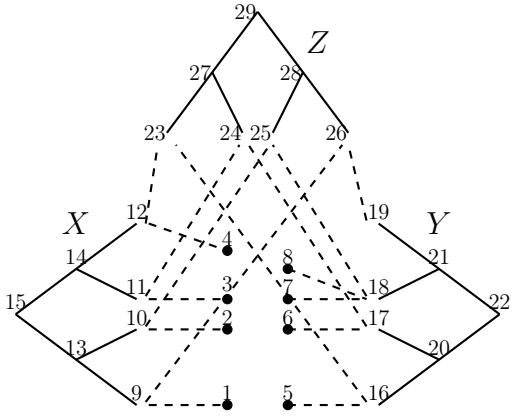


Figure 4.9: The simple example tritrophic system from Figure 4.6b with the connections between the external nodes removed.

systems under H_0 as described in the previous section, then calculating

$$p = \frac{1}{N} \sum_{i=1}^{N} I(r_i > r_{\text{obs}}),$$

where r_i is the test statistic calculated for the i^{th} randomisation and $I(r_i > r_{\text{obs}})$ is an indicator function taking the value 1 if r_i is greater r_{obs} and 0 otherwise. If $r_i = r_{\text{obs}}$, the indicator function takes the value 1 with probability 0.5. If $p \leq \alpha$ we reject H_0 at the $100\alpha\%$ significance level.

4.3 Response Matrix Calculations

It can be easily shown that our method is equivalent to calculating two separate response matrices, one for each side of the system (see Appendix E.3). We show here that our statistic does not depend on the direct connections between external nodes.

Recall Equation (4.3) where the Kirchhoff matrix, K, is partitioned in terms of the external and internal nodes of the system to produce the following submatrices

$$K = \begin{array}{c} \\ \text{E} \\ \text{I} \end{array} \begin{array}{c} \begin{array}{cc} \text{E} & \text{I} \end{array} \\ \begin{pmatrix} A & B \\ B^T & D \end{pmatrix} \end{array},$$

The internal nodes, I, can be partitioned in terms of leaf nodes, L, and ancestral nodes, C, as displayed in Figure 4.10. Each of these can then be partitioned further

in terms of Trees X and Y. This results in the following partitions: E_X, E_Y, L_X, L_Y, C_X and C_Y. Each of the submatrices in Equation (4.3) can therefore be partitioned further into submatrices, as follows.

The submatrix B in Equation (4.3) is partitioned into eight submatrices:

$$B = \begin{array}{c} \\ E_X \\ E_Y \end{array} \begin{array}{c} \begin{array}{cccc} L_X & L_Y & C_X & C_Y \end{array} \\ \begin{pmatrix} -\frac{I_X}{\epsilon} & 0 & 0 & 0 \\ 0 & -\frac{I_Y}{\epsilon} & 0 & 0 \end{pmatrix} \end{array}, \tag{4.5}$$

where I_X and I_Y are binary matrices containing the connections between between the external nodes and the leaf nodes on each tree respectively. Each connection has conductance $\frac{1}{\epsilon}$.

The submatrix A in Equation (4.3) is partitioned into four submatrices:

$$A = \begin{array}{c} \\ E_X \\ E_Y \end{array} \begin{array}{c} \begin{array}{cc} E_X & E_Y \end{array} \\ \begin{pmatrix} \Delta_{E_X} & -\frac{I}{\delta} \\ -\frac{I}{\delta} & \Delta_{E_Y} \end{pmatrix} \end{array},$$

where I is the identity matrix because each node of E_X is connected to exactly one node on E_Y. This connection has conductance $\frac{1}{\delta}$, where δ is the distance on

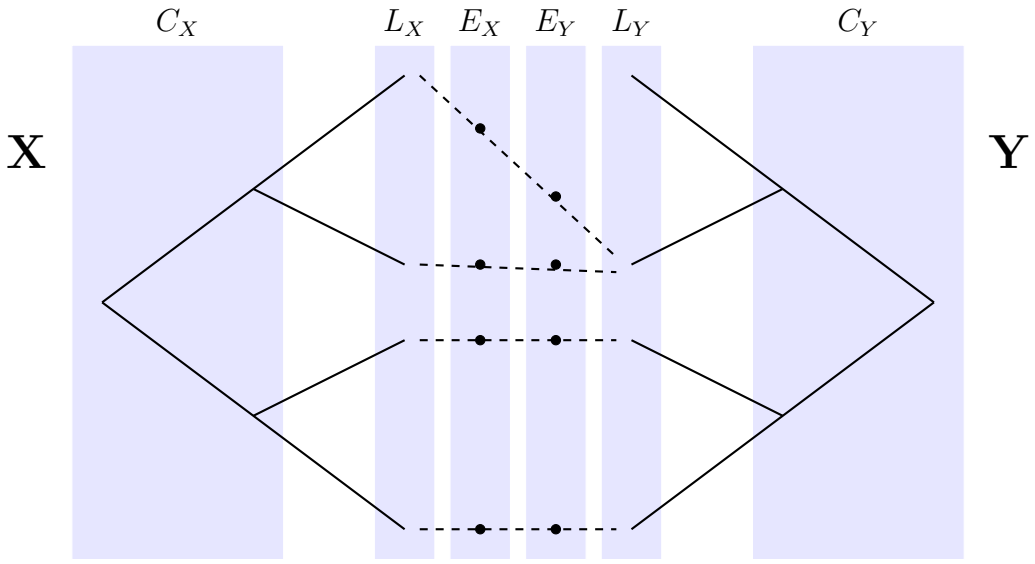


Figure 4.10: An example of how the simple bitrophic system in Figure 4.1 is partitioned in terms of the external nodes on each tree, E_X and E_Y, leaf nodes, L_X and L_Y, and ancestral nodes, C_X and C_Y.

the direct connections between the external nodes. The remaining entries, Δ_{E_X} and Δ_{E_Y}, are diagonal matrices that represent the unknown conductances on the diagonal of K.

The matrices on the diagonal of A are obtained by recalling that the rows and columns of a Kirchhoff matrix are set up to sum to zero. Therefore we have the following constraints using the submatrices in B:

$$\begin{aligned} \Delta_{E_X}\mathbf{1} - \frac{I}{\delta}\mathbf{1} - \frac{I_X}{\epsilon}\mathbf{1} &= \mathbf{0}, \\ \frac{-I}{\delta}\mathbf{1} + \Delta_{E_Y}\mathbf{1} - \frac{I_Y}{\epsilon}\mathbf{1} &= \mathbf{0}, \end{aligned} \tag{4.6}$$

where $\mathbf{0}$ and $\mathbf{1}$ represent column vectors of zero's and one's respectively. Each external node is connected to only one leaf node. Therefore each row of I_X and I_Y will contain exactly one 1 and thus $I_X\mathbf{1} = \mathbf{1}$ and $I_Y\mathbf{1} = \mathbf{1}$. It is also clear that $I\mathbf{1} = \mathbf{1}$. Therefore Equation (4.6) can be simplified and rearranged to give

$$\Delta_{E_X}\mathbf{1} = \Delta_{E_Y}\mathbf{1} = \left(\frac{1}{\delta} + \frac{1}{\epsilon}\right)\mathbf{1}.$$

From this we can fill in the diagonal elements of A as follows

$$A = \begin{array}{c} \\ E_X \\ E_Y \end{array} \begin{array}{c} \begin{array}{cc} E_X & E_Y \end{array} \\ \begin{pmatrix} (\frac{1}{\delta} + \frac{1}{\epsilon})I & -\frac{I}{\delta} \\ -\frac{I}{\delta} & (\frac{1}{\delta} + \frac{1}{\epsilon})I \end{pmatrix} \end{array}. \tag{4.7}$$

The final submatrix, D, is partitioned into sixteen submatrices:

$$D = \begin{array}{c} \\ L_X \\ L_Y \\ C_X \\ C_Y \end{array} \begin{array}{c} \begin{array}{cccc} L_X & L_Y & C_X & C_Y \end{array} \\ \begin{pmatrix} \Delta_{L_X} & 0 & -\Gamma_X & 0 \\ 0 & \Delta_{L_Y} & 0 & -\Gamma_Y \\ -\Gamma_X^T & 0 & \Gamma_{C_X} & 0 \\ 0 & -\Gamma_Y^T & 0 & \Gamma_{C_Y} \end{pmatrix} \end{array}, \tag{4.8}$$

where Γ_X and Γ_Y contain the conductances on the connections between the leaf nodes and the internal nodes on each tree. Δ_{L_X} and Δ_{L_Y} are diagonal matrices that represent the unknown conductances on the diagonal of K. Γ_{C_X} and Γ_{C_Y} are symmetric matrices containing the negative conductances between the internal nodes on each tree. The conductances on the diagonal are unknown.

The matrices on the diagonal of D are obtained by noting that the rows and columns of a Kirchhoff matrix are set up to sum to zero. Therefore we have the following constraints using the submatrices in B^T:

$$\begin{aligned}
-\frac{I_X^T}{\epsilon}\mathbf{1} + \Delta_{L_X}\mathbf{1} - \Gamma_X\mathbf{1} &= \mathbf{0},\\
-\frac{I_Y^T}{\epsilon}\mathbf{1} + \Delta_{L_Y}\mathbf{1} - \Gamma_Y\mathbf{1} &= \mathbf{0},\\
-\Gamma_X^T\mathbf{1} + \Gamma_{C_X}\mathbf{1} &= \mathbf{0},\\
-\Gamma_Y^T\mathbf{1} + \Gamma_{C_Y}\mathbf{1} &= \mathbf{0}.
\end{aligned}$$

Rearranging each constraint in terms of the matrices with unknown diagonal elements gives

$$\Delta_{L_X}\mathbf{1} = \frac{I_X^T}{\epsilon}\mathbf{1} + \Gamma_X\mathbf{1}, \tag{4.9}$$

$$\Delta_{L_Y}\mathbf{1} = \frac{I_Y^T}{\epsilon}\mathbf{1} + \Gamma_Y\mathbf{1}, \tag{4.10}$$

$$\Gamma_{C_X}\mathbf{1} = \Gamma_X^T\mathbf{1}, \tag{4.11}$$

$$\Gamma_{C_Y}\mathbf{1} = \Gamma_Y^T\mathbf{1}. \tag{4.12}$$

Equations (4.9) and (4.10) calculate the values on the diagonal of the diagonal matrices Δ_{L_X} and Δ_{L_Y}. The columns of I_X contain a 1 for every interaction connected to each leaf node on Tree X. Therefore, $\frac{I_X^T}{\epsilon}\mathbf{1}$ calculates the sum of the conductances on the interactions for each leaf node on Tree X. The rows of Γ_X correspond to the leaf nodes of Tree X. Each leaf node is connected to one ancestral node, and thus each row contains exactly one conductance, and the row sums will be the values of this conductance. Therefore, each diagonal element of Γ_{L_X} corresponds to a leaf node on Tree X, and is the sum of the conductances of branches and interactions connected to that leaf node. Similarly for Γ_{L_Y} and Tree Y.

Equations (4.11) and (4.12) are symmetric matrices containing the conductances between the ancestral nodes on Trees X and Y, respectively. To understand the diagonal component of these matrices, consider the entries of Γ_X. The columns of this matrix correspond to the ancestral nodes. Each ancestral node is connected to two leaf nodes. The column sums of Γ_X add up the conductance of the leaf node connections of each ancestral node. Therefore, each diagonal element of Γ_{C_X} corresponds to an ancestral node, and is the sum of the conductances on the connections with the ancestral node. Similarly for Tree Y.

Recall that the response matrix is calculated using the following equation

$$\Lambda_\gamma = A - BD^{-1}B^T.$$

We can work through these calculations using the partitioned matrices in Equations (4.5), (4.7) and (4.8) to obtain the partitioned response matrix in Equation (4.13). For details of these calculations see Appendix E.

$$\Lambda_\gamma = \begin{array}{c} \\ E_X \\ E_Y \end{array} \begin{array}{c} \begin{array}{cc} E_X & E_Y \end{array} \\ \begin{pmatrix} \left(\frac{1}{\delta}+\frac{1}{\epsilon}\right) I - \frac{1}{\epsilon^2}\left(I_X D_*^{11} I_X^T\right) & -\frac{I}{\delta} \\ -\frac{I}{\delta} & \left(\frac{1}{\delta}+\frac{1}{\epsilon}\right) I - \frac{1}{\epsilon^2}\left(I_Y D_*^{22} I_Y^T\right) \end{pmatrix} \end{array}, \tag{4.13}$$

where D_*^{11} and D_*^{22} are elements of the matrix inverse of D (see Appendix E.1), given by

$$\begin{aligned} D_*^{11} &= \frac{1}{\Delta_{L_X}} + \frac{1}{\Delta_{L_X}}\Gamma_X(\Gamma_{C_X} - \Gamma_X^T\frac{1}{\Delta_{L_X}}\Gamma_X)^{-1}(\Gamma_X^T\frac{1}{\Delta_{L_X}}), \\ D_*^{22} &= \frac{1}{\Delta_{L_Y}} + \frac{1}{\Delta_{L_Y}}\Gamma_Y(\Gamma_{C_Y} - \Gamma_Y^T\frac{1}{\Delta_{L_Y}}\Gamma_Y)^{-1}\Gamma_Y^T\frac{1}{\Delta_{L_Y}}. \end{aligned}$$

Our statistic is derived from the upper triangle of the highlighted matrices in Equation (4.13). Therefore our statistic only depends on the following matrices

$$\begin{aligned} &-\frac{1}{\epsilon^2} I_X D_*^{11} I_X^T, \\ &-\frac{1}{\epsilon^2} I_Y D_*^{22} I_Y^T. \end{aligned}$$

These matrices do not depend on δ, the distance on the direct connections between the external nodes, and therefore it follows that our statistic does not depend on the direct connections between E_X and E_Y. In fact, we have also shown that our method is equivalent to calculating the response matrices separately for each tree (see Appendix E.3).

4.4 Results

The performance of our method, at the bitrophic and tritrophic level, is analysed by investigating Type I error and assessing statistical power (see below). We compared the performance of our method to those proposed by Hommola *et al.* (2009) and

Mramba *et al.* (2013) at the relevant trophic level. In every simulation we set $\epsilon = 0.5$, the average branch length of the simulated trees.

4.4.1 Type I Error

Type I error arises as a result of incorrectly rejecting the null hypothesis when it is true. The probability of this is called the significance level, α, of the test. Type I error is estimated by simulating data under the null hypothesis. The rate of rejection of the null hypothesis for data simulated under it should be equal to α. We expect the p-values of data generated under H_0 to be uniformly distributed if the statistic is reliable. Therefore we expect a plot of the empirical cumulative distribution function (CDF) to be a straight diagonal line.

For both the bitrophic and tritrophic hypothesis, this corresponds to independently generating random phylogenetic trees with randomly assigned interactions (see Section 4.2.1 for the bitrophic hypothesis). The trees were generated using the `rtree` function of the R (R Core Team, 2013) package *ape* (Paradis *et al.*, 2004). In the bitrophic case we used the same parameter combinations as Hommola *et al.* (2009) and Legendre *et al.* (2002):

- 10 tips on Tree X, 10 tips on Tree Y and 10, 15, 20, and 25 interactions;
- 10 tips on Tree X, 15 tips on Tree Y and 10, 15, 20, and 25 interactions.

For each parameter combination, 1000 systems were generated. We calculated p-values with $N = 10000$ randomisations for each system using our method and the correlation method proposed by Hommola *et al.* (2009). The results for the first parameter combination, with 10 and 15 interactions, are displayed in Figure 4.11. The remaining plots for the first parameter combination, and the plots for the second parameter combination are in Appendix F.1.

For the tritrophic case we used the same parameter combinations as Mramba *et al.* (2013), with and without triangular interaction constraints:

- 10 tips on Tree X, 10 tips on Tree Y, 10 tips on Tree Z and 10, 15, 20, and 25 interactions between each pair of trees;
- 10 tips on Tree X, 10 tips on Tree Y, 15 tips on Tree Z and 10, 15, 20, and 25 interactions between each pair of trees.

For each parameter combination, 1000 systems were generated. We calculated p-values with $N = 1000$ randomisations for each system using our method and the method of Mramba *et al.* (2013). The results of our method, for the first parameter combination, with triangular interactions, are displayed in Figure 4.12, the

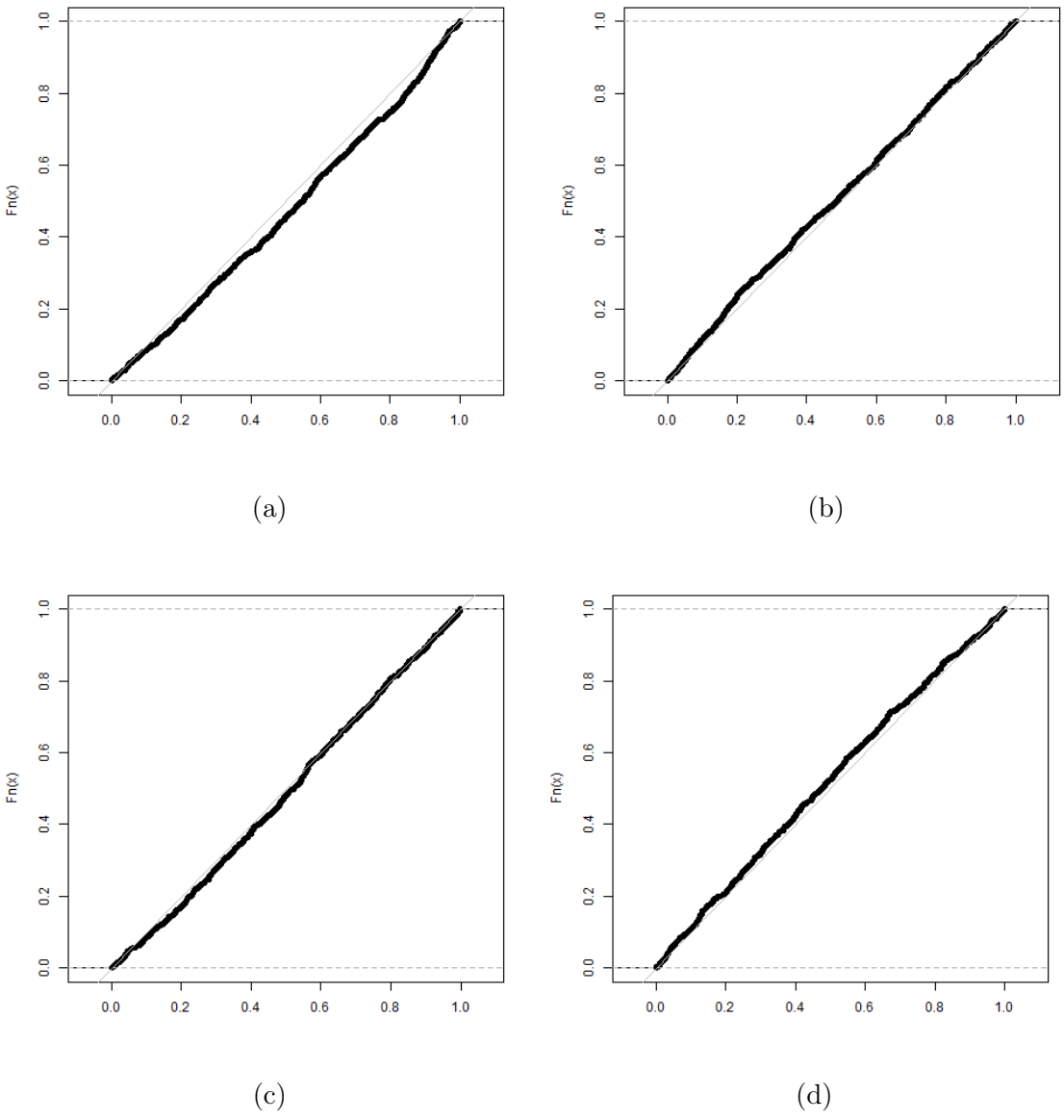


Figure 4.11: Empirical cumulative distribution functions for our p-values and Hommola *et al.*'s (2009). Each plot corresponds to simulations with 10 tips on each tree. The first column corresponds to 10 interactions simulated and the second column corresponds to 15 interactions simulated. The top row contains the p-values for our method, and the bottom row contains the p-values for the method of Hommola *et al.* (2009). The diagonal grey line is the identity line.

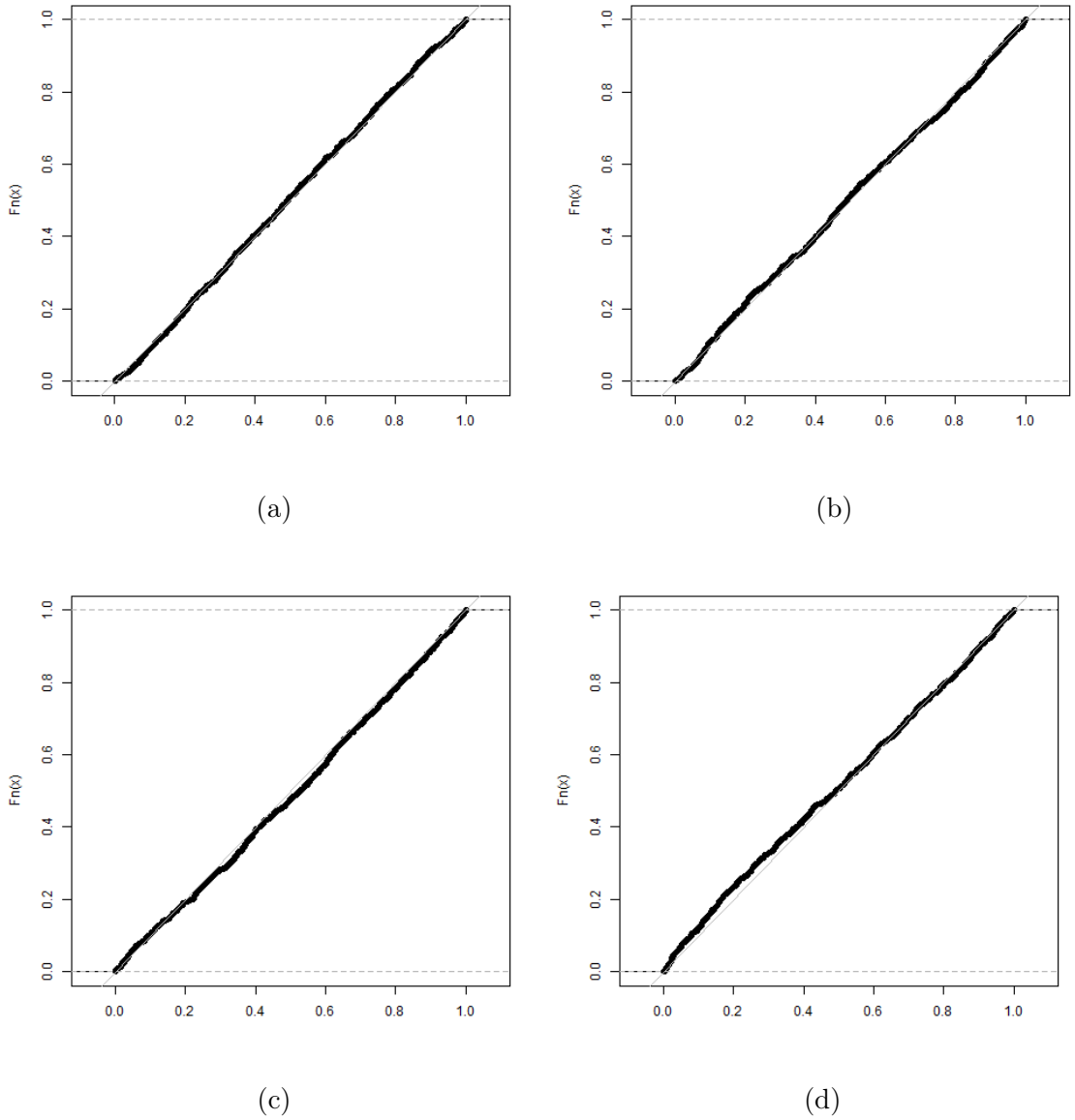


Figure 4.12: Empirical cumulative distribution functions for our tritrophic p-values. Each plot corresponds to simulations with 10 tips on each tree. Each plot represents a different number of interactions simulated. From top left to bottom right, 10, 15, 20 and 25 interactions. The diagonal grey line is the identity line.

results for the second parameter combination, with triangular interactions, are in Appendix F.1.

The empirical CDF for our p-values lies close to the desired diagonal line for all parameter combinations in the bitrophic and tritrophic cases. The same is true of the methods of Hommola *et al.* (2009) and Mramba *et al.* (2013). However, when applied to datasets where there are no constraints on the interactions, Mramba

et al.'s (2013) p-values are biased for systems where there are fewer interactions. This is because any interactions that do not form triangles are discarded. For the parameter combinations with 10 interactions, 95% and 97% of the simulated systems could not be used to calculate p-values as their interactions did not form enough triangles, as required by that method. In the case of the parameter combinations with 15 interactions; 43% and 65% of the systems could not be used.

4.4.2 Power Simulations - Bitrophic

Statistical power is the probability that the null hypothesis is correctly rejected when it is false. Statistical power has been assessed for our method as well as the correlation statistic proposed by Hommola *et al.* (2009) for the bitrophic case. We followed the simulation approaches adapted by Hommola *et al.* (2009) and Legendre *et al.* (2002) to generate data consistent with H_1. Noise is gradually added using the following three methods, and the proportion of correct rejections of the null hypothesis calculated in each case. In every simulation approach 1000 systems were generated. We calculated p-values with $N = 10000$ randomisations for each system.

Simulation Method 1: Replacing Interactions

For each simulation, Tree X and Tree Y were assigned the same randomly generated phylogenetic tree with interactions initially assigned at corresponding positions on the tree. The interactions connect each leaf node on Tree X with the same leaf node on the identical Tree Y, such that they exhibit perfect cospeciation. A percentage, ranging from 10% to 50%, of these interactions are then replaced with random non-corresponding interactions. We used the following parameter combinations:

- 10 tips on Tree X, 10 tips on Tree Y, 10 corresponding interactions, replacing 1, 2, 3, 4, and 5 random interactions
- 20 tips on Tree X, 20 tips on Tree Y, 20 corresponding interactions, replacing 2, 4, 6, 8, and 10 random interactions

Simulation Method 2: Adding Interactions

As for Simulation Method 1, Tree X and Tree Y were assigned the same phylogenetic tree and interactions assigned at corresponding positions on the tree. A number of random interactions were then added. This simulation approach was performed for the same parameter combinations as for Simulation Method 1.

Simulation Method 3: Randomise Clade Branch Lengths

We now consider the branch lengths of the phylogenies as well as the interactions. A random base tree was generated and the branch lengths randomised to produce Tree X and Tree Y. In each simulation a different number of clades on each tree are randomised. The clades were selected based on their distance from the root node; the clades furthest from the root node were randomised first. The branch lengths in each of the selected clades were randomised by replacing the existing branch lengths with new lengths sampled from the standard uniform distribution. This is the distribution used by *ape* to create the original branch lengths of the trees.

- 10 tips on Tree X, 10 tips on Tree Y, and branch lengths randomised in 1, 2, 3, 4, and 5 clades.
- 20 tips on Tree X, 20 tips on Tree Y, and branch lengths randomised in 2, 3, 4, 5 and 6 clades.

For each simulation approach, we calculated the rejection rate of the null hypothesis at the $\alpha = 0.05$ and $\alpha = 0.01$ significance levels. The rejection rate is calculated as the proportion of times that we reject the null hypothesis. Selected rejection rate plots are displayed in Figure 4.13. Rejection rate plots for Simulation Method 3 are in Appendix F.2. The rejection rates increase as the systems become more cospeciated. For each of the simulation approaches the rejection rates are higher for systems with 20 tips compared to systems with 10 tip trees. It is also clear that the rejection rates are higher for Simulation Method 2 than the other simulation approaches. For each simulation method, our rejection rate is higher than Hommola *et al.*'s (2009) in the 20 tip case, see Figures 4.13c and 4.13d. In the 10 tip case, our rejection rates are equivalent to Hommola *et al.*'s (2009). We obtain similar results at the $\alpha = 0.01$ significance level (see Appendix F.2).

At the bitrophic level our method has the ability to perform at least as well as Hommola *et al.*'s (2009) method, and in most cases performs better. We have shown that our method has greater power to detect cospeciation in systems where noise has been introduced.

4.4.3 Power Simulations - Tritrophic

Statistical power has been assessed for our method at the tritrophic level and we have also compared our method to the permutation test proposed by Mramba *et al.*

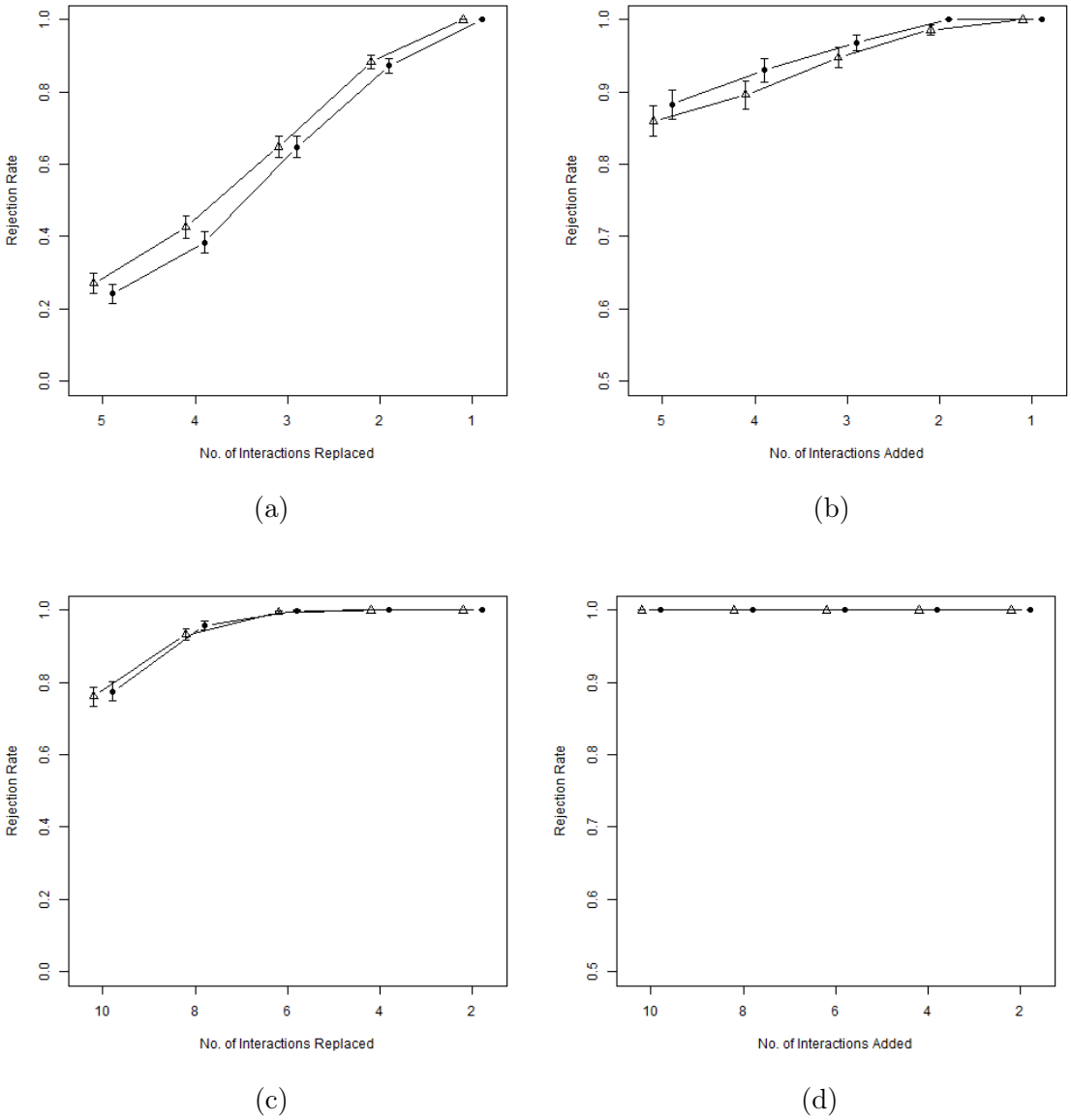


Figure 4.13: Rejection rates for the p-values generated using our method and the method of Hommola *et al.* (2009) at the $\alpha = 0.05$ significance level, under different simulation approaches. Black dots are the rates obtained using our method and triangles are the rates calculated for Hommola *et al.*'s (2009) p-values. The points are offset on the horizontal axis to prevent overlapping. Each column corresponds to a different simulation approach. The first column corresponds to Simulation Method 1 and the second column corresponds to Simulation Method 2. The top row contains the 10 tip simulations for each approach. The bottom row contains the 20 tip simulations for each approach.

(2013). We followed the simulation approaches adapted by Mramba *et al.* (2013), and repeated these without forcing the interactions to form triangles between the

three trees. In every simulation approach 100 systems were generated. We calculated p-values with $N = 10000$ randomisations for each system.

Simulation Method 1: Replacing Interactions

Trees X and Y were assigned the same randomly generated phylogenetic tree. To avoid computational issues with Mramba *et al.*'s (2013) method independent $N(0, 0.01^2)$ noise was added to the branch lengths, as described in Mramba *et al.* (2013). Interactions were initially assigned at corresponding positions between the trees, such that Tree X and Tree Y exhibit perfect cospeciation. Tree Z is unrelated to Trees X and Y, and is therefore independently generated with randomly assigned interactions between itself and Trees X and Y. The interactions between each pair of trees are then replaced with random interactions. We used the following parameter combinations:

- 10 tips on Trees X, Y and Z, 10 interactions between each pair of trees, and $1, 2, \ldots, 10$ interactions replaced between each pair of trees.
- 20 tips on Trees X, Y and Z, 20 interactions between each pair of trees, and $2, 4, \ldots, 20$ interactions replaced between each pair of trees.

Simulation Method 2: Adding Interactions

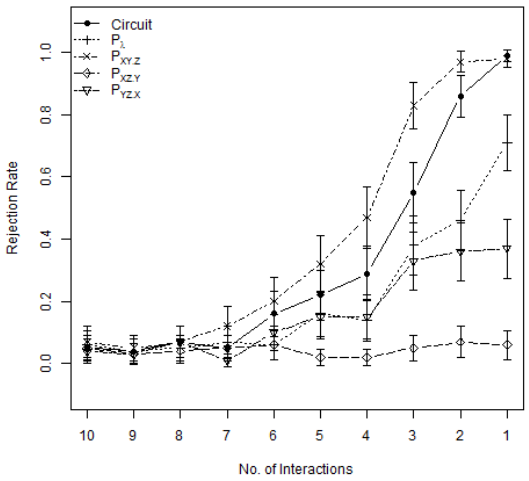
Again, Trees X and Y have the same phylogenetic tree with interactions assigned at corresponding positions. Tree Z is independently generated with random interactions between itself and Trees X and Y. In this approach, interactions were randomly added between each pair of trees. The same parameter combinations were used as in the previous simulation approach.

Our method can only be compared to Mramba *et al.*'s (2013) when the interactions between the three trees are forced to form triangles, as displayed in Figure 4.2. The above simulation approaches are performed with and without triangular interaction constraints. Selected plots of the rejection rates are displayed in Figures 4.14 and 4.15.

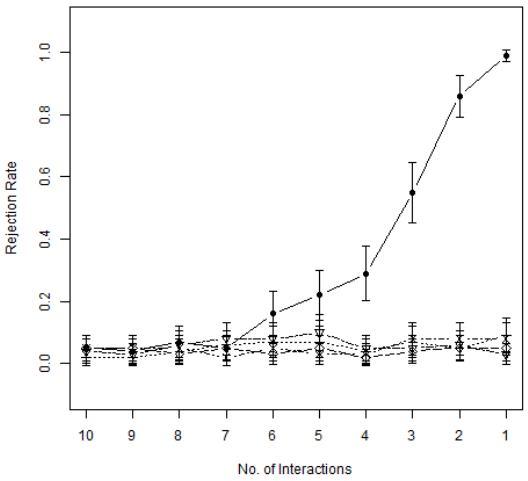
By construction, Tree Z is not involved in the cospeciation between Trees X and Y, thus permuting Tree Z reveals no effect of cospeciation. This can be seen in Figure 4.14b, as expected, the rejections rates for Mramba *et al.*'s (2013) method are all very low. We can interpret Mramba *et al.*'s (2013) p-values, defined in

Section 4.1.2, using Table 4.1. A significant value of $P_{xy.z}$ when Trees X and Y are involved in the randomisation indicates that there is cospeciation between Trees X and Y. This can clearly be seen in Figures 4.14a, 4.14c, 4.14d where the statistic corresponding to $P_{xy.z}$ is the most powerful. The statistics corresponding to $P_{xz.y}$ and $P_{yz.x}$ are less powerful because Trees X and Y are not cospeciating with Tree Z, and randomising Tree X tells us nothing about the cospeciation between Trees Y and Z. Our statistic has slightly less power than $P_{xy.z}$ under some randomisations.

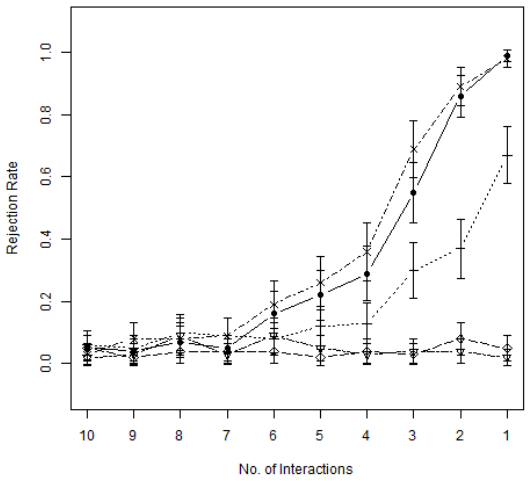
The method of Mramba *et al.* (2013) requires the permutation of every combination of trees, and four different p-values to make conclusions about cospeciation in a tritrophic system. A simple interpretation guide for the relationship between the possible permutations and the p-values is given in Table 4.1. Figure 4.14 displays the rejection rates for our p-values and Mramba *et al.*'s (2013) four different p-values for the simulation approach where we replace triangles of interactions with random triangles of interactions (see Appendix F.2 for results for the simulation approach where we add random triangles of interactions). The rejection rates are calculated at the $\alpha = 0.05$ significance level. Each plot corresponds to a different randomisation in Mramba *et al.*'s (2013) method. The power curve for our method is repeated in each plot. Figures 4.14a, 4.14b, 4.14c and 4.14d correspond to the cases where only Tree X is randomised, only Tree Z is randomised, both Trees X and Y are randomised, and all three trees are randomised, respectively.



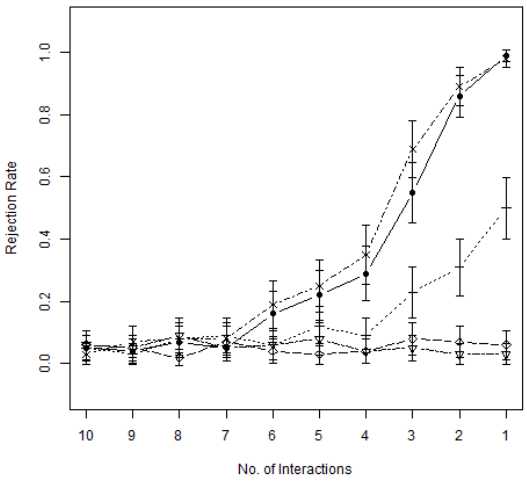
(a) Only X randomised



(b) Only Z randomised



(c) X and Y randomised



(d) X, Y and Z randomised

Figure 4.14: Rejection rates for p-values generated using our method and the method of Mramba *et al.* (2013) at the $\alpha = 0.05$ significance level, under the simulation approach where triangular interactions are replaced between three 10 tip trees. The interactions between the three trees are forced to form triangles. The horizontal axis corresponds to the number of interactions replaced between each pair of trees. Black dots are the rates obtained using our method, labelled "Circuit", and the other lines correspond to the rates calculated for the different p-values obtained under Mramba *et al.*'s (2013) method; P_λ, $P_{xy.z}$, $P_{xz.y}$ and $P_{yz.x}$.

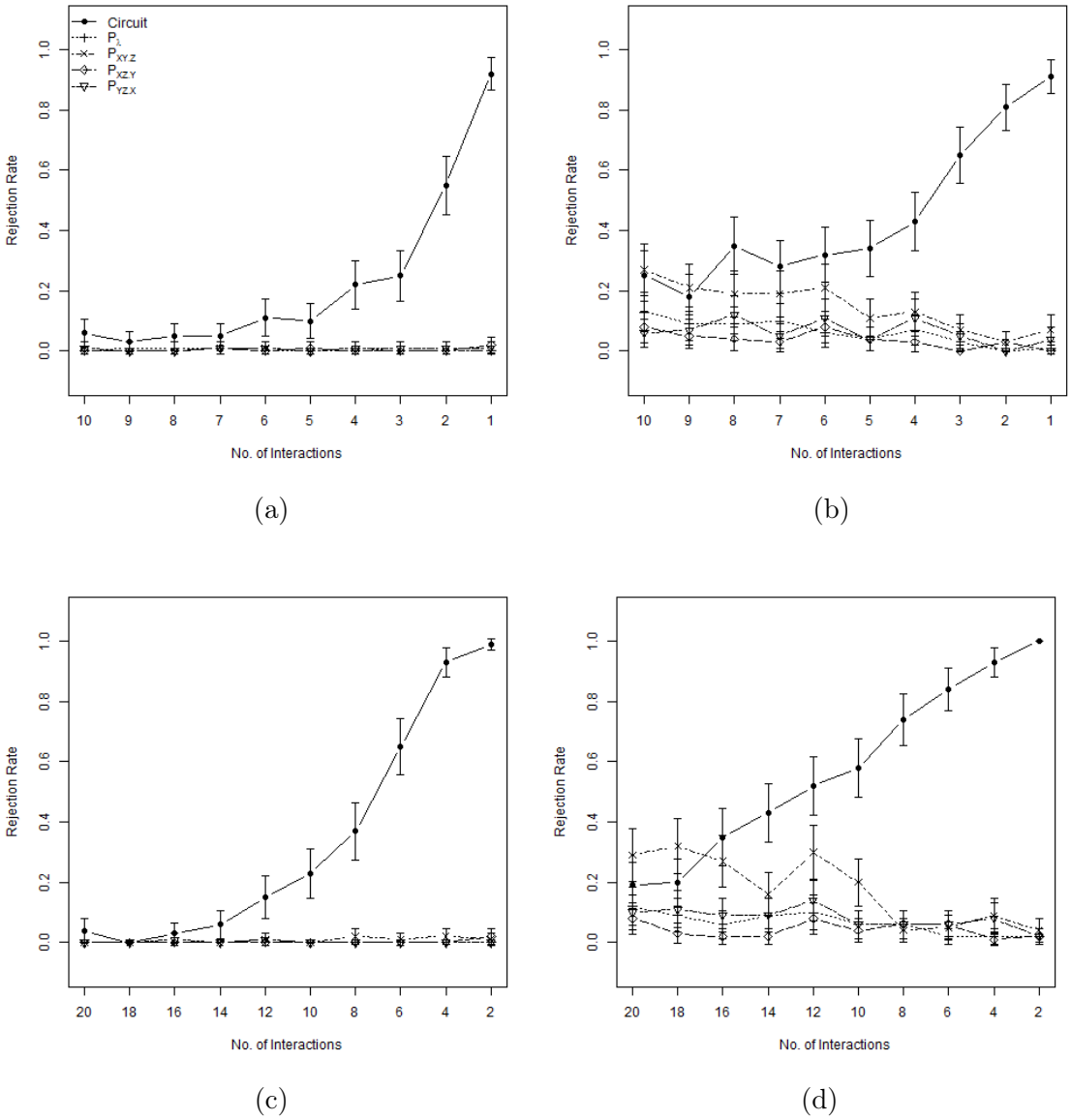


Figure 4.15: Rejection rates for p-values generated using our method and the method of Mramba *et al.* (2013) at the $\alpha = 0.05$ significance level, under different simulation approaches. Each column corresponds to a different simulation approach; replacing and adding interactions between the three trees, respectively. The horizontal axis corresponds to the number of interactions replaced or added between each pair of trees. In each simulation the interactions are not forced to form triangles. The rows correspond to the tree sizes. The first row contains the 10 tip simulations for each approach. The second row contains the 20 tip simulations for each approach. Each plot corresponds to the case where only Tree X is randomised for Mramba *et al.*'s (2013) method. Black dots are the rates obtained using our method, labelled "Circuit", and the other lines correspond to the rates calculated for the different p-values obtained under Mramba *et al.*'s (2013) method; P_λ, $P_{xy.z}$, $P_{xz.y}$ and $P_{yz.x}$.

However, in natural systems there is no restriction that the interactions form triangles between the three phylogenetic trees. Figure 4.15 displays the rejection rates, calculated at the $\alpha = 0.05$ significance level, for our method and Mramba *et al.*'s (2013) for simulations with interactions that are not constrained to form triangles. The first column of plots corresponds to simulation method 1 and the second column to simulation method 2. The rows correspond to the size of the trees; the first row is simulations involving 10 tip trees and the second row is 20 tip trees. We show only one of Mramba *et al.*'s (2013) randomisations, the case where only Tree X is randomised; other plots display very similar results. Clearly our statistic is more powerful than the method of Mramba *et al.* (2013). Similar results were obtained at the $\alpha = 0.01$ significance level (see Appendix F.2).

To calculate their p-values, the method of Mramba *et al.* (2013) must discard any interactions that do not form triangles. On average at least 60% of the interactions were discarded in every simulation approach; in most of these simulations over 80% of the interactions were discarded on average. Mramba *et al.*'s (2013) p-values cannot be calculated unless there are at least three triangles. Any p-values that cannot be calculated are not included in the calculation of the rejection rate. Therefore many of the rejection rates calculated for the method of Mramba *et al.* (2013) are calculated based on only a fraction of the systems simulated. If none of the p-values can be calculated then the rejection rate is zero.

4.4.4 Application to Real Data

We applied our method to a tritrophic dataset consisting of hostplants (H), leaf-mining moths (P) and parasitoid wasps (W) (Lopez-Vaamonde *et al.*, 2005). We set the value of ϵ on the interactions to be the average of all the branch distances on the tree they are connected to. The value of ϵ on the interactions between Trees M and W and Trees H and W is given by the average branch distance over the two trees they are connected to. We used the reconstructed phylogenetic trees calculated by Mramba *et al.* (2013). The three phylogenies and their interactions are displayed in Figure 4.16. The interactions do not all form the triangles that are necessary for Mramba *et al.*'s (2013) method; in fact 12 interactions had to be discarded.

The p-values for Mramba *et al.*'s (2013) method are given in Table 4.2, significant p-values are displayed in bold font. The rows represent the permutations, and the columns represent the different p-values of Mramba *et al.*'s (2013) method. Lopez-Vaamonde *et al.* (2005) found no evidence that the hostplant, leaf-mining moth or parasitoid wasp exhibit cospeciation at a pairwise level. By contrast, Mramba

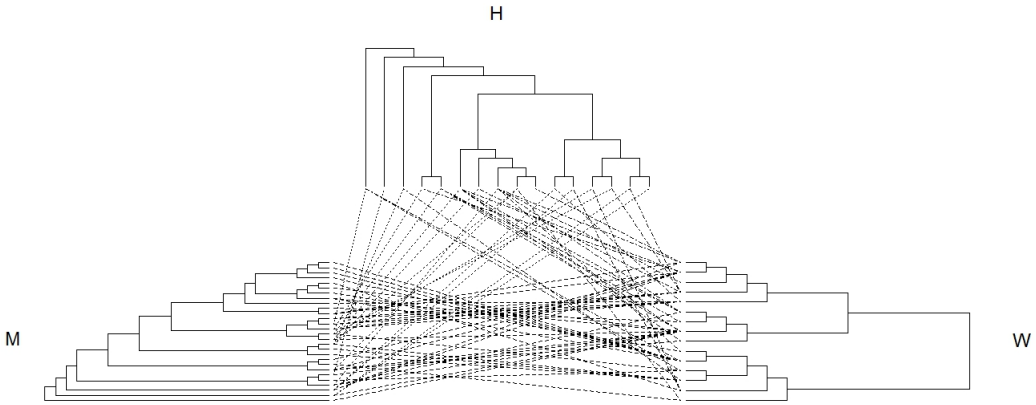


Figure 4.16: Tritrophic system consisting of hostplants (H), leaf-mining moths (M) and parasitoid wasps (W) (Lopez-Vaamonde *et al.*, 2005). The phylogenetic trees were reconstructed by Mramba *et al.* (2013). Branch lengths have not been used to plot the trees. Plots of the phylogenetic trees using the branch lengths are given in Appendix F.3. The dashed lines display the interactions between the leaf nodes of the three trees.

et al. (2013) found mixed evidence for cospeciation but conclude that the parasitoid wasp has been central in the cospeciation of the tritrophic system. Our results coincide with those of Mramba *et al.* (2013). We obtain a p-value of 0.441, with the parasitoid wasp phylogeny taking the role of Tree Z in Section 4.2.1. This indicates that there is insufficient evidence to reject the null hypothesis and therefore any

Permutation	P_λ	$P_{MW.H}$	$P_{HW.M}$	$P_{HM.W}$
H	0.134	0.908	0.156	0.152
M	0.963	0.054	0.998	0.249
W	**0.031**	0.082	**0.035**	0.982
HM	0.957	**0.028**	0.213	0.132
HW	0.127	0.248	**0.010**	0.238
MW	0.957	0.062	0.067	0.265
HMW	0.954	**0.048**	**0.012**	0.139

Table 4.2: The p-values obtained using the method of Mramba *et al.* (2013) applied to the hostplants (H), leaf-mining moths (M) and parasitoid wasps (W) dataset (Lopez-Vaamonde *et al.*, 2005). Significant p-values are highlighted in bold. The rows indicate which phylogenetic trees have been permuted. The columns correspond to the different p-values obtained using the method of Mramba *et al.* (2013).

cospeciation between the moth and hostplant is due to the parasitoid wasp driving the cospeciation in the tritrophic system.

4.5 Discussion

We have introduced a method that efficiently tests cospeciation hypotheses in tritrophic systems. We use one sophisticated permutation scheme based on weighted interactions to test our hypothesis. This is an improvement on the multiple permutation scheme required by the method proposed by Mramba *et al.* (2013). Unlike Mramba *et al.*'s (2013) method, which requires the interactions to form sets of triangles, we do not require specific interaction patterns to be formed between the three phylogenies to calculate our statistic or to perform the randomisations. As a result no information is discarded with our method, and we obtain unbiased p-values. Discarding interactions results in biased p-values for the method of Mramba *et al.* (2013). Statistical power for each method was evaluated by simulating data under the alternative hypothesis. Our method out performed Mramba *et al.*'s (2013) in all cases where the interactions were not constrained, even when noise was introduced to the data.

We successfully applied our method to a tritrophic dataset of hostplants, leaf-mining moths and parasitoid wasps. Our conclusions support those of Mramba *et al.* (2013).

We have also demonstrated that our method is effective at the bitrophic level. We observe unbiased p-values when assessing type I error. We have shown, using power calculations, that our method performs at least as well as Hommola *et al.*'s (2009), and in most cases better.

Due to the calculation of the direct distances between the external nodes, our method is not restricted to phylogenetic trees; it can still be applied when the system involves phylogenetic networks. It is also easily generalised to higher order systems.

Existing methods use a binary system to determine whether or not an association exists between two species. By setting distances, ϵ, on the interactions, our method allows the interactions to be weighted according to the user's criteria. For example, there may a degree of uncertainty surrounding the likelihood of an association existing.

In Section 4.4 we simulate systems to test the performance of our method. The trees and interactions in each system are simulated separately, and do not take into account how the system has evolved. There are many limitations to this approach.

Simulating the systems in this way does not take into account how the systems might have evolved. To simulate a cospeciating bitrophic system two identical trees are generated and interactions placed at corresponding positions between the two trees. Random bitrophic systems are simulated by generating two random trees and randomly assigning the interactions. This is not a flexible approach because the number of external nodes must be selected, and the number of interactions. In addition, it is difficult to simulate systems between these extremes. We can only partially achieve this by altering the branch lengths and randomising the clades and interactions on a tree. These disadvantages are even more pronounced at the tritrophic level, where the range of systems is more complex. In Chapter 5 we introduce a method to simulate these systems under different evolutionary scenarios over time.

Our method has been implemented using R (R Core Team, 2013) and the source code is available from: `http://www.maths.leeds.ac.uk/~stuart/research`

Chapter 5

Simulating the Evolution of Ecologically Associated Species

5.1 Introduction

There are limitations to the approaches used in Chapter 4 to simulate bitrophic and tritrophic datasets. These approaches generate random trees using the `rtree` function of the R (R Core Team, 2013) package *ape* (Paradis *et al.*, 2004), and separately generate the interactions between the trees. To simulate a cospeciating bitrophic system two identical trees are generated and interactions placed at corresponding positions between the two trees. Non-cospeciating bitrophic systems are simulated by generating two random trees and randomly assigning the interactions. This is not a flexible approach because the number of external nodes must be selected, as must the number of interactions. In addition, it is difficult to simulate systems between these extremes. We can only partially achieve this by altering the branch lengths and randomising the clades and interactions on a tree. Simulating the systems in this way does not take into account how the systems might have evolved. These disadvantages are even more pronounced at the tritrophic level, where the range of systems is more complex and subtle.

In this chapter we introduce a more realistic method to simulate bitrophic and tritrophic systems under different evolutionary scenarios. The algorithm starts with one species per lineage, that are assumed to have an ecological interaction. The joint evolution of these species is simulated by sampling the times at which evolutionary events occur from an exponential distribution. The main evolutionary events that

we are interested in are speciation, and gaining or losing ecological interactions. The occurrence of these events are controlled by a set of parameters. By experimenting with different intensities and parameter combinations, a wide range of systems with different cospeciation properties can be simulated. These systems vary from having no association to complicated patterns of interactions between tritrophic systems. We are particularly interested in tritrophic systems where one phylogenetic tree is driving the cospeciation in the system, consistent with the tritrophic null hypothesis in Chapter 4. We initially focus on simulating interacting bitrophic systems, before extending the method to the tritrophic case. However, there is also scope to generalise to higher order systems. We produce example bitrophic and tritrophic systems to display the range of systems our algorithm is able to produce. The performance of our method is evaluated by testing that the systems produced exhibit the desired level of cospeciation.

5.1.1 Existing Methodology

In Chapter 4 bitrophic and tritrophic systems were simulated using the `rtree` function in *ape* (Paradis *et al.*, 2004). `rtree` generates random phylogenetic trees by randomly splitting the edges. The branch lengths are sampled from the standard uniform distribution. This method is able to simulate individual trees independently, however it is unable to simulate an interacting system containing more than one tree. It also does not take into account how the tree might have evolved. There are many programs available to simulate individual phylogenetic trees; Geiger, TreeSample, TESS, PhyloGen, TreeSim (Harmon *et al.*, 2007; Hartmann *et al.*, 2010; Höhna, 2013; Rambaut, 2002; Stadler, 2010). All of these methods simulate under the episodic birth-death process (EBDP). In these processes, the births correspond to speciation events and the deaths corresponds to the rate of losing an existing lineage. These rates can shift, and the models include mass extinction events. Stadler (2011) gives an overview of these methods, and the EBDP. There are currently no methods available that we are aware of to simulate interacting systems of phylogenies.

5.2 Methods and Materials

Starting with a single species per phylogenetic tree, we detail an algorithmic approach to simulate the joint evolution of these associated species through time. A set of parameters control a range of evolutionary events, chosen to reflect different

coevolution scenarios. The time an event occurs is sampled from an exponential distribution, and these times are used to construct branch lengths.

5.2.1 Bitrophic Case

The simplest case is a bitrophic system consisting of two phylogenetic trees, X and Y. At time $t = 0$, we assume that there is one species per lineage, x_1 and y_1, the roots of the phylogenetic trees. These species are assumed to be interacting. We consider three events that can occur over time to shape the evolution of the system; bifurcation of a branch, and gaining or losing an interaction with a species on the other tree. These events are described in more detail below, and displayed in Figure 5.1.

Bifurcation of a branch

There are three points to consider for this event:

- A single node on one tree may speciate resulting in two descendants (Figure 5.1a).
- Alternatively, a pair of interacting nodes may speciate simultaneously, representing cospeciation between the trees (Figure 5.1b). Each species can only speciate once resulting in exactly two descendants, to form binary trees.
- When two species cospeciate, the interactions are inherited at corresponding positions, as shown in Figure 5.1b. When one node speciates, any interactions that node was involved in are inherited independently by the descendants with probability q_{XY} (the interaction is lost with probability $1 - q_{XY}$). The possible interaction placement outcomes are displayed in Figure 5.2.

Gain an interaction

An interaction may be gained between any two species on the two trees (Figure 5.1c). An interaction cannot be gained between two species where an interaction already exists at that time.

Lose an interaction

An interaction may be lost between any two species that are interacting at that time (Figure 5.1d).

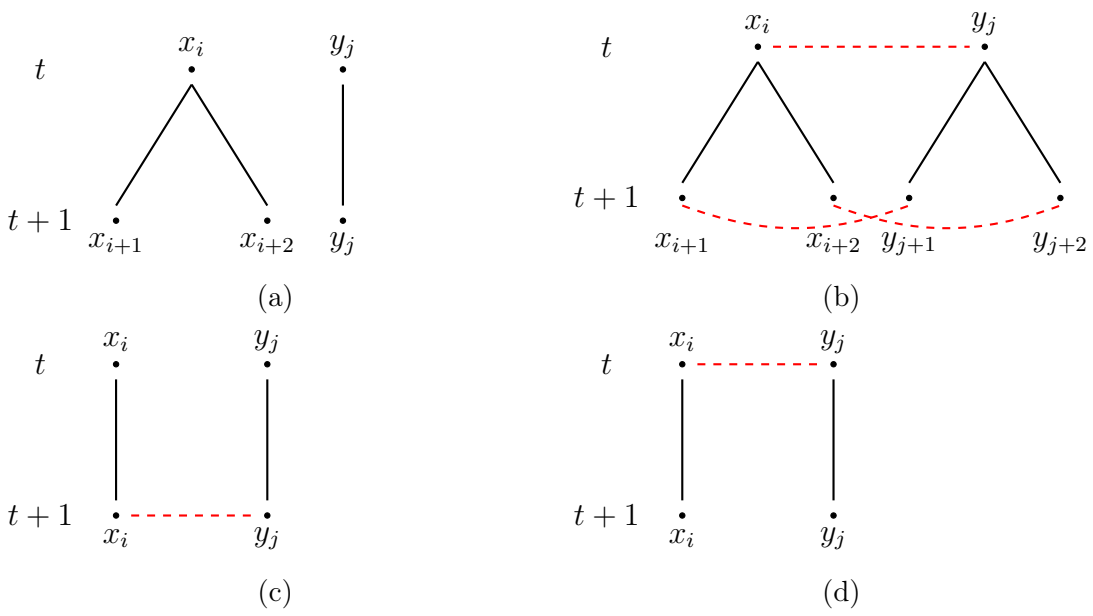


Figure 5.1: Evolutionary events included in the simulation model. (a) A node, x_i, on one tree speciates resulting in two descendants, x_{i+1} and x_{i+2}. (b) Interacting species x_i and y_j speciate simultaneously to produce two descendants on each tree, indicative of cospeciation. Interactions are inherited at corresponding positions, indicated by red dashed lines. (c) An interaction is gained between two species that were not previously interacting, x_i and y_j. (d) An interaction is lost between two interacting species, x_i and y_j.

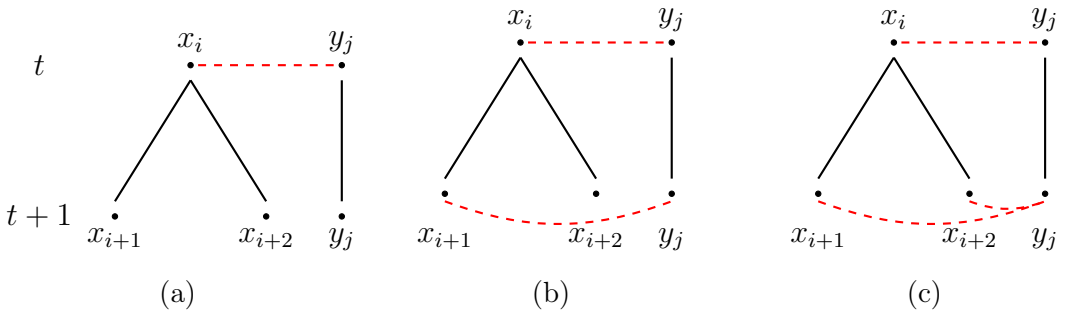


Figure 5.2: Possible interaction placement following a node, x_i, on one tree speciating to produce descendants x_{i+1} and x_{i+2}, as displayed in Figure 5.1a. Each descendant of x_i inherits the interaction with y_j with probability q_{XY}. (a) Neither of the descendants x_{i+1} or x_{i+2} inherit the interaction with y_j. (b) One of the descendants inherits the interaction. (c) Both of the descendants inherit the interaction.

Each of the events is assigned an intensity of occurring in a small time interval dt. These intensities are defined as follows.

Bifurcation of a branch

- The intensities corresponding to a single node in Tree X or Y speciating is given by $\frac{\alpha_{x_i}}{m'}$ and $\frac{\beta_{y_j}}{n'}$, where x_i and y_j correspond to node i and j on Trees X and Y respectively, and m' and n' are the number of nodes on Trees X and Y that have not yet speciated. Dividing by m' and n' stops one tree from accruing an increased chance of speciation simply by having more nodes. Every node on each tree can be allowed to have different speciation intensities if desired.

- We also have the intensity corresponding to a node in each tree speciating simultaneously, indicating cospeciation between the two species. This is given by $(\alpha\beta)_{x_i,y_j}$, where x_i and y_j correspond to node i and j on Trees X and Y respectively. Each pair of nodes can be set different intensities of cospeciating. Two species can only cospeciate if there is an interaction between them, since species that are not associated are unlikely to cospeciate.

Gain an interaction

The intensity corresponding to an interaction being gained between a species on one tree and a species on another is given by λ_{x_i,y_j}. As before, this allows different pairs of nodes to have different intensities of gaining an interaction.

Lose an interaction

The intensity corresponding to an interaction being lost between a species on one tree and a species on another is given by μ_{x_i,y_j}. Again, this allows different pairs of nodes to have different intensities of losing an interaction.

The total rate, Λ, of an event occurring in a small time interval dt is calculated by summing over all of the possible events:

$$\Lambda = \frac{\sum_i \alpha_{x_i}}{m'} + \frac{\sum_j \beta_{y_j}}{n'} + \frac{\sum_i \sum_j (\alpha\beta)_{x_i,y_j}}{\mathbf{1}^T I_{XY} \mathbf{1}} + \lambda_{x_i,y_j}(mn - \mathbf{1}^T I_{XY}\mathbf{1}) + \mu_{x_i,y_j}(\mathbf{1}^T I_{XY}\mathbf{1}), \tag{5.1}$$

where $\mathbf{1}$ is a column vector of ones, m and n are the total number of species on Trees X and Y, m' and n' are the number of species on Trees X and Y that have not speciated, and i and j sum over the nodes that could speciate in each case.

The binary matrix $(I_{XY})_{mxn}$ contains the interactions between the species on Trees X and Y. The rows correspond to species on Tree X and the columns correspond to species on Tree Y. The $(i,j)^{\text{th}}$ element of I_{XY} is equal to one if there is an interaction between species i and j, and zero otherwise.

The length of time before an event occurs can be thought of in terms of survival analysis. If the event is considered similarly to the event of death, then the probability of survival up to any given time t is equivalent to the probability of waiting until time t before an event occurs in our system. This is represented by the survival function:

$$S(t) = \Pr(T > t) = e^{-\Lambda t}, \tag{5.2}$$

where T is a random variable denoting the time of death, or in this case the time an event occurs, Λ is as defined in Equation (5.1) and t is some time.

The cumulative distribution function, or lifetime distribution function, for Equation (5.2) is given by

$$F(t) = \Pr(T \leq t) = 1 - e^{-\Lambda t}.$$

The derivative of the lifetime distribution function gives the event density; the rate of death, or in our case, the number of events per unit time. The event density is therefore calculated as follows

$$f(t) = \frac{d}{dT} 1 - e^{-\Lambda t} = \Lambda e^{-\Lambda t}.$$

Clearly, the event density follows an exponential distribution with rate parameter Λ, $T \sim \exp(\Lambda)$. To determine the time at which an event occurs we sample from this distribution.

After the time of an event has been sampled, we determine which event occurs and which nodes are involved as follows:

Bifurcation of a branch

The bifurcation events are sampled with probabilities

$$\frac{\sum_i \alpha_{x_i}}{\Lambda m'}, \frac{\sum_j \beta_{y_j}}{\Lambda n'}, \frac{\sum_i \sum_j (\alpha\beta)_{x_i,y_j}}{\Lambda \mathbf{1}^T I_{XY} \mathbf{1}}. \tag{5.3}$$

Gain an interaction

The gaining interaction events are sampled with probabilities

$$\frac{\lambda_{x_i,y_j}(mn - \mathbf{1}I_{XY}\mathbf{1})}{\Lambda}. \tag{5.4}$$

Lose an interaction

The losing interaction events are sampled with probabilities

$$\frac{\mu_{x_i,y_j}(\mathbf{1}^T I_{XY}\mathbf{1})}{\Lambda}. \tag{5.5}$$

The input parameters in the simulation model are summarised in Table 5.1.

Parameter	Description
α_{x_i}, β_{y_j}	Single node speciates
$q_{x_i y_j}$	Interaction inherited
$(\alpha\beta)_{x_i y_j}$	Two nodes cospeciate
$\lambda_{x_i y_j}$	Interaction gained
$\mu_{x_i y_j}$	Interaction lost

Table 5.1: Summary of the input parameters for the bitrophic simulation model.

Simulation:

1. Set parameters in Table 5.1.
2. Sample $t \sim \exp(\Lambda)$ where Λ is defined in Equation (5.1).
3. Sample which event occurs with the probabilities given in Equations (5.3), (5.4) and (5.5).
4. Sample which species are involved in the event with relevant probabilities.
5. Repeat from Step 1, adding the new sampled time t to the previous time, until the desired number of external nodes have been reached (or some other criteria).

5.2.2 Tritrophic Case

The tritrophic case is a simple extension of the bitrophic case. The main differences are an additional phylogenetic tree, Tree Z, and the interaction matrices, I_{XZ} and I_{YZ} containing the interactions between Tree X and Tree Z, and Tree Y and Tree Z, respectively. These differences result in the following additional possibilities for

each event.

Bifurcation of a branch

- A single node may speciate on any of the three trees. This results in an additional intensity for Tree Z, $\frac{\gamma_{z_k}}{o'}$, where z_k corresponds to node k on Tree Z, and o' is the number of nodes on Tree Z that have not yet speciated. Any interactions that node z_k is involved in with Trees X and Y are inherited by the descendants with probability q_{XZ} and q_{YZ} respectively.
- A pair of interacting nodes may speciate simultaneously between any pairwise combination of the three trees. Additional intensities for Trees X and Y cospeciating separately with Tree Z are given by $(\alpha\gamma)_{x_i z_k}$ and $(\beta\gamma)_{y_j z_k}$ respectively. Any interactions that the two speciating trees have with the third tree are inherited by the descendants with probability q_{XY}, q_{XZ} and q_{YZ} respectively.
- A node on each of the three trees may speciate simultaneously if they are all interacting, representing three-way cospeciation, as displayed in Figure 5.3. The corresponding intensity is given by $(\alpha\beta\gamma)_{x_i y_j z_k}$. Similarly to the bitrophic case, interactions are inherited at corresponding positions between each pair of trees.

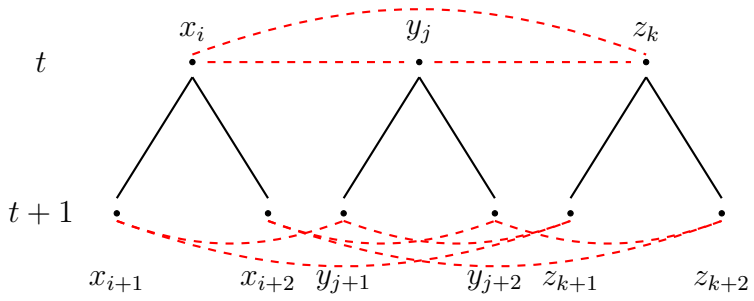


Figure 5.3: Interaction placement at time $t+1$ when interacting nodes x_i, y_j and z_k speciate simultaneously. All descendants inherit the interactions at corresponding positions between each pair of trees.

Gain an interaction

An interaction may be gained between any two species on any pair of the three trees. Therefore, we have additional intensities for nodes on Trees X and Y gaining an interaction with nodes on Tree Z, $\lambda_{x_i z_k}$ $\lambda_{y_j z_k}$ respectively.

Lose an interaction

An interaction may be lost between any two species on any pair of the three trees. Therefore, we have additional intensities for nodes on Trees X and Y losing an interaction with nodes on Tree Z, $\mu_{x_i z_k}$ $\mu_{y_j z_k}$ respectively.

The input parameters for the tritrophic simulation algorithm are summarised in Table 5.2.

Parameter	Description
α_{x_i}, β_{y_j}, γ_{z_k}	Single node speciates
$q_{x_i y_j}$, $q_{x_i z_k}$, $q_{y_j z_k}$	Interaction inherited
$(\alpha\beta)_{x_i y_j}$, $(\alpha\gamma)_{x_i z_k}$, $(\beta\gamma)_{y_j z_k}$	Two nodes cospeciate
$(\alpha\beta\gamma)_{x_i y_j z_k}$	Three-way cospeciation
$\lambda_{x_i y_j}$, $\lambda_{x_i z_k}$, $\lambda_{y_j z_k}$	Interaction gained
$\mu_{x_i y_j}$, $\mu_{x_i z_k}$, $\mu_{y_j z_k}$	Interaction lost

Table 5.2: Summary of the input parameters for the tritrophic simulation model.

5.3 Bitrophic Results

In Section 5.3.1 we use our method to simulate example systems with varying degrees of cospeciation and plot the resulting phylogenetic trees and interactions. On a larger scale, we assess whether our simulation method has the ability to produce systems with varying degrees of cospeciation by testing each system generated using our method from Chapter 4. For simplicity all of the nodes in a tree are set the same intensity for each parameter. For these simulations, the intensities of gaining and losing interactions are calibrated to ensure the average number of interactions is set at a desired level.

5.3.1 Example Systems

In the bitrophic case there are two extremes; independent systems and systems that exhibit perfect cospeciation. To simulate between these extremes, we show sample simulated trees. In all figures in this chapter, Tree X is on the left and Tree Y is on the right. In these simulations we increased $(\alpha\beta)$ in increments and decreased α and β correspondingly. Thus, the intensity of a cospeciation event was gradually increased, and the intensity of the trees speciating independently was reduced. The

value of q_{XY} is also increased gradually, to promote cospeciating interactions. The parameter combinations used to simulate each system are given in Table 5.3. The values of μ and λ for Systems 1-5 are chosen based on the parameter calibrations in Section 5.3.2.

System 1: Independent System

In this scenario, the two phylogenetic trees evolve independently and do not exhibit any cospeciation. Neither of the trees are allowed to split simultaneously, however each tree has a large intensity of bifurcating independently. Following a bifurcation event, each interaction has a 50% chance of being inherited. To add to the randomness, interactions can be gained or lost at random anywhere in the system. The resulting system is displayed in Figure 5.4a.

Systems 2-5: Intermediate Systems

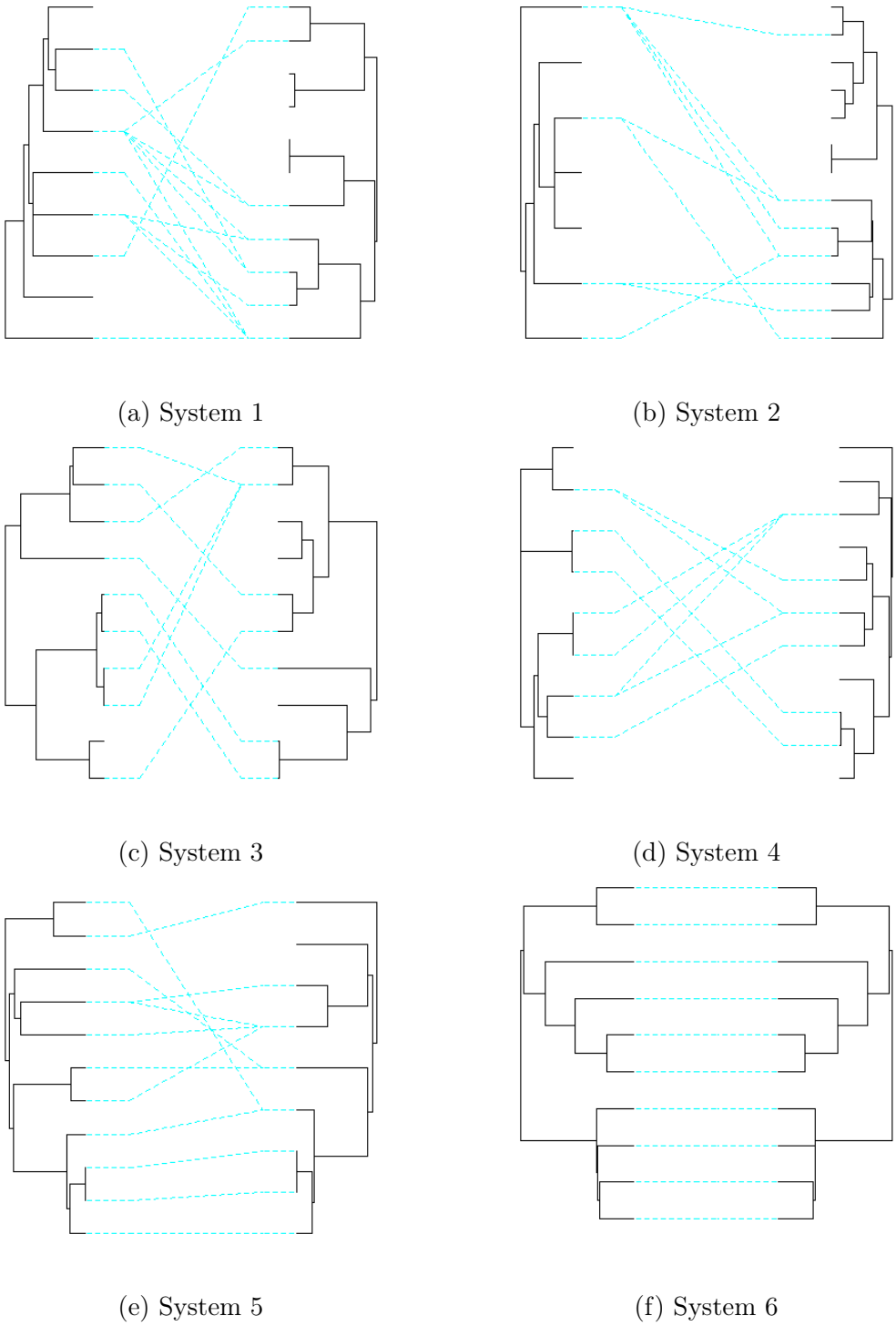
To generate systems between the extremes of System 1 and System 6, the intensity of each tree speciating independently is slowly reduced. The intensity of cospeciation is increased at the same rate as the intensity of independent speciation is reduced. The intensity of cospeciation depends on the number of interactions between nodes that have not yet speciated. To increase the chance of cospeciation, the value of q_{XY} is increased, allowing more interactions to be inherited after an independent speciation event. The resulting systems are displayed in Figures 5.4b to 5.4e.

System 6: Perfect Cospeciation

Both trees exhibit perfect cospeciation. To generate the most extreme case of this system both trees are set to speciate at the same time. No other speciation event is allowed to happen, and no interactions are gained or lost at random. An example system generated is displayed in Figure 5.4f.

5.3.2 Parameter Calibration

We want to test the ability of our method to generate systems ranging from evolving completely independently to perfectly cospeciated on a larger scale. As in the previous section, the values of α, β and $\alpha\beta$ are decreased and increased in increments between 0 and 1 respectively, as displayed in Table 5.3. System 1 represents an independent system and System 6 is a perfectly cospeciated system. Systems 2-5



(a) System 1

(b) System 2

(c) System 3

(d) System 4

(e) System 5

(f) System 6

Figure 5.4: Example systems generated by the bitrophic simulation method exhibiting various levels of cospeciation. (a) An independent system. Both trees have evolved independently, and interactions are inherited, gained and lost at random. (b)-(d) Systems between the extremes of (a) and (e). The intensity of independent speciation of each tree is gradually decreased. The intensity of cospeciation is increased. (e) A perfectly cospeciated system. Both trees have evolved simultaneously, and interactions have only been inherited at corresponding positions.

Parameters		Sys. 1	Sys. 2	Sys. 3	Sys. 4	Sys. 5	Sys. 6
Single node speciates	α β	1	0.8	0.6	0.4	0.2	0
Interaction inherited	q_{XY}	0.5	0.6	0.7	0.8	0.9	NA
Two nodes cospeciate	$\alpha\beta$	0	0.2	0.4	0.6	0.8	1
Interaction gained	λ_{XY}	0.1	0.1	0.1	0.1	0.1	0
Interaction lost	μ_{XY}	0.5	0.5	0.6	0.6	0.5	0

Table 5.3: Parameter input values used to generate the bitrophic systems displayed in Figure 5.4. The values of each parameter are set the same for all nodes in a tree.

represent the range of systems in between. It remains to determine the values of λ and μ to control the number of interactions. This is achieved by numerical simulations. For each parameter combination in Table 5.3, 100 systems are generated. The values of λ and μ are varied between 0 and 1 for each parameter combination.

The value of λ and μ for each parameter combination is selected to keep the average number of final interactions between the leaf nodes equal to the number of leaf nodes on each tree. For System 6 this is trivial, when $\lambda = \mu = 0$, the number of interactions between the leaf nodes will be equal to the number of leaf nodes on each tree. The results of the numerical simulations for Systems 1-5 are displayed in Figure G.1 in Appendix G. Each plot corresponds to a different parameter combination in Table 5.3, and displays the average number of interactions for each value of λ and μ. For Systems 1, 2 and 5, the same value of λ and μ is selected; $\lambda = 0.1$, $\mu = 0.5$. For Systems 3 and 4, $\lambda = 0.1$ and $\mu = 0.6$.

5.3.3 Rejection Rate

To confirm that the systems generated by our method are representing the range of systems we expect, they are tested using our bitrophic cospeciation method in Chapter 4. For each parameter combination in Table 5.3, 1000 systems are simulated. We calculate p-values with $N = 10000$ randomisations for each system. For each parameter combination we calculate the rejection rate of the null hypothesis at the $\alpha = 0.05$ significance level. The rejection rate is calculated as the proportion of times we reject the null hypothesis. The resulting plot is displayed in Figure 5.5.

The rejection rates increase as the systems become more cospeciated, in line with our expectations. However, the rates hardly increase from Systems 1 to 5, and then jump up at System 6. To understand this, we refer back to the power simulations in Chapter 4. In Figure 4.13 it can be seen that both our p-values and Hommola

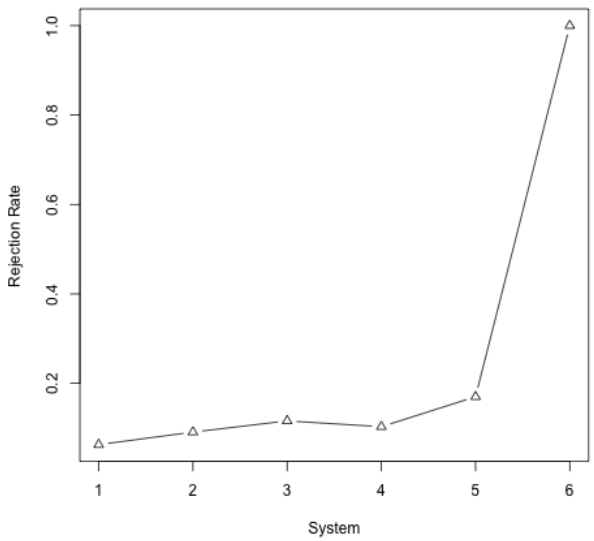


Figure 5.5: Rejection rates for the p-values generated for each parameter combination in Table 5.3 at the $\alpha = 0.05$ significance level. Systems range from Trees X and Y evolving independently to systems where Trees X and Y exhibit perfect cospeciation.

et al.'s (2009) are sensitive to interactions being replaced. In particular, replacing 5 interactions between the external nodes of a 10 tip tree system drastically reduces the rejection rate (Figure 4.13a). In addition, Figure F.5a shows that changing the branch lengths of 1-5 clades on 10 tip trees has an even larger effect on our p-values and Hommola *et al.*'s (2009). The topology of the trees does not change. We repeated these simulations and calculated p-values using the method of Hommola *et al.* (2009), and obtained very similar results. This is unsurprising as we have shown in Chapter 4 that our methods are roughly equivalent.

5.4 Tritrophic Results

The tritrophic setting is not as straightforward as the bitrophic case. There are many different systems that are consistent with the tritrophic null hypothesis in Chapter 4. As a result, there are many different ways to progress from the tritrophic null hypothesis to the corresponding alternative. In Section 5.4.1, we generate example systems to illustrate the range of systems that may be produced. We assess whether our simulation method has the ability to produce systems under the tritrophic hypotheses in Chapter 4 by testing each system generated using our method described

in Chapter 4. For simplicity all nodes in a tree share the same rate for each parameter.

5.4.1 Example Systems

We explore two possible pathways from the null to alternative hypothesis. The first, and simplest, scenario is where systems range from all three trees speciating independently, to Trees X and Y cospeciating, while Tree Z speciates independently. The parameters selected to generate these systems are displayed in Table 5.4a. The value of $\alpha\beta$ is increased, while the values of α and β are decreased. Thus, the intensity of a cospeciation event between Trees X and Y is gradually increased, and the chance of Trees X and Y speciating independently is reduced. The value of γ remains constant at 1, as Tree Z speciates independently in every case. The value of q_{XY} is increased to promote cospeciation between Trees X and Y. In all figures in this chapter, Tree X is on the left, Tree Y is on the right, and Tree Z is at the top.

Unlike the bitrophic case, we do not use parameter calibrations to determine the values of λ and μ for each pair of trees. In the tritrophic case there is a value of λ and μ for each interaction matrix. Setting up the parameter calibrations to simulate 100 systems for every combination of these ranging from 0 to 1, results in 885 780 500 systems to simulate. Computationally this would be too time consuming. Instead, parameter calibration simulations were conducted setting $\lambda_{XY} = \lambda_{XZ} = \lambda_{YZ}$ and $\mu_{XY} = \mu_{XZ} = \mu_{YZ}$. However, no choice of λ and μ was suitable to keep the expected number of interactions at a desired level for all three trees, especially as Trees X and Y became more cospeciated. This was reflected in the example systems simulated; as Trees X and Y became more cospeciated, their interaction graph contained too few interactions. To address this, the value of μ_{XY} is reduced slightly as the value of $\alpha\beta$ is increased.

The resulting systems for each parameter combination in Table 5.4a are displayed in Figure 5.6. In System 1 all three trees are speciating independently. Gradually, Trees X and Y become more cospeciated, independent of Tree Z. In System 6, Trees X and Y exhibit perfect cospeciation, while their interaction with Tree Z is random.

The second pathway through our tritrophic hypothesis ranges from systems where Tree Z is driving the cospeciation in the system, to systems where Trees X and Tree Y are cospeciating above their association with Tree Z. The parameters selected to generate these systems are displayed in Table 5.4b. The intensity of

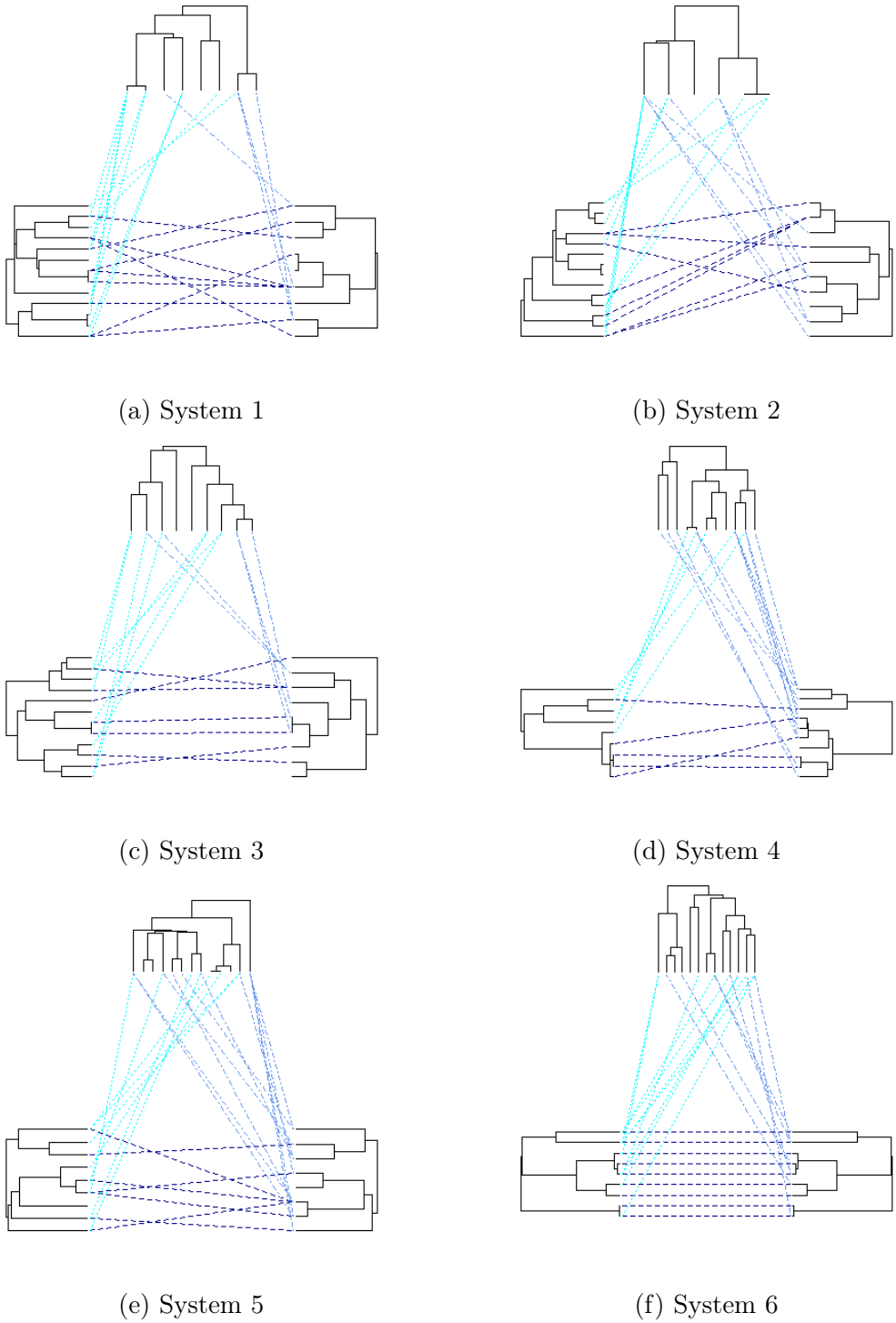
(a) Case 1

Parameters		Sys. 1	Sys. 2	Sys. 3	Sys. 4	Sys. 5	Sys. 6
Single node speciates	α	1	0.8	0.6	0.4	0.2	0
	β						
	γ		1	1	1	1	1
Interaction inherited	q_{XY}	0.5	0.6	0.7	0.8	0.9	NA
	q_{XZ}		0.5	0.5	0.5	0.5	0.5
	q_{YZ}						
Two nodes cospeciate	$\alpha\beta$	0	0.2	0.4	0.6	0.8	1
	$\alpha\gamma$	0	0	0	0	0	0
	$\beta\gamma$						
Three-way cospeciation	$\alpha\beta\gamma$	0	0	0	0	0	0
Interaction gained	λ_{XY}	0.1	0.1	0.1	0.1	0.1	0
	λ_{XZ}						0.1
	λ_{YZ}						
Interaction lost	μ_{XY}	0.6	0.6	0.5	0.5	0.4	0
	μ_{XZ}			0.6	0.6	0.6	0.6
	μ_{YZ}						

(b) Case 2

Parameters		Sys. 1	Sys. 2	Sys. 3	Sys. 4	Sys. 5	Sys. 6
Single node speciates	α	0	0	0	0	0	0
	β						
	γ		0.2	0.4	0.6	0.8	1
Interaction inherited	q_{XY}	0.5	0.6	0.7	0.8	0.9	NA
	q_{XZ}	1	0.9	0.8	0.7	0.6	0.5
	q_{YZ}						
Two nodes cospeciate	$\alpha\beta$	0	0.2	0.4	0.6	0.8	1
	$\alpha\gamma$	1	0.8	0.6	0.4	0.2	0
	$\beta\gamma$						
Three-way cospeciation	$\alpha\beta\gamma$	1	0.8	0.6	0.4	0.2	0
Interaction gained	λ_{XY}	0.1	0.1	0.1	0.1	0	0
	λ_{XZ}	0				0.1	0.1
	λ_{YZ}						
Interaction lost	μ_{XY}	0.6	0.5	0.4	0.3	0	0
	μ_{XZ}	0	0.3	0.3	0.4	0.6	0.6
	μ_{YZ}						

Table 5.4: Parameter input values used to generate example tritrophic systems. The values of each parameter are set the same for all nodes in a tree. (a) Parameter values for the first scenario; trees range from speciating independently to Trees X and Y cospeciating above their interaction with Tree Z. These systems are displayed in Figure 5.6. (b) Parameter values for the second scenario; systems range from Tree Z driving the cospeciation in the system, to Trees X and Y cospeciating above their interaction with Tree Z. These systems are displayed in Figure 5.7.



(a) System 1

(b) System 2

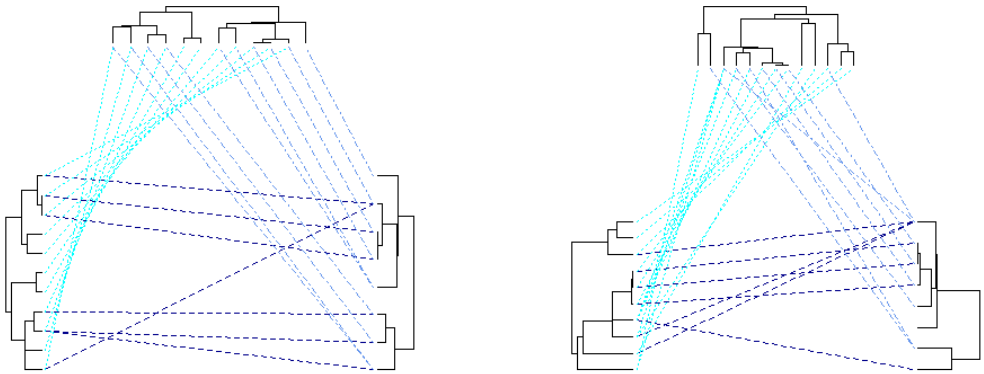
(c) System 3

(d) System 4

(e) System 5

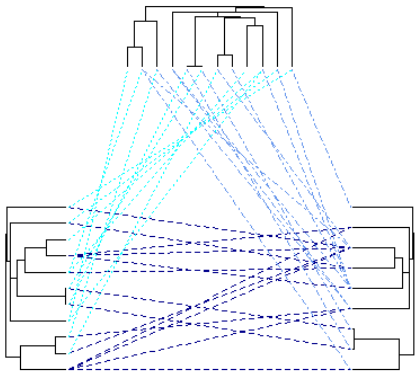
(f) System 6

Figure 5.6: Example systems generated by the tritrophic simulation method. The systems represent one pathway from the tritrophic null to alternative hypothesis detailed in Chapter 4. (a) An independent system. All three trees have evolved independently, and interactions are inherited, gained and lost at random. (b)-(e) Systems between the extremes of (a) and (f). The intensity of independent speciation of Trees X and Y is gradually decreased. The intensity of cospeciation between Trees X and Y is increased. (f) Trees X and Y are cospeciating perfectly. Tree Z has evolved independently.

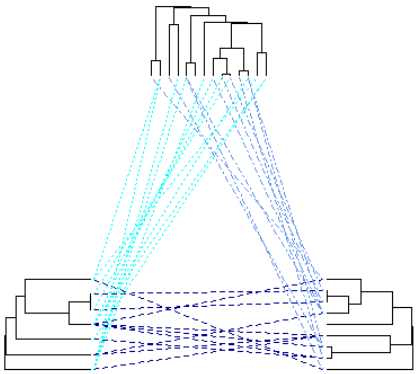


(a) System 1

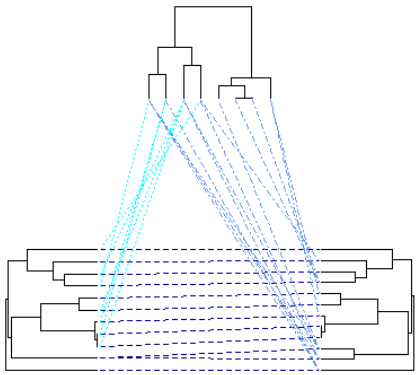
(b) System 2



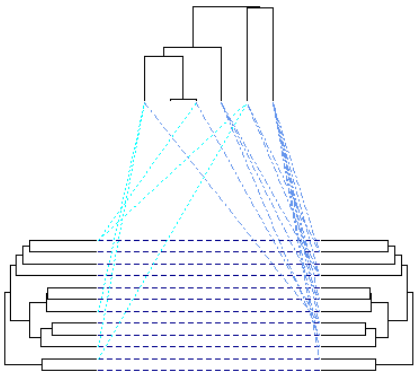
(c) System 3



(d) System 4



(e) System 5



(f) System 6

Figure 5.7: Example systems generated by the tritrophic simulation method. The systems represent one pathway from the tritrophic null to alternative hypothesis detailed in Chapter 4. (a) Tree Z is driving the cospeciation in the system, Trees X and Y are not cospeciating on their own, any cospeciation is a result of how the interactions are inherited after cospeciation with Tree Z. (b)-(e) Systems between the extremes of (a) and (f). The intensity of independent speciation of Tree Z is gradually increased. The intensity of cospeciation between Trees X and Y is increased, while the intensity of cospeciation between Trees X and Z and Trees Y and Z, and their three-way interaction is decreased. (f) Trees X and Y are cospeciating perfectly. Tree Z has evolved independently.

independent speciation of Tree Z is gradually increased. The intensity of cospeciation between Trees X and Y is increased, while the intensity of cospeciation between Trees X and Z and Trees Y and Z, and their three-way interaction is decreased.

The resulting systems for each parameter combination in Table 5.4b are displayed in Figure 5.7. In System 1 Tree Z is driving the cospeciation in the system, any cospeciation between Trees X and Y is only as a direct result of their interaction with Tree Z. In System 6, Trees X and Y exhibit perfect cospeciation, while their interaction with Tree Z is random.

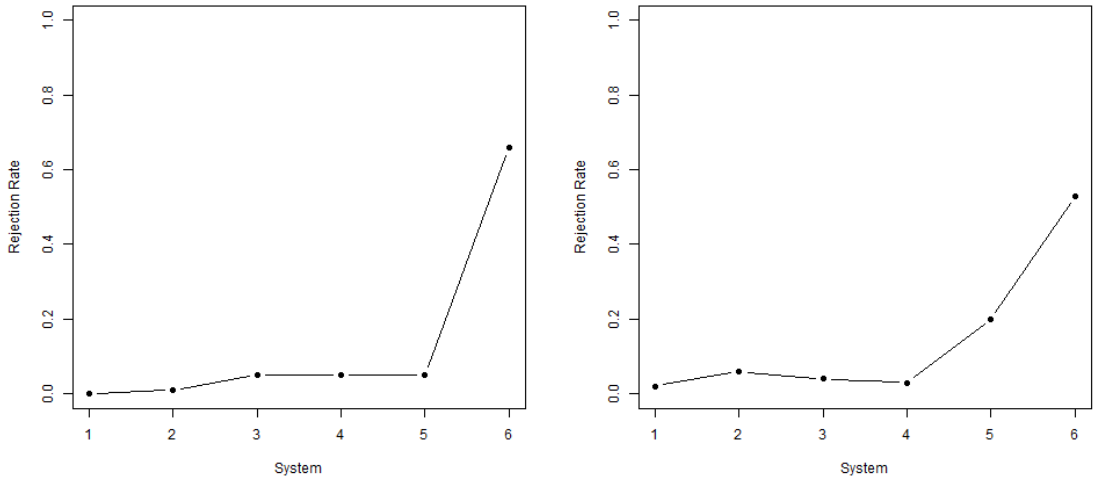
5.4.2 Rejection Rate

To test that the systems generated by our method are representing the range of systems we expect, they are testing using our tritrophic cospeciation method in Chapter 4. For each parameter combination in Table 5.4a and Table 5.4b, 100 systems are simulated. We calculate p-values with $N = 10000$ randomisations for each system. For each parameter combination we calculate the rejection rate of the null hypothesis at the $\alpha = 0.05$ significance level. The resulting plots are displayed in Figure 5.8.

The rejection rates for the first case are displayed in Figure 5.8a and the rejection rates for the second case are displayed in Figure 5.8b. In both cases, as expected, the rejection rate of the null hypothesis increases as the systems generated tend towards the alternative hypothesis. However, similarly to the bitrophic case, the increase is much slower than expected. The reasons for this are the same as in the bitrophic case. In Figure 4.15a it can be seen that our tritrophic test statistic is even more sensitive to replacing interactions than our bitrophic test statistic.

5.5 Discussion

We have introduced a method to simulate the joint evolution of complicated systems, at the bitrophic and tritrophic level. The evolution of the system is controlled by a set of parameters. Experimenting with different input values allows a wide range of systems with different cospeciation properties to be simulated. Existing methods are limited to simulating individual phylogenetic trees, and do not take the evolution of an interacting system into account. In Chapter 4 the `rtree` function is used to generate bitrophic and tritrophic systems by simulating phylogenetic trees and separately assigning interactions. Simulating a completely random or perfectly



(a) Case 1 (b) Case 2

Figure 5.8: Rejection rate plots for the p-values generated for tritrophic systems. (a) Rejection rates for each parameter combination in Table 5.4a at the $\alpha = 0.05$ significance level. Systems range from evolving independently to Trees X and Y mutually cospeciating but having no relationship with Tree Z. (b) Rejection rates for each parameter combination in Table 5.4b at the $\alpha = 0.05$ significance level. Systems range from Tree Z driving the cospeciation in the system, to Trees X and Y cospeciating to a greater extent than implied by their relationship with Tree Z.

cospeciating system is trivial, however simulating systems between these extremes is more complex, and the approach taken in Chapter 1 does not realistically reflect how evolutionary events might occur. This was only partially achieved by altering the branch lengths and randomising the clades and interactions in a perfectly cospeciating system.

We use simulations to explore the range of systems that our algorithm is able to produce. Systems are simulated that span the range of the bitrophic and tritrophic hypotheses in Chapter 4. Our method for testing multitrophic systems for cospeciation (detailed in Chapter 4) was used to analyse the simulated systems. This revealed that our method for testing cospeciation is sensitive to large changes in the systems away from cospeciation.

Our method is simple and flexible, and can be easily adapted to include different evolutionary events such as reticulate events to produce phylogenetic networks, and cannibalistic interactions. The method is also easily generalised to produce higher order systems. The algorithm is set up to end when the desired number of leaf nodes have been generated. However, other criteria of interest may be used, such as time.

We do not explore here setting different intensities for different nodes in the system. This would allow the intensities to change over time. To calibrate parameters we used numerical simulations. However, in the tritrophic setting this was too large to achieve in the time frame. Future work could explore the effect of gaining and losing interactions in tritrophic systems.

In a previous idea, we considered allowing the intensity of speciation of a node to depend on the number of interactions it is involved in. Theoretically, if a host is being infected by multiple parasites, this may drive the host to speciate. However, the converse does not necessarily hold for parasites. Setting the method up in this way, made it very difficult for the user to calibrate the parameters to represent systems of interest.

Chapter 6

Discussion

In this thesis we used statistical methods to explore, detect, and simulate coevolution in two different biological systems. At the molecular level, we explore how coevolution can be used to predict contacts in protein structures. At the species level, we analyse and simulate the coevolution between species in interacting systems of phylogenetic trees.

In Chapter 2, we carried out an exploratory data analysis of the Trypsin family of proteins. We investigated the possibility that this family of structures contains 'anchor' residues. That is, residues where the distances between these residues and every other residue in the structure is highly conserved across all of the structures in the protein family, compared to the other distances in the structure. These anchor residues were identified from the aligned distance matrices from the structural alignment produced by MUSTANG (Konagurthu *et al.*, 2006). We performed multiple tests to determine whether the anchor residues are an evolutionary feature of the trypsin family, or an artefact introduced by MUSTANG.

The tests that proved inconclusive suggested that the MUSTANG algorithm requires further exploration to determine its reliability. Konagurthu *et al.* (2006) detail a method that only uses the coordinate information of the C_α atoms. However, MUSTANG produces two different structural alignments when supplied with all of the structural information, and separately, with only the C_α atom coordinates. This suggests that either MUSTANG must incorporate the additional structural information somewhere, or that it is unreliable. The MUSTANG method does not explain how the other structural information is used, therefore a more detailed examination of the MUSTANG algorithm is required. In one test we generated a sample of artificial structures consisting only of C_α atoms. However, we discovered

that our method for generating the artificial structures was distorting the distances significantly and thus not producing a representative protein family. In addition, as discussed previously, the C_α coordinates alone produce a different alignment that is not representative of the full atom case.

If MUSTANG is reliable, aligning a large number of protein families would give insight into what range of divergences might be expected between the structures in a typical family. In this context, we would be able to re-evaluate the significance of the anchor residues in the trypsin family.

We developed a logistic regression model in Chapter 3 to identify coevolving columns in protein multiple sequence alignments. We applied our model to different artificial alignments to determine if a universally robust range of values for the elastic-net penalty, α, and the regularisation parameter, λ, could be found. We only had time to explore small simulated datasets with 30 and 50 columns, and 20, 50 and 100 sequences. For alignments with more than 50 sequences our model successfully identified the known coevolving columns in 100% of datasets, even when the amount of noise added to the coevolving columns and number of coevolving columns were varied. However, even when the number of sequences is very low our model was successful in 85% of datasets.

When applied to a selection of biological datasets we obtain mixed results. For 3 of the 7 alignments there is a combination of α and λ for which over 80% of the columns identified as coevolving by our model are in contact in three-dimensional space. The remaining 4 alignments perform slightly worse; 40–50% of the columns identified as coevolving are in contact.

It would be interesting to determine whether these predicted contacts correspond to short, medium or long range contacts, as defined in Section 3.1.1. This could then be extended to analyse the proportion of predicted short, medium and long range residues that are in contact. There was only time to apply our model to a small number of biological datasets, future work would include applying our model to a larger range of alignments.

Exploring how our method depends on the number of control sequences would also be intriguing. Four of the Pfam alignments consist of over 2000 sequences. If these sequences are from a diverse range of species it is reasonable to assume that multiple coevolution events may have occurred between a pair of residues. We did not explore this possibility in our artificial alignments, however it would be interesting to explore how our model is affected by multiple coevolution events.

In the second part of this thesis we analyse coevolution at the macroscopic level. In Chapter 4 we introduced a method to test for cospeciation in tritrophic ecological systems. We showed that our method performs comparably to a leading bitrophic method (Hommola *et al.*, 2009), and outperforms the only tritrophic method (Mramba *et al.*, 2013) that we are aware of. We do not explore scenarios where our method could be applied to higher order ecological systems. There is also scope to explore applications for our method in other complex network-like systems.

In all of the simulations, the distance on the interactions were constant and weighted equally to the branches on the phylogenetic trees in the system. Interactions are typically represented by a binary system; they either exist or they do not. By allowing the interactions to be given different distances, they can be weighted according to the likelihood of being observed. The interactions can also be weighted to give the trees or the interactions more weight as desired.

The simulation methods used to test our method are very limited and do not realistically reflect the range of coevolutionary scenarios we are interested in. The trees and interactions are simulated separately and do not take into account how the system has evolved. For example, in a bitrophic system, each tree is randomly simulated or they can be set to be identical. There is no compromise between these extremes. The interactions are placed completely at random, or in corresponding positions and gradually replaced at random to produce a slightly less perfect system. The number of external nodes on each tree, and the number of interactions must be selected in advance. These limitations are even more pronounced at the tritrophic level. To address this, we introduced a more sophisticated method for simulating these systems in Chapter 5. There are a number of small adjustments that could be incorporated into our model to tailor it to different user criteria. Currently, the simulation ends when the total number of desired leaf nodes has been reached. Alternatively, the simulation could terminate after a chosen number of time steps, or number of events.

The evolutionary events studied could easily be extended to include reticulate events to produce phylogenetic networks, and cannibalistic interactions. In addition, outside environmental or ecological pressures could be explored such as mass extinction events. Systems could be complicated further by analysing the effect of setting different rates for the nodes on a tree. For example, the rates of different evolutionary events could be set to change with time. It would also be interesting to explore simulating more than three trophic levels.

In the tritrophic case, we set the rates of gaining and losing interactions between each pair of trees to be the same. This was sufficient for testing our algorithm but

setting the rates differently for each pair of trees would be more realistic and worth exploring in future work.

Appendix A

Protein Family Selection

Before selecting the trypsin protein family, a range of possible families were considered. We required a family with a large number of sequences, and a reasonable number of structures experimentally determined, from a range of species.

The first families considered were DNA clamp loaders and DNA clamps. DNA clamp loaders load clamp proteins onto their associated DNA template strands and then disassemble the clamps after replication has been completed. Clamp proteins bind DNA polymerase to the DNA strand. The presence of DNA clamps and clamp loaders increases the rate of DNA synthesis up to 1000 fold. To carry out their function DNA clamp loaders have a specific structure composed of 5 subunits. Neuwald (2007) analysed the DNA clamp loader Replication Factor C, however only one structure has been determined for this protein. Protein domains within the subunits of the DNA clamp loaders were also considered, particularly the AAA family of ATPases. However these structures were all genetically manipulated. PCNA is a DNA clamp, again none of the structures corresponding to this protein were suitable as they are also genetically manipulated. Zinc fingers and leucine zippers are also protein families whose structures are essential to their function, however the structures determined are low quality and solved using NMR.

Marks *et al.* (2011) successfully predicted the structure of a number of protein families to 2.7-4.8Å C_α-RMSD error relative to the experimentally determined structures. The C_α atom is the backbone carbon atom to which the side chain of the residue is bonded. Thus C_α root mean squared deviation (RMSD) is a measure of the average distance between the C_α atoms of the predicted protein structure and the experimental structure. They applied their method to protein families with more than 1000 aligned sequences and at least one experimentally determined

three-dimensional structure. Due to their relative success, the families they used are suitable for analysis. First, we considered the Kunitz domain, before finally deciding on the trypsin protein family.

Kunitz Domain

The Kunitz Bovine pancreatic trypsin inhibitor domain was chosen first due to the relatively small size of the protein; around 50-60 residues. Protein domains can vary largely in size, for example the E-selectin domain consists of 36 residues, while lipoxygenase-1 consists of 692 residues. However Jones *et al.* (2008) found that 80% of protein domains tend to be less than 200 residues in size. Typically shorter domains are found on short polypeptide chains, or in multidomain proteins (Wheelan *et al.*, 2000).

The multiple sequence alignment contained sequences from a wide range of eukaryote species, and even bacteria and virus species. However, the experimentally determined structures were only from cattle and humans. As a result, the structures are identical, with 100% sequence similarity, that is every residue in every position is conserved across all of the sequences. Therefore the structures are not diverse enough to explore how they have evolved.

Appendix B

Trypsin Data

PDB ID	Non Domain Chains	Domain Range	Species
1S5S		16-238	Pig
1JIR		16-238	Cow
2QAA:A		16-236	Streptomyces Griseu
2OUA:A		47-238	Nocardiopsis Alba
2ZPQ:A		1-215	Chum Salmon
3Q76:A		16-238	Human
3K9X:D	A,C	16-238	Human
1K1I		16-238	Cow
1K1J		16-238	Cow
1L0Z		1-233	Pig
1LKA		1-233	Pig
1LPG:B	A	16-238	Human
1LQE		16-238	Cow
1LVY		16-238	Pig
1MAX		16-238	Cow
1MAY		16-238	Cow
1MKX:H	L	16-238	Cow
1MQ5:A	L	16-238	Human
1MTS		16-238	Cow
1MTU		16-238	Cow
1N6X		16-238	Cow
1N6Y		16-238	Cow
1NFU:A	B	16-238	Human
1NFW:A	B	16-238	Human
1NFX:A	B	16-238	Human
1O2I		16-238	Cow
1O2J		16-238	Cow
1O2M		16-238	Cow
1O2N		16-238	Cow

1O2O		16-238	Cow
1O2Q		16-238	Cow
1O2S		16-238	Cow
1O2T		16-238	Cow
1O2U		16-238	Cow
1O30		16-238	Cow
1O33		16-238	Cow
1O35		16-238	Cow
1O37		16-238	Cow
1O3D		16-238	Cow
1OP0		16-238	Snake
1P3E		7-213	Bacillus Intermediu
1PPZ		16-235	Fungus
1PQ5		16-235	Fungus
1PQ7		16-235	Fungus
1PQA		16-235	Fungus
1S6F		16-238	Pig
1S81		16-238	Pig
1UO6		1-233	Pig
1UTJ		16-238	Atlantic Salmon
1UTK		16-238	Atlantic Salmon
1UTL		16-238	Atlantic Salmon
1UTM		16-238	Atlantic Salmon
1V3X:A	B	16-238	Human
1XVO		16-235	Fungus
1Z6E:A	L	16-238	Human
2OQU		1-233	Pig
2OUA:B		47-238	Nocardiopsis Alba
2P3U:B	A	16-238	Human
2P95:A	L	16-238	Human
2Q1J:A	B	16-238	Human
2QAA:B		16-236	Streptomyces Griseu
2UWL:A	B	16-238	Human
2UWO:A	B	16-238	Human
2V0B		1-233	Pig
2V35		16-238	Pig
2VH6:A	B	16-238	Human
2W26:A	B	16-238	Human
2Y7Z:A	B	16-238	Human
2Y81:A	B	16-238	Human
2ZPQ:B		1-215	Chum Salmon
2ZPS		1-215	Chum Salmon
3K9X:B	A,C	16-238	Human
3KQC:A	L	16-238	Human
3MNB		16-238	Pig

3MNC		16-238	Pig
3MNS		16-238	Pig
3MO3		16-238	Pig
3MOC		16-238	Pig
3MTY		16-238	Pig
3MU0		16-238	Pig
3ODF		16-238	Pig
3Q76:B		16-238	Human
3TGI:E	I	16-238	Rat

Table B.1: Sample of 83 trypsin chains.

Appendix C

Multidimensional Scaling

Multidimensional scaling is a technique used to construct a configuration of data points in Euclidean space using the distances, similarities or dissimilarities between them. The data points are assigned coordinates in n dimensions that aim to preserve the distances between them. Define $P_1, \ldots, P_n$ to be the unknown coordinates of the n data points, then $\widehat{D}$ is the distance matrix corresponding to the set of points P and is similar to D, the distance matrix corresponding to the original data points. The method of metric multidimensional scaling is as follows:

- Construct a matrix A from the distance matrix D:

$$A = (-\frac{1}{2}d_{rs}^2).$$

- Use A to calculate the matrix B with elements

$$b_{rs} = a_{rs} - \overline{a_{r\bullet}} - \overline{a_{\bullet s}} + \overline{a_{\bullet\bullet}},$$

  where

$$\overline{a_{r\bullet}} = \frac{1}{n}\sum_{s=1}^{n} a_{rs}$$
$$\overline{a_{\bullet s}} = \frac{1}{n}\sum_{r=1}^{n} a_{rs}$$
$$\overline{a_{\bullet\bullet}} = \frac{1}{n^2}\sum_{r,s=1}^{n} a_{rs}.$$

The matrix B is calculated using $B = HAH$, where the nxn centering matrix H is given by

$$H = I - n^{-1}\mathbf{1}\mathbf{1}^T,$$

where $\mathbf{1}$ is a vector of ones.

- Find the k largest eigenvalues $\lambda_1 > \ldots > \lambda_k$ of B, with corresponding normalised eigenvectors $X = \left(X_{(1)}, \ldots, X_{(k)}\right)$.
- The coordinates of P_r are $X_r = (X_{r1}, \ldots, X_{rp})^T$, where $r = 1, \ldots, k$ are the rows of X.

If the first k eigenvalues of B are large and positive and all of the other eigenvalues are close to zero then the interpoint distances of the configuration should closely approximate the original distance matrix D (Mardia *et al.*, 1979).

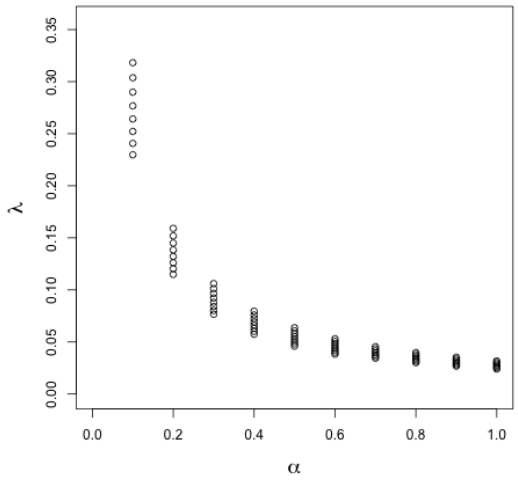
Appendix D

Additional Figures for Chapter 3

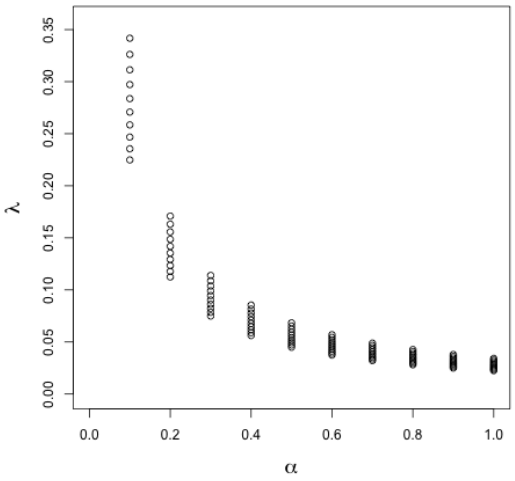
D.1 Simulated Alignments: 30 Columns, 20 Sequences

Figure D.1 displays the combinations of α and λ that identify the coevolving column scores, $s_{k,l}$, as the only non-zero scores. Each plot corresponds to a different number of coevolving columns; each with 5% noise added to the coevolving columns. Figures D.2, D.3 and D.4 display the combinations of α and λ that identify the coevolving column scores, $s_{k,l}$, as the only non-zero scores, for the 10%, 15% and 20% noise cases. Each plot corresponds to a different number of coevolving columns.

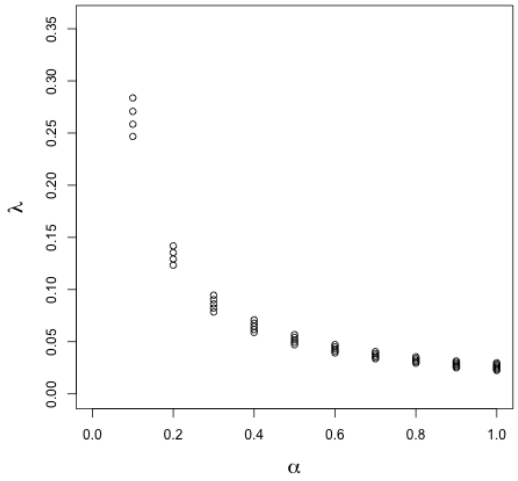
For each noise level there is no obvious pattern as the number of coevolving columns changes.



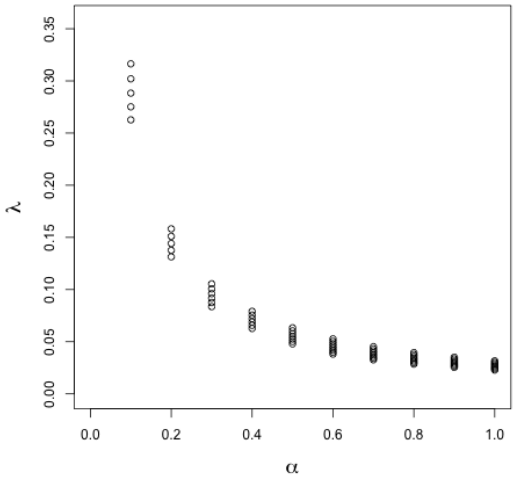
(a) Coevolving pairs=1, noise=0.05



(b) Coevolving pairs=2, noise=0.05

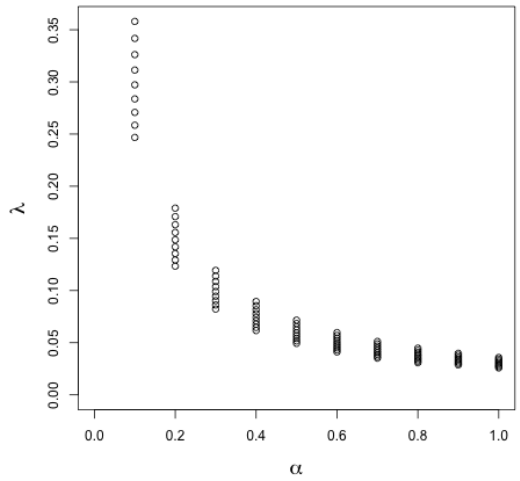


(c) Coevolving pairs=3, noise=0.05

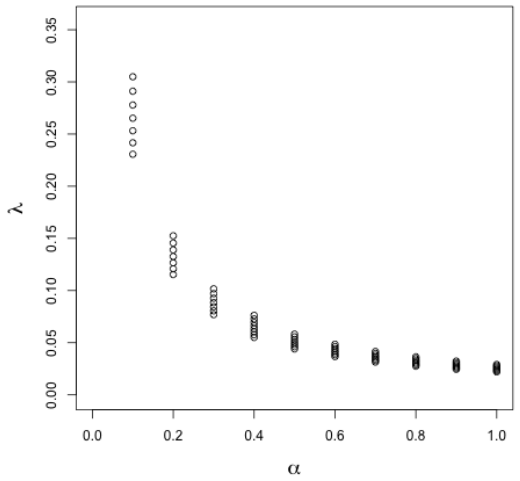


(d) Coevolving pairs=4, noise=0.05

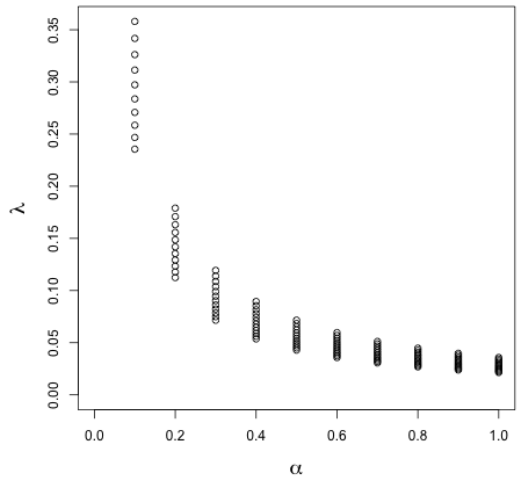
Figure D.1: Values of the elastic-net parameter, α, and the regularisation parameter, λ, that identify the coevolving column scores as the only non-zero scores for the case where 5% noise is added to the coevolving columns. Each plot corresponds to an alignment, each consisting of 30 columns and 20 sequences. The number of coevolving column pairs differs for each plot. (a) 1 coevolving pair of columns. (b) 2 coevolving pairs of columns. (c) 3 coevolving pairs of columns. (d) 4 coevolving pairs of columns.



(a) Coevolving pairs=1, noise=0.1

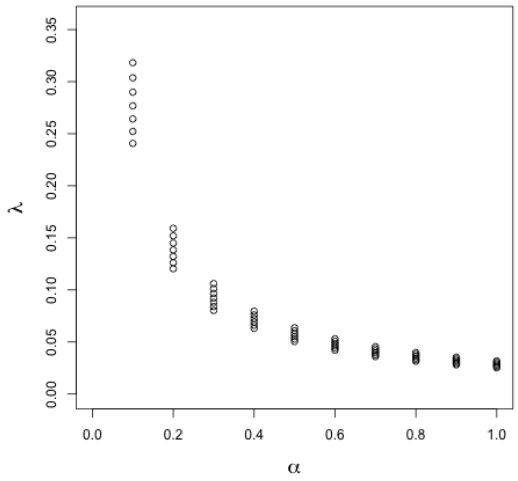


(b) Coevolving pairs=2, noise=0.1

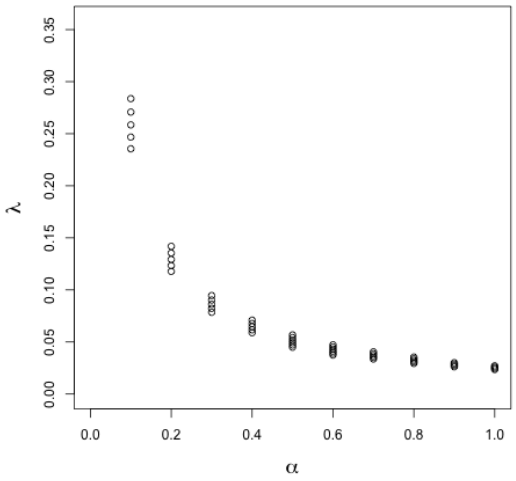


(c) Coevolving pairs=3, noise=0.1

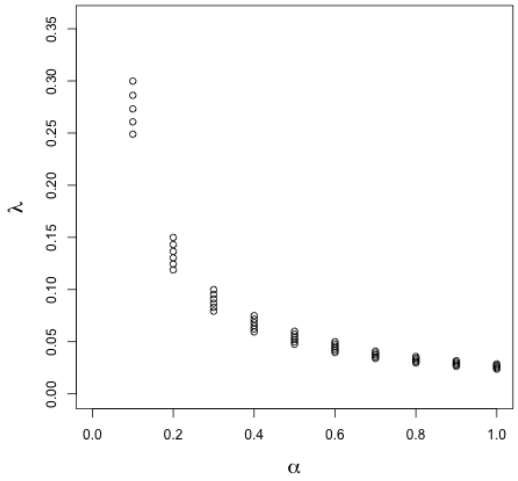
Figure D.2: Values of the elastic-net parameter, α, and the regularisation parameter, λ, that identify the coevolving column scores as the only non-zero scores for the case where 10% noise is added to the coevolving columns. Each plot corresponds to an alignment, each consisting of 30 columns and 20 sequences. The number of coevolving column pairs differs for each plot. (a) 1 coevolving pair of columns. (b) 2 coevolving pairs of columns. (c) 3 coevolving pairs of columns. There is no plot for the case with 4 coevolving pairs of columns as there was no combination of α and λ that identified only the true coevolving column scores as non-zero (see Section 3.3.1).



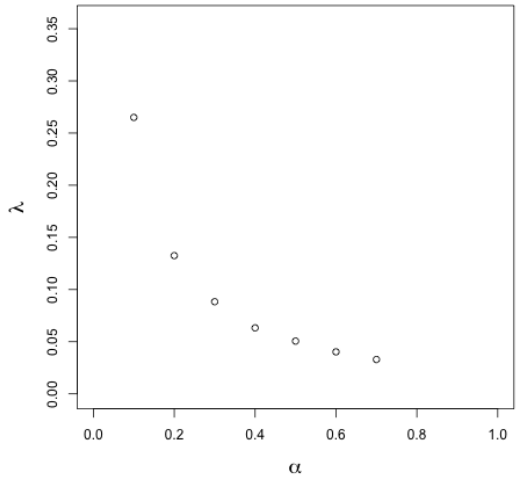
(a) Coevolving pairs=1, noise=0.15



(b) Coevolving pairs=2, noise=0.15

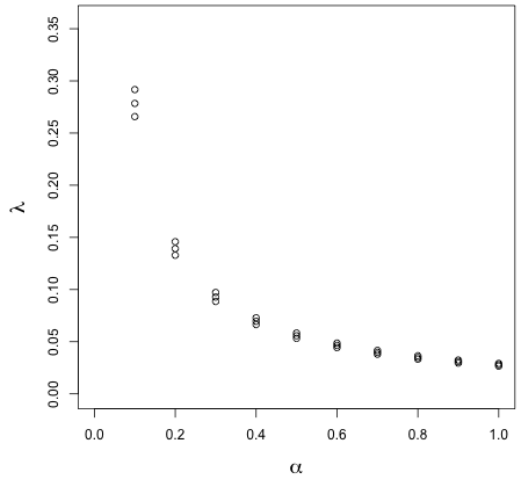


(c) Coevolving pairs=3, noise=0.15

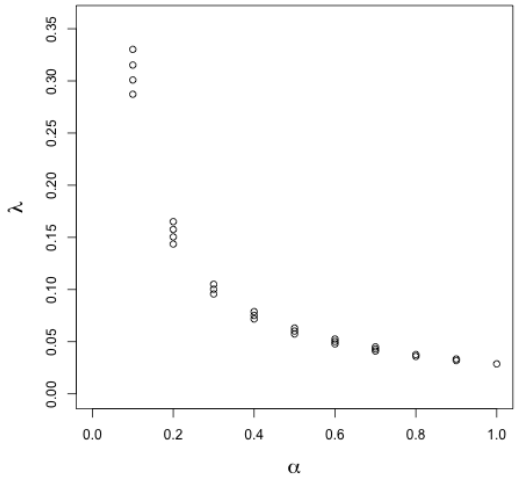


(d) Coevolving pairs=4, noise=0.15

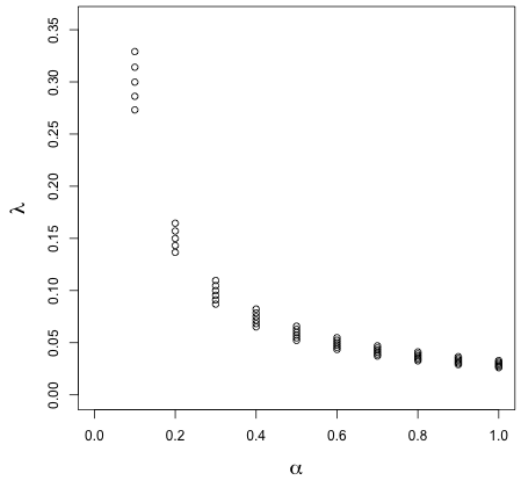
Figure D.3: Values of the elastic-net parameter, α, and the regularisation parameter, λ, that identify the coevolving column scores as the only non-zero scores for the case where 15% noise is added to the coevolving columns. Each plot corresponds to an alignment, each consisting of 30 columns and 20 sequences. The number of coevolving column pairs differs for each plot. (a) 1 coevolving pair of columns. (b) 2 coevolving pairs of columns. (c) 3 coevolving pairs of columns. (d) 4 coevolving pairs of columns.



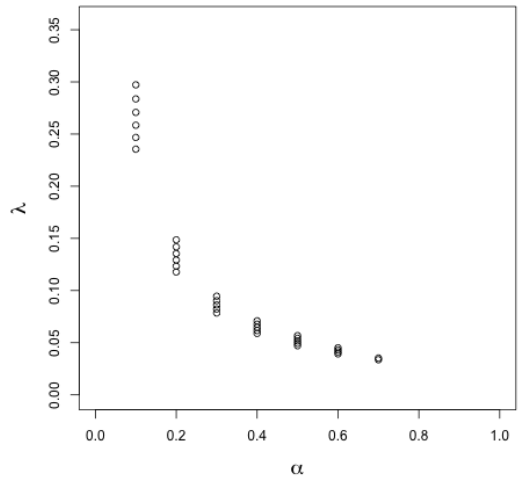
(a) Coevolving pairs=1, noise=0.2



(b) Coevolving pairs=2, noise=0.2



(c) Coevolving pairs=3, noise=0.2



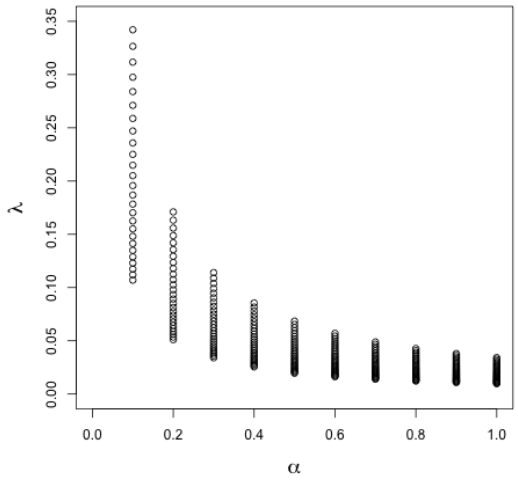
(d) Coevolving pairs=4, noise=0.2

Figure D.4: Values of the elastic-net parameter, α, and the regularisation parameter, λ, that identify the coevolving column scores as the only non-zero scores for the case where 20% noise is added to the coevolving columns. Each plot corresponds to an alignment, each consisting of 30 columns and 20 sequences. The number of coevolving column pairs differs for each plot. (a) 1 coevolving pair of columns. (b) 2 coevolving pairs of columns. (c) 3 coevolving pairs of columns. (d) 4 coevolving pairs of columns.

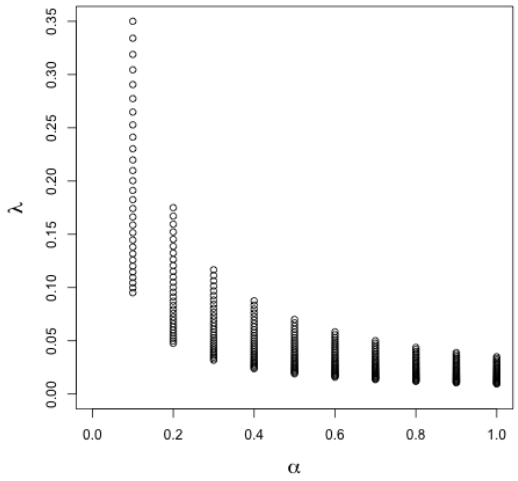
D.2 Simulated Alignments: 30 Columns, 100 Sequences

Using the same parameter combinations for the coevolving columns as the 20-sequence case, the number of sequences is increased to 100. For every alignment there are multiple optimal combinations of α and λ that successfully identify the coevolving column scores as the only non-zero scores. Figure D.5 displays the combinations of α and λ that identify the coevolving column scores, $s_{k,l}$, as the only non-zero scores. Each plot corresponds to a different number of coevolving columns; each with 5% noise added to the coevolving columns. Figures D.6, D.7 and D.8 display the combinations of α and λ that identify the coevolving column scores, $s_{k,l}$, as the only non-zero scores, for the 10%, 15% and 20% noise cases. Each plot corresponds to a different number of coevolving columns.

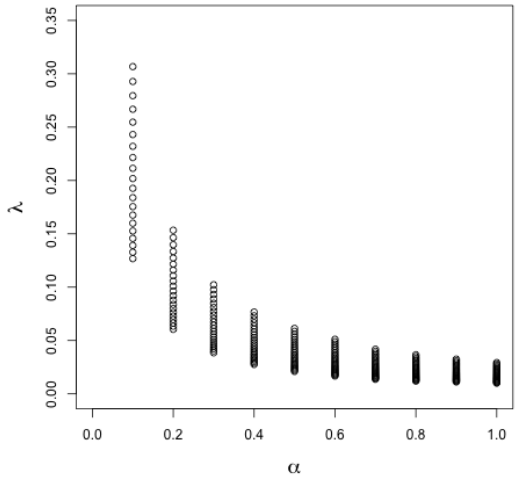
Similarly to the 20-sequence case, there appears to be no pattern in the α/λ combinations as the number of coevolving pairs or percentage of noise increases. However, the number of optimal combinations of α and λ is larger, and the range of optimal λ values is larger for each value of α.



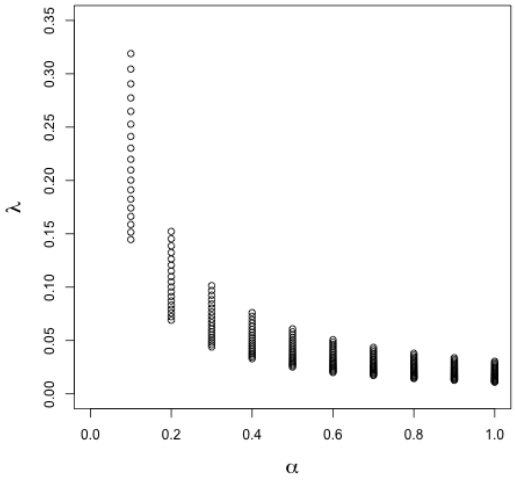
(a) Coevolving pairs=1, noise=0.05



(b) Coevolving pairs=2, noise=0.05

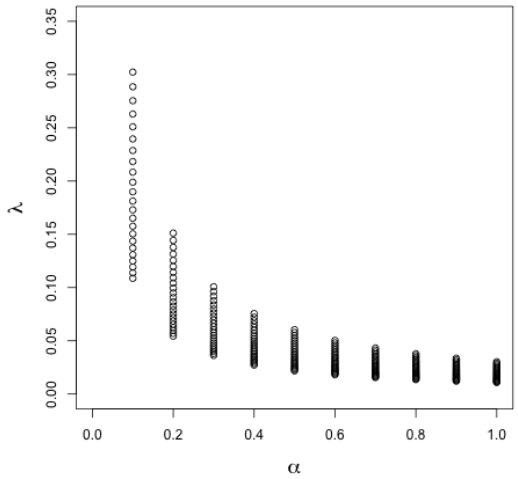


(c) Coevolving pairs=3, noise=0.05

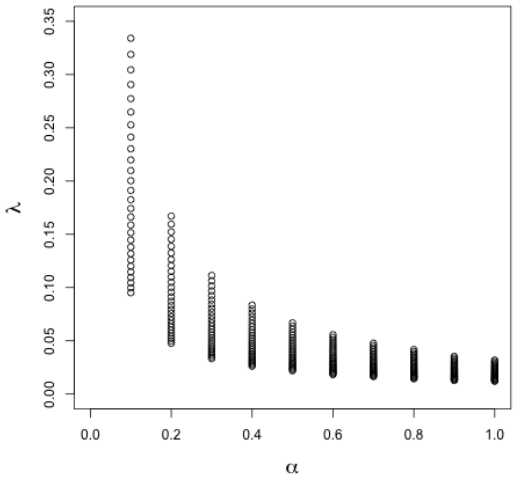


(d) Coevolving pairs=4, noise=0.05

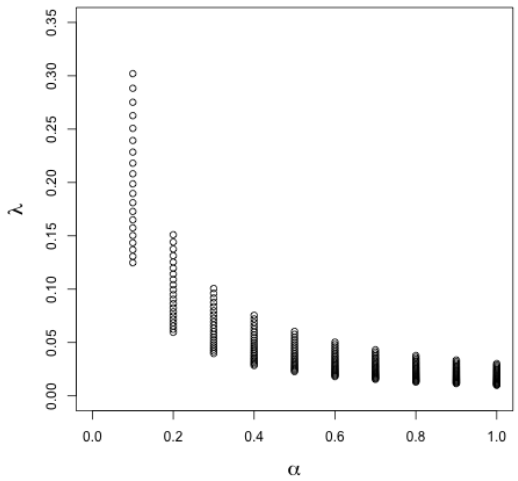
Figure D.5: Values of the elastic-net parameter, α, and the regularisation parameter, λ, that identify the coevolving column scores as the only non-zero scores for the case where 5% noise is added to the coevolving columns. Each plot corresponds to an alignment, each consisting of 30 columns and 100 sequences. The number of coevolving column pairs differs for each plot. (a) 1 coevolving pair of columns. (b) 2 coevolving pairs of columns. (c) 3 coevolving pairs of columns. (d) 4 coevolving pairs of columns.



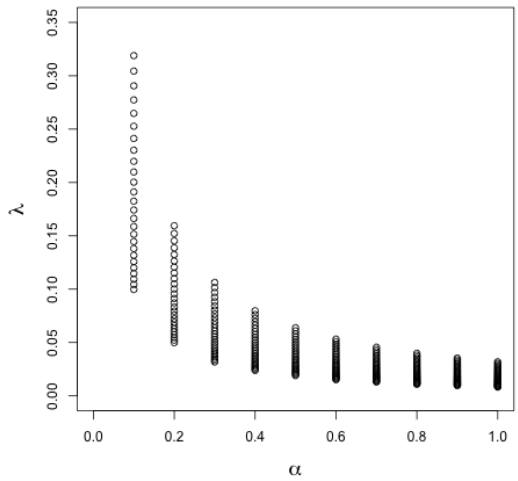
(a) Coevolving pairs=1, noise=0.1



(b) Coevolving pairs=2, noise=0.1

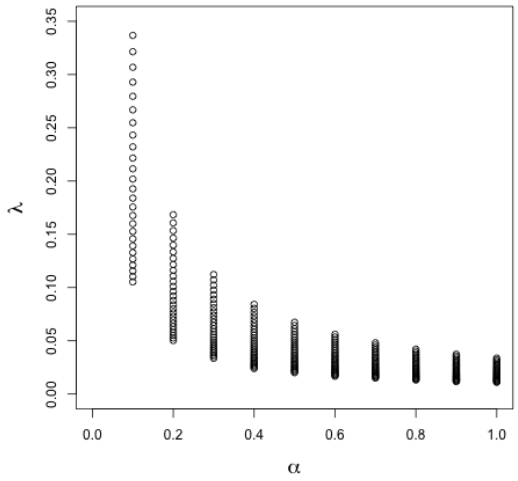


(c) Coevolving pairs=3, noise=0.1

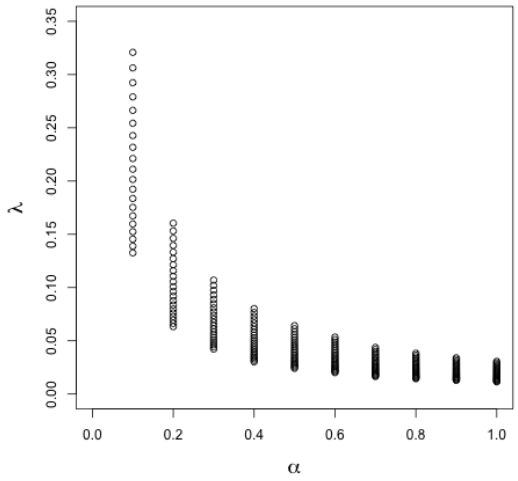


(d) Coevolving pairs=4, noise=0.1

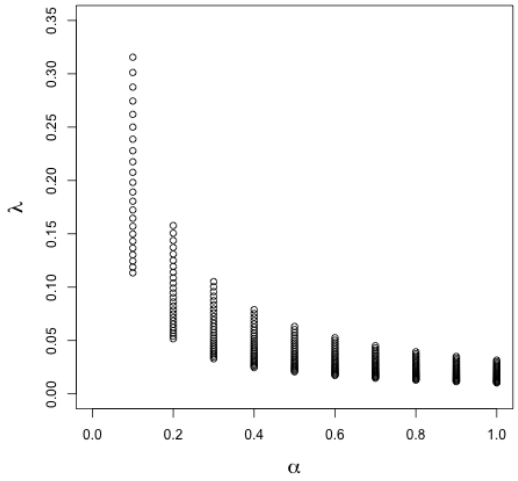
Figure D.6: Values of the elastic-net parameter, α, and the regularisation parameter, λ, that identify the coevolving column scores as the only non-zero scores for the case where 10% noise is added to the coevolving columns. Each plot corresponds to an alignment, each consisting of 30 columns and 100 sequences. The number of coevolving column pairs differs for each plot. (a) 1 coevolving pair of columns. (b) 2 coevolving pairs of columns. (c) 3 coevolving pairs of columns. (d) 4 coevolving pairs of columns.



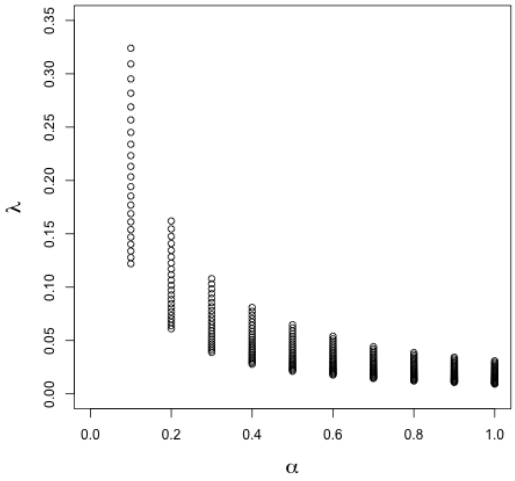
(a) Coevolving pairs=1, noise=0.15



(b) Coevolving pairs=2, noise=0.15

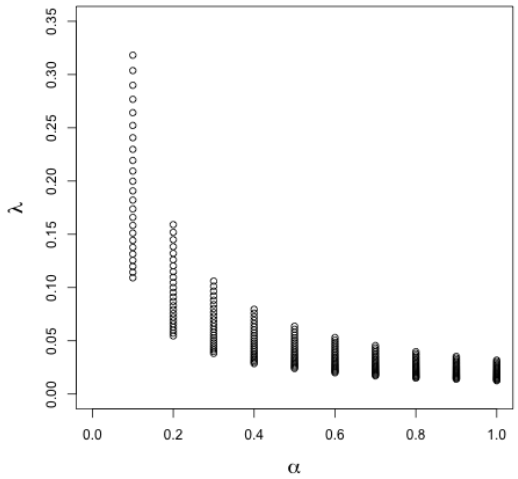


(c) Coevolving pairs=3, noise=0.15

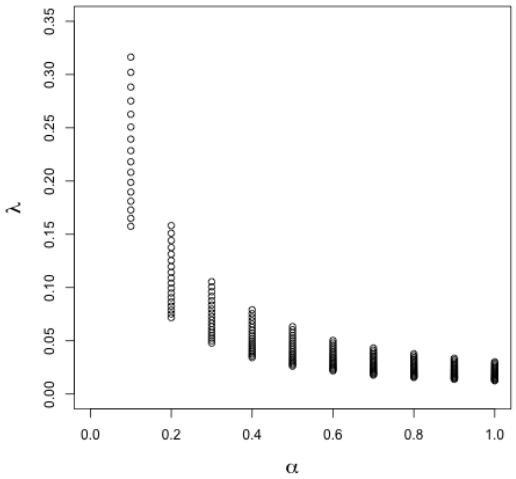


(d) Coevolving pairs=4, noise=0.15

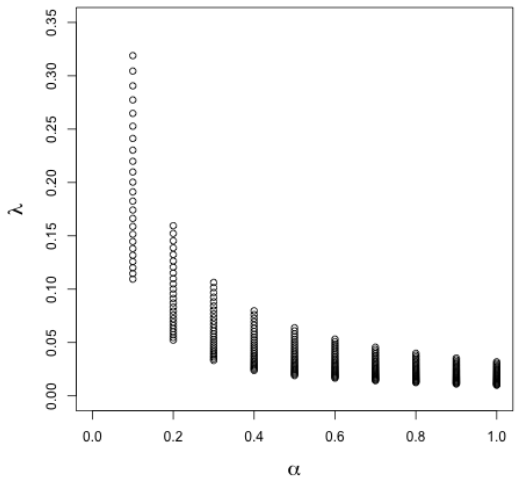
Figure D.7: Values of the elastic-net parameter, α, and the regularisation parameter, λ, that identify the coevolving column scores as the only non-zero scores for the case where 15% noise is added to the coevolving columns. Each plot corresponds to an alignment, each consisting of 30 columns and 100 sequences. The number of coevolving column pairs differs for each plot. (a) 1 coevolving pair of columns. (b) 2 coevolving pairs of columns. (c) 3 coevolving pairs of columns. (d) 4 coevolving pairs of columns.



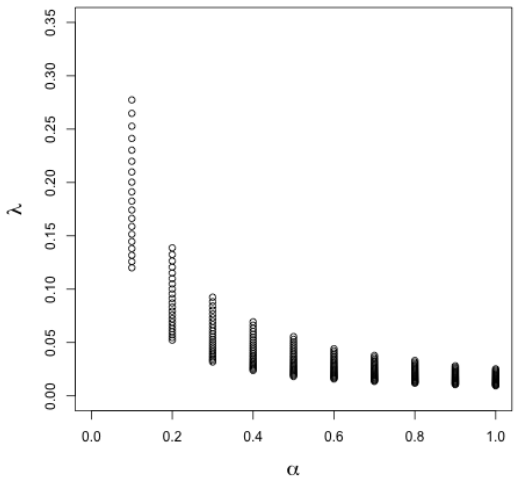
(a) Coevolving pairs=1, noise=0.2



(b) Coevolving pairs=2, noise=0.2



(c) Coevolving pairs=3, noise=0.2



(d) Coevolving pairs=4, noise=0.2

Figure D.8: Values of the elastic-net parameter, α, and the regularisation parameter, λ, that identify the coevolving column scores as the only non-zero scores for the case where 20% noise is added to the coevolving columns. Each plot corresponds to an alignment, each consisting of 30 columns and 100 sequences. The number of coevolving column pairs differs for each plot. (a) 1 coevolving pair of columns. (b) 2 coevolving pairs of columns. (c) 3 coevolving pairs of columns. (d) 4 coevolving pairs of columns.

Appendix E

Response Matrix Calculation Details for Chapter 4

The first step in calculating the response matrix for the partitioned matrices in Equations (4.5), (4.7) and (4.8) is to invert D.

E.1 Inverting D

Suppose D is partitioned into four matrices as follows

$$D = \begin{pmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{pmatrix},$$

where the partitions are given by

$$\begin{aligned} D_{11} &= \begin{pmatrix} \Delta_{L_X} & 0 \\ 0 & \Delta_{L_Y} \end{pmatrix}, \\ D_{12} &= \begin{pmatrix} -\Gamma_X & 0 \\ 0 & -\Gamma_Y \end{pmatrix}, \\ D_{21} &= \begin{pmatrix} -\Gamma_X^T & 0 \\ 0 & -\Gamma_Y^T \end{pmatrix}, \\ D_{22} &= \begin{pmatrix} \Gamma_{C_X} & 0 \\ 0 & \Gamma_{C_Y} \end{pmatrix}. \end{aligned}$$

We use the following notation for the partitions of the matrix inverse of D.

$$D^{-1} = \begin{pmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{pmatrix}^{-1} = \begin{pmatrix} D^{11} & D^{12} \\ D^{21} & D^{22} \end{pmatrix}$$

Using the blockwise inversion formula, the partitions of the inverse of D, are given by

$$\begin{aligned} D^{11} &= D_{11}^{-1} + D_{11}^{-1} D_{12} (D_{22} - D_{21} D_{11}^{-1} D_{12})^{-1} D_{21} D_{11}^{-1}, \\ D^{12} &= -D_{11}^{-1} D_{12} (D_{22} - D_{21} D_{11}^{-1} D_{12})^{-1}, \\ D^{21} &= -(D_{22} - D_{21} D_{11}^{-1} D_{12})^{-1} D_{21} D_{11}^{-1}, \\ D^{22} &= (D_{22} - D_{21} D_{11}^{-1} D_{12})^{-1}. \end{aligned} \tag{E.1}$$

Using Equation (E.1), the four partitions of D^{-1} are given by

$$D^{11} = \begin{pmatrix} \frac{1}{\Delta_{L_X}} + \frac{1}{\Delta_{L_X}} \Gamma_X (\Gamma_{C_X} - \Gamma_X^T \frac{1}{\Delta_{L_X}} \Gamma_X)^{-1} (\Gamma_X^T \frac{1}{\Delta_{L_X}}) & 0 \\ 0 & \frac{1}{\Delta_{L_Y}} + \frac{1}{\Delta_{L_Y}} \Gamma_Y (\Gamma_{C_Y} - \Gamma_Y^T \frac{1}{\Delta_{L_Y}} \Gamma_Y)^{-1} \Gamma_Y^T \frac{1}{\Delta_{L_Y}} \end{pmatrix}$$

$$D^{12} = \begin{pmatrix} \frac{1}{\Delta_{L_X}} \Gamma_X (\Gamma_{C_X} - \Gamma_X^T \frac{1}{\Delta_{L_X}} \Gamma_X)^{-1} & 0 \\ 0 & \frac{1}{\Delta_{L_Y}} \Gamma_Y (\Gamma_{C_Y} - \Gamma_Y^T \frac{1}{\Delta_{L_Y}} \Gamma_Y)^{-1} \end{pmatrix}$$

$$D^{21} = \begin{pmatrix} (\Gamma_{C_X} - \Gamma_X^T \frac{1}{\Delta_{L_X}} \Gamma_X)^{-1} \Gamma_X^T \frac{1}{\Delta_{L_X}} & 0 \\ 0 & (\Gamma_{C_Y} - \Gamma_Y^T \frac{1}{\Delta_{L_Y}} \Gamma_Y)^{-1} \Gamma_Y^T \frac{1}{\Delta_{L_Y}} \end{pmatrix}$$

$$D^{22} = \begin{pmatrix} (\Gamma_{C_X} - \Gamma_X^T \frac{1}{\Delta_{L_X}} \Gamma_X)^{-1} & 0 \\ 0 & (\Gamma_{C_Y} - \Gamma_Y^T \frac{1}{\Delta_{L_Y}} \Gamma_Y)^{-1} \end{pmatrix}$$

E.2 Calculating Λ_γ

Recall that the response matrix is calculated using the following equation

$$\Lambda_\gamma = A - B D^{-1} B^T$$

The inverse of D is comprised of 16 submatrices. To make the calculations easier, let these 16 submatrices be represented as follows

$$D^{-1} = \begin{pmatrix} D_*^{11} & D_*^{12} & D_*^{13} & D_*^{14} \\ D_*^{21} & D_*^{22} & D_*^{23} & D_*^{24} \\ D_*^{31} & D_*^{32} & D_*^{33} & D_*^{34} \\ D_*^{41} & D_*^{42} & D_*^{43} & D_*^{44} \end{pmatrix}.$$

Using this, the response matrix is calculated to be

$$\Lambda_\gamma = \begin{array}{c} \\ E_X \\ E_Y \end{array} \begin{array}{c} \begin{array}{cc} E_X & E_Y \end{array} \\ \begin{pmatrix} \left(\frac{1}{\delta}+\frac{1}{\epsilon}\right) I - \frac{1}{\epsilon^2}\left(I_X D_*^{11} I_X^T\right) & -\frac{I}{\delta} \\ -\frac{I}{\delta} & \left(\frac{1}{\delta}+\frac{1}{\epsilon}\right) I - \frac{1}{\epsilon^2}\left(I_Y D_*^{22} I_Y^T\right) \end{pmatrix} \end{array}.$$

The matrices D_*^{11} and D_*^{22} correspond to the diagonal elements of D^{11}:

$$D_*^{11} = \frac{1}{\Delta_{L_X}} + \frac{1}{\Delta_{L_X}}\Gamma_X(\Gamma_{C_X} - \Gamma_X^T\frac{1}{\Delta_{L_X}}\Gamma_X)^{-1}(\Gamma_X^T\frac{1}{\Delta_{L_X}})$$
$$D_*^{22} = \frac{1}{\Delta_{L_Y}} + \frac{1}{\Delta_{L_Y}}\Gamma_Y(\Gamma_{C_Y} - \Gamma_Y^T\frac{1}{\Delta_{L_Y}}\Gamma_Y)^{-1}\Gamma_Y^T\frac{1}{\Delta_{L_Y}}$$

E.3 Calculating Λ_γ for Tree X and Tree Y separately

In Section 4.3 the calculation of the statistic is worked through using the properties of the phylogenetic system. Here, we show that these calculations are equivalent to calculating two separate response matrices, one for each tree individually.

The Kirchhoff matrix, K_X, calculated only for the nodes in Tree X, using the partitions defined in Figure 4.10, is given by

$$K_X = \begin{array}{c} \\ E_X \\ L_X \\ C_X \end{array} \begin{array}{c} \begin{array}{ccc} E_X & L_X & C_X \end{array} \\ \begin{pmatrix} \frac{I}{\epsilon} & -\frac{I_X}{\epsilon} & 0 \\ -\frac{I_X^T}{\epsilon} & \Delta_{L_X} & -\Gamma_X \\ 0 & -\Gamma_X^T & \Gamma_{C_X} \end{pmatrix} \end{array},$$

where I is the identity matrix and I_X is a binary matrix containing the connections between the external nodes and the leaf nodes of Tree X. Each connection in I_X has conductance $\frac{1}{\epsilon}$. The matrix Γ_X contains the conductances on the connections between the leaf nodes and the internal nodes on each tree. Δ_{L_X} is a diagonal matrix that represents the unknown conductances on the diagonal of K_X, and Γ_{C_X} is a symmetric matrix containing the negative conductances between the internal nodes on Tree X. The Kirchhoff matrix, K_Y, for Tree Y will have the same general structure as K_X.

The constraints for K_X and K_Y are the same as those for the Kirchhoff matrix for the whole system, with the exception of those involving the external nodes. This

constraint no longer depends on the center of the interactions.

$$\frac{I}{\epsilon}\mathbf{1} - \frac{I_X}{\epsilon}\mathbf{1} = \mathbf{0}$$

We partition K_X into the submatrices in Equation (4.3) as follows

$$\begin{aligned} A_X &= \left(\tfrac{I}{\epsilon}\right) \\ B_X &= \begin{pmatrix} -\frac{I_X}{\epsilon} & 0 \end{pmatrix} \\ D_X &= \begin{pmatrix} \Delta_{L_X} & -\Gamma_X \\ -\Gamma_X^T & \Gamma_{C_X} \end{pmatrix} \end{aligned}$$

The response matrix for Tree X is calculated as follows

$$\begin{aligned} \Lambda_X &= A_X - B_X D_X^{-1} B_X^T \\ &= \frac{I}{\epsilon} - \frac{1}{\epsilon^2} I_X D_X^{11} I_X^T, \end{aligned}$$

where D_X^{11} is calculated using the blockwise inversion formula and is given by

$$\begin{aligned} D_X^{11} &= D_{X_{11}}^{-1} + D_{X_{11}}^{-1} D_{X_{12}} \left(D_{X_{22}} - D_{X_{21}} D_{X_{11}}^{-1} D_{X_{12}}\right)^{-1} D_{X_{21}} D_{X_{11}}^{-1} \\ &= \frac{1}{\Delta_{L_X}} + \frac{1}{\Delta_{L_X}} \Gamma_X \left(\Gamma_{C_X} - \Gamma_X^T \frac{1}{\Delta_{L_X}} \Gamma_X\right)^{-1} \Gamma_X \frac{1}{\Delta_{L_X}} \end{aligned}$$

Therefore our statistic is calculated from the upper triangle of the following matrices, since the Equations for Tree Y are equivalent:

$$\frac{1}{\epsilon^2} I_X \left(\frac{1}{\Delta_{L_X}} + \frac{1}{\Delta_{L_X}} \Gamma_X \left(\Gamma_{C_X} - \Gamma_X^T \frac{1}{\Delta_{L_X}} \Gamma_X\right)^{-1} \Gamma_X \frac{1}{\Delta_{L_X}}\right) I_X^T$$

$$\frac{1}{\epsilon^2} I_Y \left(\frac{1}{\Delta_{L_Y}} + \frac{1}{\Delta_{L_Y}} \Gamma_Y \left(\Gamma_{C_Y} - \Gamma_Y^T \frac{1}{\Delta_{L_Y}} \Gamma_Y\right)^{-1} \Gamma_Y \frac{1}{\Delta_{L_Y}}\right) I_Y^T$$

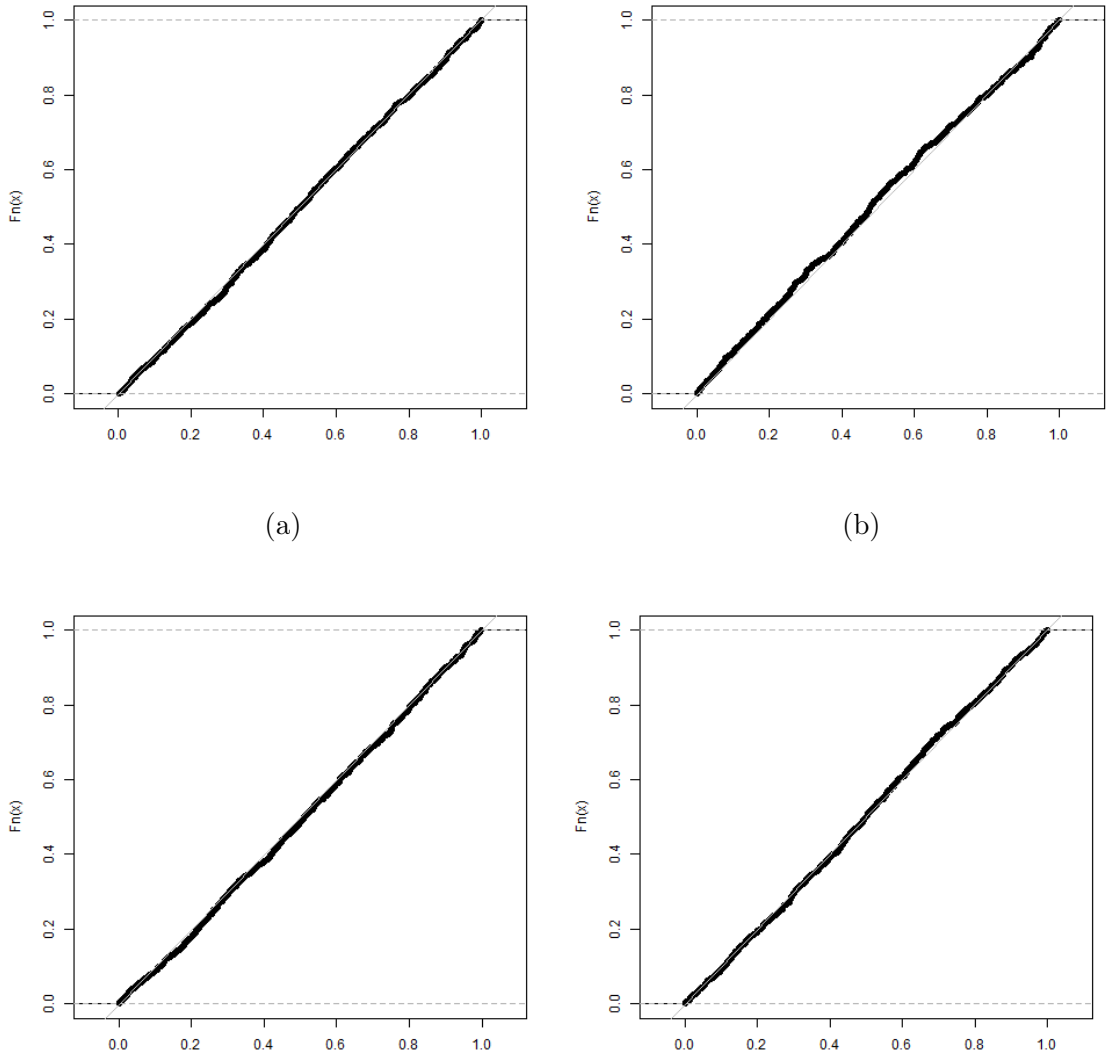
This is the same result as when the Kirchhoff matrix is calculated for the full system.

Appendix F

Additional Figures and Tables for Chapter 4

F.1 Type I Error

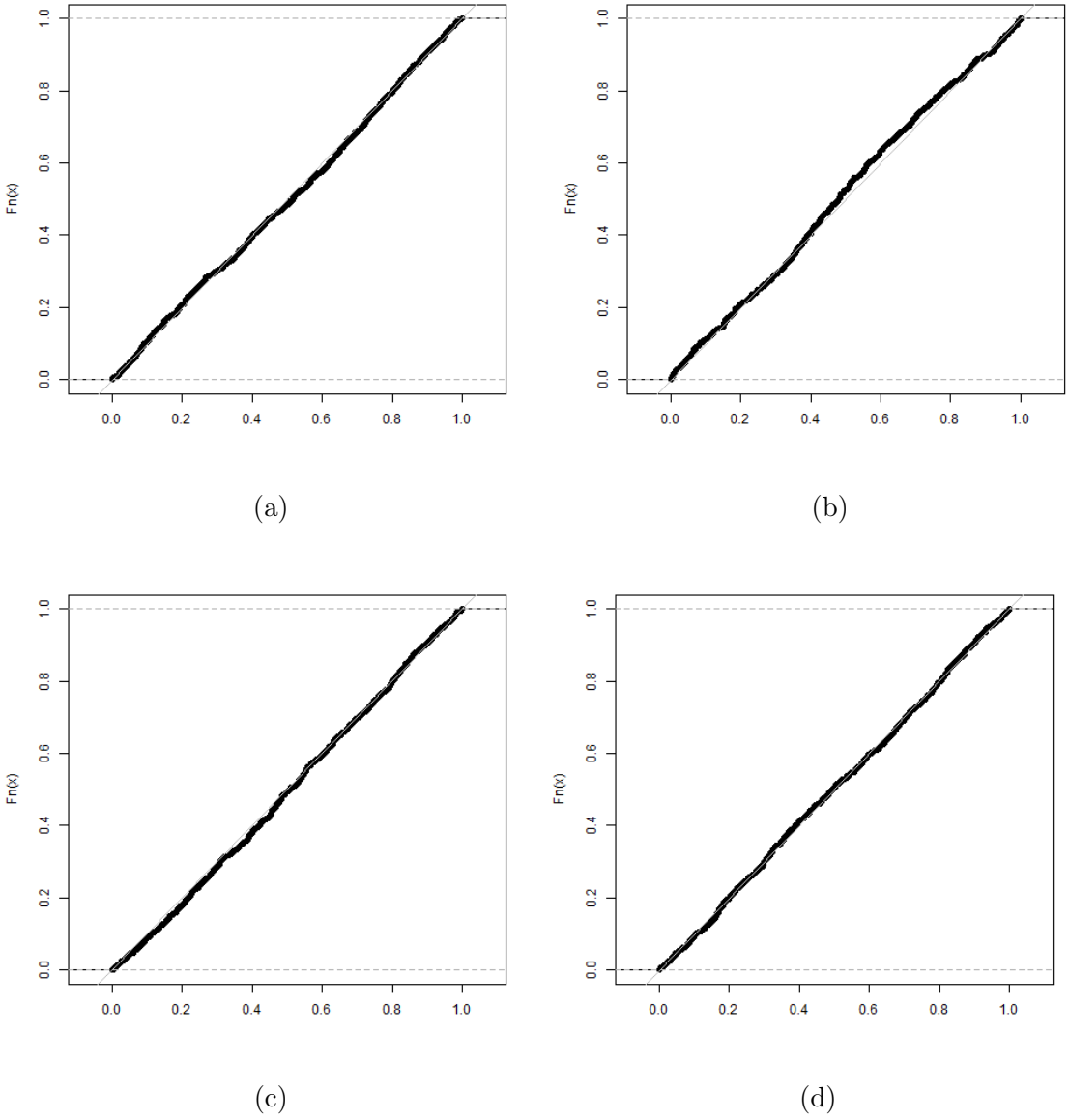
The bitrophic empirical cumulative distribution functions for the first parameter combination in Section 4.4.1 are displayed in Figure F.1. The plots for the second parameter combination are displayed in Figures F.2 and F.3. The empirical CDF for our p-values and Hommola *et al.*'s (2009) lies close to the desired diagonal line.



(a) (b)

(c) (d)

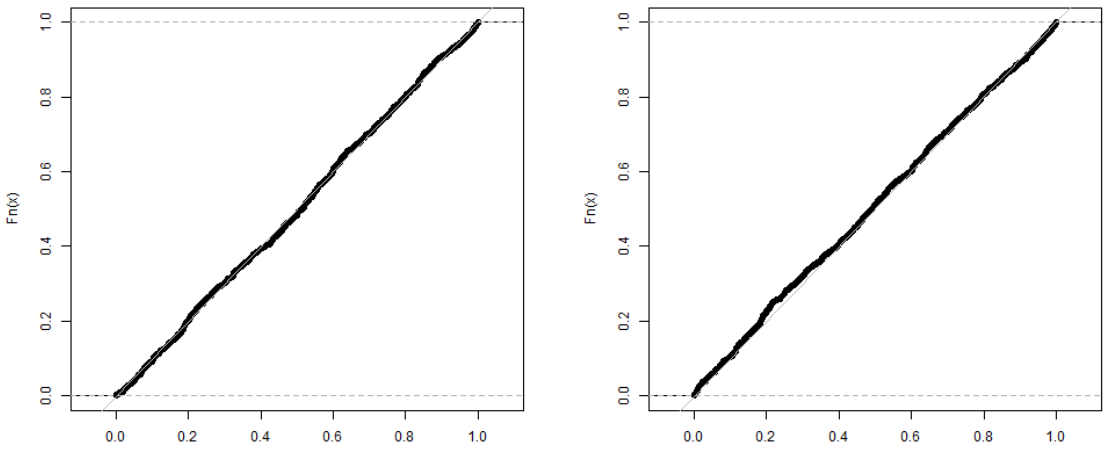
Figure F.1: Empirical cumulative distribution functions for our p-values and Hommola *et al.*'s (2009). Each plot corresponds to simulations with 10 tips on each tree. The first column corresponds to 20 interactions simulated, and the second column corresponds to 25 interactions simulated. The top row contains the p-values for our method, and the bottom row contains the p-values for the method of Hommola *et al.* (2009). The diagonal grey line is the identity line.



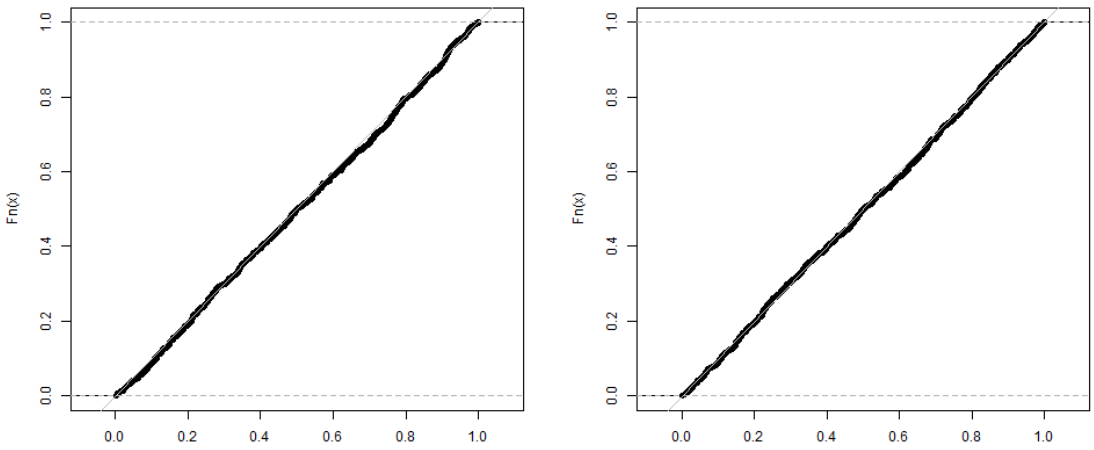
(a) (b)

(c) (d)

Figure F.2: Empirical cumulative distribution functions for our p-values. Each plot corresponds to simulations with 10 tips on Tree X and 15 tips on Tree Y. Going clockwise the plots correspond to 10, 15, 20, and 25 interactions simulated. The diagonal grey line is the identity line.



(a) (b)



(c) (d)

Figure F.3: Empirical cumulative distribution functions for Hommola *et al.*'s (2009) p-values. Each plot corresponds to simulations with 10 tips on Tree X and 15 tips on Tree Y. Going clockwise the plots correspond to 10, 15, 20, and 25 interactions simulated. The diagonal grey line is the identity line.

The tritrophic empirical cumulative distribution functions for the second parameter combination in the Type 1 Error section are displayed in Figure F.4. The empirical CDF for our p-values lies close to the desired diagonal line for all parameter combinations.

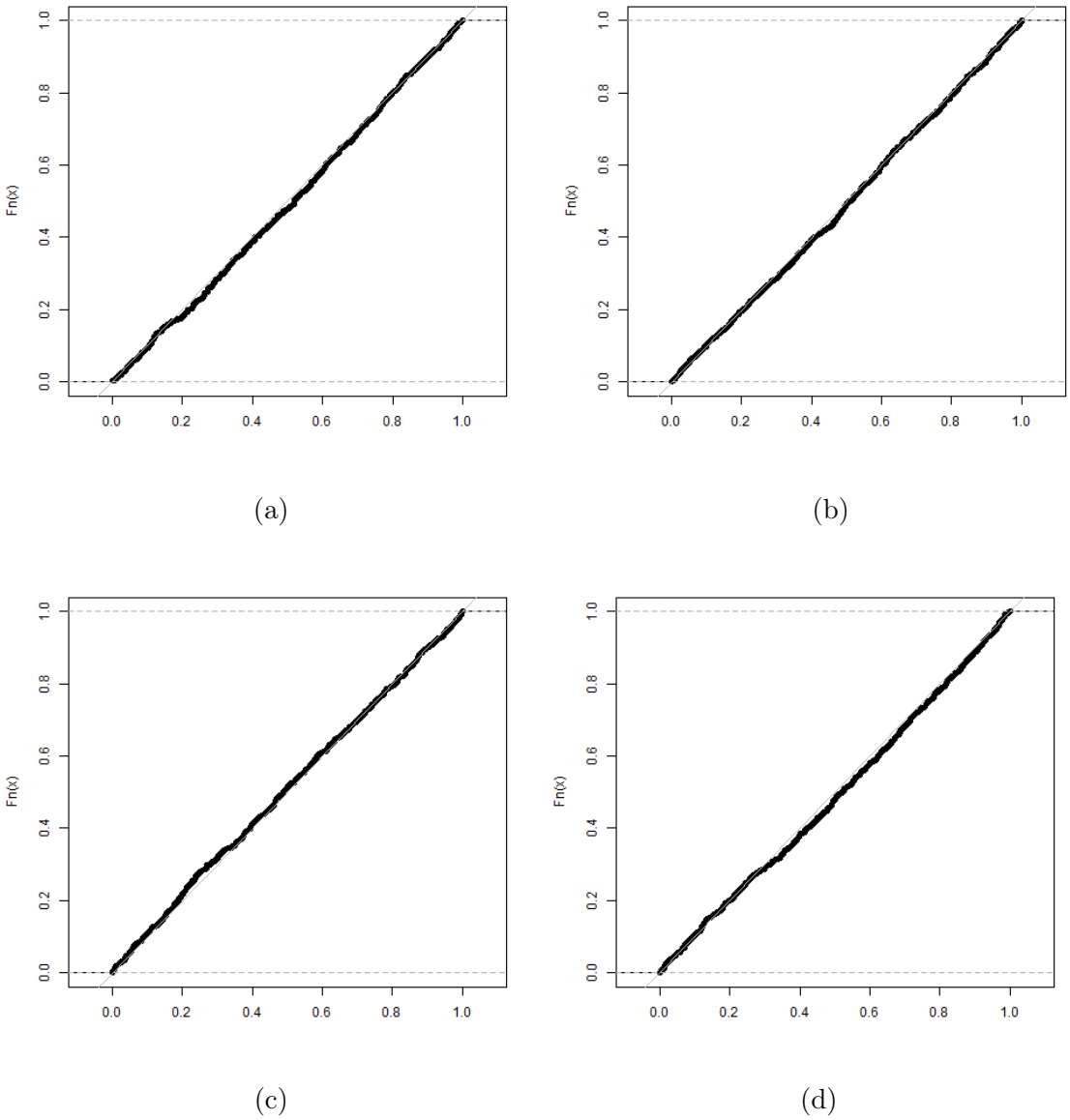
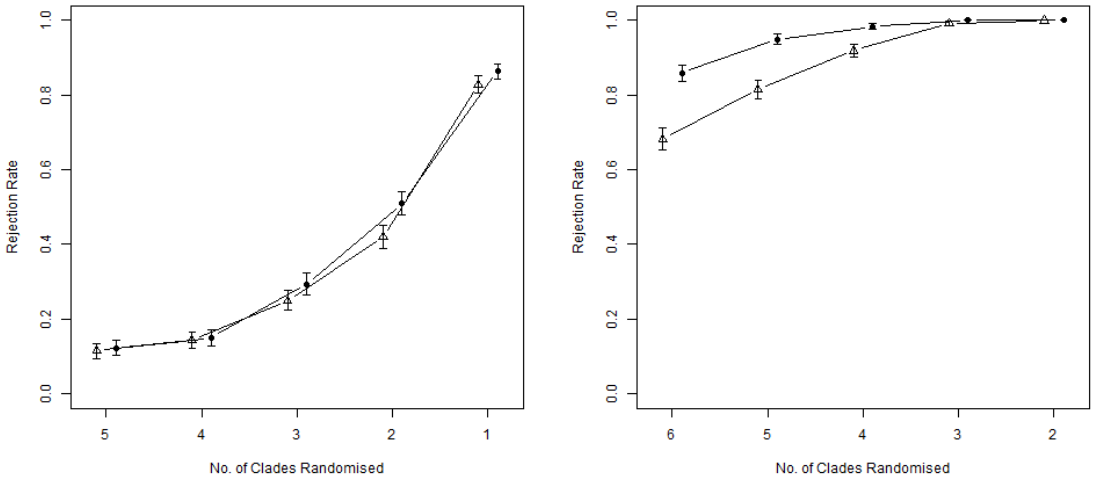


Figure F.4: Empirical cumulative distribution functions for our tritrophic p-values. Each plot corresponds to simulations with 10 tips on Trees X and Y, and 15 tips on Tree Z. Each plot represents a different number of interactions simulated. From top left to bottom right, 10, 15, 20 and 25 interactions.

F.2 Power Simulations

The rejection rate plots for Simulation Method 3 in the Power Simulations - Bitrophic section are displayed in Figure F.5. The rejection rates increase as the systems become more cospeciated and the rejection rates are higher for systems with 20 tips compared to systems with 10 tips. Our rejection rates are higher than Hommola *et al.*'s (2009) in the 10 and 20 tip case.

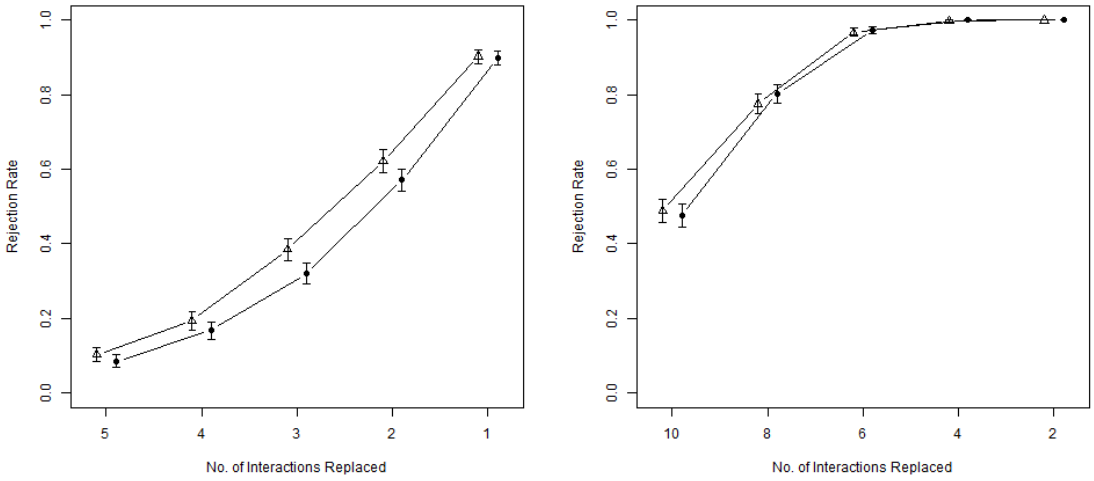


(a) (b)

Figure F.5: Rejection rates for the p-values generated using our method and the method of Hommola *et al.* (2009) at the $\alpha = 0.05$ significance level, under Simulation Approach 3. Black dots are the rates obtained using our method and triangles are the rates calculated for Hommola *et al.*'s (2009) p-values. The points are offset on the horizontal axis to prevent overlapping. The plot on the left corresponds to 10 tip simulations and the plot on the right corresponds to 20 tip simulations.

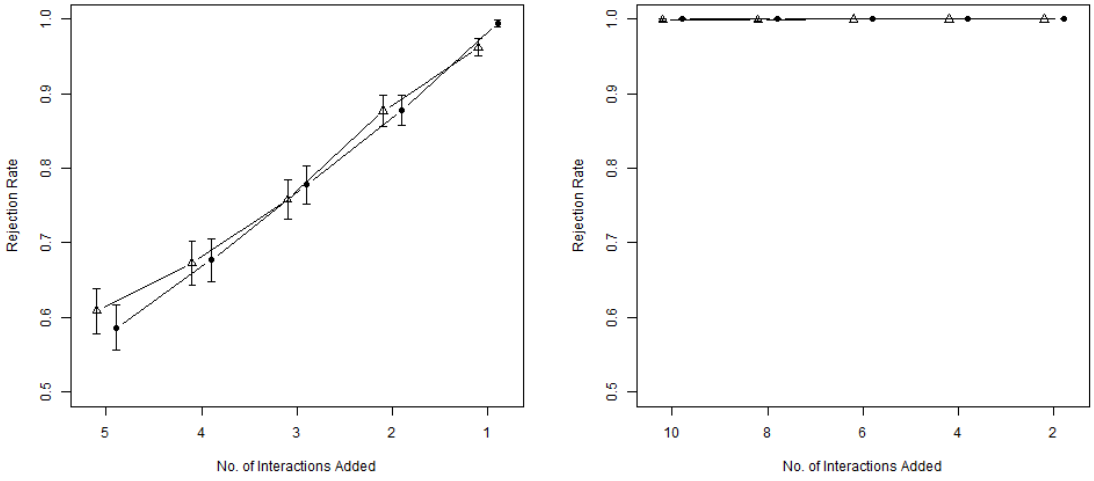
The rejection rate plots for each Simulation Method at the $\alpha = 0.01$ significance level are displayed in Figures F.6, F.7 and F.8. The rejection rates increase as the systems become more cospeciated and the rejection rates are higher for systems with 20 tips compared to systems with 10 tips. Our rejection rates are equivalent to Hommola *et al.*'s (2009) in each case.

Figure F.9 displays the rejection rates for our tritrophic p-values and Mramba *et al.*'s (2013) four different p-values for the simulation approach where we add random triangles of interactions. The rejection rates are calculated at the $\alpha = 0.05$ significance level. Each plot corresponds to a different randomisation in Mramba *et al.*'s (2013) method. The power curve for our method is repeated in each plot.



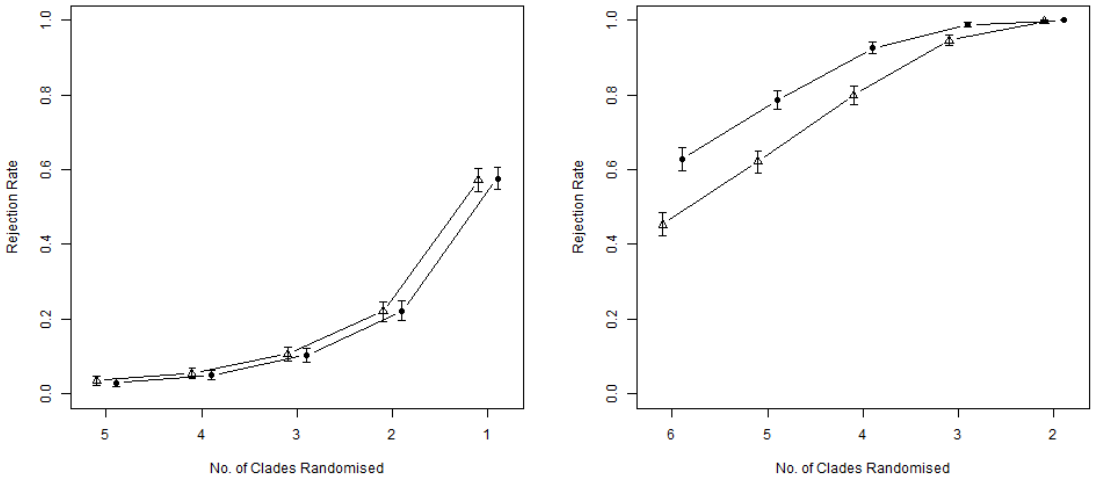
(a) (b)

Figure F.6: Rejection rates for the p-values generated using our method and the method of Hommola *et al.* (2009) at the $\alpha = 0.01$ significance level, under Simulation Approach 1. Black dots are the rates obtained using our method and triangles are the rates calculated for Hommola *et al.*'s (2009) p-values. The points are offset on the horizontal axis to prevent overlapping. The plot on the left corresponds to 10 tip simulations and the plot on the right corresponds to 20 tip simulations.



(a) (b)

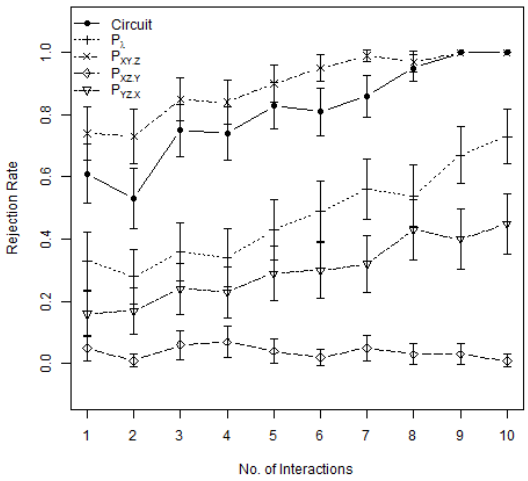
Figure F.7: Rejection rates for the p-values generated using our method and the method of Hommola *et al.* (2009) at the $\alpha = 0.01$ significance level, under Simulation Approach 2. Black dots are the rates obtained using our method and triangles are the rates calculated for Hommola *et al.*'s (2009) p-values. The points are offset on the horizontal axis to prevent overlapping. The plot on the left corresponds to 10 tip simulations and the plot on the right corresponds to 20 tip simulations.



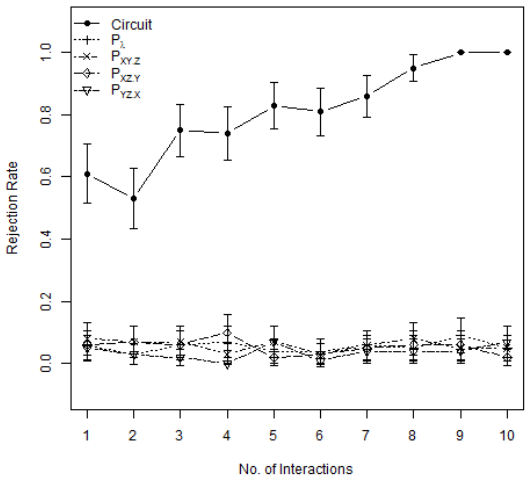
(a) (b)

Figure F.8: Rejection rates for the p-values generated using our method and the method of Hommola *et al.* (2009) at the $\alpha = 0.01$ significance level, under Simulation Approach 3. Black dots are the rates obtained using our method and triangles are the rates calculated for Hommola *et al.*'s (2009) p-values. The points are offset on the horizontal axis to prevent overlapping. The plot on the left corresponds to 10 tip simulations and the plot on the right corresponds to 20 tip simulations.

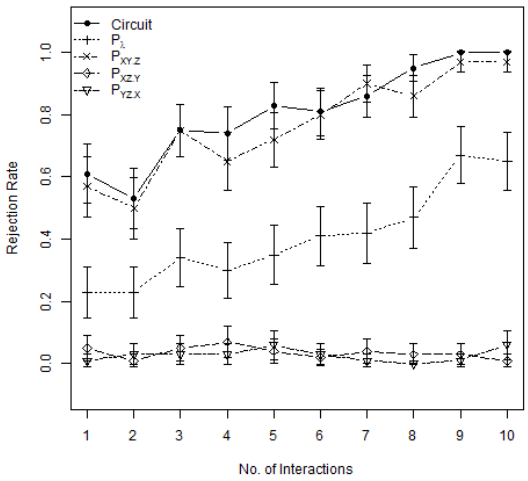
Figures F.9a, F.9b, F.9c and F.9d correspond to the cases where only Tree X is randomised, only Tree Z is randomised, both Trees X and Y are randomised, and all three trees are randomised, respectively.



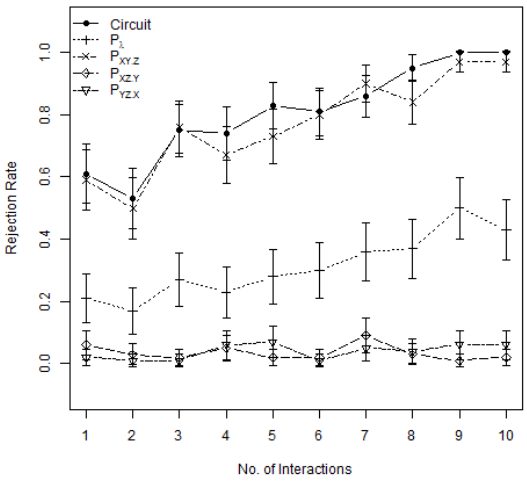
(a) Only X randomised



(b) Only Z randomised



(c) X and Y randomised



(d) X, Y and Z randomised

Figure F.9: Rejection rates for p-values generated using our method and the method of Mramba *et al.* (2013) at the $\alpha = 0.05$ significance level, under the simulation approach where triangular interactions are added between three 10 tip trees. The interactions between the three trees are forced to form triangles. The horizontal axis corresponds to the number of interactions added between each pair of trees. Black dots are the rates obtained using our method, labelled "Circuit", and the other lines correspond to the rates calculated for the different p-values obtained under Mramba *et al.*'s (2013) method; P_λ, $P_{xy.z}$, $P_{xz.y}$ and $P_{yz.x}$.

Tree Z is not involved in the cospeciation between Trees X and Y, thus permuting Tree Z reveals no effect of cospeciation. This can be seen in Figure F.9b, as expected, the rejections rates for Mramba *et al.*'s (2013) method are all very low. From Table 4.1, a significant value of $P_{xy.z}$ when Trees X and Y are involved in the randomisation indicates that there is cospeciation between Trees X and Y. This can clearly be seen in Figures F.9a, F.9c, F.9d where the statistic corresponding to $P_{xy.z}$ is the most powerful. The statistics corresponding to $P_{xz.y}$ and $P_{yz.x}$ are less powerful because Trees X and Y are not cospeciating with Tree Z, and randomising Tree X tells us nothing about the cospeciation between Trees Y and Z. Our statistic has slightly less power than $P_{xy.z}$ under some randomisations.

Figure F.10 displays the rejection rates, calculated at the $\alpha = 0.01$ significance level, for our method and Mramba *et al.*'s (2013) for simulations with interactions that are not constrained to form triangles. The first column of plots corresponds to simulation method 1 and the second column to simulation method 2. The rows correspond to the size of the trees; the first row is simulations involving 10 tip trees and the second row is 20 tip trees. We show only one of Mramba *et al.*'s (2013) randomisations, the case where only Tree X is randomised; other plots display very similar results. Clearly our statistics is more powerful than the method of Mramba *et al.* (2013).

F.3 Tritrophic Dataset

The phylogenetic trees for the Lopez-Vaamonde *et al.* (2005) dataset consisting of hostplants, leaf-mining moths and parasitoid wasps are plotted individually, with edge lengths and species labels in Figures F.11a, F.11b and F.11c respectively.

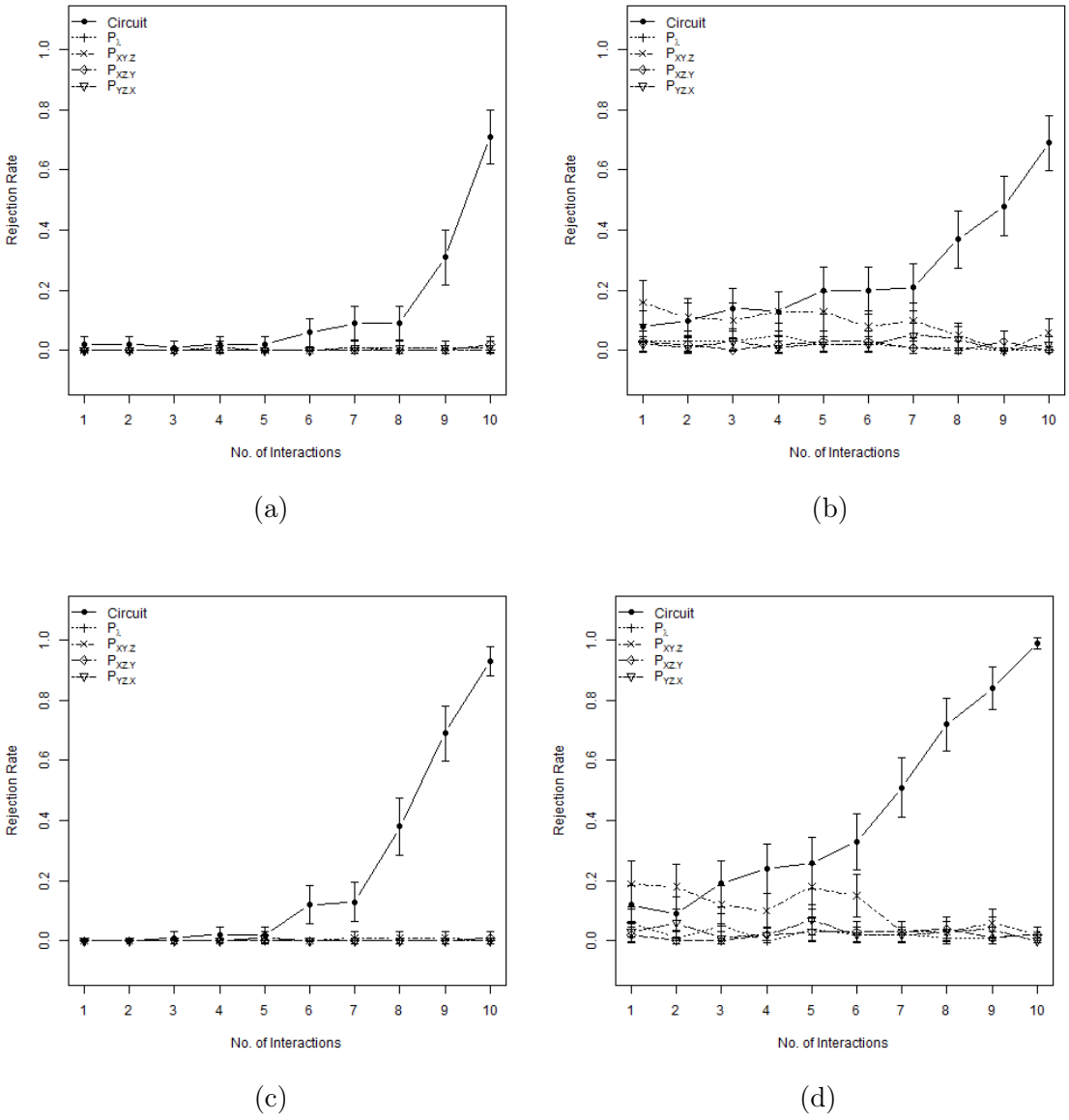
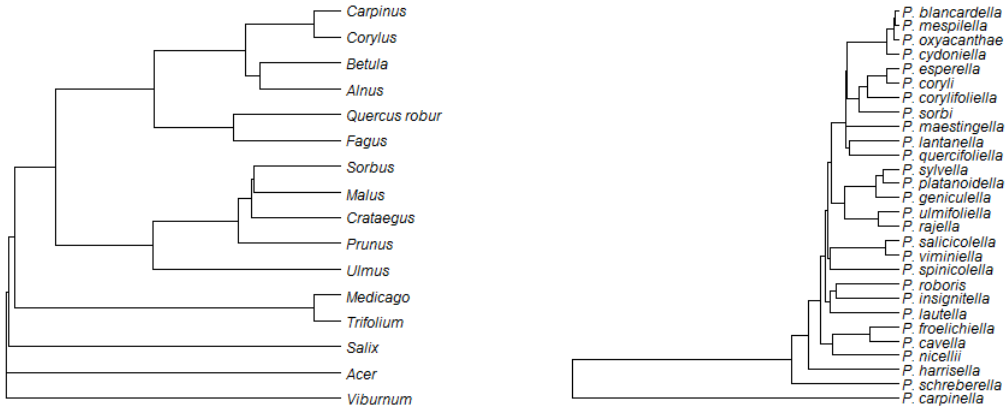
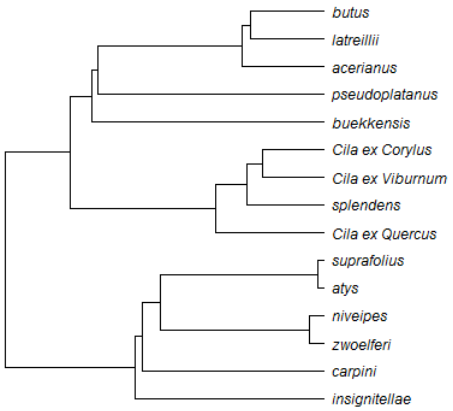


Figure F.10: Rejection rates for p-values generated using our method and the method of Mramba *et al.* (2013) at the $\alpha = 0.01$ significance level, under different simulation approaches. Each column corresponds to a different simulation approach; replacing and adding interactions between the three trees, respectively. The horizontal axis corresponds to the number of interactions replaced or added between each pair of trees. In each simulation the interactions are not forced to form triangles. The rows correspond to the tree sizes. The first row contains the 10 tip simulations for each approach. The second row contains the 20 tip simulations for each approach. Each plot corresponds to the case where only Tree X is randomised for Mramba *et al.*'s (2013) method. Black dots are the rates obtained using our method, labelled "Circuit", and the other lines correspond to the rates calculated for the different p-values obtained under Mramba *et al.*'s (2013) method; P_λ, $P_{xy.z}$, $P_{xz.y}$ and $P_{yz.x}$.



(a) Hostplant

(b) Leaf-mining moth



(c) Parasitoid wasp

Figure F.11: Individual phylogenetic trees for the hostplant, leaf-mining moth and parasitoid wasp tritrophic dataset (Lopez-Vaamonde *et al.*, 2005). Each phylogenetic tree is plotted using edge lengths. Leaf node species labels are also displayed.

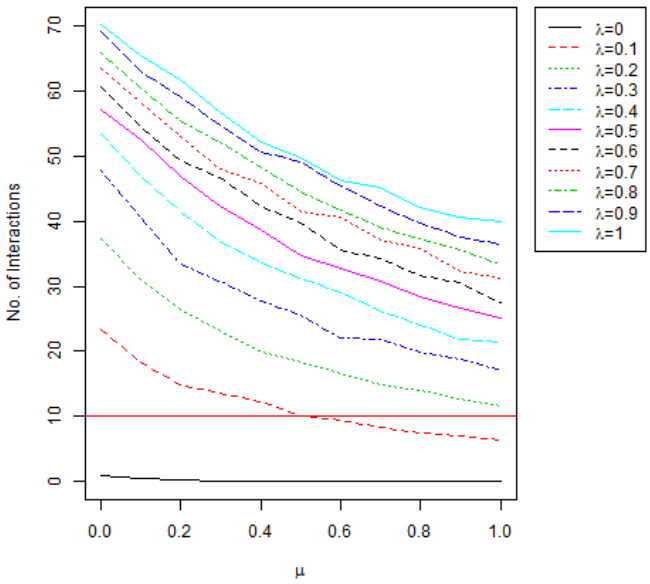
Appendix G

Additional Figures and Plots for Chapter 5

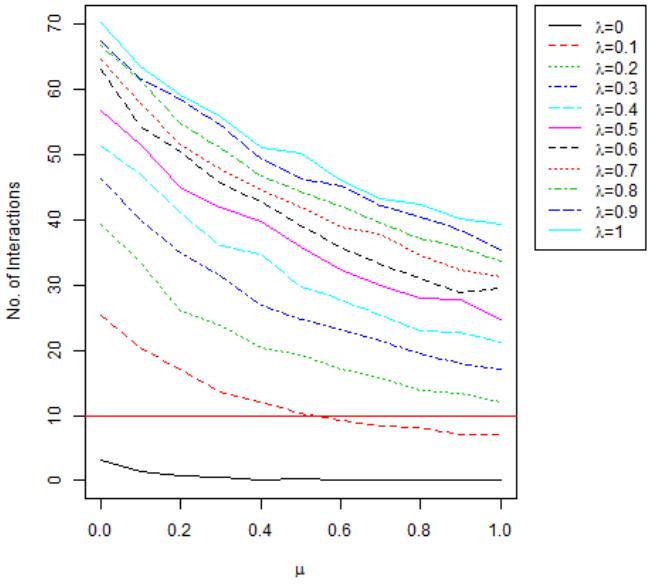
G.1 Parameter Calibration - Bitrophic

Figure G.1 displays parameter calibration plots for the numerical simulations in Section 5.3.2. For each system the parameters are set to range from evolving independently (System 1) to perfectly cospeciated (System 6) by scaling α, β and $\alpha\beta$ incrementally. The parameter combinations used for each system are given in Table 5.3. To determine the values of λ and μ we perform numerical simulations. For each parameter combination in Table 5.3, 100 systems are generated. The values of λ and μ are varied between 0 and 1 for each parameter combination.

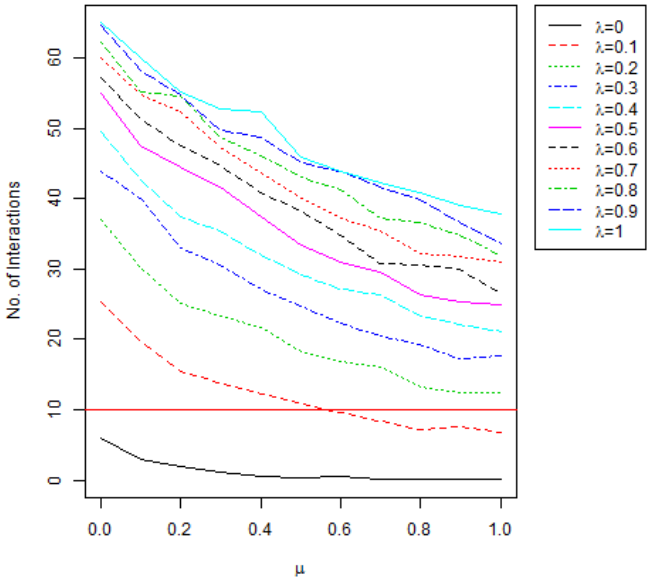
Figure G.1: Parameter calibration plots for the numerical simulations in Section 5.3.2. The parameter combinations used in plots (a)-(e) are given in Table 5.3. For each parameter combination, 100 systems are generated. The values of λ and μ are varied between 0 and 1 for each parameter combination. (a) System 1: Trees X and Y are evolving independently. (b)-(e) Systems 2-5: Trees X and Y are gradually more cospeciated.



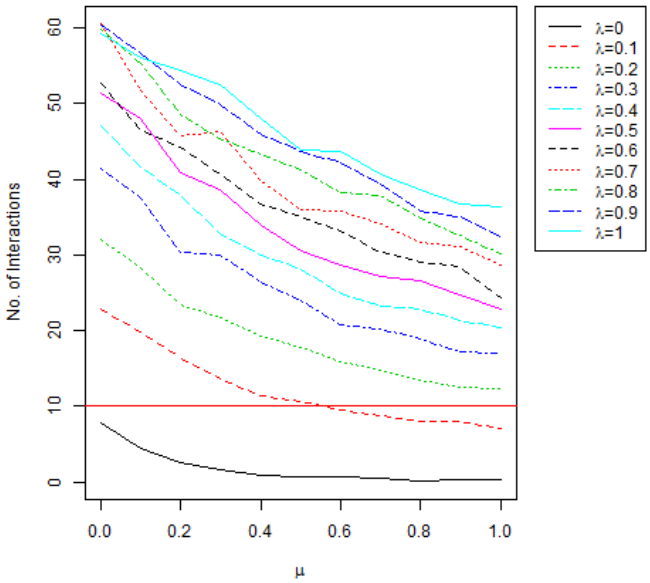
(a) System 1



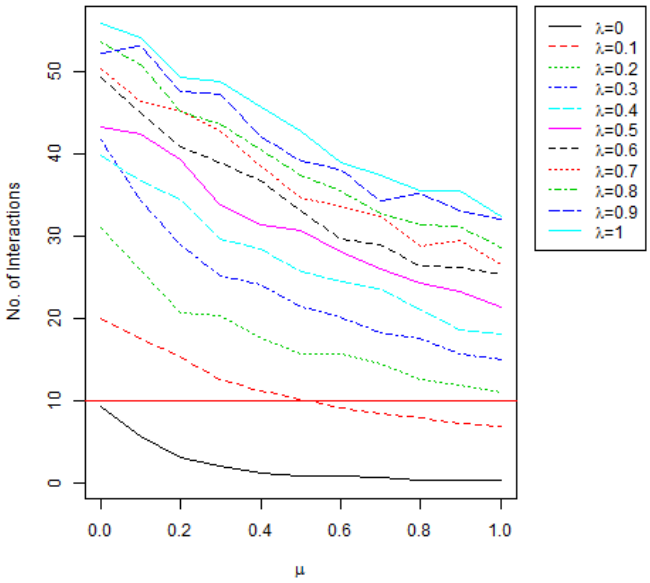
(b) System 2



(c) System 3



(d) System 4



(e) System 5

Figure G.1: (cont.)

References

Ahmad, F., Aslam, M. & Razaq, M. (2004). Chemical ecology of insects and tritrophic interactions. *Journal of Research (Science)*, **15**, 181–190.

Bateman, A., Coin, L., Durbin, R., Finn, R., Hollich, V., Griffiths-Jones, S., Khanna, A., Marshall, M., Moxon, S., Sonnhammer, E. *et al.* (2004). The Pfam protein families database. *Nucleic Acids Research*, **32**, D138–D141.

Berbalk, C., Schwaiger, C.S. & Lackner, P. (2009). Accuracy analysis of multiple structure alignments. *Protein Science*, **18**, 2027–2035.

Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T., Weissig, H., Shindyalov, I.N. & Bourne, P.E. (2000). The protein data bank. *Nucleic Acids Research*, **28**, 235–242.

Boutet, E., Lieberherr, D., Tognolli, M., Schneider, M., Bansal, P., Bridge, A.J., Poux, S., Bougueleret, L. & Xenarios, I. (2016). UniProtKB/Swiss-Prot, the manually annotated section of the UniProt KnowledgeBase: how to use the entry view. *Plant Bioinformatics: Methods and Protocols*, 23–54.

Branden, C., Tooze, J. *et al.* (1991). *Introduction to Protein Structure*, vol. 2. Garland New York.

Curtis, E.B., Morrow, J.A., Curtis, E. & a Morrow, J. (2000). *Inverse problems for electrical networks*, vol. 13. World Scientific.

Durbin, R. (1998). *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press.

Edgar, R.C. (2004). Muscle: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research*, **32**, 1792–1797.

Finn, R.D., Clements, J. & Eddy, S.R. (2011). Hmmer web server: interactive sequence similarity searching. *Nucleic Acids Research*, **39**, gkr367.

Fiser, A. (2010). Template-based protein structure modeling. *Methods in Molecular Biology*, **673**, 73–94.

Forister, M.L. & Feldman, C.R. (2011). Phylogenetic cascades and the origins of tropical diversity. *Biotropica*, **43**, 270–278.

Fourment, M. & Gibbs, M.J. (2006). PATRISTIC: a program for calculating patristic distances and graphically comparing the components of genetic change. *BMC Evolutionary Biology*, **6**, 1.

Friedman, J., Hastie, T. & Tibshirani, R. (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software*, **33**, 1–22.

Göbel, U., Sander, C., Schneider, R. & Valencia, A. (1994). Correlated mutations and residue contacts in proteins. *Proteins: Structure, Function, and Bioinformatics*, **18**, 309–317.

Hall, B.G. (2004). *Phylogenetic trees made easy: a how-to manual*. Sinauer Associates Sunderland.

Harmon, L., Weir, J., Brock, C. & Challenger, W. (2007). Geiger: A package for macroevolutionary simulation and estimating parameters related to diversification from comparative phylogenetic data. *Available via the complete R package, website http://www. r-project. org [accessed 4 July 2008]*.

Hartmann, K., Wong, D. & Stadler, T. (2010). Sampling trees from evolutionary models. *Systematic Biology*, **59**, 465–476.

Hastie, T., Tibshirani, R. & Wainwright, M. (2015). *Statistical Learning with Sparsity: The Lasso and Generalizations*. CRC Press.

Henikoff, S. & Henikoff, J.G. (1994). Position-based sequence weights. *Journal of Molecular Biology*, **243**, 574–578.

Herraez, A. (2006). Biomolecules in the computer: Jmol to the rescue. *Biochemistry and Molecular Biology Education*, **34**, 255–261.

Höhna, S. (2013). Fast simulation of reconstructed phylogenies under global time-dependent birth–death processes. *Bioinformatics*, **29**, 1367–1374.

Holm, L. & Sander, C. (1993). Protein structure comparison by alignment of distance matrices. *Journal of Molecular Biology*, **233**, 123–138.

Hommola, K., Smith, J.E., Qiu, Y. & Gilks, W.R. (2009). A permutation test of host–parasite cospeciation. *Molecular Biology and Evolution*, **26**, 1457–1468.

Huelsenbeck, J.P., Rannala, B. & Larget, B. (2000). A Bayesian framework for the analysis of cospeciation. *Evolution*, **54**, 352–364.

Jin, G., Nakhleh, L., Snir, S. & Tuller, T. (2007). Efficient parsimony-based methods for phylogenetic network reconstruction. *Bioinformatics*, **23**, e123–e128.

Jones, D.T., Buchan, D.W., Cozzetto, D. & Pontil, M. (2012). PSICOV: precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments. *Bioinformatics*, **28**, 184–190.

Jones, D.T., Singh, T., Kosciolek, T. & Tetchner, S. (2015). MetaPSICOV: combining coevolution methods for accurate prediction of contacts and long range hydrogen bonding in proteins. *Bioinformatics*, **31**, 999–1006.

Jones, S., Stewart, M., Michie, A., Swindells, M.B., Orengo, C. & Thornton, J.M. (2008). Domain assignment for protein structures using a consensus approach: characterization and analysis. *Protein Science*, **7**, 233–242.

Kearsley, S.K. (1989). On the orthogonal transformation used for structural comparisons. *Acta Crystallographica Section A: Foundations of Crystallography*, **45**, 208–210.

Klassen, G.J. (1992). Coevolution: a history of the macroevolutionary approach to studying host-parasite associations. *The Journal of Parasitology*, **78**, 573–587.

Konagurthu, A.S., Whisstock, J.C., Stuckey, P.J. & Lesk, A.M. (2006). MUSTANG: a multiple structural alignment algorithm. *Proteins: Structure, Function, and Bioinformatics*, **64**, 559–574.

Kosciolek, T. & Jones, D.T. (2015). Accurate contact predictions using covariation techniques and machine learning. *Proteins: Structure, Function, and Bioinformatics*.

Krishnan, V. & Rupp, B. (2012). Macromolecular Structure Determination: Comparison of X-ray Crystallography and NMR Spectroscopy. *eLS*.

Legendre, P., Desdevises, Y. & Bazin, E. (2002). A statistical test for host–parasite coevolution. *Systematic Biology*, **51**, 217–234.

Liu, W., Srivastava, A. & Zhang, J. (2011). A mathematical framework for protein structure comparison. *PLoS Computational Biology*, **7**, e1001075.

Lopez-Vaamonde, C., Godfray, H., West, S., Hansson, C. & Cook, J. (2005). The evolution of host use and unusual reproductive strategies in Achrysocharoides parasitoid wasps. *Journal of Evolutionary Biology*, **18**, 1029–1041.

Lovell, S.C. & Robertson, D.L. (2010). An integrated view of molecular coevolution in protein–protein interactions. *Molecular Biology and Evolution*, **27**, 2567–2575.

Makarenkov, V., Kevorkov, D. & Legendre, P. (2006). Phylogenetic network construction approaches. *Applied Mycology and Biotechnology*, **6**, 61–97.

Mantel, N. (1967). The detection of disease clustering and a generalized regression approach. *Cancer Research*, **27**, 209–220.

Mardia, K.V., Kent, J.T. & Bibby, J.M. (1979). *Multivariate Analysis*. Academic Press.

Marks, D.S., Colwell, L.J., Sheridan, R., Hopf, T.A., Pagnani, A., Zecchina, R. & Sander, C. (2011). Protein 3D structure computed from evolutionary sequence variation. *PLoS One*, **6**, e28766.

Micha, S.G., Kistenmacher, S., Mölck, G., Wyss, U., Vincent, C., Coderre, D., Hodek, I. *et al.* (2000). Tritrophic interactions between cereals, aphids and parasitoids: discrimination of different plant-host complexes by Aphidius rhopalosiphi (Hymenoptera: Aphidiidae). In *European Journal of Entomology*, vol. 97, 539–543, Institute of Entomology, Czech Academy of Sciences.

Monastyrskyy, B., D'Andrea, D., Fidelis, K., Tramontano, A. & Kryshtafovych, A. (2014). Evaluation of residue–residue contact prediction in CASP10. *Proteins: Structure, Function, and Bioinformatics*, **82**, 138–153.

Monastyrskyy, B., D'Andrea, D., Fidelis, K., Tramontano, A. & Kryshtafovych, A. (2015). New encouraging developments in contact prediction: Assessment of the CASP11 results. *Proteins: Structure, Function, and Bioinformatics*.

Moult, J., Pedersen, J.T., Judson, R. & Fidelis, K. (1995). A large-scale experiment to assess protein structure prediction methods. *Proteins: Structure, Function, and Bioinformatics*, **23**, 2–5.

Moult, J., Fidelis, K., Kryshtafovych, A., Schwede, T. & Tramontano, A. (2014). Critical assessment of methods of protein structure prediction (CASP)—round X. *Proteins: Structure, Function, and Bioinformatics*, **82**, 1–6.

Moult, J., Fidelis, K., Kryshtafovych, A., Schwede, T. & Tramontano, A. (2016). Critical assessment of methods of protein structure prediction: Progress and new directions in round XI. *Proteins: Structure, Function, and Bioinformatics*, 4–14.

Mramba, L.K., Barber, S., Hommola, K., Dyer, L.A., Wilson, J.S., Forister, M.L. & Gilks, W.R. (2013). Permutation tests for analyzing cospeciation in multiple phylogenies: applications in tri-trophic ecology. *Statistical Applications in Genetics and Molecular Biology*, **12**, 679–701.

Nelson, L.A., Davies, K.A., Scheffer, S.J., Taylor, G.S., Purcell, M.F., Giblin-Davis, R.M., Thornhill, A.H. & Yeates, D.K. (2014). An emerging example of tritrophic coevolution between flies (Diptera: Fergusoninidae) and nematodes (Nematoda: Neotylenchidae) on Myrtaceae host plants. *Biological Journal of the Linnean Society*, **111**, 699–718.

Neuwald, A.F. (2007). The CHAIN program: forging evolutionary links to underlying mechanisms. *Trends in Biochemical Sciences*, **32**, 487–493.

Nooney, C., Gusnanto, A., Gilks, W.R. & Barber, S. (2015). Do protein structures evolve around 'anchor' residues? In I.L. Dryden & J.T. Kent, eds., *Geometry Driven Statistics*, chap. 16, 311–336, John Wiley & Sons.

Page, R.D. (1996). Temporal congruence revisited: comparison of mitochondrial DNA sequence divergence in cospeciating pocket gophers and their chewing lice. *Systematic Biology*, **45**, 151–167.

Page, R.D. (2003). *Tangled Trees: Phylogeny, Cospeciation, and Coevolution*. University of Chicago Press.

Paradis, E., Claude, J. & Strimmer, K. (2004). APE: analyses of phylogenetics and evolution in R language. *Bioinformatics*, **20**, 289–290.

Perretto, M. & Lopes, H.S. (2005). Reconstruction of phylogenetic trees using the ant colony optimization paradigm. *Genetics and Molecular Research*, **4**, 581–589.

R Core Team (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Rambaut, A. (2002). Phylogen: phylogenetic tree simulator package. Department of Zoology, University of Oxford. *Oxford, UK. Available from: http://tree. bio. ed. ac. uk/software/phylogen*.

Rau, G., Mearns, A., Young, D., Olson, R., Schafer, H. & Kaplan, I. (1983). Animal C/C correlates with trophic level in pelagic food webs. *Ecology*, **64**, 1314–1318.

Remmert, M., Biegert, A., Hauser, A. & Söding, J. (2012). Hhblits: lightning-fast iterative protein sequence searching by hmm-hmm alignment. *Nature Methods*, **9**, 173–175.

Rypniewski, W., Perrakis, A., Vorgias, C. & Wilson, K. (1994). Evolutionary divergence and conservation of trypsin. *Protein Engineering*, **7**, 57–64.

Saitou, N. & Nei, M. (1987). The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular Biology and Evolution*, **4**, 406–425.

Seemayer, S., Gruber, M. & Söding, J. (2014). CCMpred—fast and precise prediction of protein residue–residue contacts from correlated mutations. *Bioinformatics*, **30**, 3128–3130.

Sonnhammer, E.L., Eddy, S.R., Birney, E., Bateman, A. & Durbin, R. (1998). Pfam: multiple sequence alignments and HMM-profiles of protein domains. *Nucleic Acids Research*, **26**, 320–322.

Sreekumar, J., ter Braak, C.J., van Ham, R.C. & van Dijk, A.D. (2011). Correlated mutations via regularized multinomial regression. *BMC Bioinformatics*, **12**, 444.

Stadler, T. (2010). Treesim: simulating trees under the birth-death model. *R package version 1.0. Available from: http://cran.r-project.org/package=TreeSim*.

Stadler, T. (2011). Simulating trees with a fixed number of extant species. *Systematic biology*, **60**, 676—684.

Stroud, R.M. (1974). A family of protein-cutting proteins. *Scientific American*, **231**, 74.

Thompson, J.D., Higgins, D.G. & Gibson, T.J. (1994). CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Research*, **22**, 4673–4680.

Thompson, R.M., Hemberg, M., Starzomski, B.M. & Shurin, J.B. (2007). Trophic levels and trophic tangles: the prevalence of omnivory in real food webs. *Ecology*, **88**, 612–617.

Várallyay, É., Lengyel, Z., Gráf, L. & Szilágyi, L. (1997). The role of disulfide bond c191-c220 in trypsin and chymotrypsin. *Biochemical and Biophysical Research Communications*, **230**, 592–596.

Wheelan, S., Marchler-Bauer, A. & Bryant, S.H. (2000). Domain size distributions can predict domain boundaries. *Bioinformatics*, **16**, 613–618.

Yang, L.W., Eyal, E., Chennubhotla, C., Jee, J., Gronenborn, A.M. & Bahar, I. (2007). Insights into equilibrium dynamics of proteins from comparison of NMR and X-ray data with computational predictions. *Structure (London, England: 1993)*, **15**, 741.

Yip, K.Y., Patel, P., Kim, P.M., Engelman, D.M., McDermott, D. & Gerstein, M. (2008). An integrated system for studying residue coevolution in proteins. *Bioinformatics*, **24**, 290–292.

Zhang, Y. & Skolnick, J. (2004). Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics*, **57**, 702–710.

Zhang, Y. & Skolnick, J. (2005). TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Research*, **33**, 2302–2309.