Modelling and comparing protein interaction networks using subgraph counts

Tiago Rito
Department of Statistics
University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy

Trinity 2012

Acknowledgements

With thanks to my supervisors, Charlotte Deane and Gesine Reinert for their guidance and positive attitude towards research.

With thanks to the Oxford Protein Informatics Group (OPIG) for providing a fun, supportive environment for research and to all that helped with the work in this thesis. In particular Mireille Gomes, Jamie Hill, Konrad Krawczyk, Sebastian Kelm and Charlotte Pattullo for comments on the manuscript. Also a note of thanks to my funding body, Fundação para a Ciência e a Tecnologia (FCT), and to the Systems Biology Doctoral Training Center (DTC).

With thanks to the extended family of the 14WS B&B for giving me a home which I looked forward to returning to every day.

Last but not least, I would like to thank my Family for the unconditional love and support.

Abstract

The astonishing progress of molecular biology, engineering and computer science has resulted in mature technologies capable of examining multiple cellular components at a genome-wide scale. Protein-protein interactions are one example of such growing data. These data are often organised as networks with proteins as nodes and interactions as edges. Albeit still incomplete, there is now a substantial amount of data available and there is a need for biologically meaningful methods to analyse and interpret these interactions.

In this thesis we focus on how to compare protein interaction networks (PINs) and on the relationship between network architecture and the biological characteristics of proteins. The underlying theme throughout the dissertation is the use of small subgraphs – small interaction patterns between 2-5 proteins.

We start by examining two popular scores that are used to compare PINs and network models. When comparing networks of the same model type we find that the typical scores are highly unstable and depend on the number of nodes and edges in the networks. This is unsatisfactory and we propose a method based on non-parametric statistics to make more meaningful comparisons. We also employ principal component analysis to judge model fit according to subgraph counts. From these analyses we show that no current model fits to the PINs; this may well reflect our lack of knowledge on the evolution of protein interactions. Thus, we use explanatory variables such as protein age and protein structural class to find patterns in the interactions and subgraphs we observe. We discover that the yeast PIN is highly heterogeneous and therefore no single model is likely to fit the network. Instead, we focus on ego-networks containing an initial protein plus its interacting partners and their interaction partners. In the final chapter we propose a new, alignment-free method for network comparison based on such ego-networks. The method compares subgraph counts in neighbourhoods within PINs in an averaging, many-to-many fashion. It clusters networks of the same model type and is able to successfully reconstruct species phylogenies solely based on PIN data providing exciting new directions for future research.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Proteins are essential molecules that perform most functions inside a living cell. They are unbranched linear polymers built up of amino-acids joined together through a peptide bond. A protein chain is normally folded into a relatively rigid, compact three-dimensional structure. This structure largely determines its role and function in the cell. In the 1940s, descriptions of the interactions between the actin and myosin proteins in the muscle cells and between regulatory metabolic enzymes started to stress the importance of protein interactions. These interactions happen using complementary patches on the surfaces of the interacting proteins which exploit electrostatic and hydrophobic forces [Chothia and Janin, 1975; Jones and Thornton, 1996]. In the 1960s, the first signal transduction cascade based on protein phosphorylation was discovered; in the 1980s it became clear that protein interactions were also fundamental to control the eukaryotic cell cycle (see Braun and Gingras [2012] for more details on the history of protein interactions). Nowadays it is generally recognised that protein interactions are at the core of the most complex cellular functions. Recent advances in experimental techniques have generated a considerable amount of protein-protein interaction (PPI) data for several organisms. The BIOGRID^{TM}database stores over 209,354 non-redundant physical interactions of 42,308 unique proteins for 38 model organisms

(in September 2012). These interactions are often integrated to form networks, with proteins as nodes and interactions as edges, forming the so-called PPI networks. This integration can be carried out by merging interactions according to the organism to which they belong, to the experimental technique that was used to generate them or to other, more specialised, arrangements.

In this context, a reliable and efficient method for comparing these large networks would be very useful [Sharan and Ideker, 2006]. Such a comparison may yield mechanistic and evolutionary insights, help identify missing links, and even aid network validation for those organisms where experimental data is scarce. Differences between a diseased and a healthy network could also lead to a greater understanding of the underlying mechanisms and causes of a disease [Ideker and Sharan, 2008]. Many methods have been proposed, but to date none offers a good solution to this problem [Alm and Arkin, 2003]. Network comparison is a computationally difficult task due to the large size of typical PPI networks. Furthermore, current PPI networks are still incomplete and rife with noise [Venkatesan et al., 2009; von Mering et al., 2002]. They tend to have a large number of false-positives and false-negatives which obscure meaningful conclusions and call for robust methods of analysis [Alm and Arkin, 2003; Huang et al., 2007].

One type of method for the comparison of networks uses short lists of summary statistics including, for instance, the degree distribution and the clustering coefficient [Costa et al., 2007]. Here we focus on comparing networks based on small subgraphs, *i.e.* small interaction patterns between 2-5 proteins. Cell biology is thought of as modular with many pathways and feedback loops being inherently seen as detachable modules [Hartwell et al., 1999]. While it has been shown that interaction patterns alone do not determine function [Ingram et al., 2006], subgraphs have been linked to modules performing coherent biological functions and hence appear to have an important role in cellular systems [Bachman and Liu, 2009; Shen-Orr et al., 2002; Wuchty et al., 2003].

With the view of comparing networks via their subgraph content, Przulj *et al.* introduced two scores based on subgraph counts. In Chapter 2 we examine the Relative Graphlet frequency distance (RGF) [Przulj et al., 2004] and the Graphlet Degree Distribution Agreement (GDDA) scores [Przulj, 2007]. The scores are used to compare networks with random graph models, assessing their fit to the empirical network based on topology alone. In order to test the applicability of the scores, we compared networks of the same random graph model. We find both RGF and GDDA scores to be dependent on the number of edges and nodes of the networks being compared and highly unstable in the region of graph density (ratio of the number of edges in the network to the number of possible edges) relevant for PPI networks. This graph density region coincides with the graph density at which subgraphs start to appear in Bernoulli random graphs, suggesting that these networks might be at the verge of the emergence of subgraphs [Rito et al., 2010]. Proving this conjecture would have implications to the ideas of network robustness and efficiency, and, ultimately, to network design. However, this would require a good model for PPI networks with respect to subgraph counts which is currently unavailable. Despite the claims in Przulj [2007], we were not able to find any network model that fits our PPI data, with the fit judged by GDDA. We then abandoned the GDDA score and used principal component analysis on subgraph counts to test the fit of new models. All models tested were still unsatisfactory.

We therefore took a step back to investigate features and patterns in the data that may uncover the main faults of current models. We discovered that even counts of basic shapes like triangles are poorly replicated by most models. To ameliorate this, in Chapter 3 we study the nature and formation of triangles in PPI networks by linking them with explanatory variables, such as protein age [Rito et al., 2012]. Our findings point to a highly heterogeneous network with selective age-dependent interaction patterns which should be taken into account when building a more realistic model for PPI networks. Finally, in Chapter 5, we propose our own method for network

comparison. The method uses statistics similar to those employed by alignment-free sequence comparison methods to compare subgraph counts of neighbourhoods within the query networks. We do not attempt to match neighbourhoods, but compare them using an averaging approach. Our algorithm gives a single pairwise distance between networks and is able to recover correct phylogenies between several species based on PPI network data alone.

We now begin by providing background information on PPI networks and the main experimental techniques used to generate them. We introduce random graph models and, since we are interested in using small subgraphs for network comparison, we also give an overview of the calculations of thresholds for the appearance of subgraphs, only available for a selected few, well-studied, random graph models. We end the chapter by presenting some characteristics of proteins that might be relevant for explaining and modelling PPI data, and with a more detailed overview of the dissertation.

1.1 Biomolecular networks

Living organisms display a plethora of interactions with complex spatial and temporal dependencies. This is true not only in the way they interact with the material world and amongst themselves, but also between their cellular components. For instance, it is remarkable that within a crowded cell environment and under high thermal noise, biomolecules can still interact - through beautifully evolution-crafted interfaces - to efficiently perform a variety of functions. These systems of interconnected entities are often represented as networks which can be modelled using *graphs*. Graphs model relationships between objects and are composed of a set of nodes (vertices) and a set of links (edges). Over the last decade a large amount of interaction data between biomolecules has become available. Representing these data as graphs, as opposed to just a collection of pairwise interactions, has proven advantageous. For instance,

PPI data can be used to predict protein structure, function or interactions themselves. Chen et al. [2008] studied the prediction of protein interactions using not only pairwise interactions, but also explicitly using the network structure, showing that incorporating information about triangles improved the predictions, *i.e*, if the nodes i,j and w form a connected triangle, the fact that j and w interact means something for predicting an individual interaction of i.

Several types of biomolecular genome-scale networks are available involving genes, proteins and metabolites [Chen et al., 2009]. The extent to which these various types of data overlap or complement each other is currently unknown.

1.2 Protein-protein interaction (PPI) networks

Here we focus on protein-protein interaction (PPI) networks. In these networks, proteins are nodes and an edge is present between pairs of proteins which are known to interact. Currently, most interaction data arises from physical evidence of high-throughput experiments such as yeast two-hybrid (Y2H) assays or tandem affinity purification followed by mass spectrometry (TAP-MS). The exact biological meaning and significance of a protein interaction depends on the particular experimental conditions, which includes not only the specific buffers used, but also the cellular condition of the organism and the particular way in which the experimental technique assesses the interactions (for more detail see Subsection 1.3).

PPI networks are regarded as a collection of interactions that *may* occur in the cell and hence lay a foundation which is useful for a wide variety of applications, some of which are briefly mentioned below:

- Prediction of PPI for species where little experimental data is available - highly connected with network comparison/alignment between multiple species; see Sharan and Ideker [2006] for a review.

- Protein function prediction and annotation using network-based approaches [Chen et al., 2008; Kourmpetis et al., 2010; Sun et al., 2010].

- Discovery of functional modules - groups of proteins which share a common function in the cell. A particularly popular application has been to combine PPI data with gene expression data to find modules which may be involved in disease [Nibbe et al., 2011].

- Identification and prediction of drug targets whose effectiveness is judged by their role in the PPI network [Klipp et al., 2010; Kuhn et al., 2008].

- Preliminary/exploratory associations between topological features and cellular roles. For instance, Ho et al. [2010] used a topology-based approach on PPI data to identify known components of melanogenesis regulatory pathways within functional genomic datasets.

On a more theoretical level, characterising PPI networks takes us a step further in our understanding of the living organism. Several properties can be studied to better understand their design such as modularity, statistical and topological properties, and roles of highly-connected proteins [Klemm and Bornholdt, 2005; Maslov and Sneppen, 2002; Valente and Cusick, 2006]. Protein interactions are far from being conserved across organisms [Lewis et al., 2012] and, although not being random [Huang et al., 2007; Rito et al., 2010], it is currently unknown if and how the cell selects for particular network designs. It is therefore important to further study not just the evolution of single protein components, but how the architecture of the PPI networks changes over time and how this links with its function [Chor and Tuller, 2006; Erten et al., 2009; Shou et al., 2011].

1.3 Experimental approaches to generation of protein interaction data

As in any other data-driven research area, it is important to understand the mechanisms of data generation so that its strengths and limitations can be understood. To date, the construction of large-scale PPI networks has been mainly driven by two techniques: yeast two-hybrid assay and affinity purification coupled with mass spectrometry. The two techniques capture different types of PPIs and therefore cannot be used to verify one another. Both techniques are error-prone and perform poorly when detecting interactions between certain types of proteins, especially amongst membrane proteins and between these and intracellular proteins [Jensen and Bork, 2008; Sanderson, 2009]. Currently, we are still at a very early stage of getting a complete, high-quality interaction map of the cell, for instance, in the DIP database (see Subsection 1.3.3) only 12.7% of the interactions found in yeast are supported by 2 or more experimental techniques, which reflects the complementarity and lack of overlap of the different methods. Recent new emergent techniques such as protein arrays and complementation assays, seem promising at bringing insight into regions of the interaction space currently unknown [Sanderson, 2009].

1.3.1 Yeast two-hybrid assay

The yeast two-hybrid assays (Y2H) were proposed by Fields and Song [1989] who were the first to realise the potential of yeast's GAL4/UAS system for detecting protein-protein interactions. Native GAL4 protein is a potent transcription activator of genes downstream of a specific DNA sequence (the Upstream Activating Sequence or UAS) that allows yeast to grow on galactose media. The GAL4 protein consists of two essential domains: an N-terminal domain which recognises and binds to the specific DNA sequence (UAS) and a C-terminal domain which does not bind DNA but is

crucial for activating transcription. The domains can be functionally separated and fused to other proteins, *i.e.*, the DNA-binding domain can be fused with a protein X (the "bait") and the activating domain can be fused with a protein Y (the "prey"). Fields and Song [1989] proved that only if X and Y are interacting partners, will the hybrid proteins establish sufficient transcriptional drive for the cells to survive in selective media. In short, in this experimental technique, if two proteins interact, the activity of a transcription factor will be restored and a downstream reporter gene will be expressed.

This discovery was followed, in 2000, by the first large-scale analyses of yeast PPIs [Ito et al., 2000; Uetz et al., 2000]. These approaches use recombinant cloning techniques to produce fusions of proteins to the different domains of GAL4, and scan the ~ 6000 potential yeast's open reading frames (ORFs) for interactions.

These data come, nonetheless, with many drawbacks. Yeast is an eukaryotic organism with cellular compartments and most PPIs occur in the chemical environment of the cytosol. The Y2H method, involving a transcription activator, tests for interactions in the nucleus, thus many proteins tested will be out of their native compartment. The biological context of the predicted interaction is also removed due to independence of endogenous expression [von Mering et al., 2002]. The former can, at least partially, account for the high incidence of false-negatives and false-positives, whilst the latter can result in meaningless false-positive interactions since both proteins are forcedly expressed (25-45% according to Huang et al. [2007]). Although the rate of false negatives was shown to affect local network alignment algorithms far more [Ali and Deane, 2010], the rate of both false negatives and false positives can influence graph density and subgraph counts. There is also a clear bias in the PPIs found: proteins which are long, unstructured and with high iso-electric points were shown to be under-represented in the Y2H data and, as one would expect, there is a relatively higher proportion of nuclear proteins among the interactions [Jensen and Bork, 2008].

1.3.2 Affinity purification-mass spectrometry

At roughly the same time that the potential of Y2H was discovered, another key advancement in functional proteomics took place: the identification of proteins using mass spectrometry (MS). Alongside Y2H assays, MS-based methods are the main contributors to the recent explosion in PPI data, especially when preceded by Tandem Affinity Purification (TAP), a more specific type of Affinity Purification (AP).

The TAP method was originally proposed by Rigaut et al. [1999] and consists in the purification of a particular protein fused to a TAP tag. As the name suggests, the TAP tag consists of two tags: the Calmodulin-binding peptide (CBP) and ProtA (two IgG-binding units of protein A of *Staphylococcus aureus*). These allow a protein to be purified from a crowded cell lysate, usually by chromatography methods. The presence of both tags is normally required for achieving a highly specific purification with low background [Rigaut et al., 1999]. After gentle purification, the TAP-fused protein (the bait) still has other interacting proteins attached (preys). The complex is then separated, usually by SDS-PAGE (sodium dodecyl sulfate polyacrylamide gel electrophoresis) and analysed using MS. The advantages of this method are that in a single experiment, and hence under a particular condition, all proteins interacting directly or indirectly with the target protein can be identified. Recent extensions allowed the technique to deliver high-throughput results by TAP-fusing the predicted ORFs of yeast by homologous recombination [Krogan et al., 2006]. This process puts the recombinant protein under the control of endogenous promotors which results in expression levels similar to the physiological ones.

The PPI data generated from these types of experiments comes from applying either one of two methods to the raw TAP-MS data: the matrix interpretation method or the standard spoke model. In the spoke model only bait-prey interactions are considered, whilst the matrix model includes all other prey-prey interactions that co-purify with the tagged bait protein.

The model choice is still a matter of debate and an active research area [Abu-Farha et al., 2008]. Since the complexes isolated by TAP-MS are stable, it is not trivial to distinguish between direct or indirect interactions or experimental artefacts. While the matrix model retrieves many more interactions, the inability to control and properly filter them explains why most studies are still employing the spoke model. This affects not only the sparseness of the data and care must be taken to minimise the numbers of false negative and false positive interactions, but the topology of the network is also particularly vulnerable to the choice of model. By definition, the matrix model is richer in highly-connected clumps of proteins and hence more heterogeneous, denser networks result, whereas the spokes model tends to favour star-like connectivities.

An interactions scoring system was recently proposed by Hart et al. [2007] that is based on comparing observed *versus* expected number of interactions in the matrix model via a probabilistic model which defines the probability of a given interaction being observed at random, also allowing data integration from several sources. By filtering the matrix model based interactions, this type of approach partially tries to limit the rate of false negatives and to deliver more accurate topologies due to better performance at discerning true prey-prey interactions from experimental artefacts.

Further to the interpretation of the experimental outcome, which can then generate networks with very different topologies [von Mering et al., 2002], the use of affinity tags in mass spectrometry techniques can affect complex formation and the particular harshness of the purification procedure may also introduce biases in the data by allowing more or less proteins to stay in the complex. Overall TAP-MS shows a small bias towards abundant and cytosolic proteins [Jensen and Bork, 2008].

1.3.3 Public databases

The generation of all these interaction data from various sources soon catalysed the appearance of online repositories. These repositories collect and organise experimental

data that support PPIs with the goal of compiling a comprehensive set of annotated data, not privileging any type of experimental evidence according to its representation or reliability [Rivas and Fontanillo, 2010]. Several public databases are now available, such as BioGRID [Breitkreutz et al., 2007], DIP [Salwinski et al., 2004] and HPRD [Keshava Prasad et al., 2009].

The Biological General Repository for Interaction Datasets (BioGRID, `http://thebiogrid.org`) database, as of September, 2012, contains 209,354 non-redundant physical interactions and 42,308 unique proteins from 38 major model organisms. These interactions are derived from both high-throughput and small-scale studies. The Database of Interacting Proteins (DIP; `http://dip.doe-mbi.ucla.edu`) provides a catalogue of experimentally determined interactions. It has about 74,089 interactions and 24,883 proteins belonging to 504 organisms (September, 2012). This database is curated both manually and computationally, the latter using a high-confidence core subset of interactions for making inferences. Finally, the Human Protein Reference Database (HPRD; `http://www.hprd.org/`) is a database dedicated solely to human proteins; it includes not just interactions but also disease associations, post-translational modifications and domain architecture. Currently (September, 2012), it has 30,047 protein entries and 39,194 interactions manually curated from the literature.

All the databases considered in this thesis are primary databases, as they only include PPIs from experimental data curated from small and large-scale published studies [Rivas and Fontanillo, 2010]. Meta-databases, like the Agile Protein Interaction Data-Analyzer [Prieto and Rivas, 2006] (APID, `http://bioinfow.dep.usal.es/apid/`), integrate and unify different public repositories. Prediction databases, such as STRING [Szklarczyk et al., 2010], contain not only experimental PPIs, but also PPIs predicted from other types of data such as gene co-expression profiles.

As the amount of interaction data increases, so does the urge to validate the incoming data and assign confidence scores to the interactions found. Manual literature

curation by experts was recently proven not to be as reliable as initially thought [Cusick et al., 2008]. Moreover, currently the overlap between different databases is very low. For human, APID reports only 853 common interactions between BIND, HPRD and BioGRID. Coverage and PPI reliability also haunts these repositories: even for the well-studied yeast with roughly 6000 proteins and predictions pointing to 13,500-30,000 interactions [Stumpf et al., 2008], we observe 56,363 interactions in BioGRID and 105,728 in APID, indicating either errors in the predictions (based on high-confidence data) or a high number of false-positives.

Efforts to standardise experimental conditions are crucial in order to achieve meaningful validation of the data since changes in these can originate different results; how much this dependency affects the data and the PIN topology is not yet understood. Many initiatives that try to account for this are now emerging such as the Proteomics Standards Initiative (PSI) [Kaiser, 2002] and the Minimum Information about a Proteomics Experiment (MIAPE) guidelines [Taylor et al., 2007]. Both include standard protocols and conduct guidelines proposed as a counter-measure for the data boom associated with the increasing readiness of high-throughput technology to discover PPIs.

Data growth will bring many biological and computational challenges on how to access, manipulate and interpret these huge datasets. Graph theory, a branch of Mathematics which studies graphs, mathematical constructions used to model pairwise relationships between objects of a set, provides the perfect foundation to tackle these problems. We now review some relevant concepts to model PPI networks as graphs.

1.4 Basic graph theory definitions

Here we model PPI data as an undirected graph with proteins as nodes and interactions as edges. This allows us to use a wide range of general formal results in graph theory whose basic terminology we now introduce. An undirected graph G with no self-loops

or multiple edges is a pair $(V(G), E(G))$ where the elements of $V(G)$ represent the set of nodes; the elements of $E(G)$ are called edges, and an edge is a two-element subset of $V(G)$. When $\{v, w\} \in E(G)$ we say v and w are *adjacent*. The *degree* of a node v, $deg(v)$, is the number of edges which have v as one of its endpoints. In a PPI network context this is simply the number of interactions of a particular protein. If $V(G)$ has v elements and $E(G)$ has e elements, then the *average degree* of a graph is defined as $d(G) = 2e/v$. The order of a graph is simply the number of elements in $V(G)$, whilst the size of a graph is the number of elements in $E(G)$.

A subgraph of $G = (V, E)$ is a graph $F = (V', E')$ whose node set V' is a subset of V and its edge set $E' \subseteq E$ connects only nodes of V'. The *maximum average degree*, $m(G)$, of a graph G is the largest average degree over all subgraphs of G. A subgraph F of G is said to be *induced* by V' if it includes exactly all the edges of G which connect the vertices of V', *i.e.*, for each pair of vertices in F and their corresponding pair in G, there will be an edge between a pair of vertices in F if there is an edge between the corresponding pair in G. For example, the only induced subgraphs of a triangle are edges. Two graphs are said to be *isomorphic* if there is a one-to-one mapping f between the vertex sets of G and H such that vertices v and w are adjacent in H if and only if $f(v)$ and $f(w)$ are adjacent in G.

We define the *graph density* ρ of a graph G, with v vertices and e edges as the ratio between the number of edges e and the number of potential edges of G, *i.e.* $\rho = e/\binom{v}{2}$.

Several network measures exist to characterise, classify and model networks [Costa et al., 2007]. These normally focus on a single network aspect and tend to the reflect, in an averaging way, the global network topology. Common network measures or summary statistics are the degree distribution, the average shortest path length and clustering coefficient. The degree distribution, $P(k)$, is the fraction of the nodes in a network with degree k, *i.e.*, the probability that a node chosen uniformly at random has k edges. Thus, $P(k)$ is obtained by dividing the number of nodes $N(k)$ with $k = 1, 2, ...$

edges by the total number of nodes N. The shortest path, d_{ij}, between nodes i and j is simply the path through the network with smallest number of edges. We can then calculate the average shortest path l such that $l = \frac{1}{N(N-1)} \sum_{i \neq j} d_{ij}$. For unconnected graphs, only the lengths of the existing paths are considered and contributing to the average. Another popular measure is the clustering coefficient. Several definitions exist, here we consider the clustering coefficient $C_i = 2n_i/(k(k-1))$, which finds the number n_i of edges connecting k neighbours of node i to each other. The network or average clustering coefficient is then $C = \frac{1}{N} \sum_i C_i$. In Section 1.7 we report some of these network measures when applied to PPI networks.

1.4.1 Network sampling

Sampling a network is a distinct problem from the classical statistical inference problem of sampling a whole population because of the existence of the two, node and edge, sets that comprise a graph. Characteristics like the degree distribution or other network properties are not exclusive properties of the nodes, but depend heavily on the network structure (edge dependencies) [Lee et al., 2006]. Several types of sampling are used and they all provide different information. The most common are node sampling, where nodes are randomly chosen with a given probability and edges between them kept; edge sampling, where edges are picked at random and their respective nodes kept, and snowball sampling. In the latter, a single node is picked at random and all the nodes adjacent to it kept up to a given radius (normally in units of edges away from the picked node). The resulting network samples are called *ego-networks* and include all the edges present in the original network between the selected nodes. As we shall see in Chapters 4 and 5, network sampling can be employed to directly measure and model heterogeneity in PPI networks as well as to compare them. Nonetheless, perhaps the most valuable concept from graph theory is that of a random graph model which we now introduce.

1.5 Random graph models

Modelling networks can be very useful not only to understand their overall topology and local interaction patterns, but also to reveal the underlying mechanisms of their formation and functionality. Random graph models specify a probability space over all possible graphs and come in handy as tools for hypothesis testing when analysing real-world networks with unknown organising principles and complex topology [Albert and Barabasi, 2002]. This approach has the underlying assumption that the network under study was generated by some unknown stochastic process(es) which can still be summarised by a relatively simple mathematical model. We now follow with a more in-depth description of some of these models, namely the Erdős-Rényi random graphs, geometric random graphs, ER mixture graphs, Exponential Random Graphs and growth models such as the duplication and divergence model and the preferential attachment; but many others exist. For more background on random graphs, see for example Bollobás [2001].

1.5.1 Erdős-Rényi random graphs

In a series of seminal papers Erdős and Rényi introduced several versions of the so-called Bernoulli or Erdős-Rényi random graph (ER) model [Erdős and Rényi, 1960]. The particular ER model $G_{n,m}$ has n labelled nodes connected by m edges which are randomly chosen from the $\frac{n(n-1)}{2}$ possible edges. A total of $C^m_{\frac{n(n-1)}{2}}$ graphs with n nodes and m edges can be constructed, each one of them being equiprobable. Another version is the $G_{n,p}$ model in which each possible edge between the n nodes is included independently at random with probability p. Graphs of the $G_{n,p}$ model have on average $\binom{n}{2}p$ edges. ER graphs need not be connected and hence can generate networks with singletons, nodes for which there are no edges. For real graphs with a known number of nodes and edges ER models have no unknown parameters. The $G_{n,m}$ model is

constructed by simply matching the number of nodes and edges to the ones of the input graph. In this particular model the choice of an edge is not completely independent, *i.e.* the choice of an edge has an effect, fortunately small, on the choice of another edge [Bollobás, 2001]. The degree of a randomly chosen node will be approximately Poisson distributed with mean $2m/n$ as $n \to \infty$. The local architecture of an ER network tends to be tree-like, more precisely, in the large network limit the probability of loops between a small number of nodes tends to zero [Bollobás, 2001]. ER models are well-studied and several theoretical results concerning subgraphs exist (see Section 1.6).

A variation of this model is an Erdős-Rényi graph with fixed degree distribution (ER-DD), also called the configuration model. Given an input graph, ER-DD is constructed to have not just the same number of nodes and edges as the input graph, but also the same degree distribution. This can be achieved by assigning the degree sequence of a given graph to the nodes, which then are sorted in decreasing order; edges are generated from the sorted nodes towards randomly selected ones, preserving the assigned degree distribution, see the Molloy and Reed construction (Durrett, 2007, page 79). This construction can create multiple edges and self-loops which may result in an impossible network if we are strict with the number of nodes and edges, and hence several experiments have to be carried out before the desired ER-DD graph is generated. These depend on the number of nodes, edges and degree distribution of the target graph, but tend to be rare for graphs with high order [Newman et al., 2001]. For a more efficient method of generating ER-DD graphs see Blitzstein and Diaconis [2011].

1.5.2 Geometric random graphs

Early works on finite geometric random graphs or proximity graphs started with the study of the topological structure formed by random clumps, i.e, potentially overlapping

objects placed at random in space (coincidence-graphs) [Hafner, 1972]. Since then, many independent research groups have worked on these graphs and several asymptotic results are now available for large finite graphs [Penrose, 2003].

Geometric n-dimensional random graphs (GEOnD) are constructed by assigning each node random coordinates in an n-dimensional box of unit volume, *i.e.*, coordinates are drawn from a uniform distribution on the unit interval [Penrose, 2003]. Points in the metric space will then correspond to graph nodes. Two nodes will be connected by an edge if the distance between them is at most r. "Distance" is an arbitrary distance norm in the space considered and can take many forms. Thus, the model requires estimation of two parameters: the spatial dimension (generally arbitrarily chosen) and the radius r, as well as the specification of the distance to be used.

According to Przulj et al. [2004], geometric graphs are typically used for modelling designed and optimised networks (as opposed to networks that emerged simply through stochastic growth processes without optimisation) such as wireless multi-hop networks and electric power-grids. Przulj et al. [2004] also applied this model to PPI networks despite the lack of a "natural" distance.

1.5.3 Erdős-Rényi mixture graphs

Empirical degree distributions of real graphs are sometimes very different from the approximate Poisson distribution implied in the classical Erdős-Rényi model. Real graphs also have higher clustering coefficients than the graphs generated by this model. Nowicki and Snijders [2001] tried to address this issue by proposing the ER "block model". Using a Bayesian approach, this model assumes that vertices belong to classes with different connectivity characteristics, but still with independent edges. It is a full probabilistic model able to estimate and predict *a posteriori* block structures, but the proposed estimation method can only deal with networks up to 200 nodes [Daudin et al., 2008]. Fitting this model requires inference on the number of classes and the mixing

parameters, *i.e.* the mixing matrix which defines the probability of connection between the different classes. Daudin et al. [2008], using a frequentist approach, provided both a fast estimation algorithm and a model selection criterion to choose the number of possible classes, making this model amenable to the typically large biological networks with thousands of nodes.

Stickiness index random graphs

PPI networks were also modelled using a tailor-made random graph model that uses a stickiness index (STICKY). This model, proposed by Pržulj and Higham [2006], is a special case of the block model [Nowicki and Snijders, 2001], *i.e.*, an Erdős-Rényi mixture graph (ERMG) with a fixed number of classes. It has two main assumptions: having a high degree is synonymous for having many interactions and any two proteins are more likely to interact if both their stickiness indices are high. These model networks are constructed by assigning a stickiness index θ_i to every node i for all N nodes where $\theta_i = \frac{deg(i)}{\sqrt{\sum_{j=1}^{N} deg(i)}}$ (see Section 1.4). Each node i is then matched with all nodes. For each match with a node j a random number r is drawn from a uniform (0,1) distribution and, if $r \leq \theta_i\theta_j$, an edge is formed, *i.e.*, the product of the two stickiness indices defines the probability of interaction. This model, originally inspired by the concept that two high-degree proteins are more likely to interact with one another than with proteins of lower degrees, was shown to outperform GEO in modelling some PPI networks under global and local measures [Pržulj and Higham, 2006].

1.5.4 Randomisation algorithms

A model commonly used for PPI networks simply applies a randomisation protocol to the network. A single randomisation step consists in picking two random edges, say A–B and C–D, and rewiring them around so that A connects to D and B connects to

C, provided the two newly chosen edges do not exist, otherwise a new pair of edges is picked. Since the procedure entails an edge swap the degree of a given node will be preserved. The original algorithm proposed by Maslov and Sneppen [2002] employs a number of randomisation steps equal to four times the number of edges of the input network. In Chapters 2 and 4 we find that networks of this model are inadequate to represent PPI networks in counts above the edge.

1.5.5 Exponential random graphs models

Exponential random graphs (ERGM), also called p^* models, were first introduced by Frank and Strauss [1986] and currently are mainly used in the social network analysis field. In this model, the empirically observed network is the outcome of a stochastic process, just a realisation from the set of possible networks with similar important characteristics. The network is perceived as a self-organising system of edges related to each other. For a fixed node set, the range of possible networks and their probability of occurrence is represented by a probability distribution on the set of all possible graphs for the node set. The generalised version of ERGMs can then theoretically represent any finite random graph model although, practically, difficulties concerning model specification and parameter estimation arise.

In this model every edge is treated as a random variable and one can introduce the notation y_{ij} for the observed value of the random variable Y_{ij} which specifies the presence ($Y_{ij} = 1$) or absence ($Y_{ij} = 0$) of an edge between nodes i and j, Y represents the matrix of all variables and y the matrix of the observed edges in the network under consideration. The probability of observing a particular graph y (observed network) in the probability distribution of all graphs on n nodes, Y, is given by the general formula

$$P(Y = y) = \frac{1}{\kappa} \exp\left(\sum_A \eta_A g_A(y) \right),$$

where κ is a normalising constant that ensures a proper probability distribution, η_A a parameter for the configuration A and $g_A(y)$ represents the network statistic of A with $g_A(y) = \prod_{y_{ij} \in A} y_{ij}$, *i.e.*, the statistic for the configuration A is the product of the variables that specify the presence or absence of that particular configuration for the nodes i and j for all n nodes in the network; $g_A(y) = 1$ is the configuration is observed in the network, 0 otherwise. The probability of observing a particular network, $P(Y = y)$, is both dependent on the statistic $g_A(y)$ and non-zero η_A for all configurations in the model. For a given configuration A, η_A is zero only if all pairs of variables in A are conditionally independent.

For this model, critically specifying dependency assumptions between the different configurations is of special importance, in fact, the only relevant configurations are the ones whose variables are contingent on each other. Models which include subgraph counts are also possible, although dependence assumptions are not always clear [Wasserman and Pattison, 1996]. In an Erdős-Rényi random graph the only dependency assumption is that all edges are independent of one another and hence the only possible configuration relates to single edges being present or not. The general formula is then reduced to

$$P(Y = y) = \frac{1}{\kappa} \exp\left(\sum_{i,j} \eta_{ij} y_{ij} \right).$$

To get the classical ER model we can further assume that the probability of an edge is equal throughout the network (homogeneity assumption) and make $\eta_{ij} = \rho$, for all i and j, a parameter controlling the graph density, and $\sum_{i,j} y_{ij}$ simply becomes the number of edges in the graph y.

1.5.6 Growth models

Models which use an iterative random graph generation process are called growth models. At each iteration, a new node is added to the graph followed by specific edge creation dynamics where the newly added node is connected to node(s) in the network. This can be done by linking to a fixed number of existing nodes, or the number of edges itself is treated as a random variable. Several options for choosing the number of new connections and their endpoints are available according to the particular model being considered. Here, we focus on two popular models for PPI networks: the preferential attachment (PA) model, and the duplication and divergence (DD) model.

1.5.6.1 Preferential attachment model and scale-free networks

Scale-free (SF) random networks is an umbrella term for networks with power-law degree distributions. The term "scale-free" refers to the fact that the shape of the distribution is preserved regardless of the scale in use. They were first proposed by de Solla Price [1965] to describe a network of citations between scientific papers. Later this model was 're-discovered' by Barabasi and Albert [1999], who called it *preferential attachment* (PA). By exploring several real-world large networks, the authors showed that in many of them the probability $P(k)$ of a node interacting with k other nodes decays as a power-law following $P(k) \sim k^{-\gamma}$, with γ generally between 2 and 3.

SF networks attempt to recreate this feature. They are constructed with two operating mechanisms: new nodes are sequentially added and attach preferentially to nodes that are already well connected, that is, the attachment probability is proportional to a power of the degree of the target node. The networks created are therefore always connected and have a small number of very highly connected nodes, generating heavy-tailed degree distributions. These scale-free type of models of network growth are popular; many real-world networks such as metabolic reaction networks [Jeong et al., 2000] and the World Wide Web [Broder et al., 2000] appear to have power-law

degree distributions and have been modelled using SF networks. Parameter estimation for these networks will depend on the particular model being used and, generally, it is not clear which estimation method is preferable; Schwoch [2008], for instance, used the method of moments for this purpose.

Despite the attractiveness of the elegant scale-invariance (also called self-similarity) property, which would provide a simple underlying principle for many real-world networks, the degree distribution of PPI networks was shown to be a poor fit to a power-law distribution [Khanin and Wit, 2006]. Furthermore, the same type of qualitative behaviour could be achieved by other heavy-tailed distributions such as truncated power-law, generalised Pareto law, stretched exponential distribution, geometric distribution or any combination of the above.

1.5.6.2 Duplication and divergence models

Network growth can also happen via node duplication. This mechanism has been used extensively to model PPI networks and proteome evolution in general, as well as peer-to-peer networks. It states that, at each iteration t, a node (parent) selected uniformly at random is duplicated with all its edges. A step of divergence (also called mutation) follows where the edges of the duplicate node (child) are deleted with probability q_{diff} (subfunctionalisation) and new edges between the duplicate node and any of the existing nodes formed with probability r/t (neofunctionalisation) [Bebek et al., 2006]. Several versions of this basic model exist. In the Bebek et al. [2006] model the node duplication is asymmetric as only edges involving the child node change during divergence. Another model proposed by Middendorf et al. [2005] specifies a symmetric node duplication model, also called duplication attachment preserving complementarity (DMR), which allows edges to be lost from both parent and child nodes upon divergence. The DMR model does not account for neofunctionalisation events and tends to create singletons; this is fixed in the Bebek et al. [2006] model but at the expense of

an extra parameter. Both models have an heterodimerisation parameter for connecting the child and parent nodes with probability q_{con}. The topology of the resulting simulated networks was found to be dependent on the seed network used to initialise the algorithm [Hormozdiari et al., 2007]. The application of these models to PPI networks found its basis in Ohno's theory of genome evolution [Ohno, 1970] which puts gene and genome duplications as the major driving force behind the creation of new genomic material; in Ohno's words: "natural selection merely modified, while redundancy created". In its most simple form, this model uses two independent parameters: the average degree of the network, easily estimated from real PPI networks and the deletion rate of newly created edges, obtained by calculating the ratio of addition and deletion rates in the proteome, which is based on the interactions of known duplicates (paralogs) [Pastor-Satorras et al., 2003]. Currently, the important question being asked is if parameters leading to graphs with topologies close to real PPI networks do have a biological meaning. For instance, Bebek et al. [2006] showed that, provided $r > 0$ and $1 - q \leq 0.58$ the model generates graphs whose degree distribution follow a power-law. Gibson and Goldberg [2011], who previously questioned the importance of neofunctionalisation [Gibson and Goldberg, 2009], re-conceptualised the model including the notion of protein domains and making it the focus of subfunctionalisation as opposed to the interactions per se. In this model, p represents the probability of a self-interacting domain being preserved and q is now, upon selection of a particular domain, the probability of losing an interaction involving that domain. This model accounts for heritable homodimers and asymmetry in the subfunctionalisation can be readily introduced. Despite these improvements, the model is still far from mimicking the topology of real PPI networks - for instance, just looking at the counts of three node cliques (triangles), the best parametrisation generates graphs with only about 50% of the counts in the real yeast PPI network [Gibson and Goldberg, 2011].

In this dissertation we concentrate on the use of subgraphs in analysing and comparing networks. For some well-studied models, it is possible to estimate when subgraphs like triangles start to appear as more edges are added to a graph and the graph density increases. We next focus on these thresholds for the appearance of subgraphs for the ER and GEO models.

1.6 Thresholds for subgraph appearances

Many theoretical properties of graphs change dramatically in a narrow range of graph density, which led to the concept of *threshold functions* [Erdős and Rényi, 1960]. Given a good model for the topology of PPI networks, threshold functions might be a valuable tool for inference on mechanisms of network design. These functions could help explaining network features such as robustness to errors, faults and attacks, which have been previously linked to network topology and subgraphs [Brady et al., 2009; Dekker and Colbert, 2004; Poncela et al., 2007]. We now start by defining a threshold function and then present some known results for estimating when with increasing graph density certain subgraphs are expected to appear, given a particular model.

Let $G_{n,f(n)}$ be a family of random graphs induced by n, the number of nodes, and $f(n)$, a function that gives edges according to the specific model. If Q is a graph property, $P(Q)$ denotes the probability that $G_{n,f(n)}$ has or belongs to Q. We say that *almost every graph* in $G_{n,f(n)}$ has the property Q if $P(Q) \to 1$ as $n \to \infty$. For a given monotone increasing property Q (such as the appearance of a certain subgraph), we define a threshold function $t(n)$ for Q as any function which satisfies

$$P(Q) \to 0 \quad \text{if} \quad \frac{f(n)}{t(n)} \to 0, \quad \text{and} \quad P(Q) \to 1 \quad \text{if} \quad \frac{f(n)}{t(n)} \to \infty. \tag{1.1}$$

Threshold functions for the ER model are not unique, although they are so within certain factors (Bollobás, 2001, p. 40).

Since we are concerned with network topology and subgraph counts, it is useful to apply these threshold functions to subgraphs and calculate, for the case of well-studied models, where these are expected to be present. For the ER random graph model $G(n, M(n))$, *i.e.* $f(n) = M(n)$, it is possible to show that the threshold function for the property of containing a fixed, non-empty graph F is $n^{2-2/m}$, where $m = m(F)$ is the maximum average degree of F (see Bollobás, 2001, p. 89). We relate $M(n)$ and the graph density ρ via $\rho = M(n)/\binom{n}{2}$.

For the ER model it is possible and more informative to calculate the graph density such that the expected number of copies of a given subgraph F is approximately 1. For a subgraph on v vertices with e edges, the approximate expected count for the subgraph under the ER model is

$$E(\text{number of occurrences}) = \lambda = \binom{n}{v} p^e (1-p)^{\binom{v}{2}-e} \sim n^v p^e / v!, \tag{1.2}$$

for small p. When the number of occurrences is well approximated by a Poisson process, as in the case for balanced graphs, $P(\text{no occurrence of subgraph}) \sim 1 - e^{-\lambda} \sim \lambda$ and hence the threshold function and the expectation formula coincide [Bollobás, 2001].

Threshold functions for GEO3D models are not so well understood. One can, nonetheless, calculate approximate threshold values for the appearance of induced subgraphs with k vertices. Analogous to the definition of threshold for ER graphs, we considered $f(n) = r(n)$, *i.e.* the number of edges will be a function of the radius in the GEO3D model. Further, monotone properties for GEO graphs have threshold functions which are almost unique [Goel et al., 2005]. Penrose [2003] showed that for a random geometric graph placed in $\mathbb{R}^d$ with n vertices and a radius r, the k-vertices subgraph count satisfies a Poisson limit when the product $n^k r^{d(k-1)}$ tends to a finite constant. The radius r can be related to the average degree α by using the gamma function $\Gamma(x)$

[Dall and Christensen, 2002],

$$r = \frac{1}{\sqrt{\pi}}\left[\frac{\alpha}{n}\Gamma\left(\frac{d+2}{2}\right)\right]^{1/d}. \tag{1.3}$$

Solving for α in (1.3) gives the threshold graph density ρ as

$$\rho = \frac{\alpha n/2}{\binom{n}{2}}. \tag{1.4}$$

In Chapter 2 we make use of these formulæ to estimate the graph density at which 2-5 node subgraphs start to appear for ER and GEO3D models with 500, 1000 and 2000 nodes (see Tables 2.2 and 2.3, respectively). There, we also present evidence that the graph density of PPI networks is roughly in the same region as the thresholds for the appearance of small subgraphs in these models which tentatively leads to hypothesising PPI networks as liminal networks. Proving this hypothesis is not trivial; a good model for PPI networks would be needed.

The next section gives an introduction to some characteristics of proteins and PPIs which are relevant to model and understand the nature of these subgraphs in PPI networks.

1.7 Characteristics of PPI data

Comparatively, we now know far more about the proteins themselves, as individual units or components, than we know about the way they interact globally and form the network topologies we observe. Little is known about the evolution of these networks. PPI networks are taken to be modular, highly organised and selected through evolution for efficiency and robustness [Klemm and Bornholdt, 2005; Valente and Cusick, 2006; Vespignani, 2003], but current models and statistics fail to illustrate why this is so. For

instance, although the modularity of cellular networks is a well-established concept, little is known about the modules and their significance. These modules span multiple scales, ranging from communities and pathways to small subgraphs. Functional homogeneity can be found at many levels [Lewis et al., 2010]. Subgraphs are sometimes referred to or confused with network motifs [Milo et al., 2002]; the latter are small interconnected subgraphs whose counts are over-represented in a given network when compared to randomised versions of it. This concept requires these randomised graphs to be a good null model for the target network, which so far has not been proved convincingly [Artzy-Randrup et al., 2004]. In fact, we provide evidence in Chapter 2 that this is probably not the case for higher-order subgraphs in PPI networks. Network motifs have been studied mainly in the context of transcriptional regulatory networks where they have a direct interpretation as regulation patterns of gene expression. However, some studies suggest that motifs might also play an important role in PPI networks [Bachman and Liu, 2009; Liu et al., 2011; Wuchty et al., 2003].

In order to concentrate on meaningful modules, a statistical model for PPI networks which appropriately approximates the occurrence of small subgraphs would be useful [Artzy-Randrup et al., 2004], but currently non-existent [Rito et al., 2010] (see Chapter 2). The lack of good models is likely to reflect a lack of understanding of the underlying network.

We next discuss some of the most popular network summary statistics used to describe the overall network, explaining their scope and limitations. We then present some characteristics of proteins (nodes) and properties of their interactions (edges) which might be of use in modelling PPI networks.

1.7.1 Global network summary statistics

Despite the advances in experimental high-throughput technology, currently available PPI networks are but a sample of the true network, generally thought of having thou-

sands to tens of thousands of nodes and a number of edges between tens and hundreds of thousands. These networks can be thought of as large, unconnected sparse graphs. The DIP database for yeast alone [Salwinski et al., 2004] (as of 10^{th} October 2010 and with self-loops removed) has 25,233 interactions between 5,213 proteins, although single studies are also often used in analyses, for instance, von Mering et al. [2002] has compiled a high confidence subset of the network consisting solely of interactions supported by more than one experimental method, which resulted in a network with 988 nodes and 2455 edges. The number of edges in PPI networks roughly scales with n^2, where n is the number of nodes of the network. This implies that PPI networks have low graph densities, typically between 0.001 and 0.005 (a value of 1 would correspond to the complete graph). The sparseness of these networks is a feature that should be taken into account when modelling them. A consequence of this sparseness is a relative inherent sensitivity to perturbation and link removal, particularly when assessing the significance of topological structures; the design of algorithms for these networks should also take this sparseness into consideration (for an example see Mirshahvalad et al. [2011]).

The relatively large sizes of PPI networks mean that network characterisation and comparison has been typically carried out using network summary statistics. Typical summary statistics are the degree distribution, the mean path length and the clustering coefficient (see Section 1.4 and Costa et al. [2007] for an overview and many more summary statistics).

More useful than reporting the average degree of nodes, roughly 2 to 5 for proteins in PPI networks, is considering the degree distribution of the network (see Section 1.4). The degree distribution has been one of the most used summary statistics for PPI networks. It can be characterised as a heavy-tailed distribution (see Figure 1.1 (*left)*) which implicitly means that most proteins have low degrees and only a few rare ones are highly connected. These high-degree proteins, so-called protein hubs, have been

linked with sequence conservation [Wuchty, 2004] and essential genes; indeed genome-wide studies have shown that deletions of hub proteins are more likely to be lethal [Yu et al., 2004]. If this is a property of the protein itself or an artefact created by certain PPIs being more important than others is still a matter of debate [He and Zhang, 2006]. In Chapter 3, we present evidence that high-degree proteins are associated with highly clustered parts of the network [Rito et al., 2012]. As for the specific shape of the distribution, due to the seemingly linear dependency of frequency with degree in a log-log plot (Figure 1.1 (*right*)), Barabasi and Albert [1999] hypothesised it to be a scale-free, power-law distribution (see Section 1.5.5.1), although this has been shown to be a crude approximation with other heavy-tailed distributions providing a goodness-of-fit which is just as good [Khanin and Wit, 2006].

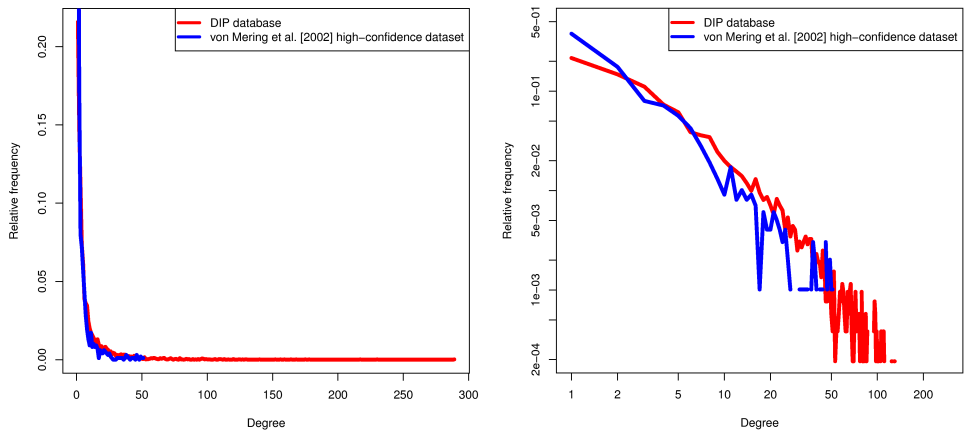


Figure 1.1: Degree distribution of yeast PPI networks with interactions taken from the DIP database [Salwinski et al., 2004] and from von Mering et al. [2002]. *(left)* Plot of the node degree versus its relative frequency. *(right)* Log-log plot of node degree versus its relative frequency. Plots were generated using the *igraph* package in R.

The average shortest-path length is also a frequently considered measure. It characterises the network overall navigability and is calculated by averaging over the shortest paths between all pairs of nodes. Given their order and size, PPI networks have rela-

tively low values of average shortest-path lengths, typically of 5 to 6.

Another common summary statistic is the clustering coefficient, which is basically the number of triangles that go through a node. The average clustering coefficient informs, in a crude way, the overall tendency of nodes to form clusters. PPI networks are known to have high average clustering coefficients which can reach values up to 0.5 (for the von Mering et al. [2002] high-confidence dataset); a value of 1 would mean that every node in the graph would be part of a triangle, *e.g.*, all cliques have clustering coefficients of 1. Low values of average shortest-path length together with high clustering coefficients are often taken as indicators to characterise the network as a small-world network, a broad type of graphs where in spite of most nodes not being adjacent to one another, most nodes can be reached from a randomly selected one through a relatively short path.

Albeit useful, global network summary statistics only provide a shallow characterisation of the network and often many different types of graphs can be obtained given a fixed set of these measures. Bottom-up strategies like the analysis of subgraph counts mentioned earlier try to ameliorate this, although, as we shall see in Chapters 4 and 5, accounting for the distribution of these summaries across the network is also important for accurately modelling the topology of PPI networks.

1.7.2 Characteristics of the nodes

In PPI networks, proteins are the nodes and currently a fair amount is known about them. These molecular machines constitute the building blocks and work-force of the cell, being responsible for controlling gene expression, most of the enzymatic catalysis, intracellular signalling and maintaining the cell's structural integrity. A full understanding of protein interactions which, ultimately, results in the overall topology of PPI networks and hence in the global network characteristics presented in the previous section, can only come from a detailed knowledge of the three-dimensional (3D)

structures of these complexes and the exact circumstances in which they are formed. However, determining 3D structures of proteins is a difficult, slow and expensive task, let alone 3D structures of protein complexes [Aloy and Russell, 2006]. Bioinformatics is expected to ease this task by discovering patterns in both PPIs and in existing 3D structures capable of informing us of the nature of these interactions.

Proteins are rich in attributes that can be used as explanatory variables for the interactions and the overall topology we observe in PPI networks. Examples of such nodal attributes include structural [Murzin et al., 1995; Orengo et al., 1997], functional [Harris et al., 2004; Mewes et al., 2002] and phylogenetic characteristics of proteins. A particularly good example of the latter is the spread of orthologs across different species. Protein orthologs are identified through their nucleotide sequences and correspond to genes sharing a common origin from a single ancestral gene in the last common ancestor of the compared genomes [Koonin, 2005]. Although several caveats come with this definition [Studer and Robinson-Rechavi, 2009], the number of orthologs of a protein given a particular phylogenetic tree, called lineage specificity, can provide us with a proxy for the "age" of a protein. Lineage specificity ranges between 0 and 1, where 0 is assigned to a protein which is only present in the proteome of interest, or in it and a few other highly related organisms and 1 corresponds to a protein whose appearance can be traced to the most recent ancient branching of the tree under consideration. Lineage specificity can be calculated by using databases of orthologs such as InParanoid [Östlund et al., 2010] and OrthoMCL [Li et al., 2003] (for examples see Winstanley et al. [2005] and Liu et al. [2011]) or by using taxonomic distributions of protein domains [Kim and Marcotte, 2008].

Protein characteristics such as protein age, native intracellular compartment, non-covalent interaction properties of its surface, type and number of interaction patches, cellular function, type of fold, amongst others, add important information to the problem of inferring PPIs [Chen et al., 2008; Sharan and Ideker, 2006]. Some characteristics

cannot be assigned to individual proteins, but are a phenomenon of the interaction. We present some of these interaction properties in the next section. Together with node attributes they may unveil topology patterns that help to explain the architecture of PPI networks.

1.7.3 Characteristics of the edges

As mentioned above, edge formation can be the result of nodal attributes and some interaction preference patterns can be detected, but the interactions (edges) themselves can have attributes that are derived from pairwise phenomena and cannot be assigned to a single protein. A fundamental characteristic of an interaction is a measure of how strong it is. This is captured by the association constant, the ratio between the rates of binding and dissociation. Obtaining these for PPIs would be helpful for understanding how the cell works, mainly because the different interaction affinities can vary from very high values, with negligible dissociation rates, to very low ones that would require non-physiological protein concentrations for sufficient amount of the complex to be detectable and hence cast doubt on its significance. Unfortunately, association constants are difficult to obtain and require laborious experiments to be conducted which are hard to scale up to a genome-wide level [Aloy and Russell, 2006].

Another type of data that might be able to provide information of which interactions are occurring in the cell at a particular condition is gene expression data. Epigenetic modifications and chromatin structure dictate which and to which extent genes are transcribed in the cell. These transcripts are then matured and eventually get translated by the ribosome into proteins. Gene expression data is mostly obtained using the oligonucleotide expression array technology [Lockhart et al., 1996] which measures the abundance of many (genome-wide) mRNA transcripts simultaneously. Although studies have shown a poor correlation of these data with actual protein abundance [Greenbaum et al., 2003].

An early attempt at integrating PPI networks and gene expression data was by Ge et al. [2001]. Using the k-means algorithm, Ge et al. [2001] clustered genes with similar expression levels and investigated the relationship between these expression clusters and the density of protein interactions within and between these clusters, finding evidence that genes with similar expression profiles (within the clusters) encode for proteins which are more likely to form physical interactions. However, this correlation is weak and highly dependent on the organism under consideration, although integrating PPI data from ortholog proteins was showed to improve it [Bhardwaj and Lu, 2005]. This correlation also seems to be stronger when we consider expression levels within network motifs, particularly as the complexity of the motif increases (both number of edges and nodes) [Bhardwaj and Lu, 2009].

Besides their physical dimension, protein interactions can also be perceived in a more functional, abstract way, namely one can look at the phenotypic effects of mutating two proteins versus the phenotype of the corresponding single mutants alone and hence measuring the importance of this synergy between a given protein pair - these are called genetic interactions. Genetic interactions derive from mutant screen assays which test whether two mutations have a combined effect on the growth of yeast that is not exhibited by either mutation alone. The state-of-the-art is the quantitative technique epistatic miniarray profiles (E-MAPs) where colony size is modelled as a multiplicative combination of the double mutant fitness, time and experimental factors such as spatial effects, nutrient competition and screen batch effects. The output is a Synthetic Genetic Array (SGA) interaction score with an associated confidence measure (p-value) [Costanzo et al., 2010].

Genetic interactions can be positive, if the double mutants have a fitness impairment that is lower than expected by combining both single mutants, or negative, if the fitness impairment is more than expected, which, in the extreme case, can lead to synthetic lethality. Only recently these functional genetic interactions are being

interpreted in the light of physical PPIs. About 40% of yeast synthetic-lethal interactions can be clearly incorporated into a physical network encompassing protein-protein, protein-DNA and metabolites "interactions". Of these genetic interactions, roughly three-and-a-half times as many are associated with between-pathway models as opposed to within-pathway, *i.e.*, most genetic interactions occur between proteins which are not directly wired-up in a densely connected set of proteins in the physical interaction network (pathway) [Kelley and Ideker, 2005]. Costanzo et al. [2010] observed an overlap of 10-20% between PPI and genetic interaction datasets, whereas the random expectation is roughly 3%. They also reinforced the conclusion that most of both positive and negative genetic interactions appear to occur between, rather than within, complexes and pathways. Recently, Bandyopadhyay et al. [2010] focused on differential genetic interaction networks. These networks are constructed by taking the difference between SGA interactions scores across different conditions. Using wild-type yeast versus yeast treated with a DNA-damage inducing agent Bandyopadhyay et al. [2010] found these differential genetic interactions to be strongly associated with physical networks, namely the sets of positive and negative genetic interactions were highly enriched in proteins known to interact. These findings can inform and enrich PPI network modelling by helping to pin-point biologically relevant modules in the network and by increasing our PPI predictive power.

1.8 Thesis overview

The ability to compare different networks is important for model selection and may provide insights into the stochastic mechanism behind the evolution of PPI networks. In Chapter 2, we begin by presenting the main results found in our analysis of the Relative Graphlet Frequency distance (RGF) [Przulj et al., 2004] and the Graphlet Degree Distribution Agreement (GDDA) scores [Przulj, 2007] used for network comparison. We find that both RGF and GDDA have a pronounced dependency on the number of edges and vertices of the networks being considered and this should be taken into account when testing the fit of models. We provide a method for assessing the statistical significance of the fit between random graph models and biological networks based on non-parametric tests finding that none of the tested models fit to the current PPI networks [Rito et al., 2010]. We also employ principal component analysis on a matrix of the standardised counts of 2-5 node subgraphs as a method to both assess model fit and to help build an understanding of where current models fail. We observe that for the yeast high-confidence PPI network, the top 5 coefficients in the first principal component, which explains 49% of the variation in the data, corresponds to counts of subgraphs which include at least one triangle in them. Here we also include the fit of popular growth models such as the preferential attachment and the gene duplication and divergence models; we also test models of the exponential random graph family. None of the models tested satisfactorily fit to the PPI networks. Notably, the GDDA and RGF scores are not stable in the region of graph density relevant to the PPI networks, which coincides with the threshold region for the appearance of small subgraphs - for the models where this can be calculated. We hypothesise that PPI networks are liminal networks, at the threshold for the appearance of subgraphs and that such threshold behaviour may be linked to the robustness and efficiency properties of the PPI networks.

The failure to model subgraph counts is probably a consequence of poor understanding of the organisation of the network. In Chapter 3, we analyse networks whose protein nodes have as label a proxy for protein age and look for patterns in the pairwise data and triangles, and at the effect of high-degree proteins on these patterns. We find that pairwise and triangle interactions between Old proteins are over-represented, even, for the triangle case, when controlling for pairwise interaction frequencies. Previous claims of homophily in interactions of the different age groups appear in our models to be solely driven by Old proteins. There is evidence for negative selection of interactions between Middle-aged and Old proteins within triangles, despite pairwise Middle-Old interactions being rather common. Most triangles consist solely of vertices with high degree. Altogether, our findings point towards an architecture of the yeast PIN that is highly heterogeneous, having connected clumps which contain a large number of interacting Old proteins and with selective age-dependent interaction patterns.

To tackle the heterogeneity we observe, in Chapter 4 we sample and characterise neighbourhoods (ego-networks) within PPI networks. We find that even basic summaries of these samples, such as their number of nodes, edges and triangles, have wide distributions which are hardly matched by current random graph models. We also observe that proteins with some particular biological properties are preferentially associated with different types of neighbourhoods. The extent to which these patterns are selected for by the cellular system is unknown and will probably require fine-tuned methods of network comparison to better understand network architecture and evolution.

In the Chapter 5 we propose our own method for network comparison. The method compares neighbourhoods within the query networks in a many-to-many fashion. It relies on occurrences of subgraphs to judge neighbourhood similarity and uses statistics akin to those used in alignment-free sequence comparison methods to combine these in a distance measure. Our method performs well at distinguishing and correctly clustering

networks of different random graph models. When applied to PPI networks we are able to construct correct phylogenetic trees solely based on PPI data. Current phylogenetic molecular data is mostly derived from sequence data. Here we show that PPI data retain evolutionary information and present new possibilities to further explore it.

Chapter 2

Network comparison using subgraphs and threshold behaviour

In this chapter I start by analysing two popular existing measures used to compare networks - the RGF and the GDDA scores. These scores are based on subgraphs counts and show high volatility in the low graph density region relevant to PPI networks. This is true even when comparing networks of the same model with each other. I then discuss the implications of this behaviour for network design and architecture and show how GDDA-based comparisons can be ameliorated using non-parametric statistics. The majority of the work in this chapter was published in Rito et al. [2010].

2.1 On the use of subgraphs for network comparison

The aim in this chapter is to compare biological networks and random graph models under the aspect of similar subgraph counts. Such subgraph counts were introduced by Milo et al. [2002] with the aim of detecting over-represented small subgraphs. They compared counts of 3-4 node connected subgraphs in real-world networks to those of

certain random networks, and called over-represented patterns *network motifs*. Typical algorithms for motif discovery consist of first counting the occurrences of subgraphs of a specific size and type in a graph, then considering isomorphisms between them and finally evaluating the statistical significance of these frequencies by comparing to those of some randomised networks (see Ciriello and Guerra, 2008 for a review). Two main tasks can be distinguished: counting the occurrences of the non-isomorphic subgraphs and evaluating their statistical significance.

Counting small connected subgraphs in large PPI networks is computationally demanding. Moreover, the number of possible n-node subgraphs increases exponentially with n, *e.g.*, for $n = 3$ we have 2 differently connected subgraphs, and 21 for $n = 5$. Przulj et al. [2004] disregarded the frequency subjacent to the definition of motifs and counted 2 to 5-node connected induced subgraphs, which they call *graphlets* (Figure 2.1). Methods have also been developed to count 6-node and 7-node graphlets (Hormozdiari et al., 2007, Grochow and Kellis, 2007). Alon et al. [2008] used a combinatorial colour coding technique to count up to 10-node non-induced subgraphs, arguing that these are more relevant to compare the incomplete and noisy networks currently available.

In this chapter, we use the software tool GraphCrunch [Milenkovic et al., 2008] to examine the use of subgraph counts for network comparison. The software uses a brute force approach to enumerate all 3-5 node connected subgraphs. To combine the distributions that result from subgraph counts, the so-called *Relative Graphlet Frequency (RGF) distance* [Przulj et al., 2004] and the *Graphlet Degree Distribution Agreement (GDDA)* have been suggested [Przulj, 2007]. Focusing our analysis on these scores, we find the statistics to have a non-monotone dependency on the number of edges and nodes of the networks being considered. As suggested in Przulj [2007], we use the GDDA score to compare PPI networks to three random graph models: Erdős-Rényi random graphs, Erdős-Rényi (ER) random graphs with fixed degree distribution

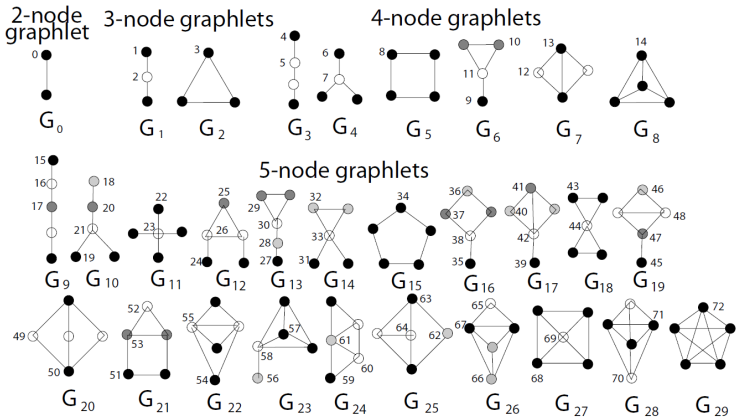


Figure 2.1: Subgraphs or graphlets with 2-5 nodes $G_0, G_1, ..., G_{29}$. The automorphism orbits are numbered 0 to 72 and the nodes belonging to the same orbit are of the same shade within a subgraph. From Przulj [2007].

(ER-DD), and geometric 3-dimensional random graphs (GEO3D). Observing that the empirical distribution of the GDDA score under these theoretical models is far from normal, we provide non-parametric test procedures to assess the model fit. We find that none of these random network models fit to the PPI networks. While we conclude that we are still far from having a satisfactory null model for PPI networks, we provide a statistical framework for assessing the fit of potential new models under the aspect of similarity of small subgraph counts. The proposed method relies only on the assumption that if a PPI network is generated by a given model, then the empirical distributions of the GDDA comparisons between the PPI network *versus* model, and between model *versus* model, will be similar. Hence, any future model proposed for PPI networks can also be tested using this method.

Strikingly, the GDDA score is not stable in the graph density region of the biological networks considered. We hypothesise that this instability arises because the observed graph densities fall in the threshold regions for the appearance of small subgraphs, under both Erdős-Rényi and geometric 3-dimensional random graph models. In this region there is high volatility in subgraph counts even for two networks which are generated

under the same model and with the same specifications. While neither of these models fit the data, we can still use their threshold regions as proxy and conjecture that the PPI networks under consideration operate near the threshold for the appearance of small subgraphs. Such behaviour would imply relatively short paths between proteins in networks, with presumably just enough alternative paths to ensure robustness, while maintaining a low edge density for efficiency. This behaviour may also have further implications in the optimal design of networks.

2.2 The RGF and GDDA scores

The RGF-distance identifies all 3-5 nodes connected induced subgraphs, also called *graphlets*, in two networks and compares the frequency of their appearance. The relative graphlet frequency is given by $N_i(G)/T(G)$, where N_i is the graphlet counts for graphlet $i \in \{1, ..., 29\}$ (see Figure 2.1) and $T(G) = \sum_{i=1}^{29} N_i(G)$ [Przulj et al., 2004]. The RGF-distance between two graphs G and H is then defined as

$$\text{RGF-distance} = \sum_{i=1}^{29} |F_i(G) - F_i(H)| ,$$

where $F_i(G) = -log(N_i(G)/T(G))$.

The Graphlet Degree Distribution Agreement (GDDA) [Przulj, 2007] is based on *orbit degree distributions*, which are based on the automorphism orbits of the 29 graphlets on 2-5 vertices, as follows. Automorphisms are edge-preserving bijections from a graph to itself, and together they form a permutation group. An *automorphism orbit* is a node that represents this group. Within the 29 graphlets, 73 different orbits can be found (see Figure 2.1) and each one will have an associated orbit degree distribution.

An orbit i from graphlet G_j has *orbit degree* k in the graph G if there are k copies of G_j in G which involve orbit i. In [Przulj, 2007] the term *graphlet degree distribution* is used instead of *orbit degree distribution*, but as orbits are counted, in our view the

latter term is more appropriate. For example, considering a simple 2-star graph as our main graph G (graphlet G_1 in Figure 2.1), we would have an orbit degree distribution for orbit 0 (an edge) of 2 node counts for orbit degree 1 (the outer two nodes) and one count for an orbit degree 2 (the middle node); the orbit degree distribution of orbit 1 would be two counts for an orbit degree 1, and for orbit 2 we would have one count for an orbit degree 1.

Let $d_G^j(k)$ be the sample distribution of the node counts for a given orbit degree k in a graph G and for a particular automorphism orbit j. In our previous example, where $G = G_1$, we obtain $d_{G_1}^0 = (2, 1, 0, ..., 0)$; $d_{G_1}^1 = (2, 0, 0, ..., 0)$; $d_{G_1}^2 = (1, 0, 0, ..., 0)$ and $d_{G_1}^i = 0, \quad \text{for} \quad i = 3, \ldots, 72$. This sample distribution is then scaled by $1/k$ in order that large degrees do not dominate the score, and normalised to give a total sum of 1,

$$N_G^j(k) = \frac{d_G^j(k)/k}{\sum_{\ell=1}^{\infty} d_G^j(\ell)/\ell}.$$

The comparison $D^j(G, H)$ of two graphs G and H with respect to j is simply the Euclidean distance between the two scaled and normalised vectors N, which is scaled by $1/\sqrt{2}$ to be between 0 and 1, as pointed out in Przulj [2010]; the resulting expression is

$$D^j(G, H) = \frac{1}{\sqrt{2}} \left(\sum_{k=1}^{\infty} [N_G^j(k) - N_H^j(k)]^2 \right)^{\frac{1}{2}}.$$

This distance is then turned into an agreement by subtracting from 1, and the agreements are combined into a single value by taking the arithmetic mean over all j, yielding the GDDA,

$$\text{GDDA} = \frac{1}{73} \sum_{j=0}^{72} (1 - D^j(G, H)).$$

GraphCrunch, a software developed by N. Pržulj and co-workers also calculates a variant of GDDA using the geometric mean (see Appendix A.2).

The random graphs used in the experiments of this chapter were generated using the internal generators of GraphCrunch. GraphCrunch [Milenkovic et al., 2008] is an open-source software tool that compares large real-world networks with random graph models. These are automatically generated to have the same number of nodes and edges (to within 1%) as those of the real-world network being compared. This has to be taken as approximate; with a simple 12-star as input, ER-DD graphs with 10, 11 and 12 edges are generated. As well as many global standard properties, the software supports the local statistics RGF-distance and GDDA. The software has been used for a wide range of applications among which assess parametric models for PPI networks [Przulj, 2007], protein structure networks [Milenkovic et al., 2009] and brain functional networks [Kuchaiev et al., 2009].

A typical network comparison output based on GDDA generated by GraphCrunch is shown in Figure 2.2. Six PPI networks were considered; 2 yeast and 4 human. Table 2.1 summarises some characteristics of these datasets. Multiple edges between the same nodes were collapsed into one and self-loops were excluded from all network data. BioGRID interaction data for human (release 2.0.55, www.thebiogrid.org) was filtered using the key-words "Affinity Capture-MS" and "Two-hybrid" and divided into two distinct data sets: BG_MS and BG_Y2H respectively. The query networks were compared with 100 random graphs of each model - ER, ER-DD and GEO - which were automatically generated by GraphCrunch.

Table 2.1: Protein-protein interaction networks analysed in this chapter.

Label	#nodes	#edges	Experiment type	Organism	Reference
YIC [1]	796	841	Yeast two-hybrid	*Saccharomyces cerevisiae*	Ito et al. [2000]
YHC [1]	988	2455	TAP MS	*Saccharomyces cerevisiae*	von Mering et al. [2002]
HS [1]	1705	3186	Yeast two-hybrid	*Homo sapiens*	Stelzl et al. [2005]
HG [1]	3134	6725	Yeast two-hybrid	*Homo sapiens*	Rual et al. [2005]
BG_MS	1923	3866	Affinity Capture-MS	*Homo sapiens*	BioGRID 2.0.55 (filtered)
BG_Y2H	5057	9442	Yeast two-hybrid	*Homo sapiens*	BioGRID 2.0.55 (filtered)

[1]These data sets were also considered by Przulj [2007].

2.3 Empirical distributions of GDDA

The standard GraphCrunch output of Figure 2.2 compares PPI networks with random model networks using GDDA. The plot shows the highest GDDA for the GEO3D random graph model type for all the networks, followed by ER-DD and ER models. While Przulj [2007] would now conclude that GEO3D is the best-fitting model for PPI networks, we shall see that due to the threshold behaviour of the networks such conclusion is not statistically justifiable.

According to Przulj [2007], a perfect score can be achieved when comparing networks of the same random model type. Przulj [2007] found the mean GDDA of comparing ER *versus* ER, ER-DD *versus* ER-DD or GEO-3D *versus* GEO-3D to be 0.84 ± 0.07, where 0.07 denotes one standard error. This was updated in Przulj [2010] where they found the highest score for two GEO-3D networks to be 0.95 ± 0.002.

To address how to interpret the output from a graph comparison based on GDDA, first for both the ER model and the GEO3D model, graphs of 500, 1000 and 2000 vertices with increasing graph density were generated using the internal generators from GraphCrunch. The graphs were subsequently used as query networks in the software and compared with 50 networks of the same model to ascertain typical GDDA scores if the model is correct.

The results for comparing ER *versus* ER and GEO3D *versus* GEO3D networks with 500, 1000 and 2000 nodes across a wide range of graph densities are summarised

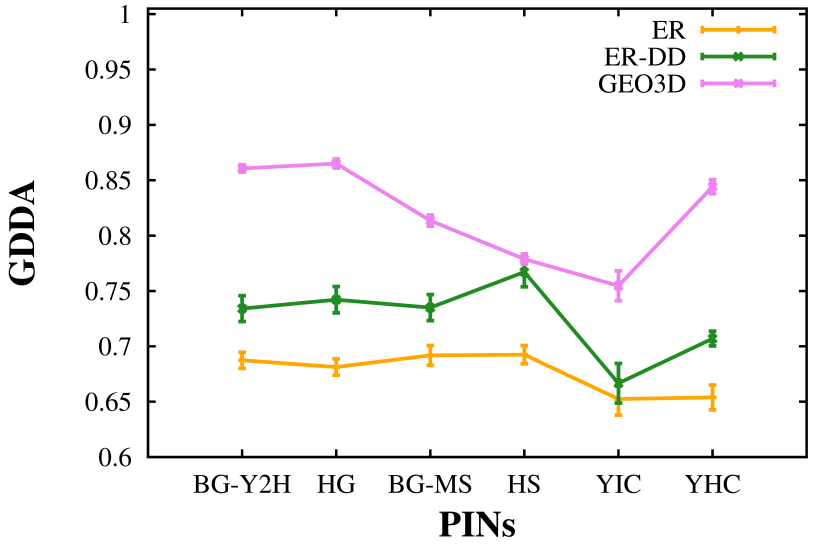


Figure 2.2: Graphlet Degree Distribution Agreement (GDDA) plot between 6 PPI networks and their corresponding random model networks. All the points in the plot are averages of comparisons between the query network and each of 100 generated model networks. The error bars represent 1 estimated standard deviation below and above the average point.

in Figure 2.3 using GDDA and in Figure 2.4 using RGF. Similar results for GDDA with geometric mean can be found in Appendix A.1.

In contrast to Przulj [2007], we find that the RGF and GDDA values have not only striking differences amongst different model types, but also a pronounced dependency on the number of vertices of the network. For a specific graph, drawn from one model type and with a fixed number of vertices, we also observe a strong dependency of the average GDDA score with graph density when comparing to graphs of the same type and with the same number of vertices. Furthermore, these dependencies are not monotone. Also, although the RGF score does not rely on automorphism orbits like the GDDA-based scores, we still see a very similar dependency in both cases (Figure 2.4). Note that this score is not an agreement, but a distance.

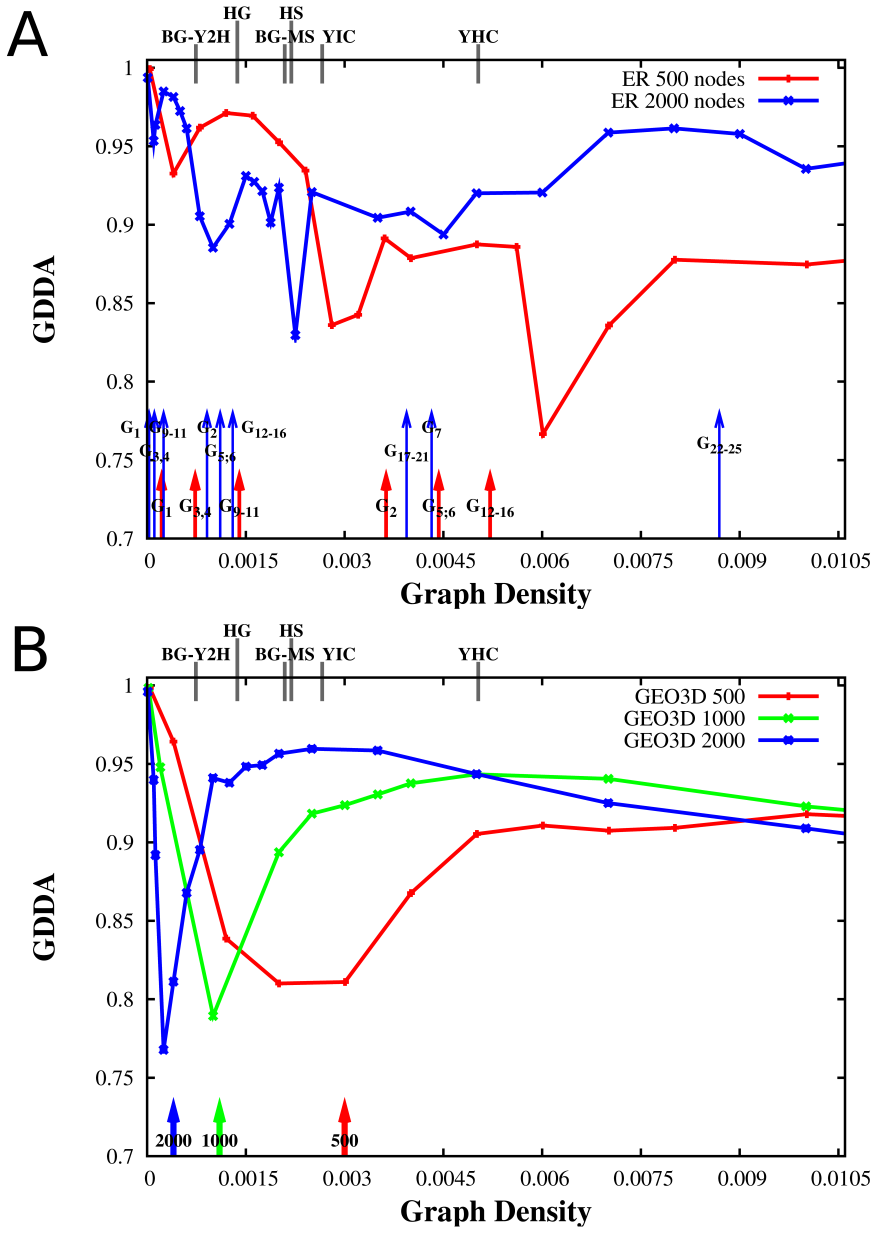


Figure 2.3: Dependency of GDDA for model *versus* model comparisons on the number of vertices and edges of a network. GDDA of ER *versus* ER *(A)* and GEO3D *versus* GEO3D *(B)* graphs with 500, 1000 and 2000 vertices are plotted against graph density. Each value represents the average agreement of 50 networks. The graph densities of the PPI networks considered (see Table 2.1) are indicated on the top x axis. In (A), the graph density values where the expected number of occurrences of a specific graphlet is approximately equal to one, for an ER graph with 500 and 2000 nodes respectively are indicated by the short and long arrows along the x axis. In (B), the thresholds for the appearance of 3-node graphlets are indicated for the GEO3D graphs with 500, 1000 and 2000 nodes.

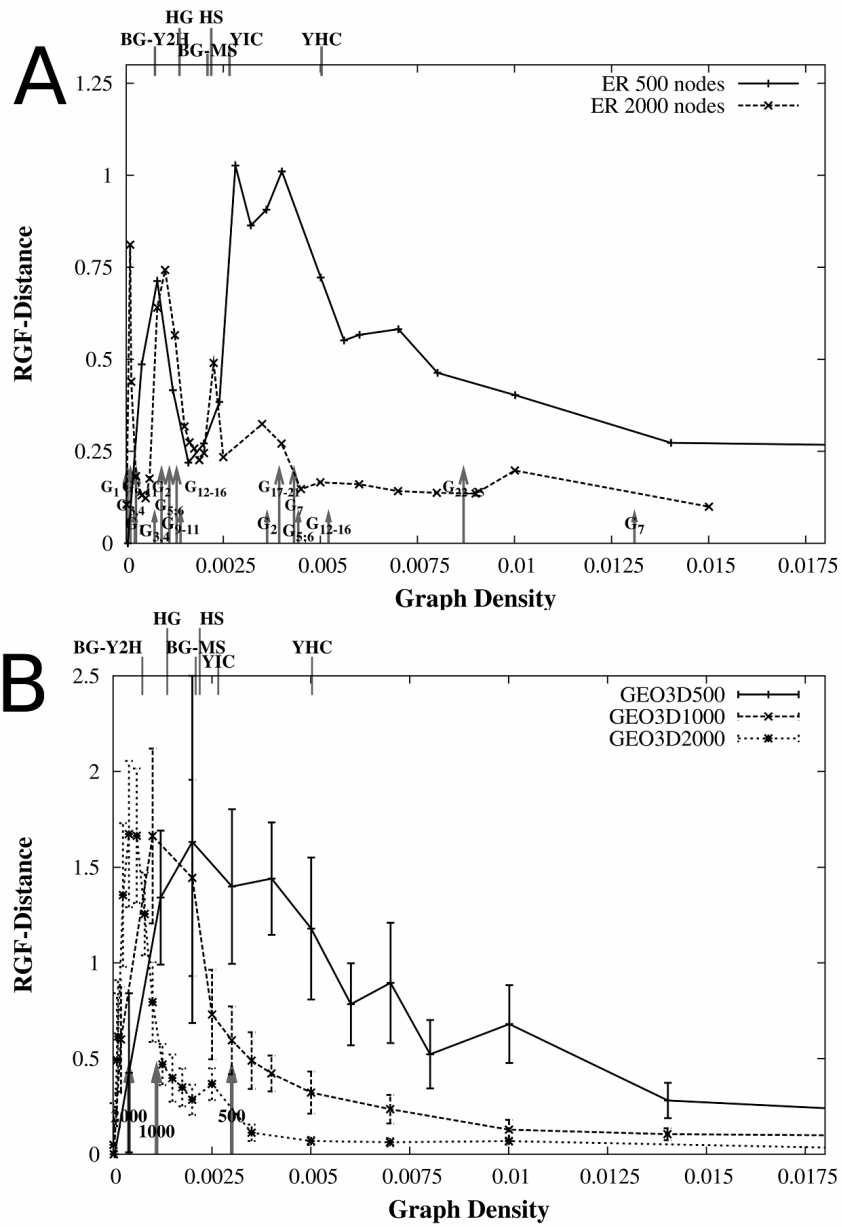


Figure 2.4: RGF-distance dependence on the number of vertices and edges of a network in model *versus* model comparisons. Average agreements of ER *versus* ER *(A)* and GEO-3D *versus* GEO-3D *(B)* graphs with 500, 1000 and 2000 vertices are plotted against graph density. Each value represents the average agreement of 50 networks. The graph density of the PPI networks considered (see Table 2.1) is indicated in the top x axis. In (A), the thresholds for the appearance of graphlets for an ER graph with 500 and 1000 nodes are pointed out along the x axis. In (B), the thresholds for the appearance of 3-node graphlets are indicated for the GEO3D graphs with 500, 1000 and 2000 nodes; although error bars are not statistically informative, they were included to give a sense of the variability present in the GDDA values considered.

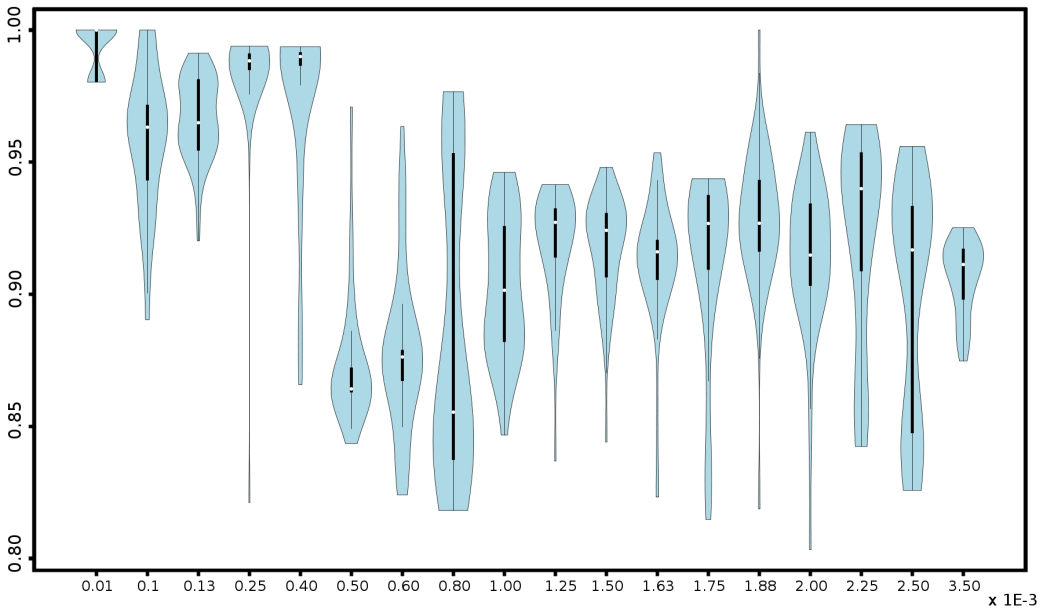


Figure 2.5: Violin plots with the distribution of GDDA values when comparing ER *versus* ER graphs with 2000 vertices against graph density. The white dots indicate the average score of each distribution.

For easier readability and because a normal approximation does not hold, we omit the error bars from the plots of Figure 2.3 and Figure 2.4A. To gauge these specific distributions we show in Figure 2.5 another run of ER *versus* ER in a violin plot. The "violins" represent smoothed distributions of GDDA scores of comparing 2000 nodes ER graphs with themselves for a zoomed region of graph density (0 to 0.00350).

In Figure 2.3A, for ER *versus* ER comparisons, in the region of graph density 0-0.01 we observe high volatility in the GDDA score. For higher graph densities the score appears to become more stable (see Appendix A.2). This volatility may be related to the natural appearance of small subgraphs, which is itself dependent on the number of nodes. In Table 2.2 we show the threshold functions for the property of containing one specific graphlet which were defined and calculated for ER networks with 500, 1000 and 2000 vertices (see Chapter 1).

The threshold values of the different 3-5 node graphlets for an ER graph with 500

Table 2.2: Graph density values for expecting approximately one copy of the graphlets $G_1, \ldots, G_{29}$ in ER networks with 500, 1000 and 2000 vertices.

Graphlets	500	1000	2000
G_1	0.00022	0.00008	0.00003
G_2	0.00363	0.00182	0.00091
G_3 - G_4	0.00073	0.00029	0.00011
G_5 - G_6	0.00443	0.00221	0.00111
G_7	0.01309	0.00752	0.00432
G_8	0.02696	0.01698	0.01070
G_9 - G_{11}	0.00140	0.00059	0.00025
G_{12} - G_{16}	0.00521	0.00261	0.00130
G_{17} - G_{21}	0.01251	0.00702	0.00394
G_{22} - G_{25}	0.02340	0.01426	0.00869
G_{26} - G_{27}	0.03741	0.02426	0.01573
G_{28}	0.05390	0.03667	0.02495
G_{29}	0.07218	0.05104	0.03609

and 2000 vertices are indicated in Figure 2.3A. For all graphs tested, the instability region in the GDDA score includes most of these thresholds.

For GEO3D *versus* GEO3D comparisons, one sees an instability in the score for small graph density which, after recovery, seems to slowly decrease again. Comparisons of GEO3D with 500 vertices for higher graph densities (up to 0.4) suggest that the score becomes more stable, although slowly increasing (see Appendix A.2). The asymptotic results also appear to be related to the score instability (Table 2.3; Figure 2.3B).

Table 2.3: Approximate graph density threshold values for the appearance of k-vertices graphlets in GEO3D networks with 500, 1000 and 2000 vertices.

Graphlets	500	1000	2000
3-vertices	0.0030	0.0011	0.0004
4-vertices	0.0085	0.0033	0.0013
5-vertices	0.0142	0.0060	0.0025

The most dramatic change in the score occurs when 3-node subgraphs start to appear; the appearance of 4- and 5-node subgraphs seems to have a much lower influence on the score. Strikingly, all the PPI networks under consideration are in the region of graph density populated by thresholds in both ER and GEO3D models. This invites the conjecture that PPI networks operate near the threshold for appearance of small

subgraphs. Unfortunately, no good model yet exists of PPI networks and so further work will be needed to confirm this conjecture.

It is worth noting that the specific GDDA values presented in Figure 2.3 may vary precisely because the specific graphs being generated for a particular comparison can be very diverse, especially in the region of high volatility (graph density between 0-0.01).

The instability of GDDA scores makes it difficult to interpret the output presented in Figure 2.2, not just because the typical score is different for each model type, but also because it is a function of the number of vertices and edges of the specific network being analysed. We find that the empirical distribution of GDDA in the region of interest, even in model *versus* model comparison, is not close to normal, indeed not even unimodal. This finding again can be explained by the network parameters being close to thresholds for the appearance of small subgraphs. Thus this threshold behaviour seriously affects the statistical inference from subgraph counts for network comparison and the conclusions which can be drawn from such subgraph count comparison.

2.4 Assessing the statistical relevance of the fit

Here we propose a new protocol for assessing model fit based on GDDA. Several same model *versus* model comparisons with roughly the same number of vertices and edges should be carried out in order to assess the best obtainable score for this specific case. GDDA should then be calculated between the query network and graphs from the model network. Model fit can be evaluated by gauging the differences between the distributions of agreement scores resulting from query network *versus* model and model *versus* model comparisons. We suggest the Monte Carlo non-parametric test for assessing whether the two independent samples of GDDA scores, one resulting from comparisons between query network *versus* model and the other from model *versus* model, come from the same distribution. Alternatively, the Wilcoxon rank-sum test

can be employed (see Appendix A.3).

More formally, the method can be described as follows. Given an input graph with n vertices and e edges, and a random graph model 1,

(1) Generate M graphs, say $M = 99$, from model 1 with about n vertices and e edges.

(2) For each one of these, carry out comparisons with N graphs generated from the same model and record GDDA; call the result *Sample A*. Here we use $N = 99$.

(3) Calculate the GDDA between the input graph and the N graphs from model 1, call the result *Sample B*.

(4) A histogram of *Sample A versus Sample B* may already show a clear separation of the two samples, making it obvious that the suggested model 1 is not a good fit, see Figure 2.6 for an illustration.

(5) For a statistical test, which tests for the null hypothesis that the two samples come from the same distributions against the general alternative that the distributions of the two samples are not the same, we employ a Monte Carlo test (see Appendix A.3 for details). Here, *Sample A* records M averages of the N comparisons, whereas *Sample B* consists of one observation: the average GDDA over the N comparisons of the input network *versus* model. The lowest obtainable p-value is then $1/(M + 1)$.

(6) We also employ a Wilcoxon rank-sum test, which tests for the alternative that the distribution of *Sample A* is a shifted version of the distribution of *Sample B* (see Appendix A.3). This test is more powerful than the Monte Carlo test, but tests against a less general alternative.

Figures 2.6A and B show histograms of GDDA values for comparisons between the PPI network BG-MS (see Table 2.1) *versus* 99 GEO3D and 99 ER-DD model networks

respectively. Both models have a virtually zero Wilcoxon p-value (there is no overlap between the distributions). A Monte-Carlo test was performed with 999 values, each an average of 30 model *versus* model agreements (M=999, N=30). In both cases a p-value of 0.001 was obtained, which is the smallest possible p-value for this test with 1000 observations. Although the mean of the empirical distribution is closer to ER-DD than to GEO-3D, the means are too far away to draw any useful conclusions. The large distances instead point to both models being inadequate and incommensurable to the network under consideration. The same results were obtained for the other 5 PPI networks of Table 2.1 (see Appendix A.3 for the p-values and histograms). The STICKY model [Pržulj and Higham, 2006] was also tested with similar results (see Appendix A.3). Hence, we conclude that none of the tested models fit the data.

To verify that our method is indeed capable of classifying a network, we took a GEO3D graph as input and compared it with other GEO3D networks, Figure 2.6C. The distribution overlap is clear and the Monte Carlo test gives a p-value of 0.24 (M=99, N=99). Figure 2.6C also illustrates the possible bias that can occur when just one model graph is used in same model *versus* model comparison. We emphasise that the graphs used for Figure 2.6C had the same number of vertices and graph density as BG-MS, and hence they are also in the threshold region, which may account for the relatively low p-value. We also report the GDDA values when one compares an ER-DD query network with ER model networks to show how the method behaves for two closely related models, see Figure 2.6D. A large overlap between the GDDA values is observed. The p-value for the Monte Carlo test with 100 values (M=99, N=99) is 0.15; hence for a single graph from the ER-DD model, our method cannot reject at the 10 % level the (reasonable) null hypothesis that the graph comes from an ER model.

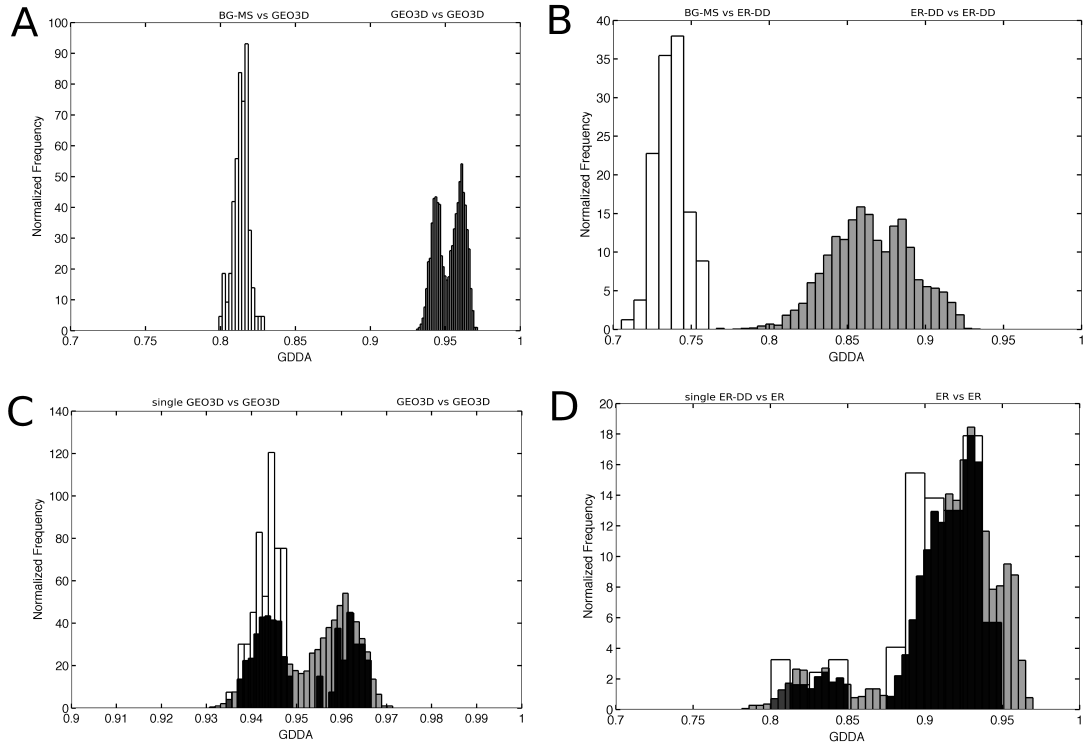


Figure 2.6: Normalised histograms of GDDA values. (*A*) Histograms of GDDA values between PPI network BG-MS *versus* 99 GEO3D model networks *(white)* and GDDA of 30 GEO3D, each *versus* 99 GEO3D *(grey)* (*B*) Histograms of GDDA values between PPI network BG-MS *versus* 99 ER-DD model networks *(white)* and GDDA of 30 ER-DD, each *versus* 99 ER-DD *(grey)* (*C*) Histograms of GDDA values between a single GEO3D graph *versus* 99 GEO3D *(white)* and GDDA of 30 GEO3D, each *versus* 99 GEO3D *(grey, the overlap is shown in black.)* (*D*) Histograms of GDDA values between a single ER-DD graph *versus* 99 ER *(white)* and GDDA of 30 ER, each *versus* 99 ER *(grey, the overlap is shown in black.)* All networks have approximately the same number of vertices and edges as BG-MS, with graph density of 0.00209. All images were generated by Matlab.

2.5 Principal component analysis as a method of network comparison

Judging by the results of the method presented in the previous section, with respect to subgraph counts, all tested models behave quite poorly. A plot with counts of triangles *versus* the counts of squares suffices to separate the models, see Figure 2.7.

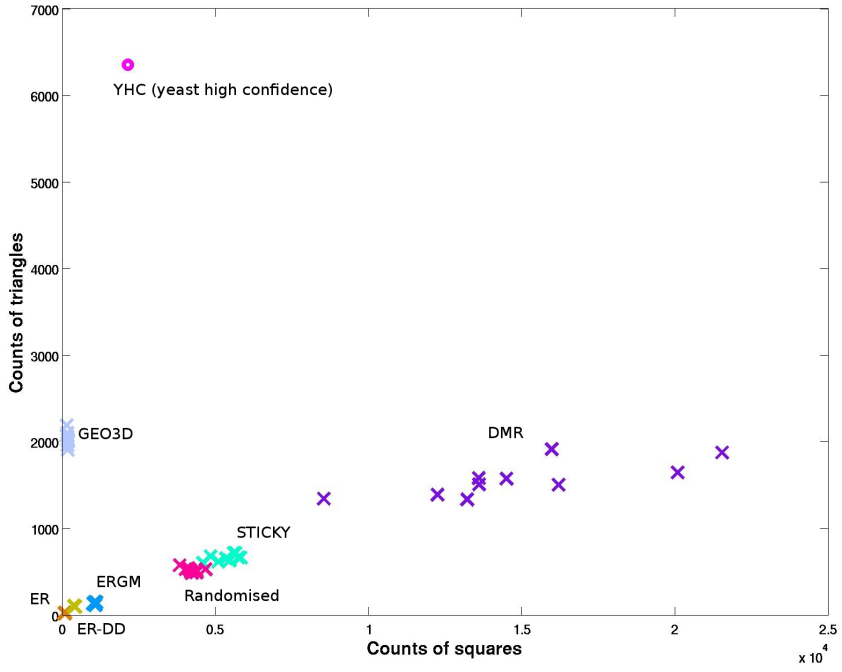


Figure 2.7: Counts of triangles *versus* counts of squares for several models (indicated in the plot) parameterised for the yeast high-confidence PPI network (YHC) [von Mering et al., 2002].

Besides the ER, ER-DD, GEO3D and stickiness index (STICKY) models employed in the previous sections, we also test exponential random graphs (model 6 of Table B.2 in Appendix B), a gene duplication model (DMR) [Middendorf et al., 2005] and the randomisation algorithm by [Maslov and Sneppen, 2002] (see Chapter 1). The parameterisation of ER, GEO3D and DMR is solely done using the number of nodes

and edges of YHC (see Table 2.1). DMR was generated as in Middendorf et al. [2005] with the parameters $q_{mod} = 0.62$ and $q_{con} = 0.1$, using a 50-node seed as described by Hormozdiari et al. [2007]. We simulate 10 networks of each network model to represent it in our analyses. Figure 2.7 shows that the models considered are incapable of reproducing both the number of squares and triangles found in YHC. All triangle counts are greatly under-estimated and the number of squares are either above or below the number of occurrences for YHC.

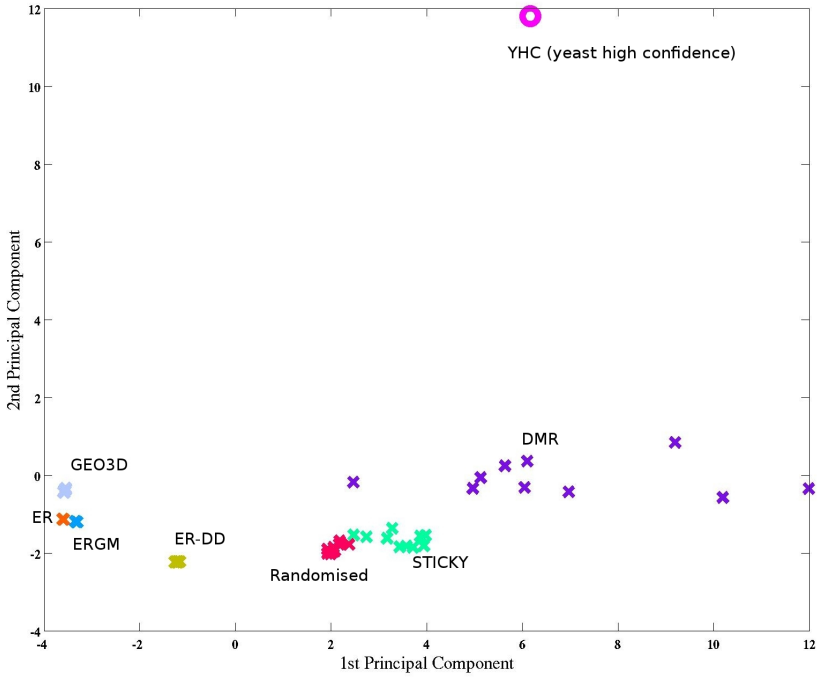


Figure 2.8: Projection of the first two components of principal component analysis on the normalised subgraph counts of 2-5 node subgraphs for different models parameterised for the yeast PPI network YHC.

To understand where the models are failing we employed principal component analysis (PCA) performed on a matrix with the standardised counts of all 2 to 5-node induced subgraphs. Each column of the matrix corresponds to a network of a specific

model and a column for YHC is added at the end. The raw subgraph counts are divided by the standard deviation of the counts for that particular subgraph across different models (one standard deviation value per row). Figure 2.8 depicts a 2D plot of the first two principal components; we can see that the models are quite separated from the YHC PPI network. The first component explains 49% of the data and the second 36%. In this case, the top 5 coefficients in the first component correspond to the subgraph shapes seen in Figure 2.9.

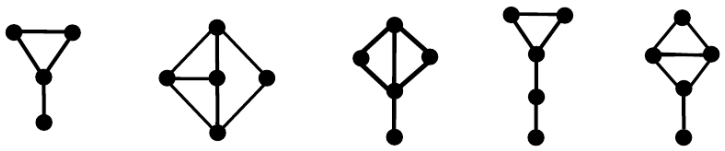


Figure 2.9: Subgraphs corresponding to the Top 5 coefficients of the first principal component in the PCA of Figure 2.8. This component explains 49% of the data.

A similar analysis can be performed for any other PPI network. The overall results indicate that shapes containing triangles frequently feature in the top coefficients of the first and second principal component. In the case of YHC, although triangles are not in the top 5 coefficients, all the top shapes contain at least one triangle. When we include different PPI networks in the same PCA plot we observe that PPI networks can vary more amongst themselves than a given PPI network from its respective models, implying that these analyses are dataset specific. This also illustrates the different level of completeness and noise between the various PPI networks. A model which reproduces the typically high clustering coefficients of these networks is likely to contain a higher number of these shapes. We note that these shapes seem to be best at distinguishing between models and PPI networks. This observation supported the study of the nature of triangles in PPI networks presented in the next chapter.

2.6 Threshold behaviour

In our analysis we have found that none of the theoretical models considered are suitable models for the PPI networks analysed. However, we provide a statistical framework for comparing real-world networks to other theoretical models using non-parametric statistics.

Our results on GDDA scores suggest that PPI networks are situated in a region of graph density close to the threshold behaviour of the models analysed. *Saccharomyces cerevisiae* has ∼6,600 protein coding genes (www.yeastgenome.org) and is predicted to have about 25,000-35,000 interactions [Stumpf et al., 2008]; such a network would have a graph density between 0.0011 and 0.0016. For *Homo sapiens*, estimates of about 25,000 genes (Human Genome Project) and 650,000 PPI [Stumpf et al., 2008] would also lead to graph densities around 0.002. Both these networks would be placed in the threshold region for the appearance of G_8 as well as $G_{17} - G_{27}$ under the ER model. This may suggest that globally many pathways between proteins are essentially unique, with just a few alternative routes; cliques of size 4 and most graphlets on 5 vertices are unlikely to appear. Such an architecture would render the network both efficient (not too many edges) and robust (alternative pathways are available).

The use of subgraphs for network comparison has nonetheless, many open questions such as the use of induced *versus* non-induced subgraphs [Alon et al., 2008] and their biological significance, and the inevitable over-counting problem that occurs when one limits the size of the subgraphs considered.

2.7 Conclusions

We have shown that typical values of GDDA, gauged by same model comparison, depend on the number of edges and nodes of the underlying graph.

We propose a statistical method for assessing model fit based on GDDA. Although none of the suggested models fit any of the data sets, we provide the basis for statistical comparison with other models. Principal component analysis of standardised subgraph counts also puts models very far from PPI networks and this appear to be related with shapes containing triangles. Note that testing for network similarity is phrased here in terms of small subgraphs. If the interest was rather on, say shortest distances in PPI networks, then the conclusions could be very different.

Curiously, the GDDA score is particularly unstable in the graph density region between 0 and 0.01, which encompasses most of the PPI networks currently available. We provide the plausible explanation that this is due to thresholds for the appearance of small subgraphs.

Using these thresholds in ER and GEO3D models as proxy, we suggest that PPI networks themselves tend to operate near the thresholds for the appearance of small subgraphs. That is, the network will start to have a few alternative paths between proteins, but not many. This observation may lead to further conjectures about optimal design of networks, accounting for these critical regimes.

Chapter 3

Protein age and degree: the node and the edge models

In the last chapter I presented evidence that current network models cannot accurately reproduce the counts of subgraphs found in PPI networks. Here I look at protein age as an explanatory variable for those protein interactions finding an over-representation of pairwise and triangle interactions between "Old" proteins. The results point towards an architecture of PPI networks that is highly heterogeneous, having connected clumps which contain a large number of interacting Old proteins along with selective age-dependent interaction patterns. The majority of the results presented in this chapter have been published in Rito et al. [2012].

3.1 Protein age and PPI networks

Several high-throughput studies in *Saccharomyces cerevisiae* (yeast) contribute to make its interaction data the most extensive and, to date, the most complete eukaryotic PIN existent [Jensen and Bork, 2008; Sambourg and Thierry-Mieg, 2010]. Thus, in this chapter we focus on yeast. To increase our understanding of the network topology it is reasonable to include explanatory variables for interactions. Guided by the principle

that cell biology should be viewed in the light of evolution, we concentrate on one evolutionary explanatory variable where previous correlations with interactions have been found - protein "age". The age of a protein is an abstract theoretical concept of little importance for the organism and there is no single, optimal way of estimating the "age" of a protein. Here, as a proxy for age, we use lineage specificity [Winstanley et al., 2005] based on InParanoid's [Östlund et al., 2010] ortholog identification. Lineage specificity values range between 0 and 1, where 0 is assigned to a protein which is only present in yeast, or in yeast and a few other highly related organisms, and 1 corresponds to a protein whose appearance can be traced back to the most ancient branching of the tree under consideration. In view of the distribution of lineage specificity we declare proteins with a lineage specificity of at least 0.8 as "Old", those with a lineage specificity of at most 0.2 as "Young", and all others as "Middle-aged". Other age assignments are possible; the approach by Kim and Marcotte [2008], which we also consider, assigned Pfam domains to each protein in yeast and took its taxonomic distribution as the age groups. According to its youngest Pfam domain a protein can then be classified as ABE, AE/BE,E or F, where ABE is the oldest age group including proteins found in Archae, Bacteria and Eukaryota, and F the youngest, only including Fungi-specific proteins.

Here we provide an in-depth analysis of protein age patterns found in edges and triangles of the yeast PIN. We assess their statistical significance according to what would be expected by chance alone given the age frequencies found in the PIN and also, for the case of triangles, given the age frequencies observed in the currently available pairwise data. To our knowledge this second approach has not been applied previously, although it seems more natural as we have not only information on node frequencies, but also detailed information on the types of interactions we currently observe. When analysing the relationship between network models and age-dependent interaction patterns, Kim and Marcotte [2008] focused on interaction pairs and on the propensities of proteins in

similar age groups to interact with each other. They found that the interaction density is highly protein age-dependent. Liu et al. [2011] considered age patterns found in network motifs, but only reached the coarse-grained conclusion that proteins of the same age group tend to form motifs which are functionally homogeneous and densely interconnected. Qin et al. [2003] also looked at the interaction patterns within and between age categories using a randomisation protocol as a null model which shuffled age assignments to nodes while fixing the node frequencies [Maslov and Sneppen, 2002]; they find support for a PIN which conserves age-homogeneous clusters and whose topology depends on the evolution of the organism.

As there is a positive association between age and degree [Winstanley et al., 2005], we also investigate whether the patterns change when restricting attention to proteins with high degree, and equally when restricting attention to proteins with low degree.

Given the large amount of data, many of our tests indicate significant differences from what would be expected given the node frequencies, and from what would be expected given the edge frequencies; hence we focus on the most highly significant differences. These are, across data sets, firstly, that Old-Old edges and Old-Old-Old triangles are highly over-represented. Secondly, triangles with edges involving one Middle-aged and one Old protein are highly under-represented. Thirdly, 83% of all triangles consist of proteins with degree at least 10. We dispense with the notions that proteins like to interact with proteins of the same age, and that high-degree proteins are connected to many low-degree proteins in a star-shaped fashion. Instead our findings lead to viewing the PIN as highly heterogeneous, with dense regions containing a large number of Old proteins, and with complex selection patterns.

3.2 Relative age calculation

In this chapter we work with the protein interaction dataset for *Saccharomyces cerevisiae* obtained from the Database of Interacting Proteins (DIP) [Salwinski et al., 2004], version 2010-10-10, comprising 25,233 interactions between 5,213 proteins. These data were purged of self-interactions and translated to the SGD systematic name nomenclature. The small number of interactions with no name matches were discarded. The final DIP dataset consisted of 5,092 proteins and 24,693 interactions. A high-quality subset of DIP, called DIP_CORE, was also considered, as well as another subset of DIP where interactions discovered using tandem affinity purification followed by Mass Spectrometry were omitted (see Appendix C).

The (relative) age of a protein is here calculated by lineage specificity [Winstanley et al., 2005] (`http://www.stats.ox.ac.uk/~abeln/foldage/`) using the ortholog identification of InParanoid 7.0 [Östlund et al., 2010] in 99 eukaryotic species plus *Escherichia coli*. A parsimony age that considers the possibility of horizontal gene transfer by making parsimonious allocations of gain and loss events is calculated for each protein using the occurrence pattern and the respective tree of the 100 species as described in Winstanley et al. [2005]. Values of relative age range from 0 to 1, where 0 corresponds to a protein that is only present in yeast or in yeast and a few other highly related organisms on the same branch as yeast, and 1 to a protein whose appearance can be traced to the most ancient branching of the tree considered.

3.3 Distribution of protein age and categorisation

The relative age of 5,884 proteins in the proteome of yeast was calculated by lineage specificity using the normalised tree and ortholog identification of InParanoid. The distribution of protein ages is presented in Figure 3.1. Note that not all of these proteins will have interaction data associated with them.

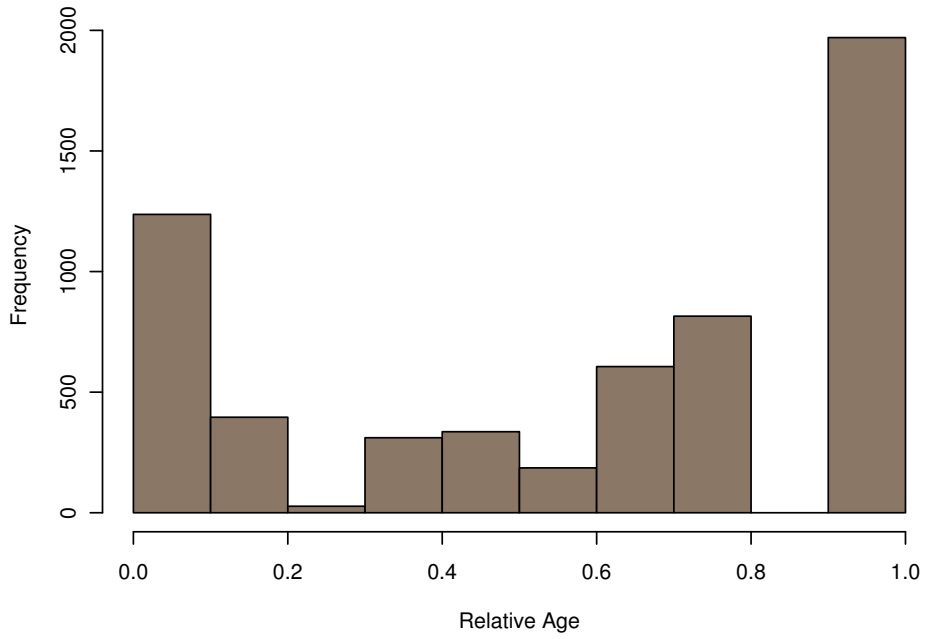


Figure 3.1: Distribution of the relative protein age of 5,884 proteins in the yeast proteome. Middle-aged proteins are defined as the ones with relative ages greater or equal to 0.2 and less than or equal to 0.8. Proteins above or below this cut-off are put in the Old or Young set, respectively.

Next we bin the ages in this distribution into three categories: "Old" proteins, which tend to have relative age values near 1 and are therefore represented in most genomes considered; "Young" proteins, with age values near zero which are specific to yeast or yeast and highly related species; and "Middle"-aged proteins, those which are neither "Old" or "Young". More specifically, proteins with relative age > 0.8 are called "Old", proteins with relative age < 0.2 are called "Young", and proteins with relative age between 0.2 and 0.8 inclusive are called "Middle-aged". While Figure 3.1 shows that the Old cut-off is natural, this is less clear in the Young case, thus we test variation in this boundary with a different Young cut-off of 0.4 and find that our conclusions are robust (see these results in Appendix C). Protein ages obtained from Kim

and Marcotte [2008] were also considered by collapsing the sets ABE, AE/BE into a set of Old proteins, the set E into Middle-aged proteins and Fu and N into a set of Young proteins (see Appendix C).

In DIP, the original 0.2 Young cut-off assignment yields 1,706 Old proteins, 1,917 Middle-aged proteins, and 1,146 Young proteins. Due to nomenclature issues there were 323 proteins in the PIN for which no age was available; these proteins and their interactions were discarded (565 of 24,693 interactions in DIP). The final DIP interaction data sets consists of 4,769 proteins and 24,128 interactions. The three protein age sets in the yeast DIP PIN possess very different connectivity patterns. Figure 3.2 depicts a box plot (outliers were not plotted) showing the variation in degree distribution for the categories; the degree of a protein is simply its number of interactions.

In agreement to what has previously been found, older proteins have, on average, a higher degree than younger ones [Wuchty et al., 2003]. It was also shown that high-degree proteins tend to be old [Winstanley et al., 2005]. Biologically, old genes have been loosely associated with large protein size, strong selection pressure as well as high intron density and expression level [Wolf et al., 2009].

3.4 Age patterns

After describing the age distribution across all proteins in the proteome of yeast, we now compute age-dependent edge and triangle frequencies and assess whether the observed patterns can be explained by protein age frequencies alone. Next we assess whether the observed triangle patterns can be explained by the relative frequencies of protein interactions alone. For this, we devised a software capable of calculating the absolute and relative frequencies of the nodes, edges and triangles for the possible combinations of different types of categories. To control for degree as a possible confounding factor

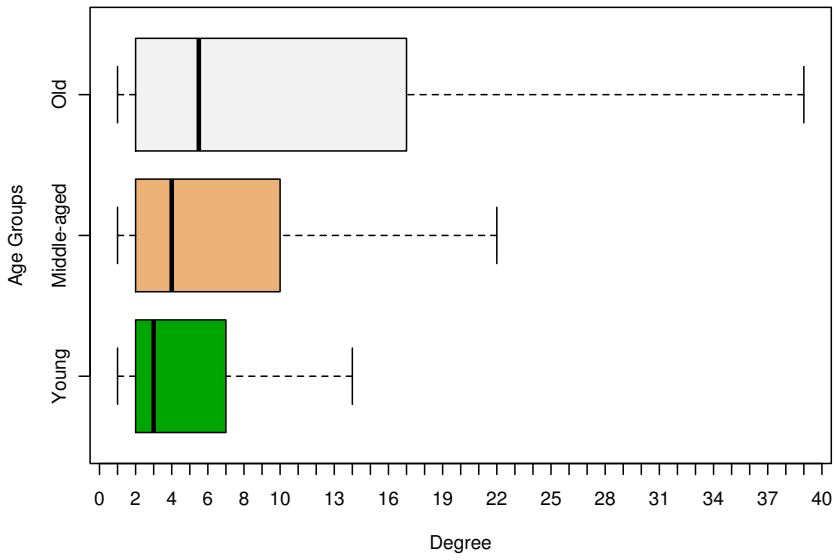


Figure 3.2: Degree distribution for the yeast proteins in each age category. For each age set the middle line represents the median of the data and the bottom and top of the box correspond to the 25th and 75th percentile respectively. The bottom and top whiskers represent the lowest/ highest datum within 1.5 times the interquartile range of the lower/ upper quartile, respectively. Outliers are not plotted. For the Old set of proteins, there are 123 proteins above degree 39 (top whisker), the highest degree of this set is 229. For the Middle-aged set, there are 184 proteins above degree 22, the maximum degree value is 281. For the Young set, there are 105 proteins above degree 14, of which the most extreme value is 98.

the software also allows for selecting nodes above or below a certain degree and we repeat the analysis considering only the interactions amongst high-degree proteins. We mainly present results for the DIP (Database of Interacting Proteins) dataset and for a single categorisation of protein age, although we also consider the high-confidence DIP-CORE PIN and different age categories; as our findings are fairly robust we only mention the latter data sets when their patterns deviate from the patterns in DIP. Table 3.1 summarises the number of proteins in each age group, for the different DIP-derived data sets considered, not taking edge categories into account. In the next section, we

present the null models based on node and edge frequencies which allow us to assess the significance of the observed patterns.

Table 3.1: Absolute frequencies of proteins categorised by age on each network and their number of nodes, edges and triangles.

	DIP	DIP_10	DIP_25	ANTI_10	ANTI_25
Old	1,706 (36%)	631 (48%)	273 (56%)	641 (32%)	1,190 (34%)
Middle-aged	1,917 (40%)	504 (38%)	164 (34%)	808 (40%)	1,425 (41%)
Young	1,146 (24%)	187 (14%)	47 (10%)	568 (28%)	886 (25%)
Number of proteins	4,769	1,322	484	2,017	3,501
Number of edges	24,128	14,053	5,793	1,700	6,328
Number of triangles	18,274	15,187	7,389	120	1,640

DIP_x denotes the DIP dataset only considering the proteins with a degree greater or equal to x and the links between them. ANTI_x denotes the complementary network with proteins with a degree of less than x. Here we consider $x = 10$ and $x = 25$.

3.4.1 Models for edges and triangles

Node frequency model

To assess whether the observed edge distributions and triangle distributions can be explained by the relative frequencies of proteins of different ages in the data set without making use of network information, we test null models based on nodal frequencies against general alternatives. Here we abbreviate the ages as O, M, Y and order them as $O > M > Y$; edges and triangles are annotated with the ages of the nodes involved.

Let (p_O, p_M, p_Y) denote the vector of the probability of seeing an Old, Middle-aged, or Young protein in the data set of interest. The null model for edges is that the probability p_{ij} of an edge (i, j) is just the product of the probabilities of the nodes;

$$p_{ij} = \begin{cases} p_i^2 & \text{if } i = j, \\ 2p_i p_j & \text{if } i < j, \end{cases}$$

such that $\sum_{i \in \{O,M,Y\}} \sum_{j \geq i} p_{ij} = 1$.

We estimate these probabilities by their maximum-likelihood estimator under the

null model, namely their relative frequencies (f_O, f_M, f_Y); thus the dimension of the parameter space under the null hypothesis is 2. Under the general alternative the probability of an edge is estimated by its relative frequency, $f_{i,j}$, for $i \leq j \in \{O, M, Y\}$, hence the dimension of the parameter space under the alternative is 5. We use a chi-square goodness-of-fit test with $5 - 2 = 3$ degrees of freedom to assess the evidence for the null hypothesis.

For the case of triangles, the null hypothesis is again that node probabilities alone can explain the numbers of triangles we see; the probability p_{ijk} for a triangle (i, j, k) with $i \leq j \leq k \in \{O, M, Y\}$ would be

$$p_{ijk} = \begin{cases} p_i^3 & \text{if } i = j = k, \\ 3p_i p_j^2 & \text{if } i < j = k, \\ 6p_i p_j p_k & \text{if } i < j < k, \end{cases}$$

such that $\sum_{i \in \{O,M,Y\}} \sum_{j \geq i} \sum_{k \geq j} p_{ijk} = 1$. We use a chi-square goodness-of-fit test with $9 - 2 = 7$ degrees of freedom to assess the evidence for the null hypothesis.

We report the observed and expected counts for each edge and triangle, assess their contribution to the chi-square statistic separately and report whether its contribution alone would have lead to a rejection of the null hypothesis; these p-values are presented in the Appendix C. To account for multiple testing we also take the Bonferroni correction.

Edge frequency model

To assess whether the observed triangle distributions can be explained by the relative frequencies of edges between proteins of different ages without making use of any further network information, we also test the null model based on edge frequencies against the general alternative. Under the null model the probability of a triangle is just the

product of the probabilities of its edges, conditioned on the event that the edges can form a triangle;

$$p_{ijk} = \frac{1}{p} \times \begin{cases} p_{ii}^3 & \text{if } i = j = k, \\ 3p_{ij}^2 p_{jj} & \text{if } i < j = k, \\ 6p_{ij}p_{jk}p_{ik} & \text{if } i < j < k, \end{cases}$$

where

$$p = \sum_{i \le j \le k} p_{ij}p_{jk}p_{ik}.$$

We use a chi-square test of goodness-of-fit with $9 - 5 = 4$ degrees of freedom. We also report the observed and expected counts for each triangle separately, and whether its contribution alone would have lead to a rejection of the null hypothesis; the p-values are included in the Appendix C.

3.4.2 Age patterns in pairwise interactions

Given the higher average degree of Old proteins (Figure 3.2), it is perhaps unsurprising to find that about 60% of all interactions happen between Old and Middle-aged or Old proteins. Figure 3.3 depicts the observed age-labelled pair frequency and, for comparison, the corresponding expected frequency under a model which assumes that the node frequencies can explain the edge frequencies; this is the node frequency model introduced in the previous section.

Despite the differences between observed and expected frequencies giving, for all cases, a significant p-value under a Chi-squared test, caution in the interpretation is advisable as with large amounts of data even small departures from the null hypothesis, with relatively no relevance given the overall pattern, may be highly significant [Cox, 2006, p.42]. We find that while the product of the nodal frequencies does not predict the interactions very well for most types of interactions, the most striking deviation is that Old proteins interact far more frequently with Old proteins than expected (so that

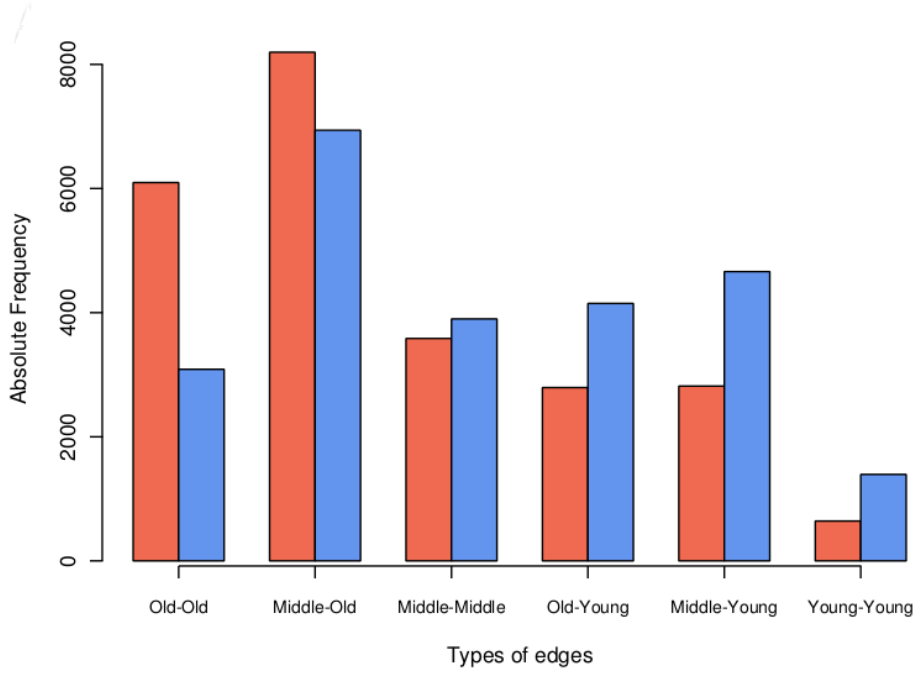


Figure 3.3: Age-dependent patterns in the pairs of the yeast PIN. Observed *(red)* and expected *(blue)* frequencies under the node model for the different types of age-dependent patterns found in edges of the PIN. All differences between observed and expected are significant under a chi-squared test. "O" indicates an Old protein, "M" a Middle-aged one and "Y" a Young protein. All differences are by themselves significant under a chi-squared test. All p-values are reported in the Appendix C.

most of the other types of interaction are just slightly under-represented). The runner-up of highly over-represented interactions in DIP is Middle-Old, but this results is not replicated in the high-confidence DIP-CORE dataset. Several previous studies have reported a preference for proteins of the same age to interact with one another [Liu et al., 2011; Wuchty et al., 2003]. If we collapse our data to consider only two types of interactions, those between proteins in the same age categories and those between proteins of different categories we replicate this result, see Appendix C. Our more detailed results reveal that the same-same interactions are driven solely by the over-representation of Old-Old interactions in the dataset; Young-Young and Middle-Middle interactions are

actually slightly under-represented. This finding is robust when removing interactions found by tandem-affinity purification coupled to mass spectrometry (see Appendix C). The results vary slightly when the age definitions by Kim and Marcotte [2008] are considered; then all of Old-Old, Middle-Old and Middle-Middle are over-represented, suggesting that in mapping between the ages there is overlap between our definition of Old and the Middle-aged and Old proteins under Kim and Marcotte [2008].

3.4.3 Age patterns in triangles

In view of the high clustering coefficients of PINs [Gibson and Goldberg, 2011] and the importance of triangles in interaction and function prediction [Chen et al., 2008], we now consider the age patterns on triangles. We identified the 18,274 triangles present in the complete DIP dataset. These were split according to the combination of the different age-categories of their constituent proteins. Figure 3.4 shows the observed absolute frequencies of all types of triangles formed from our three-way age classification in blue. Triangles between Old proteins and triangles between a Middle-aged protein and two Old ones account for over 50% of the total number.

At the level of triangle subgraphs we observe a repeat of the patterns seen in interactions; compared to what would be expected under a node frequency model (red in Figure 3.4), Old-Old-Old triangles (like Old-Old interactions) are highly over-represented. To assess whether this over-representation of Old triangles could be due to the large number of Old-Old interactions, we compare the observed counts to the counts which would be expected under an edge frequency model. The results of this comparison are shown in purple in Figure 3.4. Taking the Bonferroni correction into account, for DIP, all the differences are significant under a Chi-squared test and rejected at the 0.5% level, with the exception of M-M-M triangles under the node frequency model. This points to a network that is far from forming its interactions at random. Old triangles are over-represented compared to node frequencies and, once again, the proposed

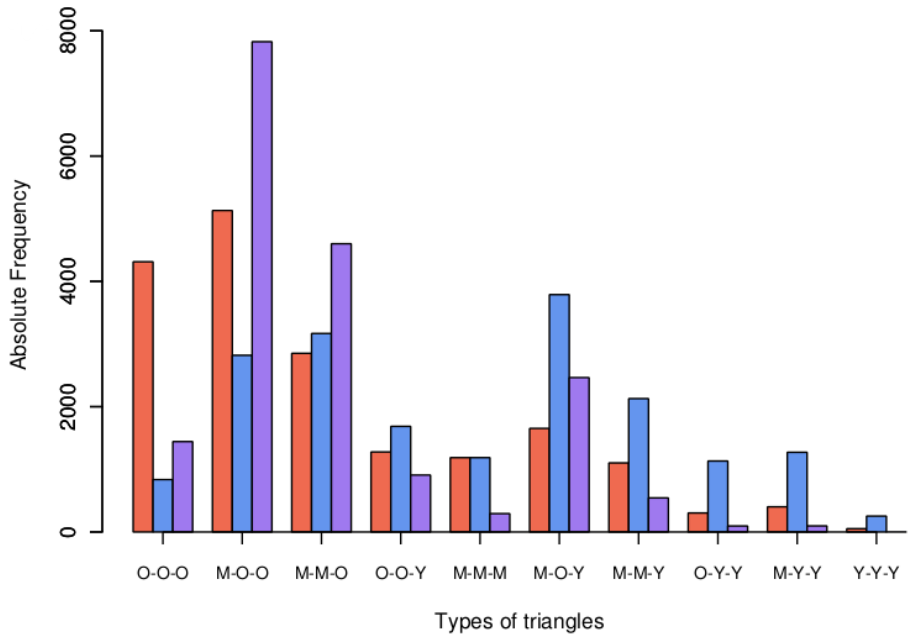


Figure 3.4: Age-dependent patterns in the triangles of the yeast PIN. Observed *(red)* and expected frequencies under the node *(blue)* and the edge *(purple)* models for the different types of age-dependent patterns found in triangles. "O" indicates an Old protein, "M" a Middle-aged one and "Y" a Young protein. All differences are by themselves significant under a chi-squared test, apart from M-M-M under the node model. All p-values are reported in the Appendix C.

homophily is driven by Old-Old-Old triangles. However, when we take edge frequencies into account, not only are all triangles involving proteins of the same age group over-represented, but so are all triangles involving a Young protein (apart from the Old-Middle-Young case). Given these patterns, homophily may be misleading. The pattern here is rather that all triangles which contain at least one Middle-Old edge are consistently under-represented, suggesting some form of negative selection for such triangles, especially for the Middle-Old-Old triangles. These results are also valid for DIP-CORE. Age-dependent triangle patterns based on the age definitions of Kim and Marcotte [2008] show deviations due to the age differences, but we still observe Old-

Old-Old over-representation, and Middle-Old-Old under-representation under the edge model. While evolution suggests many Middle-Old interactions in order to maintain relative stability of the network, evolution may nudge proteins to find new interaction partners among the same age group. Young proteins may not have had enough time to find new interaction partners.

3.5 The influence of high-degree proteins on age patterns

PINs have a broad degree distribution [Khanin and Wit, 2006] with a relatively small number of proteins having high degree. As high degree proteins tend to be older [Winstanley et al., 2005], high degree could provide an alternative explanation for the over-representation of patterns involving Old proteins. To control for high degree, for the DIP database, we generate two sets of proteins, those with degree greater or equal to 10 (DIP_10) and those with a degree greater or equal to 25 (DIP_25). For each set we rebuild the network, considering only the interactions between proteins within the set. We also select the complementary sets of proteins which we call Anti_10 and Anti_25 (see Table 3.1). Even with the partition of the dataset, Old-Old edges are over-represented given the node-frequencies in all of DIP_10, DIP_25, Anti_10 and Anti_25 (although in the last two cases Middle-Middle and Middle-Old are also over-represented, respectively). We conclude that high degree does not fully explain the over-representation of Old-Old.

Over 83% of all triangles are observed between high degree proteins (greater or equal to 10 interactions), suggesting that the current notion of hub proteins does not capture the network. Hubs are often thought of as high-degree proteins connected to proteins of far lower degree. Here we observe instead that high degree proteins are connected to proteins of similar high degree. This suggests that the PIN contains many dense clusters of high degree proteins connected via lower degree proteins.

Concerning age-dependent interactions, in high degree triangles we still observe the over-representation of Old-Old-Old triangles, but now it is only triangles with two Middle-Old edges that are under-represented under the edge model (this pattern is seen for both DIP_10 and DIP_25, notably for Middle-Old-Old). Even for triangles between low-degree proteins (Anti_10), Old-Old-Old triangles are over-represented (under both node and edge models) but now, alas, the only other pattern that is significant is the over-representation of triangles between proteins of the same age groups, under the edge model. These patterns are also seen for Anti_25.

3.6 Limitations and implications of our analysis for PINs

There are two caveats that come with our analysis, one related to the definition of age and the other with the effects of errors in the PIN.

Our age definition is a measure of lineage specificity dependent on sequence homology. Here we do not take the uncertainty in tree reconstruction into account; for statistical issues regarding tree reconstruction see for example Tavaré [2004]. Moreover, if a particular protein family diverges rapidly, we would not recognise its members across multiple species by sequence similarity, thus proteins with a Young age may contain new proteins alongside members of old protein families that are rapidly diverging. Conversely, the Old proteins set will tend to contain proteins from relatively stable, in sequence terms, protein families. It is hence possible that the behaviour of Old proteins described here is instead the behaviour of sequence-stable proteins.

The PIN itself is the other potential confounding factor. It has been extensively reported [Deane et al., 2002; Huang and Bader, 2009; Sprinzak et al., 2003] that PINs contain a large number of false positives and also a potentially even larger number of false negatives [Ali and Deane, 2010]. The node frequencies are calculated from the relatively error-free genomic data, whereas edge and triangle frequencies rely on the

PIN. Thus, the observed differences could be due to the error strewn PIN. While we have chosen to work with the yeast DIP data because the yeast PIN is thought to be the most complete currently available, even for yeast the dataset is far from complete [Jensen and Bork, 2008].

It is reasonable to ask how our findings relate to biological models such as that of gene duplication and divergence described in the introduction chapter. We consider a version of the duplication model [Bebek et al., 2006] designed to avoid the creation of singletons and proved to give heavy-tailed degree distributions. As described in Chapter 1, in this model, at each iteration t a node is chosen uniformly at random (parent) and duplicated by adding a new node to the graph (child) which retains all edges incident to the parent node. The divergence is then introduced at each iteration by (i) each edge in the child node is deleted independently with probability q; (ii) each node in the graph is independently connected to the child node with probability r/t. A final step consists in assuring that, if a singleton is produced, this node will be connected to at least one uniformly chosen random node. We choose the seed graph to be the sub-network of DIP comprising all 1,706 Old proteins and the 6,096 edges between them (singletons may be present). The model parametrisation is similar to Hormozdiari et al. [2007] with $q = 0.635$, $r = 0.33$. A total of 30 networks are grown to an order of 4,769 nodes, having an average size of 23,827 edges. The age labelling process is then carried out in order to preserve the same Old:Middle-aged:Young proportions as found in the DIP data set.

When comparing the age patterns of the generated networks with those of the DIP, although we observe significant differences in the various age-dependent types of both edges and triangles, the general pattern is maintained in the edges, whereas the number of triangles and the proportions between the different types is strikingly different from DIP (see Appendix C). The resulting networks have less than 50% of the number of triangles found in DIP. The observed pattern cannot be explained by

this gene duplication and divergence model and hence there is reason to assume that the underlying biological mechanism of edge formation is more complicated than that of gene duplication and divergence alone. Further studies are therefore needed to understand how these age-dependent patterns come to be and the limitations which they impose on available functional topologies for the network.

Our results support the idea of a stratified, highly heterogeneous network with highly connected clumps of Old proteins. As an example, Figure 3.5 shows a so-called ego-network centred on YFL021W, a transcriptional regulator associated with multiple nitrogen catabolic genes; the ego-network shows the protein itself, its interacting partners, and the interacting partners of those interacting partners - all proteins within graph distance 2 of the original protein. The prominent clump of Old proteins on the right-hand side of the figure contains proteins associated with the GO term "proteolysis involved in cellular protein catabolic process", most of them being units of the proteosome or associated proteins.

3.7 Conclusions

Protein age, as estimated by lineage specificity, is a readily available explanatory variable to start probing the nature of connectivities we currently see in the yeast protein-protein interaction network. An analysis of the pairwise data shows an over-representation of Old-Old pairs given the frequency of the different age-labelled proteins in the network, even when controlling for age. This over-representation of interactions between older proteins alone explains results on pooled protein age groups which suggested that proteins prefer to interact with proteins within the same age group.

Old-Old-Old triangles are also over-represented under the node model, but once the edge frequencies are taken into account (edge model), the conclusions change. Now all triangles involving proteins of the same age group are over-represented, but so are all

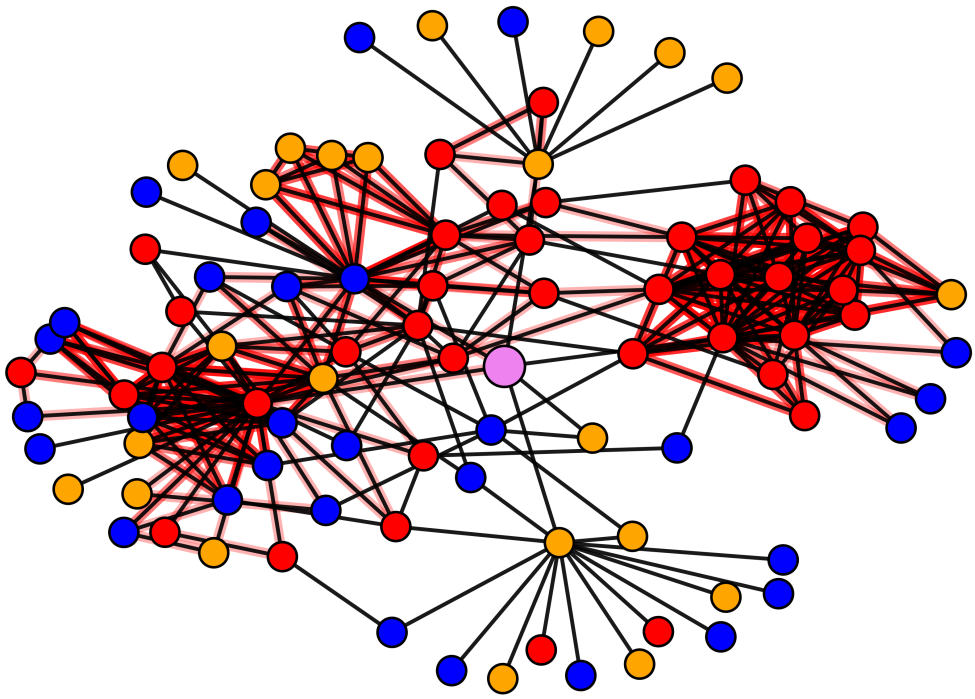


Figure 3.5: Example of an ego-network extracted from yeast DIP with nodes coloured by age. The ego-network is based on the protein 'YFL021W' *(violet)* with its interacting partners as well as their interacting partners. Old proteins are coloured *red*, Middle-aged proteins in *blue* and Young proteins in *orange*. Edges involved in at least one triangle are highlighted in *red*. The prominent clump of Old proteins on the right-hand side contains mostly subunits of the proteosome or proteins associated with it.

triangles involving a Young protein, again casting doubt on the concept of homophily for all protein interactions. The consistent pattern is instead that triangles containing at least one Middle-Old edge were under-represented under the edge model, suggesting negative selection or segregation of Older triangles. Considering only higher degree proteins, Old-Old-Old triangles are still over-represented, while triangles containing Middle-Old edges account for significantly fewer triangles than expected under the edge model. Moreover most triangles in the network are preserved in the sets of high-degree proteins, suggesting dense clustering between high-degree proteins. A gene duplication and divergence model was unable to explain the triangle patterns, pointing to more

complex biological mechanisms of proteome growth.

In conclusion the network contains biological features which cannot be explained by protein frequencies, or protein interaction frequencies, alone. Instead, our findings point to an architecture of the yeast PIN that is highly heterogeneous, with most of triangles being associated with high-degree proteins, and with selective age-dependent interaction patterns. Any network model which tries to capture the behaviour of small subgraphs should take this inhomogeneous and beyond-pair-dependence structure into account.

Chapter 4

Ego-networks

Interactions between Old proteins are over-represented and most triangles in PPI networks (PINs) are amongst proteins with degree 10 or more. The selective age-dependent interaction patterns we observe suggest a very heterogeneous network architecture. In this chapter we focus on small network samples, two-step ego-networks, in both PINs and random graph models. Aiming to start understanding the source of network heterogeneity in PINs, we also relate topological features of these network neighbourhoods to biological characteristics of proteins.

4.1 Ego-networks of a network

In this chapter we explore the interaction patterns in the vicinity of a protein by employing a snow-ball type of sampling method. In this sampling approach a single node is picked in the network (the centre node or ego), then all the nodes directly connected to it are picked, as well as all the edges between these (1st step, 1st layer of nodes picked). In the next step, all nodes connected to the nodes picked in the last step are chosen (2nd step) as well as all the edges between them. The process continues until the desired number of layers or nodes are sampled. The subnetworks resulting from this process are denoted as ego-networks of radius n, where n is the number of

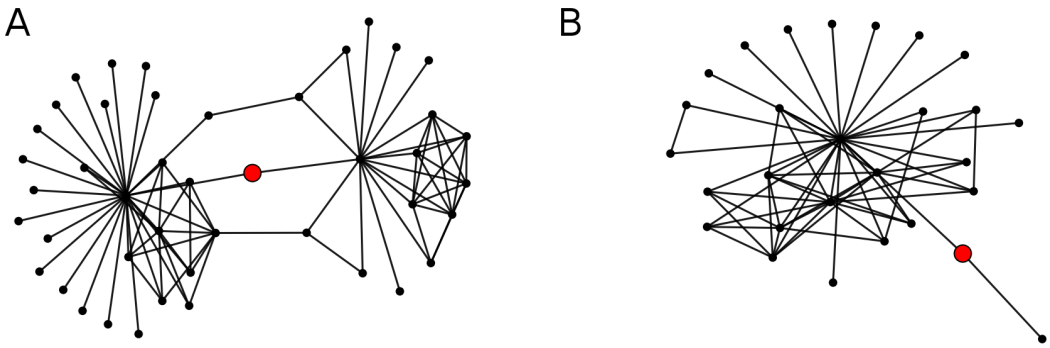


Figure 4.1: Example of two ego-networks with radius 2 from the yeast PIN DIP data set. The seeds/ ego-node for the ego-networks were *A*) YDR196C and *B*) YDR406W which are coloured red.

layers or steps used to build the subnetwork. For instance, the ego-network of radius 2 of a protein/ node p is a (sub)network consisting of all nodes within two edges of p that includes all the edges between those nodes. Figure 4.1 depicts two examples of such ego-networks.

This type of sampling, when applied to every node in the network, results in many node overlaps amongst the ego-networks. Snowball sampling is known to find nodes in proportion to their eigenvector centrality [Newman, 2012]. Empirically, the sampling tends to pick high-degree nodes in relatively short steps due to their high connectivity and hence, regardless of the initial node picked, theses nodes are more likely than others to be included in a given ego-network [Lee et al., 2006].

4.1.1 The heterogeneity of ego-networks

In Chapter 3 we found that over ∼80% of the total number of triangles of the network are observed between high-degree proteins. This finding suggests a very heterogeneous network architecture, with dense clumps of Old proteins interceded by Young ones. To gauge the heterogeneity of a PPI network we devise a three-dimensional plot based on the number of nodes, edges and triangles contained in ego-networks with radius 2 (all

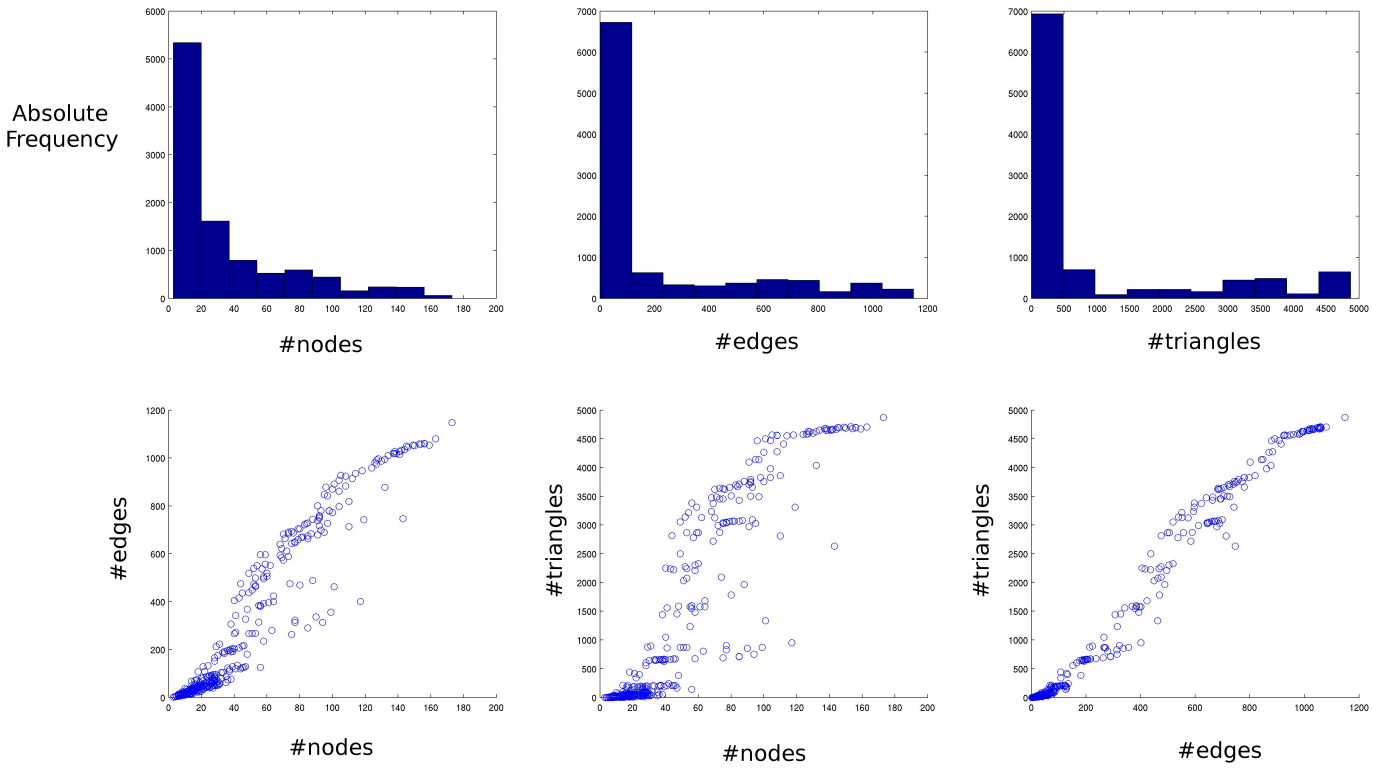


Figure 4.2: Histograms of the number of nodes, edges and triangles of 10,000 ego-networks of radius 2 from the YHC PPI network. Below the histograms are correlation plots between these counts.

nodes within a distance of two edges from the randomly picked ego-node are selected and all the edges amongst all the selected nodes considered). A histogram of these three counts for 10,000 random ego-network samples from the YHC data set (see Table 2.1) is presented in Figure 4.2 (up). Figure 4.2 (down) shows correlation plots between these basic summary statistics; we observe that the number of nodes, edges and triangles have strong interdependencies.

The degree of heterogeneity of YHC can be fully appreciated in the three-dimensional plot of Figure 4.3A as a wide surface formed by the diverse samples of the PPI network. Network models can be also distinguished from PPI networks with respect to their de-

gree of heterogeneity and sampling characteristics. Figure 4.3B shows the same plot as Figure 4.3A, but for a randomized YHC network using the Maslov and Sneppen [2002] algorithm. This process has a pronounced effect on the topology and heterogeneity of the network (mainly by destroying triangles) and hence this model should not be used for inferences on the network beyond pairwise characteristics. Models like GEO3D and ER have even poorer sampling surfaces, as shown in the Figure 4.3C.

4.2 The set of all ego-networks of a network

The random sampling presented in the last section may, despite repeated sampling, not contain all possible ego-networks. Thus, we focus on the set of all 2-step ego-networks, one for each node in the network (see Figure 4.1 for examples). Here we use the larger DIP yeast PIN data to illustrate the richness of these data and how it can be used to mine biologically relevant patterns.

The yeast DIP PIN is composed of 5,078 distinct proteins, 5,020 of which are involved in neighbourhoods with more than two nodes and two edges. Figure 4.4A and B show histograms of the number of nodes and edges of these 5,020 ego-networks. As in Figure 4.2, we see a negative exponential-like distribution for these measures – most of the ego-networks have relatively few nodes and edges, whilst a few have a considerable number of them, probably due to the inclusion of high-degree nodes which boost the number of nodes and edges of an ego-network. Figure 4.4C and D show that these ego-networks have average degrees between 2 and 18, lower values being more frequent, and that most ego-networks have low graph densities.

Ego-networks of radius 1, *i.e.*, only the immediate neighbours of a protein and the interactions between these, have skewed distributions to lower numbers of nodes and edges and tend to have a more pronounced decay (heavy tail). On the other hand, ego-networks of greater radius contain a larger number of nodes and edges in their ego-

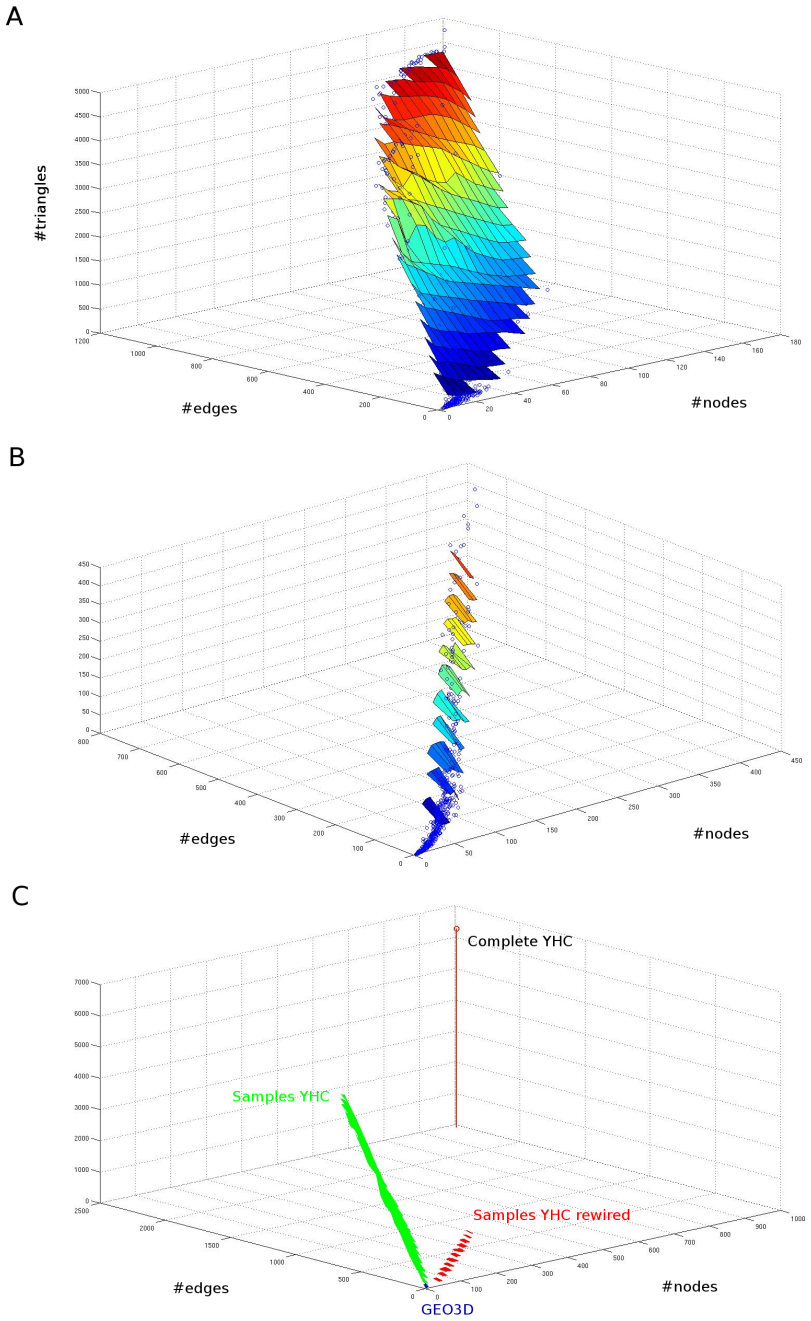


Figure 4.3: Three-dimensional plot of the number of nodes, edges and triangles of 10,000 ego-networks taken from **(A)** YHC, **(B)** a randomized/ rewired YHC network and **(C)** of YHC *(green)*, a randomized network of YHC *(red)* and GEO3D *(blue)*. The surface in (A) and (B) is created by triangulation over the different data points and its colouring has no particular meaning besides showing the increase of triangle counts, with higher counts having warmer tones (red).

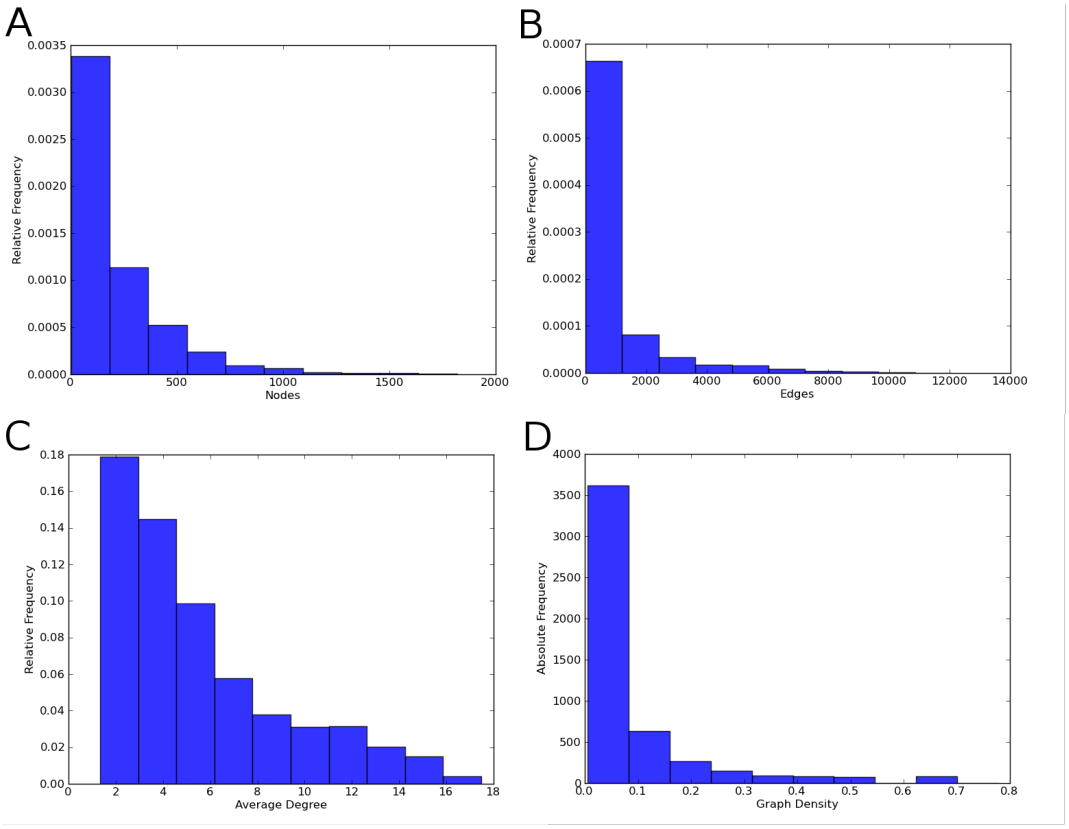


Figure 4.4: Histograms of summary statistics of all 2-step ego-networks of the yeast DIP PIN. The network summary statistics considered are *A)* number of nodes, *B)* number of edges, *C)* average degree and *D)* graph density.

networks, but these are more homogeneous and display smoother distributions (light tail). This is not unexpected since the average shortest path length of all node pairs in the giant component of the overall DIP yeast network is 3.98 and the greatest possible shortest path (network diameter) is only 10. With ego-networks of 3 or 4 steps, the probability that most nodes in the network will be included is quite high and indeed for the case of radius 3 some ego-networks already contain around 20,000 edges, almost the total number of edges in the network which is approximately 22,000. Small ego-networks of radius 1 can be probed directly from the edge list and provide relatively little information about the protein's functional surroundings, hence our focus on 2-step ego-networks.

As we observed in Figure 4.3, measures like the number of nodes in all the 2-step ego-networks of a network *versus* their respective graph density and another summary statistic such as the number of triangles, result in unique landscapes that are very specific to the particular network being analysed. Figure 4.5 shows some examples of these 3D plots for PINs of the DIP database. Note that we can use counts for subgraphs other than triangles; see for instance Figure 4.5B.

By observing the xy plane in Figure 4.5 alone we understand that most ego-networks with a high number of nodes occur at relatively low graph densities; they still have a large number of edges and account for high subgraph counts, but this number is small when compared to the number of edges they could hold. Ego-networks at high graph density only have a few nodes, by definition, these represent small network cliques or nearly-complete small subnetworks. Different PINs, with different levels of coverage and completion, display different numbers of subgraph counts and each plot of Figure 4.5 is unique. The equivalent landscapes for random graph models are typically poorer and more homogeneous than those of PINs. Figure 4.6 shows the same plot as Figure 4.5A, but using as input networks of random graph models generated to match the number of nodes and edges of the yeast DIP PIN. We consider a three-dimensional geometric random graph [Penrose, 2003] and a randomised version of the yeast PIN [Maslov and Sneppen, 2002] (see also Figure 5.12 for similar plots with the Erdős-Rényi and gene duplication and divergence models).

The 2-step ego-networks in the geometric model network (Figure 4.6A) have small sizes when compared to the yeast DIP PIN (Figure 4.5A). These ego-networks contain at most 70 nodes whilst the ego-networks of the yeast DIP PIN contain up to 1,500 nodes. The numbers of triangles in the ego-networks of these two networks are also strikingly different: ego-networks of the geometric model network contain up to ~ 1000 triangles, whereas in the yeast DIP PIN these counts go up $8,000$.

The randomised model network contains large ego-networks with graph densities

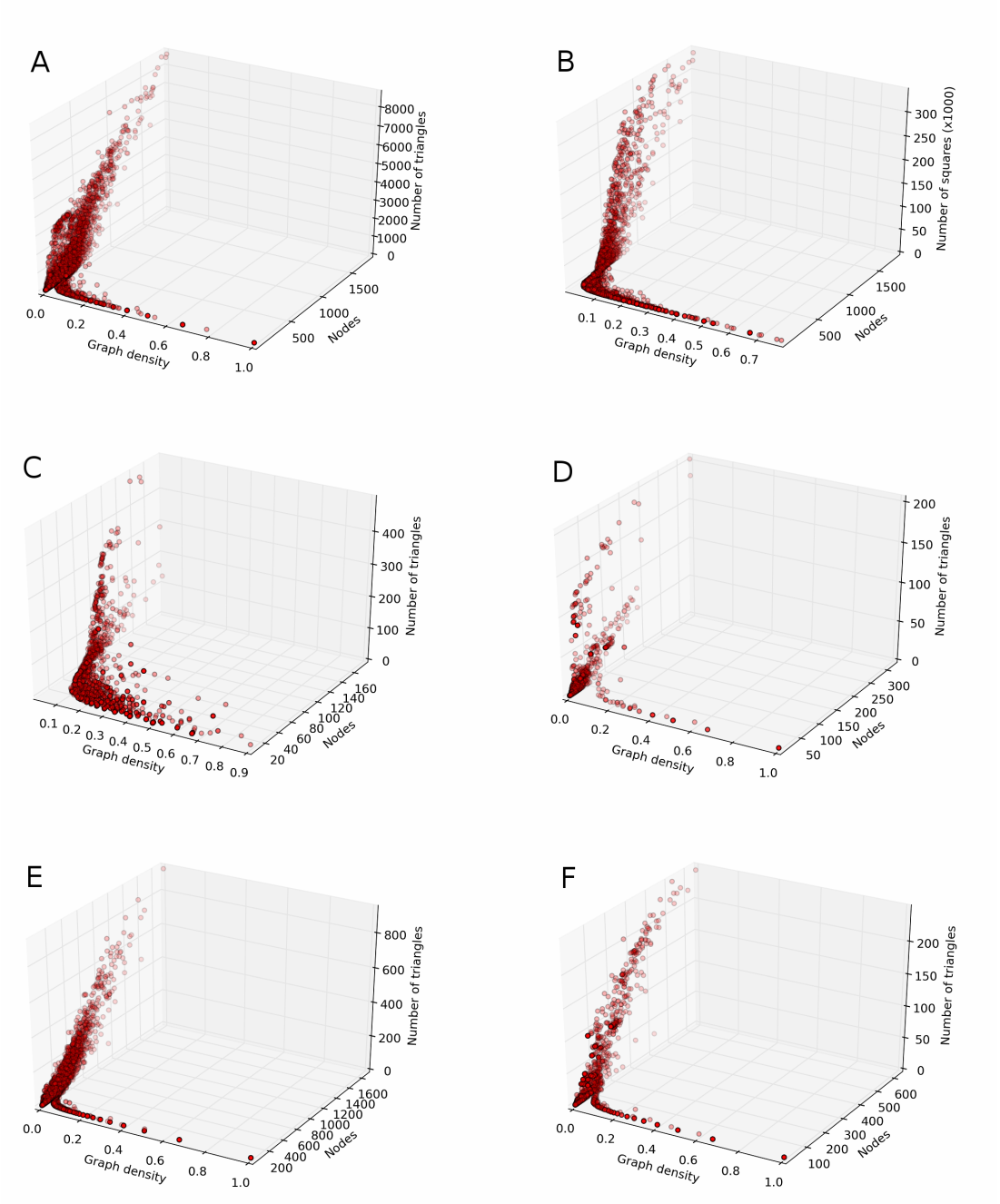


Figure 4.5: Three-dimensional plots of network summary statistics for all ego-networks, each represented by a dot, of a given network. All the plots have number of nodes and graph density in the xy plane. The z axis shows *A*) the number of triangles for the yeast DIP PIN data set; *B*) the number of squares for the yeast DIP; *C*) the number of triangles for the yeast DIP-Core; *D*) the number of triangles for the human DIP; *E*) the number of triangles for the fly DIP and *F*) the number of triangles for the worm DIP.

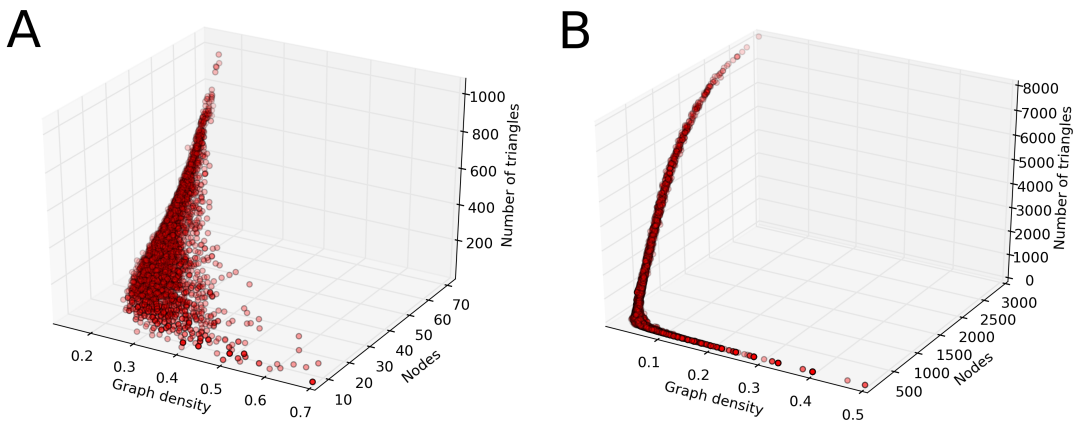


Figure 4.6: Three-dimensional plots of the graph density *versus* number of nodes and triangles counts of all ego-networks with more than 5 nodes of networks simulated from a A) three-dimensional geometric random graph model, and B) randomisation model. Each dot represents one ego-network. The networks were generated to match the number of nodes and edges of the yeast DIP PIN.

up to 0.5 (Figure 4.6B). In this model network the number of triangle counts of its ego-networks reaches $\sim 8,000$ resembling those of the yeast PIN. Nonetheless, the triangle counts of the entire network are considerably larger in the PIN ($14,933$ *versus* $8,171$). Perhaps the most interesting feature is that the ego-networks of this randomised model are very homogeneous and appear to follow a specific trend. Most of the random graph models do not model the network heterogeneity found in PINs. The ego-networks in these models are similar amongst themselves and the variety of neighbourhoods found in PINs is lost.

4.3 Biological features of proteins and their neighbourhoods

This section describes some results on associating neighbourhoods and protein characteristics.

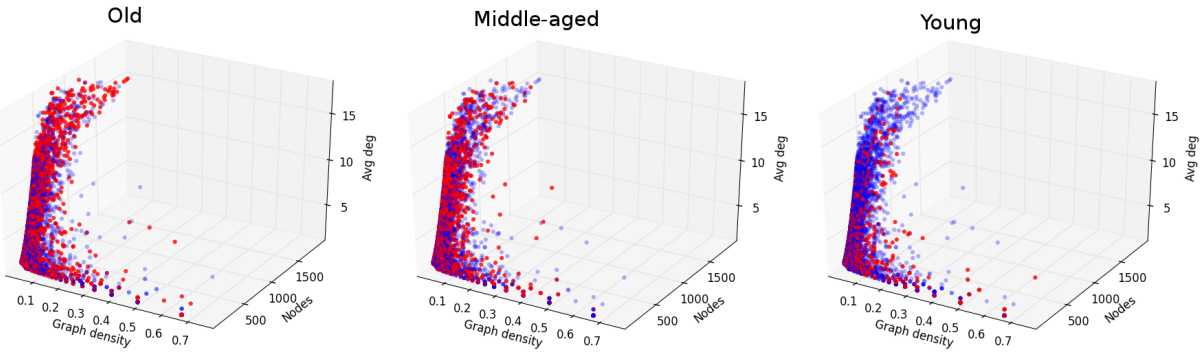


Figure 4.7: Three-dimensional plots of the number of nodes *versus* graph density and average degree of all ego-networks in the yeast DIP PIN. All individual ego-networks are coloured blue except for the ones of a particular age category, Old, Middle-aged and Young (left to right), which are highlighted in red.

4.3.1 Protein age and network neighbourhoods

In Chapter 3 we described an in-depth analysis of the protein age patterns found in the edge and triangle subgraphs of the yeast PIN. Here we relate protein age with topological properties of neighbourhoods using the yeast DIP PIN data. Employing the snow-ball sampling approach used in the previous sections, we build two-step ego-networks for each protein and relate summary statistics of these ego-networks with the age of the central proteins (egos) used to construct them. Here we calculate protein age as in Chapter 3 (see Section 3.2). Figure 4.7 shows three-dimensional plots akin to the ones presented in Figure 4.5 with graph density and number of nodes of the ego-networks in the xy plane, but with average degree on the z axis. The different plots show the ego-networks based on the age of their ego-protein.

From a total of 5051 ego-networks with an age tag in the yeast DIP PIN, 1998 are built around Old proteins, 1910 Middle-aged proteins and 1143 Young proteins. Figure 4.7 shows that Old proteins have ego-networks which are larger and have higher average degree. To better characterise the distribution of the number of nodes found

in these ego-networks we report the median value, which for Old-based ego-networks is 210.5 nodes. The neighbourhoods of Middle and Young proteins tend to have lower average degrees and fewer nodes with medians of 160 and 121 nodes, respectively. We also monitored the group of ego-networks for which no age was available. This group had the smallest neighbourhoods (Median=76 nodes), which may be due to them being poorly studied leading to a lack of annotation data. The topological complexity of the neighbourhoods can be judged by the occurrences of subgraphs within them. Here we focus on the number of triangles in these ego-networks. Figure 4.8 shows histograms of the number of triangles found in ego-networks around proteins of a given age category.

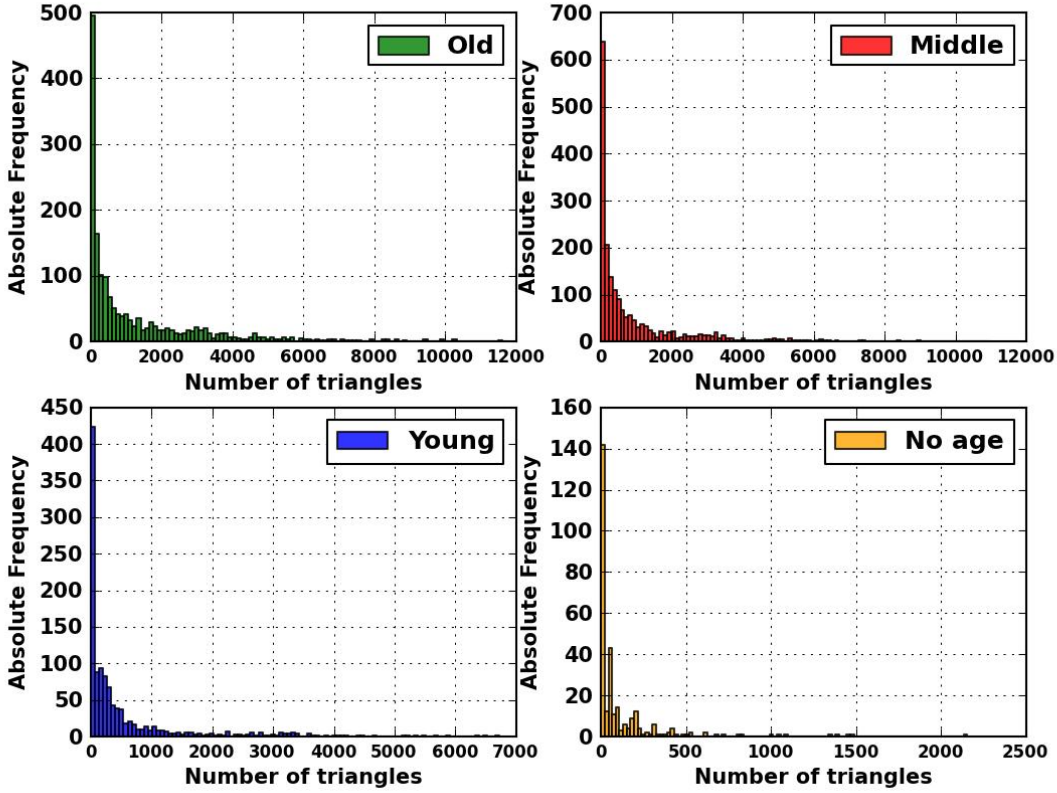


Figure 4.8: Histograms of the number of triangles found in ego-networks of the yeast DIP PIN according to the age of the centre protein.

Neighbourhoods of Old proteins have more triangles in them (median=456.5 triangles) then the ones of Middle-aged (median=311.5 triangles) or Young proteins (median=186 triangles). Proteins with no age tag have the least triangles within them with a median of 35 triangles. These results confirm and expand our findings in Chapter 3. There, we concluded that the edges and triangles between Old proteins are over-represented even controlling for the node and pairwise interaction frequencies we observe. Here we observe a similar trend with Old proteins being associated with larger and more complex neighbourhoods. We can hypothesise that Old proteins tend to be part of complex, highly-connected neighbourhoods perhaps involved in basic molecular functions which are essential for the cell, whereas Young proteins may tend to be involved in new, adaptive functions and hence are normally found in less connected, smaller network surroundings.

4.3.2 Network neighbourhoods and other biological features

A similar analysis can be conducted for any biological property of the central protein in the ego-network. A second characteristic we consider is the structural class of the ego-protein. We use the Structural Classification of Proteins (SCOP) database [Murzin et al., 1995], release 1.75, to annotate the proteins in the yeast DIP PIN. SCOP is a manually-curated database that uses visual inspection of structures as well as automated tools to characterise, compare and cluster protein structures together in an hierarchical classification. Here we focus on the root level which comprises several structural classes based on secondary structure: *all alpha*, *all beta*, *alpha and beta* (a/b; α-helices intersperse with β-strands), *alpha plus beta* (a+b; α-helices and β-strands segregated in different parts of the structure), *multi-domain* (proteins with different folds for which no homologues are known), *small proteins*, *coiled coil proteins* and *membrane and cell surface proteins and peptides*. We assigned a SCOP class to every protein in the yeast DIP PIN, proteins with multiple class were counted multiple times so that they

are present in the several class groups that characterise them. Of the 5051 proteins in the PIN data, 1946 did not have structural information (*"No SCOP tag"* group). For those proteins with known structure (3105), we assigned 3954 terms across the 8 classes and for all proteins within each class we analyse the distribution of network properties of their ego-networks. As with protein age, we report the median of these distributions for the number of nodes, edges and triangles. These values, their dependency with each other and with the number of proteins in that class is presented in Figure 4.9. The results put *all beta* proteins in the largest ego-networks with high number of nodes, edges and triangles. These are closely followed by *all alpha*, *a+b*, and *coiled coil*. A second cluster constituted of *small* and *a/b* is found in neighbourhoods displaying intermediate medians for nodes, edges and triangles. Proteins within the *membrane*, *multi-domain* classes and without SCOP classifications group have the least number of nodes, edges and triangles in their ego-networks.

Our results show that SCOP class and size of the ego-network are not unrelated. Hence an analysis similar to the one presented in Chapter 3 could be carried out to gain further insight into protein interaction networks. Other explanatory variables could also be considered, such as intra-cellular location.

4.4 Conclusions

Ego-networks sample the surroundings of a protein in PPI networks. When considering the set of all possible ego-networks of the yeast PIN, we observe a wide topological diversity. We judge this diversity in three-dimensional plots of the number of nodes, graph density and subgraph counts, *e.g.* number of triangles, of these ego-networks. In contrast to PINs, networks of random graph models, such as the ER, GEO3D, DMR models or randomised networks, display ego-networks which are typically smaller and less diverse.

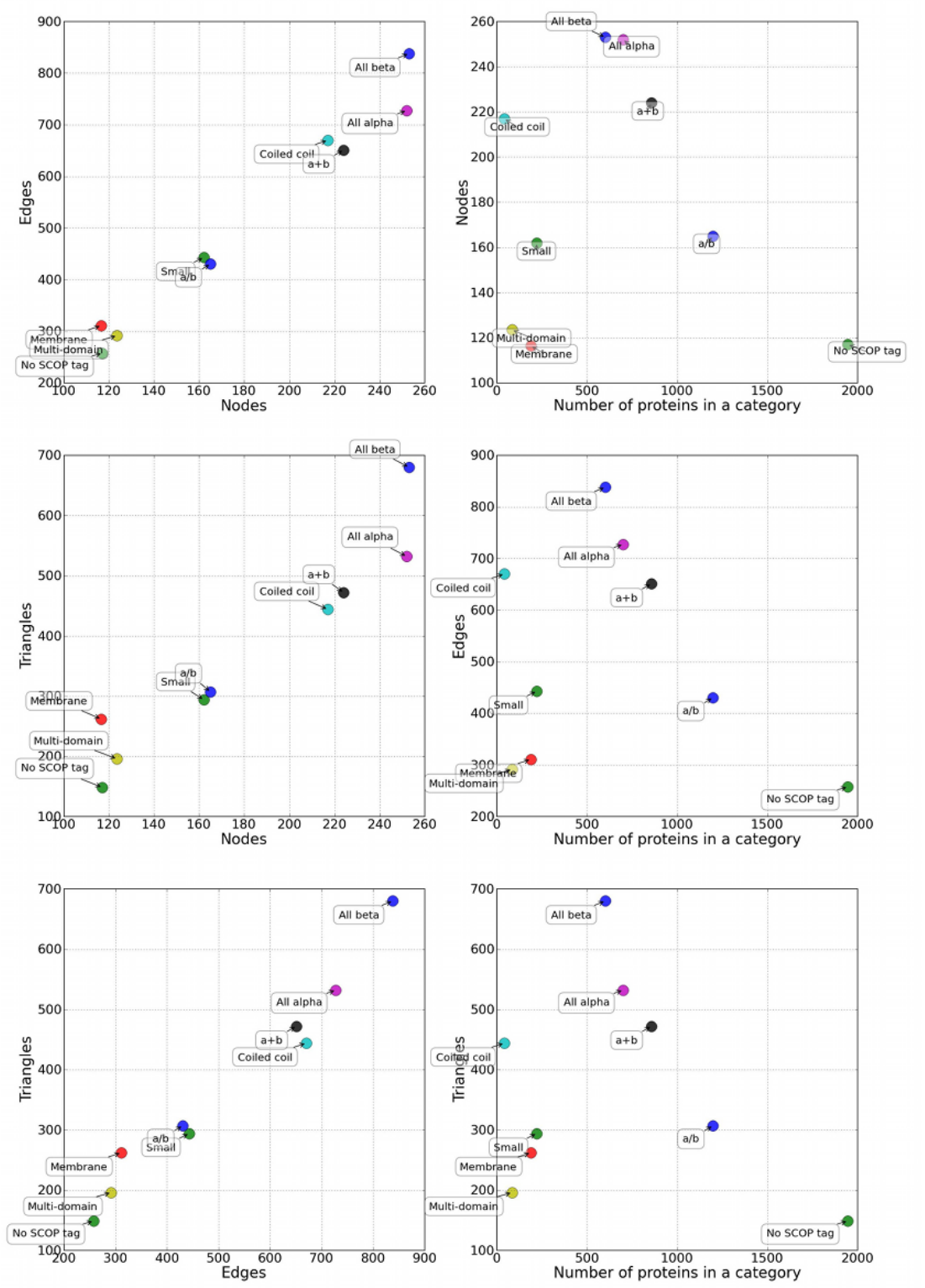


Figure 4.9: Median values of the distributions of the number of nodes, edges and triangles for ego-networks constructed around proteins of a particular structural SCOP class.

In this chapter we associate biological properties of proteins with their network neighbourhood. On average, Old proteins have neighbourhoods which are roughly twice as large and contain over twice as many triangles than the neighbourhoods of Young proteins.

We also find a relationship between neighbourhood size and SCOP structural class. Proteins of the structural class "all beta" or "all alpha" are often the centre of large, dense step-two ego-networks. To gauge the significance of these results, an analysis similar to the one for protein age in Chapter 3 could be carried out.

In the next chapter we explore how these ego-networks and their subgraph counts can be used for network comparison.

Chapter 5

Alignment-free method for network comparison

Within a PIN, a rich variety of network neighbourhoods can be found. In this chapter I employ these samples in a new, alignment-free method to compare networks. The method counts subgraphs in all protein neighbourhoods of a network and compares them with another query in an averaging, many-to-many approach. Using only network data the method can recreate phylogenies between species and opens many possibilities for further studies.

5.1 Comparison in biology

Most of our knowledge of evolution at a molecular level, and arguably one of the biggest contributions that computer science has made to biology, comes from the ability to compare biological sequences of proteins or nucleic acids with each other. This comparative, evolutionary paradigm came to be thanks to both the unprecedented generation of sequencing data and to the parallel steady development of computational algorithms.

5.1.1 Sequence comparison

Sequencing refers to the process of determining the primary structure of an unbranched biopolymer. The *sequence* of the polymer is a symbolic linear representation of the order in which its monomers exist in the actual three-dimensional structure of the molecule. Between the 1950s and 1970s, the seminal works of Edman and Sanger [Edman, 1950; Sanger and Coulson, 1975] among many others, resulted in robust experimental techniques capable of routine generation of sequencing data. Sanger's efforts to determine the sequence of insulin in 1955, the first complete protein sequence, and the development of the relatively fast "dideoxy" technique for DNA sequencing (1975), led the scientific community to actively pursue the sequencing of many other molecules. Further improvement and automation of these techniques, particularly DNA sequencing, to high-performance and high-throughput versions culminated in recent sequencing projects such as "The Human Genome Project" (1990–2006), "The 1000 Genomes Project" (2008 – ongoing) as well as the full genome sequencing of many species.

Soon after the first experimental sequencing data became available, many theoretical models and algorithms started being developed to aid data analysis and interpretation. Jukes and Cantor [1969] were the first to propose a model describing the rates of nucleotide changes during evolution. The model postulated a Markov process for the substitution of base pairs assuming equal frequency and mutation rate for all base pairs. Many variations and improvements of this model are still used nowadays, forming the basis of phylogenetic analysis based on sequence comparisons. Dayhoff first proposed the reconstruction of evolutionary history from sequence data [Dayhoff, 1969] and pioneered a probabilistic model of protein evolution with the construction of the first amino-acid substitution matrices (Point Accepted Mutation, PAM matrices) [Dayhoff et al., 1978].

Together with the deluge of sequence data and the first models of protein evolution came the increasing need for efficient algorithms to compare sequences. Needleman

and Wunsch [1970] applied dynamic programming to find the optimal global alignment between two sequences (pairwise). The algorithm constructs a "trace-back" matrix based on a similarity score that penalises sequence mismatches and gaps. The pathway within the matrix with least penalties can then be traced back and corresponds to the (or an) optimal alignment. Despite their usefulness, global alignments perform poorly on distantly related species with only some localised similarities between them. To tackle this, Smith and Waterman [1981] introduced a different similarity scoring system whose focus was on local alignments. The algorithm compares segments of all possible lengths of the entire sequence and chooses the one with maximum score. It can find optimum local alignment segments between sequences at the expense of finding the best global score. Although the method guarantees optimal alignments, it is time consuming and computationally expensive. Feng and Doolittle [1987] proposed CLUSTAL, a progressive algorithm giving approximate results to the much harder problem of multiple sequence alignment (finding similarities between many sequences simultaneously). BLAST [Altschul et al., 1990], the most cited algorithm's article in bioinformatics in the 1990s, is also a local alignment method which is based on an approximation of the Smith-Waterman algorithm. BLAST uses an heuristic approach that by sacrificing the accuracy and precision of the results is able to make faster comparisons. The method revolutionised molecular biology by allowing researchers to submit a query sequence and get fast responses identifying library sequences that are similar to the query above a certain threshold. The results are probabilistic and each sequence match has an associated E_{val} representing the probability of a single hit by chance alone. These results are typically used to make phylogenetic and functional inferences on the query by finding more studied sequences in different species, targets for structural homology modelling among others.

Before BLAST was proposed in 1990, many other sequence comparison tools independent from sequence alignment were also developed. These included the calculation

of global properties in sequences such as their hydropathic character [Kyte et al., 1982]. Blaisdell [1986] devised a measure of similarity between sets of sequences not requiring sequence alignments. The measure compared distributions of adjacent pairs or triplets in a sequence and was the first alignment-free sequence comparison method. Stormo and Hartzell [1989] also stressed the importance of finding features in DNA sequences independent from their alignments and pioneered the search of sequence motifs and domains.

Nowadays, alignment-free methods are increasingly used due to their speed and ability to cluster sequences that are too distant from each other to produce meaningful alignments. The algorithm presented in this chapter for network alignment is inspired in these alignment-free sequence comparison methods and more details on these now follow.

5.1.1.1 Alignment-free sequence comparison methods

Alignment-free sequence comparison is still a growing field. This is mainly due to the realisation of the biases in the traditional methods which are commonly used to model sequences such as Hidden Markov models or Bayesian theory-based models. These tend to see biological molecules as long linear sequences, assuming conservation of contiguity, thereby ignoring their 3D structure and the long-range interactions in it. Alignment methods also cannot usually cope with the fluidity of certain, usually short, sequence elements due to recombination events [Vinga and Almeida, 2003]. These make them poor at aligning distantly related species and cis-regulatory regions that precede genes - mainly due to the short motifs that compose them. Alignment-free methods also tend to be faster than alignment methods and are now being used to analyse the large amount of data produced by next-generation sequencing technology [Song et al., 2012] or as filtering steps prior to alignment-based methods.

Here we focus on alignment-free methods based on word frequencies. We can think

of a sequence as a collection of symbols from an alphabet, say $\{A, T, C, G\}$. Words represent smaller segments of that sequence with a given length k. Alignment-free methods based on words entail statistics that combine counts of occurrences (with overlap) of these words, k-tuples (also called k-mers or k-words) in a given set of candidate sequences.

Let S be a sequence with $n(S)$ letters drawn from a finite alphabet with length r. A segment of k letters is called a k-mer and the set $W_k = (w_{k,1}, w_{k,2}, ...w_{k,X})$ of all possible k-mers of S has $X = r^k$ elements. A sliding window can be run through the sequence from the position 1 to $n - k + 1$ and the number of occurrences for each element of W_k found. Let X_w represent the sum of the counts of W_k for S and Y_w the counts of a different sequence E with $n(E)$ letters. A popular similarity statistic, D_2, can compare these counts such that

$$D_2 = \sum_{W_k} X_w Y_w.$$

Intuitively, the more related the sequences are, the more similar their k-mer content will be and higher values of D_2 will be obtained. Nonetheless, this simple statistic suffers from lack of standardisation and is dominated by background noise [Lippert et al., 2002].

Reinert et al. [2009] proposed a modified statistic, called D_2^S, which self-standardises the counts for the sequences. Heuristically, if the word counts are not rare, then the counts vector would be asymptotically normally distributed (with non-trivial covariance matrix). From Shepp, L. [1964] it follows that D_2^S should then be approximately normal with mean zero, and indeed D_2^S was shown to be approximately normally distributed without being dominated by single-sequence noise. More formally, let $p_{w_k} = \prod_{i=1}^{X} p_{w_i}$ be the probability of occurrence of W_k for a given sequence. The counts are then standardised around the mean by

$$\tilde{X}_w = X_w - (n(S) - k + 1)p_w \text{ and } \tilde{Y}_w = Y_w - (n(E) - k + 1)p_w$$

and the D_2^S statistic defined as,

$$D_2^S = \sum_w \frac{\tilde{X}_w \tilde{Y}_w}{\sqrt{\tilde{X}_w^2 + \tilde{Y}_w^2}}.$$

In this chapter we describe how similar statistics can be used for network comparison using counts of subgraphs in networks, but first we review the current methods used in the field of network comparison.

5.1.2 Network comparison

Many recent advances in high-throughput technologies are driving an exponential increase of interaction data, often modelled as graphs. An analogy can be made with the earlier days of the sequence analysis field where many strategies were proposed to tackle the vast amount of data being generated. Analysing and comparing the various types of interaction data promises a paradigm shift in biology akin to the one triggered by sequence comparison [Sharan and Ideker, 2006]. Nonetheless, for a coherent incorporation of this information into useful models of cellular function we need meaningful and efficient algorithms to carry out the comparative analyses. Here we focus on protein-protein interaction networks (PINs), which are but one example of such data providing an additional layer of understanding on biological systems. As mentioned in Chapter 1, the main techniques to probe protein interactions were proposed in the 1990s and 2000s with the development of the two-hybrid screening [Fields and Song, 1989; Ito et al., 2000; Uetz et al., 2000] and affinity purification followed by mass spectrometry (TAP-MS) [Krogan et al., 2006; Rigaut et al., 1999]. Interestingly, publicly available databases of these data were created shortly after its arrival, e.g. DIP [Xenarios et al., 2000] or BioGRID [Breitkreutz et al., 2003], unlike sequence data

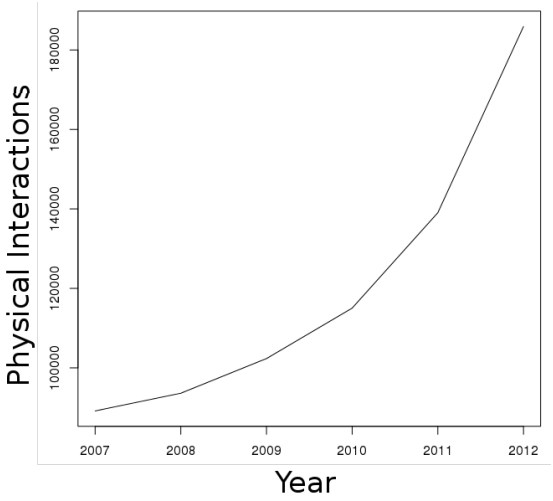


Figure 5.1: Total number of physical interactions found in the BioGRID database [Breitkreutz et al., 2007] from 2007 to 2012.

where databases such as GeneBank (1993) or Swiss-Prot (2002) were only established 30 to 40 years after the first sequence data became available. Since its appearance, the amount of PIN data continues to grow exponentially. Figure 5.1 shows the number of physical, experimentally-determined protein interactions deposited in the BioGRID database [Breitkreutz et al., 2007] since 2007.

Global network summary statistics like the degree distribution or clustering coefficient aid in the characterisation of PINs, but are of little use in comparing networks since they are too broad a descriptors and lack biological interpretation. For instance, many different networks can usually be found that share a similar value for a given global summary statistic, e.g. the degree distribution. Milo et al. [2002] proposed a connection between specific subgraphs and biological function and coined the term network motif – a subgraph that is found to be over-represented with respect to an ensemble of suitable random networks. How PINs can be compared based on network motifs is not trivial and much theoretical work relative to their distribution, standardisation and counting algorithms remains to be done. The analysis of motifs and global properties in networks happened relatively early when comparing to the

timeline of the sequence world. Also, unlike sequence comparison, the first network matching algorithms [Dandekar et al., 1999; Ogata et al., 2000] were proposed before the establishment of PIN databases albeit, to date, there is still no single best method for doing this (for more details see the next section). Crucially, we lack robust evolutionary models describing protein interactions and their evolution. Models similar to the Jukes-Cantor substitution model or Hidden-Markov models, have been difficult to achieve due to the incompleteness of the data and the non-linearity of networks [Sharan and Ideker, 2006]. For instance, computer science has essentially solved the graph matching problem and several efficient algorithms now exist that retrieve the exact or best subnetwork matches between two networks although the integration of these into biologically meaningful alignments has proven far more difficult. The construction of multiple network alignments is also still under active research. Comparing multiple networks requires an extremely fast algorithm that is able to cope with the size of the multiple datasets. The different sizes and degrees of noise and incompleteness in the PINs of the various species also present several challenges. The next section describes in greater detail the efforts made so far to compare PINs.

5.1.3 Alignment-based methods

Current methods for comparing PINs are mainly alignment-based methods and can be separated into local or global alignment algorithms. These methods are typically formulated with two objective functions whose trade-off resolves the alignment. These objective functions reflect two, often competing, goals: one of ensuring that proteins that perform similar biological functions are matched, this is usually done using BLAST similarity scores between protein pairs, and one of conserving network topology, generally by maximising the number of conserved edges.

Local network alignment methods focus on finding similar subnetworks between the different query networks at the expense of a global match. These subnetworks are

typically unrelated regions of graph isomorphism and these methods resemble the quest of finding conserved motifs in sequences. For instance, Network BLAST-M [Kalaev et al., 2009] greedily finds independent, highly conserved subnetworks given a set of query networks considering the phylogeny between them. Ali and Deane [2009] argued that sequence similarity alone might be insufficient to accurately match proteins that are functionally similar and incorporated a protein functional similarity measure in a match-and-split local alignment algorithm. Using gene ontology annotation terms in combination with sequence similarity they were able to retrieve subnetworks that contained more conserved interactions and were more functionally coherent than when using sequence similarity alone. Nonetheless, even using this combined information, when aligning a 10,000 node human PIN with a 5,000 node yeast PIN only 303 clusters were found, corresponding to 1479 unique proteins. The alignments obtained through these methods often exclude most proteins in the networks as only subnetworks with high similarity are considered [Singh et al., 2008], and the possibility of a one-to-many mapping of subnetworks makes it non-trivial to establish clear relationships between the different query networks [Shih and Parthasarathy, 2012; Singh et al., 2008].

Global alignment methods consistently map proteins across networks and try to find the best overall compromise between sequence and topology aiming to align as many nodes as possible across the networks.

IsoRank [Singh et al., 2008] has the intuition that if two nodes, each of a different PIN, have a high sequence similarity and align well, than their neighbours should also align well. This is posed as an eigenvalue problem involving a large similarity score matrix which is iteratively updated and optimised. The score has a non-zero value for every node-pair involving nodes from different query networks and reflects similarity at both sequence and network topology levels with a tunable parameter that decides the relative weight of each of these. The IsoRank algorithm innovation was that it introduced flexibility in matching nodes whose sequence similarity was relatively poor,

but whose network topology match was relatively good. With no sequence information, the algorithm simply returns the maximum common subgraph (the largest subgraph common to all candidate networks) between the networks. IsoRankN [Liao et al., 2009], an improved version of IsoRank, uses a spectral clustering method to produce a final common subgraph, typically consisting of multiple components.

Graemlin 2.0 [Flannick et al., 2009] is a multi-step local alignment method that produces global alignments. It uses a feature-based scoring system that includes node features, such as node deletion, duplication and mutation, and edge features, such as edge conservation. The parameters are learned automatically from a training set with a given phylogenetic relationship and then optimised for the candidate set of networks. The last step of the algorithm takes a set of disjoint local alignments and assigns probabilities for a node class to belong to a given local alignment. Intuitively, a node class will be kept in a given alignment if the scoring function increases upon reassignment.

PINALOG [Phan and Sternberg, 2012] first explored the use of community detection and matching in global network alignment in an algorithm that uses both sequence similarity and GO annotation terms. The communities (dense subnetworks) of a PIN are first detected and scored against all others of a different query PIN. Similar communities have a high number of inter-species proteins with high similarity scores. The hungarian method is then used to map them to each other. Similar proteins in mapped communities constitute seed pairs for an extension mapping algorithm which takes into account network topology in addition to protein sequence and functional similarity. Compared to IsoRank, PINALOG is able to retrieve more conserved interactions (3319 *versus* 717) and more interologs (460 *versus* 136) in a human–yeast alignment.

Shih and Parthasarathy [2012] recently provided a different method which greatly speeds up multiple global network alignment by including an independent pre-processing step that computes a similarity matrix for all proteins using sequence information. The

intuition here reinforces and furthers the motto of IsoRank by assuming that highly similar proteins, sequence-wise, will provide good match-seeds. These are expanded and merged in subsequent steps where network topology is taken into account.

The final alignment produced by these methods can then be used to find functional, as opposed to sequence-based only, conserved components or modules across species, as well as functional ortholog predictions. It can also be used as a phylogenetic tool providing a protein interaction view of relationships between species/ networks that complements the traditional sequence-based one [Chor and Tuller, 2006; Erten et al., 2009].

5.1.4 Problems with current methods and possible solutions

PPI network alignment methods are all based on the loose premises that the respective orthologs of two interacting proteins also interact, the so-called *interologs*, or that orthologs will share neighbourhood topology. Most current algorithms such as IsoRank use a combination of these assumptions to increase the alignment coverage. Although there is evidence for the existence of such conserved interactions across species [Matthews et al., 2001], particularly in proteins with high sequence similarity, this is by no means always the case and there is growing evidence that we might have to rethink functional homology. Mika and Rost [2007] tested how accurate homology-based inferences of PPI across-species are and found that only for extremely high values of sequence similarity these are justified, and even then, the accuracy of transferring interactions across-species for proteins that have percentage sequence identity values $> 70\%$ was just around $10 - 15\%$. In a recent study, Lewis et al. [2012] also investigated the validity of inferring interactions across-species taking the noise and incompleteness of the data into account. Given the interaction A–B in one species, they look for the interaction A'–B' in another species using *blastp* reciprocal hits and a variable threshold of sequence homology to find A' and B'. They found that the fraction of correctly

transferred interactions is at most 3% at $E_{val} \geq 10^{-10}$. Even at the fairly stringent criterium of $E_{val} \geq 10^{-70}$ only 30% of the interactions were correct and these correspond to species that have recently diverged such as yeast and *S.pombe* and is far lower for all other species' pairs.

The most recent global alignment method [Shih and Parthasarathy, 2012] used 7 species in the DIP database and achieved a high coverage of 24,119 proteins (out of 25,555), but only 17,365 edges are conserved out of 78,611. For the same dataset, IsoRankN finds 4,696 conserved edges out of the 54,364 possible ones between the homologous proteins matched across species. Whilst this might be due to genuine evolutionary factors, the problem remains that currently we have no clear way of assessing how good an alignment is due to the lack of direct functional experimental evidence. Enrichment analysis of GO or KEGG Ontology annotation terms within protein classes/ clusters try to tackle this weakness, but these terms are often too general to provide a clear picture of how aligned nodes match functionally across species and many methods are now beginning to use them in order to improve protein matching across species making validation more complex.

Ali and Deane [2010] studied the effect of errors and incompleteness in alignments of simulated networks. Using models such as the Duplication-Divergence model [Pastor-Satorras et al., 2003] they found incompleteness of the data, false-negatives, to be much more of a hindrance to correct network alignments than the presence of false-positives, which results in relatively fewer alignment errors. They estimated that only nearly complete networks ($> 90\%$) can produce reliable alignments for cross-species comparisons. Perhaps the most striking observation of the study was that even using complete, error-free networks results in very little overlap between networks (small alignment) after evolution at currently accepted rates. If we accept their models of network and orthology evolution, aligning a yeast-like network (5,000 nodes and 21,000 edges) with a human-like network (9,100 nodes and 33,000 edges) results, using the

ideal, error-free, complete networks, in alignments with just about 2,000 nodes. If this scenario were to be confirmed or approximated, this would render alignment-based cross-species comparisons very difficult to make and to interpret.

In this chapter we introduce a method that dispenses the notion of sequence homology or functional one-to-one matches to compare networks. Current evidence points to a rate of change in PPIs far larger than expected and close matches seem not to be the rule, but rather the exception or at least far less plausible [Lewis et al., 2012; Shou et al., 2011]. The method presented in the next section assumes that similar neighbourhoods do exist, but makes no attempt at matching them.

5.2 *Egotif* – an alignment-free method for network comparison

We propose a method for network comparison based on the core concept that similar networks will on average contain similar local neighbourhoods within them. For each query network we build a two-step ego-network for every one of its nodes – our description of its network neighbourhoods. We then compare these between networks of interest in an averaging, many-to-many, fashion. Sequence homology thresholds or interolog assumptions to match particular proteins of a network with another are purposely dispensed. Here we use subgraph counts as the raw material for making neighbourhood comparisons since they provide a sensitive measure of the neighbourhood's topology and may be linked with its functionality [Milo et al., 2002]. The method is akin to alignment-free sequence comparison methods and very different in spirit to current network alignment methods as it does not try to align proteins or match particular network neighbourhoods.

Figure 5.2 illustrates the differences between our approach to network comparison and current standard methods. Network alignment methods find the largest possi-

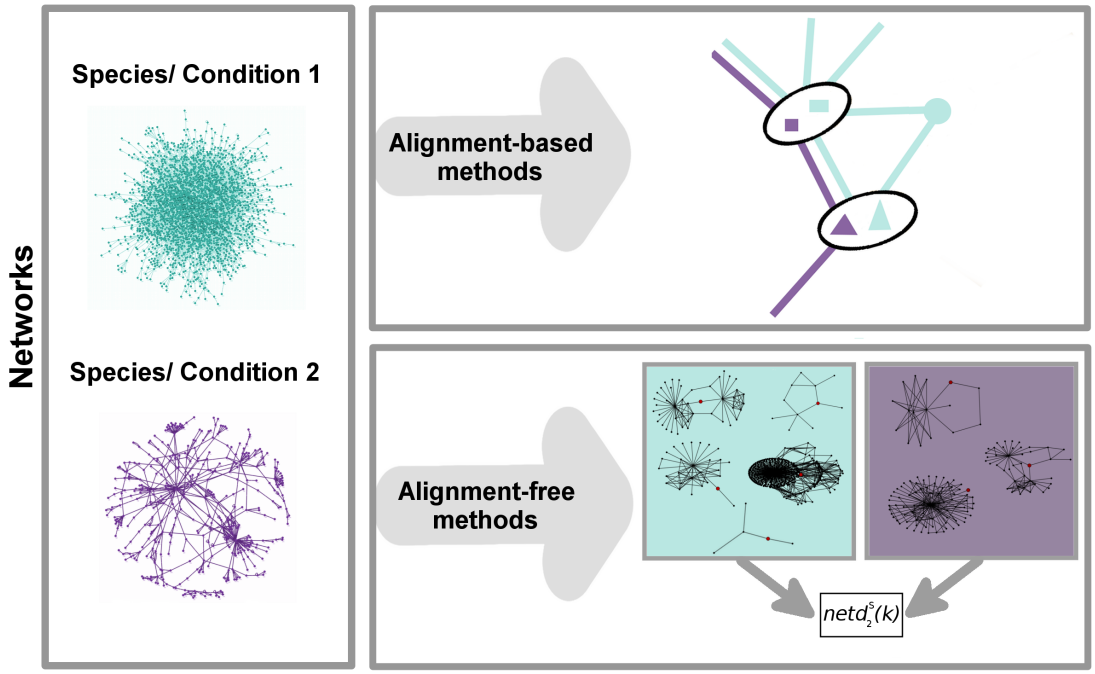


Figure 5.2: Network comparison using alignment *versus* alignment-free methods. *Upper box)* Network alignment methods try to match proteins across the different query networks attending to both their sequence and network topology. This results in a consensus network with the conserved nodes and edges across the networks. *Lower box)* Alignment-free methods do not match proteins and comparison is based on specific features of the networks. Egotif compares networks using subgraph counts in their local neighbourhoods (ego-networks). The comparison yields a single number, $netd_2^S(k)$, reflecting the average neighbourhood similarity between the query networks.

ble overlap between a set of networks relying on sequence homology or other functional proxies to match proteins while making compromises with network topology. An alignment-free network comparison method identifies specific network features and clusters the queries according to these. The alignment-free approach gives less information on the individual proteins than an alignment-based method, but sometimes this may not be required. Above all, alignment-free methods provide an alternative, complementary way of assessing relationships between the different networks. Using subgraph counts, our method would, for instance, compare the triangle counts of the ego-networks in species 1 and 2, and cluster them together according to these. Triangles alone are a limited descriptor of a neighbourhood's topology and our algorithm considers the counts of all k-node induced subgraphs for $k = 3$, 4 and 5. As mentioned in Chapter 1, an induced k-node subgraph includes all edges between the k nodes considered. For instance, in the $k = 3$ count vector, a triangle will have a single count for a triangle and the three two-stars included within it will not be considered. Figure 5.3 shows all subgraphs considered in characterising individual neighbourhoods. These

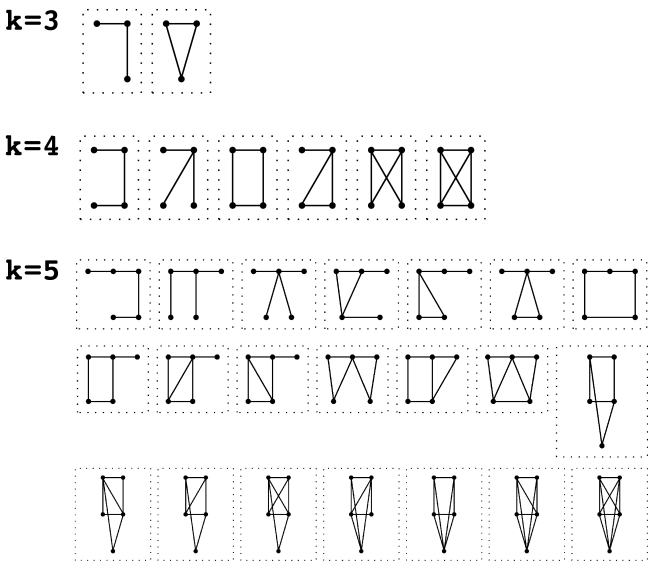


Figure 5.3: Set of all k-node induced subgraphs. The counts of these shapes are involved in formulating the $netD_2^S(k)$ statistic used in our network comparison algorithm.

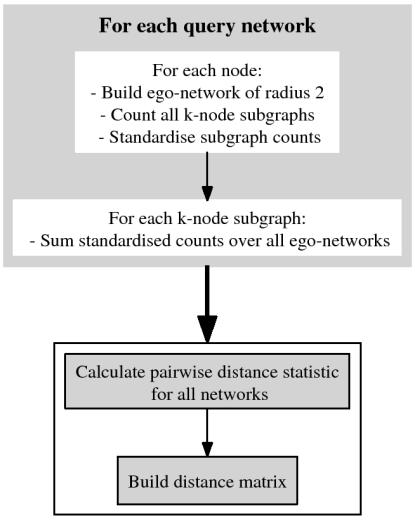


Figure 5.4: Overview of the proposed alignment-free network comparison algorithm.

are split into three vectors according to the number of nodes k of their subgraphs, for instance, a triangle will be the last subgraph included in the counts vector for $k = 3$, the other being the star with 3 nodes. The counting algorithm employed uses a brute force enumeration approach [Przulj, 2007]. Some overlap is only observed between subgraphs of different k, for example, a network consisting solely of a 4-node clique will have one count for its own shape (6th subgraph for $k = 4$ in Figure 5.3), but will also include 4 triangle occurrences (2nd subgraph for $k = 3$ in Figure 5.3). In Chapter 2 we discussed how orbits and subgraph occurrences of the entire network can be used for network comparison, more specifically for comparing to random graph models. The previously considered RGF-distance [Przulj et al., 2004] is simply based on the Euclidean difference of the logarithm of the subgraph occurrences and does not reflect the local heterogeneity within biological networks that we observe.

Figure 5.4 gives a general overview of the algorithm. As previously mentioned, our algorithm starts by extracting for each query network the set of neighbourhoods of all proteins. Here we define the neighbourhood of a protein as a two-step (radius 2) ego-network centred around that protein (see Chapter 4). For each ego-network we then

calculate the number of occurrences of all 3 to 5-nodes induced subgraphs, 29 subgraphs in total (Figure 5.3). Each node is hence associated with a k-nodes subgraph count vector for its corresponding neighbourhood (ego-network with radius 2). An important step of the algorithm, discussed in more detail in the next section, is the standardisation of subgraph counts. Once each neighbourhood of a node has been associated with a count vector for all subgraphs contained in it, we have to standardise these counts according to the size and density of the ego-network.

For each k-node subgraph, we sum the standardised counts of all ego-networks in the query network. The final sum vectors have length 2 for $k = 3$, 6 for $k = 4$ or 21 for $k = 5$ (see Figure 5.3). These sum vectors are then used as input to statistics similar to those of alignment-free sequence comparison methods to calculate a distance matrix between the query networks. This distance reflects the average similarities/ differences in the subgraph counts of their respective ego-networks. No other data besides network data is used as input to the method. Alignment-based methods often require not only sequence similarity scores, but also functionally annotated data to improve protein matching; our method relies only on the information given by an edge list and hence, all types of network data can be used. The devised algorithm is coarse-grained in the sense that it does not provide information on specific proteins or neighbourhoods. It aims to give clear relationships between a set of networks and hence be capable of phylogenetic tree reconstruction. In the next section we describe in detail our attempts at finding an expectation for subgraph counts and how they can be used for standardisation.

5.2.1 Expectation of subgraph counts in neighbourhoods

In statistics we often compare observed values with what is expected given either the best available probabilistic model for the data or resorting to empirical methods to determine the expectation. In the case of alignment-free sequence comparison it was shown that without proper standardisation of word counts, the resulting distances

between sequences will be dominated by single-sequence noise [Lippert et al., 2002]. In this case all we would be measuring is the departure of each sequence from the background, missing the objective of sequence comparison. For network comparison it is therefore reasonable to assume that it is important to achieve a good standardisation of the subgraph counts with respect to the ego-network's size and graph density. A key issue is to be able to calculate the expectation of each subgraph count in a ego-network so that χ^2-like statistics can be devised and the performance of the standardisation assessed. In this section we present some ideas on how this standardisation might be carried out.

We propose a method based on the notion of a gold-standard network. In the context of protein interaction data, this gold-standard network can, for instance, correspond to a high-confidence PIN. In the previous chapters we showed that no simple model gives reasonable results for an accurate estimation of the number of subgraph occurrences of the entire network [Rito et al., 2010]. Although this may not be the case for models of our smaller ego-networks, we propose nonetheless an empirical method through which these occurrences can be estimated. The method requires a gold-standard network and can readily provide approximate mean number of subgraph occurrences per ego-network according to its size and graph density.

In order to calculate the expected number of subgraph occurrences in an ego-network of a given graph density we first create a histogram of all the ego-networks of a gold-standard network. The aim is to discretise the ego-network's graph density space as finely as possible, while preserving in each bin enough ego-networks so that their average subgraph counts can still capture the overall trend of the network. We start by creating 100 equally spaced bins in the graph density interval $[0, 1]$ and then enforce that every bin has at least 5 ego-networks in it. This is done by merging the under-populated bins with the least crowded adjacent bin, or, in the case of a tie, with the bin of lower graph density; the first and last bin are treated last. The bin occupancy, *i.e.*, the number

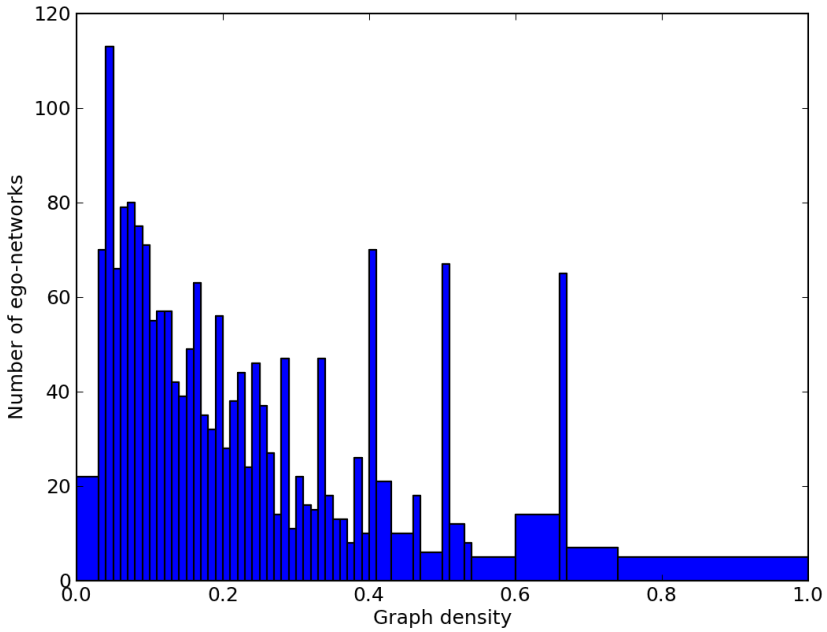


Figure 5.5: Bar plot depicting the graph density binning and the number of ego-networks in each bin for the yeast DIP-Core data set.

of ego-networks in each graph density bin, is then network specific depending on the distribution of ego-networks that compose it and their graph density. Figure 5.5 shows the binning of the yeast DIP-Core data set.

The bars in the plot represent the graph density binning and their height the number of ego-networks (occupancy) in that bin. We can see that most ego-networks in the yeast DIP-Core data have a graph density between 0 and 0.2. Due to the scarcity of ego-networks at high densities several bins were pooled together to satisfy our minimum number of ego-networks requirement.

We then calculate the expectation per ego-network of each subgraph in a given graph density bin. Let w denote a particular induced subgraph with k nodes and N_w its number of occurrences. Let Q represent a gold-standard network with q nodes and hence q ego-networks. Each i ego-network contains n_i nodes, e_i edges and has graph density d_i, with $d_i = \binom{n_i}{2}^{-1}(e_i)$.

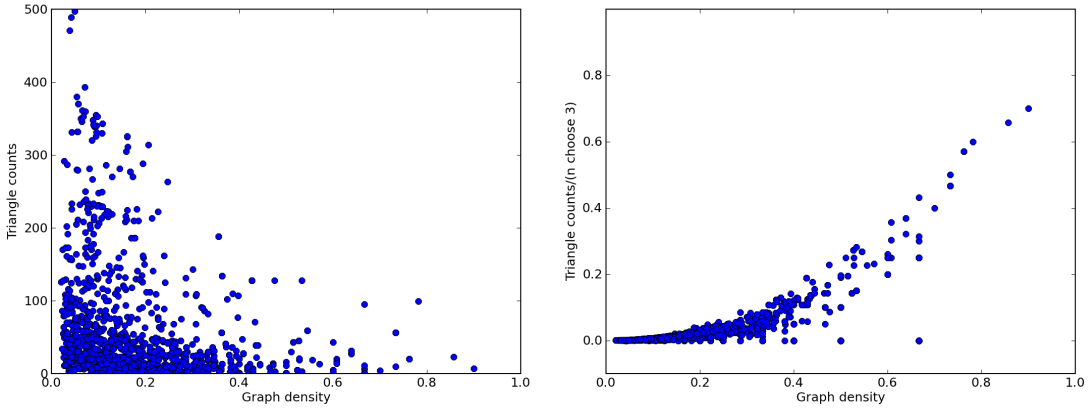


Figure 5.6: (*left*) Plot of the number of triangle occurrences ($N_{\triangle,i}(Q)$) in each ego-network of the yeast DIP-Core network (Q) *versus* their graph density. (*right*) Scaled triangle occurrences of the same ego-networks according to $\frac{N_{\triangle,i}(Q)}{\binom{n_i}{k}}$ with $k = 3$.

For each graph density bin ρ, we scale the occurrences $N_{w,i}(Q)$ in an ego-network i of Q according to the number of possible k-node subgraphs the ego-network can have, which in turn depends on its number of nodes n. The plot of Figure 5.6 shows the effect of this scaling step on triangle counts of the ego-network of the yeast DIP-Core network. The scaling is successful at standardising the counts by the size of the ego-networks. Future refinements of the method should include a study of different possible ways of scaling the counts and their effect on the results obtained.

The average of the scaled $N_{w,i}(Q)$ for all q ego-networks in the graph density bin is then calculated such as

$$E_w(Q,\rho) = \frac{1}{|\{i \in \{1,...,q\} : d_i \approx \rho\}|} \sum_{\substack{i=1...q: \\ d_i \approx \rho}} \frac{N_{w,i}(Q)}{\binom{n_i}{k}},$$

where $d_i \approx \rho$ means that d_i is in the graph density bin ρ.

The value $E_w(Q,\rho)$ of a given subgraph w is calculated per ego-network, due to the $\binom{n_i}{k}$ scaling. The Q and ρ remind us that E_w is specific to a given gold-standard network Q and of a given graph density bin. Figure 5.7 shows a bar plot for the values

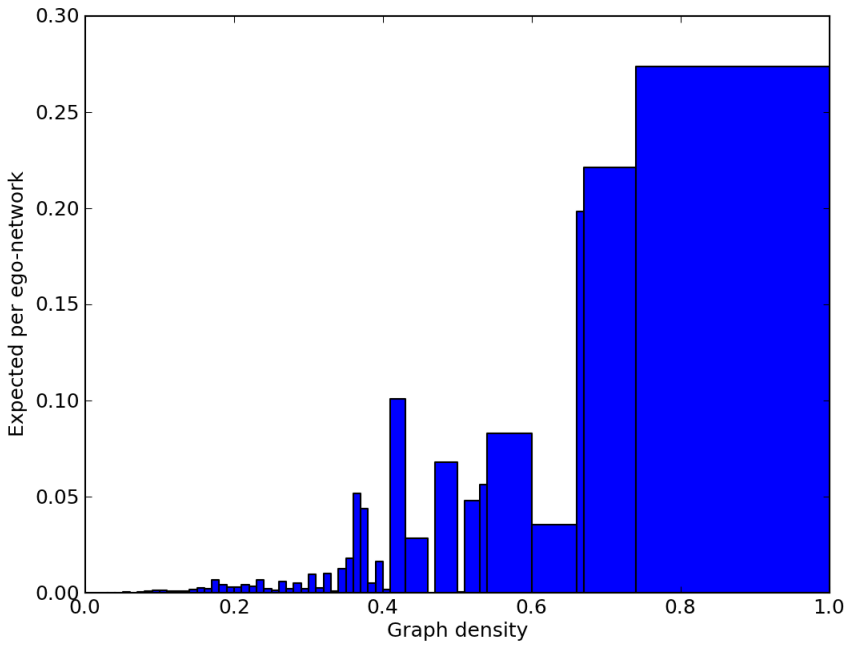


Figure 5.7: Bar plot with the expectation $E_w(Q, \rho)$ of each graph density bin. The $E_w(Q, \rho)$ values are per ego-network for the case of triangles in the yeast DIP-Core data-set.

of E_w for triangles in the DIP-Core data-set. At higher graph densities the expectation increases, which matches our intuition from the raw data (Figure 4.5 C) that dense ego-networks tend to have high triangle counts.

The final expectation for a query ego-network is calculated from E_w and its number of possible k-node subgraphs such that

$$E_p(w) = \binom{s}{k} E_w(Q, \rho),$$

where s is the number of nodes of the ego-network centred around a node p with graph density $d_p \approx \rho$. Subgraph counts can then be standardised by subtracting the respective expected $E_p(w)$ value from the occurrences in each ego-network. Consider a query network G, Figure 5.8 shows the $(Observed - Expected)/\sqrt{Expected}$ or $(N_{w=\triangle,i}(G) - E_p(w))$ divided by $\sqrt{E_p(w)}$ values of triangle $(w = \triangle)$ counts in the ego-networks of the yeast DIP-Core dataset and the yeast DIP using the DIP-Core dataset

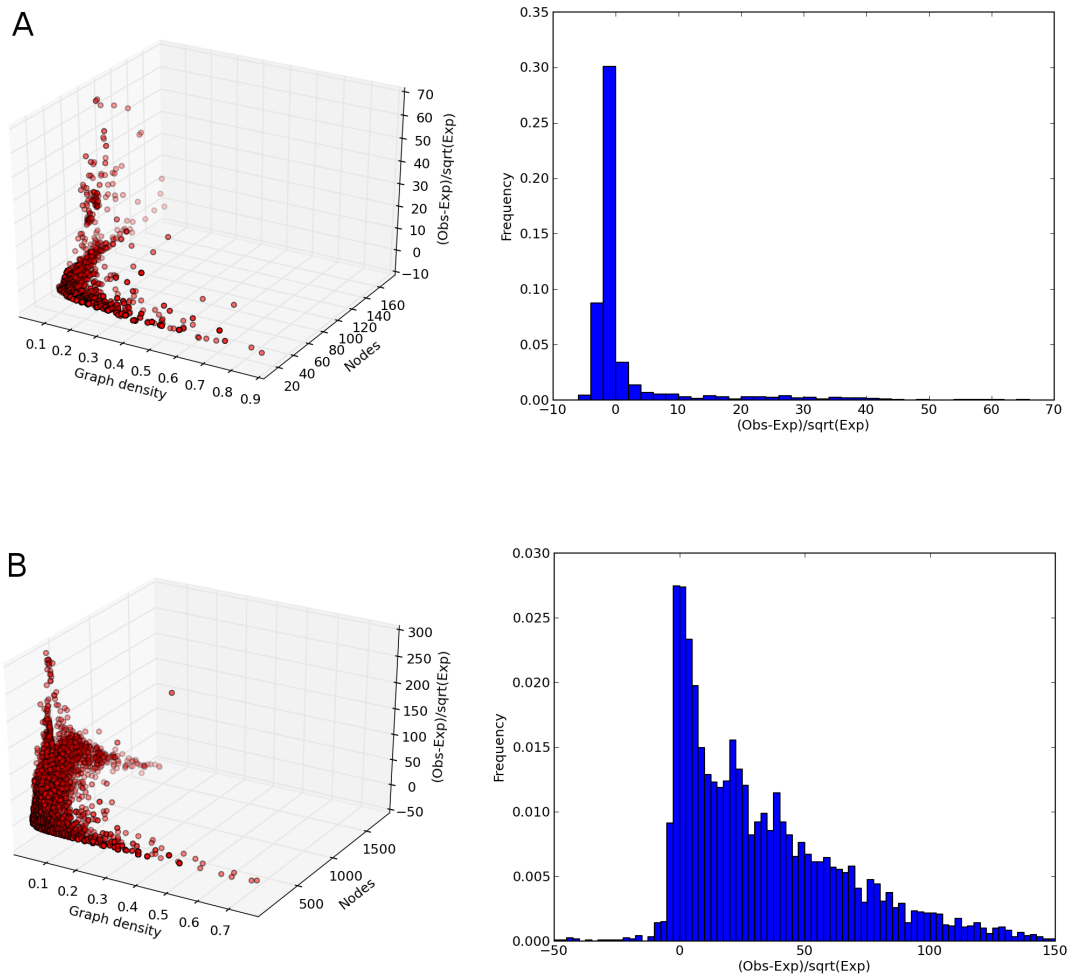


Figure 5.8: Three-dimensional plots of the $(Observed - Expected)/\sqrt{Expected}$ values for the triangle counts of each ego-network and their corresponding histogram for A) yeast DIP-Core using yeast DIP-Core as gold-standard and B) yeast DIP using yeast DIP-Core as gold-standard.

as Q – employed in calculating $E_p(w)$. Heuristically, if we assume that the counts approximately follow a Poisson distribution, then the variance is approximately equal to the mean and the division by $\sqrt{E_p(w)}$ is reasonable to construct the z-statistic.

We observe that some information is lost in the binning process since using yeast DIP-Core as a gold-standard to itself results in many non-zero values of the χ^2-like statistic, the predictions for the complete yeast DIP data-set also seem to be mostly under-estimated. Nonetheless, in both examples, positive, negative and zero values are

observed which indicates that the centring of the data is performing well enough to test our comparison algorithm. Future refinements of this method should probably include improvements in this step, for instance, a smoothing function could be applied to the E_w values across the graph density range, see Figure 5.7.

The expectation of the occurrences of a given subgraph could also be achieved via simulation from a random graph model. Given a full probabilistic random graph model where parametrisation could be readily computed, we could simulate a network for each specific ego-network and use the occurrences of subgraphs in these as our expectation to standardise the counts. This approach is nonetheless computationally more expensive and our attempts using the configuration model showed no significant improvements over our empirical method.

5.2.2 Detailed network comparison algorithm

More formally, the entire method of network comparison using a gold-standard network can be described as follows. As before, let w denote a particular induced subgraph with k nodes and $N_w(G)$ its number of occurrences in a given ego-network of a network G. Now consider a candidate network G with node set $V(G)$. For each p in $V(G)$,

1. Build an ego-network of radius 2 around and including p;

2. Calculate for the p ego-network its number of nodes s, graph density d_p and the subgraph count vector, $N_{w,p}(G)$, for each k, with $k = 3$, 4 and 5;

3. From the previous section we have then an expectation $E_w(Q, \rho)$, with $\rho \approx d_p$, per graph density interval, per ego-network for each of the 29 subgraphs (see Figure 5.3). We now calculate the expectation for this particular ego-network using the previously estimated E_w such that,

$$E_p(N_{w,p}(G)) = \binom{s}{k} E_w(Q, \rho);$$

4. Repeat steps 1-3 for every p in G and define the statistic

$$S_w(G) = \sum_{p \in V(G)} \Big(N_{w,p}(G) - E_p(N_{w,p}(G)) \Big).$$

We can now consider multiple networks and compare them by employing the D_2^S, originally developed for alignment-free comparison of sequences [Reinert et al., 2009]. This can be done directly using the centralised counts S_w of each network. For a different candidate network H, we define three $netD_2^S(k)$ statistics according to the particular subgraphs that are being considered. Let $A(k)$ be the set of all w subgraphs with k number of nodes. Here we consider statistics for $k = 3$, 4 or 5. The centralised counts S_w from G and H are then employed to calculate $netD_2^S(k)$ such that:

$$netD_2^S(k) = \tfrac{1}{M} \sum_{w \in A(k)} \left(\tfrac{S_w(G) S_w(H)}{\sqrt{S_w(G)^2 + S_w(H)^2}} \right)$$

where

$$M = \sum_{w \in A(k)} \left(\tfrac{S_w(G)^2}{\sqrt{S_w(G)^2 + S_w(H)^2}} \right) \sum_{w \in A(k)} \left(\tfrac{S_w(H)^2}{\sqrt{S_w(G)^2 + S_w(H)^2}} \right),$$

so that $netD_2^S(k) \in [-1, 1]$ by the Cauchy-Schwarz inequality. The subgraph occurrences considered by each $netD_2^S(k)$ can be seen in Figure 5.3. The corresponding distance statistic is defined as

$$netd_2^S(k) = \tfrac{1}{2}(1 - netD_2^S(k)) \in [0, 1].$$

This distance can then be used to build the distance matrix for all query networks. The distance matrix establishes relationships between the networks according to the subgraph counts of their respective ego-networks and is a new method for network comparison that does not require any data external to the network data. The next

section describes how this method can be used to cluster networks of different random graph models and form relationships between these.

5.3 *Egotif* can separate different random graph model types

A good method for network comparison should be able to discern between networks of different random graph models and cluster them together according to these models. In our method, as described above, we first build an ego-network around each node of a network and compute its respective subgraph count vector. The vectors are then used to calculate a distance matrix, a symmetric matrix containing the $netd_2^S(k)$ distances for each network pair, for all candidate networks. The distance matrix can be used to cluster the candidate networks and the clustering represented by dendrograms. Here we consider two popular methods for distance analysis: the unweighted pair-group method with arithmetic mean (UPGMA) [Sokal and Michener, 1958] and neighbour-joining (NJ) [Saitou and Nei, 1987].

UPGMA is an heuristic greedy method which creates one cluster per network and sequentially merges the nearest pair of clusters by directly using the distance matrix until only two clusters remain. The branch length within a cluster is simply the original distance between the merged clusters halved. At each step, and to account for merging, the distance matrix is re-calculated. The distance of a newly formed cluster to the remaining ones being the mean distance between all elements of each cluster. UPGMA requires the distance metric to be ultrametric, satisfying the *three point condition*, to produce consistent trees. A metric is generally required to satisfy the triangle inequality which states that given the distances $d(x, z)$ and $d(z, y)$, the distance $d(x, y)$ is at most the sum of the first two; an ultrametric is a stronger version where $d(x, y) \leq max(d(x, z), d(z, y))$. Using UPGMA on non-ultrametric data will still

result in an ultrametric tree, but there is no guarantee of convergence to the correct tree. UPGMA produces rooted trees with all leave nodes at the same distance from the root, a consequence of its averaging algorithm which assumes the rate of change to be constant for all lineages over all time, the so-called '*molecular clock hypothesis*'.

Neighbour-joining is also a step-wise clustering algorithm, but it does not require an ultrametric distance matrix. It starts with each element in a star-like tree and recursively merges the least-distant pairs of elements according to a modified distance matrix Q. Q is constructed to represent not only the distance between two elements, but also their divergence with every other elements. Let $d(i,j)$ represent the distance between the elements i and j, the matrix Q is then calculated for each pair of elements such that

$$Q(i,j) = d(i,j) - (r_i + r_j)/(N-2),$$

where N is the total number of elements, $r_i = \sum_{k=1}^{N} d(i,k)$ and $r_j = \sum_{k=1}^{N} d(j,k)$. When two elements are joined, a common ancestral node replaces them in the tree, whose size is then iteratively reduced until only two nodes remain. The trees produced by NJ are unrooted and unique for a given distance matrix.

First we test our method on simulated networks of various models including the Erdős-Rényi random graph, $G_{n,m}$, model (ER) [Erdős and Rényi, 1960], the geometric 3-dimensional model (GEO) [Penrose, 2003], the stickiness index model (STICKY) [Pržulj and Higham, 2006] and the duplication and divergence model (DD) described by Middendorf et al. [2005]. For all simulated networks the parameters were chosen in order to have the number of nodes and edges match those of the yeast DIP PIN data set, although some models create self-loops and disconnected nodes leading to slight discrepancies between them. Using the yeast DIP PIN we also simulated from the Erdős-Rényi with fixed degree distribution model (ER-DD) [Newman, 2003] and the

Table 5.1: Networks generated by several random graph models to match the yeast DIP PIN. For ER and GEO we also include networks with higher graph density.

Networks	#Nodes	#Edges	Graph Density
Yeast DIP	5208	24914	0.001837
ER1–3	5208	24914	0.001837
ER high	1000	10000	0.020020
ERDD1–3	5208	24914	0.001837
Rand1–3	5208	24914	0.001837
GEO1–3	5204	24914	0.001840
GEO high	1000	5324	0.010658
STICKY1	4653	24853	0.002296
STICKY2	4652	25060	0.002316
STICKY3	4644	25241	0.002341
DD1	4760	23351	0.002061
DD2	4756	23558	0.002083
DD3	4751	23053	0.002043

randomisation algorithm of Maslov and Sneppen [2002] (Rand) which shuffles edges of an input network while preserving its original degree distribution (see Chapter 1 for more information about these models). A summary of the networks considered and some of their characteristics can be found in Table 5.1.

As we rely on subgraph counts to discern differences between networks, one could argue that our method will be biased if we compare networks with very different graph densities. For higher graph densities we expect more edges and higher subgraph counts. To test the effect of graph density we also included the "*ER high*" and "*GEO high*" graphs with different number of nodes and a graph density which is about an order of magnitude higher than that of the current yeast DIP PIN.

We calculated the distances $netd_2^S(k)$, with $k = 3, 4$ and 5, for all networks in Table 5.1 using the yeast DIP core dataset as our gold-standard network and applied both UPGMA and NJ to the resulting distance matrices. We disregard branch lengths when plotting the trees and focus only on their topology. The corresponding trees for the case of $k = 3$ are shown in Figure 5.9.

For both NJ- and UPGMA-based trees we observe a nearly perfect clustering of the networks. NJ sequentially starts by clustering the GEO networks, followed by the ER and DD networks (Figure 5.9(*left*)). Despite the DD being a more sophisticated

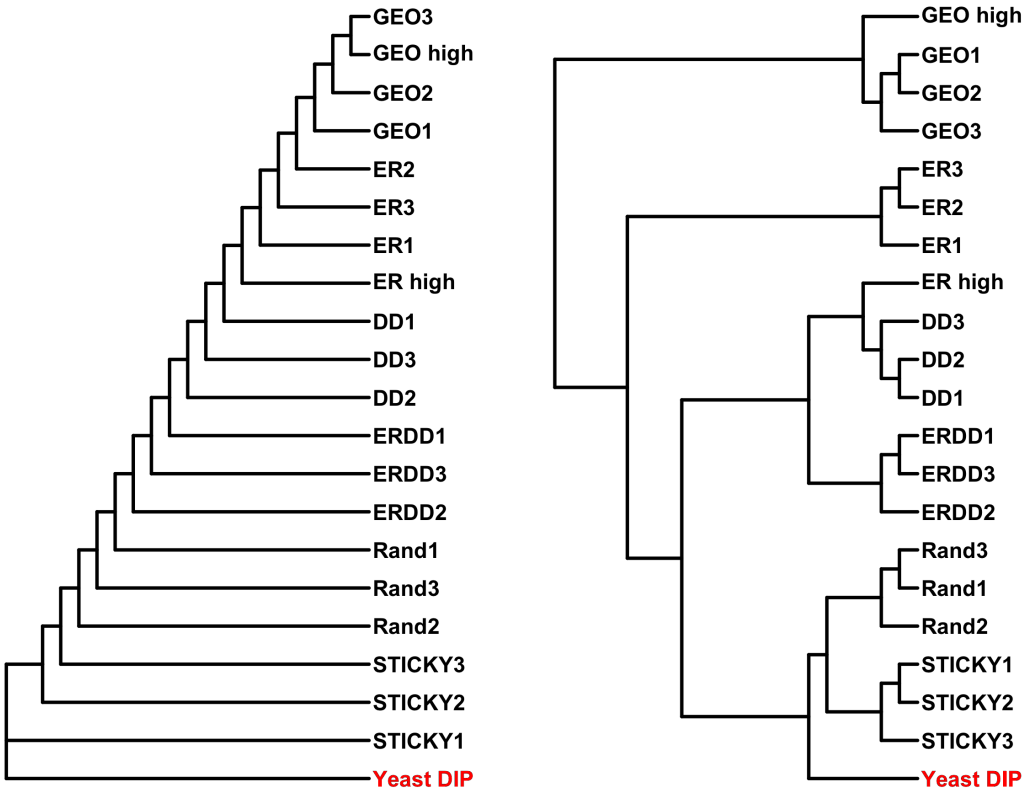


Figure 5.9: (*left*) Phylogenetic tree based on Neighbour-Joining (NJ) of the simulated networks listed in Table 5.1 and the yeast DIP PIN. (*right*) Phylogenetic tree of the same data, but based on the unweighted pair-group method with arithmetic mean (UPGMA). Both trees are based on the $netd_2^S(3)$ distance which considers all 3-node subgraphs.

model for PINs, the yeast PIN is put next to the STICKY, Rand and ERDD models. This may be because these three models require the degree distribution of the PIN as extra information to build their networks. NJ sequentially adds networks to the initial cluster. The sequence in which the networks are added to this cluster is correct in the sense that networks of the same model type are adjacent, but it results in a very "smeared" tree topology. Given that our distance matrix is not ultrametric, it is surprising that UPGMA appears to perform better at clustering the networks into meaningful clades according to their model type (Figure 5.9(*right*)). Here we observe that the Yeast PIN is put as an out-group within the STICKY and Rand clade. GEO

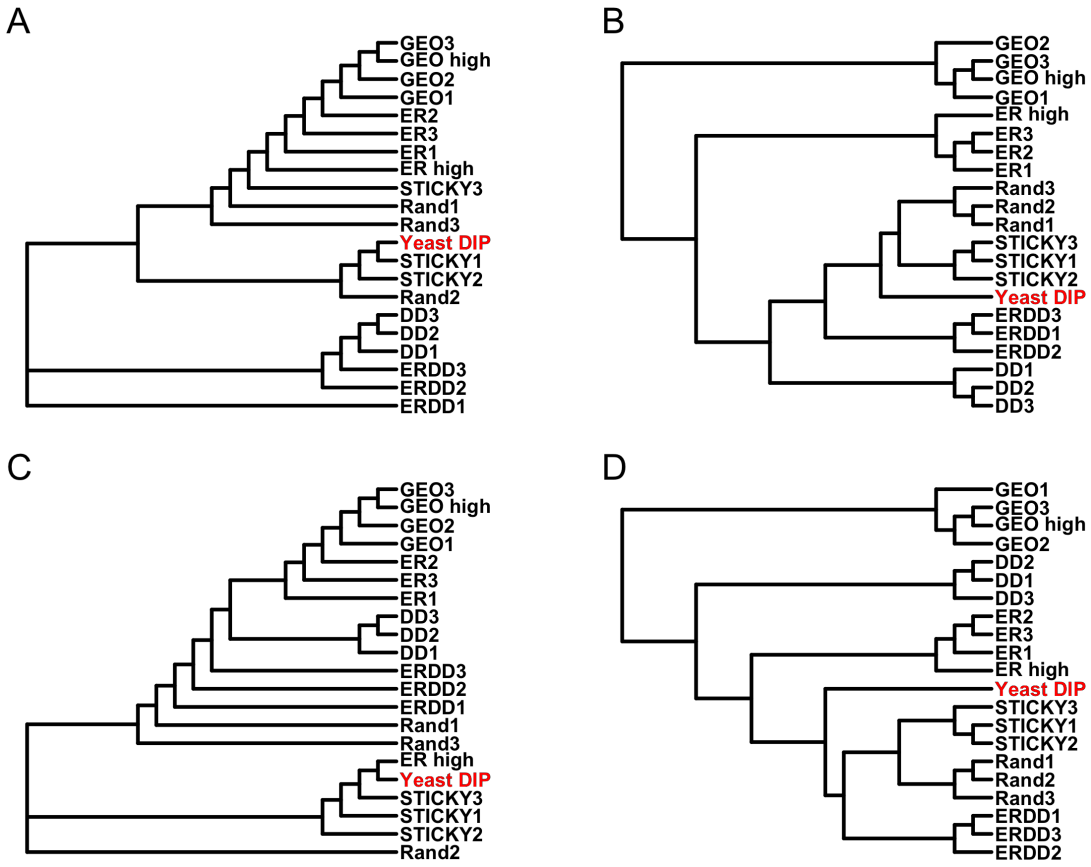


Figure 5.10: Phylogenetic trees based on the distance $netd_2^S(k)$ for $k = 4$, A) and B), and $k = 5$, C) and D). Phylogenetic trees A) and C) are based on Neighbour-Joining (NJ) of the simulated networks listed in Table 5.1 and the yeast DIP PIN. The trees B) and D) are of the same data, but based on the unweighted pair-group method with arithmetic mean (UPGMA).

and ER appear as the most distant models from the PIN. Figure 5.9, based on the distance $netd_2^S(3)$, is essentially considering the counts of only two subgraphs – a 3-node path and a triangle (see Figure 5.3). Nonetheless, a meaningful clustering of the data can already be achieved with our many-to-many ego-networks comparison method. We also consider the cases where $k = 4$ (comprising a total of 6 subgraphs) and $k = 5$ (21 subgraphs) using both NJ and UPGMA. The phylogenetic trees for these are in Figure 5.10.

The general observation that UPGMA is better at forming clades which are coherent

with model type still holds for $k = 4$ and 5, and the clustering according to either of these is generally good. The yeast PIN data set is consistently closer to Rand, STICKY, DD and ERDD (Figure 5.10B) or to STICKY, Rand and ERDD (Figure 5.10D). "ER high", which was misplaced into the DD clade for the $k = 3$ case, is now correctly put next to the other ER networks for both $k = 4$ or 5 (Figure 5.10B and D), although in the NJ $k = 5$ case it appears closer to the Yeast PIN (Figure 5.10C).

To see how the choice of the gold-standard network affects our results, we repeated the experiments using as gold-standard the ER1 and DD1 networks. The UPGMA trees for the case of $k = 5$ are shown in Figure 5.11. In both cases the algorithm correctly forms clades which are coherent by model type, although the relationship between these clades changes slightly.

When using ER1 as gold-standard (Figure 5.11A), the yeast PIN is put next to networks from ERDD and DD and all the others form a separate clade. When using DD1 (Figure 5.11B), the PIN, much like the previous trees using yeast DIP-Core as gold standard, is also in the larger clade of Rand, STICKY and ERDD; ER and GEO form separate clades.

We interpret these changes in clade positioning in terms of the sensitivity of the distance measure. As discussed before, the point of a gold-standard network is to calculate the expectation of a particular subgraph count per ego-network in a given graph density bin. These expectation values are then used to calculate final expectations for a particular ego-network (according to its number of nodes) which are subtracted from the observed values. The expectation values obtained from one particular gold-standard are fixed for a given run across all networks and only the final, ego-network specific, expectation values change. As such, there are two requirements for a good gold-standard network: 1) it should possess enough ego-networks to populate all graph density bins (Figure 5.5 shows that this is harder for higher graph densities, and bins often have to be pooled in this region) and 2) the particular dependency of the subgraph counts of

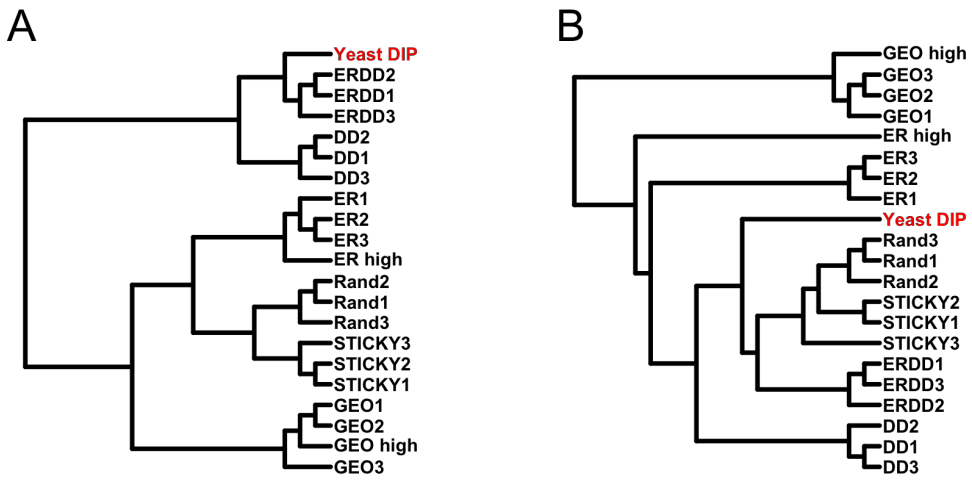


Figure 5.11: Phylogenetic trees of the networks of Table 5.1 using as gold-standard A) the ER1 network and B) the DD1 network. Both trees are for the $k = 5$ case and were produced using UPGMA.

the ego-networks in the gold-standard network with their number of nodes and edges should be as close as possible to that of the candidate networks (for instance, the plots of Figure 4.5A and Figure 5.12 suggest very different trends for the case of triangles). The latter is not necessary, but is desirable as it will ensure maximum sensitivity. This is of importance since the landscapes of ego-networks can be very different from network to network. For instance, Figure 5.12 shows all the ego-networks with more than 5 nodes in ER1 and DD1 in a plot similar to the ones of Figure 4.5, where we plot the graph density, number of nodes and the number of triangles of ego-networks.

Although DD1 (Figure 5.12 ($right$)) and yeast DIP-Core (Figure 4.5C) have similar shapes, the number of triangles in DD1 is about an order of magnitude higher. The ER1 network (Figure 5.12 ($left$)) has a very different, limited, landscape where ego-networks have at most 6 triangles and 200 nodes, with only two ego-networks reaching graph densities above 0.2. Using this network as gold-standard would mean that the expectation per ego-network for graph densities higher than 0.2 would be calculated

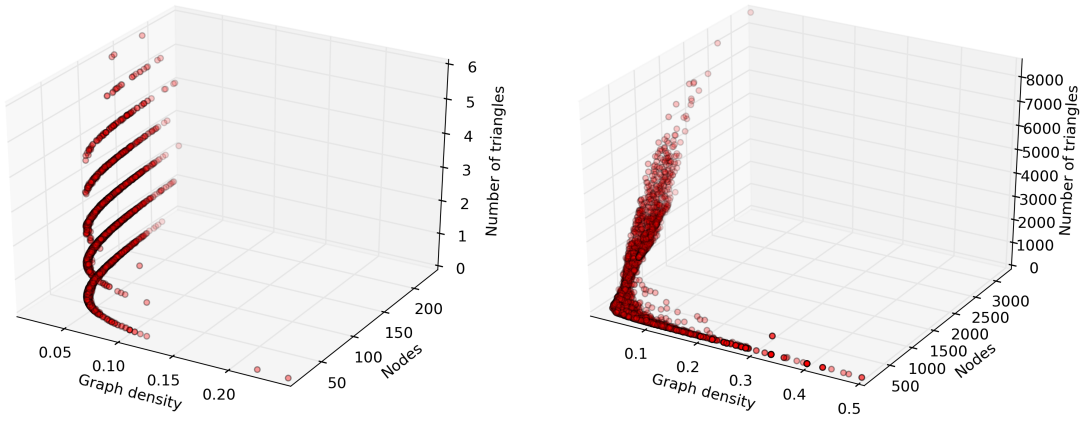


Figure 5.12: Three-dimensional plots of the graph density, number of nodes and triangles of all ego-networks with more than 5 nodes for the networks (*left*) ER1 and (*right*) DD1. Each dot represents one ego-network.

based on the subgraph counts of the last bin which will span most of the graph density space. For a more concrete example, consider the standardisation of the triangle subgraph counts in an ego-network with 4 nodes, 5 edges ($\rho = 0.83$) and 2 triangles. Using the yeast DIP core as gold-standard this ego-network would fall in the last graph density bin from 0.74 to 1; with an expectation of 0.201 calculated based on the 5 ego-networks in DIP-core that populate this bin. The final expectation in this case would then be $\binom{4}{3} * 0.201 = 0.8$ and to the S_w sum would be added $2 - 0.8 = 1.2$. If DD1 was being used as gold-standard, we would take the expected value, 0.44, of the last bin which corresponds to a graph density from 0.41-1 combining the information of 9 ego-networks in DD1. The final expectation would be 1.76 and the S_w sum would increase by $2 - 1.76 = 0.24$. For this concrete case, DD1 was more effective in the standardisation, although overall this assessment would be dependent on all ego-networks of the particular candidate network. Moreover this would also be dependent on the set of candidates and ideally the gold-standard should achieve a good standardisation for all.

Table 5.2: Network summary statistics of PIN data in DIP and BioGRID.

		DIP				BioGRID physical			
Species	# Genes	Nodes	Edges	Coverage	ρ*1000	Nodes	Edges	Coverage	ρ*1000
Homo sapiens (Hsap)	21224	3090	3750	14.56	0.79	14334	64241	67.54	0.63
Mus musculus (Mmus)	21638	1024	1058	4.73	2.02	4571	7780	21.12	0.74
Drosophila melanogaster(Dmel)	13917	7565	22800	54.35	0.80	7975	34630	57.30	1.09
Caenorhabditis elegans (Cele)	20517	2672	3995	13.02	1.12	2900	4594	14.13	1.09
Saccharomyces cerevisiae (Scer)	6692	5078	22103	86.22	1.71	5987	68509	89.46	3.82
Schizosaccharomyces pombe (Spom)	4929	-	-	-	-	1788	3701	36.28	2.32
Arabidopsis thaliana (Atal)	17000	-	-	-	-	5645	12388	33.21	0.78
Escherichia coli (Ecoli)	4377	2968	11604	67.81	2.64	-	-	-	-
Helicobacter pylori (Hpyl)	1589	714	1361	44.93	5.35	-	-	-	-

Sources : for *Hsap*, *Mmus*, *Dmel*, *Cele* and *Scer* we used the "known genes" information at the Ensembl project website (these numbers correspond to Ensembl genes for which at least one known transcript has been annotated); for the rest of the species, the following web page was used `http://users.rcn.com/jkimball.ma.ultranet/BiologyPages/G/GenomeSizes.html`.

5.4 Phylogenies from protein interaction data

The averaging nature of our method, which instead of trying to obtain one-to-one neighbourhood matches between the different networks has a many-to-many approach, provides a measure of similarity between networks which is independent of sequence homology. This approach should also be more robust to noise and data incompleteness than homology-based methods, since our only assumption is that similar networks will share on average more topologically similar neighbourhoods. PIN data is known to be noisy [Huang et al., 2007; Venkatesan et al., 2009; von Mering et al., 2002] and the amount of available data is just now becoming sufficient for organism-wide comparisons making it a challenging test set for our method. Obtaining phylogenies of organisms or of cell-states based on network data would provide a biological perspective on these which would be both new and completely free from sequence data. This would further complement phylogenies derived from phenotypic traits with another type of molecular data.

Table 5.2 summarises the size and coverage of the currently existing PIN data. Only species having at least 500 physical interactions were considered, according to DIP [Salwinski et al., 2004] (2012-02-28 data set) and BioGRID [Breitkreutz et al., 2007] (v3.1.88) databases. Coverage is here a rough estimate of how many proteins

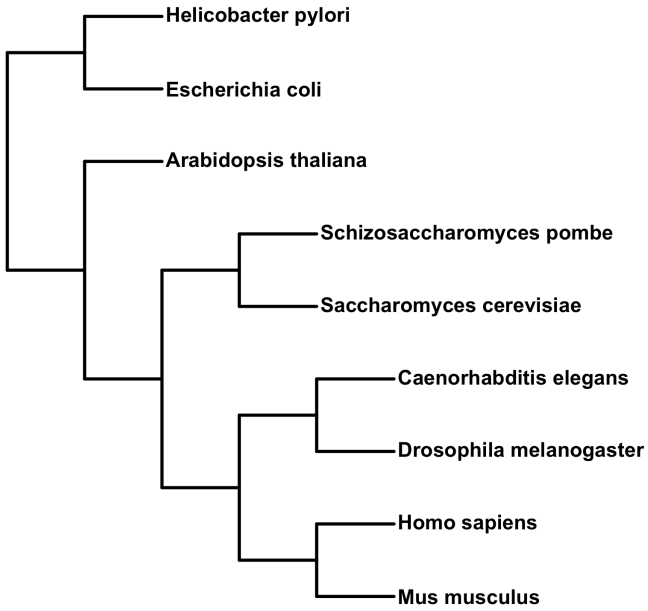


Figure 5.13: Phylogenetic tree of the species considered based on NCBI taxonomy database.

have been probed for interactions given the expected proteome of the organism. We define it as a percentage by taking the number of proteins (nodes of the network) divided by the estimated number of genes in the genome of the organism at hand. The currently accepted phylogeny between the species of Table 5.2 is depicted in Figure 5.13. The tree is based on the NCBI taxonomy database, which incorporates a variety of phylogenetic resources including molecular and morphological characters.

We analyse the PIN data of these species with our alignment-free network comparison method with the aim of understanding the resulting clustering in light of NCBI phylogeny and the characteristics of their PINs. For simplicity we only present UPGMA-based trees, since these were better than NJ-based trees at producing meaningful clades for the network models. As yeast (Scer) is the organism for which most data is available we consistently use here the high-confidence yeast DIP-Core data set as the gold-standard network required by the method. We also included an ER graph

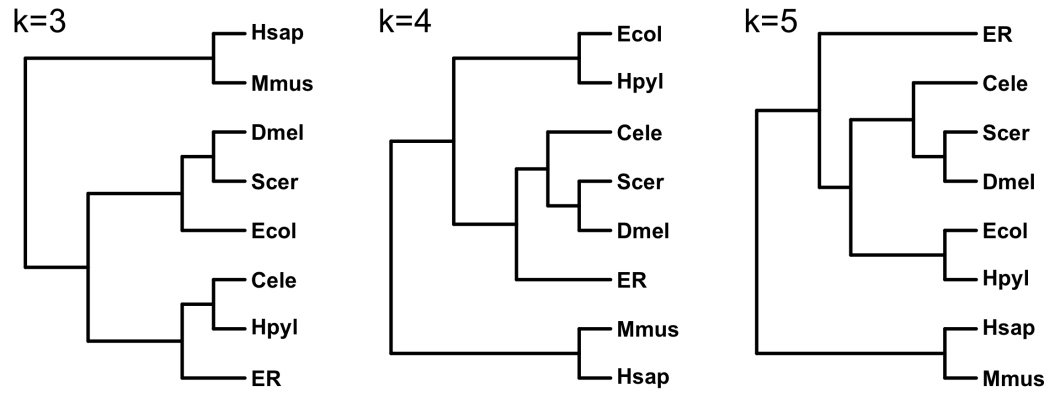


Figure 5.14: UPGMA-based phylogenetic trees of all species with > 500 physical interactions in the DIP database. The trees are based on the distance $netd_2^S(k)$ for $k = 3$, 4 and 5, respectively.

with 2000 nodes and 2000 edges in the trees as an out-group. Figure 5.14 shows the tree corresponding to all species in the DIP database. We observe that human (Hsap) and mouse (Mmus) are consistently put together for all values of k despite their differences in number of nodes and edges, coverage and graph density. For both $k = 4$ and $k = 5$ Ecoli and Hpyl are correctly put together and fly (Dmel), worm (Cele) and yeast (Scer) form a coherent cluster. In $k = 5$ human and mouse are further away from all other PINs than the chosen out-group. The $k = 3$ case shows many inconsistencies relative to the NCBI tree which is not unexpected given the results on the network models, which show that only two subgraph counts results in a distance which lacks discriminatory power.

Although we expect the method to be relatively robust to data incompleteness, it should still require a minimum of complete neighbourhoods (in terms of both nodes and edges) to be able to correctly estimate distances. To test this we rebuild the tree considering only species with a coverage greater than 10% (Figure 5.15). As mentioned, we define coverage as the fraction of the proteins in the network out of the total number of known genes for that species. In the 10% coverage case, we still observe human (Hsap) further away from the other PINs than an ER random graph. Fly is also preferentially

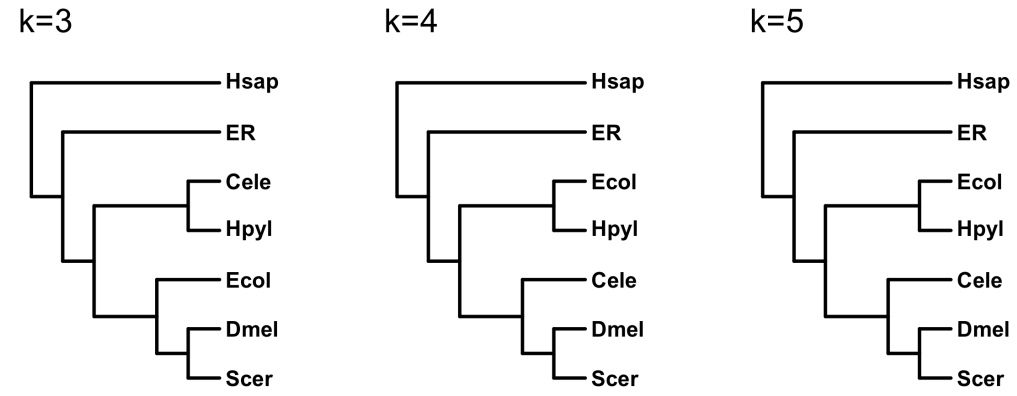


Figure 5.15: UPGMA-based phylogenetic trees of species with a genome coverage $> 10\%$ in the DIP database. The trees are based on the distance $netd_2^S(k)$ for $k = 3$, 4 and 5, respectively.

put together with yeast (Scer) than worm (Cele).

We then increase our data requirement criterium for 15% coverage (Figure 5.16). This results in a loss of 3 species due to data incompleteness. However, out of the 236 possible rooted trees with 5 leaves [Felsenstein, 1978], the correct clustering is obtained with fly (Dmel) next to yeast (Scer) and Hpyl next to Ecoli. It is important to note that the clustering is not following any trivial indicator we can find, based on coverage alone, Ecol would be next to Scer in Figure 5.16; based on number of nodes or edges alone human (Hsap) would be much closer to Ecol and Cele than Mmus (Figure 5.14); also Dmel and Hsap do not cluster based on graph density, neither do the high density data sets such as Hpyl, Mmus and Ecoli.

Part of the incompleteness issue could be solved by combining different databases, but these have different curation protocols and hence different degrees of quality which may introduce biases in the results that are hard to control for. Nonetheless, we tried including HPRD [Keshava Prasad et al., 2009], a manually-curated database for human protein interaction data. This network, "HPRD", can be added to our set of DIP networks with $> 15\%$ coverage. HPRD includes 9,223 nodes and 36,631 edges ($\rho * 1000 = 0.86$), and hence has much higher coverage than the excluded DIP one,

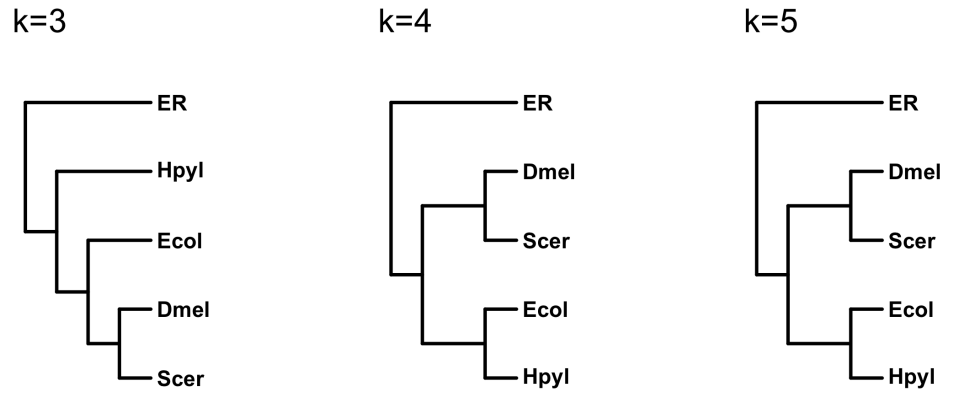


Figure 5.16: UPGMA-based phylogenetic trees of species with a genome coverage $> 15\%$ in the DIP database. The trees are based on the distance $netd_2^S(k)$ for $k = 3$, 4 and 5, respectively.

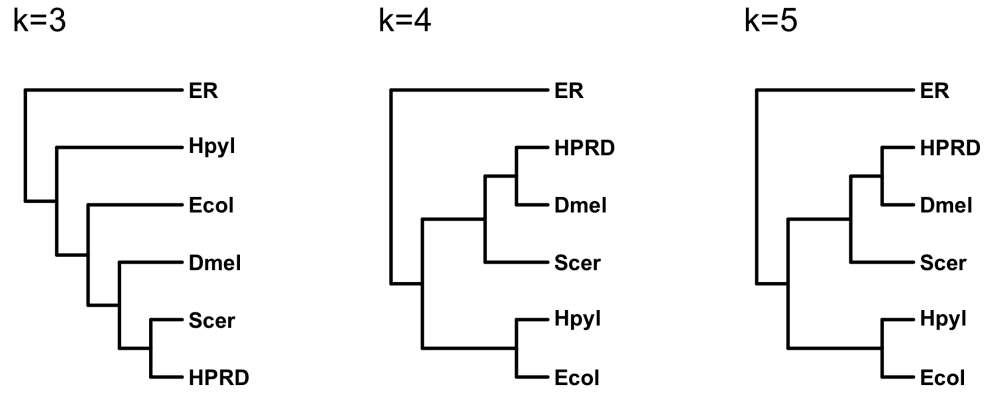


Figure 5.17: UPGMA-based phylogenetic trees of species with a genome coverage $> 15\%$ in the DIP database and the HPRD human network. The trees are based on the distance $netd_2^S(k)$ for $k = 3$, 4 and 5, respectively.

43.92% *versus* 14.71%. Figure 5.17 shows that this dataset is put closer to Dmel and Scer, as in the NCBI tree.

These results are also seem to be robust to our choice of gold-standard. Figure 5.18 shows the same data as Figure 5.17 but using a three-dimensional GEO model with 5000 nodes and 24369 edges instead of the yeast DIP-Core network. Although the clades are less defined, the networks are sequentially clustered in the correct order. Using the ER outgroup with 2000 nodes and 2000 edges as the gold-standard results in the same clustering as Figure 5.17.

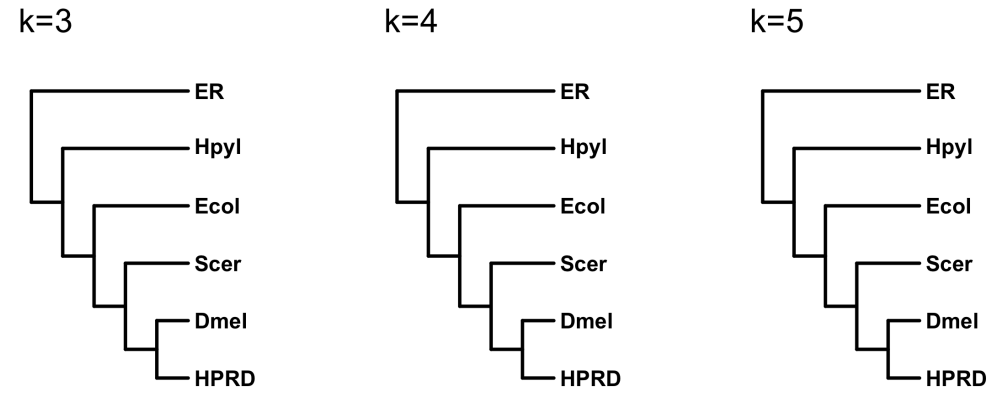


Figure 5.18: UPGMA-based phylogenetic trees of species with a genome coverage $> 15\%$ in the DIP database and the HPRD human network. The gold-standard network used was a GEO3D model with 5000 nodes and 24369 edges. The trees are based on the distance $netd_2^S(k)$ for $k = 3$, 4 and 5, respectively.

Table 5.3: PIN data from BioGRID separated by experiment type.

	BioGRID Two-hybrid screening				BioGRID Affinity purification/ MS			
Species	Nodes	Edges	Coverage	ρ*1000	Nodes	Edges	Coverage	ρ*1000
hsap	6572	15901	31.30	0.74	9881	26316	47.05	0.54
mmus	1175	1508	5.11	2.19	2335	3573	10.15	1.31
dmel	7250	23427	25.89	0.89	2346	10814	8.38	3.93
cele	2830	4453	13.02	1.11	24	49	0.11	177.54
scer	3472	8079	60.17	1.34	4479	39033	77.63	3.89
spom	513	537	10.41	4.09	1051	2030	21.32	3.68
atal	4314	8950	25.38	0.96	1421	2245	8.36	2.23
ecoli	-	-	-	-	-	-	-	-
hpyl	-	-	-	-	-	-	-	-

We have also run our method on networks from BioGRID, another general interactions depository. These networks are larger than those in DIP and are expected to be noisier due to the inclusion of more high-throughput data, especially two-hybrid screening derived interactions. As we saw in previous chapters, the experimental technique behind the generation of the data has a by-design influence on the topology of the resulting network which may affect our results. For this reason we separate the BioGRID data into two main categories, according to the experimental technique used: two-hybrid screening and affinity purification followed by mass spectrometry. Some characteristics of the resulting networks are summarised in Table 5.3.

We first focus on the BioGRID networks resulting from two-hybrid screening. Fig-

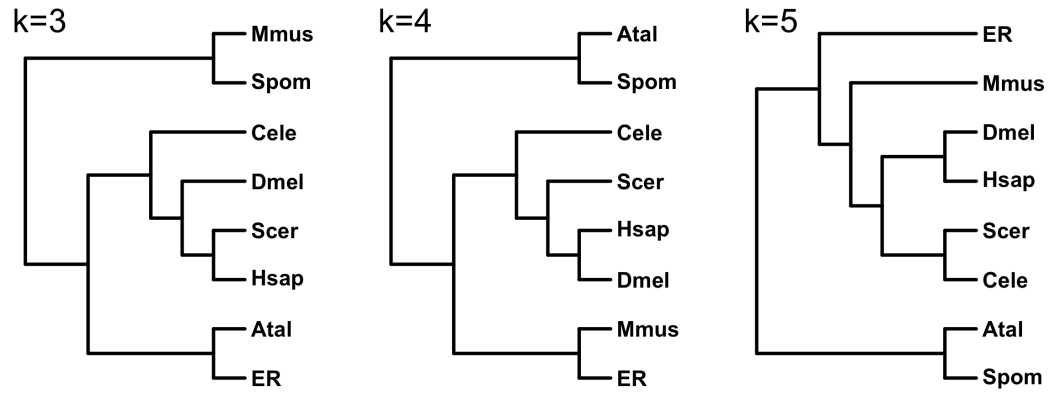


Figure 5.19: UPGMA-based phylogenetic trees of all species in the BioGRID database filtered for two-hybrid screening-based interactions only. The trees are based on the distance $netd_2^S(k)$ for $k = 3$, 4 and 5, respectively.

ure 5.19 shows the dendrograms obtained for all species considered. Unlike DIP, the clustering of species is far poorer, as judged by the NCBI tree. Hsap, Dmel, Scer and Cele are consistently put together in the same clade. Mmus, Atal and Spom PINs appear to be very different from the others and always lie in the extremities of the tree next to the outgroup.

To test if this is caused by lack of data we adopted the same approach as in DIP and only considered networks with a coverage higher than 15%. The trees are represented in Figure 5.20. The clustering in this case appears more sensible and, for $k = 4$ and $k = 5$, Hsap is consistently put next to Dmel, followed by Scer and Atal. Curiously, the only plant PIN (Atal) seems to be further away from the others than these are from an ER random graph.

Mass Spectrometry (MS) based data is fundamentally different since proteins are co-purified with a bait protein and a whole complex may be detected. Individual interactions have to be decided *a posteriori* and it is often not clear when an interaction is a direct bait-prey one or an indirect prey-prey. The trees derived from these data are shown in Figure 5.21.

For the MS-based data the clustering is far worse than our previous results. In the

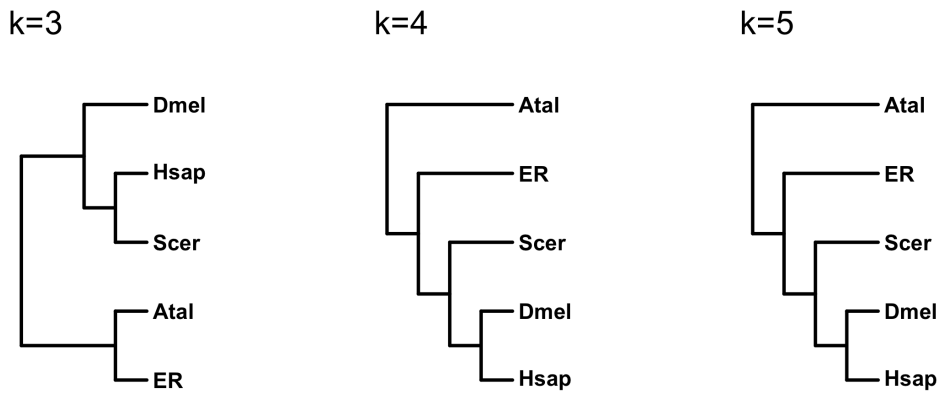


Figure 5.20: UPGMA-based phylogenetic trees of all species with a genome coverage > 15% in the BioGRID database filtered for two-hybrid screening-based interactions only. The trees are based on the distance $netd_2^S(k)$ for $k = 3$, 4 and 5, respectively.

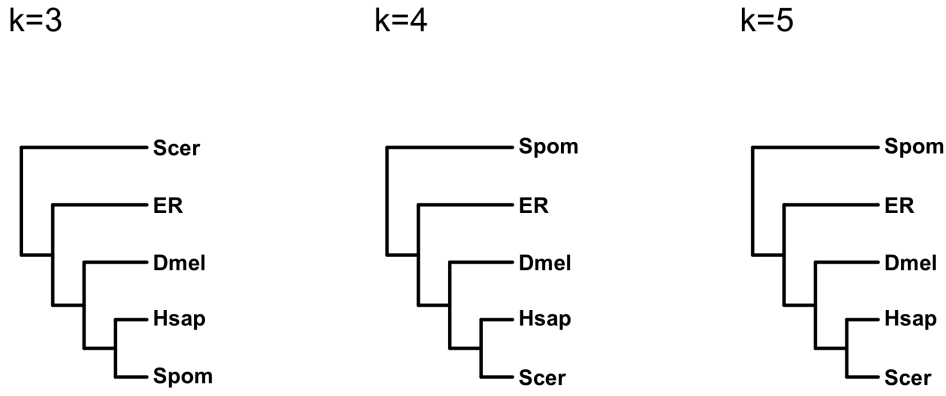


Figure 5.21: UPGMA-based phylogenetic trees of all species with a genome coverage > 15% in the BioGRID database filtered for affinity purification/ mass spectrometry-based interactions only. The trees are based on the distance $netd_2^S(k)$ for $k = 3$, 4 and 5, respectively.

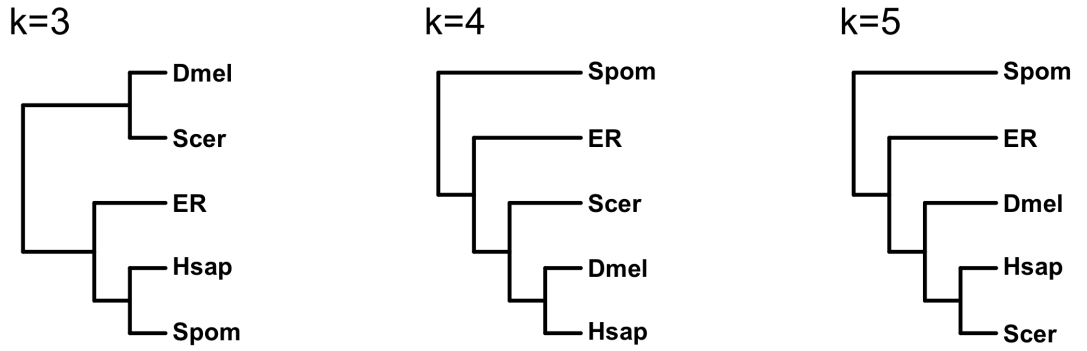


Figure 5.22: UPGMA-based phylogenetic trees of all species with a genome coverage > 15% in the BioGRID database filtered for affinity purification/ mass spectrometry-based interactions only. The gold-standard used here was *not* the yeast DIP-Core data set but the yeast BioGRID MS data. The trees are based on the distance $netd_2^S(k)$ for $k = 3$, 4 and 5, respectively.

case of $k = 4$ and $k = 5$, Hsap is put with Scer. The Spom dataset appears to be very different from all others and stands further away from the rest of the PINs than the ER outgroup chosen. We note however that these results were obtained using the yeast DIP-Core PIN as gold-standard, which includes many interactions derived from Y2H experiments. When we use the yeast BioGRID MS data set as gold-standard, the correct clustering is obtained for $k = 4$ as can be seen in Figure 5.22. This finding stresses the importance of choosing the a sensible gold-standard network to obtain meaningful comparisons.

Curiously, a NJ-based tree of the data using the yeast DIP-Core as gold-standard seems to perform better at representing the data that its equivalent UPGMA-based tree, see Figure 5.23. For instance, in case of $k = 4$ Hsap is correctly put with Dmel; Spom is still close to an ER graph than Scer. This suggests that the choice of the tree construction method may also be playing a role at clustering the species correctly and further studies are needed to investigate this dependency.

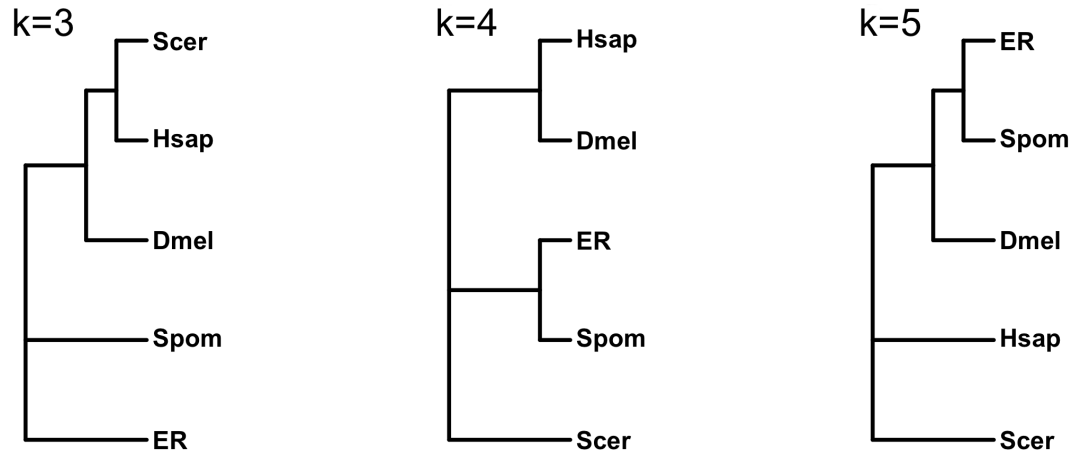


Figure 5.23: NJ-based phylogenetic trees of all species with a genome coverage $> 15\%$ in the BioGRID database filtered for affinity purification/ mass spectrometry-based interactions only. The trees are based on the distance $netd_2^S(k)$ for $k = 3$, 4 and 5, respectively.

5.5 Conclusions

Overall our results suggest that current PIN data is abundant enough to derive correct relationships between species. From PINs with a genome coverage of at least 15%, our method is able to deduce correct phylogenies between species without resorting to sequence homology. It assumes that species that are more related will on average share more neighbourhoods which are topologically similar. We do this without requiring exact or similar one-to-one matches, but by adopting a many-to-many approach which compares neighbourhoods in a given graph density region. Our approach is more robust to noise and to the incompleteness of the data than standard network alignment techniques. It provides an additional way to compare networks which is not based on exact topological matches or compromises with sequence homology. The results show that there is phylogenetic information in PINs without additional biological data.

It is tempting to assign a biological interpretation to the algorithm: a given biological function is normally credited to a community of interacting proteins which act together to perform that function in the cell. Closely related species will have, on aver-

age, more of these communities in common. Our method appears to be detecting this phenomenon by discretising the network space into ego-networks.

The results presented are but a proof-of-principle of this concept raising many questions whose answer is beyond the scope of this thesis, remaining as future work. These include questions on the branch lengths, which were ignored here. Do the distances between branches carry any biological meaning? And if so, what is the best tree building method to show them? In most of the trees presented, we observed big differences in the branch lengths between the different clades which will present challenges in standardisation and interpretation. For simplicity, our distance measures were confined to subgraphs of order k for $k = 3$, 4 and 5. One could imagine though that the best measure will be achieved for a specific combination of subgraphs or for a specific weighting of these, much like the usage of k-word statistics for sequence comparison.

The use of a gold-standard network could also be changed by adopting a network model for ego-networks which would approximately reproduce the subgraph counts found. As we discretise the ego-network landscape in terms of graph density, different random graph models could be used in different regions. This would result in smaller S_w sums and a more accurate comparison although this would probably still be highly variable with the type of candidate networks.

Finally, the method can be applied to any group of networks whose clustering can (ideally) result in meaningful relationships between them. For instance, social networks where the type of neighbourhoods found in facebook-derived data *versus* offline friendship data can be quite different and may vary with the popularity, age and other features of the central node (ego). Another case would be to investigate whether other types of biological networks such as transcription or genetic interaction networks also cluster in a phylogenetically meaningful fashion.

Chapter 6

Conclusions and future directions

Looking at living systems from a genome-wide perspective is becoming more and more common. Typical experiments now involve probing the abundance and interactions of multiple molecules at a time. One example of such data is protein-protein interaction networks (PINs). These are large sparse graphs which, despite still being incomplete, start to form a comprehensive body of data from which inferences on biological systems can be made.

In this thesis we explored two major topics in the study of biological networks: network comparison; and the connection between network topology and characteristics of proteins. This chapter summarises the main conclusions of the thesis and discusses the limitations of our results as well as possible directions for future research. The underlying theme throughout the dissertation is the use of small subgraph counts, as these are a sensitive description of the network topology and are conjectured to relate to biological functional modules.

As network data becomes more available, we will need fast algorithms to compare these graphs and, ideally, cluster them in a biologically meaningful way. Such algorithms are currently not available. One limitation is that there is no good random graph model capable of reproducing the network architectures we observe in PINs.

In Chapter 2 we analyse two popular measures: GDDA and RGF. These scores,

based on subgraph (or orbit) counts, are used to compare and assess the fit of random graph models to PINs. We find that network comparison based on simple distance measures of subgraph counts is probably too naive. When networks of the same model type are compared with each other the scores show high instability. The typical score is different for each model and has a non-monotone dependency with the number of nodes and edges of the networks, making network comparison comparison based on RGF or GDDA very difficult to interpret. We propose a new method to ameliorate graph comparisons using GDDA based on non-parametric statistics. We find that no model tested fits to current PINs. In the same chapter we also use principal component analysis to show most models fail at replicating the counts of small subgraphs found in PINs, even for triangles (the highest order clique after the edge itself).

GDDA and RGF score instability may be related to the fact that the networks are near thresholds for the appearance of subgraphs. Using simulated networks of the Erdős-Rényi (ER) and geometric (GEO) random graph models we observe that the region of greatest instability coincides with the same graph density region at which subgraphs are expected to appear. PINs also occur in this graph density region and therefore we hypothesise that they are at the threshold for subgraph appearance, rendering them both robust and efficient. We bring forward the concept of liminal networks to characterise PINs. Here, liminality does not only imply being at a given graph density but, given an underlying network growth model, it entails being in the graph density region where more complex small subgraphs start to appear. The subgraph counts of networks around this threshold region should approximately follow a Poisson distribution.

A valid self-critique, already hinted at in Chapter 2, is that the graph density threshold values for subgraph appearances used as proxy for the threshold values in PINs were from the ER and GEO models despite these not representing PINs in terms of subgraph counts. As currently no good model exists for PINs, we cannot confirm the

threshold conjecture. Future attempts may involve new and more complex models such as the domain-based gene duplication model [Gibson and Goldberg, 2011] or the crystal growth model [Kim and Marcotte, 2008]. Even if these models are not mathematically well understood and formulæ for the calculation of subgraph appearance thresholds do not exist, an alternative method for empirical detection of thresholds may be possible. One such method could consist in testing the distribution fit of subgraph counts; we expect these to be quite different below (approximately Poisson), around, and above (approximately Normal) the threshold region for subgraph appearance.

We reason that the lack of good models for PINs is likely to reflect our ignorance on how the cell organises and selects interactions. Thus, we search for correlations between network structure (nodes, high-degree nodes, edges, triangles and neighbourhoods) and biological characteristics of proteins (protein age and structural class).

In Chapter 3 we presented an in-depth analysis of protein age and PINs. We devised a method that analyses the occurrences of age-dependent patterns in edges and triangles and assesses their significance based on what we would expect from chance alone given the age-distribution in nodes and edges we observe. We found that subgraphs in PINs have heterogeneous protein age patterns. These cannot be matched by gene duplication models. Previous studies support a network architecture where proteins of the same age tend to interact with each other. We find that only interactions between Old proteins are consistently over-represented and these may be driving the claim for homophily. When we expand our analysis to neighbourhoods (two-step ego-networks, see Chapter 4) we find compatible results. Old proteins are mostly associated with larger neighbourhoods which have over twice the number of triangles within them than the neighbourhoods of Young proteins. These complex neighbourhoods are also predominately centred around proteins belonging to the "all beta" or "all alpha" SCOP structural class.

High-degree proteins do not typically display star-like connectivities as they are

often portrayed in the literature, but display high clustering coefficients, retaining the majority of triangles found in PINs. Overall these results point to PINs being highly heterogeneous; this finding is confirmed in Chapter 4 with the study of ego-networks. The method applied here can also be easily extended to other subgraphs and other explanatory variables than protein age, leading to the potential discovery of other significant and more discriminatory interaction patterns. By studying the fit of exponential random graphs to PINs (see Appendix B) we also note that assortative mixing by protein attributes such as protein age and function cannot alone account for the clustering observed in PPI networks.

PINs and model networks are very different with respect to their ego-networks characteristics. PINs tend to have large, diverse neighbourhoods (ego-networks), whereas their equivalent model networks with the same number of nodes, edges and even degree distribution show poor and homogeneous samples (see Chapter 4). In Chapter 5 we explore these neighbourhoods to devise a new network comparison method. Our method, *Egotif*, is similar in spirit to alignment-free sequence comparison methods and quite different from most current alignment-based approaches. We first tested the method with an array of model networks and observed that it is very successful at clustering networks of the same random graph model together. UPGMA-based phylogenetic trees appear to be better than NJ-based trees at forming meaningful clades between model types which use the same input information.

In Chapter 5 we also compare PINs amongst themselves. Using species data sets from DIP, HPRD and BioGRID with a genome coverage of at least 15%, Egotif is able to cluster the different species in phylogenetic trees identical to the ones given by NCBI. This is surprising since the method uses no biological data other than the PINs themselves and leads to the conclusion that there is information about evolution in the topologies of PINs. The relatively low coverage used also supports the view that the method is quite robust to noise and incompleteness, unlike most alignment-based

methods. Future work should include studies on the scaling and standardisation of counts. Effects of the choice of gold-standard set on the clustering performance should also be further investigated, for instance, preliminary studies show that using a mass spectroscopy (MS)-based network to compare other MS-based PINs results in a better clustering for the case of the BioGRID data sets. It would also be interesting to see how other types of network data cluster according to the method.

In summary, this thesis starts by exposing the problems and limitations of current scores for network comparison. We propose an improved comparison method based on these scores and find that current models do not fit to PINs. In order to understand the nature of protein interactions we then present several results linking protein age and protein structural class to network topology.

We end by introducing a new conceptual method for network comparison. *Egotif* compares neighbourhoods of networks in an averaging fashion and is able to reconstruct correct phylogenies between species based on PIN data alone. Future research on understanding the nature of subgraphs together with the design of better network models is expected to refine network comparison methods even further allowing real-time, online database query-based comparisons across a variety of organisms and conditions.

References

M. Abu-Farha, F. Elisma, and D. Figeys. Identification of Protein–Protein Interactions by Mass Spectrometry Coupled Techniques. *Protein–Protein Interaction*, 110:67–80, 2008. 12

R. Albert and A.L. Barabasi. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74(1):47–97, 2002. 17

W. Ali and C.M. Deane. Functionally guided alignment of protein interaction networks for module detection. *Bioinformatics*, 25(23):3166–3173, 2009. 107

W. Ali and C.M. Deane. Evolutionary analysis reveals low coverage as the major challenge for protein interaction network alignment. *Molecular BioSystems*, 6(11):2296–2304, 2010. 10, 77, 110

E. Alm and A.P. Arkin. Biological networks. *Current Opinion in Structural Biology*, 13(2): 193–202, 2003. 4

N. Alon, P. Dao, F. Hajirasouliha, I.and Hormozdiari, and S.C. Sahinalp. Biomolecular network motif counting and discovery by color coding. *Bioinformatics*, 24(13):1367–4803, 2008. 42, 60

P. Aloy and R.B. Russell. Structural systems biology: modelling protein interactions. *Nature Reviews Molecular Cell Biology*, 7(3):188–197, 2006. 33, 34

S.F. Altschul, W. Gish, W. Miller, E.W. Myers, D.J. Lipman, et al. Basic local alignment search tool. *Journal of molecular biology*, 215(3):403–410, 1990. 101

Y. Artzy-Randrup, S.J. Fleishman, N. Ben-Tal, and L. Stone. Comment on" Network Motifs: Simple Building Blocks of Complex Networks" and" Superfamilies of Evolved and Designed Networks". *Science*, 305(5687):1107, 2004. 29

P. Bachman and Y. Liu. Structure discovery in ppi networks using pattern-based network decomposition. *Bioinformatics*, 25(14):1814, 2009. 4, 29

S. Bandyopadhyay, M. Mehta, D. Kuo, M.K. Sung, R. Chuang, E.J. Jaehnig, B. Bodenmiller, K. Licon, W. Copeland, M. Shales, et al. Rewiring of genetic networks in response to dna damage. *Science*, 330(6009):1385, 2010. 36

A.L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439): 509, 1999. 23, 31

G. Bebek, P. Berenbrink, C. Cooper, T. Friedetzky, J. Nadeau, and SC Sahinalp. The degree distribution of the generalized duplication model. *Theoretical Computer Science*, 369(1-3): 239–249, 2006. , 24, 25, 78, 198

N. Bhardwaj and H. Lu. Correlation between gene expression profiles and protein–protein interactions within and across genomes. *Bioinformatics*, 21(11):2730, 2005. 35

N. Bhardwaj and H. Lu. Co-expression among constituents of a motif in the protein-protein interaction network. *Journal of bioinformatics and computational biology*, 7(1):1, 2009. 35

B.E. Blaisdell. A measure of the similarity of sets of sequences not requiring sequence alignment. *Proceedings of the National Academy of Sciences*, 83(14):5155–5164, 1986. 102

J. Blitzstein and P. Diaconis. A sequential importance sampling algorithm for generating random graphs with prescribed degrees. *Internet Mathematics*, 6(4):489–522, 2011. 18

B. Bollobás. *Random Graphs*. Cambridge University Press, 2001. 17, 18, 26, 27

A. Brady, K. Maxwell, N. Daniels, and L.J. Cowen. Fault tolerance in protein interaction networks: stable bipartite subgraphs and redundant pathways. *PLoS One*, 4(4):e5364, 2009. 26

P. Braun and A.C. Gingras. History of protein-protein interactions: from egg-white to complex networks. *Proteomics*, 12(10):1478–1498, 2012. 3

B.J. Breitkreutz, C. Stark, M. Tyers, et al. The grid: The general repository for interaction datasets. *Genome Biology*, 4(3):R23, 2003. 104

B.J. Breitkreutz, C. Stark, T. Reguly, L. Boucher, A. Breitkreutz, M. Livstone, R. Oughtred, D.H. Lackner, J. Bahler, V. Wood, et al. The BioGRID interaction database: 2008 update. *Nucleic Acids Research*, 36(suppl 1):D636–D640, 2007. xi, 13, 105, 131

A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener. Graph structure in the web. *Computer Networks*, 33(1-6):309–320, 2000. 23

S. Chatterjee and P. Diaconis. Estimating and understanding exponential random graph models. *Arxiv preprint arxiv:1102.2650*, 2011. 169, 170

L. Chen, R.S. Wang, and X.S. Zhang. *Biomolecular networks: methods and applications in systems biology*. John Wiley & Sons Inc, 2009. 7

P.Y. Chen, C.M. Deane, and G. Reinert. Predicting and validating protein interactions using network structure. *PLoS Computational Biology*, 4(7):e1000118, 2008. 7, 8, 33, 74

B. Chor and T. Tuller. Biological networks: Comparison, conservation, and evolutionary trees. In *Research in Computational Molecular Biology*, pages 30–44. Springer, 2006. 8, 109

C. Chothia and J. Janin. Principles of protein-protein recognition. *Nature*, 256(5520):705–708, 1975. 3

G. Ciriello and C. Guerra. A review on models and algorithms for motif discovery in protein-protein interaction networks. *Briefings in Functional Genomics and Proteomics*, 7(3):147–156, 2008. 42

L. da F. Costa, F.A. Rodrigues, G. Travieso, and P.R.V. Boas. Characterization of complex networks: A survey of measurements. *Advances in Physics*, 56(1):167–242, 2007. 4, 15, 30

M. Costanzo, A. Baryshnikova, J. Bellay, Y. Kim, E.D. Spear, C.S. Sevier, H. Ding, J.L.Y. Koh, K. Toufighi, S. Mostafavi, et al. The genetic landscape of a cell. *Science*, 327(5964): 425, 2010. 35, 36

D.R. Cox. *Principles of Statistical Inference*. Cambridge University Press, 2006. 72

M.E. Cusick, H. Yu, A. Smolyar, K. Venkatesan, A.R. Carvunis, N. Simonis, J.F. Rual, H. Borick, P. Braun, M. Dreze, et al. Literature-curated protein interaction datasets. *Nature Methods*, 6(1):39–46, 2008. 14

J. Dall and M. Christensen. Random geometric graphs. *Physical Review E*, 66(1):16121–16130, 2002. 28

T. Dandekar, S. Schuster, B. Snel, M. Huynen, and P. Bork. Pathway alignment: application to the comparative analysis of glycolytic enzymes. *Biochemical Journal*, 343(1):115–124, 1999. 106

J.J. Daudin, F. Picard, and S. Robin. A mixture model for random graphs. *Statistics and Computing*, 18(2):173–183, 2008. 19, 20

M. O. Dayhoff. Computer analysis of protein evolution. *Scientific American*, 221:86–95, 1969. 100

M.O. Dayhoff, R.M. Schwartz, and B.C. Orcutt. A model of evolutionary change in proteins. *In Atlas of Protein Sequences and Structure*, 5:345–352, 1978. 100

D.J. de Solla Price. Networks of scientific papers. *Science*, 149(3683):510, 1965. 23

C.M. Deane, Ł. Salwiński, I. Xenarios, and D. Eisenberg. Protein interactions. *Molecular & Cellular Proteomics*, 1(5):349–356, 2002. 77

A.H. Dekker and B.D. Colbert. Network robustness and graph topology. In *Proceedings of the 27th Australasian conference on Computer science-Volume 26*, pages 359–368. Australian Computer Society, Inc., 2004. 26

R. Durrett. *Random graph dynamics*. Cambridge Univ Pr, 2007. ISBN 0521866561. 18

P. Edman. Method for determination of the amino acid sequence in peptides. *ACTA Chemica Scandinavica*, 4(283-293):7, 1950. 100

P. Erdős and A. Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, 5:17–61, 1960. 17, 26, 124

S. Erten, X. Li, G. Bebek, J. Li, and M. Koyutürk. Phylogenetic analysis of modularity in protein interaction networks. *BMC bioinformatics*, 10(1):333, 2009. 8, 109

J. Felsenstein. The number of evolutionary trees. *Systematic Biology*, 27(1):27–33, 1978. 134

D.F. Feng and R.F. Doolittle. Progressive sequence alignment as a prerequisitetto correct phylogenetic trees. *Journal of molecular evolution*, 25(4):351–360, 1987. 101

S. Fields and O. Song. A novel genetic system to detect protein-protein interactions. *Nature*, 340(6230):245, 1989. 9, 10, 104

J. Flannick, A. Novak, C. B. Do, B.S. Srinivasan, and S. Batzoglou. Automatic parameter learning for multiple local network alignment. *Journal of Computational Biology*, 16(8): 1001–1022, 2009. 108

O. Frank and D. Strauss. Markov graphs. *Journal of the American Statistical Association*, pages 832–842, 1986. 21

H. Ge, Z. Liu, G.M. Church, and M. Vidal. Correlation between transcriptome and interactome mapping data from saccharomyces cerevisiae. *Nature genetics*, 29(4):482–486, 2001. 35

T.A. Gibson and D.S. Goldberg. Questioning the ubiquity of neofunctionalization. *PLoS computational biology*, 5(1):e1000252, 2009. 25

T.A. Gibson and D.S. Goldberg. Improving evolutionary models of protein interaction networks. *Bioinformatics*, 27(3):376–382, 2011. 25, 74, 145

A. Goel, S. Rai, and B. Krishnamachari. Monotone properties of random geometric graphs have sharp thresholds. *Annals of Applied Probability*, 15(4):2535–2552, 2005. 27

D. Greenbaum, C. Colangelo, K. Williams, and M. Gerstein. Comparing protein abundance and mrna expression levels on a genomic scale. *Genome Biology*, 4(9):117, 2003. 34

J.A. Grochow and M. Kellis. Network motif discovery using subgraph enumeration and symmetry-breaking. *Lecture Notes in Computer Science*, 4453:92–106, 2007. 42

R. Hafner. The asymptotic distribution of random clumps. *Computing*, 10(4):335–351, 1972. 19

M.S. Handcock. Assessing degeneracy in statistical models of social networks. *Working Paper no.39*, 2003. 170

MA Harris, J. Clark, A. Ireland, J. Lomax, M. Ashburner, R. Foulger, K. Eilbeck, S. Lewis, B. Marshall, C. Mungall, et al. The gene ontology (go) database and informatics resource. *Nucleic acids research*, 32(Database issue):D258, 2004. 33

G.T. Hart, I. Lee, and E.M. Marcotte. A high-accuracy consensus map of yeast protein complexes reveals modular nature of gene essentiality. *BMC Bioinformatics*, 8(1):236, 2007. 12

L. H. Hartwell, J. J. Hopfield, S. Leibler, and A. W. Murray. From molecular to modular cell biology. *Nature*, 402(6761 Suppl):C47–52, Dec 2 1999. 4

Xionglei He and Jianzhi Zhang. Why do hubs tend to be essential in protein networks? *PLoS Genet*, 2(6):e88, 06 2006. 31

H. Ho, T. Milenković, V. Memišević, J. Aruri, N. Pržulj, and A.K. Ganesan. Protein interaction network topology uncovers melanogenesis regulatory network components within functional genomics datasets. *BMC Systems Biology*, 4(1):84, 2010. 8

F. Hormozdiari, P. Berenbrink, N. Pržulj, and S.C. Sahinalp. Not all scale-free networks are born equal: the role of the seed graph in PPI network evolution. *PLoS Computational Biology*, 3(7):e118, 2007. 25, 42, 58, 78

H. Huang and J.S. Bader. Precision and recall estimates for two-hybrid screens. *Bioinformatics*, 25(3):372–8, 2009. 77

H. Huang, B.M. Jedynak, and J.S. Bader. Where have all the interactions gone? estimating the coverage of two-hybrid protein interaction maps. *PLoS computational biology*, 3(11):e214, 2007. 4, 8, 10, 131

R.M. Hummel, D.R. Hunter, and M.S. Handcock. A steplength algorithm for fitting ergms. *Technical Reports and Preprints*, (10-03), 2010. 170

D.R. Hunter and M.S. Handcock. Inference in curved exponential family models for networks. *Journal of Computational and Graphical Statistics*, 15(3):565–583, 2006. 171

T. Ideker and R. Sharan. Protein networks in disease. *Genome Research*, 18(4):644–652, 2008. 4

P.J. Ingram, M.P.H. Stumpf, and J. Stark. Network motifs: structure does not determine function. *BMC Bioinformatics*, 7:1–12, 2006. 4

T. Ito, K. Tashiro, S. Muta, R. Ozawa, T. Chiba, M. Nishizawa, K. Yamamoto, S. Kuhara, and Y. Sakaki. Toward a protein–protein interaction map of the budding yeast: a comprehensive system to examine two-hybrid interactions in all possible combinations between the yeast proteins. *Proceedings of the National Academy of Sciences*, 97(3):1143–1147, 2000. 10, 47, 104

L.J. Jensen and P. Bork. Biochemistry: Not comparable, but complementary. *Science*, 322 (5898):56–57, 2008. 9, 10, 12, 63, 78

H. Jeong, B. Tombor, R. Albert, ZN Oltvai, and A.L. Barabasi. The large-scale organization of metabolic networks. *Nature*, 407(6804):651–654, 2000. 23

S. Jones and J.M. Thornton. Principles of protein-protein interactions. *Proceedings of the National Academy of Sciences*, 93(1):13–20, 1996. 3

T.H. Jukes and C.R. Cantor. Evolution of protein molecules. *Mammalian protein metabolism*, III:21–132, 1969. 100

J. Kaiser. Proteomics: public-private group maps out initiatives. *Science*, 296(5569):827, 2002. 14

M. Kalaev, V. Bafna, and R. Sharan. Fast and accurate alignment of multiple protein networks. *Journal of Computational Biology*, 16(8):989–999, 2009. 107

R. Kelley and T. Ideker. Systematic interpretation of genetic interactions using protein networks. *Nature biotechnology*, 23(5):561, 2005. 36

T.S. Keshava Prasad, R. Goel, K. Kandasamy, S. Keerthikumar, S. Kumar, S. Mathivanan, D. Telikicherla, R. Raju, B. Shafreen, A. Venugopal, et al. Human Protein Reference Database–2009 update. *Nucleic Acids Research*, 37(suppl 1):D767–D772, 2009. 13, 134

R. Khanin and E. Wit. How scale-free are biological networks. *Journal of Computational Biology*, 13(3):810–818, 2006. 24, 31, 76

W.K. Kim and E.M. Marcotte. Age-dependent evolution of the yeast protein interaction network suggests a limited role of gene duplication and divergence. *PLoS Computational Biology*, 4 (11):e1000232, 2008. 33, 64, 67, 74, 75, 145

K. Klemm and S. Bornholdt. Topology of biological networks and reliability of information processing. *Proceedings of the National Academy of Sciences of the United States of America*, 102(51):18414–18419, 2005. 8, 28

E. Klipp, R.C. Wade, and U. Kummer. Biochemical network-based drug-target prediction. *Current Opinion in Biotechnology*, 2010. 8

E.V. Koonin. Orthologs, paralogs, and evolutionary genomics 1. *Annu. Rev. Genet.*, 39:309–338, 2005. 33

Y.A.I. Kourmpetis, A.D.J. van Dijk, M.C.A.M. Bink, R.C.H.J. van Ham, C.J.F. Ter Braak, and I. Friedberg. Bayesian Markov Random Field Analysis for Protein Function Prediction Based on Network Data. *PloS one*, 5(2):e9293, 2010. 8

N.J. Krogan, G. Cagney, H. Yu, G. Zhong, X. Guo, A. Ignatchenko, J. Li, S. Pu, N. Datta, A.P. Tikuisis, et al. Global landscape of protein complexes in the yeast Saccharomyces cerevisiae. *Nature*, 440(7084):637–643, 2006. 11, 104

O. Kuchaiev, P.T. Wang, Z. Nenadic, and N. Pržulj. Structure of brain functional networks. *Conf Proc IEEE Eng Med Biol Soc*, pages 4166–4170, 2009. 46

M. Kuhn, C. Von Mering, M. Campillos, L.J. Jensen, and P. Bork. STITCH: interaction networks of chemicals and proteins. *Nucleic Acids Research*, 36(Database issue):D684, 2008. 8

J. Kyte, R.F. Doolittle, et al. A simple method for displaying the hydropathic character of a protein. *Journal of molecular biology*, 157(1):105–132, 1982. 102

S.H. Lee, P.J. Kim, and H. Jeong. Statistical properties of sampled networks. *Physical Review E*, 73(1):016102, 2006. 16, 84

A.C.F. Lewis, N.J. Jones, M.A. Porter, and C.M. Deane. The function of communities in protein interaction networks at multiple scales. *BMC Systems Biology*, 4(100):1–14, 2010. 29

A.C.F. Lewis, N.J. Jones, M.A. Porter, and C.M. Deane. What evidence is there for the homology of protein-protein interactions? *PLoS Computational Biology*, 8(9):e1002645, 2012. 8, 109, 111

L. Li, C.J. Stoeckert, and D.S. Roos. Orthomcl: identification of ortholog groups for eukaryotic genomes. *Genome research*, 13(9):2178, 2003. 33

C. Liao, K. Lu, M. Baym, R. Singh, and B. Berger. IsoRankN: spectral methods for global alignment of multiple protein networks. *Bioinformatics*, 25(12):i253–i258, 2009. 108

R.A. Lippert, H. Huang, and M.S. Waterman. Distributional regimes for the number of k-word matches between two random sequences. *Proceedings of the National Academy of Sciences*, 99(22):13980–13989, 2002. 103, 116

Z. Liu, Q. Liu, H. Sun, L. Hou, H. Guo, Y. Zhu, D. Li, and F. He. Evidence for the additions of clustered interacting nodes during the evolution of protein interaction networks from network motifs. *BMC evolutionary biology*, 11(1):133, 2011. 29, 33, 65, 73, 172

D.J. Lockhart, H. Dong, M.C. Byrne, M.T. Follettie, M.V. Gallo, M.S. Chee, M. Mittmann, C. Wang, M. Kobayashi, H. Norton, and E.L. Brown. Expression monitoring by hybridization to high-density oligonucleotide arrays. *Nature biotechnology*, 14(13):1675–1680, 1996. 34

P.W. Lord, R.D. Stevens, A. Brass, and C.A. Goble. Investigating semantic similarity measures across the gene ontology: the relationship between sequence and annotation. *Bioinformatics*, 19(10):1275–1283, 2003. 172

S. Maslov and K. Sneppen. Specificity and stability in topology of protein networks. *Science*, 296(5569):910, 2002. 8, 21, 57, 65, 86, 89, 125

L.R. Matthews, P. Vaglio, J. Reboul, H. Ge, B.P. Davis, J. Garrels, S. Vincent, and M. Vidal. Identification of Potential Interaction Networks Using Sequence-Based Searches for Conserved Protein-Protein Interactions or Interologs. *Genome Research*, 11(12):2120–2126, 2001. 109

H.W. Mewes, D. Frishman, U. Güldener, G. Mannhaupt, K. Mayer, M. Mokrejs, B. Morgenstern, M. Münsterkötter, S. Rudd, and B. Weil. Mips: a database for genomes and protein sequences. *Nucleic acids research*, 30(1):31–34, 2002. 33, 171

M. Middendorf, E. Ziv, and C.H. Wiggins. Inferring network mechanisms: the drosophila melanogaster protein interaction network. *Proceedings of the National Academy of Sciences*, 102(9):3192–3197, 2005. 24, 57, 58, 124

S. Mika and B. Rost. Protein-protein interactions more conserved within than across species. *PLoS Computational Biology*, 2:e79, 2007. 109

T. Milenkovic, J. Lai, and N. Przulj. Graphcrunch: a tool for large network analyses. *BMC Bioinformatics*, 9:70–81, 2008. 42, 46, 161

T. Milenkovic, I. Filippis, M. Lappe, and N. Pržulj. Optimized null model for protein structure networks. *PLoS ONE*, 4(6):e5967, 2009. 46

R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs: simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002. 29, 41, 105, 111

A. Mirshahvalad, J. Lindholm, M. Derlen, and M. Rosvall. Significant communities in large sparse networks. *Arxiv preprint arXiv:1110.0305*, 2011. 30

M. Morris, M.S. Handcock, and D.R. Hunter. Specification of exponential-family random graph models: terms and computational aspects. *Journal of Statistical Software*, 24(4):1548, 2008. 169

A.G. Murzin, S.E. Brenner, T. Hubbard, and C. Chothia. Scop: a structural classification of proteins database for the investigation of sequences and structures. *Journal of molecular biology*, 247(4):536–540, 1995. 33, 94

S.B. Needleman and C.D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of molecular biology*, 48(3):443–453, 1970. 100

M. E. J. Newman, S. H. Strogatz, and D. J. Watts. Random graphs with arbitrary degree distributions and their applications. *Phys. Rev. E*, 64:026118, 2001. 18

M.E.J. Newman. The structure and function of complex networks. *Arxiv preprint cond-mat/0303516*, 2003. 124

M.E.J. Newman. *Networks: An Introduction.* Oxford University Press; Oxford, UK, 2012. 84

R.K. Nibbe, S.A. Chowdhury, M. Koyutürk, R. Ewing, and M.R. Chance. Protein–protein interaction networks and subnetworks in the biology of disease. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 3(3):357–367, 2011. 8

K. Nowicki and T.A.B. Snijders. Estimation and prediction for stochastic blockstructures. *Journal of the American Statistical Association*, 96(455):1077–1087, 2001. 19, 20

H. Ogata, W. Fujibuchi, S. Goto, and M. Kanehisa. A heuristic graph comparison algorithm and its application to detect functionally related enzyme clusters. *Nucleic Acids Research*, 28(20):4021–4028, 2000. 106

S. Ohno. *Evolution by gene duplication.* Springer Verlag, New York, 1970. 25

C.A. Orengo, AD Michie, S. Jones, D.T. Jones, MB Swindells, and J.M. Thornton. Cath-a hierarchic classification of protein domain structures. *Structure*, 5(8):1093–1109, 1997. 33

G. Östlund, T. Schmitt, K. Forslund, T. Köstler, D.N. Messina, S. Roopra, O. Frings, and E.L.L. Sonnhammer. Inparanoid 7: new algorithms and tools for eukaryotic orthology analysis. *Nucleic Acids Research*, 38(Suppl. 1):D196–203, 2010. 33, 64, 66

R. Pastor-Satorras, E. Smith, and R.V. Solé. Evolving protein interaction networks through gene duplication. *Journal of Theoretical biology*, 222(2):199–210, 2003. 25, 110

M. Penrose. *Random Geometric Graphs.* Oxford University Press, 2003. 19, 27, 89, 124

H.T.T. Phan and M.J.E. Sternberg. Pinalog: a novel approach to align protein interaction networksimplications for complex detection and function prediction. *Bioinformatics*, 28(9): 1239–1245, 2012. 108

J. Poncela, J. Gómez-Gardeñes, LM Floría, and Y. Moreno. Robustness of cooperation in the evolutionary prisoner's dilemma on complex networks. *New Journal of Physics*, 9:184, 2007. 26

C. Prieto and J. Rivas. APID: Agile Protein Interaction DataAnalyzer. *Nucleic Acids Research*, 34:W298, 2006. 13

N. Przulj. Biological network comparison using graphlet degree distribution. *Bioinformatics*, 23(2):177–183, 2007. 5, 37, 42, 43, 44, 46, 47, 48, 114

N. Przulj. Biological network comparison using graphlet degree distribution. *Bioinformatics*, 26(6):853–854, 2010. 45, 47

N. Pržulj and D.J. Higham. Modelling protein–protein interaction networks via a stickiness index. *Journal of the Royal Society Interface*, 3(10):711–716, 2006. 20, 55, 124, 164

N. Przulj, D. G. Corneil, and I. Jurisica. Modeling interactome: scale-free or geometric? *Bioinformatics*, 20(18):3508–3515, Dec 12 2004. 5, 19, 37, 42, 44, 114

H. Qin, H.H.S. Lu, W.B. Wu, and W.H. Li. Evolution of the yeast protein interaction network. *Proceedings of the National Academy of Sciences of the United States of America*, 100(22): 12820–4, 2003. 65

G. Reinert, D. Chew, F. Sun, and M.S. Waterman. Research Articles Alignment-Free Sequence Comparison (I): Statistics and Power. *Journal of Computational Biology*, 16(12):1615–1634, 2009. 103, 122

G. Rigaut, A. Shevchenko, B. Rutz, M. Wilm, M. Mann, and B. Séraphin. A generic protein purification method for protein complex characterization and proteome exploration. *Nature Biotechnology*, 17(10):1030–1032, 1999. 11, 104

T. Rito, Z. Wang, C.M. Deane, and G. Reinert. How threshold behaviour affects the use of subgraphs for network comparison. *Bioinformatics*, 26(18):i611–i617, 2010. 5, 8, 29, 37, 41, 116

T. Rito, C.M. Deane, and G. Reinert. The importance of age and high-degree, in protein-protein interaction networks. *Journal of Computational Biology*, 19(6):785–795, 2012. 5, 31, 63

J. Rivas and C. Fontanillo. Protein–Protein Interactions Essentials: Key Concepts to Building and Analyzing Interactome Networks. *PLoS Computational Biology*, 6(6), 2010. 13

G. Robins, P. Pattison, Y. Kalish, and D. Lusher. An introduction to exponential random graph (p*) models for social networks. *Social Networks*, 29(2):173–191, 2007. 169

J.F. Rual, K. Venkatesan, T. Hao, T. Hirozane-Kishikawa, A. Dricot, N. Li, G.F. Berriz, F.D. Gibbons, M. Dreze, N. Ayivi-Guedehoussou, et al. Towards a proteome-scale map of the human protein–protein interaction network. *Nature*, 437(7062):1173–1178, 2005. 47

N. Saitou and M. Nei. The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular biology and evolution*, 4(4):406–425, 1987. 123

L. Salwinski, C.S. Miller, A.J. Smith, F.K. Pettit, J.U. Bowie, and D. Eisenberg. The database of interacting proteins: 2004 update. *Nucleic Acids Research*, 32(suppl 1):D449–D451, 2004. 13, 30, 31, 66, 131

L. Sambourg and N. Thierry-Mieg. New insights into protein-protein interaction data lead to increased estimates of the s. cerevisiae interactome size. *BMC Bioinformatics*, 11(1):605, 2010. 63

C.M. Sanderson. The Cartographers toolbox: building bigger and better human protein interaction networks. *Briefings in Functional Genomics and Proteomics*, 8(1):1–11, 2009. 9

F. Sanger and A.R. Coulson. A rapid method for determining sequences in dna by primed synthesis with dna polymerase. *Journal of molecular biology*, 94(3):441–448, 1975. 100

K. Schwoch. Analysing network statistics: Spectral analysis of networks. *MSc Dissertation*, 2008. 24

R. Sharan and T. Ideker. Modeling cellular machinery through biological network comparison. *Nature Biotechnology*, 24(4):427–433, 2006. 4, 7, 33, 104, 106

R. Sharan, I. Ulitsky, and R. Shamir. Network-based prediction of protein function. *Molecular systems biology*, 3(88), 2007. 172

S.S. Shen-Orr, R. Milo, S. Mangan, and U. Alon. Network motifs in the transcriptional regulation network of Escherichia coli. *Nature Genetics*, 31(1):64–68, 2002. 4

Shepp, L. Normal functions of normal random variables. *SIAM Review*, 6:459, 1964. 103

Y. Shih and S. Parthasarathy. Scalable global alignment for multiple biological networks. *BMC Bioinformatics*, 13(Suppl 3):S11, 2012. 107, 108, 110

H. Shimazaki and S. Shinomoto. A method for selecting the bin size of a time histogram. *Neural Computation*, 19(6):1503–1527, 2007. 164

C. Shou, N. Bhardwaj, H.Y.K. Lam, K. Yan, P.M. Kim, M. Snyder, and M.B. Gerstein. Measuring the evolutionary rewiring of biological networks. *PLoS Computational Biology*, 7(1): e1001050, 2011. 8, 111

R. Singh, J. Xu, and B. Berger. Global alignment of multiple protein interaction networks with application to functional orthology detection. *Proceedings of the National Academy of Sciences*, 105(35):12763–12768, 2008. 107

T.F. Smith and M.S. Waterman. Identification of common molecular subsequences. *Journal of Molecular Biology*, 147:195–197, 1981. 101

T.A.B. Snijders. Markov chain monte carlo estimation of exponential random graph models. *Journal of Social Structure*, 3(2):1–40, 2002. 169

T.A.B. Snijders, P.E. Pattison, G.L. Robins, and M.S. Handcock. New specifications for exponential random graph models. *Sociological Methodology*, 36(1):99–153, 2006. 171

R. Sokal and C. Michener. A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin*, 38:1409–1438, 1958. 123

K. Song, J. Ren, Z. Zhai, X. Liu, and M. Deng. Alignment-Free Sequence Comparison Based on Next Generation Sequencing Reads : Extended Abstract. *RECOMB*, 7262:272–285, 2012. 102

E. Sprinzak, S. Sattath, and H. Margalit. How reliable are experimental protein-protein interaction data? *Journal of Molecular Biology*, 327(5):919–923, 2003. 77

U. Stelzl, U. Worm, M. Lalowski, C. Haenig, F.H. Brembeck, H. Goehler, M. Stroedicke, M. Zenkner, A. Schoenherr, S. Koeppen, et al. A human protein-protein interaction network: a resource for annotating the proteome. *Cell*, 122(6):957–968, 2005. 47

G.D. Stormo and G.W. Hartzell. Identifying protein-binding sites from unaligned dna fragments. *Proceedings of the National Academy of Sciences*, 86(4):1183, 1989. 102

R.A. Studer and M. Robinson-Rechavi. How confident can we be that orthologs are similar, but paralogs differ? *Trends in Genetics*, 25(5):210–216, 2009. 33

M.P.H. Stumpf, T. Thorne, E. de Silva, R. Stewart, H.J. An, M. Lappe, and C. Wiuf. Estimating the size of the human interactome. *Proceedings of the National Academy of Sciences*, 105 (19):6959–6964, 2008. 14, 60

D. Sun, J. Fan, H. Zhao, and B. Luo. Inferring Protein Annotation from Topological Structural Analysis on Protein Interaction Network. In *Bioinformatics and Biomedical Engineering (iCBBE), 2010 4th International Conference*, pages 1–4. IEEE, 2010. 8

D. Szklarczyk, A. Franceschini, M. Kuhn, M. Simonovic, A. Roth, P. Minguez, T. Doerks, M. Stark, J. Muller, P. Bork, et al. The STRING database in 2011: functional interaction networks of proteins, globally integrated and scored. *Nucleic Acids Research*, 2010. 13

S. Tavaré. Ancestral inference in population genetics. In J. Picard, editor, *École d'Été de Probabilités de Saint-Flour XXXI-2001*, volume 1837 of *Lecture Notes in Mathematics*, pages 1–188. Springer-Verlag, New York, 2004. 77

C.F. Taylor, N.W. Paton, K.S. Lilley, P.A. Binz, R.K. Julian, A.R. Jones, W. Zhu, R. Apweiler, R. Aebersold, E.W. Deutsch, et al. The minimum information about a proteomics experiment (MIAPE). *Nature Biotechnology*, 25(8):887–893, 2007. 14

P. Uetz, L. Giot, G. Cagney, T.A. Mansfield, R.S. Judson, J.R. Knight, D. Lockshon, V. Narayan, M. Srinivasan, P. Pochart, et al. A comprehensive analysis of protein–protein interactions in Saccharomyces cerevisiae. *Nature*, 403(6770):623–627, 2000. 10, 104

G. Upton and I. Cook. *Understanding statistics*. Oxford University Press, 1996. 164

A.X.C.N. Valente and M.E. Cusick. Yeast protein interactome topology provides framework for coordinated-functionality. *Nucleic Acids Research*, 34(9):2812–2819, 2006. 8, 28

K. Venkatesan, J.F. Rual, A. Vazquez, U. Stelzl, I. Lemmens, T. Hirozane-Kishikawa, T. Hao, M. Zenkner, X. Xin, K.I. Goh, et al. An empirical framework for binary interactome mapping. *Nature methods*, 6(1):83–90, 2009. 4, 131

A. Vespignani. Evolution thinks modular. *Nature Genetics*, 35(2):118–119, 2003. 28

S. Vinga and J. Almeida. Alignment-free sequence comparison–a review. *Bioinformatics*, 19 (4):513–523, 2003. 102

C. von Mering, R. Krause, B. Snel, M. Cornell, S.G. Oliver, S. Fields, and P. Bork. Comparative assessment of large scale data sets of protein-protein interactions. *Nature*, 417(6887):399–403, 2002. 4, 10, 12, 30, 31, 32, 47, 57, 131, 170, 171

S. Wasserman and P. Pattison. Logit models and logistic regressions for social networks: I. an introduction to markov graphs andp. *Psychometrika*, 61(3):401–425, 1996. 22

H.F. Winstanley, S. Abeln, and C.M. Deane. How old is your fold? *Bioinformatics*, 21(Suppl. 1):i449–458, 2005. 33, 64, 65, 66, 68, 76

Y.I. Wolf, P.S. Novichkov, G.P. Karev, E.V. Koonin, and D.J. Lipman. The universal distribution of evolutionary rates of genes and distinct characteristics of eukaryotic genes of different apparent ages. *Proceedings of the National Academy of Sciences of the United States of America*, 106(18):7273–7280, 2009. 68

S. Wuchty. Evolution and topology in the yeast protein interaction network. *Genome research*, 14(7):1310–1314, 2004. 31

S. Wuchty, Z.N. Oltvai, and A.L. Barabasi. Evolutionary conservation of motif constituents in the yeast protein interaction network. *Nature Genetics*, 35(2):176–179, 2003. 4, 29, 68, 73, 172

I. Xenarios, D.W. Rice, L. Salwinski, M.K. Baron, E.M. Marcotte, and D. Eisenberg. Dip: the database of interacting proteins. *Nucleic Acids Research*, 28(1):289–291, 2000. 104

H. Yu, D. Greenbaum, H. Xin Lu, X. Zhu, and M. Gerstein. Genomic analysis of essentiality within protein networks. *TRENDS in genetics*, 20(6):227–231, 2004. 31

Appendix A

Other GDDA comparisons and model fit

A.1 Model *versus* model comparisons using other subgraph-based scores

A.1.1 GDDA using the geometric mean

GraphCrunch [Milenkovic et al., 2008] also calculates a Graphlet Degree Distribution Agreement (GDDA) by taking the geometric mean instead of the arithmetic mean. That is, with $D^j(G,H)$ defined in Chapter 2 as being the standardised Euclidean distance between the two scaled and normalised graphlet count vectors the networks G and H for a specific automorphism orbit j,

$$\text{GDDA (geometric mean)} = \left(\prod_{j=0}^{72} \left(1 - D^j(G,H)\right) \right)^{1/73}.$$

Again, the GDDA score using the geometric mean depends on the number of vertices and on the number of edges in the network. Figure A.1.1 depicts model *versus* model comparisons.

A.2 Model *versus* model comparisons at high graph densities using GDDA

Model *versus* model comparisons for GEO3D and ER-DD were carried out for higher graph densities - up to 0.4 for GEO3D and 0.05 for ER (Figure A.2). These plots suggest that for higher graph densities the score is more stable and less sensitive to the appearance of subgraphs.

A.3 Assessing model fit for all PPI considered

To assess how well a random model network fits a given query network, we employ a protocol which consists in the comparison of two samples using two non-parametrics tests: a Monte Carlo test and a Wilcoxon Rank-Sum test. Both of these tests apply to two samples and test the null hypothesis that the samples come from the same distribution. The Monte Carlo test uses as alternative that the samples come from different distributions, whereas the Wilcoxon

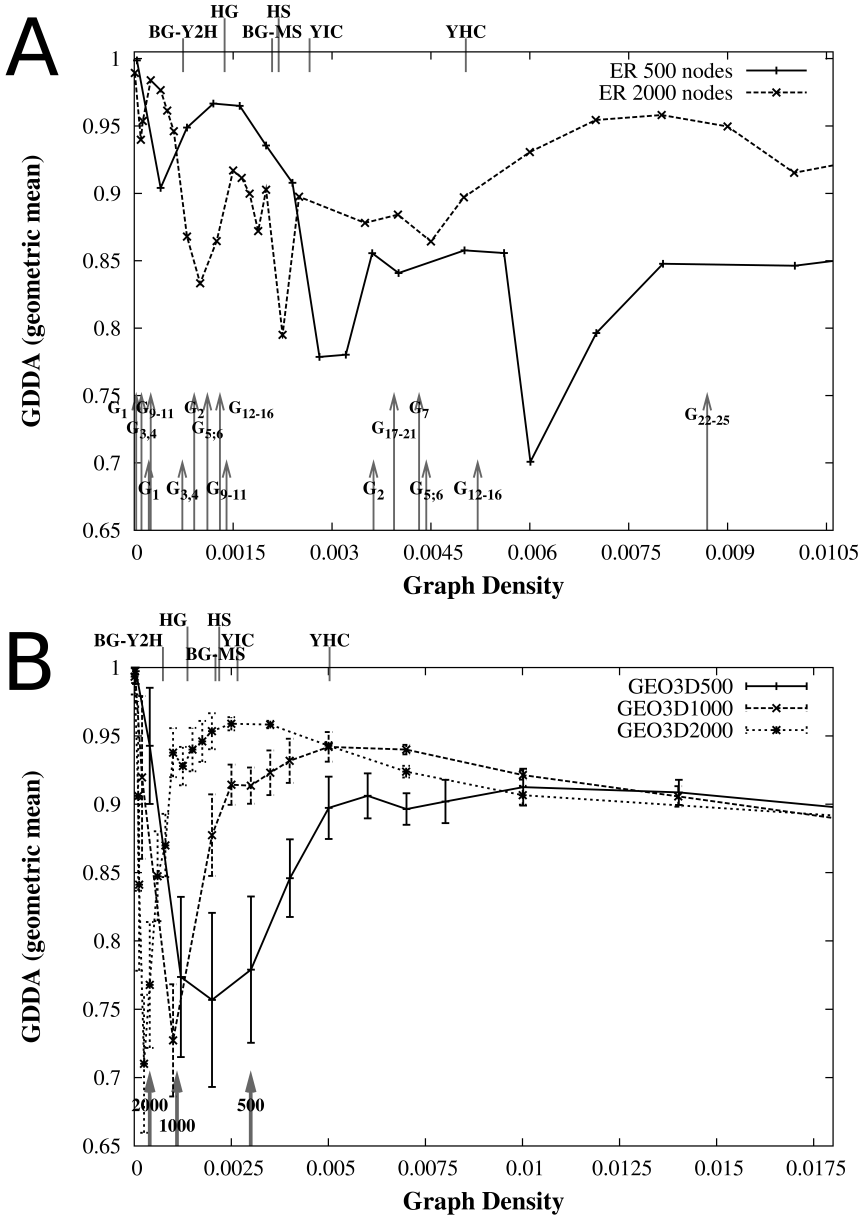


Figure A.1: GDDA (geometric mean) dependence on the number of vertices and edges of a network in model *versus* model comparisons. Average agreements of ER versus ER *(A)* and GEO-3D versus GEO-3D *(B)* graphs with 500, 1000 and 2000 vertices are plotted against graph density. Each value represents the average agreement of 50 networks. The graph density of the PPI networks considered (see Table 2.1) is indicated in the top x axis. In (A), the thresholds for the appearance of the several 3-5 nodes graphlets for an ER graph with 500 and 1000 nodes are pointed out along the x axis. In (B), the thresholds for the appearance of 3-node graphlets are indicated for the GEO graphs with 500, 1000 and 2000 nodes; although error bars are not statistically informative, they were included to give a sense of the variability present in the GDDA values considered.

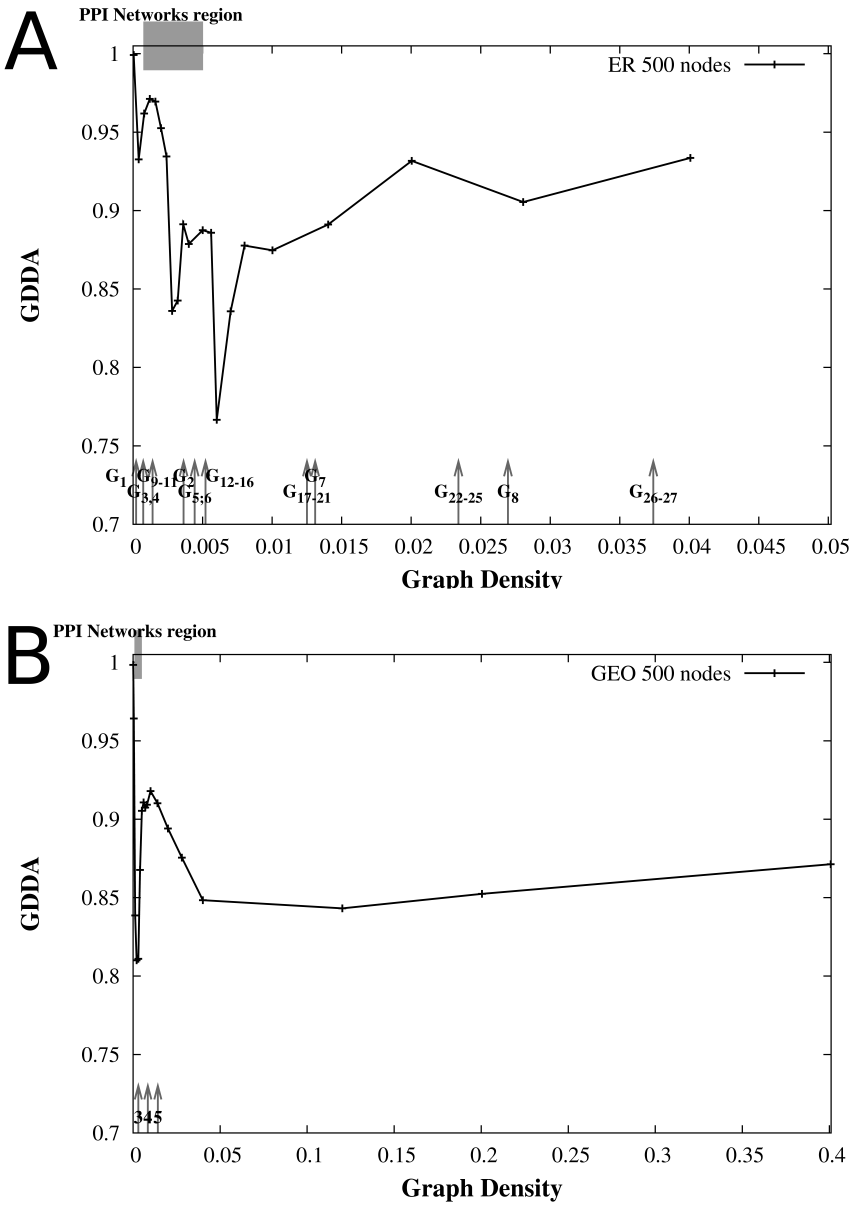


Figure A.2: GDDA dependence on the number of vertices and edges of a network in model *versus* model comparisons for higher graph densities. Average agreements of ER versus ER *(A)* and GEO3D versus GEO3D *(B)* graphs with 500 vertices are plotted against graph densities up to 0.05 for ER and 0.4 for GEO3D. Each value represents the average agreement of 50 networks. The graph density region of the PPI networks considered (see Table 2.1) is indicated in the top x axis. In (A), the thresholds for the appearance of the several 3- to 5-node graphlets for an ER graph with 500 nodes are pointed out along the x axis. In (B), the thresholds for the appearance of 3, 4 and 5-node graphlets are indicated for the GEO3D graphs with 500 nodes.

Rank-Sum test uses as alternative that the two different distributions are shifted versions of each other, with a non-zero shift.

For the Monte Carlo test, for our query network we generate N random graphs from the given model, so that the number of edges and vertices are within 1% of the corresponding numbers for the query network.

We calculate the GDDA scores between the query network and each of these N random graphs and take the average over these GDDA agreements, resulting in one number; call it S_0 . We then generate M random graphs from the given model and use each of these as pseudo-query input graph; repeating the above procedure for each of these M random graphs, we hence obtain M averages over N GDDA scores; call these $S_1, ..., S_{99}$. Now we order the vector $(S_0, ..., S_{99})$ in increasing order, $S_{(0)} \leq S_{(1)} \leq ... \leq S_{(99)}$. If S_0 is the k^{th} of these numbers, then the p-value of the test is $\frac{k}{N+1}$. Thus, if $N = 99$ and if S_0 is smaller than all of $S_1, ..., S_{99}$, then the p-value is 0.01. With $N + 1 = 100$ observations, this is the lowest possible p-value for the test. For a more powerful test we employ the non-parametric Wilcoxon Rank-Sum test. Here we directy compare the GDDA scores, without averaging. For the query network, we generate N random graphs as before and store the resulting GDDA scores as an N-vector, we call this *Sample B. Sample A*, the result of M graphs from a model each compared with other N graphs from the same model, can comprise all the resulting $M \times N$ GDDAs. The two samples are then used to perform the test [Upton and Cook, 1996].

The p-values obtained upon comparison of the PPI considered with GEO3D and ER-DD random graph models are shown in Table A.1. Histograms of the 99 averages of GDDA against the observed value can be seen in Figures A.3 and A.3 for several particular comparisons. The histograms of the samples used in the Wilcoxon Rank-Sum test are shown in Figure A.3. The optimal bin size was calculated using the Matlab function sshist [Shimazaki and Shinomoto, 2007].

For the PPI networks YHC and BG-MS we also tested the STICKY model [Pržulj and Higham, 2006]. This model assumes that the probability of interaction increases with the expected degree of the vertices. Our results are the same as for ER, ER-DD and GEO comparisons: the p-values are 0.01 for the Monte Carlo test and 6.68×10^{-66} for the Wilcoxon test. The histograms of the samples used in the Wilcoxon Rank-Sum test are shown in Figure A.3.

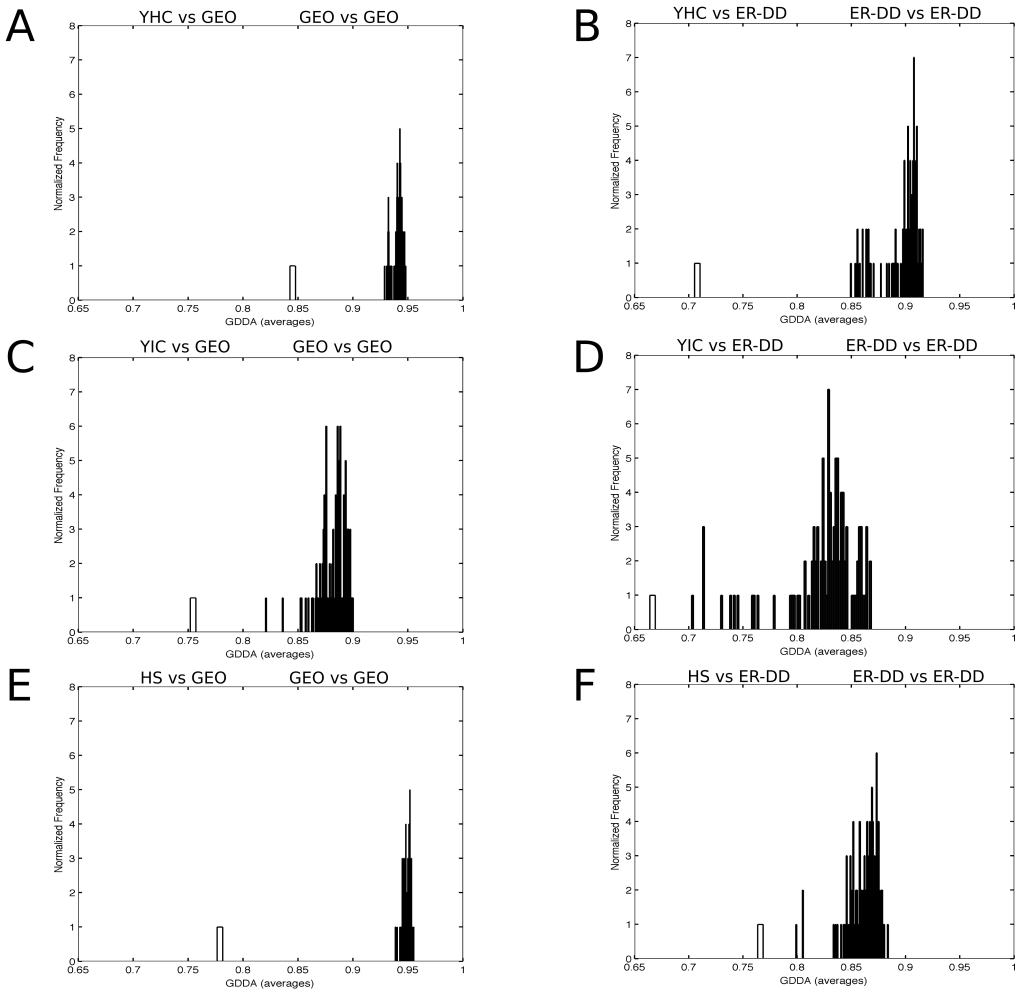


Figure A.3: Normalised histograms of average GDDA values. The *white* bar represents the observation in the Monte Carlo test, *i.e.* the average over N comparisons with the query network.

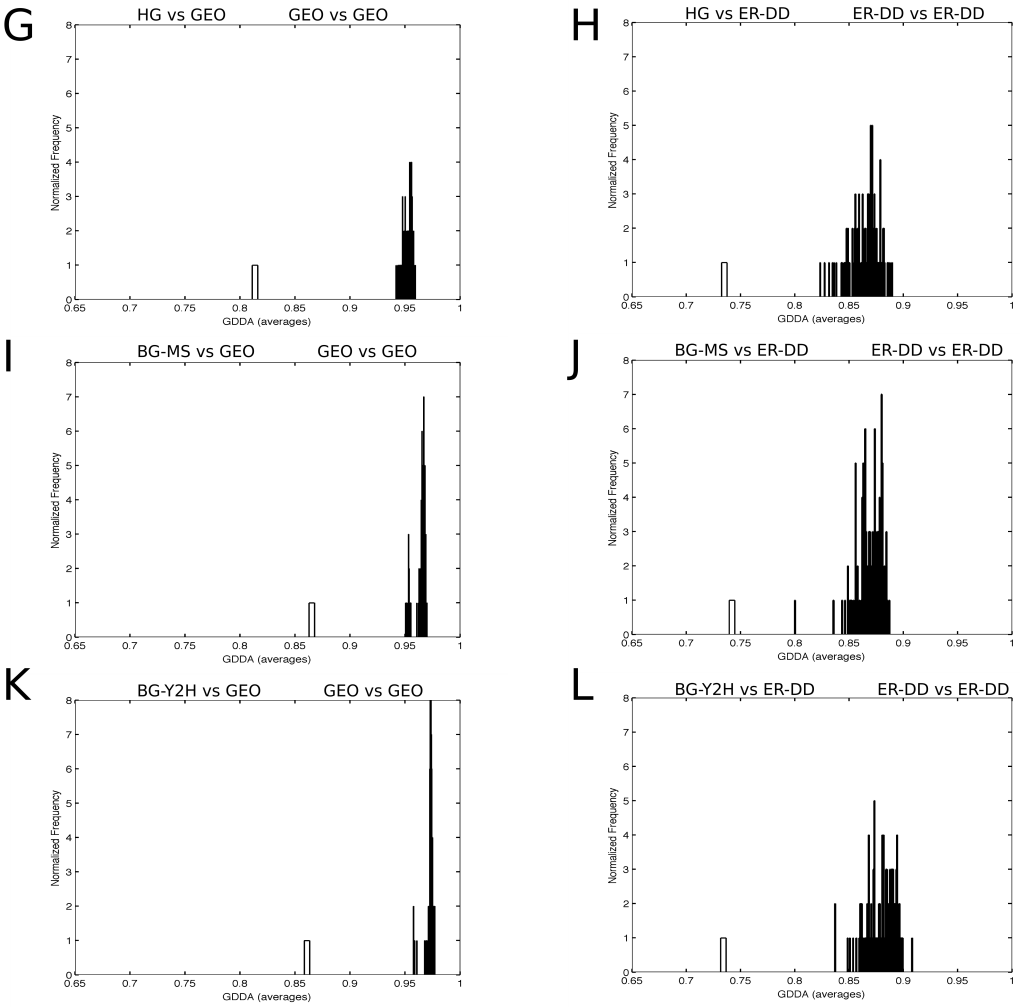


Figure A.4: Normalised histograms of average GDDA values. The *white* bar represents the observation in the Monte Carlo test, *i.e.* the average over N comparisons with the query network.

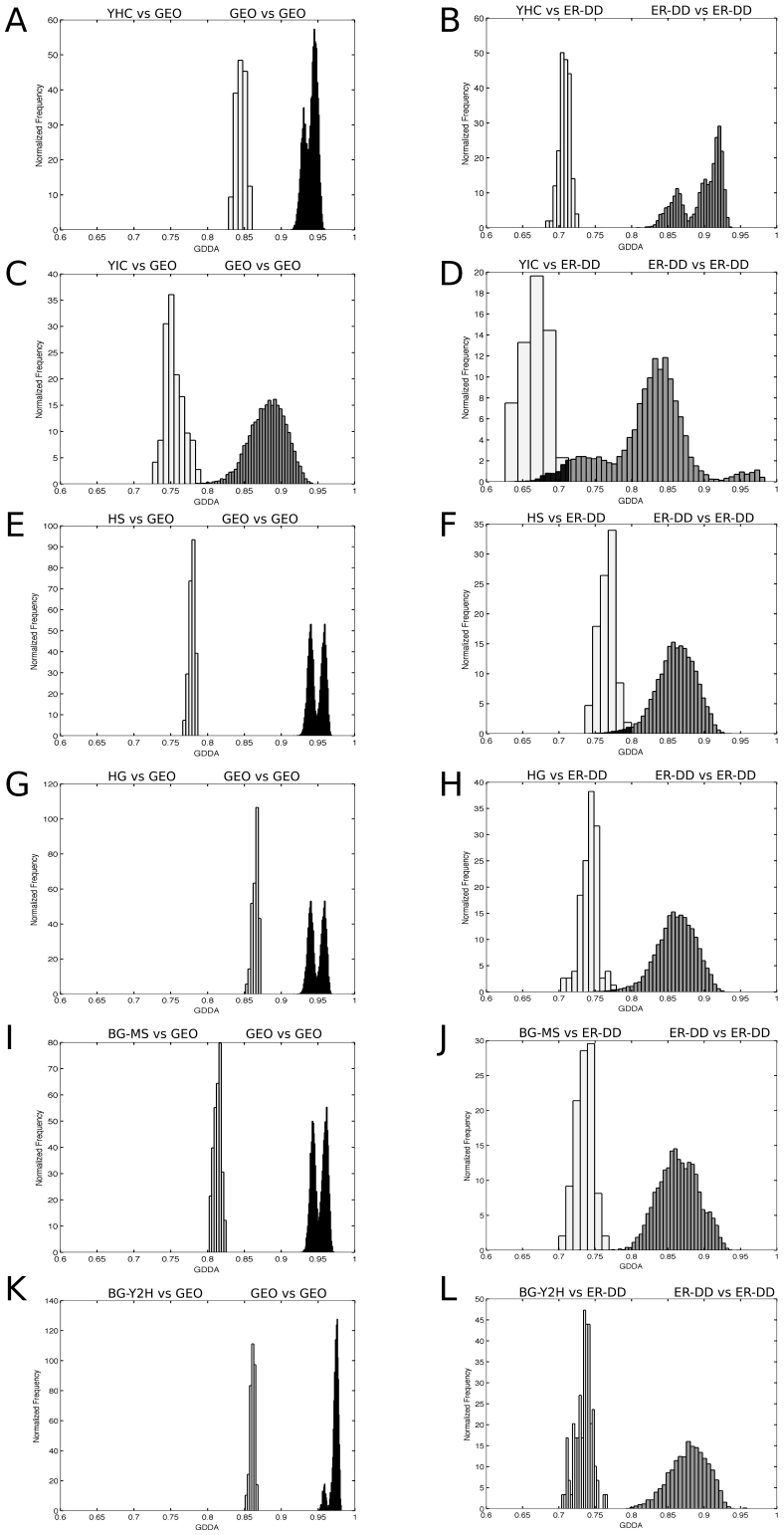


Figure A.5: Normalised histograms of GDDA values. Histograms of GDDA values between a PPI network (see Table 1) and 99 graphs of a model network are represented by the bars in *white*. Histograms of GDDA values between graphs of the same model network, 99x99, are shown in *grey*.

Table A.1: Assessing Model Fit: p-values obtained by employing Monte Carlo and Wilcoxon rank-sum tests for all PPI considered against GEO3D and ER-DD random graph models. The corresponding histograms can be found in Figures 4 and 5 in the reference letter for Monte Carlo and Wicoxon rank-sum tests respectively.

Model	Query	Monte Carlo	Wilcoxon	Reference letter
GEO3D	YHC	0.01	6.68×10^{-66}	A
ER-DD	YHC	0.01	6.68×10^{-66}	B
GEO3D	YIC	0.01	6.81×10^{-66}	C
ER-DD	YIC	0.01	1.31×10^{-64}	D
GEO3D	HS	0.01	6.68×10^{-66}	E
ER-DD	HS	0.01	5.25×10^{-65}	F
GEO3D	HG	0.01	6.68×10^{-66}	G
ER-DD	HG	0.01	8.15×10^{-66}	H
GEO3D	BG-MS	0.01	6.68×10^{-66}	I
ER-DD	BG-MS	0.01	7.09×10^{-66}	J
GEO3D	BG-Y2H	0.01	6.68×10^{-66}	K
ER-DD	BG-Y2H	0.01	6.68×10^{-66}	L

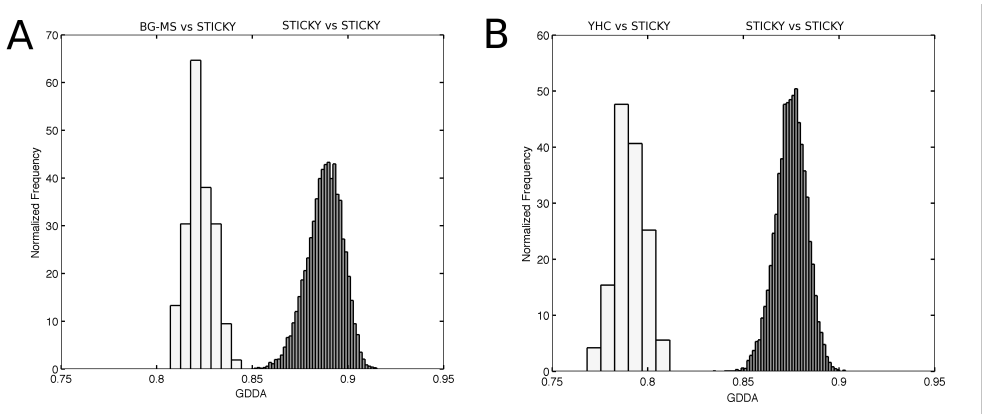


Figure A.6: Normalised histograms of GDDA values between the PPI networks BG-MS and YHC, and 99 graphs of the STICKY model are represented by the bars in *white*. Histograms of GDDA values between graphs of the same model network, 99x99, are shown in *grey*, *i.e.* STICKY *versus* STICKY comparisons.

Appendix B

Exponential random graphs

B.1 Exponential random graphs and PPI networks

Here we test the fit of Exponential Random Graphs Models (ERGMs) to PPI networks.

In Chapter 1 we pointed out that within ERGMs many classes of models are possible according to the dependency hypothesis specified. The strength of ERGMS resides in this specification of local forces which build and dictate the global structure of a network [Robins et al., 2007].

Here we use the R package ergm part of statnet to specify, fit and simulate ERGMs. A particular model is defined by a combination of terms, each specifying a Markov-like statistic whose associated parameters will be estimated. For instance, a model with the term "edges" specifies a single-parameter model where the vector of statistics $g_A(y)$ (see Chapter 1) will only contain the number of edges of the network. This model assumes an equal probability for all edges and it is essentially an ER model with graph density as the relevant parameter to estimate. Many other statistics exist and multiple models can be defined by combining them; for a comprehensive list of the statistics available in ergm see [Morris et al., 2008]. Parameter estimation in ERGMs is hard and currently an active area of research. Here we use mostly Markov Chain Monte Carlo (MCMC) methods implemented in ergm. Snijders [2002] first proposed a Gibbs sampling method to approximate random draws from a given Markov random graph model. A random graph is considered to be a Markov graph if for a fixed number of nodes, non-incident edges are independent conditional on the rest of the graph. We represent a network with g nodes by its adjacency matrix $y = (y_{ij})_{(1 \leq i,j \leq g)}$ with zero diagonal terms ($y_{ii} = 0$ for all i). This notation is similar to the one in Chapter 1 where y_{ii} specifies the existence or not of an edge between the nodes i and j. After the initial matrix $Y^{(1)}$ is chosen, we can then define a Gibbs sampler which randomly updates all elements of y_{ij}, one per draw, with all the others left unchanged. The stochastic process of updating the whole matrix defines the Markov chain such that the matrices $Y^{(t)}$ and $Y^{(t+1)}$ only differ in one element, the one updated in step t. Asymptotically, $Y^{(t)}$ tends towards the distribution of the random graph model specified. During model parameter estimation the statistics are updated as the conditional distribution switches to or from $y_{ij} = 1$ and $y_{ij} = 0$.

In addition to the problem that the relevant specific model is unknown for each particular case, parameter estimation is difficult due to the unknown normalisation constant κ and because similar distributions on graphs can be obtained by very different parameters. For instance, Chatterjee and Diaconis [2011] proved that when modelling transitivity (stronger tendency to complete a 2-star subgraph and forming a triangle *versus* forming an edge with a random

node, related with triangle counts) using just edges and triangle counts as sufficient statistics (configurations) for a positive value of $\eta_{triangle}$ the graph behaves like the ER model in the dense graph limit, while for negative values the graph becomes approximately bipartite.

Another common issue with these models, linked with the previous one, is that of model degeneracy, an ill-defined term describing estimation problems of using Markov Chain Monte Carlo methods where the chain gets "stuck" at parameters whose networks are either highly dense or the empty graph [Handcock, 2003; Hummel et al., 2010]. By studying phase transitions at the dense graph limit in the ERGM with edges and triangles as sufficient statistics, Chatterjee and Diaconis [2011] provided the first rigorous proof of degeneracy in this model, a sharp jump from very sparse to nearly complete graphs, skipping the intermediate regime - arguably the more interesting one.

Here our main focus is the final performance of a model. This performance is judged by comparing the average numbers of edges and triangles of networks simulated from a given model with those of the target network. As parametrisation and simulation can be computationally intensive for some models, here we use as target PPI network the smaller, high-confidence yeast PPI network assembled by only considering interactions that are supported by two or more types of experimental methods [von Mering et al., 2002]. This network, which we call "YHC", has 988 proteins and comprises $2,455$ interactions which form $6,353$ triangles. Our goal here is to test whether models are capable of generating networks with architectures similar to those of PPI networks, regardless of their order.

All models considered are indicated in Table B.1. Together with the model specification we also report some non-default control parameters of the MCMC algorithm and indicate the average counts of 50 simulated networks for each model. To ensure that the simulated networks are well away from the input network and had time to converge we use a burn-in of 10^7 for all simulations.

Table B.1: Different exponential random graph model specifications and their average number of edges and triangles of 50 simulated networks. The "exact" corresponds to the value $2,455$, the number of edges in YHC. The target number of triangles in YHC is $6,353$.

Model	**Specification**	**MCMC Sample size**	**Constraints**	**Average #Edges**	**Average #Triangles**
1	edges + kstar(2) + triangles	10^5	–	$\sim 3,100$	$> 30,000$
2	edges + triangles	10^5	–	~ 5000	$> 10,000$
3	edges + gwesp(5)	10^5	–	$\sim 2,100$	$\sim 3,000$
4	edges + altkstar(5)	10^5	–	$\sim 1,000$	$< 1,000$
5	edges + altkstar(5, fixed) + gwesp(5, fixed)	10^5	–	$\sim 3,200$	$> 35,000$
6	kstar(2) + triangles	10^5	fix #edges	*exact*	$\sim 9,800$
7	gwesp(3)	10^5	fix #edges	*exact*	$\sim 1,500$
8	gwdsp(5) + gwesp(5)	$2 * 10^5$	fix #edges	*exact*	$\sim 4,625$
9	gwesp(2)	$2 * 10^5$	fix #edges	*exact*	~ 500

First we attempted to fit a classic Markov model with edges, 2-stars and triangles as sufficient statistics. This corresponds to Model 1 in Table B.1. As mentioned by Hummel et al. [2010], this model cannot be fitted using MCMC-based methods resulting in the problem of degeneracy (see Chapter 1). The resulting networks are denser and over-estimate the number of edges and triangles in the network since for a positive 2-star coefficient, adding edges to high-degree nodes will increase the 2-star counts considerably.

The poor fit of an ERGM can have two reasons: either the model did not converge and hence the sampled networks are not from the specified distribution, or the model parameters

used to simulate the network are just too far away from the true maximum likelihood estimator (MLE). To prevent the first case we made sure the estimation procedure for all models presented in Table B.1 had converged by examining plots of the MCMC chain run; also for the simulation case, as mentioned, we use the generous burn-in of 10^7 for all models. As for the second reason, even if the simulating algorithm is producing representative sample networks, due to wrong parametrisation the samples just are not equal to those that would have been produced under the MLE. This case is more difficult to control since degeneracy may not be due to the shortcomings of MCMC but to the model not being properly specified. For instance, the closure of 2-stars (formation of triangles) may not occur homogeneously across all pairs of nodes in the network as specified by the term "triangles". To account for this we recur to two options: the use of node attributes that specify differential tendencies of groups of nodes to form triangles (see next subsection) and to consider the recently proposed alternating k-statistics for triangles and k-stars devised by Snijders et al. [2006] which aim to decrease the marginal impact of triangles on edge formation. Unlike the triangle census and clustering coefficient, a count is produced for every edge, resulting in a distribution of counts. For instance, to specify the formation of triangles (two nodes "sharing" a partner) the geometrically weighted edgewise shared partner (GWESP) statistic is defined as

$$w(y;\tau) = e^{\tau} \sum_{i=1}^{n-2} \left\{1 - (1 - e^{\tau})^{i}\right\} P_i. \tag{B.1}$$

This is the Hunter and Handcock [2006] re-parametrised form of the original alternating k-triangles statistic which gives each additional shared partner a declining, geometrically weighted, positive impact on the probability of two nodes forming an edge. The P_i stands for the number of interacting node pairs that have exactly i friends in common. The additional parameter τ controls the geometric rate of decline on the probability of an edge as the number of shared partners increase. Equivalent statistics exist for the case of k-stars. From Table B.1 we can see that these new statistics help to prevent the explosion of edges and triangles. Nonetheless, the fit of these two basic network descriptors is far from good. The best fit, with the additional constraint of fixing the number of edges, was Model 8 which is still short in $\sim 1,700$ triangles when compared to YHC. In the next section we follow-up the conjecture that the homogeneity assumption might be the reason behind these poor results and explore the effect of node attributes.

B.1.1 Node attributes alone cannot explain the topology of PINs

We tested the hypothesis that node attributes could help to understand the underlying forces for the network to form triangles. We labelled the nodes of YHC with their species coverage, a crude proxy for "protein age", according to the number of kingdoms in which a given protein is found. Each node is labelled with a number from zero to three reflecting the number of kingdoms that protein is absent [von Mering et al., 2002]. A protein found in Prokaryotes, Eukarya and Fungi will have number zero. A protein which is found in other Fungi will have number 1 and if it is yeast specific it will have number 3. See Chapter 3 for details on how protein age can be calculated. We also used the MIPS [Mewes et al., 2002] function category of each protein as labels. There are a total of 13 functional categories, some examples are "energy production", "translation", "transcription" and "stress and defence".

The ERGMs with node attributes considered are indicated in Table B.2. These include a "nodematch" term for nodal attribute mixing. This term tests for uniform homophily in the different attribute categories, or in other words, it tracks within-category edges, regardless of

Table B.2: Different exponential random graph model specifications including protein attributes such as protein age and functional category. The "exact" corresponds to the value $2,455$, the number of edges in YHC. The target number of triangles in YHC is $6,353$.

Model	**Specification**	**MCMC Sample size**	**Constraints**	**Average #Edges**	**Average #Triangles**
1	edges + nodematch(age)	$5 * 10^5$	–	$\sim 2,400$	~ 30
2	edges + nodematch(age) + gwesp(0, fixed)	$5 * 10^5$	–	$\sim 6,000$	$\sim 2,500$
3	edges + nodematch(MIPS)	$5 * 10^5$	–	$\sim 2,450$	~ 40
4	nodematch(age) + gwesp(5, fixed)	$5 * 10^5$	fix #edges	*exact*	$\sim 8,460$
5	nodematch(MIPS) + esp(3)	$5 * 10^5$	fix #edges	*exact*	~ 250
6 [1]	nodematch(age, MIPS)	$1 * 10^7$	fix #edges	*exact*	~ 40
7 [1]	nodematch(age, MIPS) + gwesp(5, fixed)	$5 * 10^5$	fix #edges	*exact*	$\sim 41,500$

[1]The estimation and MCMC run for these models is shown in Figure B.1A and B. For all cases the burn-in in estimation was $1.5 * 10^4$. The maximum number of iterations ranged from 20 to 50.

the category. This adds one network statistic to the model which counts the number of edges that share the same label. In the first model of Table B.2 we test whether homophily in age, previously identified [Liu et al., 2011; Wuchty et al., 2003], suffices to create networks with high number of triangles. In Model 6 we test this together with the homophily amongst nodes belonging to the same MIPS functional category, in this case whilst fixing the number of edges of the networks.

Figure B.1A shows the MCMC trace of the two assortative mixing parameters of "age" and "MIPS" in Model 6 and the histogram of the number of triangles in 50 networks generated from the same parametrised ERGM model. Although the models correctly estimate the graph density of the network (for Model 6 this is a constraint), both can reproduce the homophily between the different labels without forming triangles in the networks, which, on average have 30 and 40 triangles. Model 7 considers both node attributes and the new GWESP statistic for triangles. The networks simulated from this model, albeit with the same number of edges greatly over-estimate the number of triangles in YHC, see Figure B.1B. With $\sim 2,100$ more triangles than YHC, Model 4 performs best, although, as shown in Table B.3, it is not really able to reproduce the mixing matrix found in YHC or the homophily therein for protein age.

Table B.3: Assortative mixing matrix by protein age in $(left)$ YHC and $(right)$ the average matrix for the 50 simulated networks of Model 4.

	0	**1**	**2**	**3**
0	771	798	177	32
1	798	391	189	48
2	177	189	29	14
3	32	48	14	6

	0	**1**	**2**	**3**
0	368	1123	294	137
1	1123	180	207	100
2	294	207	15	28
3	137	100	28	3

Previous studies showed functional homophily in protein interactions and so-called "guilt-by-association" methods are commonly used to predict protein function [Lord et al., 2003; Sharan et al., 2007]. The nature of the relatively high clustering coefficients found in PPI networks have been neglected and may be tempting to attribute them to the fact that proteins of the same function tend to interact. From Models 1 and 3, we show that the network could reproduce homophily in functional and protein age categories without the formation of triangles. This finding also motivated the more in-depth study of the nature of triangles with respect to

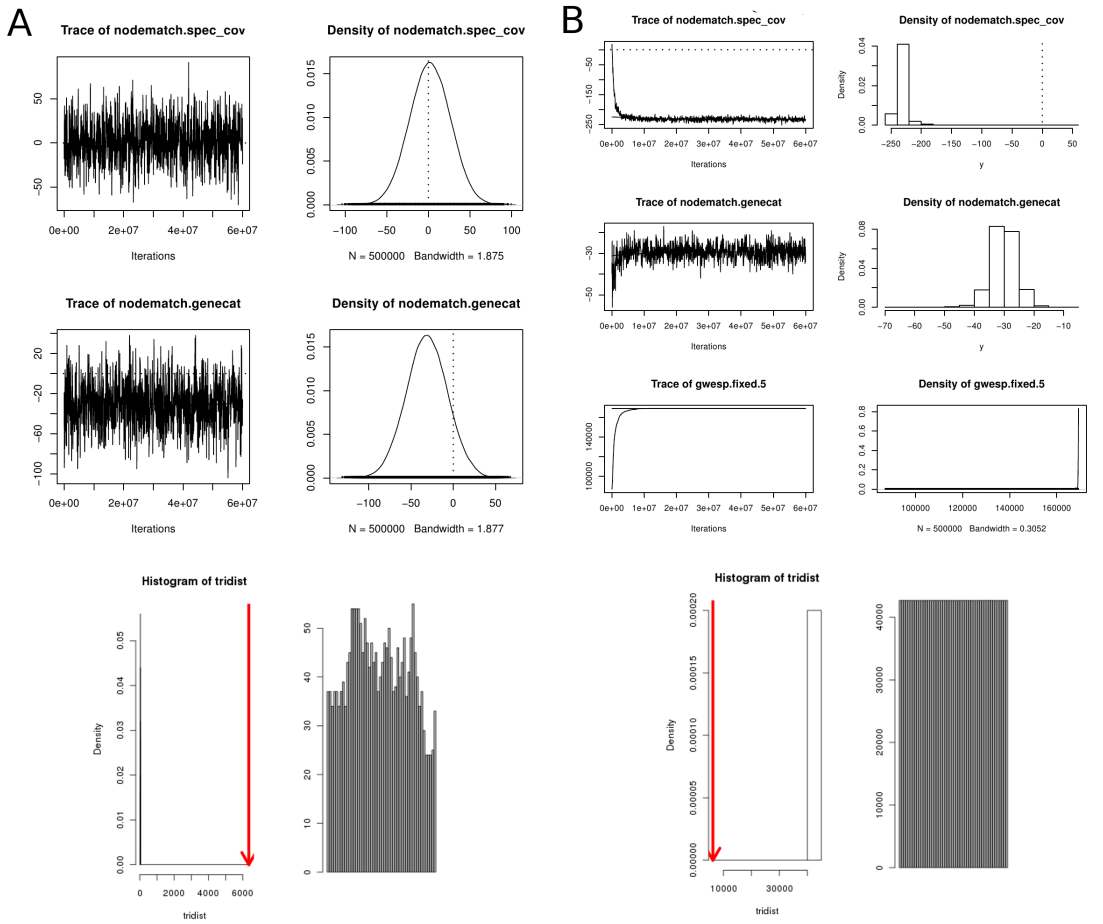


Figure B.1: ERGM with assortative mixing by protein age and MIPS functional category. *(Top)* Summary of MCMC samples; *(Bottom)* Histogram of number of triangles in 50 networks generated from the specific ERGM model. The red arrows indicate the number of triangles in the YHC PPI network. **(A)** ERGM with the assortative mixing parameters alone. **(B)** ERGM identical to (A) but also including the GWESP statistic.

protein age presented in Chapter 3.

B.2 On the fit to small ego-networks

Parameter estimation in ERGMs for large networks is computationally very intensive and since modelling PPI heterogeneity is also our goal we explore in this section the use of ERGMs to fit ego-networks. Here we specified the ERGM as a curved exponential random graph model and again make use of the alternating k-triangle (GWESP) and the alternating k-star (GWDSP) statistics, fixing the parameter for the number of edges. The model appears to perform relatively well in ego-networks that are either sparse or dense. Figure B.2 shows the example of 2 ego-

networks from YHC and the histograms of number of triangles present in 50 networks generated from that specifically parametrised ERGM model.

We also fitted 80 ego-networks from YHC, with the sole requirement that they have more than 10 nodes in them, to the same ERGM model. Curiously, the parameter space occupied by these ego-networks is relatively small (see Figure B.3) and the alternating k-triangle (GWESP) statistic is relatively stable across ego-networks with very distinct number of triangles (albeit the number of edges is not estimated but fixed). This could mean that by just specifying the number of edges and nodes of a given ego-network, the same set of parameters could be used to model many ego-networks across the network, but further investigation is required.

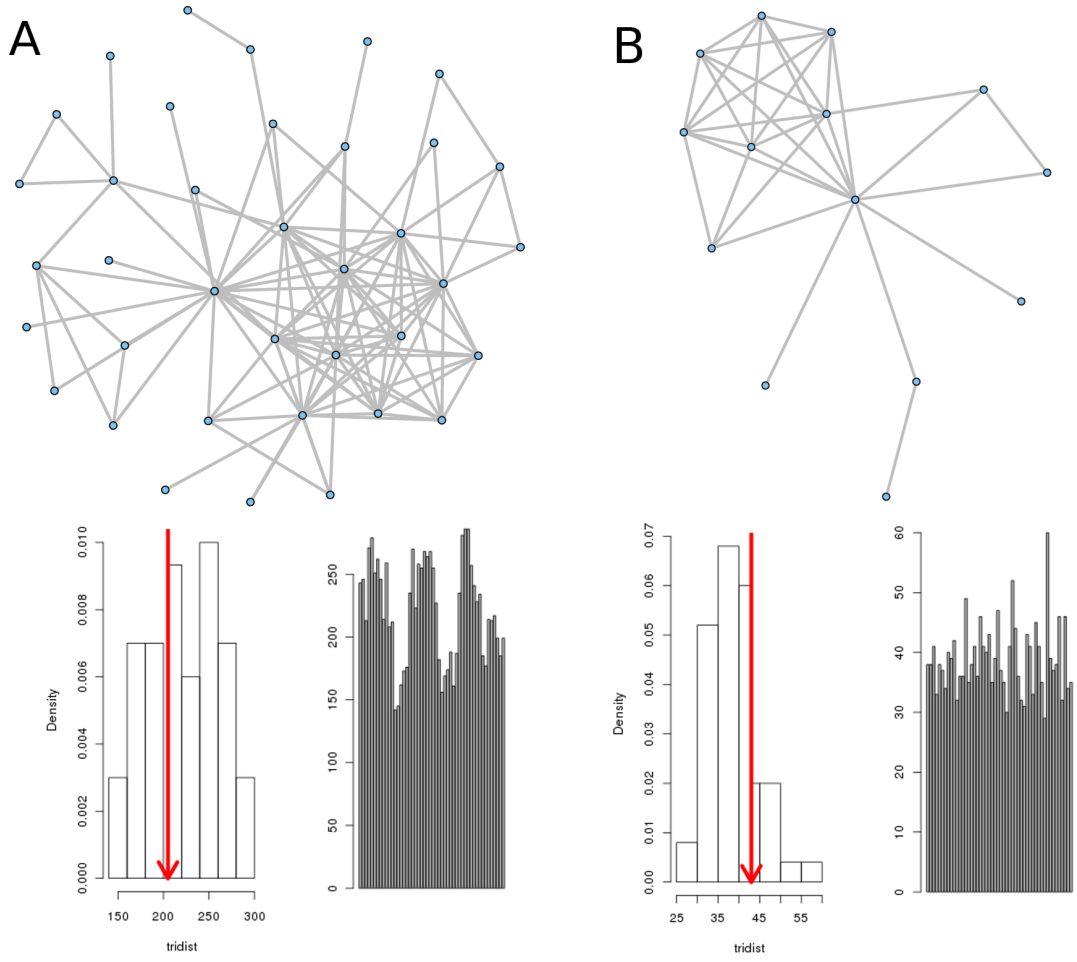


Figure B.2: ERGM with the alternating k-triangle and the alternating k-star statistics for the ego-networks (A) and (B) taken from YHC. Below each ego-network is the corresponding histogram of the number of triangles found in 50 networks generated with particular parameters estimated for that ego-network and this ERGM model. The red arrows indicate the number of triangles in the original ego-network.

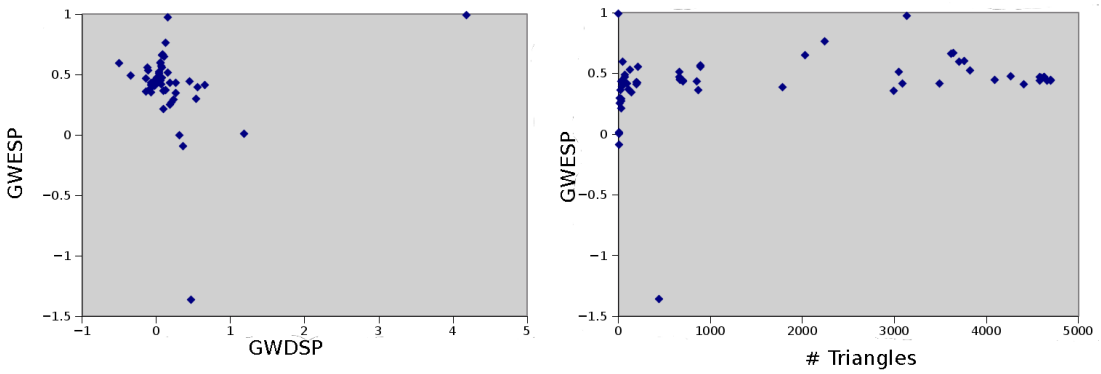


Figure B.3: *(Left)* Plot of the parameters GWESP and GWDSP estimated for 80 ego-networks of YHC. *(Right)* Variation of the values of the GWESP parameter with the number of triangles of 80 ego-networks from YHC.

Appendix C

Age-dependent patterns of the yeast PIN

Here we present the detailed results of our analysis, separated by dataset. We use "O" to denote an Old protein, "M" for a Middle-aged protein and "Y" for a Young one. We report the observed and expected absolute frequencies for the edges given the node frequencies and for the triangles under both the node and edge models, as well as the p-values under a chi-squared goodness-of-fit test as described in Chapter 3. Expected values are truncated to the nearest integer. The p-values in the tables refer to the individual contributions in the chi-squared test; they show whether a test based on this contribution alone, would already be rejected. We use the scientific notation for these, where "aE-b" denotes "$a \times 10^{-b}$".

For each dataset, we show two plots, one for the age-dependent patterns in the pairs (A) and one for patterns in the triangles (B) of the yeast PIN. The observed frequencies are always coloured in red. Expected frequencies under the node model are coloured in blue for both edge and triangle patterns; expected frequencies under the edge model for triangles are coloured in purple. All values of zero represent limits to computer precision.

The different datasets considered are:

- Complete DIP
- DIP_10 (considering proteins with degree greater or equal to 10)
- Anti_10 (considering proteins with degree less than 10)
- DIP_25
- Anti_25
- DIP-CORE
- DIP-CORE_5
- DIP with TAP-MS interactions omitted
- Complete DIP with a different protein age cut-off
- DIP considering Kim and Marcotte (2008) age definition
- Complete DIP and Gene Duplication and Divergence networks

C.1 Complete DIP

The tables in this section refer to Figures 3.3 and Figures 3.4 in the main text. For edges, the p-value for the chi-squared test with 3 degrees of freedom is zero. The contributions from the different edges are as follows.

	Observed	Expected given nodes	p-values
O-O	6096	3088	0.00E+000
M-O	8196	6939	4.35E-049
M-M	3584	3899	1.28E-005
O-Y	2793	4148	1.22E-095
M-Y	2817	4661	7.64E-158
Y-Y	642	1393	1.74E-087

If we collapse the edge types into edges containing proteins of the same age and edges containing proteins of different ages we get the following frequencies.

	Observed	Expected	p-values
Two of a kind	10322	8380	2.82E-097
One of a kind	13806	15748	1.17E-051

For triangles, the p-values are: under the node model, for the chi-squared test with 7 degrees of freedom, zero; under the edge model, for the chi-squared test with 4 degrees of freedom, zero. The contributions from the different triangles are as follows.

	Observed	Expected given nodes	p-values
O-O-O	4313	837	0.00E+000
M-O-O	5131	2820	0.00E+000
M-M-O	2852	3169	4.67E-005
O-O-Y	1277	1686	1.62E-018
M-M-M	1187	1187	1.00E+000
M-O-Y	1653	3789	1.02E-255
M-M-M	1103	2129	1.43E-102
O-Y-Y	304	1132	1.20E-126
M-Y-Y	402	1273	2.25E-124
Y-Y-Y	52	254	2.86E-031

	Observed	Expected given nodes	p-values
O-O-O	4313	1443	0.00E+000
M-O-O	5131	7825	1.65E-199
M-M-O	2852	4601	1.53E-142
O-O-Y	1277	909	2.98E-031
M-M-M	1187	293	0.00E+000
M-O-Y	1653	2465	1.24E-056
M-M-M	1103	543	2.45E-123
O-Y-Y	304	96	8.23E-097
M-Y-Y	402	97	4.77E-205
Y-Y-Y	52	2	0.00E+000

Results collapsed into patterns formed by proteins of the same age groups; homophily would be in this case a natural conclusion.

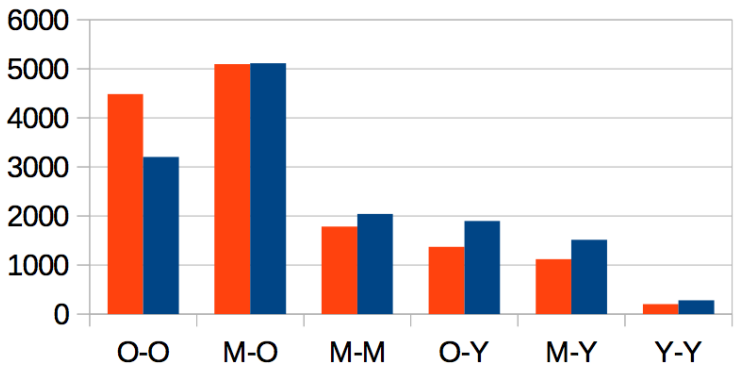
	Observed	Expected given nodes	p-values
Three of a kind	5552	2277	0.00E+000
Two of a kind	11069	12208	5.32E-020
One of a kind	1653	3789	1.02E-255

	Observed	Expected given edges	p-values
Three of a kind	5552	1738	0.00E+000
Two of a kind	11069	14071	2.48E-137
One of a kind	1653	2465	1.24E-056

C.2 DIP_10 (considering proteins with degree greater or equal to 10)

For edges, the p-value for the chi-squared test with 3 degrees of freedom is 4.88E-178. The contributions from the different edges are shown in the next figure and the particular p-values indicated in the following table.

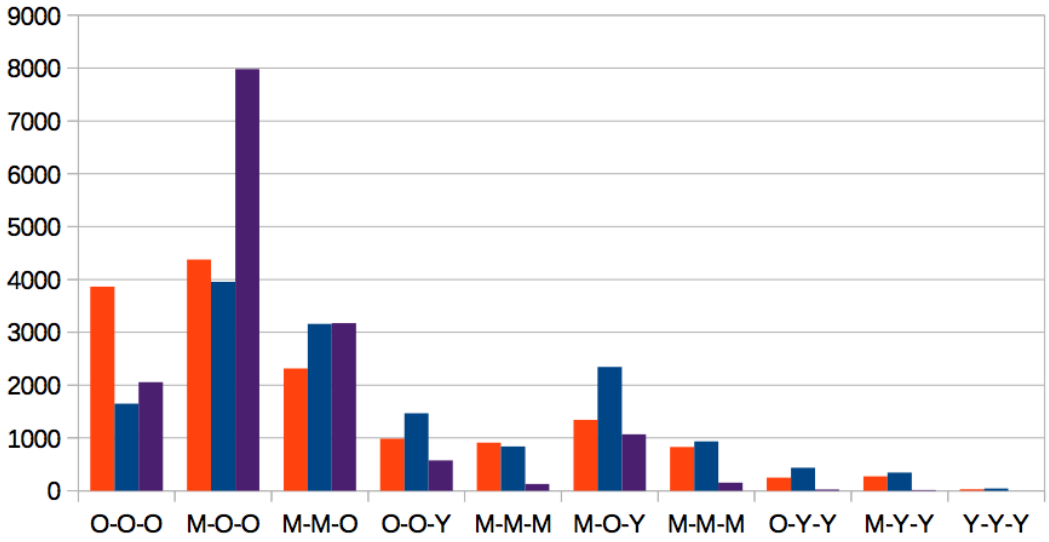
	Observed	Expected given nodes	p-values
O-O	4485	3202	3.47E-111
M-O	5098	5114	9.97E-001
M-M	1784	2043	3.69E-007
O-Y	1371	1898	1.79E-031
M-Y	1116	1516	1.07E-022
Y-Y	199	281	2.47E-005



For triangles, the p-values are: under the node model, for the chi-squared test with 7 degrees of freedom, zero; under the edge model, for the chi-squared test with 4 degrees of freedom, zero. The contributions from the different triangles are shown in the next figure and the particular p-values indicated in the following tables.

	Observed	Expected given nodes	p-values
O-O-O	3863	1651	0.00E+000
M-O-O	4377	3957	1.68E-007
M-M-O	2316	3161	3.93E-045
O-O-Y	988	1468	1.32E-030
M-M-M	914	842	5.12E-001
M-O-Y	1343	2345	1.87E-088
M-M-M	829	937	8.86E-002
O-Y-Y	249	435	1.65E-014
M-Y-Y	276	348	3.96E-002
Y-Y-Y	32	43	9.02E-001

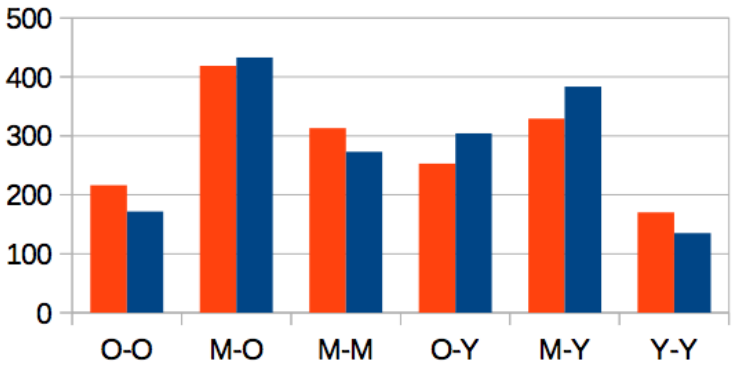
	Observed	Expected given edges	*p*-values
O-O-O	3863	2059	0.00E+000
M-O-O	4377	7982	0.00E+000
M-M-O	2316	3175	3.88E-049
O-O-Y	988	577	5.30E-062
M-M-M	914	130	0.00E+000
M-O-Y	1343	1068	1.67E-014
M-M-M	829	152	0.00E+000
O-Y-Y	249	26	0.00E+000
M-Y-Y	276	17	0.00E+000
Y-Y-Y	32	0	0.00E+000



C.3 Anti_10 (considering proteins with degree less than 10)

For edges, the p-value for the chi-squared test with 3 degrees of freedom is 2.04E-09. The contributions from the different edges are shown in the next figure and the particular p-values indicated in the following table.

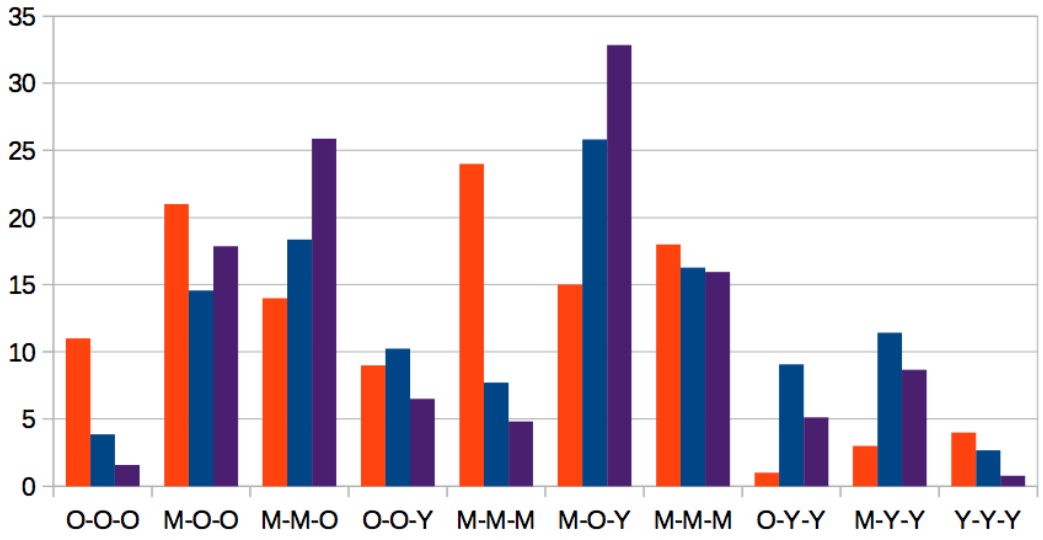
	Observed	Expected given nodes	p-values
O-O	216	172	9.60E-003
M-O	419	433	9.31E-001
M-M	313	273	1.16E-001
O-Y	253	304	3.44E-002
M-Y	329	384	5.13E-002
Y-Y	170	135	2.69E-002



For triangles, the p-values are: under the node model, for the chi-squared test with 7 degrees of freedom, 1.20E-12; under the edge model, for the chi-squared test with 4 degrees of freedom, 1.04E-35. The contributions from the different triangles are shown in the next figure and the particular p-values indicated in the following tables.

	Observed	Expected given nodes	p-values
M-O-O	21	15	8.99E-001
M-M-O	14	18	9.94E-001
O-O-Y	9	10	1.00E+000
M-M-M	24	8	1.46E-005
M-O-Y	15	26	7.17E-001
M-M-M	18	16	1.00E+000
O-Y-Y	1	9	4.10E-001
M-Y-Y	3	11	5.14E-001
Y-Y-Y	4	3	9.99E-001

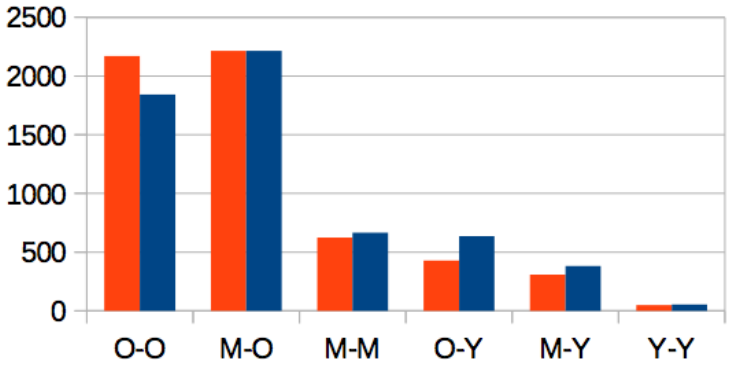
	Observed	Expected given edges	*p*-values
O-O-O	11	2	1.93E-011
M-O-O	21	18	9.68E-001
M-M-O	14	26	2.44E-001
O-O-Y	9	7	9.17E-001
M-M-M	24	5	9.68E-016
M-O-Y	15	33	4.59E-002
M-M-M	18	16	9.92E-001
O-Y-Y	1	5	5.06E-001
M-Y-Y	3	9	4.48E-001
Y-Y-Y	4	1	9.00E-003



C.4 DIP_25 (considering only proteins with degree greater or equal to 25)

For edges, the p-value for the chi-squared test with 3 degrees of freedom is 9.54E-31.The contributions from the different edges are shown in the next figure and the particular p-values indicated in the following table.

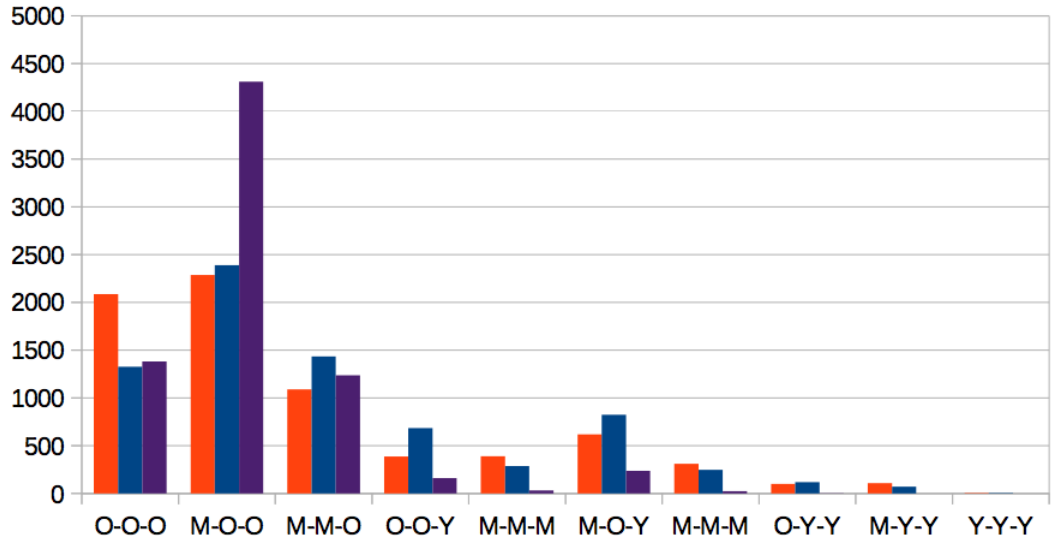
	Observed	Expected given nodes	p-values
O-O	2171	1843	1.32E-012
M-O	2214	2214	1.00E+000
M-M	623	665	4.46E-001
O-Y	428	635	1.64E-014
M-Y	309	381	3.37E-003
Y-Y	48	55	8.49E-001



For triangles, the p-values are: under the node model, for the chi-squared test with 7 degrees of freedom, 4.10E-164; under the edge model, for the chi-squared test with 4 degrees of freedom, zero.The contributions from the different triangles are shown in the next figure and the particular p-values indicated in the following tables.

	Observed	Expected given nodes	p-values
O-O-O	2088	1326	1.74E-090
M-O-O	2287	2390	7.31E-001
M-M-O	1090	1436	3.09E-015
O-O-Y	387	685	7.85E-025
M-M-M	389	287	7.69E-006
M-O-Y	620	823	1.45E-008
M-M-M	312	247	1.73E-002
O-Y-Y	101	118	9.33E-001
M-Y-Y	108	71	6.74E-003
Y-Y-Y	7	7	1.00E+000

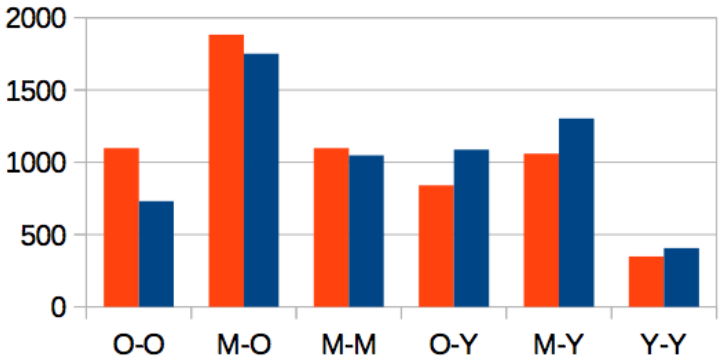
	Observed	Expected given edges	*p*-values
M-O-O	2287	4310	2.84E-204
M-M-O	1090	1237	1.59E-003
O-O-Y	387	161	2.47E-067
M-M-M	389	33	0.00E+000
M-O-Y	620	237	2.13E-132
M-M-M	312	24	0.00E+000
O-Y-Y	101	4	0.00E+000
M-Y-Y	108	2	0.00E+000
Y-Y-Y	7	0	0.00E+000



C.5 Anti_25 (complementary set of DIP_25, considering only proteins with degree less than 25)

For edges, the p-value for the chi-squared test with 3 degrees of freedom is 7.73E-66. The contributions from the different edges are shown in the next figure and the particular p-values indicated in the following table.

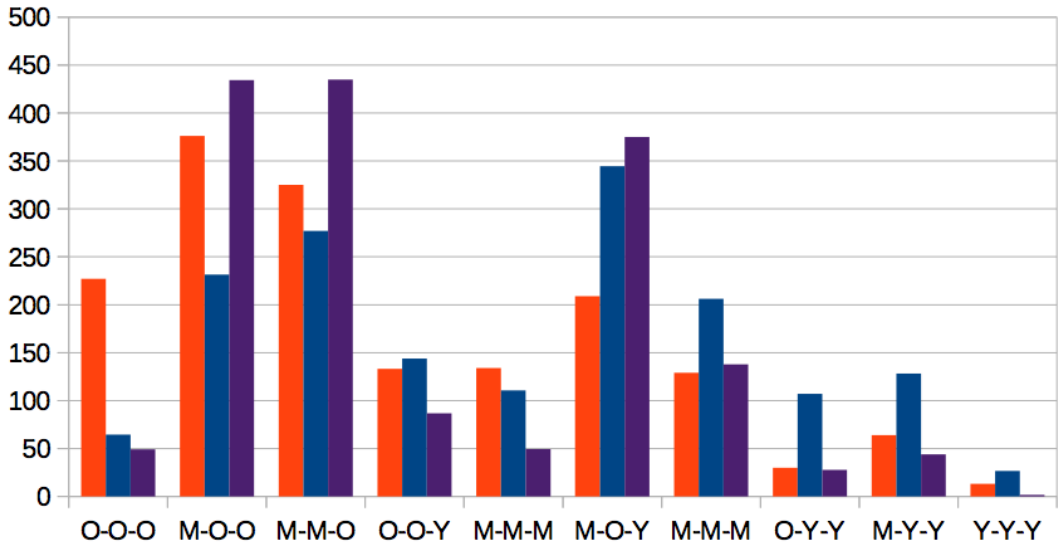
	Observed	Expected given nodes	p-values
O-O	1097	731	1.86E-039
M-O	1883	1751	1.89E-002
M-M	1098	1048	5.03E-001
O-Y	841	1089	3.55E-012
M-Y	1060	1304	7.11E-010
Y-Y	349	405	5.00E-002



For triangles, the p-values are: under the node model, for the chi-squared test with 7 degrees of freedom, 3.87E-145; under the edge model, for the chi-squared test with 4 degrees of freedom, 1.52E-218. The contributions from the different triangles are shown in the next figure and the particular p-values indicated in the following tables.

	Observed	Expected given nodes	p-values
O-O-O	227	64	1.33E-084
M-O-O	376	231	1.02E-016
M-M-O	325	277	3.07E-001
O-O-Y	133	144	9.97E-001
M-M-M	134	111	6.65E-001
M-O-Y	209	345	3.22E-009
M-M-M	129	206	1.48E-004
O-Y-Y	30	107	1.18E-009
M-Y-Y	64	128	3.74E-005
Y-Y-Y	13	27	4.35E-001

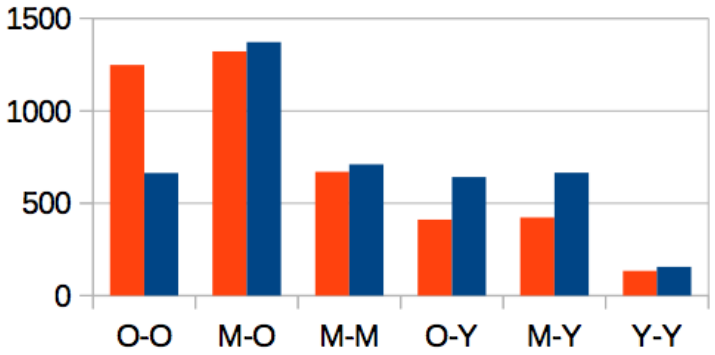
	Observed	Expected given edges	*p*-values
M-O-O	376	434	9.70E-002
M-M-O	325	435	1.41E-005
O-O-Y	133	87	5.56E-005
M-M-M	134	49	1.75E-030
M-O-Y	209	375	4.23E-015
M-M-M	129	138	9.67E-001
O-Y-Y	30	28	9.95E-001
M-Y-Y	64	44	5.36E-002
Y-Y-Y	13	2	5.47E-017



C.6 DIP-CORE (the high-confidence subset of DIP)

For edges, the p-value for the chi-squared test with 3 degrees of freedom is 1.78E-150. The contributions from the different edges are shown in the next figure and the particular p-values indicated in the following table.

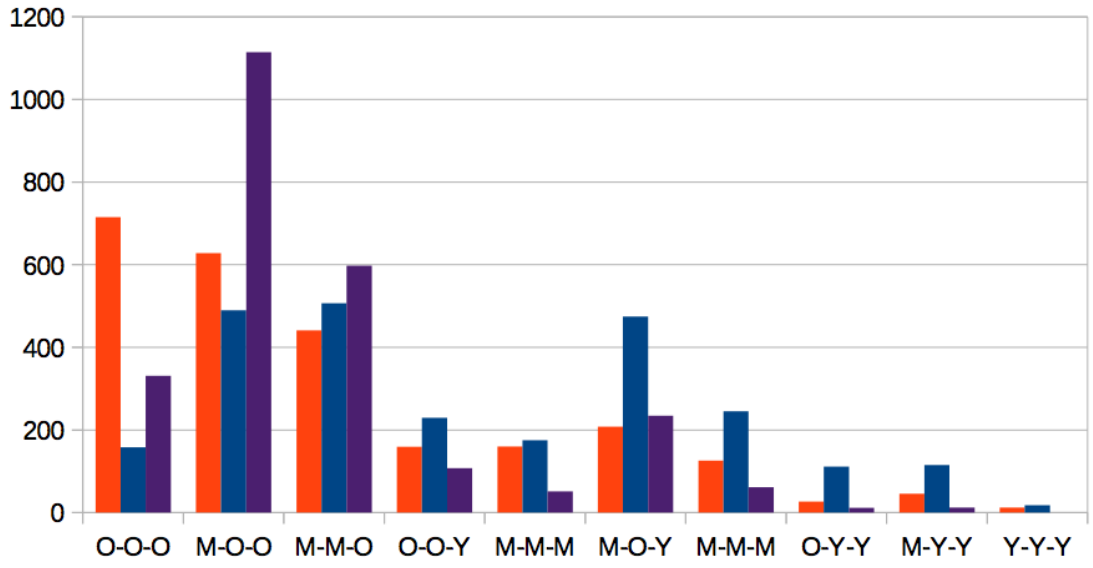
	Observed	Expected given nodes	p-values
O-O	1249	663	9.99E-112
M-O	1323	1373	6.12E-001
M-M	670	710	5.15E-001
O-Y	411	643	5.37E-018
M-Y	423	665	5.62E-019
Y-Y	134	156	3.89E-001



For triangles, the p-values are: under the node model, for the chi-squared test with 7 degrees of freedom, zero; under the edge model, for the chi-squared test with 4 degrees of freedom, 5.30E-317.The contributions from the different triangles are shown in the next figure and the particular p-values indicated in the following tables.

	Observed	Expected given nodes	p-values
O-O-O	715	158	0.00E+000
M-O-O	628	490	1.91E-006
M-M-O	441	507	2.89E-001
O-O-Y	159	229	3.07E-003
M-M-M	160	175	9.90E-001
M-O-Y	208	474	4.87E-029
M-M-M	126	245	3.54E-010
O-Y-Y	27	111	2.85E-011
M-Y-Y	46	115	7.01E-007
Y-Y-Y	12	18	9.62E-001

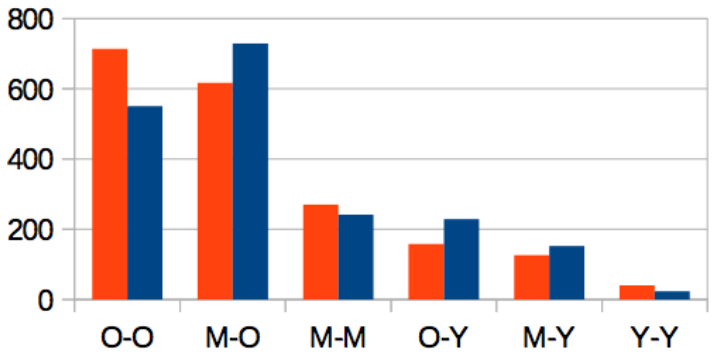
	Observed	Expected given edges	*p*-values
O-O-O	715	331	5.05E-095
M-O-O	628	1115	8.14E-045
M-M-O	441	598	2.49E-008
O-O-Y	159	108	6.06E-005
M-M-M	160	51	4.94E-049
M-O-Y	208	235	5.58E-001
M-M-M	126	61	3.91E-014
O-Y-Y	27	12	3.61E-004
M-Y-Y	46	12	2.58E-019
Y-Y-Y	12	0	7.37E-070



C.7 DIP-CORE_5 (considering only proteins with degree greater or equal to 5)

For edges, the p-value for the chi-squared test with 3 degrees of freedom is 6.64E-24. The contributions from the different edges are shown in the next figure and the particular p-values indicated in the following table.

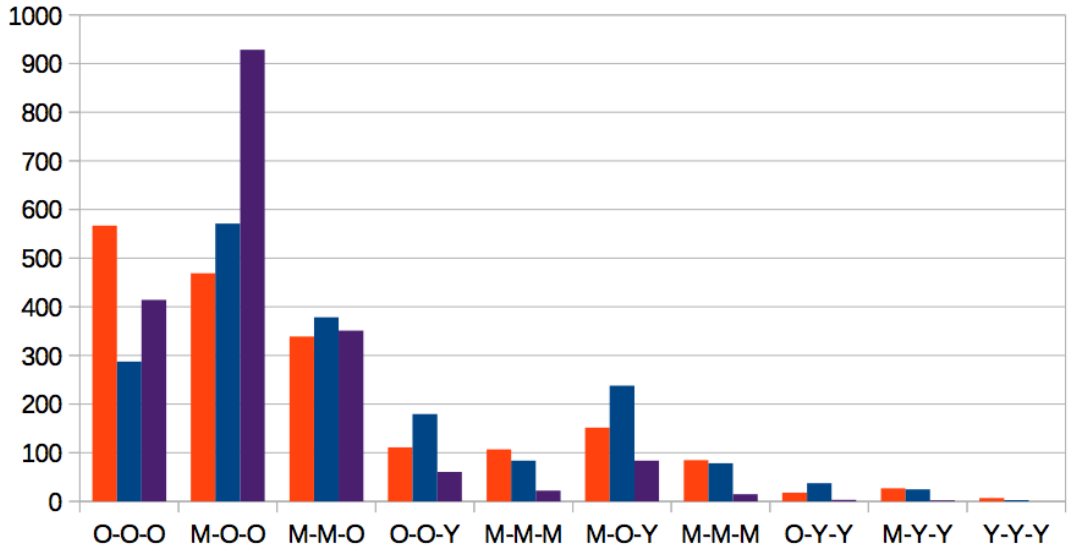
	Observed	Expected given nodes	p-values
O-O	714	550	2.49E-011
M-O	617	729	1.88E-004
M-M	270	241	1.84E-001
O-Y	158	229	1.63E-005
M-Y	126	152	1.13E-001
Y-Y	40	24	4.20E-003



For triangles, the p-values are: under the node model, for the chi-squared test with 7 degrees of freedom, 3.03E-77; under the edge model, for the chi-squared test with 4 degrees of freedom, zero. The contributions from the different triangles are shown in the next figure and the particular p-values indicated in the following tables.

	Observed	Expected given nodes	p-values
O-O-O	567	287	6.51E-055
M-O-O	469	571	1.05E-002
M-M-O	339	379	7.65E-001
O-O-Y	111	180	4.67E-004
M-M-M	107	84	4.76E-001
M-O-Y	152	238	6.14E-005
M-M-M	85	79	9.99E-001
O-Y-Y	18	37	1.86E-001
M-Y-Y	27	25	1.00E+000
Y-Y-Y	7	3	3.81E-001

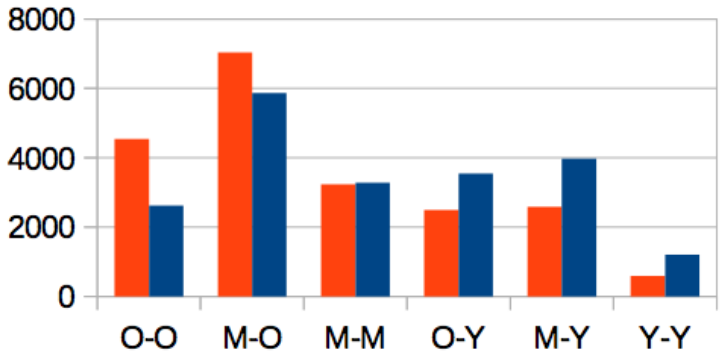
	Observed	Expected given edges	*p*-values
O-O-O	567	415	1.95E-011
M-O-O	469	929	4.45E-048
M-M-O	339	351	9.81E-001
O-O-Y	111	61	2.43E-008
M-M-M	107	22	8.02E-068
M-O-Y	152	84	2.96E-011
M-M-M	85	15	6.97E-072
O-Y-Y	18	3	9.18E-013
M-Y-Y	27	2	2.83E-060
Y-Y-Y	7	0	3.68E-141



C.8 DIP without TAP-MS data

For edges, the p-value for the chi-squared test with 3 degrees of freedom is zero. The contributions from the different edges are shown in the next figure and the particular p-values indicated in the following table.

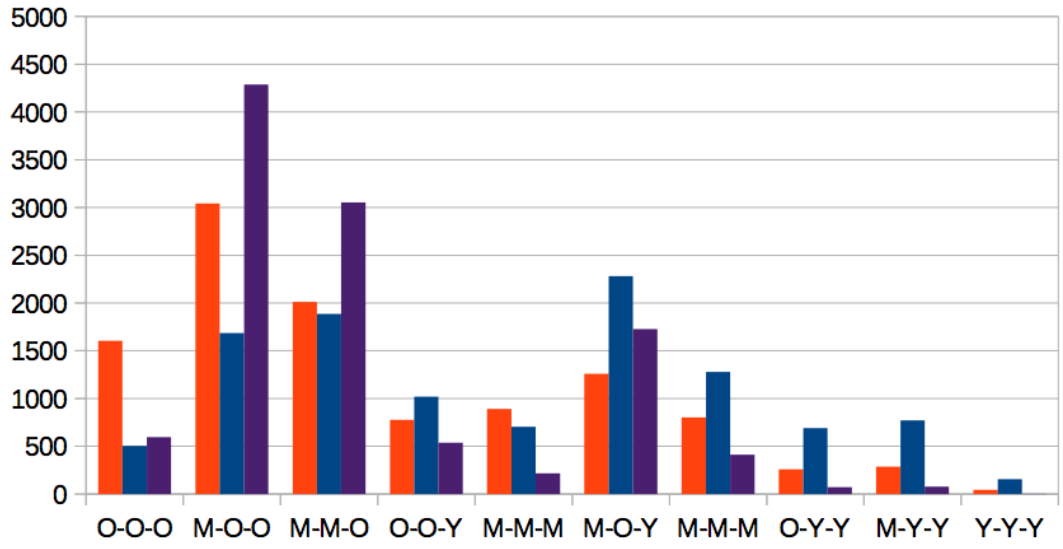
	Observed	Expected given nodes	p-values
O-O	4545	2622	1.71E-305
M-O	7043	5870	1.65E-050
M-M	3238	3286	8.75E-001
O-Y	2491	3549	4.92E-068
M-Y	2587	3973	2.04E-104
Y-Y	596	1201	1.02E-065



For triangles, the p-values are: under the node model, for the chi-squared test with 7 degrees of freedom, zero; under the edge model, for the chi-squared test with 4 degrees of freedom, zero. The contributions from the different triangles are shown in the next figure and the particular p-values indicated in the following tables.

	Observed	Expected given nodes	p-values
O-O-O	1605	502	0.00E+000
M-O-O	3043	1686	9.48E-232
M-M-O	2011	1887	3.19E-001
O-O-Y	778	1019	5.97E-010
M-M-M	890	704	2.18E-008
M-O-Y	1257	2281	3.19E-095
M-M-M	801	1277	6.95E-035
O-Y-Y	258	690	1.44E-054
M-Y-Y	287	772	6.18E-062
Y-Y-Y	43	156	7.01E-015

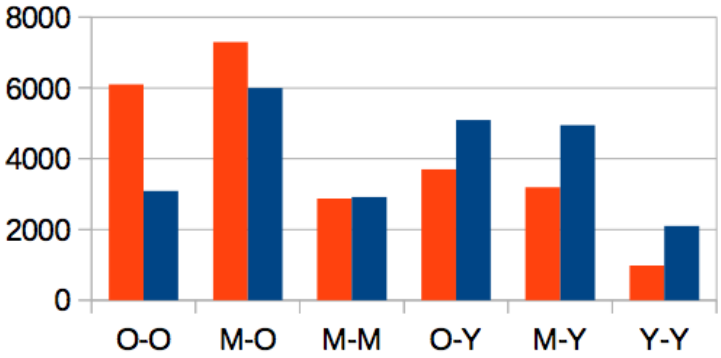
	Observed	Expected given edges	*p*-values
O-O-O	1605	595	0.00E+000
M-O-O	3043	4287	7.76E-077
M-M-O	2011	3054	7.85E-076
O-O-Y	778	536	1.20E-022
M-M-M	890	215	0.00E+000
M-O-Y	1257	1726	1.37E-026
M-M-M	801	412	3.52E-078
O-Y-Y	258	70	4.28E-107
M-Y-Y	287	76	6.56E-126
Y-Y-Y	43	1	9.50E-279



C.9 Complete DIP with a different protein age cut-off (Middle proteins now have a relative age between 0.4 and 0.8 inclusive)

For edges, the p-value for the chi-squared test with 3 degrees of freedom is zero. The contributions from the different edges are shown in the next figure and the particular p-values indicated in the following table.

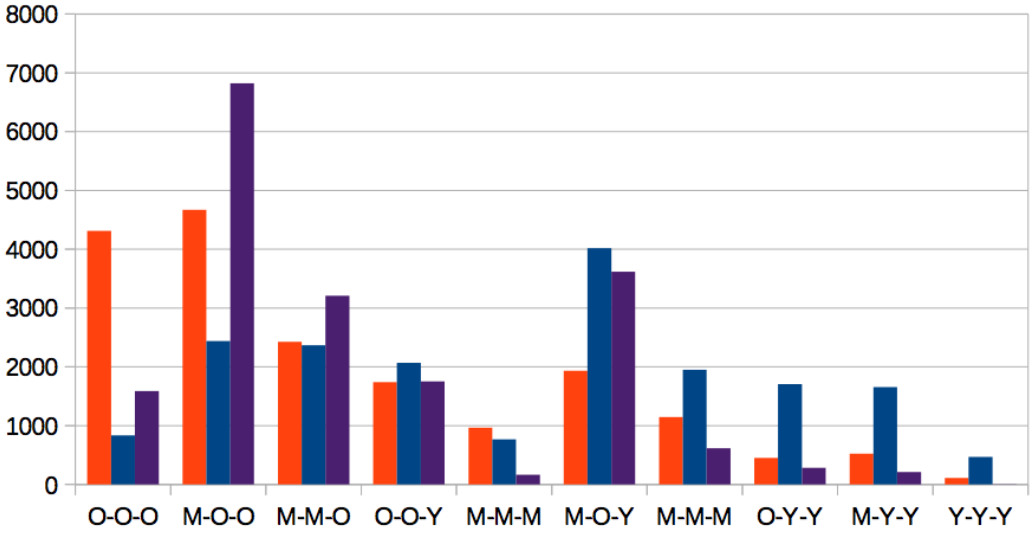
	Observed	Expected given nodes	p-values
O-O	6096	3088	0.00E+000
M-O	7292	5998	3.12E-060
M-M	2870	2913	8.90E-001
O-Y	3697	5089	3.01E-082
M-Y	3188	4943	9.41E-135
Y-Y	985	2097	1.62E-127



For triangles, the p-values are: under the node model, for the chi-squared test with 7 degrees of freedom, zero; under the edge model, for the chi-squared test with 4 degrees of freedom, zero. The contributions from the different triangles are shown in the next figure and the particular p-values indicated in the following tables.

	Observed	Expected given nodes	p-values
O-O-O	4313	837	0.00E+000
M-O-O	4670	2438	0.00E+000
M-M-O	2425	2368	9.86E-001
O-O-Y	1738	2068	4.15E-009
M-M-M	966	767	6.06E-009
M-O-Y	1934	4018	4.25E-229
M-M-M	1147	1951	1.14E-067
O-Y-Y	450	1705	4.25E-195
M-Y-Y	521	1656	1.27E-163
Y-Y-Y	110	468	2.01E-055

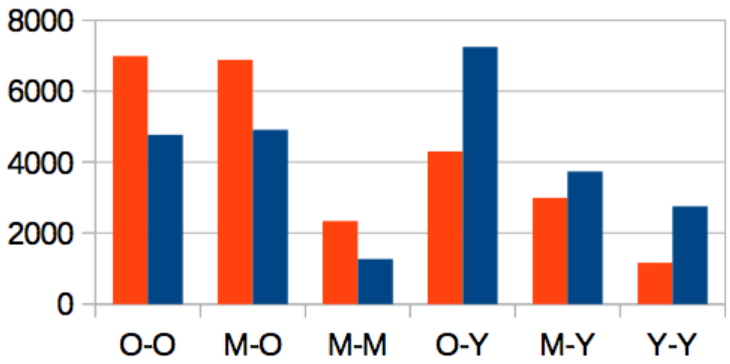
	Observed	Expected given edges	*p*-values
O-O-O	4313	1589	0.00E+000
M-O-O	4670	6822	1.41E-145
M-M-O	2425	3212	1.40E-040
O-O-Y	1738	1753	9.98E-001
M-M-M	966	166	0.00E+000
M-O-Y	1934	3617	2.98E-168
M-M-M	1147	614	6.69E-099
O-Y-Y	450	283	2.57E-020
M-Y-Y	521	211	1.29E-097
Y-Y-Y	110	7	0.00E+000



C.10 DIP data using the age definition of Kim and Marcotte (2008) and collapsing it in three age categories (see Chapter 3)

For edges, the p-value for the chi-squared test with 3 degrees of freedom is zero.The contributions from the different edges are shown in the next figure and the particular p-values indicated in the following table.

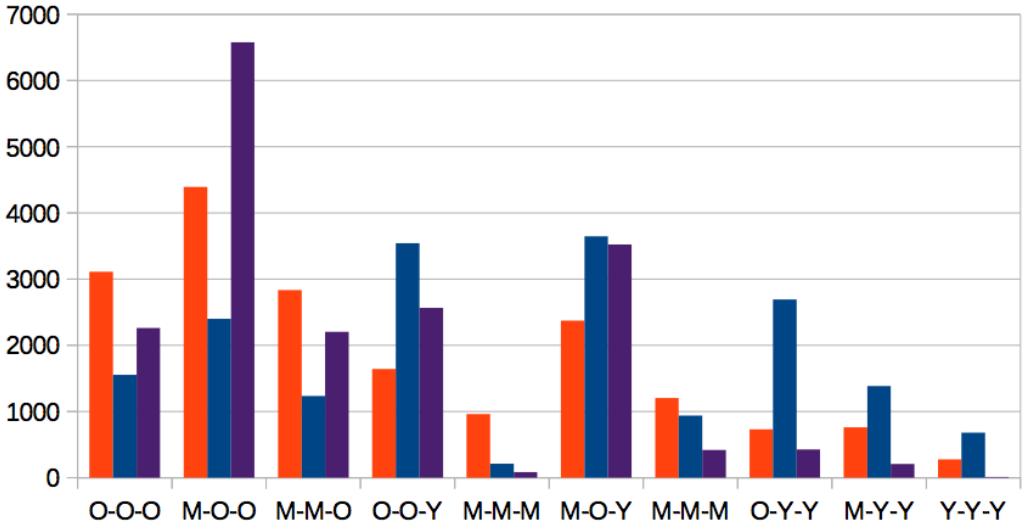
	Observed	Expected given nodes	p-values
O-O	6992	4770	3.56E-224
M-O	6882	4910	2.83E-171
M-M	2343	1264	1.94E-199
O-Y	4298	7248	4.78E-260
M-Y	2998	3731	5.07E-031
Y-Y	1164	2754	1.23E-198



For triangles, the p-values are: under the node model, for the chi-squared test with 7 degrees of freedom, zero; under the edge model, for the chi-squared test with 4 degrees of freedom, zero. The contributions from the different triangles are shown in the next figure and the particular p-values indicated in the following tables.

	Observed	Expected given nodes	p-values
O-O-O	3111	1554	0.00E+000
M-O-O	4391	2400	0.00E+000
M-M-O	2836	1235	0.00E+000
O-O-Y	1641	3542	4.66E-216
M-M-M	961	212	0.00E+000
M-O-Y	2373	3647	5.54E-092
M-M-M	1203	939	1.83E-013
O-Y-Y	731	2691	3.32E-304
M-Y-Y	762	1385	8.51E-057
Y-Y-Y	278	682	5.86E-048

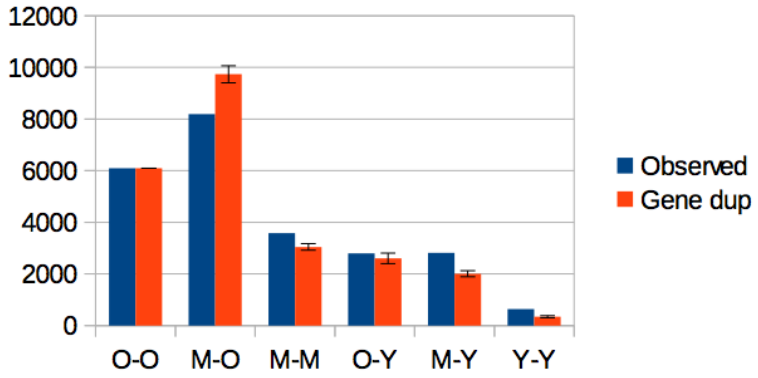
	Observed	Expected given edges	*p*-values
O-O-O	3111	2264	2.28E-067
M-O-O	4391	6579	3.16E-156
M-M-O	2836	2205	5.18E-038
O-O-Y	1641	2566	6.18E-071
M-M-M	961	85	0.00E+000
M-O-Y	2373	3524	4.76E-080
M-M-M	1203	418	0.00E+000
O-Y-Y	731	427	1.35E-045
M-Y-Y	762	208	0.00E+000
Y-Y-Y	278	10	0.00E+000



C.11 Complete DIP data set *versus* 30 networks generated by a gene duplication and divergence model [Bebek et al., 2006]

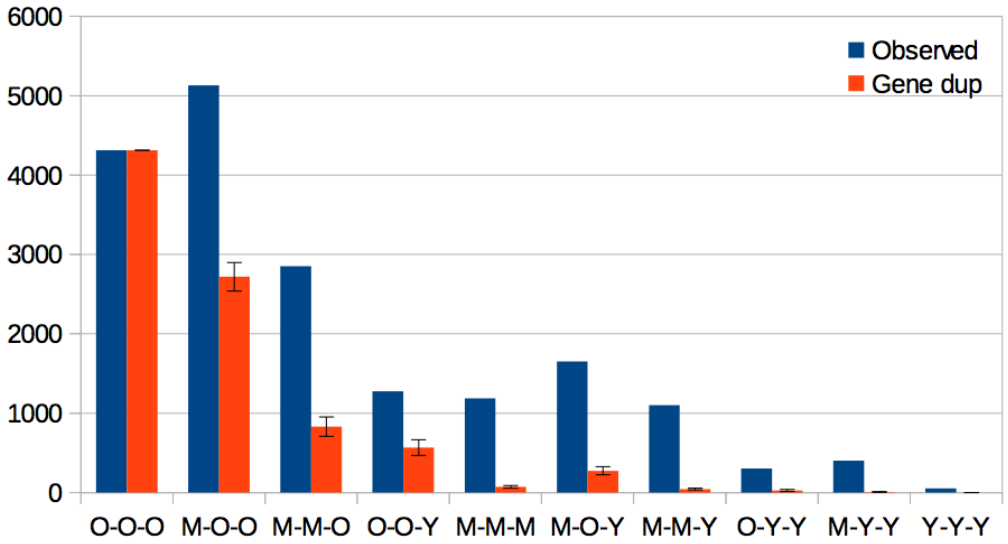
Here we report the age-dependent patterns found by growing the network through gene duplication and divergence. The "Observed" columns represent the counts found in the complete yeast DIP and the "Gene dup" columns are average counts over 30 generated networks, the error bars in the plots represent the standard deviation of these. O-O edges and O-O-O triangles are by construction equal to DIP. The differences between the various types of edges and triangles and their particular p-values are represented in the following figure and tables.

	Observed	Gene dup	p-values
O-O	6096	6096.0	–
M-O	8196	9734.2	1.70E-050
M-M	3584	3042.9	3.33E-019
O-Y	2793	2603.8	1.73E-002
M-Y	2817	2005.9	9.79E-069
Y-Y	642	343.7	6.50E-054



The contributions from the different triangles are shown in the next figure and the particular p-values indicated in the following tables.

	Observed	Gene dup	p-values
O-O-O	4313	4313.0	–
M-O-O	5131	2718.7	0.00E+000
M-M-O	2852	831.0	0.00E+000
O-O-Y	1277	566.8	8.76E-186
M-M-M	1187	73.8	0.00E+000
M-O-Y	1653	275.3	0.00E+000
M-M-Y	1103	43.9	0.00E+000
O-Y-Y	304	27.1	0.00E+000
M-Y-Y	402	7.9	0.00E+000
Y-Y-Y	52	0	–


6000
5000
4000
3000
2000
1000
0
Observed
Gene dup
O-O-O
M-O-O
M-M-O
O-O-Y
M-M-M
M-O-Y
M-M-Y
O-Y-Y
M-Y-Y
Y-Y-Y