MATHEMATICAL CHARACTERIZATION OF INTERFACE INTERACTION NETWORKS

A thesis submitted to Johns Hopkins University in conformity with the requirements for the degree of Master of Science in Engineering

by
Benjamin H. Shapiro

Baltimore, Maryland
May, 2016

Abstract

Biological systems carry out complex functions such as DNA transcription, cell division, and signaling within and between cells through the collective effort of a diverse range of proteins. In these functional pathways, proteins both cooperate and compete with one another to bind their partners, such that the outcome depends on the protein concentrations, regulation of binding sites, and the number of partners per protein. A tool to help scientists visualize some of these interdependences is known as a protein-protein interaction (PPI) network. In these networks, nodes represent proteins and a line is drawn between two nodes if the proteins interact. These networks have allowed for the visualization and analysis of complex systems yet still fail to accurately capture the competition between proteins.

Interface-interaction networks (IINs) are a modification of PPI networks that capture such competition. These networks are distinctive because they are constrained by the parent PPI network, and they therefore have novel attributes that have not been previously characterized theoretically. In these networks, nodes represent binding sites and two nodes are connected if the binding sites interact. The structure and topology of the IIN may reflect evolutionary pressures on individual proteins because proteins evolve to recognize and bind their functional partners in specific ways. If the IIN is under evolutionary pressure, it should exhibit unique structural properties. To recognize unusual and unexpected features of a network, we must understand the topology of these networks that would arise under purely random conditions, which will be the goal of this thesis.

We will begin by rigorously defining the space of all possible IINs and then enumerate such a space. We will find the number of possible networks to be on the order of 10^{177}. Using this knowledge, we will develop a statistical test to determine if an IIN's topology is under selective pressure. This test will be applied to the clathrin-mediated endocytosis (CME) IIN in yeast and will allow us to conclude

that this network is under selective pressure.

Finally, we will characterize the global structural properties of the random networks and compare them to the CME IIN. We will find that the CME IIN has a unique scale free distribution. We will conclude with a brief discussion on local motifs and the difficulties involved with delineating their prevalence in IINs.

Research Advisor: Dr. Margaret Johnson **Reader:** Dr. Amitabh Basu

Acknowledgement

I am incredibly grateful for the support and mentorship of Professor Margaret Johnson. She introduced me into the world of research and her guidance and encouragement has not only allowed me to write this thesis but has changed my goals and aspirations. It has been a truly incredible experience to learn from someone who is so passionate about the pursuit of new knowledge and teaching others.

I would also like to thank David Holland for all of his help in this research and the many hours he has dedicated to helping me compile and debug code. He has taught me so much and is always willing to take the time to provide assistance.

Lastly, I would like to thank my family for their unending support and love. They have always been by my side and have done so much to help me accomplish my goals.

Contents

List of Tables

List of Figures

1 Introduction

Protein-protein interaction (PPI) networks have traditionally been used to render a visual representation of the complex interactions between a system of many proteins. In these networks, nodes represent the proteins in a biological system and edges represent the binding interactions between proteins. The use of these networks goes much further than visualizing complex systems. The spatial patterns in the networks can be mathematically quantified to predict a wide range of phenomenon such as the molecular basis of disease [12].

However, these PPI networks fail to adequately describe the spatial arrangement of binding interactions. This is because no distinction is made between binding sites. For example, if two proteins interact with a third, the PPI network will not distinguish if these proteins bind to the same spot on the third protein or to different spots. These interactions are vital to understanding competition between proteins.

1.1 Interface-Interaction Networks

Interface-interaction networks (IINs) capture the dimensionality between binding sites that is lost in PPI networks. In an IIN, nodes represent binding sites and edges represent interactions between binding sites. IINs can therefore be seen as resolved PPIs. Rather than the looking at the interactions at the level of a protein as a PPI does, the IIN considers interactions at the level of the interfaces for each protein. Like a PPI network, the structural characteristics of an IIN can be quantified to predict biological function.

In previous work the clathrin-mediated endocytosis (CME) IIN in yeast was constructed using both structure-based computational approaches and biochemical data [14]. The IIN was then analyzed for specific features and the significance of these features was quantified via the comparison with random networks of the

same size. The possible origins of these features are discussed in Section 1.2.

The structure of the IIN was characterized by its degree distribution, density, modularity, and local motif structure, including clustering. Although the degree distribution was similar to a PPI network in the sense of being scale-free, the other features were quite distinct.

To quantify the significance of these features and the probability they occur by random chance, the CME IIN was randomized through a rewiring process that swapped edges between nodes. This procedure allowed the network to maintain the total number of edges and nodes along with the degree distribution.

Using the above procedure, randomly generated IINs were constructed from the CME IIN. It was found that the CME IIN contained certain characteristics that were difficult to sample under randomness. For example, the CME IIN contained interesting global properties such as a diverse distribution of module sizes that were distributed according to a power law and a low clustering coefficient. In addition, the network contained a unique distribution of local structural elements such as an abundance of 4-node hubs and a scarcity of 4-node chains.

These properties suggest that there are evolutionary mechanisms acting on the topology of the CME IIN. More specifically, these features were hypothesized to be a result of evolutionary pressures to minimize nonspecific interactions [13].

1.2 Evolutionary Pressures & Origins of IIN Structural Features

Previous work by Johnson et al. [13] discussed how network structure must follow from function. For example, one would expect PPI networks to maintain global connectivity so different module pathways can communicate with one another. On the other hand, it is not imperative that an IIN maintain such a global connectivity if the overlaying PPI network accomplished such a task. It was hypothesized

that the major functional property of an IIN is to maximize proper binding. It follows from this that network structure should maximize highly specific (functional) interactions while minimizing non-specific (non-functional) interactions. To accomplish such a task, the network topology must be under evolutionary pressure.

A Monte Carlo global optimization procedure was developed to sample for networks that favor certain local features while maintaining the same degree distribution. The procedure was employed for a range of generated networks to classify which network features allow highly selective protein binding interactions. For each network, the interfaces were assigned sequences that were then mutated and optimized to ensure strong functional binding and weak non functional interactions as described in [13].

The properties of the IINs with the highest selectivity and binding were very unique. It was found that larger networks result in overall higher binding specificity despite the quadratic increase in nonspecific interactions. Furthermore, the more scale-free the degree distribution, the higher the overall binding specify, up to a point. Locally, the optimized networks with high specificity have fewer cliques, more hub motifs, and very few chain motifs (hub motif is 3 interfaces all incident to a fourth and a clique is a sub network in which all proteins are pairwise adjacent).

1.3 Random IINs

An IIN is a network within a PPI network. Therefore, IINs are constrained by their parent protein network. Up to this point, the simulations discussed to sample random IINs have preserved the degree distribution but have not preserved the constraints of the protein network. We are interested in exploring the space of possible IINs given a PPI network. Figure 1 displays a simple example of all possible IINs for a protein network with 3 proteins and 2 interactions. As more edges and nodes are added to the protein network, the space of IINs gets increasingly

more complicated.

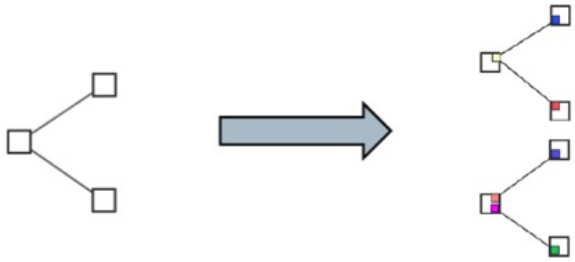

Figure 1: An example of a PPI network with 3 proteins (left) and its two possible interface networks (right) overlaid on the PPI network.

Little research has been done regarding the entire space of IINs given a PPI network. While, as previously discussed, we have a good idea of what evolution should select for in an IIN, we know little of what an IIN would look like without such pressure. It is important to characterize such random networks to understand the topological properties of IINs that should arise without evolutionary pressure. Furthermore, characterizing such networks will give additional probabilistic and statistical approaches to classifying whether there exists pressure on structural properties of an IIN.

To characterize such a space, however, is difficult. IINs are subject to a wide array of node specific constraints that cannot be relationally expressed. Furthermore, there is no function that maps between the two types of networks. For a given PPI network, there are many IINs. Given an IIN, one would not be able to determine its parent PPI network, as there could be many that produced it. There are two approaches to combat these challenges.

One method is to computationally sample IINs constrained to a PPI network through a Monte Carlo method. Recently, the Johnson Group laboratory has developed such a method. In this method, a PPI network is inputted and the simulation transverses the space of all possible IINs given the PPI network. To accomplish this, the method begins with an IIN identical to the inputted PPI net-

work, for example. The algorithm then creates a new network with an interface duplication or an interface combination at each iteration. In an interface duplication, an interface is split into two interfaces and edges are randomly assigned in accordance with the principles of detailed balance to the two new interfaces. In an interface combination, two interfaces in the IIN are combined into one interface and all edges incident to each interface are now incident to the combined interface. It is important to note that interface duplications and combinations can only occur with interfaces incident to edges that are all incident to the same protein in the parent PPI network.

The Monte Carlo method accepts or rejects a move, such as interface duplication or combination, based on the Boltzmann weight $e^{\frac{-f}{k_b T}}$. By setting the temperature to infinity, all possible moves are accepted, generating randomized, constrained, IINs.

Another method to characterize the space of IINs constrained to a PPI network is through theoretical calculations, which are described in this thesis. A main interest is how many IINs are possible for a given PPI network. We will develop the theoretical framework for this calculation and carry out the calculation on the CME PPI network. Surprisingly, the space of IINs is enormous — with small protein networks having more possible IINs than atoms in the universe!

Furthermore, we want to develop a generalized statistical method to determine if some outside pressure shapes a given IIN's topology. We will create a method that is easily generalizable to all IINs. We will apply this method to the CME IIN and discover that the topology of the IIN is under selective pressures, which fits with previous results of the Johnson Group laboratory.

Lastly, we would like to explore the local and global characteristics of random IINs and understand how these structures compare with IINs found in nature. We expect distinct features compared to fully unbound networks, however, the extent of the effects of the constraints is unclear. We will specifically examine

these properties in the context of the CME IIN. We will develop a method to parameterize a scale-free distribution so we can measure the relative scale-freeness of sampled IINs and the CME IIN. We can, in turn, answer the question of whether the degree distribution of the CME IIN is a result of simply the constraints of the parental PPI network or a result of specific pressures acting on the topology of the IIN. We will conclude by discussing local motifs that we expect to see in random IINs.

It is important to keep in mind the motivations of such a theoretical framework extend beyond IINs and PPI networks. Our methodology can be applied to any network that can be modeled in such a constrained fashion. One example of a network with similar constraints is the network of airports. One can view terminals or runways as interfaces and airports as proteins. Airline logistics relies heavily on graph theory and our model might find use in this industry. Another example is social network groups. One can view an individual as a protein connected by edges to other individuals they know. Interfaces can represent social circles such as business acquaintances, family, and friends. These are only a few of the additional applications but we suspect there are many more.

2 Enumeration

In the following section we will develop a framework to enumerate the number of IINs given a PPI network. First, we will define the concept of randomness in a constrained IIN. Next, we will reduce the enumeration problem to a question of partitioning sets. Finally we will derive, a simple, closed form, solution.

2.1 Randomness

While a PPI network displays the interactions between proteins, an IIN displays the interactions between specific binding sites on proteins. To create an IIN, one must first have a PPI network and determine what interfaces on the constituent proteins interact to form the edges of the PPI network. One then uses this information to form a new network in which interfaces become the nodes and edges represent the interactions between specific interfaces.

There are many different ways in which a protein can have interfaces. For example, a protein may interact with 10 proteins through one binding site or through 10 separate binding sites. In the former case the IIN would contain one node for that protein incident to 10 edges while in the latter it would contain 10 nodes, each incident to one edge.

We would like to know the number of IINs possible for a given PPI network. To do this, we must first consider how to generate some IIN from the space of all possible IINs for a PPI network. We must generate the network in such a way that any possible IIN is created with equal probability.

The motivation of such a method comes from a popular model of generating a random network given some number of nodes and edges. This model for generating random networks is known as the Erdős — Rényi model [7]. In this model, to generate a random network that has n nodes and N edges, the connectivity is chosen with equal probability from the $\binom{\binom{n}{2}}{N}$ possible combinations of edge place-

ments. However, this model falls short for enumerating the space of IINs for a given PPI.

To delve further into this concept, specific notation will be introduced. Define G as some PPI network. Furthermore, let $E(G)$ and $V(G)$ be the edge and node sets respectively for this network and define I_G to be the set of all possible IINs for G.

The conditions for the Erdős — Rényi model are not met for creating random IINs as, even though for all $X, Y \in I_G, |E(X)| = |E(Y)|$, it is not necessarily true that $|V(X)| = |V(Y)|$. A protein in a PPI network could have some number of interfaces inclusively between 1 and the degree of that protein.

Let us now further examine our set of network I_G. This set contains every possible IIN for a PPI network G. To form such a set we could look at every $v \in V(G)$ and form a family v_G which contains all combinations of interfaces and ways in which edges can be incident to such interfaces for protein v. Picking one combination of interfaces/edges from each v_G will form a unique IIN. Furthermore, constructing IINs from all possible combinations of picking one element from each v_G for all $v \in G$ will form I_G. This is trivial to prove using the law of total probability.

Pick some random IIN $X \in I_G$. Define $a, b \in V(G)$. The first thing to note is that the number of interfaces defined in X for a is independent from the number defined for b. That is, $P(X$ contains r interfaces for $a|X$ contains m interfaces for $b) = P(X$ contains r interfaces for $a)$ for some $r, m \in \mathbb{N}$. Furthermore, the connectivity of edges to interfaces of protein a is independent from that of b. These are direct results of our algorithm to generate I_G in the proceeding paragraph.

We have now laid the groundwork to calculate $|I_G|$ through these independence conditions. Our strategy will be to model a specific protein and then extend our model under the independence conditions.

2.2 Bell Numbers

We will take a detour from networks to introduce an important combinatoric concept known as Bell numbers.

Definition. Bell Numbers: We denote the n^{th} Bell number as B_n. This value represents the number of ways to partition a set with n elements into non-empty, disjoint subsets.

$$B_{n+1} = \sum_{k=0}^{n} \binom{n}{k} B_k, \quad B_0 = 0 \tag{1}$$

[19]

2.3 Solution for the Number of IINs Given a PPI Network

Now that we have developed the independence condition in random IINs for the number of interfaces assigned to each protein and have introduces Bell numbers, we are ready to calculate $|I_G|$.

Theorem. The cardinality of I_G : Given a PPI network G, with the i^{th} protein represented by v_i, the number of IINs possible is as follows:

$$|I_G| = \prod_{i=1}^{|V(G)|} B_{deg(v_i)} \tag{2}$$

Proof. Take $u \in V(G)$. Next, we create a set, M, of all edges incident upon node u represented as follows: $\{e \in M | e \in E(G) \wedge e \in u \quad \text{(e is incident to u)}\}$ There can be $1, 2, \cdots, deg(u)$ interfaces on this protein. We want to calculate the number of possible ways edges can be incident on this node (i.e: are all edges incident upon one interface, on unique interfaces, or some other combination).

Without loss of generality we will assume that protein u has t interfaces. Each interface must have some set of edges incident upon it and the union of all such sets must be equal to M. All sets must be pairwise disjoint as we can not have one edge incident to two interfaces on the same node. Therefore, M is the disjoint

union of all sets. By definition, a t partition creates t disjoint non-empty subsets. The number of ways edges can be incident upon the t interfaces is therefore the number of t partitions that can be created from the set M. By taking the sum of the number of partitions from $t = 0$ to $t = deg(u)$ we arrive at $B_{deg(u)}$.

In the previous calculation we have assumed that an edge can be incident to only one interface on a specific protein. However, a network with self-loops does not fit this criterion as a self-loop edge may be incident to two different interfaces or be twice incident to the same interface on the same protein. We will generalize our proof to include networks with self-loops.

Take $u \in V(G)$. Suppose protein u has one or more self-loops. Because self-loops can be incident upon any interface on a protein we can treat a self-loop as two separate edges, both incident to u. Thus, for every self-loop we add one extra unique edge to M. However, we will adopt the convention in this thesis that a self loop adds 2 to the degree of a node. This means that we arrive back to the same result of $B_{deg(v_i)}$.

So far we have determined the number of possible ways interfaces could appear with respective edge assignments in a specific node of the PPI network. We know from our previous discussion on randomness that both the number of interfaces and the connectivity of interfaces for each v_i is independent of one another from a network chosen uniformly from I_G. Therefore, because the structure of one protein is independent of the structure of any other protein, the number of IINs, $|I_G|$, that can be created is simply the product of the number of ways each protein can be structured.

□

2.4 Illustrated Example with the CME Network

It would be of interest to apply our result to a real world PPI network to determine the exact number of possible IINs for such a network. We will use the CME PPI

Table 1: CME PPI degree distribution

Degree	Number of Proteins
1	4
2	6
3	8
4	7
5	4
6	5
7	4
8	5
9	1
11	2
12	3
13	2
15	1
16	1
18	1
20	1
24	1

network whose degree distribution is displayed in Table 1. Furthermore, for each value in the left column of Table 1, the corresponding Bell number was calculated and is displayed in Table 2.

On inspection, Bell numbers grow incredibly fast. However, it has been shown that a Bell number, B_n is bounded by $(\frac{.792n}{ln(n+1)})^n$ [6]. To calculate the number of IINs possible we simply apply our formula by taking the product of each Bell number, B_n, raised to the power of the respective amount of proteins that have a degree equal to n. In doing so, we arrive at the following solution:

$$\text{Number of Possible IINS for the CME PPI network} = 9.8 \times 10^{176}$$

For your reference, this is approximately 1×10^{173} times the number of calories Michael Phelps eats in a day [16], or about 1×10^{93} times the number of atoms in the universe! This is a huge number of possible networks. In coming chapters we will analyze the types and features of the networks that populate this very large

Table 2: Bell Numbers

N	B_N
1	1
2	2
3	5
4	15
5	52
6	203
7	877
8	4,140
9	21,147
11	678,570
12	4,213,597
13	27,644,437
15	1,382,958,545
16	10,480,142,147
18	682,076,806,159
20	51,724,158,235,372
24	445,958,869,294,805,289

space.

It is also worth noting that the number of possible IINs is larger for a PPI network with a scale free degree distribution than a PPI of the same size but with a binomial degree distribution. For example, if the 56 protein of the CME network were connected by the same number of 186 edges, but each protein had 6 edges with a few making up the difference, one observes a significantly smaller (although still huge!) number of networks.

3 Randomness of the CME IIN

It was of great interest to determine whether the CME IIN is under evolutionary pressure. In previous work it has been shown that the frequency of motifs of the CME IIN significantly differed from those of randomly generated IINs which points to such pressures [14]. To arrive at this conclusion, sophisticated rewiring procedures and Monte Carlo methods were created. The algorithms took a great deal of time and computational expense to deploy. Furthermore, these algorithms never considered the constraints introduced by the parent PPI network.

We would like to generate a generalized statistical test that can be used without sophisticated computational tools to determine if such evolutionary pressures exist over any given IIN. Furthermore, we want this statistical test to take into account the structural properties introduced by the parental PPI network. That is, we want to know if the IIN is under evolutionary pressure independent of whether the PPI network is under selective pressure. We will construct such a test and we will validate our test on the CME IIN that we know is under such pressure.

The statistic we have defined has another practical use in quantifying whether an individual protein has an unexpected number of interfaces given its number of interaction partners. The metric is based only on the number of ways to partition the interactions of a protein and none of the biophysical properties of a protein that determine binding. Nonetheless, it is a potentially useful and straightforward way to single out proteins with unexpectedly many or few interfaces in a consistent manner.

It is important to note that the test we develop here treats proteins independently, and therefore is not measuring the same degree of randomness as the previous work, which considered the connectivity of the IIN. For example, this test is not designed to compare unusual motif structure because it does not consider pairs of proteins. Instead, it tests whether the partitions of the protein interactions

into interfaces is random or not.

3.1 Formalization

We will use the notation introduced in the previous chapter. We let G represent some protein interaction network and I_G represent the set of all possible IINs of G. Furthermore, we pick some $C \in I_G$ to represent the actual interface network of G found in nature. Our question is to determine whether C is under some evolutionary pressure. We can rationalize that, if such a pressure existed, then we would expect unique features to exist over the topology of C. These unique features are selected by evolution to maximize or minimize some property throughout the network, such as minimizing nonspecific interactions [13]. This implies that if C is under evolutionary pressure, the number of interfaces on $a, a \in V(G)$, is dependent on the number of interfaces on $b, b \in V(G)$. This implication puts us in a unique position to create a statistical test that leverages our results developed in the previous section.

3.2 Global Hypotheses

We must create some null hypothesis that, if we reject, implies dependence between the number of interfaces on each protein of G. Furthermore, the distribution of some test statistic under our null hypothesis must be easily quantifiable. We will propose the following global hypotheses and develop a method of multiple testing to reject or accept the global null hypothesis.

Null Hypothesis. C results from a random partitioning of the protein network, G, into interfaces (i.e: chosen uniformly from I_G)

Alternative Hypothesis. C results from a nonrandom partitioning of the protein network, G, into interfaces

In other words, the null hypothesis states that we expect C to be chosen from I_G uniformly while the alternative hypothesis states the C is chosen from I_G according to some other distribution. More rigorously, it means under the null hypothesis $P(X = C)$ is uniformly distributed where X is a random variable, $X \in I_G$, in contrast to some other distribution under the alternative hypothesis. If we reject the null hypothesis then it means that we have a higher probability of picking networks with certain features not reflected when picking networks uniformly from I_G.

We note that this test only considers the partitioning of interfaces and not the actual specific connectivity of interfaces. That is, failure to reject the null hypothesis doesn't necessarily mean that C isn't under pressure in other ways. Pressures that this test may not pick up on, for example, are whether certain domains of interfaces favor interactions with other domains. However, if we reject the null hypothesis, our test implies that pressures exist on the number of interfaces each protein has.

To use these hypotheses we must create a test static, U, on some feature of the network and calculate its distribution under the null hypothesis. However, finding such a feature and determining its distribution under the null hypothesis is an incredibly difficult task to do analytically. Luckily, we will develop an approach to avoid calculating a global test static by reducing the problem to one of multiple testing.

3.3 Multiple Testing

We will introduce a multiple testing framework that can be applied to our proposed problem [10]. Let $H_1, ..., H_n$ be a collection of hypotheses that we are interested

in testing. We will define a global hypothesis to be as follows:

$$H_{global} = \bigcap_{i \in I} H_i \quad , \quad I = \{1, 2, ..., n\} \tag{3}$$

The global hypothesis is true whenever all $H_i, i \in I$, are true. However, if there exists any $H_j, j \in I$, that is not true then H_{global} is not true.

We would like to test the global hypothesis at some significance level α by testing each null hypothesis, H_j , at some level β dictated by α. We denote α as the family wise error rate. Using a Bonferroni correction [18] the family wise error rate is bounded by α by setting $\beta = \frac{\alpha}{n}$ This bound is very conservative and therefore should only detect highly non-random or specialized solutions. A less stringent criterion could be implemented using the Benjamini Hochberg [5] procedure which controls the false detection rate rather than the family wise error rate.

In summary, we can test a much larger (global) hypothesis by breaking such a hypothesis into smaller hypotheses whose intersection results in the global hypothesis. A significance level, α, is then set for our global hypothesis which defines the probability of type I error. When this hypothesis is broken into the family of smaller hypothesis, α becomes known as the family wise error rate. Each smaller hypothesis can be testing using a Bonferroni correction at a significance level of $\beta = \frac{\alpha}{n}$ which will ensure that our family wise error rate remains bounded by α. If we reject any smaller hypothesis then we also reject our global hypothesis.

3.4 Breaking Down the Problem

We will now define smaller hypotheses whose intersection is the previous global null hypothesis we proposed. Recall that our null hypothesis was that C is a IIN formed from random partitioning - chosen uniformly form I_G. Let $|V(G)| = n$. We propose the following null hypotheses for $i \in \{1, 2...., n\}$:

H_i**.** The number of interfaces of v_i defined by C is random where v_i is the i^{th} protein of G

We note that the intersection of all of these hypotheses results in the hypothesis that for all proteins $v \in G$, the number of interfaces on each defined by the IIN C is random. Also, recall from the previous chapter that for some C chosen uniformly from I_G, the number of interfaces and edge assignments to such interfaces for each protein in G are pairwise independent! This means we can test each H_i by analyzing each protein's interface splitting pattern in isolation of all others.

We will walk through the strategy for some protein $v \in V(G)$ and derive the necessary distributions. We know that v can have any number of interfaces inclusively between 1 and $deg(v)$ from our previous discussion. However, we know that some of the possibilities for the number of interfaces on v are more likely than others. For example, there is only one-way to partition the edges incident to v into one interface while there are many ways to partition the edges incident to v among some number of interfaces greater than 1. We want to derive this exact probability distribution. That is, we want to find $P(interfaces(v) = X)$ where $X \in \{1, 2, ...deg(v)\}$. We have actually done most of the legwork already for this calculation in our enumeration of I_G. However, we are missing one key combinatoric concept that we will now introduce:

Definition. Stirling Numbers of the Second Kind: The number of ways to partition n items into k non-empty subsets. This value will be denoted as $S(n, k)$.

$$S(n,k) = \frac{1}{k!}\sum_{i=0}^{k}(-1)^i\binom{k}{i}(k-i)^n \tag{4}$$

[17]

It is easy to see then that:

$$B_n = \sum_{k=0}^{n} S(n,k) \tag{5}$$

It is now a simple jump to determine $P(interfaces(v) = X)$. We know that there are $B_{deg(v)}$ possible interface and edge combinations for v as proven in section 2. Furthermore, we would like to know how many ways there are to partition $deg(v)$ edges among X interfaces. This is just the Stirling number, $S(deg(v), X)$. To reiterate, $P(interfaces(v) = X)$ is simply the number of possible ways v can have X interfaces over the total number of all possible interface and edge combinations for v. We can conclude the following:

$$P(interfaces(v) = X) = \frac{S(deg(v), X)}{B_{deg(v)}} \tag{6}$$

Using this expression we can generate the probability mass function for interfaces given a protein and its corresponding degree. In Figure 2 the probability mass function for the number of interfaces under our null hypothesis for a protein of degree 20 is displayed. The distribution is unimodal with a clear positive skew as the maximum density is found with the protein having only 8 interfaces. Figures 3 and 4 display distributions for other degree values. The skew is more pronounced for proteins with higher degree.

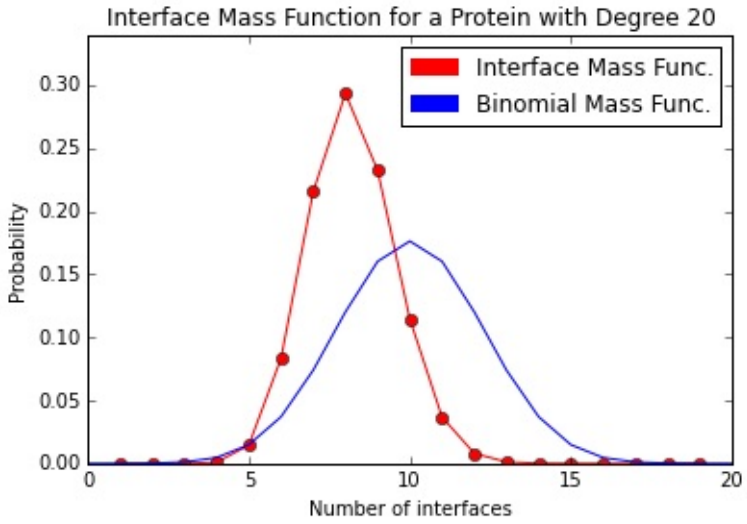

Figure 2:
The probability mass function for $P(interfaces(v) = X)$ where $degree(v) = 20$ (v represents a protein) is plotted in red. The distribution is unimodal and highly left skewed. For reference, the binomial distribution with $n = 20$ and $p = 1/2$ is plotted in blue. It is clear that the interface probability distribution's mass is contained almost entirely between only a few values of X in contrast to the binomial distribution which has more mass contained within its tails.

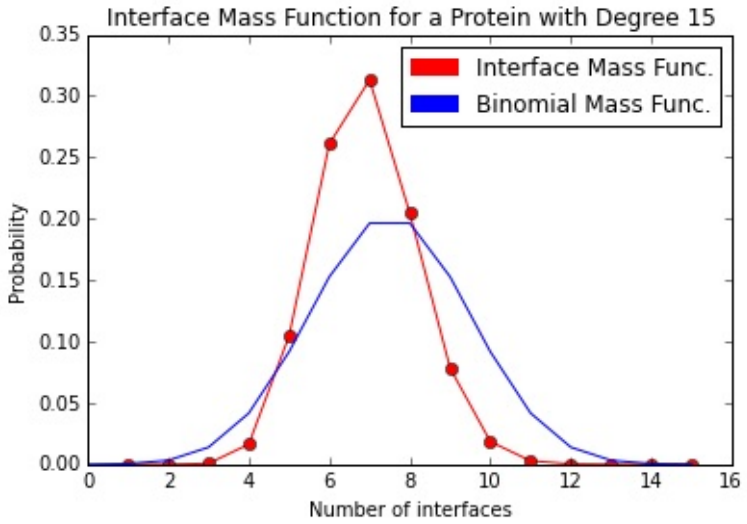

Figure 3:
The probability mass function for $P(interfaces(v) = X)$ where $degree(v) = 15$ (v represents a protein) is plotted in red. The distribution is unimodal and slightly left skewed. It is very similar to the distribution in which $degree(v) = 20$. For reference, the binomial distribution with $n = 15$ and $p = 1/2$ is plotted in blue.

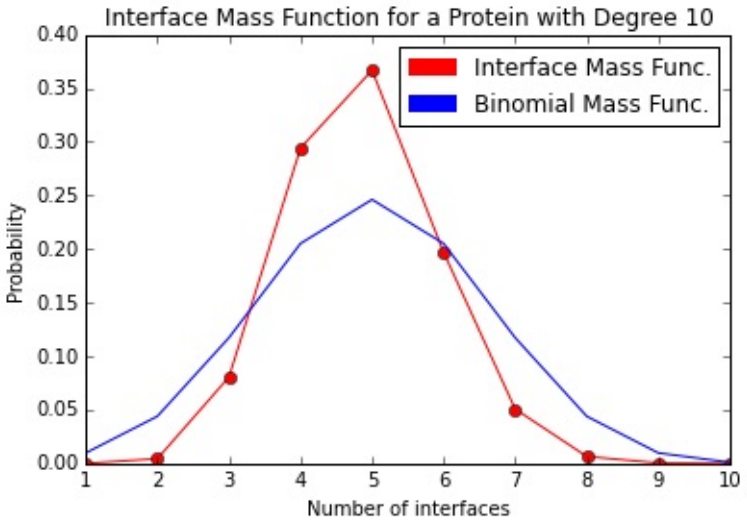

Figure 4:
The probability mass function for $P(interfaces(v) = X)$ where $degree(v) = 10$ (v represents a protein) is plotted in red. The distribution is unimodal and and left skewed like the previous distributions for $degree(v) = 15$ and $degree(v) = 20$. For reference, the binomial distribution with $n = 10$ and $p = 1/2$ is plotted in blue.

3.5 Test Statistic for a Protein Interface Distribution

Let us create a test statistic under H_i which will allow us to reject H_i at some significance level β. An obvious test statistic, U, is the number of interfaces present in the IIN, C, which are on protein v_i. The distribution of U is given by Equation 6.

We will perform a two-sided test using our test statistic such that we reject the null hypothesis only when the probability of seeing a value more extreme than U is less than β. This equates to calculating the two-sided tail area of our distribution for the interfaces of v_i. The following equations calculate this two-sided area:

$$\text{For } U \neq median\{1, 2..., deg(v_i)\}:$$

$$P(t \leq median\{1, 2..., deg(v_i)\} - |U - median\{1, 2..., deg(v_i)\}|) \quad + \\ P(t \geq median\{1, 2..., deg(v_i)\} + |U - median\{1, 2..., deg(v_i)\}|) \quad \text{(Eq. 7)}$$

$$\text{When } U = median\{1, 2..., deg(v_i)\}:$$

$$\text{P(more extreme (or equal) number of interfaces than } U) = 1$$

The tail probabilities are illustrated for $U = 8$ in Figure 5 on $deg(v_i) = 20$. The asymmetry and skew of the distribution are highly evident as little area lies beyond $t = 13$ compared with the density below $t = 8$.

We have thus developed a test to accept or reject each H_i. We will illustrate using a simple example then extend our testing to the global hypothesis for the CME IIN.

We note that the partitioning of edges on each protein only takes one form (i.e. a protein with 20 edges has 2 interfaces), but the origins of each edge are not considered, nor is the number of edges per interface. Hence, when the whole network of proteins is considered, the statistic actually describes a whole family of IINs, of which the target IIN is one instance. If our statistic proves the family

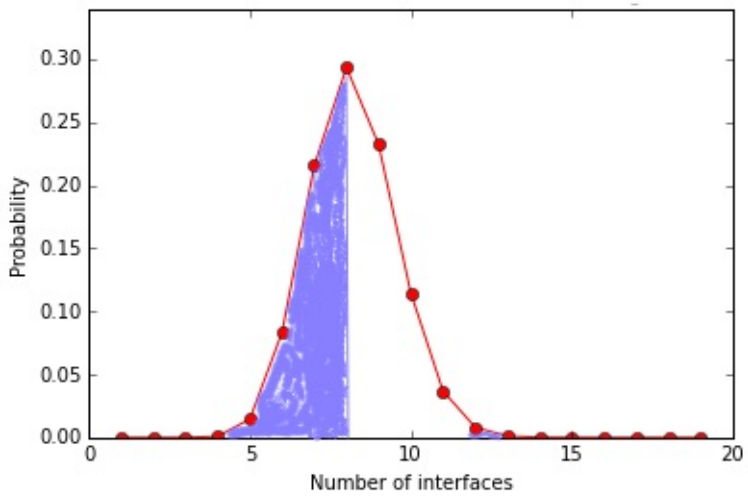

Figure 5:
The probability mass function for $P(interfaces(v) = X)$ where $degree(v) = 20$ (v represents a protein). The shaded area represents the tail probabilities of Eq. 7 for the test statistic $U = 8$.

is non-random, this also indicates our IIN is non-random. However, if it finds the family is consistent with a random sampling, then further testing would be needed to establish the uniqueness of the target IIN.

3.6 Hypothesis Testing Example

Suppose we have some protein network G and $v \in G$ has a degree of 20 and has 2 interfaces. We would like to test the following hypotheses:

Null Hypothesis. The 2 interfaces on v are a result of randomness

Alternative Hypothesis. There is nonrandom process that results in v having 2 interfaces

We now calculate the probability that v would have 2, or a more extreme amount of interfaces under randomness. In this case our test statistic, U, is 2. We use Equation Eq. 7 to calculate this probability: $P(t \leq 10.5 - |2 - 10.5|) + P(t \geq 10.5 + |2 - 10.5|) = P(t \leq 2) + P(t \geq 19) = \frac{S(20,1)}{B_{20}} + \frac{S(20,2)}{B_{20}} + \frac{S(20,19)}{B_{20}} + \frac{S(20,20)}{B_{20}}$ Using computational software this expression can be evaluated to arrive at the

solution of $1.013\,992\,335\,290\,11 \times 10^{-8}$. We can clearly reject the null hypothesis at the level of $\beta = .01$. Because seeing this number of interfaces on such a highly connected node is so rare we conclude that there is some non-random process that resulted in the protein using only two interfaces for its many partners.

3.7 Testing On the CME IIN

We will perform the previous hypothesis test on each protein in the CME IIN and its corresponding number of interfaces. Table 3 gives information about each protein tested and the resultant p value, where the p value is calculated using Eq. 7. We would like to control the family wise error rate at $\alpha = .05$. To do this, we apply a Bonferroni correction and reject the null hypothesis for each protein at significance level $\beta = \frac{.05}{56} = 8.9 \times 10^{-4}$. We are only able to reject the null hypothesis of protein ACT1 which had a p value of 2.6×10^{-5}. However, because we have rejected one null hypothesis we can reject our global hypothesis at a significance level of $\beta \leq .05$!

The reader may be a little concerned at our result - we rejected the global null hypothesis even though we only rejected one of the intersecting hypotheses. This is a common pitfall in statistics - analyzing the resultant p values rather than examining the hypotheses tested [1]. In fact, it makes little difference if we reject 1 or all of the null hypotheses in regards to our statistical test. Rejecting more simply lower the type I error. To examine type I error in any more capacity rather than to reject a null hypothesis according to some threshold α lends little meaning and may bias statistical results if used to readjust tests or guide in the use of new tests.

Therefore, we can conclude that the CME protein interactions are partitioned into a set of interfaces that does not result from randomness. This partitioning describes a family of IINs, and the CME IIN C is a specific instance of this family. Hence the IIN C deviates from a randomly sampled IIN, even without having to

consider its specific connectivity! This observation is in line with the results of Johnson et al.[13], which found that the specific topology of the IIN is not random but under evolutionary pressure.

Table 3: Multiple Testing On Each CME Protein

Protein Name	Degree of Protein	Number of Interfaces	P Value [Eq. 7]
ACT1	24	4	2.6×10^{-5}
RVS167	4	2	1
APP1	9	2	1.2×10^{-2}
PRK1	12	3	2.1×10^{-2}
ARK1	6	2	0.23
SAC6	1	1	1
CRN1	6	4	1
TWF1	4	4	0.13
COF1	2	2	1
PFY1	1	1	1
LAS17	16	11	0.39
SLA2	8	5	1
AIM21	6	3	1
CAP1	4	4	0.13
AIM3	5	2	0.52
AKL1	6	2	0.23
APM4	3	3	0.4
APL3	3	3	0.4
SLA1	20	10	1
LSB3	15	3	1.7×10^{-3}
RVS161	1	1	1
AIP1	2	2	1
ABP1	7	15	0.84
ARC19	5	5	3.8×10^{-2}
YAP1801	4	4	0.13
LSB5	2	2	1
ARC35	3	3	0.4
CHC	8	3	0.34
ENT2	5	3	1
SCP1	2	2	1
APL1	3	3	0.4
ARP3	13	7	1
CLC1	2	2	1
CAP2	4	4	0.13
PAL1	1	1	1
EDE1	8	5	1
ASP2	3	3	1
VRP1	7	5	0.61
GTS1	5	2	0.52
INP52	3	3	0.4
YAP1802	7	5	.61
END3	4	2	1
SCD5	4	2	1
ARC15	3	3	1
ARC40	8	3	0.60
ARP2	13	7	1
ARC18	2	2	1
SYP1	6	5	0.24
PAN1	18	7	0.50
BBC1	7	2	9.8×10^{-2}
BZZ1	5	1	3.8×10^{-2}
YSC84	14	3	4.2×10^{-3}
MY05	13	4	5.9×10^{-2}
BSP1	7	4	1

4 Degree Distribution

So far we have been able to enumerate the number of possible IINs for a given PPI network and then use this information to create a statistical test that can determine if an IIN is formed through random interface splitting. However, we have not discussed the global characteristics of randomly formed IINs, that is, properties that appear throughout the entire network. For example, a question of interest would be what kind of connectivity patterns are seen in random interface networks? Do most interfaces in these random networks interact with only a few other interfaces or many interfaces? How many interfaces do we even expect to see in a network given some PPI? What would be the expected number of interactions each interface has in the network?

If we can answer these questions, then we can go back to an IIN that we classified as non random from our statistical test in the previous chapter and analyze how its properties depart from those of random IINs. These differences then can help us determine network function.

Here we will derive an algorithm for characterizing a network degree distribution with a single parameter, the attachment exponent, described further below. The degree distribution cannot be fit to a function directly to extract this parameter, because there is not a single function that describes the range of distributions one may observe. Instead, the attachment exponent defines how a network can be generated. Our algorithm is unique because it prevents the inclusion of orphan nodes, which allow the network to be effectively constructed with fewer nodes. This biases the comparison with the target network. To explain the method, we first introduce the degree distribution and its common forms.

4.1 What is a Degree Distribution?

We want to understand the connectivity structure, or topology, of a network. Yet, there can be an incredible amount of interfaces comprising a network and analyzing each in isolation, as we have previously done, gives little information on network structure. We need a tool that can compare the connectivity of interfaces across the entire network. Luckily for us, such a tool exists and is known as a degree distribution.

As we previously discussed in Chapter 2, the degree of a node in a network is the number of edges it has incident to itself. Therefore, each neighbor to a node contributes one to its degree and each self-loop contributes two to its degree. It is clear then to see that the degree is a measure of the connectivity of an interface.

We would like to make an informative plot of all of the degrees in a network. To do this, we first count the frequencies of each degree value appearing in the network and then normalize each value by the number of nodes in the network. Next, we create a density plot for a network by plotting the degree against its relative proportion in the network. An example of a degree distribution is given in Figure 6.

A degree distribution allows us to see immediate patterns of connectivity across the entire network. We can see what degrees are favored and which ones are rare. Furthermore, we can see if there is a probability distribution that can be fit to the data to predict relative frequencies.

4.2 Popular Degree Distributions

Degree distributions on many networks are well researched and characterized. In this section we will introduce two of the most popular and common distributions.

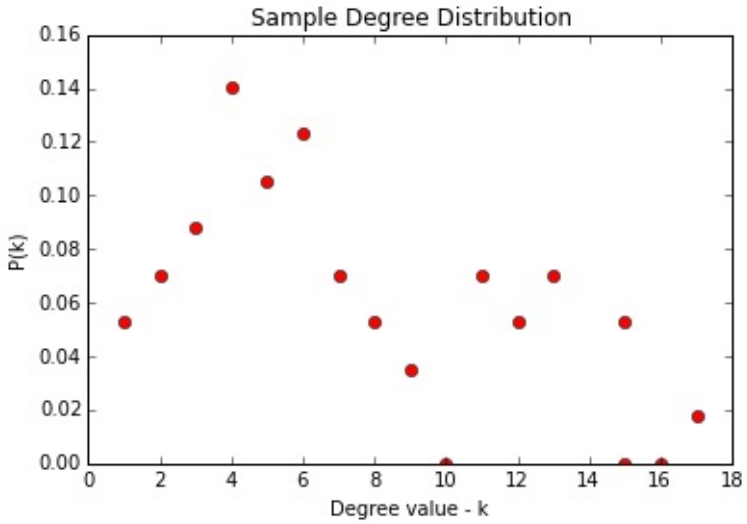

Figure 6:
A sample degree distribution for a network with 57 proteins and 197 edges. We see that the most frequent degree in this network is degree 4, comprising approximately 14% of nodes.

4.2.1 Binomial and Poisson Topology

We will discuss the degree distribution for traditional random networks. There are two popular models for such networks, the Erdős - Rényi model which we previously discussed, and the binomial model [9]. We will begin by exploring the binomial model which is a more simplistic view of a random network than the Erdős - Rényi model.

In the binomial model we fix a certain number of nodes, n, in a network and any two nodes will be adjacent with a probability of p, where $0 \leq p \leq 1$. A node will then have a degree k with probability $p^k(1-p)^{n-k-1}$. Because there are $\binom{n-1}{k}$ ways to pick pairs of nodes to be adjacent, we arrive at the degree distribution in Equation 8.

$$P(degree = k) = \binom{n-1}{k} p^k (1-p)^{n-k-1} \tag{8}$$

Figure 7 shows a sample network that was generated using the binomial model for random networks. A binomial degree distribution is also shown in Figure 8. As

$n \to \infty$ and $p \to 0$ the limiting distribution of a binomial is a Poisson distribution with $\lambda = np$. We note that this is equivalent to $np << n$ - meaning that the average degree of a node is much less than the total number of nodes in the network as the number of nodes grows to infinity [2].

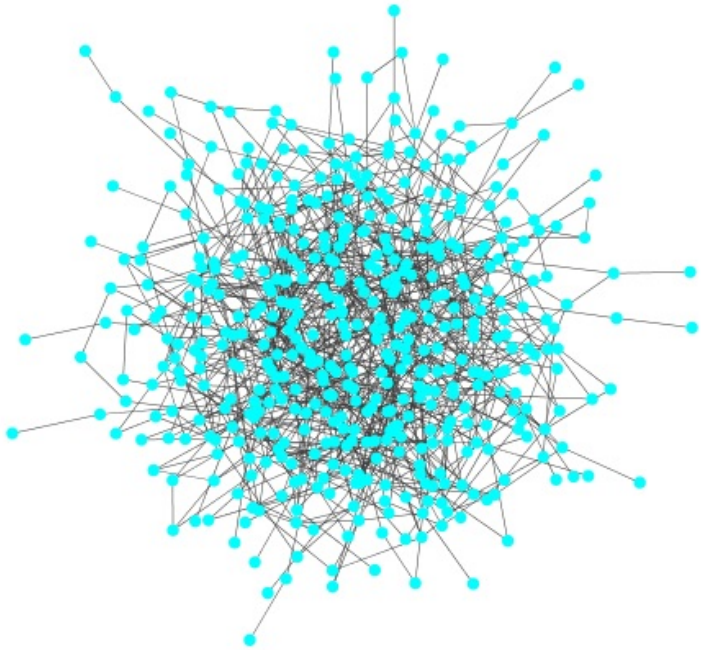

Figure 7:
An example of network generated using the binomial random network model

If, instead of letting edges vary, we fix the number edges, we arrive at the Erdős and Rényi model for random networks [8]. In this model, to generate a random network that has n nodes and N edges, the connectivity is chosen with equal probability from the $\binom{\binom{n}{2}}{N}$ possible edges. We can develop an equation that calculates the probability that a node has degree k as we did with the binomial model.

Determining such a probability is equivalent to determining the probability that a node is connected to exactly k others - we will assume no self loops for the sake of simplicity. There are $\binom{n-1}{k}$ ways to pick k neighbors for a given node. Furthermore, we must calculate the number of ways that other nodes can be connected. There are $N - k$ edges left in the network which can be randomly

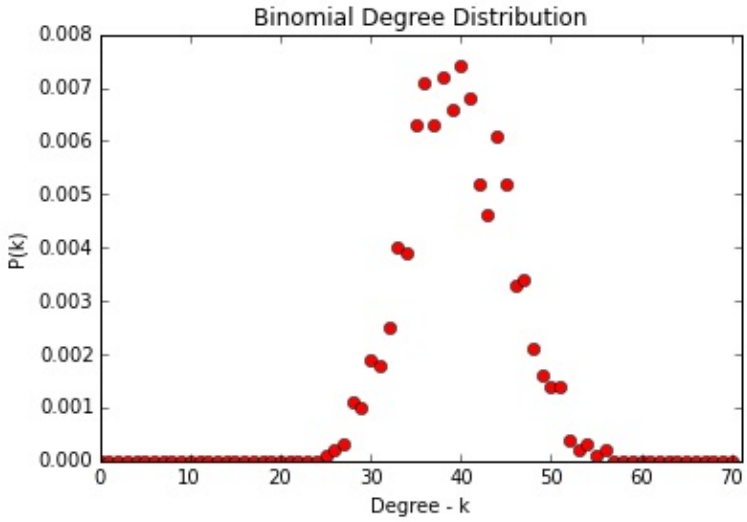

Figure 8:
A sample degree distribution for a network generated using the binomial model with 200 nodes and $p = .2$.

connected to any of the other $n-1$ nodes. Therefore, we must chose $N-k$ edges from the $\binom{n-1}{2}$ edges left to chose. This results in $\binom{\binom{n-1}{2}}{N-k}$ possible ways to distribute the remaining edges among the other $n-1$ nodes. It then follows from this that the probability of any node in the network having degree k is given by:

$$P(degree = k) = \frac{\binom{n-1}{k}\binom{\binom{n-1}{2}}{N-k}}{\binom{\binom{n}{2}}{N}} \tag{9}$$

The binomial model is typically easier to work with than the original Erdős - Rény model because we do not constrain ourselves to a set number of edges. Assigning a probability to each edge allows us to use a wide array of simple statistical techniques to extract network properties.

Unfortunately, the popularity of these networks as modeling tools is a result of their simplicity and not their prevalence in nature [2]. The scale-free model we will introduce next poses the opposite difficulties; it is highly prevalent in nature but far from simple.

4.2.2 Scale-Free Topology

If a network has a binomial or a Poisson degree distribution we expect there to be few outliers. We will illustrate this concept using the binomial model. Note that the variance of a binomial distribution is $np(1-p)$ and the standard deviation is $\sqrt{np(1-p)}$. It is easy to see that $\sqrt{np(1-p)} << n$.

For example, let us have a network with 1,000 nodes and we chose $p = \frac{1}{2}$ to maximize the variance. The average degree of this network is $np = (1,000)\frac{1}{2} = 500$. This variance of the degree of each node is simply $np(1-p) = (1,000)\frac{1}{2}^2 = 250$. The standard deviation is then $\sqrt{250} \approx 16$.

A binomial distribution with large n can be approximated using a normal distribution. The area of a normal distribution beyond 4 standard deviations of the mean represents less than .1% of the total area of the distribution. Thus, there is less than a .1% chance that a node will have a higher degree than 564!

This seems like an unrealistic model for natural networks as most networks that we are familiar with have outliers far from the mean degree. We will introduce a model, known as a scale-free network, that more accurately captures this observation.

A scale-free network is when the network's degree distribution follows a power law. That is, the fraction of nodes, $P(k)$, with degree k is approximately equal to k raised to some exponent γ. Equation 10 displays this relationship.

$$P(k) \sim k^{-\gamma}, \quad 2 \leq \gamma \leq 3 \tag{10}$$

This relation creates a heavy tailed degree distribution in which most nodes have a low degree but there are some nodes with extremely high degrees, well above the mean degree. We call such outlier nodes, hubs. A picture of a scale-free network is shown in Figure 9 and a degree distribution in Figure 10.

It turns out that these networks are seen throughout nature. Whether it be

at the level of the cell or on the scope of society and the internet, scale-free networks model most behaviors and interactions [4]. One possible explanation for the abundance of such networks was proposed by Albert-László Barabási with his model of preferential attachment [3]. In this model, networks continuously expand with the addition of new nodes and the new nodes are attached preferentially to existing nodes of higher degree. This scheme can be seen as "the rich get richer" - the higher the connectivity of a node, the more connections it will gain in the future. This model reproduces scale free distributions and serves as a possible explanation as to why such an abundance of real world phenomenon can be modeled with these networks.

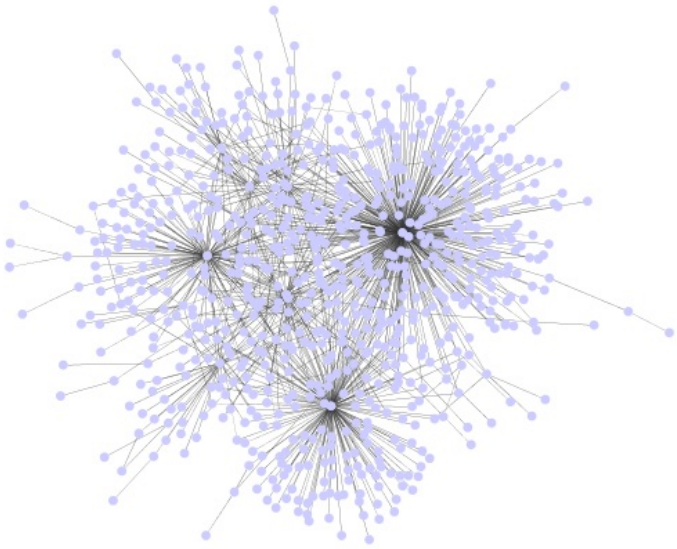

Figure 9:
An example of a scale-free network. Notice there are a few highly connected node called hubs and many nodes of low degree connected to the hubs.

Another interesting property may also be a reason why these networks are found in high abundance. Scale-free networks are highly robust against random failure due to their hub like structure. The chance that a hub fails is small compared to a less connected node. A less connect node is less vital to the functioning of the network and its failure will not be detrimental to the connectivity of the system. This is an especially important feature of protein networks whose com-

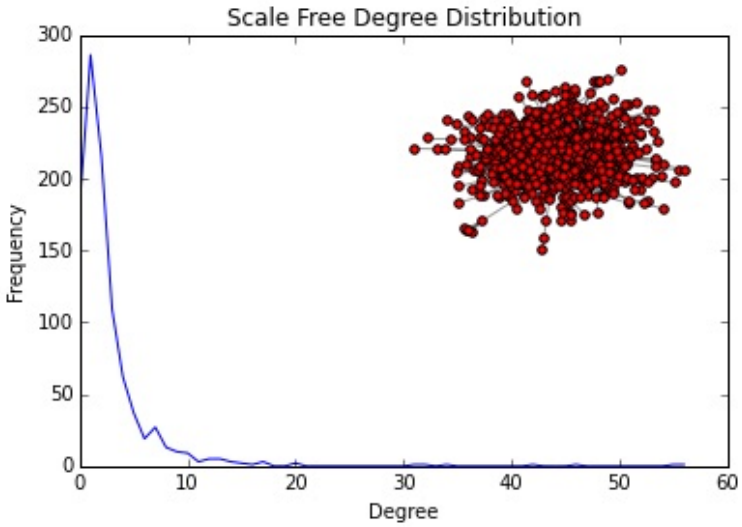

Figure 10: An example of a scale-free network degree distribution with 1,000 nodes and $\gamma = 2.5$. Note the very long tail of the distribution. A small version of the network is shown in the upper right corner.

ponents are susceptible to mutations.

4.3 Finding the Degree Distribution of a Random IIN

We would like to find some averaged degree distribution over all possible IINs for a given PPI network. This is a difficult task because our enumeration methods up to this point have only provided information on the splitting pattern of interfaces for a protein without tracking edge assignments and corresponding degrees. To mathematically track such features is a task for which the author has not found a solution.

Rather than taking a probabilistic approach like we have done up to now, we will take advantage of 21^{st} century technology. A Monte Carlo method was used which samples IINs constrained to a given PPI network. The method accepts or rejects a move, such as interface duplication or combination, based on the Boltzmann weight $e^{\frac{-f}{k_b T}}$. By setting the temperature to infinity all possible moves were accepted, generating randomized IINs.

If this algorithm was run for an infinite amount of time, it would transverse

the entire space of all possible IINs for the inputted PPI network. Unfortunately, because this thesis has a deadline, we cannot run the algorithm forever and must only use a sample of networks from the space generated by this algorithm. It is important to note that this sample is an unbiased sampling. Using this sample, we can measure average degree distributions of random networks and compare the similarity to those found in nature. Figure 11 displays sampled IINs produces from this algorithm using the CME PPI network as the input.

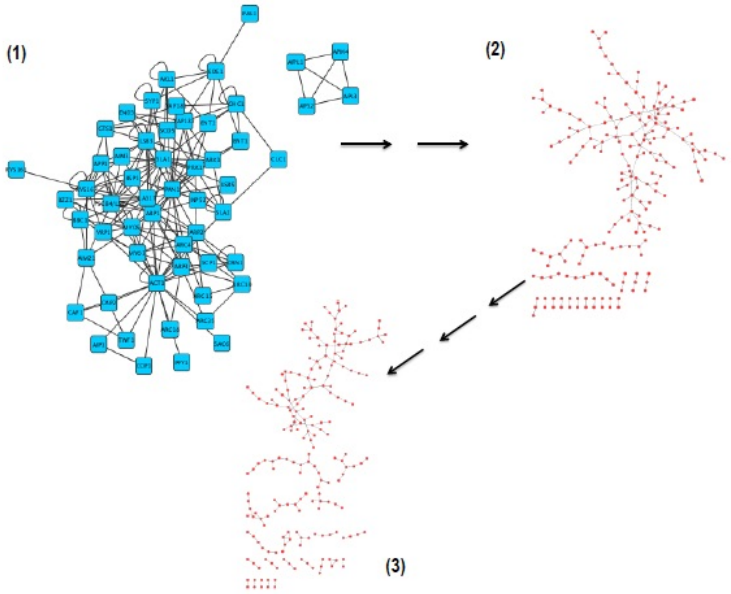

Figure 11: (1) Clathrin-mediated endocytosis in yeast PPI network used as an input to the Monte Carlo method (2) IIN generated after 100,000 iterations (3) IIN generated after 1,000,000 iterations

4.4 Generation of Networks with Specific Degree Distributions

It has been shown that PPI networks are scale-free [20]. A question of interest is whether IINs possess this property and if this property arises in randomly generated IINs.

Typically, a degree distribution is determined to be scale free by taking the log transform of the normalized degree frequencies and using these as the response

variables in a first order linear regression. The scale-free parameter, γ, can then be estimated using the coefficients from the fit. Unfortunately, this method performs poorly on sparse networks with heavy tailed distributions which are two characteristics of IINs. Also, our sampling may well produce networks that are not scale-free at all.

We would like to have a way to classify any degree distribution, whether it is power-law or not, and get an idea of whether it is more similar to a power law or a binomial degree distribution. These distributions then describe the scale-free versus the binomial network model.

One method would be to work backwards - generate networks with known parameters and compare them to our unknown degree distribution. We will develop this method over the coming pages.

Goh et al. introduced a simple algorithm to generate scale-free networks with a fixed number of edges and nodes [11]. In this algorithm, one begins with a set of n nodes and m edges. Each node is assigned a label from 1 to n, which we will denote i. We then assign the weight of $P_i = i^{-\alpha}$ to the i^{th} labeled node, where $0 \leq \alpha \leq 1$. Two nodes are then chosen with a probability equal to the their normalized weights, $\frac{P_i}{\sum_{j=1}^{n} P_j}$. An edge is added between these two vertices if one does not already exist. This picking and connecting continues until m edges have been added into the network.

When α=0, the method generates binomial networks. As α increases, the degree distribution becomes more like a power-law, and at that point will generate a scale-free network. We refer to alpha as the (preferential) attachment exponent, and use it as the single metric to characterize the degree-distributions of our sampled networks.

While this algorithm creates scale free networks, the networks it produces may have orphaned nodes - nodes that have a degree of 0. By construction, an IIN cannot have orphaned nodes. Remember, we define a node as a binding site.

Clearly a binding site must be connected to another binding site! Comparing an IIN to a network produced by the above algorithm is like comparing Coke and Diet Coke, there is no comparison! We must alter this algorithm to produce a fully connected network.

To combat this problem, edges were initially removed and later used to connect orphans back into the network. Therefore, rather than starting with m edges, we start with $m - R$ edges and once the algorithm terminates, we use the remaining R edges to connect orphaned nodes back into the network. However, finding the best value of R posed a problem. If R was too large, we would end up with many orphans and our algorithm would have a large bias reconnecting them back into the network. On the other hand, if R was too small, we may not have enough edges to reconnect orphaned nodes back into the network. To solve this problem, we want to find the optimal R. That is, we want to find when the expected number of orphans of a network produced using the Goh et al. algorithm with $m-R$ edges equals R. This would leave us exactly one edge to connect each orphan back into the network - thereby minimizing both the initial number of edges to remove and the number of orphans created.

To calculate this value of R, we must first determine the expected number of orphans in a network produced by the algorithm. This calculation is displayed below:

Consider a network with n nodes and m edges produced using the algorithm

$$Let \quad X_i = \begin{cases} 0 & if \quad degree(i) > 0 \\ 1 & if \quad degree(i) = 0 \end{cases} \tag{11}$$

The approximate probability node i is never picked is:

$$\left(1 - \frac{i^{-\alpha}}{\sum_{j=1}^{n} j^{-\alpha}}\right)^{(2m)} \tag{12}$$

We can use this result to find the following expectation:

$$E[Orphans|n, m, \alpha] = E\left[\sum_{i=1}^{n} X_i\right] = \sum_{i=1}^{n} E[X_i] = \sum_{i=1}^{n} \left(1 - \frac{i^{-\alpha}}{\sum_{j=1}^{n} j^{-\alpha}}\right)^{(2m)} \tag{13}$$

Initially, removing $E[Orphans|n, m, \alpha]$ edges will result in more orphans than removed edges. A recursive definition must be created for the optimal number of edges, R, to remove.

$$a_o = \left(1 - \frac{i^{-\alpha}}{\sum_{j=1}^{n} j^{-\alpha}}\right)^{(2m)} \quad a_k = \left(1 - \frac{i^{-\alpha}}{\sum_{j=1}^{n} j^{-\alpha}}\right)^{2(m-a_{k-1})} \tag{14}$$

$$a_k \longrightarrow R \quad as \quad k \to \infty$$

To generate a network that is fully connected with some parameter, α, we can use our recursive algorithm to determine the initial amount of edges, R, to remove. We then can run the Goh et al. network generating algorithm using $m - R$ edges. Upon termination of the algorithm, we reconnect all orphaned nodes with the R removed edges. If there are too many orphans, the network is discarded and a new one is generated.

4.5 Measuring α

We want to be given some IIN and determine the attachment exponent alpha that characterizes the distribution. Goh et al. estimated that the relationship between α and γ to be $\gamma = \frac{1-\alpha}{\alpha}$ [11]. We note that $2 \leq \gamma \leq 3$ when $\frac{1}{4} \leq \alpha \leq 1$. The closer α is to 1, the lower γ - this will decrease the number of hubs but increase their connectivity. We also note that as α approaches 0, γ approaches ∞. As γ gets larger, the network becomes random because nodes are chosen and connected to each other with equal probabilities.

To determine α for a given IIN with n nodes and m edges, we choose $\alpha = \{0, .1, .2, .3, ..., 1\}$ and generate 30 networks using each α value. We therefore generate a total of 330 networks, each with n nodes and m edges, over 11 different α values.

Next, degree distributions for each network are calculated. We take all 30 degree distributions that correspond to a given α and calculate the average degree distribution. Doing so for each value of α gives us 11 average degree distributions. A cumulative density function (CDF) is then created using each of these average degree distributions where a CDF is simply the function $F_X(x) = P\{X \leq x\}$ where X represents a randomly chosen node's degree.

The degree distribution for the IIN whose α parameter is being estimated is also calculated along with the corresponding CDF. This CDF is then compared to each CDF that corresponds to a different value of α by calculating the square errors between the distributions. The IIN is determined to have the α that corresponded to the CDF that yielded the lowest square error.

4.6 Example: Determining α for the CME IIN

We will walk through an example of determining the α value for the CME IIN. First, the degree distribution for the CME IIN was calculated along with average

degree distributions for 30 networks generated for each $\alpha = \{0, .1, .2, .3, ..., 1\}$ with the same number of edges and nodes as the CME IIN. A plot of this is shown in Figure 12.

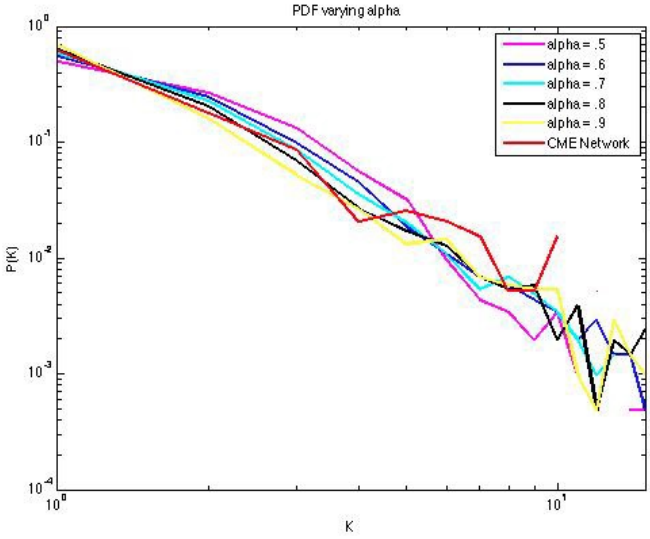

Figure 12: The average degree distributions for $\alpha = \{.5, ..., 1\}$ plotted with the degree distribution for the CME network IIN. For ease of visualization, we have not plotted the degree distributions for $\alpha \leq .4$.

Next, we calculate the CDF corresponding to each of these degree distributions. Figure 13 displays this plot. We note that the CDF functions are much smoother than their corresponding degree distributions. This smoothness allows for a better and more consistent comparison.

The squared error of each CDF from the generated networks with the CDF of the CME IIN is calculated. Figure 14 displays these errors. We pick the minimum error which came from the CDF corresponding with $\alpha = .7$. This large value indicates that the CME IIN is, in fact, scale free! It was further shown in [13] that the IIN CDF could be fit to a power-law distribution using a rigorous statistic and that the network is therefore scale-free. Our next task is to determine whether this was a result of randomness or selective pressure.

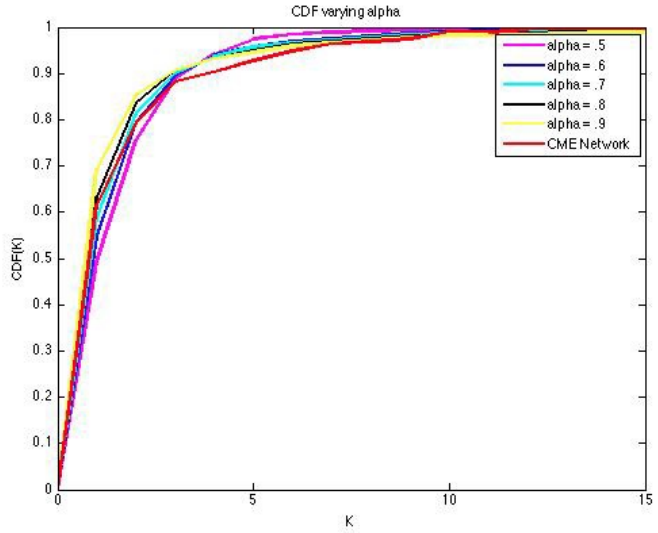

Figure 13: The CDFs corresponding to the average degree distributions for $\alpha = \{.5, ..., 1\}$ plotted with the CDF for the CME IIN. For ease of visualization, we have not plotted the CDFs for $\alpha \leq .4$.

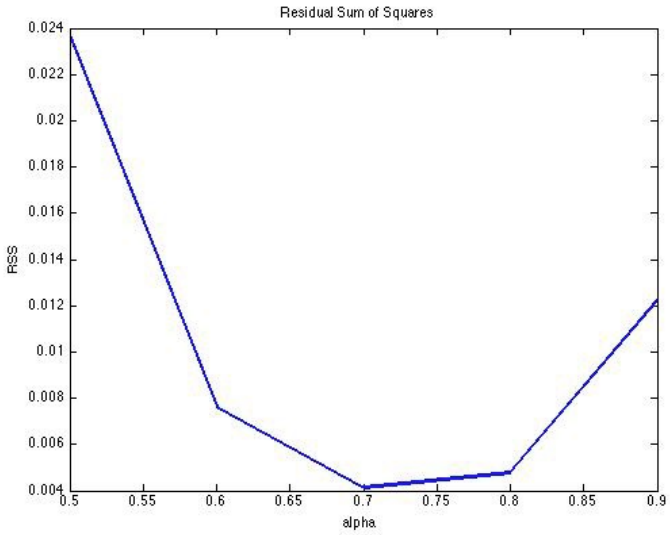

Figure 14: The squared errors from the comparison of the CDFs generated with known α to that of the CME IIN with an unknown α. For ease of visualization, we have not plotted the CDFs for $\alpha \leq .4$.

4.7 Random IIN Degree Distributions

We just found that the CME IIN is given by a large attachment exponent of $\alpha = .7$. However, we cannot conclude immediately that this is a result of evolutionary pressure. This simply may be a result of the constraints of the overlaying PPI network. To determine if this is the case, we can generate random interface networks constrained to the CME PPI network using our Monte Carlo method with the temperature set to infinity and measure the α value of a sample of such random networks.

We chose a sample of $2,000$ random networks from $2,000,000$ we generated. The α parameter for each network was measured. Figure 15 displays the frequencies of each α value.

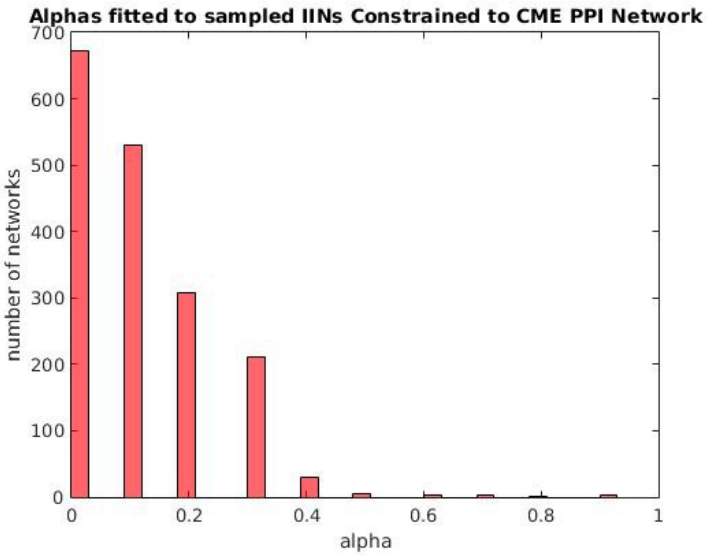

Figure 15: Depicts frequency of α fitted on 2,000 randomly sampled IINs

We see that very few networks were found to have an α of greater than .3 from Figure 15. In fact, the networks with high α values are most likely a result of networks chosen before our Monte Carlo method equilibrated. The low prevalence of large α networks means it is hard to randomly sample scale free IINs. This indicates that the scale free property of the CME IIN is probably not a result of randomness under the constraints of the overlaying PPI network but a function of some other topological pressure specific to the IIN.

4.8 Average Degree of Random IINs

Another interesting question to ask regarding the global structure of these random IINs is what we expect the average degree to be. So far, we have used computational methods to fit some parameter α to the degree distribution of a network. In turn, we found that random IINs generated from the CME PPI are not scale free.

The average degree is simply the average of all degrees in the network. It tells us what the expected number of edges each node will be incident to. We pose the next question: Would it be possible to calculate the expected average degree of a random IIN given some PPI network? We will show, while at first the solution may seem trivial, it is in fact not. Computational methods will then be used to estimate such an average.

We know from Equation 6 that for a given node, v, in some PPI network, G, the probability that v splits into X interfaces is as follows:

$$P(interfaces(v) = X) = \frac{S(deg(v), X)}{B_{deg(v)}}$$

We would like to determine the expected number of interfaces v splits into.

This is a simple calculation which is displayed in Equation 15.

$$E[interfaces(v)] = \sum_{j=0}^{\infty} P(interfaces(v) = j) * j = \sum_{j=1}^{deg(v)} \frac{S(deg(v), X)}{B_{deg(v)}} * j \quad (15)$$

We can actually use this expectation to determine the expected number of interfaces in a PPI network, G, with n proteins under random conditions. Equation 16 follows from the linearity of the expectation operator.

$$E[interfaces \quad in \quad G] = E[\sum_{j=1}^{n} interfaces(v_j)] = \sum_{j=1}^{n} E[interfaces(v_j)] \quad (16)$$

We will determine the expected number of interfaces in the CME IIN under random conditions using Equation 16. Table 4 gives the expected number of interfaces under random conditions for the nodes of varying degrees in the CME PPI network. We simply multiply each expected value by the number of proteins with the corresponding degree in the CME PPI network and take the sum over all of these values.

We get that the expected number of interfaces in the CME IIN under randomness is 192. The actual CME IIN has 195 interfaces, which is very close to this value. This is surprising because the networks have such different degree distributions and we rejected the hypothesis that the CME IIN was of random origins. We suspect that the closeness of these number results from similar means between the power-law degree distribution and the distribution of the sum of all interface-splitting patterns under randomness.

Now that we have finished calculating the expected number of interfaces in an IIN we can turn our attention to our main goal of calculating the expected average

Table 4: Expected Interface Count

Degree	Expected Number of Interfaces
1	1
2	1.5
3	2
4	2.47
5	2.90
6	3.32
7	3.72
8	4.11
9	4.48
11	5.21
12	5.56
13	5.91
15	6.58
16	6.91
18	7.55
20	8.18
24	9.40

degree. That is, we want to calculate Equation 17.

$$E\left[\frac{2 * |E(G)|}{n}\right] \tag{17}$$

Where $2 * |E(G)|$ represents two times the number of edges in the protein network, G. At this point, the reader is probably thinking that all we need to do is sum the expected number of interfaces for each protein and divide $|2*E(G)|$ by the result. This is what the author assumed before doing the math out. Unfortunately, we forgot that the inverse of an arithmetic mean and the inverse of the harmonic mean are not equal, Equation 18 displays this.

$$E\left[\frac{1}{n}\right] \neq \frac{1}{E[n]} \tag{18}$$

This means the expected average degree of the CME IIN is not simply $\frac{2*(206)}{192} \approx 2.15$. Finding $E[\frac{1}{n}]$ is extremely difficult - one would need to convolve n probability distributions together (one for each protein in the PPI network) in order to

determine the probability distribution of n and subsequently $\frac{1}{n}$.

While a general formula may not be possible, we can use our Monte Carlo method with the temperature set at infinity and look at the distribution of average degrees for randomly generated IINs for a given PPI network. Figure 16 displays such a distribution for IINs generated randomly constrained to the CME PPI network. Over $2,000$ IINs were chosen over 2 million iterations of the simulation.

We see that the distribution looks approximately normal, centered at 2, with a right skew. There are some outliers with a higher degree than 2.5 but we expect such outliers are artifacts of measuring networks before our Monte Carlo method had equilibrated.

The real CME IIN had an average degree of 2.11 which is a very likely value according to our randomly generated networks. Again, we suspect such closeness between our random distribution of expected average degrees and that of the real CME IIN to be a result of similar moments of the underlying probability distributions.

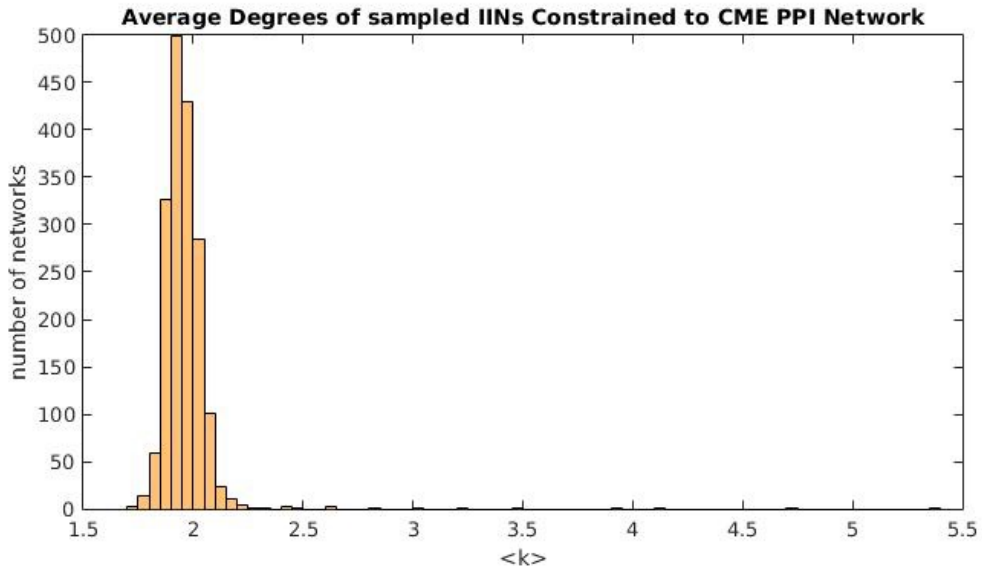

Figure 16: Shows the average degree, $< k >= 2 * Edges/Nodes$, and the frequency of the networks generated with $< k >$. There were a total of $2,000$ networks measured over 2 million iterations with networks chosen after every $1,000$ generations.

4.9 Note on Local Motifs

We have described global structures of random IINs but have not mentioned any local motifs we expect to see in such networks. A motif is a subgraph of a network that recurrently appears throughout the network. A few common motifs are cliques, triangles, squares, and chains. We will only discuss triangles and squares here but our analysis can be extended to the other subgraphs.

Let us first define what a triangle and square are. A triangle is simply 3 nodes, each pairwise adjacent. A square is 4 nodes connected by 4 edges to form what looks like a square. Figure 17 and 18 display pictures of these motifs.

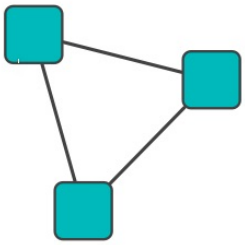

Figure 17: Example of a triangle subgraph

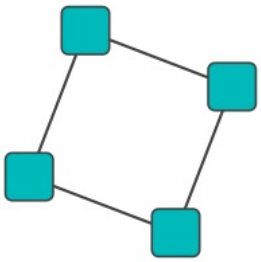

Figure 18: Example of a square subgraph

An interesting question is how many triangles and squares do we would expect to see in a random IIN chosen from I_G (the space of all possible IINs) given a PPI network G. This is a difficult question to answer and we will illustrate why. We first, however, will determine a simple upper bound for the number of squares and triangles that can appear in an IIN.

We propose the following upper bound: Assuming no self loops in the PPI network, a corresponding IIN can have at most the number of triangles and squares as the number contained in its parental PPI network. This statement can be quickly proven by contradiction. Suppose our statement was false. This means that we can find an additional triangle or square in the IIN network that is not present in the PPI network. However, if 3 interfaces are pairwise adjacent in the IIN network then 3 proteins must be pairwise adjacent in the PPI network forming a triangle. If there is a square in the IIN network then by the same reasoning, there must be a corresponding square with the overlaying proteins that is formed in the PPI network. By contradiction, our initial statement must hold true.

Finding the exact expectation of the number of triangles and squares is very difficult. First, there can be self-loops in a real PPI network. Second, triangles and squares can share edges with other triangles and squares. Lastly, determining the probability of a triangle or square being preserved from the parental PPI network to the IIN is highly dependent on the number edges incident to each protein forming the triangle or square.

This is an area that further research must be conducted on. Local features of IINs are highly important to understanding network function. Having an understanding of these structures that we would not expect under randomness can allow us to pinpoint special functions of an IIN. To accomplish such a task, Monte Carlo methods can be employed like we have done previously to calculate motif frequency and approximations can be determined.

5 Conclusions

We showed that it is possible to enumerate the number of IINs in a PPI network using Bell numbers, and that this number is enormous. It also depends on the topology of the PPI network, where a random PPI network has a smaller number of IINs than a scale free PPI network of the same size.

We subsequently devised a new statistic for determining if an observed interface distribution for a given PPI network is representative of a random, or uniformly sampled network. Our statistic is simple to use and when applied to the CME IIN showed that even when considering only the partitioning of the protein interactions between interfaces, the network is not consistent with a randomly generated one.

In addition, we developed a new algorithm for characterizing the degree distribution of sampled networks using a single parameter, the attachment exponent. Our method does so by generating networks defined by this attachment exponent. It takes particular care to prevent the inclusion of orphans in the network, and does so by using a recursive formula we derived for maximal efficiency.

Finally, we performed MC sampling to characterize the topologies of IINs possible for a given PPI network, such as the CME PPI network, and to specifically quantify the degree distributions of these topologies. We used our algorithm and found that most networks sampled have a degree distribution most consistent with a binomial distribution ($\alpha \leq 0.3$). In contrast, the CME IIN has a value of 0.7, representing a rare type of network topology. These results demonstrate that the biological IIN develops under specific evolutionary constraints that drive the production of hub interfaces.

Our formulas and algorithms are general and can be applied to any networks, not just the CME PPI studied most heavily here. The methods described here have already been applied to the analysis of another biological network that describes growth factor signaling in humans through the ErbB pathway [15].

References

[1] M Baker. Statisticians issue warning over misuse of p values. *Nature*, 531(7593):151–151, 2016.

[2] Albert-László Barabási. Network science. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 371(1987):20120375, 2013.

[3] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[4] Albert-László Barabási et al. Scale-free networks: a decade and beyond. *science*, 325(5939):412, 2009.

[5] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 289–300, 1995.

[6] Daniel Berend and Tamir Tassa. Improved bounds on bell numbers and on moments of sums of random variables. *Probability and Mathematical Statistics*, 30(2):185–205, 2010.

[7] Paul Erdős and Alfréd Rényi. On random graphs. *Publicationes Mathematicae Debrecen*, 6:290–297, 1959.

[8] Paul Erdős and Alfréd Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hungar. Acad. Sci*, 5:17–61, 1960.

[9] Edgar N Gilbert. Random graphs. *The Annals of Mathematical Statistics*, 30(4):1141–1144, 1959.

[10] Jelle J Goeman and Aldo Solari. Multiple testing for exploratory research. *Statistical Science*, pages 584–597, 2011.

[11] K-I Goh, B Kahng, and D Kim. Universal behavior of load distribution in scale-free networks. *Physical Review Letters*, 87(27):278701, 2001.

[12] Trey Ideker and Roded Sharan. Protein networks in disease. *Genome research*, 18(4):644–652, 2008.

[13] Margaret E Johnson and Gerhard Hummer. Evolutionary pressure on the topology of protein interface interaction networks. *The Journal of Physical Chemistry B*, 117(42):13098–13106, 2013.

[14] Margaret E Johnson and Gerhard Hummer. Interface-resolved network of protein-protein interactions. *PLoS Comput Biol*, 9(5):e1003065, 2013.

[15] Christina Kiel, Erik Verschueren, Jae-Seong Yang, and Luis Serrano. Integration of protein abundance and structure data reveals competition in the erbb signaling network. *Science signaling*, 6(306):ra109–ra109, 2013.

[16] WebMD. The olympic diet of michael phelps. `http://www.webmd.com/diet/20080813/the-olympic-diet-of-michael-phelps`. Accessed: 2016-3-24.

[17] Eric W Weisstein. Stirling number of the second kind. *triangle*, 7:8, 2002.

[18] Eric W Weisstein. Bonferroni correction. 2004.

[19] Eric W Weisstein. Bell polynomial. 2005.

[20] Soon-Hyung Yook, Zoltán N Oltvai, and Albert-László Barabási. Functional and topological characterization of protein interaction networks. *Proteomics*, 4(4):928–942, 2004.

Curriculum Vitae

I was born in Falmouth, Maine on May 29th, 1994, where I spent the entirety of my childhood. I attended Johns Hopkins University and was accepted into the combined Bachelor and Master's program through the Department of Applied Mathematics and Statistics. I expect to receive both degrees during May of 2016 and be initiated into Phi Beta Kappa. I will be attending in the fall the Alpert Medical School of Brown University to pursue a medical doctorate.

I enjoy the broad application of applied math. Outside of the research that I have conducted in this thesis, I have worked as a software developer, creating real-estate analytic tools. I also have conducted network science research through the Maine Medical Center Research Institute. As a team member in the Qualcomm Tricorder XPRIZE competition, I have developed algorithms for the diagnosis of atrial fibrillation and sleep apnea. Currently, I work as a data scientist and software developer for The Student Doctor Network. Outside of academia and work, I enjoy running and a good slice of pizza.