2016

Identifying Parameters for Robust Network Growth using Attachment Kernels: A case study on directed and undirected networks

Ahmed F. Abdelzaher

Virginia Commonwealth University, abdelzaheraf@vcu.edu

DISSERTATION IDENTIFYING PARAMETERS FOR ROBUST NETWORK
GROWTH USING ATTACHMENT KERNELS: A CASE STUDY ON DIRECTED AND
UNDIRECTED NETWORKS

A Dissertation submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy at Virginia Commonwealth University.

by

AHMED FAROUK ABDELZAHER

M.S., The University of Southern Mississippi - Aug 2008 to May 2010

Director: Dissertation Preetam Ghosh,

Associate Professor, Department of Computer Science

Virginia Commonwewalth University

Richmond, Virginia

July, 2016

Acknowledgements

I am thankful to the committee members: Dr. Vojislav Kecman, Dr. Thang Dinh and Dr. Kevin Pilkiewicz, who agreed to serve on the committee. Special thanks to Dr. Preetam Ghosh, my advisor since Fall 2011, for his continuous support and to Dr. Michael Mayo, my collaborator, for his support.

I am grateful to all my parents, brother, sister, other family members, many friends who have supported me either whole-heartedly or reluctantly. I am also thankful to the free and open source community that developed software I have been using for many years, and for the numerous online research resources that continue to help me grow as a researcher and developer.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Abstract

DISSERTATION IDENTIFYING PARAMETERS FOR ROBUST NETWORK GROWTH USING ATTACHMENT KERNELS: A CASE STUDY ON DIRECTED AND UNDIRECTED NETWORKS

By Ahmed Farouk Abdelzaher

A Dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at Virginia Commonwealth University.

Virginia Commonwealth University, 2016.

Director: Dissertation Preetam Ghosh,
Associate Professor, Department of Computer Science

Network growing mechanisms are used to construct random networks that have structural behaviors similar to existing networks such as genetic networks, in efforts of understanding the evolution of complex topologies. Popular mechanisms, such as preferential attachment, are capable of preserving network features such as the degree distribution. However, little is known about such randomly grown structures regarding robustness to disturbances (e.g., edge deletions). Moreover, preferential attachment does not target optimizing the network's functionality, such as information flow. Here, we consider a network to be optimal if it's natural functionality is relatively high in addition to possessing some degree of robustness to disturbances. Specifically, a robust network would continue to (1) transmit information, (2) preserve it's connectivity and (3) preserve internal clusters post failures. In efforts to pinpoint features that would possibly replace or collaborate with the degree of a node as criteria for preferential attachment, we present a case study on both; undirected and directed networks. For undirected networks, we make a case study on wireless sensor networks in which we outline a strategy using Support Vector Regression. For Directed networks, we formulate an Integer Linear Program to gauge the exact transcriptional regulatory network optimal structures, from there on we can identify

variations in structural features post optimization.

CHAPTER 1

INTRODUCTION

1.1 Complex Networks

The dynamics of complex networks are derived using graph theoretical measurements that are deduced from the topology of the network entities and their relationships. For example, science collaboration networks are portrayed using nodes that represent scientists or authors, and links that connect pairs of nodes that coauthored an article [1]. Unlike engineered networks such as wireless sensor networks [2] and airline transportation networks [3], science collaboration networks fall under the "small world" category of complex networks due to their smaller average over the ensemble of shortest connected paths through a network. Networks subscribing to the same category, such as the World Wide Web, cell structures networks, protein-protein interaction networks, the Internet, and infectious disease networks have all been analyzed for path lengths, cluster formations, degree distributions, and evolutionary patterns [1, 4, 5, 6, 7, 8, 9, 10, 11].

Gene regulatory networks (GRNs) also belong to this category. Understanding the dynamical consequences implied by the architecture of GRNs has been one of the major goals in systems biology and bioinformatics, as it can provide insights into, e.g., disease dynamics and drug development [12, 13]. In gene-regulatory networks, the nodes portray products of genes or transcription factor proteins within a cell, and a set of directed bonds which each denote pairs of nodes that interact by altering the activity of the target gene [14] parameterized by the biological processes of translation and transcription [15]. Unlike engineered communication networks (as in [16]),

1

GRNs exhibit a unique withstanding property–a phenomenon known as "Biological Robustness" [17, 18], which describes an ability of individual genes to adapt to and potentially resist disturbances to gene activity based, in part, on their connectivity to other genes of the network [19]. Such a useful property could be potentially exploited to design engineered networks with similar communication properties [20, 21, 22, 23].

1.2   Attachment Kernels

One way of understanding the dynamics of such complex networks, is to compare them with random models of similar properties, like for example: number of nodes and edges, and the clustering coefficient. Random models can be generated using preferential attachment schemes and their emergence are well studied [24, 1, 25, 26]. Consider random models vs. an existing complex network (or real), where the above characteristics have shown to be comparable. Consider the case that when contrasting the same real network with the generated models, it was observed that the average shortest path was much smaller in real vs. random. Knowing that both networks have much in common, one can deduct topological criteria that caused the differences in connectivity, hence, prove significance attributed to real structures. For example, insight into the emergence of The Internet was given based on contrasting the real network with model networks grown at random where few of the nodes were densely connected representing hubs or servers, and several were loosely connected representing clients [25, 1].

Preferential attachment schemes are usually governed by a mathematical formulae termed *Attachment kernels*. Attachment kernels are criteria for growing a network by which an expansion favors one part of a network (e.g., a node) more than another. Perhaps the most popular attachment kernel is that of the BA-model [27] which uses the degree of the node as a criteria, hence the attachment kernel is based on degree.

BA was formulated to construct random "scale-free" networks where rich nodes remain rich and poor nodes remain poor after the expansion. A rich node is that of a high degree, i.e., having degree much higher than the average, while a poor node has a degree much lower than the average.

The BA-model considers an initial substrate network of one or more nodes. The network grows in successive iterations where each incoming node can attach with one or more edge(s) to distinct nodes within the substrate. A candidate node within the substrate will have a probability of connecting with an incoming node equal to the *ratio of it's degree to the total degrees* of the network. Essentially, this ratio formulates the attachment kernel.

Network growing models are useful because they provide means for modeling empirical networks as random networks in attempt to study structure and dynamics [25]. Certain models, utilizing preferential attachment [27] and others [28], have succeeded in preserving the "scale-free" property as well as small average distances. Other network growing mechanisms were capable of forming structures with intended degree distributions [29] and clustering coefficients [30] by setting a priori for these metrics. Moreover, we have recently been able to combine attachment kernels based on in- and out- degrees to create scale-free directed networks with high motif counts [31].

Perhaps these models are useful, however, they only take topology in consideration. Little work has been done towards utilizing growing mechanism for optimizing network performance. For example, setting a priori for high clustering coefficient does not guarantee optimal routing in wireless sensor networks. It would be useful if these grown networks where tested for function in addition to the structure they intend to mimic. This fact partially motivates our work in this dissertation where we would like to identify network features that would optimize information flow.

1.3 What is Network Robustness?

The term "Robustness" usually refers to any system preserving a particular performance after suffering from some loss or disturbance. For example the human body can preserve a performance metric like body temperature even when subject to cold environments. Many engineered systems try to adopt biological behavior due to that fact [32, 33, 34]. The term Biological Robustness was first coined by Kitano et al. [17, 18] after exploiting properties of GRNs. The authors observed that these networks can maintain signaling transductions and gene expressions post external perturbation, hence they are considered to be "robust".

Some of the above have also addressed network robustness. For example the authors of [27] consider modeling the Internet's robustness to attacks [24]. To do this, randomly chosen nodes are deleted, and the network diameter is recorded after each deletion to form a 'diameter vs. network size plot'. The same is applied to another arbitrary network of similar number of nodes and edges, and the gradients of both plots are compared. This scenario is known as random deletions- a way describing random attacks and failures. Eventually, the authors concluded that the internet is robust because the diameter's plot rises at a relatively low gradient post node failures. This essentially means that "the Internet will not suffer that much if some terminal or server failed, and packets will eventually find their way to their destinations within reasonable time".

But does the random failure scenario suffice for assessing network robustness? No, there are several aspects to review in the domain of network robustness in addition to performance (information flow) post failures, and as a result, researchers have yet to establish a unified quantification for network robustness. For example, many suggest that robustness should be measured using other metrics. For example [35]
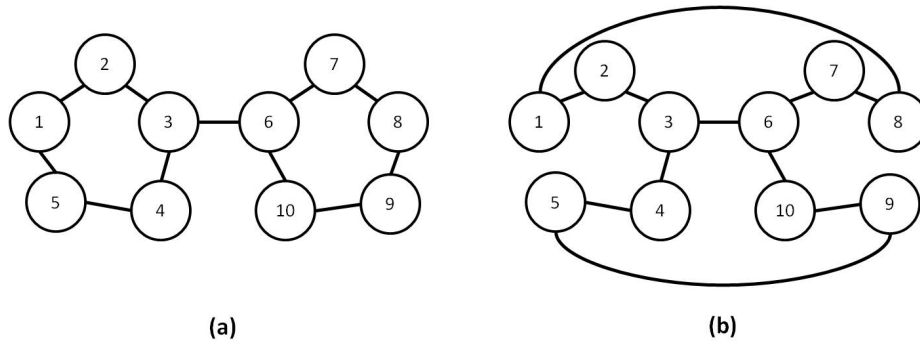
Fig. 1. Toy example.

suggests that connectivity should be assessed via observing the change in "network efficiency" (a variant of the average shortest path) post random nodal deletions (to simulate errors). Others consider different forms of disturbances; works assessing modular failures (using deletions) [36], in addition to link fragility (using fractional deactivation) [37] have been considered.

Consider the toy example given in Figure 1 [38]. Both have similar average degree, clustering coefficients and degree distributions. The only difference is that the average path length in (a) and (b) are 2.56 and 2.44 respectively. Topology (a) can be divided into two components by removing link 3-6, which makes topology (b) more robust towards link failures. On the other hand, if robustness was evaluated using strength of ties, e.g., in social networks, topology (a) would be considered more robust, because it is easier to divide it in two equally sized groups. What if nodes 1 in each network where infected with a virus? Because of links 1-8 in (b), a virus would spread faster to every node in the network, which makes topology (a) more robust. How about routing efficiency in case of congestion of link 3-6? In this case topology (b) will be able to reroute information from groups 1-2-3-4-5 to 6-7-8-9-10 through links 5-9 and 1-8, and vice versa. This makes topology (b) more robust.

It is clear that robustness needs to be understood according to the network's be-

havior post disturbance. Some topologies suffer greatly, others sustain their abilities. However, which metric should you consider? The answer to this question motivates our research. Here, we examine a collection of metrics: together they cover the (1) network's performance, e.g., packet transmission, (2) the network's connectivity, e.g., the average shortest path, and (3) it's ability to form internal communities, e.g., clusters.

CHAPTER 2

MOTIVATION AND CONTRIBUTIONS

There is no particular formula or attachment kernel that can guarantee randomly generated networks with optimal performances, connectivity and modularity- in other words "robustness". In order to do so, firstly, structural features that affect the above should be identified, which can be potential candidates for attachment kernel criteria. Secondly, Once attachment kernels have been formulated, comprehensive testing of different random failure scenarios should be applied to grown structures. This dissertation mainly considers the first of which, the testing component is considered for future works of this research. We present some attachment kernel testing cases for Wireless Sensor Networks growth in Chapter 6.

Static measures like connectivity and strength of tie formations are universally applied for any network. However, dynamic performance like signaling flux or packet transmission would differ according to the nature of the network itself. Therefore, we explain our strategy for pinpointing those structural features on different networks. (1) For an undirected network case study, we consider **Wireless Sensor Networks**, while the (2) directed network case study, we consider **Transcriptional Regulatory Networks**. We use **Network Simulator 2** to evaluate routing efficiency for wireless sensor and transcriptional networks, although, for this research component results for transcriptional networks will be reported in future works.

It is important to note that the focus of this research is to lay different strategies to pinpointing features for enhancing networks' performances and/or expansion. This dissertation provides different methods to consider when tackling this problem

for undirected and directed networks. Networks considered like Transcriptional regulatory or wireless sensor networks, are two of many different existing networks that might have different performance expectations. However, a universal performance that can be addressed for example, is the ability to transmit information within the topology given. We will explain later in this dissertation how information transfer efficiency can be modeled as packets flowing through a wireless sensor network for both directed and undirected networks.

Because we address network expansion using attachment kernels and we argue that preferential attachment can depend on more than one parameter, we give two example algorithms for growing networks. For these algorithms, we have observed comparable topologies to the real networks, and details are discussed in chapters 3 and 4. The second part of this thesis adopts the same growing schemes, however, we try to pinpoint different features to consider for preferential attachment. For this we consider wireless sensor networks and the features are selected. Preliminary testing cases prove our hypothesis, that it is possible to use attachment kernel with features other than the degree of the nodes to build random scale-free topologies yet exhibiting some degree of robustness. The third part of this thesis supports the random model generation by enhancing a substrate network by re-wiring of the edges. Because the final grown system depends heavily on the initial substrate [25], this part of the dissertation is also necessary.

2.1 Motivation for Case Study 1 on Wireless Sensor Networks

Wireless Sensor Networks (WSNs) have many applications that can help humanity, however, they are accompanied with many challenges during their emplacement-mainly attributed to inefficient packet transmission and non-robust structures. Identifying features that make WSNs robust will enhance their deployment structures. In

order to motivate the strategy for selecting robustness relevant features for WSNs, we must first define the different aspects of WSN robustness. These aspects are defined as follows:

1. Performance: A WSN should maintain reasonable packet transmission rates post disturbances such as edge deletions and elevated channel noise levels.

2. Connectivity: A WSN should maintain high connectivity post disturbances such as edge deletions.

3. Internal Communities: A WSN should maintain it's ability to form internal clusters post disturbances such as edge deletions

Problem Definition 1. There is no comprehensive strategy that can pinpoint the structural features necessary for preserving the different aspects of Wireless Sensor Network robustness.

Having a strategy for selecting robustness features will help identify new attachment kernels for WSN growth and robust WSN design. Such attachment kernels can help with computational experiments on different WSN structures, and eventually save costs in terms of labor and time during their physical deployment. For example, consider a hypothetical WSN structure deployed in the field, and that we need to determine the most efficient connection for the next 10 sensors to be added to the network. Network growing algorithms using attachment kernels that considers all aspects of WSN robustness can computationally determine different structures to be considered.

2.2 Motivation for Case Study 2 on Transcriptional Regulatory Networks

Our motivation for the transcriptional regulatory network's case study is two-fold. The first of which, motivates the use of network growing mechanisms for directed

networks. The second motivates the necessity for deducing structural feature varia-
tions for optimized performance directed networks.

2.2.1 Motivating Network Growing Mechanisms

Most of the popular network generation algorithms like the BA-, ER-, and the
configuration model are applied to growing undirected substrates, little is applied for
undirected networks. Therefore, a random model generator for directed networks is
required.

Understanding relationships between architectural properties of gene-regulatory
networks (GRNs) has been one of the major goals in Systems Biology and Bioinfor-
matics, as it can provide insights into, e.g., disease dynamics and drug development.
Such GRNs are characterized by their scale-free degree distributions and existence of
network motifs i.e., small-node subgraphs that occur more abundantly in GRNs than
expected from chance alone. Because these transcriptional modules represent build-
ing blocks of complex networks and exhibit a wide range of functional and dynamical
properties, they may contribute to the remarkable robustness and dynamical stability
associated with the whole of GRNs. Here, we developed network-construction mod-
els to better understand this relationship, which produce randomized GRNs by using
transcriptional motifs as the fundamental growth unit in contrast to other methods
that construct similar networks on a node-by-node basis. Because this model pro-
duces networks with a prescribed lower bound on the number of choice transcriptional
motifs (e.g., downlinks, feed-forward loops), its fidelity to the motif distributions ob-
served in model organisms represents an improvement over existing methods, which
we validated by contrasting their resultant motif and degree distributions against ex-
isting network-growth models and data from the model organism of the bacterium
Escherichia coli. These models may therefore serve as novel test-beds for further eluci-

dating relationships between the topology of transcriptional motifs and network-wide dynamical properties.

2.2.2 Motivating Optimizing an Undirected Network Performance

We already know that Transcriptional Regulatory Networks exhibit scale-free topologies which gives them structural robustness [24]. Moreover, WSNs adopting TRN topologies have shown better performances in-terms of packet transmission, energy consumption and end-to-end delays [21]. However, a question remains unanswered; is whether or not a TRN structure, like that of *E. coli*, is optimal. And if not, can this structure be improved? Improving existing TRN topologies will assist in designing robust networks. For example most network growing mechanism consider incremental attachments based on existing features of the substrate networks- initial networks before the attachments. Having an optimal substrate will serve better down the attachment process as compared with the case of beginning with a non-optimal substrate.

Given any arbitrary directed network, an optimal performance topology should minimize the overall distances from transmitter to receiver nodes (nodes with no out-degree). One metric that can capture this performance is the average shortest path [24, 1]. However, in order to achieve optimal robustness the degree distributions (in- and out-degrees) should also be preserved. The above serves as basis for our problem definition and motivation.

Problem Definition 2. Given an undirected network with a particular in-degree, out-degree distributions, and an average shortest path among pairs of transmitter and receiver nodes, an algorithm that can preserve the distributions provided having the average shortest path a minimum is required.

2.3 Research Contributions

Contributions of this work can be summarized in the following points:

1. Constructing different schemes for incremental network expansions using combinations of attachment kernels; one scenario describes single node attachments (Chapter 3) and the other described motif-based attachments (Chapter 4).

2. Presenting a scenario for evaluating network robustness via random edge deletions, in addition to providing proof for bio-inspired network robustness (chapter 6).

3. Providing a machine learning strategy using support vector regression for identifying robustness features in both; bio-inspired and randomly generated wireless sensor networks, thereby providing an undirected network case study (Chapter 7).

4. Providing an integer linear programming formulation that can be a tool for detecting variations in structural features of transcriptional regulatory networks evolving from there original state to an optimal structure, thereby providing a directed network case study (Chapter 8).

5. Discuss how selected features can be utilized as preferential attachment kernels in the context of future works (chapter 9).

Furthermore, literature reviews on bio-inspired wireless sensor networks and the different topologies considered are given in chapters 5 and 6.1 respectively.

CHAPTER 3

GROWING DIRECTED NETWORKS USING SINGLE NODE ATTACHMENT KERNELS

The material of this chapter was published in [31]. Here we construct random directed scale-free networks which exhibit similar structural distributions of that of the TRN of the bacterium *E. coli*. Using the transcriptional network of the bacterium Escherichia coli, we investigate motif connectivity by reconstructing the distribution of genes participating in feed-forward loop motifs from its largest connected network component. We contrast these motif participation distributions with those obtained from model networks built using the preferential attachment mechanism employed by many biological and man-made networks. We report that, although some of these model networks support a motif participation distribution that appears qualitatively similar to that obtained from the bacterium *E. coli*, the probability for a node to support a feed-forward loop motif may instead be strongly influenced by only a few master transcriptional regulators within the network. From these analyses we conclude that such master regulators may be a crucial ingredient to describe coupling among feed-forward loop motifs in transcriptional regulatory networks.

3.1 Motifs

It was discovered that genetic networks host repeating patterns of smaller sub-networks, termed motifs [39], that occur far more frequently than would be expected in randomized networks with the same degree sequence. These patterns are thought to be the basic building blocks of complex networks [40]. While much attention has

13

been directed toward the study of their individual functions, both experimentally (e.g., autoregulatory motifs [41]) and theoretically [42], much less is known relating their coupling and positions within the network to its robustness.

Feed-forward loops are one of the most common motifs in genetic networks and are well studied in a variety of biological contexts. In a genetic network, if one gene is linked to another, then it may either enhance or repress the expression level of the target gene, respectively termed up- and down-regulation. A feed-forward loop consists of three genes or nodes, the first of which regulates a second, and both of these co-regulate a third (Figure 11). Recently, Alon and collaborators [42] discovered that individual feed-forward loops possess interesting dynamical properties, such as signal delay and pulse generation. Although it is not generally clear how coupling among these motifs affects the overall network function, several groups are beginning to move in this direction. For example, exhaustive experiments with the bacterium *E. coli*, in which 598 gene promoters were altered to rewire its genetic network, showed that most of these new connections are tolerated by the bacteria [43]. Mathematical modeling of gene transcription and translation has also been used to investigate the relationship between coupling and function among differing motif configurations [44, 45, 41]. However, a requisite for using these results to understand complex features at the network level, such as robustness, is a more basic understanding of how motifs are coupled together and distributed throughout such transcriptional networks.

3.2   Participation of Genes in Motifs Distributed Throughout a Network

Here we consider the transcriptional regulatory network of the bacterium *E. coli* as a prototypical genetic network, by which we mean that genes interact with one another when transcription products affect the trans-activation of other target genes by interacting with their promoter regions. Not only are all connections among

14

genes in *E. coli*'s genetic network well validated by experiments (e.g., [39]), but these data are also easily sampled using the software tool GeneNetWeaver [46], first introduced to aid the development of more accurate gene regulatory network inference algorithms. *E. coli*'s genetic network supports 23 disjoint subnetworks that together form a network of 1565 genes and 3758 links, and it is not completely connected. Based on this observation we restrict our analyses to its largest connected component (LCC), which is sparse, supporting 1477 genes and 3671 directed links.

For each gene in the LCC of this genetic network, we count how many feed-forward loop motifs a gene participates in as one of its three elements, illustrated as nodes i, j, or k in Figure 11. A Java module was developed here to identify feed-forward patterns in the network, independent of whether one gene up- or down-regulates another. So we did not distinguish between, for example, coherent and incoherent feed-forward loops in the counting procedure. Motifs were compared to one another to ensure that they were only counted once for each gene. These steps were repeated for the model networks built from procedures described below.

3.3   Degree Distributions for Growing Networks

Because *E. coli*'s LCC is a directed network, it supports two distinct distributions that together describe the total-degree distribution. For a network of n nodes, these are (i) the fraction of the network hosting K-many outgoing links, $p(K, R, n)$, termed the out-degree distribution, and (ii) the fraction of the network hosting R-many incoming links, $q(K, R, n)$, termed the in-degree distribution.

The growth of several man-made or technological networks, such as citation, internet, actor, and scientific co-authorship networks has been measured before [26], and their growth was modeled by a scheme that adds links to new nodes in a way that depends on the degree of a candidate node of the existing network- a mechanism

for network evolution termed preferential attachment [25]. Although these man-made networks have been observed to grow according to preferential attachment, gene networks in *E. coli* and other organisms may instead evolve in response to environmental stressors realized as horizontal gene transfers or gene duplication events [47]. While these and other mechanisms may indeed drive transcriptional network growth, it remains unclear what role they play in the creation and persistence of genetic motifs. Because preferential attachment offers a simplified view of network growth and has been relatively well studied, we employ it here to develop formulas for the creation of directed networks, wherein the network evolution is determined by an attachment kernel taking one of several forms explained below, either linear, power-law, or sigmoid types.

Consider a network of n nodes, wherein each of its nodes labeled by the subscript $i = 1, 2, ..., n$ hosts K_i outgoing links and R_i incoming links. A randomized network is grown by adding nodes one at a time, increasing its size by exactly one node during each round of attachment (also termed a simulation step). These new nodes are attached to the existing network by an average of m directed links to candidate nodes of the network, chosen with equal probability among all existing network nodes. The probability for an edge to link a candidate node i with the new one directed from the candidate to the new one is generally given by $A(K_i, R_i)$, wherein K_i and R_i label the out- and in-degrees of the candidate node, respectively. The probability for a link to be drawn from the new node to a candidate node i is similarly given by $B(K_i, R_i)$. These probabilities are normalized against all nodes of the existing network, and are termed attachment kernels [48].

The expected number of K degree nodes that will have a new bond added in the next iteration can be written as $np(K, n)mA(K)$, wherein p and A are assumed to be independent of the nodes in-degree R. Using this expression, a master equation may

Table 1. Normalized attachment kernels used to create the model networks.

Functional Type	Attachment Kernel (e.g., $A = a/Z$)			
	a	$z = \sum_i a_i$	b	$z = \sum_i b_i$
Linear	K	$\sum_K Kp(K)$	R	$\sum_K Kp(K)$
Power-law ($\gamma = 0.8$)	K^γ	$\sum_K K^\gamma p(K)$	R^γ	$\sum_K K^\gamma p(K)$
Sigmoid	$K/(K+R)$	$\sum_K \sum_R \frac{Kp(K)q(R)}{K+R}$	$R/(K+R)$	$\sum_K \sum_R \frac{Rp(K)q(R)}{K+R}$

be written that describes the evolution of this out-degree distribution:

$$(n+1)p(K, n+1) - np(K, n) = np(K-1, n)mA(K-1) - np(K, n)mA(K), \quad (3.1)$$

Equation 3.1 holds for all cases except $K = m$, which describes the links extending from the new node to the existing network. For this case we have

$$(n+1)p(m, n+1) - np(m, n) = 1 - np(m, n)mA(m). \quad (3.2)$$

Equations 3.1 and 3.2 are difficult to solve exactly. In light of this difficulty we instead simulated the growth algorithm directly using computational means using attachment kernels listed in Table 1, and described by the algorithm given below. Nevertheless, by using suitable approximations for Eqs 3.1 and 3.2, we can infer a general form for the degree distribution; however, the exact relationship reflecting the frequency of degrees observed for network nodes depends strongly on the specific form of the attachment kernel, as demonstrated here.

By taking an approximation valid for very large networks, $n \to \infty$, we can solve for the degree distribution near this limiting value. Here we label $p(K, \infty) = p(K)$, so that $(n+1)p(K, n+1) - np(K, n) \sim p(K)$. Then, Eqs 3.1 and 3.2 become [49]

17

$$p(K) = np(K-1)mA(K-1) - np(K)mA(K), \quad \text{and} \qquad (3.3)$$

$$p(m) = 1 - np(m)mA(m). \qquad (3.4)$$

Equations 3.3 and 3.4 can be solved to give

$$p(K) = \frac{1}{nmA(K)} e^{\sum_{i=m}^{K} 1/nmA(i)}, \qquad (3.5)$$

wherein the attachment kernel is small, i.e., for $A(K) > 1/nm$. Equation 3.5 can be further reduced when the actual dependence of A (or B) on the out- or in- degrees is known. For more details, Please refer to [31] and Appendix B.

3.4 Algorithm to Generate Model Networks

Synthetic networks are grown step-wise according to the following protocol. First, a candidate node, denoted by subscript i here, is chosen randomly with equi-probability from the existing network of size n. Next, a link directed from the candidate node to the new one is drawn if a number selected at random from an equi-probable distribution on the interval $d \in (0,1)$ generally satisfies $d \leq A(K_i, R_i)$. This process is then repeated for a link to be drawn from the new node to the candidate, wherein a newly drawn random number from this same distribution instead generally satisfies $d \leq B(K_i, R_i)$. These steps were repeated $m_i - 1$ times, wherein m_i is another number drawn at random, and the final sequence of such numbers after S growth steps $\{m_l : l = 1, 2, ..., S\}$ satisfies the following exponential distribution:

$$\rho(m_i) = \left(f^{1/(1-m_0)-1}\right) f^{-m_i/(1-m_0)} \qquad (3.6)$$

Parameters here are chosen so that $\rho(m_i = m_0)/\rho(m_i = 1) = f$, with the values

$f = 1/4$ and m_0 varied for creation of the model networks between 2, 3, and 4, which skews the distribution toward larger values of average m_i. The average number of links chosen per growth step, m, is given in terms of these parameters as

$$m = \sum_{m_i=1}^{\infty} m_i \rho(m_i) = \frac{1}{1 - f^{1/(m_0-1)}} \tag{3.7}$$

So, in view of this expression the average number of links supported by model networks built using $m_i = 2, 3$, and 4 is approximately $m = 1.33, 2$, and 2.7, respectively.

The form of this distribution of link enumerations, Eq. 3.6, was chosen partly because the majority of *E. coli*'s genes support only 1 or 2 links, rather than many more. Computer experiments using other link distributions, such as $m_i =$ constant, generated motif participation distributions in greater variance with the *E. coli* distributions than generated using Eq. 3.6 (data not shown here). We note that model networks were built over a seed network of eight nodes fully connected supporting 42 links. This ensures that early in the growth process, when the network is small, it is much less likely for values of m_i to force the creation of duplicate links. That is, more than one link of the same direction connecting two nodes is not permitted.

3.5 Maximum Likelihood Estimation of Cumulative Distribution Functions

Many features of interest in biology when subjected to repeated measurement show a cumulative probability distribution that follows power-law type mathematical relationship [50]. For reasons discussed above, the in-, out-, or total-degree distributions of a network may support a power-law type tail depending on the form of the attachment kernel used to build it (e.g., 3.5). However if there are no a priori theoretical considerations to predict whether experimental data should best fit to a

particular distribution, then curve-fitting methodologies are commonly used to justify empirical relationships among features in these data. It is known, for example, that using a least squares based optimization algorithm does not accurately determine whether the data are power-law distributed Hoogenboom et al., [50, 51].

Addressing this problem, Hoogenboom et al. [51] presented a maximum likelihood estimation based approach that determines whether data are power-law distributed or not. For illustration, let $p(k; \gamma)$ be an out-degree distribution function that depends on a parameter γ, such as $p(K; \gamma) \sim K^{-\gamma}$. A likelihood function is then defined from this distribution so that $L(\gamma) = \prod K p(K; \gamma)$. To find the parameter that best fits the experimental data, this likelihood function is maximized with respect to it. To carry out these analyses on the motif participation and degree distributions extracted from the experimental and synthetic networks described above, we employed MATLAB implementations of the maximum likelihood estimation method of Hoogenboom et al. as described by Clauset et al. [50].

3.6 Results and Discussion

3.6.1 Cumulative Degree Distributions

Figure 2 illustrates the cumulative degree distributions of one representative network generated computationally using the attachment kernels listed in Table 1 for varying distributions of the link enumeration as given by Eq. 3.6, contrasted against the associated distributions arising from the *E. coli* network (black circles). Straight lines are the result of the maximum likelihood estimation of the validity of a power-law fit to these cumulative distributions, $p(\text{degree} \geq K)$, which measures, for example, the probability that observation of the out-degree for any network node is greater than K. The cumulative distribution is related to the degree distribution,
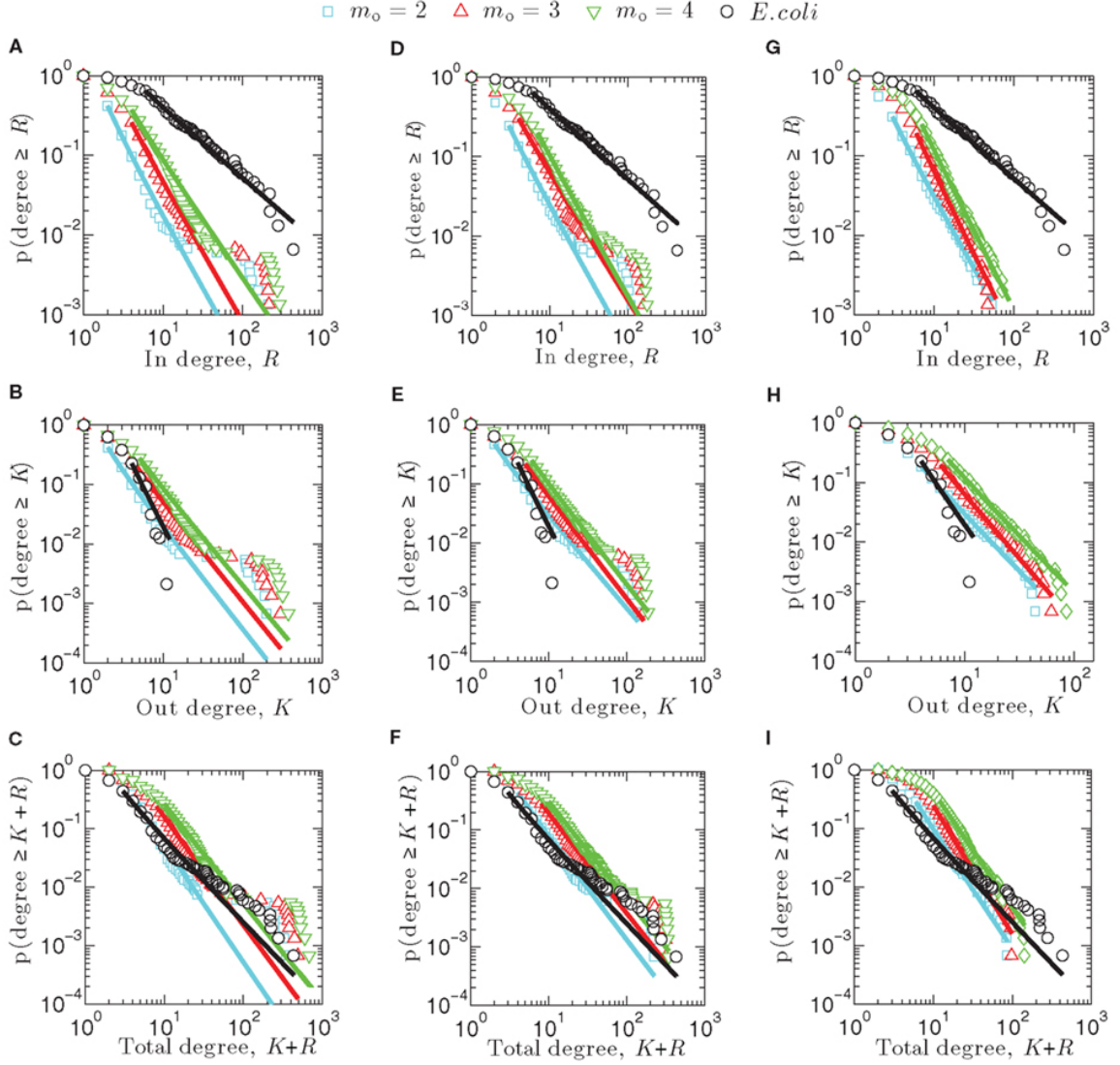
Fig. 2. Cumulative degree distributions for synthetic networks created using linear attachment kernels (A-C), power-law kernels (D-F), and sigmoidal type kernels (G-I).

$p(K)$, by the equation

$$p(\text{degree} \geq K) = \sum_{i=K}^{\infty} p(i) \tag{3.8}$$

Table 2. Scaling exponents α, defined for cumulative distribution functions $p(\text{feature} \geq x) \sim x^{-\alpha}$, identified using the maximum likelihood fitting procedure explained in the section Materials and Methods for the degree and motif participation distributions illustrated in Figures 2 and 3

Distributions		Features			
		In-degree	Out-degree	Total-degree	Motif
Model Networks	Linear	2.768	2.697	2.976	1.828
	Power-law	2.745	2.538	2.737	1.848
	Sigmoid	2.746	2.863	2.995	1.886
Experimental network	*E. coli*	1.871	3.492	2.407	2.008

Data are collected here for networks with $m_0 = 2$

Similar equations exist relating in and total-degree distributions to their associated cumulative distributions.

In, out, and total cumulative degree distributions arising from the linear attachment kernel are displayed here in Figures 2 A-C. Notably, scaling exponents for power-law type equations fit to these distributions, such as $p(\text{degree} \geq K) \sim K^{\alpha}$, do not differ greatly between $m_0 = 2, 3$, or 4; exponents are collected for $m_0 = 2$ networks (cyan in Figure 2) into Table 2. A point-wise inspection of the cumulative total-degree distribution over its whole domain $K + R$, however, closely resembles that for *E. coli* (Figure 2 C), while the cumulative in- and out-degree distributions do not match qualitatively with *E. coli* very well. This observation is consistent with power-law (Figures 2 D-F) and sigmoidal (Figures 2 G-I) attachment kernel constructed networks.

As expected, when more links are added (e.g., $m_0 = 4$) the distributions illustrated in Figure 2 are shifted more toward the right, demonstrating that nodes
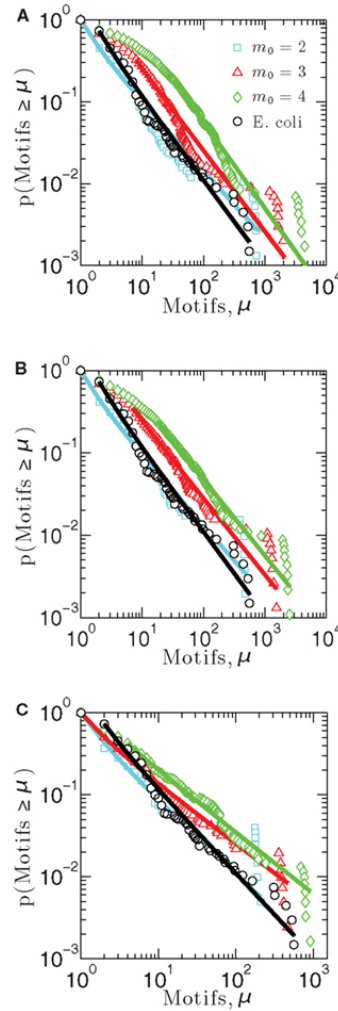
Fig. 3. Cumulative distribution functions measuring the probability that a measurement made on a network node gives a number of motifs greater than μ, for each of three model networks built using (A) linear attachment kernels, (B) power-law kernels, and (C) sigmoidal kernels.

of such networks support larger degrees merely because the density of links has increased. The form of these distributions remains similar, however, appearing to be mostly independent of the choice of m_0. For example, in networks built using the linear (Figures 2 A-C) and power-law (Figures 2 D-F) attachment kernels, a plateau arises in the cumulative distribution that persists across a decade or so in each degree

type. This plateau describes a region in the degree (i.e., x-axes of Figure 2) for which there is constant probability that a measurement of a node's degree gives a greater value than the considered one. Given the definition of the cumulative distribution, Eq. 3.8, the existence of the plateaus mean the degree distributions for the model power-law networks are bimodal, with a longer plateau indicative of a longer span in the degree between maxima of the degree distribution.

The cumulative total-degree distribution for *E. coli*, illustrated by black circles Figures 2 C,F,I, begins to moderately vary from the power-law fit obtained from the maximum likelihood estimation method (black line) at approximately $K + R = 20$, lasting until approximately $K + R = 200$. This variance is not strictly indicative of a plateau, but does hint that power-law-type factors may be ingredients in the evolutionary pressures leading to the shape of the final transcriptional network distribution. Interestingly, preferential attachment mechanisms have indeed been suggested for this purpose yielding scale-free protein-interaction networks (see, e.g., [4]). It was also shown that highly connected genes evolve more slowly (and are therefore older) than their loosely connected peers and that co-expressed genes evolve at similar rates ([52]). (There are, however, exceptions to this conclusion in the case of protein-interaction networks, e.g., [53]) These data suggest preferential attachment contributes to transcriptional network evolution, lending weight to our conclusion that a moderate departure from scale-free topology observed in the *E. coli* (Figures 2 C,F,I) cumulative total-degree distribution data is consistent with a power-law-type preferential attachment growth mechanism. However, the reason even minor bimodality should present in the *E. coli* transcriptional network topology remains unknown.

3.6.2 Participation of *E. coli* Genes in Feed-Forward Loop Motifs

Participation of *E. coli* genes in Feed-forward loops (FFLs) is another distribution which illustrates how many genes are part of one FFL, and how many genes are part of two FFLs, and so on till the maximum number of FFL participation. Figure 3 illustrates the cumulative motif participation distributions for networks constructed using each of the three attachment kernels: linear (Figure 3 A), power-law (Figure 3 B), and sigmoid (Figure 3 C). As with the distributions of Figure 2, scaling exponents for these motif participation distributions are also collected into Table 2.

As the number of motifs associated with a node, μ, increases, the probability that a node will host a greater number of such motifs decreases for all networks (Figures 3 A-C) - a result consistent with the *E. coli* data (depicted with black circles). As expected, when more links are added on average per growth step (i.e., increasing m_0), or more generally as the network density increases, feed-forward loop motifs are more likely to be created by the attachment procedure. This is the reason these cumulative motif participation distributions mostly shift toward the right in Figures 3 A-C with increasing m_0. While differences between the cumulative distribution scaling exponents for these representative networks built using $m_0 = 2$ and *E. coli*'s motif participation distribution are the largest of any m_0 values considered here, these $m_0 = 2$ networks nevertheless more closely resemble the overall *E. coli* motif distribution. Of these, the $m_0 = 2$ network of Figure 3 A provides the closest match to the *E. coli* data for the kernels considered here.

CHAPTER 4

GROWING DIRECTED NETWORKS USING MULTIPLE NODE ATTACHMENT KERNELS

We have seen in chapter 3 that the combined probabilities of the attachment kernels can be used to give random networks with comparable distributions to the original. The attachment kernels were based on nodal properties (e.g., in- and out degrees), the number of linking edges to the substrate were derived from a random probability distribution 3.7, and nodes were attached one at a time. Here we take a different approach, where we consider smaller structural attachments (generally, of sizes = 1, 2, and 3 nodes) at a time. These smaller structures are referred to as "downlinks", and considered structural motifs within *E. coli* [39].

In the supplementary materials of [40], the authors enumerate all possible 3-6 node transcriptional motifs. Among the most common transcriptional motifs observed in GRNs of the model bacterium *Escherichia coli* (herein *E. coli*) and the baker's yeast *Saccharomyces cerevisiae* (herein labeled Yeast), are feed-forward loops (FFLs) and bifans (BFs), which can be observed natively in Figures 4 a and b. An FFL is hierarchically composed of three genes, a top-level "parent" gene which regulates two "child" genes, wherein one of the child genes regulates the other. This specific topology allows for interesting dynamical consequences, such as pulses, signal delays, and irreversible speed-ups [42]. In contrast, BFs constitute four genes, two of which simultaneously regulate the other two; these motifs have been reported as constituents of dense overlapping regulons in the GRN "backbone" responsible for vital life functions, such as nutrient metabolism and bio-synthesis [54].

26
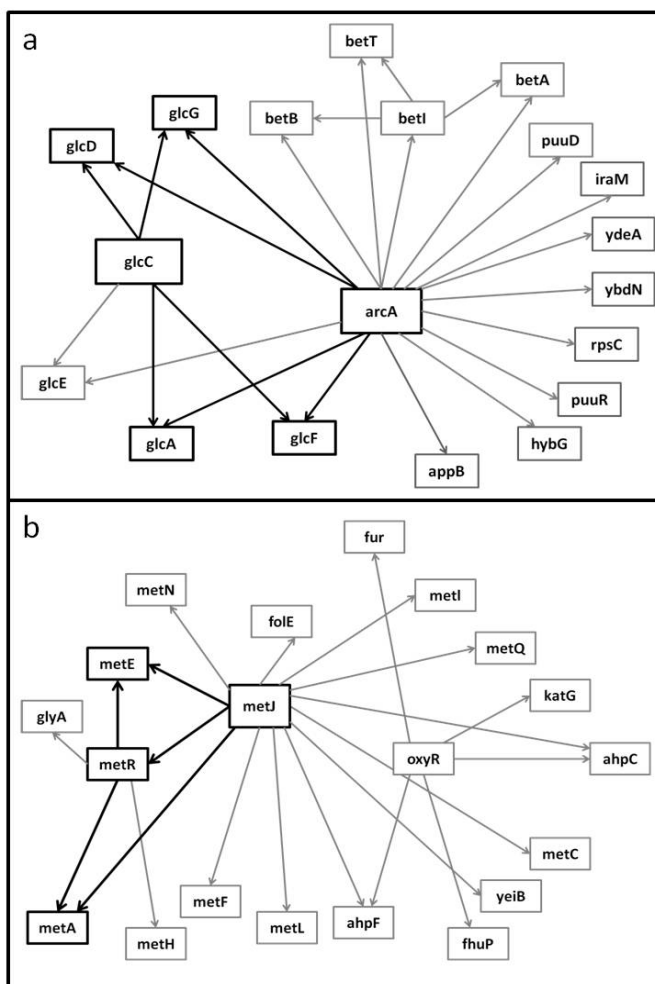
Fig. 4. Embedded within sample GRN subgraphs of *E. coli*, the topological represen-
tation of (a) bifans. Here transcription factors arcA amd glcC coregulate glcD
and glcG. On the other hand (b) the feed-forward loop constitutes a tran-
scription factor (such as metJ) that regulates both a gene (metE) and another
transcription factor (metR). The regulated transcription factor co-regulates the
same gene (metR → metE).

It is notable to point out that many motifs are a product of the coupling be-
tween the subnetworks illustrated in Figure 5: the uplink, the downlink, and the
three chain. For instance, a BF can be viewed as two downlinks coupled by sharing
both child genes, while an FFL can be viewed as an uplink or a downlink sharing all
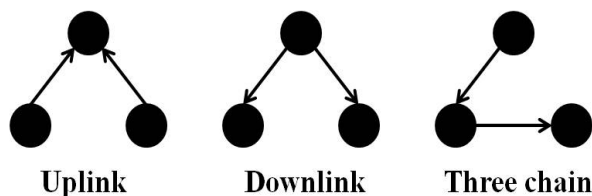
Fig. 5. The three-node two-edge motif substructures.

three genes with a three-chain. Moreover, we have conducted computational analysis to estimate the percentages of the gene-regulatory interactions that participate in these components for an *E. coli* GRN. We observed that 54.7% of interactions are involved with FFLs, 82% with BFs, 99.4% by downlinks, 83.9% by uplinks and 78.3% by three-chains. Given these data for *E. coli*, we hypothesize that downlinks represent a primary component in the evolution of GRN topology. Despite that the impacts of motif-coupling on the functionality of GRNs remains largely mysterious, some results have been reported in this particular area. For example, investigations of gene coupling for different motif patterns have been conducted using mathematical modeling of transcription and translation in order to reveal substructure functionalities [44, 45, 41]. Additionally, experiments have revealed that bacteria can endure a great deal of regulatory interaction rewiring via manipulation of protein-binding DNA sequences [43].

In this dissertation, we extend the prior algorithm of Chapter 3 based on the following two criteria:

1. Our modified preferential attachment algorithm was oblivious to the distinction of the two different types of nodes in transcriptional networks: genes and transcription factors (TFs). Since transcriptional networks only allow TF-to-TF and TF-to-gene edges, a distinction between these biological classes that restricts allowed bonds may improve fidelity of the "grown" networks to that

from *E. coli* or other GRNs.

2. The previous algorithm considered attachment of one node at a time to the substrate network for growth following the general premise of preferential attachment. However, this failed to generate the correct FFL motif distribution of the nodes in the grown network as compared to the GRN of *E. coli*. In this dissertation, we consider the attachment of an entire downlink motif at a time using a preferential attachment methodology. One or more of the three nodes of the incoming downlink may be shared with selected nodes in the substrate network resulting in the growth of the network by one (if two vertices are shared between the incoming downlink and substrate network) or two nodes (if one vertex is shared between the incoming downlink and substrate network) or zero nodes (if all three vertices of the incoming downlink is shared with corresponding three vertices in the substrate) at a time. The motivation for a downlink-based preferential attachment model stems from an observation that 99.4% of the nodes in the GRN of *E. coli* participate in downlinks.

The algorithm produces more comparable results for different subnetworks of *E. coli*, and the results are illustrated in terms of the maximum likelihood estimation (chapter 3.5). Furthermore, the algorithm discussed in this chapter was published in [55].

4.1 Transcriptional Network Datasets

To evaluate the fidelity of artificially constructed networks, we sampled subnetworks from the entire body of the *E. coli* transcriptional network, herein referred to as "target networks." As mentioned above, we defined two types of nodes arranged hierarchically in these GRNs, classified as either (a) genes or (b) transcription factors, and defined such that genes reflect a regulatory terminus wherein they do not

regulate other nodes (i.e. have no outgoing links), and transcription factors are nodes that regulate genes. Consequently, there are three possibilities for the class of nodes that constitute a downlink motif:

1. three transcription factors (herein TTT);

2. a transcription factor regulating two genes (herein TGG); or

3. a transcription factor which regulates another transcription factor and a gene (herein TTG).

All transcriptional interactions of *E. coli* GRNs have been validated experimentally [39], and target networks have been rendered using GeneNetWeaver [46]. GeneNetWeaver provides options for sampling subnetworks from the GRNs of both *E. coli* and *S. cerevisiae*. For simplicity, we have removed them from the target networks considered here.

4.2   Vertex-based motif networks and downlink coupling

Conventional preferential attachment models estimate the attachment probability from the degree of single candidate nodes in the target networks. However, to conceptualize a downlink-based preferential attachment method, which is a collection of nodes, we must first identify a way to express a downlink motif from the substrate network into a single, effective "lumped" node.

To achieve this we propose to apply a network transformation to the *E. coli* LCC, defined so that each node of the transformed network represents a downlink derived from the LCC; downlink "nodes" are connected to others with edges weighted by the number of nodes shared between the two downlink motifs. For example, two downlink motifs that share a single node would equate with two nodes connected
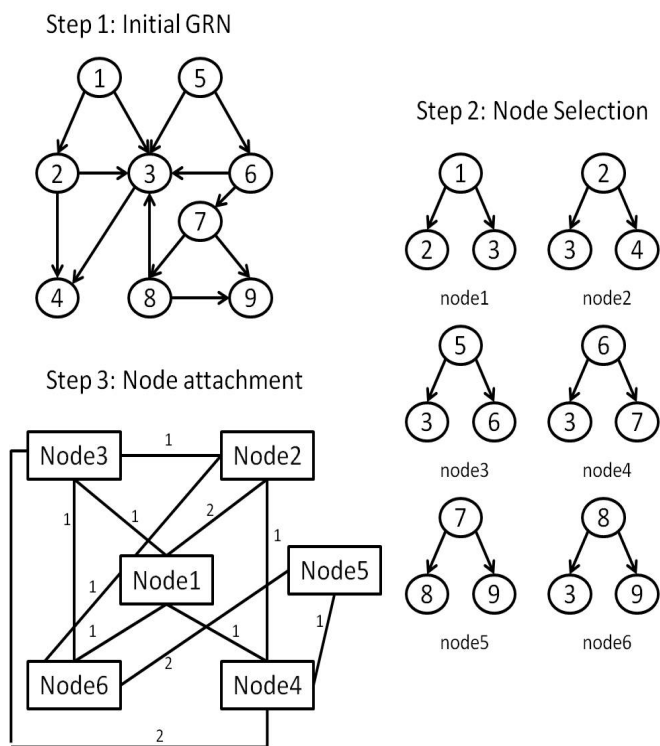
Fig. 6. The steps for forming the VMN from a GRN: (Step 1) An initial GRN is considered. (Step 2) A list of the downlink structures is derived from the GRN giving each downlink structure it's unique id. (Step 3) Each downlink's constituent nodes are contrasted with every other downlink's nodes. Downlinks form topological interactions in the VMN if they have at least one common node. The strength of the interaction is equivalent to the number of shared nodes between the corresponding downlinks.

by a single link of unit weight. Herein we term such a resultant network, a vertex-based motif network (VMN). An illustration of this graph transformation is shown in Figure 6. VMNs are therefore manifestly undirected networks. Although *E. coli* is sparse ([56]), its equivalent VMN contains many more nodes due to the approximately 278,000 downlinks supported in the network, most of which share nodes due to the hierarchical nature of the *E. coli* GRN. Therefore, its VMN is dense

Figure 7 contrasts differences in the total degree distributions of three sample
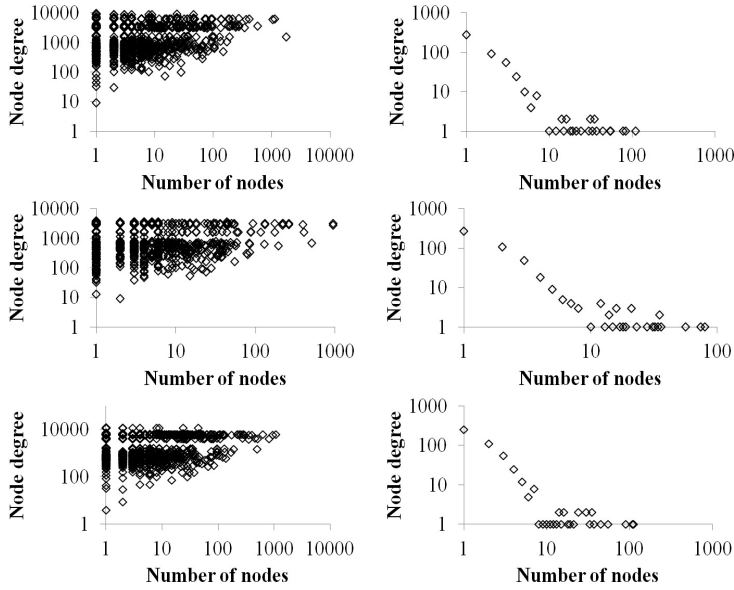
Fig. 7. A plot of the number of nodes (vertical axis) vs. the cumulative degrees (horizontal axis) of VMNs (left) as compared to their respective GRNs (right).

GRN subnetworks of sizes $n = 500$ (right panels) with their corresponding VMNs (left panels). Some VMNs reached as much as 400-fold the number of nodes as their original subnetwork. Finally, we note that degree distributions exhibited by VMNs indicate an absence of correlation in the abundance of shared vertices among downlink motifs.

4.3 Data Representation

Computationally, we have represented GRNs and VMNs using square matrices, respectively labeled G and V. A GRN link from node j and incident on node k is represented by $G_{jk} = 1$, and the absence of such connection is represented by $G_{jk} = 0$, similar to an adjacency matrix. Because GRN links carry no weight, the matrix G may only hold values of 0 and 1. In G of size n, the downlink count S_{DL} can be

determined mathematically using the equation:

$$S_{DL} = \frac{1}{2} \sum_{a=1}^{n} \sum_{b=1}^{n} \sum_{c=1}^{n} [G_{ab} \cap G_{ac}]. \tag{4.1}$$

However, V differs from G in that it is symmetric with elements given by the weights 0, 1, 2 and 3, depending on the number of vertex overlaps between one downlink and another. Therefore, $V_{lm} = V_{ml} = 0$ if downlinks l and m do not share any nodes, $V_{lm} = V_{ml} = 1$ if downlinks l and m share one node, and so on.

4.4 Algorithm for Network Growth

A subnetwork of a target network, termed a "substrate," accumulates one downlink per attachment step. Table 3 illustrates possible downlink-to-downlink attachments, as based on the number of vertices shared between a candidate and incoming downlink motif (DLs). In order to determine the appropriate attachment, the following steps are considered.
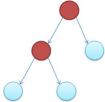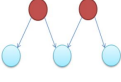
4.4.1 Step 1 - Determine candidate downlink type

In order to select an existing downlink from the substrate network as a candidate for attachment, its type needs to be specified. We denote the sums of the three downlink types as N_{TGG}, N_{TTG} and N_{TTT}, such that

$$S_{DL} = N_{TGG} + N_{TTG} + N_{TTT}. \tag{4.2}$$

Using Eq. 4.2, the probability that a selected candidate downlink is of type TGG, TTG, or TTT is determined by, $P_{TGG} = \frac{N_{TGG}}{S_{DL}}$, $P_{TTG} = \frac{N_{TTG}}{S_{DL}}$, and $P_{TTT} = \frac{N_{TTT}}{S_{DL}}$ in that order. These probabilities are later used as selection kernels to determine the type of candidate downlink. A random number, r_1, is generated with uniform probability on the interval $r_1 \in [0, 1]$. If $0 \leq r_1 < P_{TGG}$, a TGG downlink is considered

33

Table 3. Every type of potential downlink-to-downlink attachment.

Category	Pattern id	Pattern graph	Attachment description	Applicable DL-DL combinations
One node attachment	P1		Root TF coupling.	TGG-TGG, TTG-TGG, TTT-TGG, TGG-TTG, TTG-TTG, TTT-TTG, TGG-TTT, TTG-TTT, TTT-TTT
	P2		leaf TF to root TF coupling.	TTG-TGG, TTT-TGG, TGG-TTG, TTG-TTG, TTT-TTG, TGG-TTT, TTG-TTT, TTT-TTT
	P3		leaf gene coupling.	TGG-TGG, TTG-TGG, TGG-TTG, TTG-TTG, TTT-TTG, TTG-TTT, TTT-TTT
Two node attachment	P4		(1) Root TF coupling, and (2) one leaf gene coupling.	TGG-TGG, TTG-TGG TGG-TTG, TTG-TTG, TTT-TTG, TTG-TTT, TTT-TTT
	P5		(1) Root TF couples with leaf TF, and (2) one leaf TF couples with root TF.	TGG-TTG, TTG-TTG, TTT-TTT
	P6		(1) Leaf TF couples with root TF, and (2) one leaf gene couples with leaf node.	TTG-TGG, TGG-TTG, TTG-TTG, TTT-TTG, TTT-TTT
	P7		(1) Leaf gene couples with leaf gene, and (2) one leaf gene couples with leaf gene.	TGG-TGG, TTG-TTG, TTT-TTT
Three node attachment	P8		(1) Root TF couples with leaf TF, and (2) one leaf TF couples with root TF, and (3) one leaf gene couples with leaf gene.	TTG-TTG, TTT-TTT

Transcription factors are red, and genes are blue.

as a candidate for attachment. If $P_{TGG} \leq r_1 < P_{TGG} + P_{TTG}$, a TTG downlink is considered. Otherwise a TTT is considered for attachment.

4.4.2   Step 2 - Selection of candidate downlink

A VMN is created from the downlinks subscribing to the type selected in Step 1 and the preferential attachment mechanism is employed ([25]). A random downlink l

is picked with uniform probability, and it's degree centrality is calculated as follows:

$$C_l = \frac{\sum_{a=1}^{t-1} V_{la}}{\sum_{a=1}^{t} \sum_{b=1}^{t} V_{ab}}, \tag{4.3}$$

wherein t represents the total number of downlinks in the VMN. Next, a random number $0 \le r_2 \le 1$, is compared with C_l such that if $r_2 < C_l$, l is selected as a candidate downlink. On the other hand, if the condition is not satisfied another downlink is picked at random and the process is repeated.

4.4.3 Step 3 - The type of incoming downlink

Incoming downlinks may be either of the three downlink types, generated at random with uniform probability.

4.4.4 Step 4 - The number of shared nodes

A similar strategy to that of Step 1 is implemented, except that the probability distribution depends on the number of shared nodes between pairs of downlinks and not the number of each type of downlink. There are $S_{pair} = S_{DL}(S_{DL} - 1)/2$ total cases of downlink pairs sharing nodes, each of which can share 0,1, 2, or 3 nodes. Since our model does not account for disjoint components, we ignore the cases where downlink pairs share no nodes. We denote the number of pairs sharing 1, 2 and 3 nodes as N_{s1}, N_{s2}, and N_{s3} respectively. Consequently the probabilities for node sharing can be determined by $P_{s1} = \frac{N_{s1}}{S_{DL}}$, $P_{s2} = \frac{N_{s2}}{S_{DL}}$, and $P_{s3} = \frac{N_{s3}}{S_{DL}}$. Next a third random variable $0 \le r_3 \le 1$ will be compared with the ranges $(0, P_{s1})$, $(P_{s1}, P_{s1} + P_{s2})$ and $(P_{s1} + P_{s2}, P_{s1} + P_{s2} + P_{s3})$ respectively to determine the number of shared nodes as was done in Step 1.

Table 4. Applicable Downlink to Downlink attachments for a given candidate down-link, incoming downlink and number of vertex overlaps.

	DL-DL combination	Applicable patterns	DL-DL combination	Applicable patterns	DL-DL combination	Applicable patterns
One node attachment	TGG - TGG	{P1, P3}	TTG-TGG	{P1, P2, P3}	TTT-TGG	{P1, P2}
	TGG - TTG	{P1, P2, P3}	TTG-TTG	{P1, P2, P3}	TTT-TTG	{P1, P2, P3}
	TGG - TTT	{P1, P2}	TTG-TTT	{P1, P2, P3}	TTT-TTT	{P1, P2, P3}
Two node attachment	TGG - TGG	{P4, P7}	TTG-TGG	{P4, P6}	TTT-TGG	NA
	TGG - TTG	{P4, P5, P6}	TTG-TTG	{P4, P5, P6, P7}	TTT-TTG	{P4, P6}
	TGG - TTT	NA	TTG-TTT	{P4}	TTT-TTT	{P4, P5, P6, P7}
Three node attachment	TGG - TGG	NA	TTG-TGG		TTT-TGG	NA
	TGG - TTG	NA	TTG-TTG	{P8}	TTT-TTG	NA
	TGG - TTT	NA	TTG-TTT		TTT-TTT	{P8}

4.4.5 Step 5 - The attachment pattern

Knowing the candidate downlink, the type of incoming downlink and the number of nodes to be shared (or overlapped), we can use Table 4 to proceed with an attachment. For example, having selected a candidate TGG, an incoming TTG, that will share two nodes, from Table 4 we are only allowed to proceed with three attachment patterns {P4, P5, P6}. Each pattern is given an equal probability of being chosen (here 1/3). A process similar to the random number generated in Steps 1 and 4 is used to determine which pattern will be chosen.

4.5 Results and Discussion

4.5.1 Fidelity of the downlink-based preferential attachment mechanism

We extracted 5 different target networks of 100 nodes from the *E. coli* LCC using the GeneNetWeaver software in the manner explained above. We extracted substrate subnetworks upon which to "grow" new networks from these target networks of relative sizes equal to 10, 20, 30, and 40 nodes. We sampled 5 substrates of each

size, resulting in a total of 20 substrate subnetworks per target network derived from the *E. coli* LCC. Each substrate network was grown to a size of 100 nodes using two algorithms: (i) the attachment kernel (linear, power-law and sigmoidal) method as presented in [31]; and, (ii) the downlink based attachment mechanism explained above.

For networks generated using the downlink based preferential attachment mechanism we calculated the 3 types of downlink attachment probabilities in two ways. In the first method, termed "target attachment," values for the fraction of downlinks of each type, $P_{TGG}, P_{TTG}, P_{TTT}$, and fractions of downlinks that share one (P_{s_1}) two (P_{s_2}), and three (P_{s_3}) vertices were all calculated from the target networks derived from the *E. coli* LCC. This method is biased, given that we must use the structure of the biological networks to inform that of the "grown" networks. The second method, termed "substrate attachment," calculates the same probabilities as the first method, but iteratively from the current state of the grown network. This method is unbiased, in the sense that it is ignorant of the final topology of the target network.

Degree distributions of the "grown" networks were fitted to the data using a power-law equation, and each of the two methods was compared individually to the fitted exponents of the biological networks as a measure of their fidelity. Exponents, γ, were estimated for in-, out-, total degree distributions (Table 5), but also for distributions relating the participation of nodes in downlink substructures (Table 6). A lower value for the difference in fitted exponents suggests a higher fidelity of the attachment model to the properties of the "target" biological network. As can be seen from Table 5, fidelity of the degree distributions between grown and target networks is higher for downlink-based attachment mechanisms as compared to the attachment kernel method of [31].

Error bounds for the distribution of nodes participating in downlink substruc-

tures show similar traits to that observed for the degree distributions. Out of the five substrates, the fifth network had marginally better distributions when grown with single node attachments for the same reasons explained above. Additionally, using the probabilities calculated from the target network (i.e., "target attachment, Tables 5 and 6) does not always lead to higher fidelity, as can be seen in the fourth and fifth networks. This is again because quite a few nodes do not participate in downlink structures and hence the probability distributions from the goal network make the counts skewed.

4.5.2   Evolutionary mechanisms and downlink-based network growth

Preferential attachment mechanisms have been suggested, sometimes in addition to other mechanisms (e.g., duplication events), as models of evolutionary formation of gene-regulatory ([57]), protein interaction ([58]), and metabolic networks ([59]). For gene-regulatory networks, mutations to DNA bases may alter the affinity of DNA-binding proteins or cis-regulatory modules to result in rewiring or admission of novel regulatory interactions ([60]). It is plausible that evolutionary mutations to DNA sequences result in creation of whole downlink transcriptional modules over a single generation, given the local nature of cis-regulatory mutation mechanisms and the potential for gene duplication events. For example, base-pair mutations can alter the availability of new binding sites, which manipulates the "distance" between interacting sites via insertion or deletion of cis-regulatory modules or sub-functionalization due to regional duplications, among others ([60]). At the system level, correlations between mutations over successive generations may be needed to consistently evolve new cis-regulatory modules and gene-regulatory interactions. However, even a node-by-node attachment mechanism (i.e., DNA sequence mutations that result in a single novel gene-regulatory interaction) holds potential for multiple novel gene-regulatory

38

interactions formed over a single generation ([57]), which may explain the fewer nodes in the GRN observed to not participate in downlink modules. This can be linked to the error bounds generated for the fifth substrate, where results are marginally better for single node attachments; in this network only approximately 80% of the nodes participated in downlink motifs as opposed to $\geq$90% for networks labeled 1-4.

It is currently difficult to directly test hypotheses regarding network 'growth' mechanisms due to experimental difficulties in manipulating the evolution of transcriptional networks in microorganisms such as bacteria. An attempt to experimentally emulate the "bottom up" approach employed in many attachment or duplication based network growth mechanisms, such as the motif-based attachment method proposed in this paper, may be therefore impractical with current technologies. One alternative might be to reverse the growth process. Transcriptional regulatory networks, such as the *E. coli* network dataset analyzed here, serve as target states of the growth mechanisms; beginning with these fully formed networks and sequentially "deactivating" regulatory interactions between genes and transcription factors may provide valuable insight into the processes that formed them. For example, protein production could be suppressed with RNAi tailored to specific mRNA, thereby eliminating a regulatory interaction by preventing protein proliferation; another strategy could be to target a transcription factor's activated state, perhaps by interfering with phosphorylation/dephosphorylation reactions through crosstalk ([61]), thus modulating its binding affinity to the correct DNA sequence and preventing gene activation. As a proof of principle, some experimental efforts have already succeeded in extensively "rewiring" *E. coli*'s transcriptional regulatory network ([43]). Even so, future work is needed to predict dynamical consequences of adding or removing regulatory interactions specific to the attachment mechanism (in our case, regulatory interactions associated with downlink motifs), which could be evaluated using these or other

experimental methods.

Recent developments in "in vitro" circuit design using microfluidic cell-free systems for the rapid prototyping of synthetic genetic networks as a "biomolecular breadboard" for molecular programming ([62]) is another promising avenue for experimentally validating the network growth principles proposed here. The biomolecular breadboards project has successfully synthesized different types of feed-forward loop motifs ([63]) and can be extended to design coupled FFL circuits. Similarly, such synthetic biology circuits of coupled downlink motifs can experimentally validate the dynamical consequences of our proposed network growth method thereby creating new hypotheses on whether coupled downlinks exhibit any preferences in natural selection. Currently however, this can only be achieved at a smaller scale by synthesizing small networks of connected downlinks.

Table 5. Statistics for the difference between power-law exponents of candidate and target network's degree distributions resulting from either the Attachment Kernel method reported in ([31]), or from the Downlink Attachment method reported here.

Attachment Probability	Networks														
	1			2			3			4			5		
	In	Out	Tot.	In	Out	Tot.	In	Out	Tot.	In	Out	Tot.	In	Out	Tot.
Attachment Kernel Method															
Linear	0.91	0.94	0.81	0.25	0.55	0.18	0.86	0.74	0.63	1.18	0.87	0.75	0.8	1.92	0.21
	±0.6	±0.6	±0.6	±0.3	±0.2	±0.1	±0.4	±0.3	±0.6	±0.5	±0.4	±0.7	±0.5	±0.2	±0.3
Power-law	1.09	1.08	0.99	0.23	0.57	0.16	0.8	0.71	0.73	1.09	0.99	0.46	0.88	1.91	0.19
	±0.5	±0.5	±0.7	±0.2	±0.2	±0.1	±0.4	±0.4	±0.7	±0.5	±0.2	±0.6	±0.6	±0.2	±0.4
Sigmoidal	0.92	0.98	0.97	0.42	0.63	0.15	1.01	0.66	0.82	1.25	0.65	1.09	0.62	1.91	0.3
	±0.6	±0.5	±0.7	±0.3	±0.1	±0.1	±0.5	±0.3	±0.6	±0.5	±0.4	±0.6	±0.5	±0.2	±0.2
Downlink Attachment Method															
Target Attachment	0.08	0.96	0.13	0.38	0.21	0.07	0.62	0.12	0.1	0.22	0.44	0.07	1.89	1.9	0.35
	±0.1	±0.6	±0.1	±0.0	±0.3	±0.1	±0.5	±0.1	±0.0	±0.2	±0.3	±0.0	±0.0	±0.0	±0.3
Substrate Attachment	0.16	1.4	0.61	0.37	0.69	0.02	0.38	0.9	0.36	0.48	0.94	0.37	1.9	1.9	0.41
	±0.1	±0.2	±0.4	±0.0	±0.2	±0.0	±0.0	±0.0	±0.6	±0.7	±0.2	±0.3	±0.0	±0.0	±0.4

Table 6. Statistics or the difference between fitted power-law exponent for candidate and target networks' distributions of genes participating in downlinks.

	Networks				
Attachment Probability	1	2	3	4	5
Attachment Kernel Method					
Linear	1.17	1.19	0.79	1.32	0.33
	±0.5	±0.5	±0.4	±0.4	±0.3
Power-law	0.9	1.27	0.72	1.07	0.51
	±0.5	±0.6	±0.1	±0.5	±0.2
Sigmoidal	1.43	1.1	0.86	1.56	0.15
	±0.0	±0.3	±0.1	±0.1	±0.1
Downlink Attachment Method					
Target Attachment	0.67	0.43	0.34	0.75	0.63
	±0.2	±0.1	±0.0	±0.4	±0.6
Substrate Attachment	0.75	1.2	0.34	0.69	0.62
	±0.3	±0.5	±0.0	±0.4	±0.6

CHAPTER 5

APPLICATIONS: BIO-INSPIRED WIRELESS SENSOR NETWORKS

Wireless Sensor Network (WSN) simulations constitute a big portion of this research because they are used to determine *In-Silico* network routing efficiency. This chapter serves as a motivation for considering WSNs adopting bio-topologies vs other networks as well as shed light on "Biological Robustness". Moreover the mapping from *In-Silico* to WSN is provided.

5.1 Biological Robustness and Applications

A biological network is a graphical representation of a system of interacting organic components such as cells, tissues, organs, etc. For example two proteins binding are represented with two nodes (the proteins) and a bidirectional link (the interaction) in a protein-protein interaction network [64]. Principles such as swarm intelligence [65], artificial immune systems [66], cellular signaling networks and many others [32] have existed for millions of years. Hence, their evolution resulted in developing captivating features such as robustness to internal component failures, acclimating to external changes, self organization and efficient signaling, collaborative tasking and their ability to evolve through learning [32, 33, 34].

All the above conceptualize what is known as 'Biological Robustness' [17]- a distinctive quality mainly attributed to their bio-topologies [18]. As a result, many research has been directed towards adopting these topologies to solve networking related problems. For example, 'Ant Colony Optimization' is used in 'swarm intelligence' to enhance routing protocols in WSNs and techniques for distributed search

and optimization [65]. A proposed self organizing WSN, i.e adopting cellular signaling network schemes, exhibits optimal topologies post nodal failure, thereby guaranteeing optimal packet transmission [67]. Moreover, WSNs can be mapped to an artificial immune system, where B-cells can be viewed as sensor nodes, antibodies as sensor data, T-cells as rate control parameter, antigen as estimation distortion and natural extinction as packet loss [68]. Many other bio-inspired networking applications can be found in [34].

5.2 Biological vs. Exponential topologies

Precisely what makes such bio-topologies optimal? "Sparseness" or "loose connectedness" [56]. Biological networks degree distributions can be viewed using a power law, $p(k) \sim k^{-\gamma}$ [69]. In a log-log plot of the different nodal degrees (in- out- and cumulative) vs. the nodes that have such degrees, $p(k)$ will results in a steep negative slope. Therefore, the bulk of the nodes have degrees much lower than the average degree, $\langle k \rangle$, while few nodes have degrees much higher than $\langle k \rangle$ (i.e hubs) [24]. This formation results in loosely connected components. Prime examples of such graphs are the transcriptional networks of *E. coli* and Saccharomyces Cerevisiae (herein Yeast), where most genes have no outgoing edges and single incoming edges.

Most biological networks including *E. coli*'s and Yeast's transcriptional regulatory networks (TRNs) have $2 < \gamma < 3$ [69]- a property which classifies a network as 'scale-free' (SF) [24]. Having multiple low degrees to few hubs formations creates an increased probability of having random attacks targeting lower degree nodes, which therefore results in loss of insignificant number of links [36]. Similarly with edge rewiring, the relative damage to the whole network will not be afflictive. However targeted hub attacks can be costly and result in disjointing the network [70].

Other topological classifications belong to two other major categories, namely

44

Erdö-Rényi (ER) [71] and small-world (SW) [28] networks. ER considers graphs extracted out of a complete network with a prescribed number of nodes and edges, provided the edges are selected with equal probability. SW networks are inclined towards minimizing the number of hops between pairs of nodes during their growth. In contrast with SF networks, interconnections grow adopting 'the rich get richer and the poor get poorer' criterion. ER and SW networks typically have tightly connected components, which is different from that of scale-free networks. The probability of a selected node of a particular degree is $p(k) \sim \langle k \rangle$, therefore making random attacks as equally costly as targeted attacks [24].

5.3 Mapping Transcriptional Regulatory into Wireless Sensor Networks

Similarities between TRNs and WSNs are explained through an activity named transcription. During transcription, genes process signals from nearby neighbors in the form of transcription factors, proteins of antithetic degrees that stimulate/inhibit genes to/from generating mRNA molecules, which encrypt a chemical to be synthesized at the receptor gene [23]. Therefore TRN nodes communicate and alter one another decisions through sending signals (transcription factors), which are in return processed into output signals (mRNAs), a process which is similar to WSNs were sensors receive packets from nearby neighbors with packet forwarding instructions to other nigh destination points (sensors). As a result, any node in a network (TRN and WSN) can affect the decision of every other node and the overall performance of the grid in instances of nodal collapse.

A single transcription factor can regulate other transcription factors and genes per unit time, while genes can never regulate other nodes. These molecules having half-lives $T_{1/2} = \ln (2)/k$ [72], were k represents the decay rate constant, are subject to degradation. Similarly in the case of WSNs, packets are forwarded from source nodes

45

to sink nodes are dropped if they exceed the queue length. Therefore, genes can be considered as sink nodes and transcription factors as source nodes. On that account, we describe one measure of robustness in WSNs that adopt biological topology as the ability for each node in the network to deliver information to their local sink nodes with minimal packet loss.

Furthermore, WSNs adopting TRN topologies have proven to perform better in terms of packet transmission than randomly generated networks both; in small-[73] and large-scale [21] networks. Because of such reasons, we incorporate WSNs adopting transcriptional network graphs to aid in selecting topological features that promote efficient packet receival rates.

Here WSN Transmission scenarios were evaluated using NS-2 simulations [74]. A duplex-link is established between the nodes (based on input files) that have edges in TRNs. A bandwidth of 1 Mb is assigned for each link in this simulation, with a packet interval of 1 ms. Ten simulations are executed for every category of observation using a flooding routing protocol. Information is transmitted at periodic intervals and the packet receipt rate at the sink node is recorded. A packet-loss model is used to evaluate the performance of each network of fixed size, in which packet-loss of 0 - 50% is considered.

A note on the packet loss model. The loss model percentage defines the level of channel noise along the paths for which packets traverse a WSN, i.e. every link in the network. Packets transmitted in a 0% loss model will experience no drop along the transmission links, while a 100% channel noise (loss) will guarantee a failed transmission. The loss model is a function of the individual links where each link has the exact noise level.

CHAPTER 6

EVALUATING NETWORK ROBUSTNESS IN WIRELESS SENSOR NETWORKS

As mentioned in section 1.3, the main aspects of network robustness are namely performance, connectivity and tie formations. In this section we discuss metrics that capture these phenomena in WSNs. Moreover, we analyze the networks ability to preserve such characteristics post disruptions. Here we consider consecutive random edge deletions as disruptions. A similar approach was used for connectivity analysis in [24] for which the authors have concluded that scale-free networks are more robust than exponential (or random) networks as their diameters suffer at a lesser rate post random node deletions.

Random failure analysis are conducted for WSNs adopting *E. coli*'s scale-free topologies vs. WSNs adopting exponential topologies generated at random of similar sizes (i.e., nodes and links). Networks of node sizes $= 100, 200, .., 600$ were generated for this study. Each network starts with an initial number of links and undergoes a single random edge deletion, then it's metrics are recorded (e.g clustering coefficient)- this process is repeated for 60% the number of link.

6.1 Types of Networks And Data Representation

In this section we explain the process of acquiring the different types of networks, namely; transcriptional regulatory, scale-free, and random networks.

6.1.1 Transcriptional Regulatory Networks

TRNs are represented using a digraph $G(V, E)$, where each node in V represents a gene and arcs in E represent a regulation of one gene to another. In this study we limit our research to the topology of TRNs and do not consider the regulation types, i.e whether or not a regulation is stimulatory or inhibitory. Furthermore, a TRN is composed of two types of proteins: (1) transcription factors (or TFs) and (2) regulated genes, where a transcription factor can regulate other TFs and genes, however, genes do not regulate others. This is not to be confused with gene regulatory networks (GRNs) where gene-gene edges are possible.

All our sample networks are derived from the TRNs. A sample TRN is needed as input for generating SF networks (section 6.1.2), while it's dimensions (nodes and edges) are required for ER networks (section 6.1.3). TRN subnetworks of user defined sizes are rendered using GeneNetWeaver [46] tool- originally developed as an application to help researchers in evaluating network inference algorithms.

The skeletons which embodies all TRN extracted subnetworks are that of the complete TRNs of *E. coli* and Yeast, both available on GeneNetWeaver tool. Both networks are considered scale-free (section 5.2) and are completely mapped. *E. coli* is composed of 23 disjoint components together forming 1565 nodes and 3758 edges, while Yeast is a single network of 4441 nodes 12873 edges. Depth First Searches (DFS) are used to ensure that extracted subnetworks have no isolated nodes. Furthermore, self loops associated with regulatory TFs capable of up/down regulating themselves are deleted to avoid computational overheads- though crucial for TRN functionality, they do not contribute to WSN performances.

6.1.2 Scale-free Networks

SF networks are products of randomizing the extracted TRNs (section 6.1.1). Network randomization is done using the method of "switches" [40], which preserves the degrees of the nodes. The TRN remains SF, however, it is an entirely different network. The randomization starts with four distinct nodes A, B, C and D, where an edge is incident from A unto B and another from C unto D. Both edges are split in half and rewired such that A becomes incident on D and C becomes incident on B. This process is repeated a minimum of $|E|$ times. The end result is a network having the same γ, however, substructural counts like for instance a three chain loop (e.g., A→B→C→A), will not be the same- also referred to as a "randomized network". **Note**, these substructures are termed motifs and are explained in section 7.1.5.

6.1.3 Random Networks

For several of the TRNs extracted from *E. coli* and Yeast, random networks of similar dimensions are generated using a variant of the ER model [71]. The process starts by prescribing a number of nodes n and edges m. Next, m edges are selected with equal probability from the completely connected n-node graph. The resulting network is checked for connected-ness using DFS, if the network contains any isolated components the selection process is repeated, otherwise the resulting network suffices. An example of the different types of networks is given in Figure 8. **Note**, this algorithm is exactly the same as the ER model, however, ER considers disjoint network sampling. In our case, we disregard a network if the random sample contains disconnected components.

(a) Transcriptional   **(b) Scale-free**   **(c) Random**
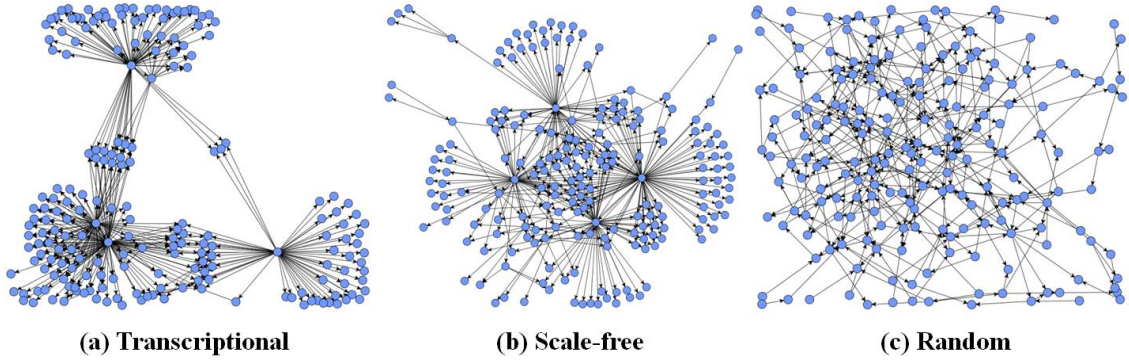
Fig. 8. A sample 200 node (a) transcriptional subnetwork of *E. coli* (regulation type not shown), (b) a scale-free randomization of similar degree sequences, and (c) a random network of similar dimensions ($|V|$ and $|E|$).

6.1.4   Network Representation

Our generated sample networks are represented using a directed graph $G(V, E)$. V is composed of a set of source genes $u \in V$, and destination genes $v \in V$, while E contains the set of edges $e(u, v) \in E$: where gene u regulates gene v. G is stored in a $|V| \times |V|$ adjacency matrix A, where $\forall e(u, v) : A_{uv} = 1$ and $A_{uv} = 0$ otherwise. The undirected adaptation of G, $G'(V, L)$, where $|L| \leq |E|$, is composed of the same vertices, however $l(u, v)$ denotes a bidirectional link between u and v. A $|V| \times |V|$ matrix M stores G', where $\forall l(u, v) : M_{uv} = M_{vu} = 1$ and 0 otherwise. In this regard, M is symmetric and A is not, however A may contain more information about a network. Simplifying to an undirected case is necessary for simpler computations like for example calculating the global clustering coefficient.

6.2   Performance: Packet Transmission

For this study the NS-2 setup considered utilizes topologies where the node with the largest degree is a sink, and every other node is a transmitter. It's imperative that a WSN is capable of re-routing packets to destination nodes post link failures,

which happens often due to line-of-sight obstructions [75]. Although, it is expected that received packets are reduced after eliminating edges, the performance reduction rate should be kept at a minimum for robust topologies. Therefore, we consider the average packet loss in packets per edge knockout. Moreover, a robust network should maintain performances when subject to elevated loss model percentages.

The mean loss in packets for different loss models of 0, 20, 35 and 50% are recorded in Table 7. For each network and loss model, the initial packet receival rate was recorded and the successive rates where also recorded post each edge deletion in a vector fashion. These values are then normalized with respect to the initial transmission rate (current transmission divided by initial transmission), and the differences between successive packet transmission rates are stored in another vector. The means of the resulting vectors for each network and loss models are recorded in Table 7.

Almost every case in *E. coli* performed better than Random as their percentage mean packet losses are less. Furthermore, E.coli shows more robustness towards elevated channel noises. For example, the 300 node case in *E. coli* maintains the same average packet loss throughout the different channel noises, while in the random network cases; the average packet loss almost goes to double of it's initial loss value. For these two observations we conclude that bio-inspired networks are more robust than ER networks in terms of WSN transmission efficiency.

6.3   Connectivity: Average Shortest Path

The average shortest path measures the average number of hops it takes to travel along the shortest paths between pairs of nodes in the network. Many consider the metric to be the most universal measure for assessing information transport efficiency [27]. Others adapt variations of the shortest path for assessing connectivity such as network efficiency (inverse) [76], or the diameter (maximum) [24]. We denote

Table 7. Normalized percentage mean packet transmission loss vs. loss model percentages for Bio- and random modules.

	E. coli				Random			
$|V|$	0%	20%	35%	50%	0%	20%	35%	50%
100	0.128	0.131	0.135	0.137	0.177	0.185	0.204	0.246
200	0.057	0.082	0.085	0.072	0.064	0.1	0.1	0.174
300	0.064	0.064	0.064	0.064	0.064	0.088	0.099	0.123
400	0.04	0.037	0.037	0.04	0.04	0.04	0.04	0.05
500	0.024	0.024	0.025	0.025	0.02	0.028	0.03	0.033
600	0.03	0.031	0.031	0.031	0.035	0.038	0.04	0.041

the average shortest path,

$$\langle d \rangle = \frac{2}{n(n-1)} \sum_{i=1}^{|V|} \sum_{j=1}^{|V|} d_{ij}, \quad d_{ij} \neq \infty, \tag{6.1}$$

where d_{ij} is the shortest path between nodes i and j.

Values of $\langle d \rangle$ are plotted against $|L|$ for different values of $|V|$ in Figure 9. For all the sample networks, $\langle d \rangle$ rises at a higher rate in the random networks as compared to E. coli's networks. While these are similar cases to the nodal knockout experiment mentioned in [24] using the diameter as a performance metric, we observe that this phenomena is repeated for edge knockouts as well. This suggests that robustness of SF networks extend beyond the effects of nodal disconnection from their immediate neighbors since nodal knockout can vary in terms of edge deletions, however single edge deletions are constant for both TRN and ER topologies.

Furthermore, the graph of $|V| = 600$ suggests that the random network goes through a series of disconnections close to the end of the edge deletion process which
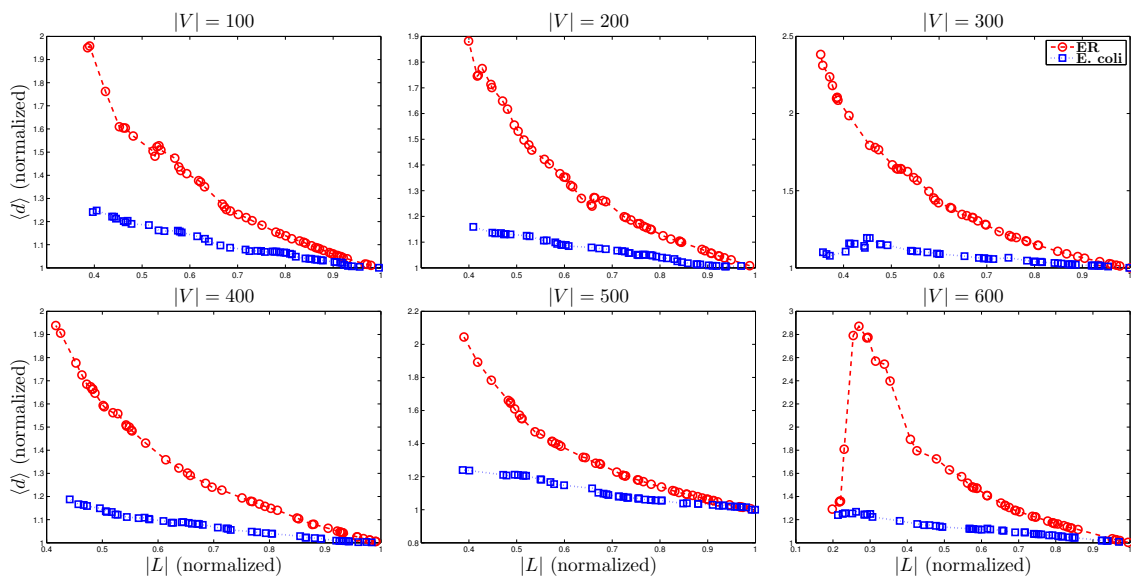
Fig. 9. Edge knockout analysis for *E. coli* subnetworks (blue) vs. randomly generated (red) networks of similar $|V|$ and $|L|$. The plots depicts the performance of the average shortest paths, $\langle d \rangle$ vs. the fraction of the edges remaining in the networks.

explains the sudden drop in $\langle d \rangle$. Essentially, $\langle d \rangle$ gets recorded for the surviving largest component. Due to the above, and because $\langle d \rangle$ is sustained in *E. coli* as compared to random networks, we conclude bio-inspired networks are more robust than random networks in terms of connectivity.

6.4 Tie formations: Global Clustering Coefficient

The global clustering coefficient, C^G, measures the tendency for a network to form internal communities. Scale-free biological topologies exhibit high clustering coefficients and small average shortest paths [77, 78]. Moreover, evidence provided in [28] show that real-world networks such as collaboration and micro organism neural networks create denser ties than ER networks, and are more resilient to edge re-wiring. C^G; computed using the formula [(3×number of triangles)/number of connected triplets],

or computationally,

$$C^G = \sum_{i=1}^{|V|} \sum_{i=1}^{|V|} \sum_{i=1}^{|V|} \frac{M_{ij} \wedge M_{jk} \wedge M_{jk}}{(M_{ij} \wedge M_{jk}) \vee (M_{ij} \wedge M_{ik}) \vee (M_{jk} \wedge M_{ik})}, \qquad (6.2)$$

in an undirected graph. In general, a network with relatively high C^G is considered to have densely connected ties, or in other words, nodes form larger aggregates with their immediate neighbors.

In WSNs, clustering is a prominent area of research as it can be utilized for enhancing their efficiency. For example, experimenting with cluster sizes can be useful for avoiding the "hotspot" problem [79], where some nodes gather more load during operation and die faster. Using clusters, local nodes curate data for cluster heads, which in return can relay packets to base (or sink). Several works have been produced for experimentation with clusters and cluster heads in both, heterogeneous [80, 81] and homogeneous [79, 81, 82, 83] WSN orientations.

Though it is already established that biological networks exhibit internal communities more than that of random networks [28], our analysis show how well can these networks sustain their initial clustering coefficient, herein C_0^G. During edge failures the clustering fluctuates, therefore proves irrelevant to robustness (not like $\langle d \rangle$ where almost each knockout show a vivid increase). For this reason we are more interested in the absolute difference $|C_0^G - C^G|$ to indicate how far did the clustering deviate from the original. Figure 10, presents the mean and the maximum of this quantity. The chart shows that for $|V| \geq 200$, *E. coli* modules are capable of sustaining their clustering better than that of the random networks, However, random networks outperforms *E. coli* for $|V| = 100$. This is common because for smaller SF samples it is sometime the case that subnetworks exhibit exponential traits [84]. In most cases however, we observe that *E. coli*'s clustering is more robust to change.
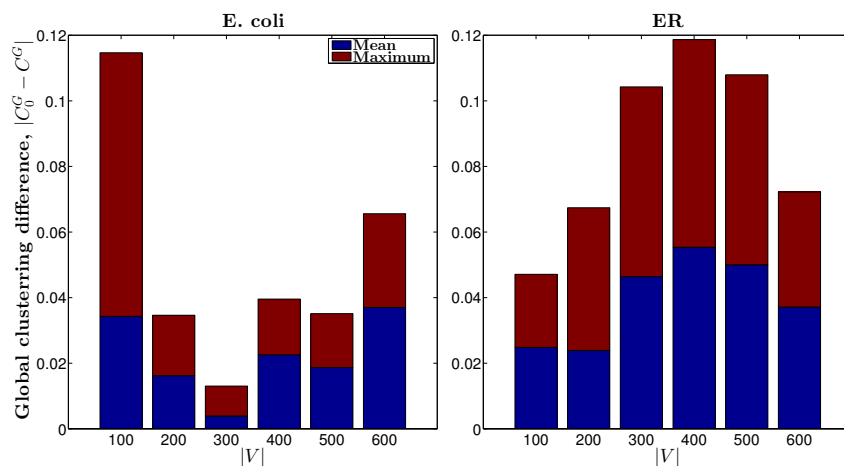
Fig. 10. The deviation of the global clustering coefficient C^G from the original network's C_0^G plotted against different network sizes $|V|$ for *E. coli* subnetworks and generated ER networks of similar dimensions. Define a vector V that holds the quantities $|C_0^G - C^G|$ calculated post every edge elimination. The mean (blue) and max (maroon) for V are given.

CHAPTER 7

CASE STUDY 1: METHODOLOGY FOR SELECTING ROBUSTNESS FEATURES IN WIRELESS SENSOR NETWORKS

In this chapter we describe the methodology for ranking network features that influence packet transmission in WSNs. Ranking is achieved via feature coefficient weights of the SVR prediction hyperplanes. 3590 graphs of $50 \leq |V| \leq 1450$ were considered as separate data points for SVR learning, out of which 1590 networks were TRNs extracted from *E. coli* and Yeast using GeneNetWeaver tool. The remaining 2000 networks were SF derived from 1000 TRN subnetworks using the methodology outlined in sections 6.1.2. **Note**, the SVR is used as basis for ranking features that influence WSN robustness using the normalized weights for linear models methods outlined in [85]. The prediction phase (section 7.2.3) is used for the sole purpose of establishing a reliable model.

7.1 Feature Selection

Our premise is that the graph theoretical properties influence the packet transmission efficiency using the WSN simulation scheme described in Section 5.3. Features are calculated from the sample networks matrix using both directed (G) and their undirected (G') adaptations depending on the feature considered. In addition to features mentioned, $\langle k \rangle$, $\langle d \rangle$ and C^G, all sample features have been determined computationally; their concepts are described in the following.

7.1.1 The Degree Index

Consider vector K, which holds the degrees for nodes in an undirected network. The Degree Index is defined as the ratio of the average nodal degree, K_{avg}, to the highest degree, K_{max} as follows;

$$DI = \frac{K_{\mathrm{avg}}}{K_{\mathrm{max}}}. \tag{7.1}$$

The packet transmission performance depends heavily upon the degree of the sink node selected [86, 73, 21]; specifically hub nodes, having relatively larger degrees than the average, usually exhibit better packet receipt rates when selected as sink nodes. Furthermore, the average packet receipt rates does not monotonically increase with the selection of sinks having higher degrees, however, the highest degree node almost always gives best performance.

Based on this observation, we hypothesize that the Degree Index is a suitable metric that can characterize packet transmission efficiency. As $DI \to 1$, the network becomes more tightly connected, such that the nodes would gain less significance in comparison to one another to act as sinks (i.e., the network does not rely heavily on fewer potential sink nodes). Furthermore, as DI increases for any particular sub-graph, the percentage of packets received decrease making the benefits of sink node selection (having highest degree) largely redundant.

7.1.2 Hub Node Density

The density of the hub nodes measures the territorial occupation of the adjacency matrix grid by the higher degree nodes as a fraction of the total number of edges. We hypothesize that the hub nodes are the hot spot traffic management zones as they have more packets hopping through them. We can determine this quantity as follows:

$$HND = \frac{1}{|L|} \sum_{i}^{n_h} \left[2 \sum_{j=g_i}^{n_g} A_{ij} + \sum_{j=h_i}^{n_h} A_{ij} \right] \tag{7.2}$$

were n_h is the number of hub nodes, n_g is the number of genes (receivers), g_i is the index of nodes outside the set of hub nodes, and h_i is the index of the hub nodes.

7.1.3 Sink coverage

The sink coverage measures the percentage of nodes that have a direct link to the sink node,

$$SC = \frac{K_{max}}{|V|} \tag{7.3}$$

When node a tries to send packets to node b through node c lying along the path P_{ab}, packets are queued at c before they get forwarded to b, which in return can be dropped if packets exceed the queue length at c. However, if c did not exist in P_{ab}, packets will not be discarded due to multi-hops; SC is a feature that captures such scenarios.

7.1.4 Network Density

The Network Density is traditionally a measure of the territorial occupation of a communication network, calculated as the ratio of the sum of the edge lengths to the surface area occupied by the network grid [87]. Since our simulations do not account for edge weights, we simplistically consider the nodes to be equidistant, having unit lengths of one. Our measure accounts for how many links occupy the M grid and determine the Network Density as follows:

$$ND = \frac{2|L|}{|V|(|V| - 1)} \tag{7.4}$$

where, ND ranges between $(n-1)/n^2$ for a Star configuration (with a single hub, provided every other node is solely connected to the hub with one edge) or a tree, and $1 - (1/n)$ for a fully connected sub-graph excluding the self-loops.

Previous work [86] suggests that tightly connected graphs does not necessarily have higher packet receipt rates. For example, complete networks generate more interference than star topologies. The performance of the networks having the same ND are comparable because the properties of the sink nodes selected after the randomizations are still preserved in the sub-graphs. Moreover, trees (low ND), do not necessarily exhibit high packet transmission. For example, a bus topology subjects packets to multi-hops, which in return increases the chances of packet drops.

7.1.5   The Motif Densities

The repetitive "motif" substructures have significant contributions to WSN performance and functionality (as we have shown earlier in [22, 21]), as well as affecting robustness in biological networks [18]. Although various types of motifs have been identified previously in biological networks, we consider the "most significant" ones for our model: the Feed-Forward Loop (FFL) and the Bi-fan [40]. These two motifs significantly outnumber similar sub-structures when mined from the TRN of *E. coli* (or Yeast) in comparison to other randomized networks, and hence they are believed to have a significance in biological networks in general.

In addition to statistical significance, motifs are known to exhibit peculiar functions. For example, FFLs are notable for their ability to deliver vital functions such as delay responses times in genes, irreversible speed up, or create pulses [42]. BFs are the building blocks of dense overlapping regulons, which are considered to be the backbone for gene regulatory networks, sharing global functions such as: stress response, nutrient metabolism, or bio-synthesis of key classes of cellular components [54].

59
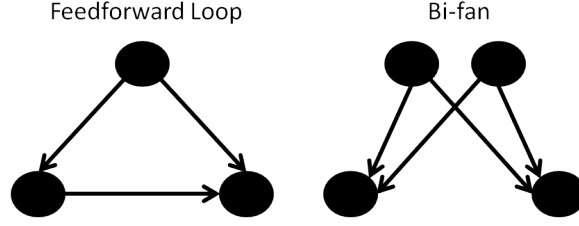
Fig. 11. Structures of significant motifs; the FFL and BF.

Figure 11 shows the FFL and bi-fans structures as mentioned above, for which we hypothesize that their relative abundance in the network should make an important feature to consider in our regression model. We propose a way of identifying a networks motif density; their counts divided by their possible occurrences. In this regard we can define both the FFL and bi-fan densities as,

$$
FFLD = \frac{1}{\binom{|V|}{3}} \sum_{i=1}^{|V|} \sum_{j=1}^{|V|} \sum_{k=1}^{|V|} \big[(A_{ij} \wedge A_{jk} \wedge A_{ik})
$$
$$
\wedge \neg (A_{ji} \wedge A_{kj} \wedge A_{ik}) \big], \tag{7.5}
$$

and,

$$
BFD = \frac{1}{\binom{|V|}{4}} \sum_{i=1}^{|V|} \sum_{j=1}^{|V|} \sum_{k=1}^{|V|} \sum_{l=1}^{|V|} \big[(A_{ij} \wedge A_{ik} \wedge A_{lj} \wedge A_{lk})
$$
$$
\wedge \neg (A_{ji} \wedge A_{ki} \wedge A_{jl} \wedge A_{kl}) \big], \tag{7.6}
$$

in that order.

7.1.6 Average Closeness Centrality

The closeness centrality c_i measures the relative closeness of node i to every other node in the network. A close node is capable of delivering information quicker

to other nodes [88]. Central nodes in biological networks are crucial because they play the roles of 'organizational hubs' [89]. The closeness centrality of a node is computed by,

$$c_i = \frac{n-1}{\sum_{i=1}^{|V|} d_{ij}}, \quad d_{ij} \neq \infty, \tag{7.7}$$

having values $(0, 1]$, with 0 meaning i is disconnected from the network, and 1 meaning node i has a direct link to every node.

An increase in c_i signifies an increase in communicative efficiency, however, it does not guarantee robustness to random attacks. For example, if all nodes have $c_i = 1$, i.e a complete graph, a random attack on any node would cost the network $|V| - 1$ links. For the entire network we compute the average closeness as,

$$\langle c \rangle = \frac{n-1}{\sum_{i=1}^{|V|} d_{ij}}, \quad d_{ij} \neq \infty. \tag{7.8}$$

7.1.7 Average Local Clustering Coefficient

The average local clustering coefficient, $\langle C^L \rangle$, formally defined as

$$\langle C^L \rangle = \frac{1}{|V|} \sum_{i=1}^{|V|} \frac{\sum_{j=1}^{|V|} A_{ij}}{S_i(S_i - 1)}, \tag{7.9}$$

where S is a vector holding the number of surrounding neighbors for each vertex i. The metric measures the number of edges connecting the nodes as a fraction of the number of possible edges that could exist in their local communities. At the individual node level, this metric can be used to quantify relative nodal participation in embedded clusters.

7.2 Regression Model

To summarize, we consider 11 features: $\langle k \rangle$, DI, SC, ND, HND, $\langle c \rangle$, $\langle C^L \rangle$, C^G, $\langle d \rangle$, $FFLD$ and BFD. In the next subsections we discuss the next steps towards generating the regressor and ranking our features. Those steps include **pre-processing** of the features generated, **training** and **validating** the SVR.

7.2.1 Pre-processing

In order to check for feature relevancy, pair-wise correlations were considered. Pearson product-moment correlation coefficient is used to measure trend similarities (or correlations) between any two feature vectors $F1$ and $F2$ as follows,

$$\rho(F1, F2) = \frac{1}{n-1} \sum_{i=1}^{n} \left(\frac{F1_i - \mu_{F1}}{\sigma_{F1}} \right) \left(\frac{F2_i - \mu_{F2}}{\sigma_{F2}} \right), \tag{7.10}$$

where n is the size of F, $\mu_{...}$ the mean and $\sigma_{...}$ the standard deviation. Features are considered highly correlated if $\rho(F1, F2) \geq 0.94$ or $\rho(F1, F2) \leq -0.94$. Because correlated features add computational overheads to SVR learning and performance, one of each correlated feature vector pairs is eliminated. For example, the average in-, out-degree and the diameter where eliminated because their effects are compensated for via the average cumulative degree and shortest path. The remaining features are correlated with ρ values shown in the grey-scale map of Figure 12 where $\rho(F1, F2)$ values are given by,

$$R = \begin{pmatrix} 1 & \rho(F1, F2) \\ \rho(F2, F1) & 1 \end{pmatrix}. \tag{7.11}$$

Additional, eliminations are considered for outliers. Data points having packet transmission values that fall outside the range of mean $\pm [3 \times$ standard deviation$]$ were

$\langle k \rangle$	DI	SC	ND	HND	$\langle c \rangle$	$\langle C^L \rangle$	C^G	$\langle d \rangle$	FFLD	BFD
1.00	0.20	-0.05	-0.06	-0.31	0.32	-0.74	-0.28	0.18	0.02	0.19
0.20	1.00	-0.81	0.07	-0.65	0.91	-0.78	0.26	0.90	-0.19	-0.06
-0.05	-0.81	1.00	0.31	0.65	-0.78	0.54	-0.15	-0.82	0.45	0.30
-0.06	0.07	0.31	1.00	0.20	-0.13	-0.00	0.28	-0.17	0.71	0.60
-0.31	-0.65	0.65	0.20	1.00	-0.64	0.63	-0.01	-0.66	0.32	0.14
0.32	0.91	-0.78	-0.13	-0.64	1.00	-0.81	0.17	0.92	-0.26	-0.10
-0.74	-0.78	0.54	-0.00	0.63	-0.81	1.00	0.01	-0.74	0.11	-0.08
-0.28	0.26	-0.15	0.28	-0.01	0.17	0.01	1.00	0.23	0.22	0.03
0.18	0.90	-0.82	-0.17	-0.66	0.92	-0.74	0.23	1.00	-0.31	-0.14
0.02	-0.19	0.45	0.71	0.32	-0.26	0.11	0.22	-0.31	1.00	0.60
0.19	-0.06	0.30	0.60	0.14	-0.10	-0.08	0.03	-0.14	0.60	1.00

Fig. 12. Grey-scale showing correlations of feature pairs and their values. Consider Equation 7.10, with $F1$ and $F2$ values for features 1 and 2; $-1.0 \leq \rho(F1, F2) \leq 1.0$. Numbers in the color map are due to R, where 1.0 correspond to strong positive correlation (black), -1.0 strong negative correlation (white) and 0 for no correlation (gray).

removed. Furthermore, features are scaled by subtracting them from their means and dividing them by their standard deviations to enhance the learning process.

7.2.2 Training

After generating the packet transmission rates, the values of the loss models were stored in a y vector of $s \times 1$ dimension for s subgraphs. If X is a $s \times 11$ matrix, then we can denote $X^{(s)}$ as the sth row of X, which holds the values of the training data (or topological features) of the sth training sample (or network). X and y are used as input parameters for generating the model's weights vector (w) and the bias (b) using LibSVM[90]. Once the model is generated, any approximation O_i can be predicted with any input X_i, using $O_i = X_i w + b$, and the influences of the different features on the models output are ranked by analyzing the magnitudes of their corresponding

weights. Other parameters include setting the svm type to an epsilon-SVR with a linear kernel. The training was repeated several times with varying values of the hyper parameters; C (the quadratic optimization constraint) and ϵ ($\pm$ error range). The model produced by the parameters which yields the minimum cross-validation error was used as basis for feature ranking and selection. Furthermore, as we observe that the values recorded from the different loss models are exactly the same for a queue length of 1, there was no need to account for any difference in the loss models. The model set-up and results with varying queue lengths in the NS-2 simulations will be discussed elsewhere.

7.2.3 Validation

The SVR is validated using a 10 k-fold cross validation error, for which training is assumed 10 times at random on 90% of the sample space and prediction is performed on the remaining 10% during each iteration. The error percentage of the difference between the desired average packet receipt rates and the predicted values is calculated by,

$$error = 100 \times \frac{|y - y_{apx}|}{|y|} \tag{7.12}$$

were y_{apx} represents the model's approximated (predicted) values. The error is accumulated after each iteration and then averaged across the 10 folds to give us the cross validation error. Using the same parameters, training is performed across the whole sample space and prediction was again performed on the whole data set to calculate the data set error (or total error). The SVR errors for the different loss models are given in Table 8.

64

Table 8. Support vector regression accuracy for the different data sets and packet transmission loss model.

Loss model(%)	TRN		SF		ER		Combined	
	Cross(%)	Total(%)	Cross(%)	Total(%)	Cross(%)	Total(%)	Cross(%)	Total(%)
0	1.5	0.1	2.0	0.2	1.2	0.1	1.9	0.5
20	1.9	0.7	2.2	1.0	1.4	0.3	2.1	1.1
35	2.3	0.1	2.3	1.1	1.7	0.8	2.5	4.7
50	3.0	4.7	2.5	0.2	2.2	4.7	3.2	8.8

7.3 Feature Ranking in Wireless Sensor Networks

The relative importance of the network features are identified from w, which holds the coefficients of the regressor's hyper-plane. Because these coefficient values differ according to feature space and type, it is relevant to rank them relative to the most influential, $|w|_{max}$. In this regard, a feature rank is considered based on the magnitude of,

$$rank_i = \frac{w_i}{|w|_{max}}, \qquad (7.13)$$

where i is the index of the coefficient (feature weight). Furthermore a coefficient can be either positive or negative, meaning it can raise or lower the linear hyper-plane (SVR).

Feature influence is depicted in Figure 13. In general, We have observed (results not shown) that the relative importance of the features used in our model can be ranked in descending order as follows: DI, $\langle d \rangle$, ND, C^G, HND, SC, $\langle k \rangle$, $\langle C^L \rangle$, $\langle c \rangle$, $FFLD$, and BFD. DI has maximum influence, while BFD has the least. Out of the 11 features; $\langle k \rangle$, SC, HND, and $\langle C^L \rangle$ raise the SVR, the remaining cause it to descend.

65
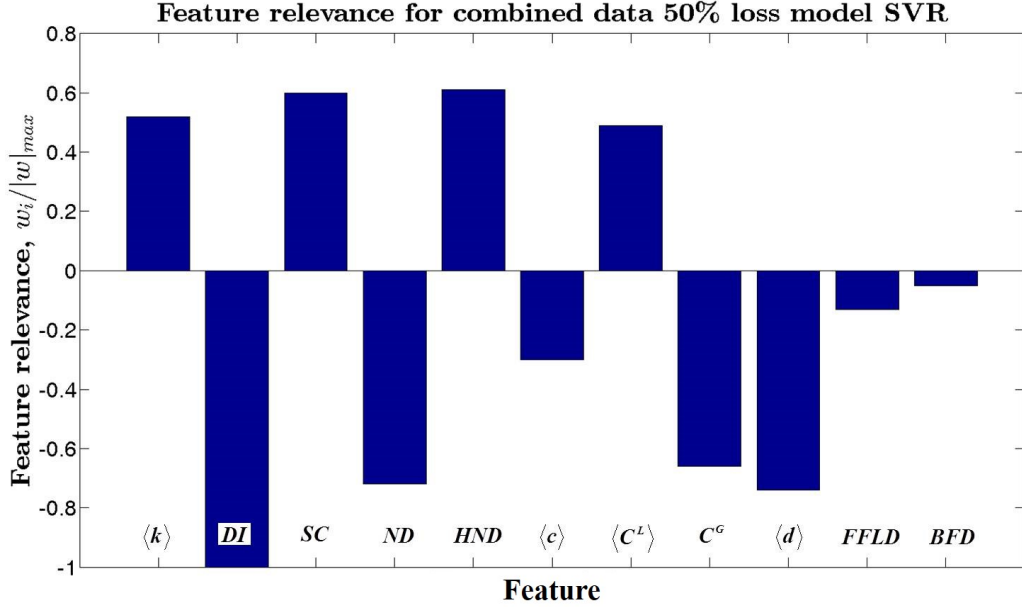
Fig. 13. Consider the maximum of the weight coefficients w_{max}, which is DI in this case. Feature relevance to the maximum is the ratio of it's coefficient w_i to the magnitude of the maximum $|w|_{max}$.

Several observations can be deduced from Figure 13 via pair wise comparisons. In regards to the degree distribution, the fact that $\langle k \rangle$ is relatively medium positive, and DI is at maximum negative, suggests that although the degree should be balanced, the sink node should have a far larger degree than every other node. The value of the average degree should be mainly compensated for by the maximum degree. In agreement with these two, SC is at a relatively high positive suggesting that the sink should have a direct link to large number of nodes, while the rest of the links should be distributed among the nodes themselves.

Regarding the network's tightness of connectivity, we observe that a less dense network is better for transmission, as explained due to the negative-ness of ND, $FFLD$ and BFD. In contrast with these three, HND is shown to be relatively high positive, which suggests that the network should be dense among the hub nodes

66

and loose among the rest. This can be explained using hierarchal cascades of star formations, where the network enclosed by the hub nodes is eventually dense as compared to that enclosed by the others (like in Figures 8 (a) and (b)).

Furthermore, C^G is shown to be negative while $\langle C^L \rangle$ is shown to be positive, meaning that "although there should be a decrease in the number of cliques, there should be an increase at which neighbors of nodes connect to one another". While the first part of the statement suggests that the network should be tightly connected, the second part suggests that these cliques should not overwhelm the network. Combining both statements, suggests that there should be a distributed local star formations across the network.

For connectivity, in terms of link hops: the negative $\langle d \rangle$ and $\langle c \rangle$ show that it is best for nodes to be as far as possible from one another. However, SC suggests that these nodes should be closer to the sink. In order to achieve both criteria, the network should be sparse and densely connected among the hub node regions.

7.4   Example: Testing Attachment Kernels

From Figure 13, the two most influential features affecting the packet transmissions are; DI and SC. The negative DI suggests that $K_{\max}$ should be much higher than K_{avg}, which is a characteristic of scale-free topologies. To accomplish scale-free random growth, we use BA as part of the attachment kernel. The probability of a incoming node attaching to an existing node a within the substrate network is given by,

$$\text{kernel(BA)} = \frac{K_a}{\sum_{i=1}^{n_{\text{sub}}} K_i} \tag{7.14}$$

, where $K_{..}$ is the degree of a node and n_{sub} the size of the substrate network.

On the other hand, a positive SC suggest that the sink should have relatively
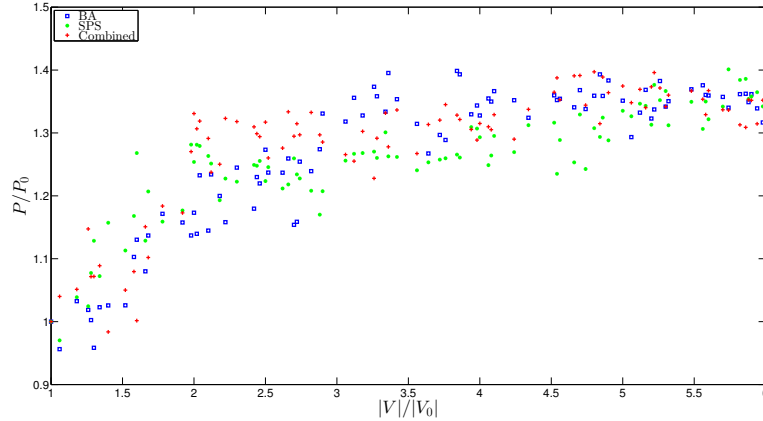
Fig. 14. Robustness comparisons for packet transmission. Scatter plot of the normalized packet transmission efficiency vs. the normalized size of the network. Here, P represents the packet receipt post nodal additions, and P_0 the initial performance.

high number of connections. Based on this observation, we hypothesize that the shortest path to the sink should be a criteria for attachment. In this case, the kernel should have a reverse effect, where a higher probability of attachment is given to nodes that are closer to the sink and lower probability is assigned to distant ones- in other words, using closeness centrality. The shortest path to sink kernel is given by,

$$\text{kernel(SPS)} = \frac{1/(d(s)_a + 1)}{\sum_{i=1}^{n_{\text{sub}}} 1/(d(s)_i + 1)} \tag{7.15}$$

, where $d(s)_{..}$ is the shortest path from a given node to the sink, i.e. the highest degree node within the substrate.

For simplicity, combining both kernel effects during the growth process, can be achieved by assigning both probabilities equal weights. Though extensive experimenting is required for acquiring the necessary features and their assigned weights, this example is intended to give the reader an initial idea of how the kernels can be uti-

68
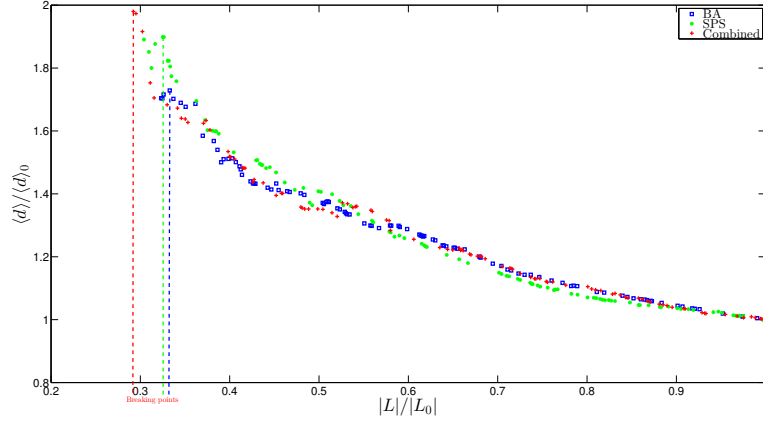
Fig. 15. Robustness comparisons for connectivity. Edge knockout analysis for randomly grown networks using BA and the proposed attachment kernels; SPS and combined. The plot depicts the performance of the normalized average shortest path, $\langle d \rangle / \langle d \rangle_0$, vs. the normalized number of links in the network, $\langle L \rangle / \langle L \rangle_0$.
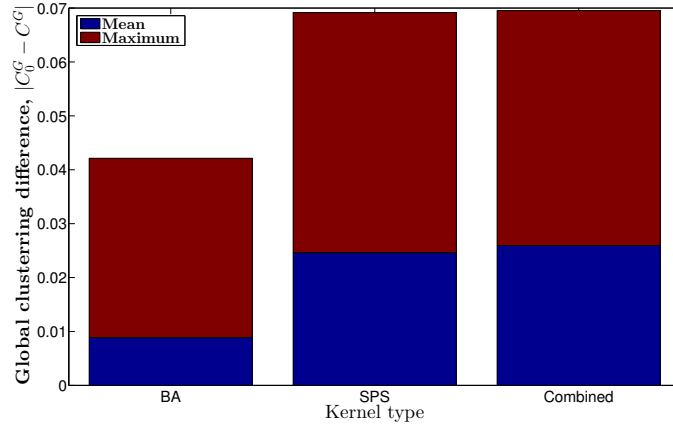
Fig. 16. Robustness comparisons for modularity. The deviation of the global clustering coefficient C^G from the original network's C_0^G plotted against different network sizes $|V|$ for BA, SPS and combined kernel grown networks.

lized. For more information on the subject, please refer to our previous works in [31, 55]. In this regard, we denote the combined kernel as,

$$\text{kernel(Combined)} = \frac{1}{2}\left[\frac{K_a}{\sum\limits_{i=1}^{n_{\text{sub}}} K_i} + \frac{1/(d(s)_a + 1)}{\sum\limits_{i=1}^{n_{\text{sub}}} 1/(d(s)_i + 1)}\right]. \qquad (7.16)$$

Results for packet transmission, connectivity and modularity are depicted in Figures 14,15 and 16 in that order. In this experiment an initial *E. coli* subnetwork of 50 nodes is grown using all three kernels.

For packet transmission, the improvement for all three approaches are comparable, however the average improvement for SPS and combined are higher than that of BA. Moreover, Figure 15 shows that all three approaches have comparable increases in $\langle d \rangle$ per edge knockout, however the combined and SPS approaches sustain a singly connected component past the breaking point of that of BA. Additionally, BA outperforms the other kernels in terms of sustaining the clustering coefficient deviation as depicted in Figure 16.

CHAPTER 8

CASE STUDY 2 - METHODOLOGY FOR SELECTING ROBUSTNESS FEATURES IN TRANSCRIPTIONAL REGULATORY NETWORKS

In this chapter we describe the methodology for pinpointing the structural features that affect TRN robustness. Sample TRN and exponential subnetworks of similar dimensions ($|V|$ and $|E|$) are considered for optimization using ILP which returns different subnetworks with similar degree distributions but smaller average shortest paths (optimized network). Optimal structures will have variations in feature magnitudes. Features which show significant change are considered as candidates for attachment kernel criteria. Moreover, an Integer Linear Program is given, which serves as a solution to the problem stated in the motivation of section 2.2.2, which answers the question of whether or not the TRN of *E. coli*; is an optimal network. And if it is not, how can we make it so?

8.1 Methods

Here we introduce our performance metric, the average shortest path and other related network features. Their utilization in determining an optimal network using integer linear programming is explained in the following subsections.

8.1.1 Metrics and criteria for optimal topology

We adopt a version of the average shortest path, $\langle d \rangle$, as a measure for network robustness, which has been widely used as a robustness metric for many complex systems [1]. In this dissertation, $\langle d \rangle$ is defined as the distance, in number of "hops"
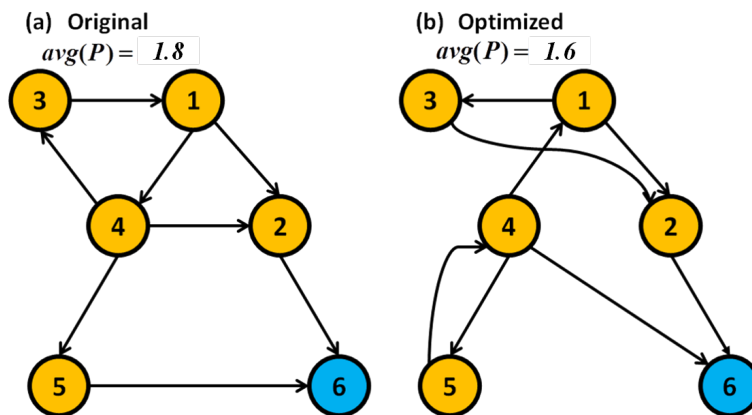
Fig. 17. (a) An exemplary random network and (b) it's post-optimization output from the integer linear programming algorithm. The in- and out-degrees of each node in both are identical, and the existence of routes from the TFs (nodes 1-5) to their target regulated gene (node 6) have been preserved. In this example, the optimized network (b) exhibits a smaller average shortest path, $\langle d \rangle$, than the exemplary network (a).

(i.e., single movements between adjacent nodes) required to connect a transcription-factor node to a regulated gene, and averaged over all such pairs in a network. We restrict our analyses to the connected paths from transcription factors to genes. With these conditions, $\langle d \rangle$ can be expressed by the equation:

$$\langle d \rangle = \frac{1}{p} \sum_{i}^{n_t} \sum_{j}^{n_g} d_{ij}, \ \text{with} \ d_{ij} < \infty, \tag{8.1}$$

wherein d_{ij} denotes the shortest path between nodes i and j, p the number of connected TF-gene pairs, n_t the number of TFs, and n_g the number of genes in the network. The type of the regulatory interaction, either up- or down-regulating, was not considered in Eq. (8.1).

We note that $\langle d \rangle$ is slightly different from other definitions of the average shortest path, wherein p has been taken as the number of connected pairs without regard for their type. However, the p of 8.1 is distinct from these by its consideration of biology,

that it reflects consideration of only a subset of the total number of possible paths by eliminating any contributions to $\langle d \rangle$ between regulated genes. For a detailed derivation, please refer to Appendix A.

An optimal topology is one wherein a metric of the topology is at an extremum. As explained in the next section, this criteria is the average shortest path, $\langle d \rangle$, for a given network $G(V, E)$, wherein V is the set of all networked nodes (TFs and regulated genes) and E denotes the set of edges connecting them. An output of the integer linear programming algorithm (explained below) is an "optimized" version of G, $G_0(V, E_0)$, which hosts an identical set of vertices, number of edges, $|E| = |E_0|$, identical degree distribution as G, and the same number (and type) of feed-forward loop transcriptional network motifs, but with potentially different average shortest path, $\langle d_0 \rangle$. To determine whether the topology of G is optimal, we compare $\langle d \rangle$ to that of its optimized counterpart, $\langle d_0 \rangle$, using the following metric, δ:

$$\delta = \frac{|\langle d \rangle - \langle d_0 \rangle|}{\langle d \rangle}, \tag{8.2}$$

Therefore, $\delta = 0$ corresponds to the result that G exhibits a perfectly optimal topology.

8.1.2 Optimizing Topology with Integer Linear Programming

Finding a network with optimal topology is a challenging problem. On the one hand, a brute-force search approach is intractable due to an exponentially large search space on the order of $\mathcal{O}(2^{|V|^2})$-many different graphs. On the other hand, meta-heuristic methods, such as Simulated Annealing [91] and Genetic Programming [92, 16], cannot guarantee an optimal solution. To address this problem, we propose a new approach to finding an optimal graph topologies for small and moderate size networks based on integer linear programming (ILP).

Many graph-theoretical problems are solvable using integer linear programming [93], such as the shortest path, vertex coverage, maximum flow, and minimum cost-flow problems. This is possible, because these problems can be expressed in terms of linear relationships which together form a polytope enclosed by their intersections. From this polytope, it is possible to identify an extremum of a cost function. Moreover, ILP can be useful to identify cases wherein solutions are not feasible with other methods, and their implementation can be facilitated using freely available academic software, such as IBM ILOG CPLEX optimizer [94].

We consider a linear program which identifies a new graph G_0 with minimum $\langle d_0 \rangle$ based on an input G, subject to the following constraints:

1. Connectivity between a TF and a regulated gene "target' must be preserved from G, which ensures invariance of the overall paths, but not necessarily the path-lengths;

2. Degree distributions between G and G_0 are identical;

3. The number of feed-forward loop transcriptional network motifs does not exceed that of G, despite any variance in topology.

In particular, we hold the number of feed-forward loops below a threshold during the optimization process, because we have previously identified that path-length metrics may be significantly affected if feed-forward loop transcriptional motifs are "added" to the network topology [95]. The detailed equations for the linear constraints are discussed below.

An example of an "optimized" 6-node network is given in Figure 17. The optimal solution (Fig. 17(b)) preserves connections between TF-gene pairs, in- and out-degrees for each node, and the number of feed-forward loop transcriptional motifs, but exhibits a $\langle d \rangle$ smaller by approximately 11%.
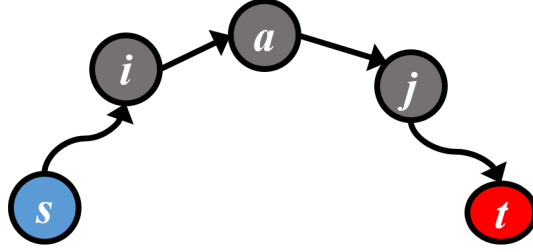
Fig. 18. A representation of the shortest path between nodes s and t. Since edge $i \rightarrow j$ does not exist, then $i \rightarrow j$ is not included in the shortest path from s to t, and indicator $Z_{ij}^l = 0$. However, the shortest path goes through $i \rightarrow a$ and $a \rightarrow j$, hence $Z_{ia}^l = Z_{aj}^l = 1$. Here, l references the connected pairs s and t, where $l = 1..f$ (number of connected pairs, $|P|$).

8.2   Integer Linear Programming Formulation

Given a TRN network abstracted as a directed graph $G = (V, E)$ where $V = \{v_1, v_2, ..., v_n\}$ and $E = \{e_1, e_2, ..., e_m\}$, where $n = |V|$ and $m = |E|$. Let d_i^{in} and d_i^{out} be the in- and out-degree of node i for $i = 1..n$. Let $N_G = \{v_i \mid d_i^{\text{out}} = 0\}$ be the set of gene nodes, $N_{\text{TF}} \equiv V - N_G$ be the set of transcription factor (TF) nodes.

Define $P = \{(s_i, t_i) \mid s_i \in N_{\text{TF}}, t_i \in N_G$, provided there exists a path from s_i to $t_i\}$. Denote by $d(u, v)$ the minimum distance, i.e the length of the shortest path from u to v. Therefore, the average shortest path

$$\text{avg}(P) = \frac{1}{|P|} \sum_{i=1}^{|P|} d(s_i, t_i) \tag{8.3}$$

among the pairs indexed by i. We want to verify that: "Among the networks with same in- and out-degree distributions of similar or less FFL abundances, the TRN network's $\text{avg}(P)$ is among the smallest".

Problem definition: the Optimal Topology Design (ODT) problem is defined as, "given a network G of an in-degree distribution $D^{\text{in}} = \{d_1^{\text{in}}, d_2^{\text{in}}, ..., d_n^{\text{in}}\}$, $D^{\text{out}} = \{d_1^{\text{out}}, d_2^{\text{out}}, ..., d_n^{\text{out}}\}$, an embedded motif abundance $M(G)$, and a an average shortest

path avg(P) among pairs of P, obtain an optimal network G_0, such that $D_0^{\text{in}} = D^{\text{in}}$, $D_0^{\text{out}} = D^{\text{out}}$, $M(G_0) \leq M(G)$, and avg(P) is a minimum."

ILP formulation: consider $P = \{(s_1, t_1), (s_2, t_2), ..., (s_f, t_f)\}$ where $f = |P|$. Denote objective term $Z^l = \{0, 1\}$ for some index l. The ILP is formulated as follows; $\forall i = 1..n$, $\forall j = 1..n$ and $\forall l = 1..f$ consider,

$$\textbf{Min.} \quad \sum_{l=1}^{f}\sum_{i \neq j}^{n} Z_{ij}^{l} \tag{O1}$$

$$\textbf{S.t.} \quad Z_{ij}^{l} \leq x_{ij} \tag{C1}$$

$$\sum_{i=1}^{n} Z_{ij}^{l} - \sum_{j=1}^{n} Z_{ji}^{l} = \begin{cases} 1 & \text{if } i = s_l \\ -1 & \text{if } i = t_l \\ 0 & \text{otherwise} \end{cases} \tag{C2}$$

$$x_{ii} = 0 \qquad \forall i = 1..n \tag{C3}$$

$$\sum_{j=1}^{n} x_{ij} = d_i^{\text{out}} \qquad \forall i = 1..n \tag{C4}$$

$$\sum_{j=1}^{n} x_{ji} = d_i^{\text{in}} \qquad \forall i = 1..n \tag{C5}$$

$$\sum_{i \neq j \neq k} y_{ijk} \leq M(G) \tag{C6}$$

$$x_{ij} + x_{jk} + x_{ik} \leq 2 + y_{ijk} \tag{C7}$$

$$\left.\begin{aligned} x_{ij} &\geq y_{ijk} \\ x_{jk} &\geq y_{ijk} \\ x_{ik} &\geq y_{ijk} \end{aligned}\right\} \quad \forall i \neq j \neq k \tag{C8}$$

$$x_{ij}, \, y_{ijk}, \, Z_{ij}^{l} \in \{0,1\} \tag{C9}$$

Z_{ij}^l (O1-C2) is an indicator variable suggesting that edge(i, j) is part of the shortest path between a source node s and a destination node t; defined as follows,

$$Z_{ij}^l = \begin{cases} 1 & \text{edge } i \rightarrow j \text{ is within a shortest path} \\ 0 & \text{otherwise} \end{cases} . \tag{8.4}$$

The objective to minimize the expression $\sum_{i \neq j}^n Z_{ij}^l$ of (O1), together with constraint (C2) and excluding the indicator l, illustrate the standard linear programming formula for capturing the shortest path from a given s to an existing t [96]. Based on Eq. 8.3, we would like the ILP to capture the average shortest path between pairs of P. This is achieved in (O1) using the summation $\sum_{l=1}^f ..$ which guarantees ILP operation for all pairs(s, t).

For the **degree distribution**: let $x_{ij} \in \{0, 1\}$ be the indicator

$$x_{ij} = \begin{cases} 1 & \text{if there is an edge from } i \rightarrow j \\ 0 & \text{otherwise} \end{cases} . \tag{8.5}$$

Because self loops do not constitute significance in this study, the ILP avoids creating them using constraint (C3), which sets $x = 0$ for all node indicators. The out- and in- degree distributions are maintained using constraints (C4) and (C5) in this order.

Furthermore, we would like to guarantee that if nodes i and j are part of the shortest path between s and t, and if edge does not exist between nodes i and j and, Z_{ij} will not be included within the shortest path (Figure 18). The above is explained in lemma 8.2.1.

Lemma 8.2.1. *If there is no edge(i, j), then Z_{ij} can not be on the shortest path from i to j.*

Proof. If $x_{ij} = 1$, then by constraint (C1) $Z_{ij}^1 \leq 1$.
If $x_{ij} = 0$, then by constraint (C1) $Z_{ij}^1 \leq \infty$.

78
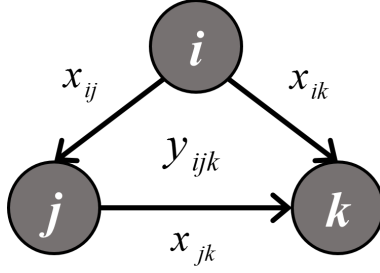
Fig. 19. A representation of the Feed-forward loop motif. Indicators x_{ij}, x_{jk} and x_{ik} designate the existence of edges $i \to j$, $j \to k$ and $i \to k$ in this order. Y_{ijk} designate a single FFL formed due to the intersection of the edges.

□

$$y_{ijk} = \begin{cases} 1 & \text{if } (i, j, k) \text{ is an FFL} \\ 0 & \text{otherwise} \end{cases} \quad . \tag{8.6}$$

The condition FFL count $\leq M(G)$ can be formulated using constraints (C6-C8). (C6) guarantees that the summation of all FFL indicators should not exceed that of the original FFL count $M(G)$. Furthermore, constraints ($C7$ and $C8$) guarantee an FFL structure to be an intersection between the edges $i \to j$, $j \to k$ and $i \to k$ as shown in Figure 19. The above can be proved using lemma 8.2.2.

Lemma 8.2.2. $y_{ijk} = 1$ *if and only if* $x_{ij} = x_{jk} = x_{ik} = 1$

Proof. If $y_{ijk} = 1$, then $x_{ij} + x_{jk} + x_{ik} = 3$, and by ($C8$) and ($C9$) $x_{ij} = x_{jk} = x_{ik} = 1$. If $x_{ij} = x_{jk} = x_{ik} = 1$, then by ($C7$) $y_{ijk} = 1$

□

Theorem 8.2.3. *The ILP returns an optimal solution for the OTD problem.*

Furthermore, the ILP collectively comprised of $5[n(n-1)(n-2)]+2n(n-1)+3n+1$ constraints, where n is the size of the network.
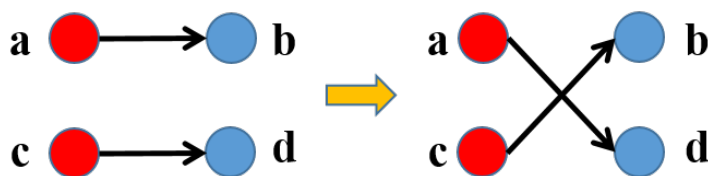
Fig. 20. A representation of a possible switch; $a \to b$ with $c \to d$. All nodes retain their original degrees post switching.

8.3 Algorithms

In this section, we discuss two heuristic methods for approximating the output of the ILP: a (1) Local search, and a (2) Simulated annealing approach. Our goal is to be able to computationally optimize a given G of sizes $n > 100$ within reasonable time. Solutions produced exhibit δ values not too far off from that of optimal $G_0(V, E_0)$ structures assembled using the ILP.

8.3.1 Local Search Approach

Algorithm 1 uses an input G graph and P pairs to produce an approximated optimized output graph G_0. Instruction 2 uses exhaustive search among pairs of edges within the graph to identify edge pair candidates for a 'good switch'. In addition to having a possible switch that preserves the degree distribution, as shown in Figure 8.3.1, a good switch must reduce or maintain the $\text{avg}(P)$. In this effect a good switch is characterized by;

- having the degree distributions; in- and out-degrees, of the nodes involved are preserved,

- $\text{avg}(P)$ decreases or remains the same,

- the motif abundance remains less than or equal to $M(G)$ after the switching.

If a good switch is determined, the edges will be swapped as illustrated in instructions 2 and 3. In order to determine the eligibility of a switch, $avg(P)$ is calculated post switching an edge pair. In cases where $avg(P)$ increases, the edges will be reverted to their original positions, and the output of the condition in instruction 2 will be false. Furthermore, the run-time for $\mathcal{O}[|V|^2|E|^2(|V|^{3/2} + N_{tf}(|E| + V log|V|))]$, where N_{tf} is the number of transcription factors.

Input: Original network, G; P

Output: Optimized network, G_0

  1: **for all** pairs of edges (e_1, e_2) **do**

  2:    **if** if (e_1, e_2) is a good switch **then**

  3:        switch e_1 and e_2

  4:    **end if**

  5: **end for**

Algorithm 1: Optimizing a directed network using Local Search.

8.3.2 Simulated Annealing Approach

An improvement over the local search is considered using simulated annealing-given in Algorithm 2. The algorithm uses an input G graph, P pairs, small scalar ϵ and the number of possible switches to produce an approximation G_0- generally $1 \times 10^{-5} \leq \epsilon \leq 1 \times 10^{-7}$. Other factors affecting the performance of the algorithm like the temperature T, which can allow instances of a "bad switch" as illustrated in instruction 7. A bad switch will cause an increase in $avg(P)$ after the edge swap. Therefore a bad switch is characterized by;

- having the degree distributions; in- and out-degrees, of the nodes involved are preserved,

- avg(P) increases,

- the motif abundance remains less than or equal to $M(G)$ after the switching.

If a bad switch is determined (instructions 6-8), the swap will be considered if a random variable $r = [0, 1]$ satisfies the condition $r \leq \alpha$. Moreover, the algorithm run-time is $\mathcal{O}[T|V|^2|E|^2(|V|^{3/2} + N_{tf}(|E| + V log|V|))]$.

Input: Original network, G; $\epsilon > 0$; P; possible switches

Output: Optimized network, G_0

1: $T \leftarrow 2/\text{possible switches}$

2: **repeat**

3:     **for all** pairs of edges (e_1, e_2) **do**

4:         **if** if (e_1, e_2) is a good switch **then**

5:             switch e_1 and e_2

6:         **else**

7:             switch e_1 and e_2

8:             switch e_1 and e_2 with a probability, $\alpha = T\dfrac{\text{avg}(P)_{\text{before switch}}}{\text{avg}(P)_{\text{after switch}}}$

9:         **end if**

10:     **end for**

11: **until**

Algorithm 2: Optimizing a directed network using Simulated Annealing.

Figure 8.3.2 illustrates results of the integer linear programming (ILP), local
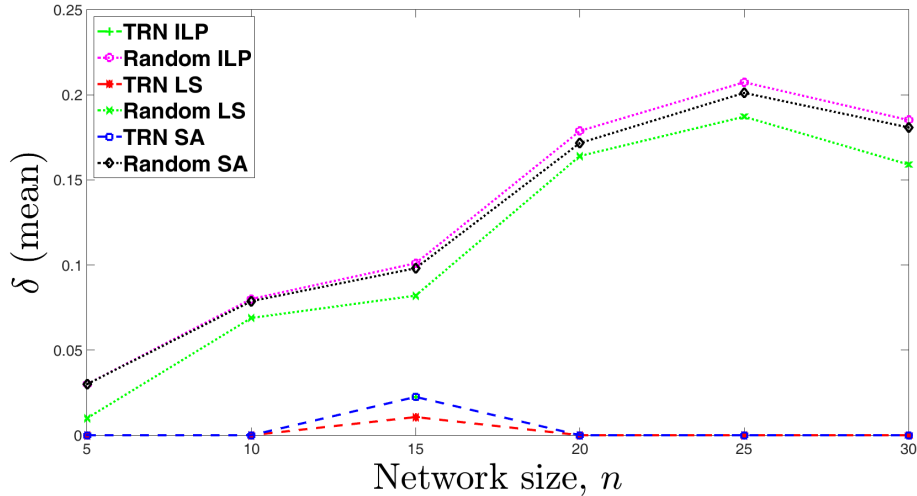
Fig. 21. Mean δ (Eq. (8.2)) evaluated using the integer linear programming (ILP), local search (LS) and simulated annealing (SA) methods, for transcriptional-regulatory subnetworks sampled from E. coli of size $n = 5, 10, 15, 20, 25,$ and 30, compared against optimal solutions found from randomized versions of these networks. If the input network is optimal, then $\delta = 0$.

search and simulated annealing based optimizations of sampled TRNs and their randomly generated networks of similar $|V|$ and $|E|$. The mean of the δ's (Eq. 8.2) are plotted for the different network sizes, while the standard deviation error bars are ignored to avoid obscurity. We find that the SA algorithm gives more reliable δ than that of the LS method, i.e. SA gives δ values closer to that of the ILP.

8.4 Demonstrating Transcriptional Network Connectivity

From the ILP results of Figure 8.3.2, we observe that very few of the subnetworks sampled from E. coli, of any size, exhibit an average path length in their optimized topologies that is smaller than already supported by these networks (green plot). This result should be contrasted by our attempts to optimize randomized networks (magenta), which demonstrate that average shortest path lengths computed for optimized

topologies were significantly reduced over their non-optimized input network.

Moreover, the same comparison should be applied for larger subnetworks that undergo SA based optimization, which can be shown in Figure 8.4. Again, E. coli subnetworks prove to supersede the optimality of their random network versions. Although δ can reach up to 7%, it is a far better performance than that of random structures, which can have $\delta \geq 40\%$.

These results indicates that E. coli network topologies are already well-suited to minimize the average shortest path between transcription factors and their regulated genes. There may be plausible reasons why an evolved transcriptional-regulatory network may experience pressure to minimize the number of regulatory interactions between the regulating proteins and the terminal genes. Common modes of genetic evolution, such as gene duplication and divergence, alters the degree and connectivity of networked protein-coding genes; while these are manifestly local topological alterations, the path-length encompasses regulatory interactions that transcend a gene's local neighborhood, suggesting that network dynamics manifest with system-level properties might play a role in whether an organism's progeny survives. Although dynamics may play a role in correlating topological characteristics at network scales, there is evidence that node-degrees in both biological and non-biological networks are correlated by a geodesic distance of approximately three steps [97]; however, it is as-yet unclear what general mechanism underlies such a correlative relationship.

8.5   Feature Selection

Going back to the previous motivation, which stated that TRNs exhibit robust topologies and therefore have many useful applications like in Bio-inspired wireless sensor networks. The ILP discussed provides experimental evidence that such networks' performances in terms of minimizing distances from transmitter to receiver
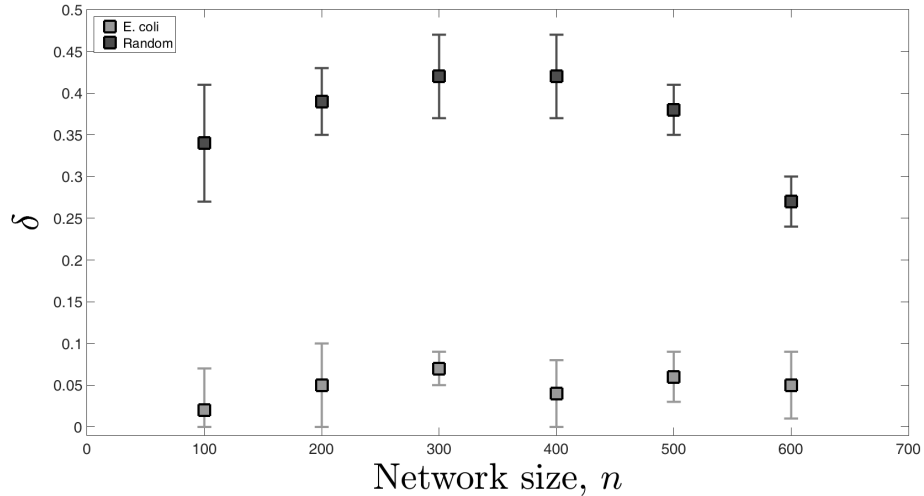
Fig. 22. Mean δ (Eq. (8.2)) evaluated using the simulated annealing algorithm, for transcriptional-regulatory subnetworks sampled from E. coli (light grey boxes) of size $n = 100, 200, 300, 400, 500$, and 600, compared against optimal solutions found from randomized versions of these networks (dark grey).

nodes, can be improved. The initial TRNs preserve their degree distributions, hence, their power-law exponent (maintained scale-free property)- determined by the maximum likelihood estimation. This begs the question; what changed to make $\mathrm{avg}(P)$ smaller.

We select features based on metrics increase/decrease in value due to the networks' optimizations. We report the average gain in feature values in Table 9. As expected from the ILP, the farness of the nodes (or sum of distances from other nodes) decreased as depicted by the diameter's decrease and the average closeness's increases in the average gain.

Interestingly, the average local clustering was almost not changed while the global clustering decreases. This means that nodes keep their neighbors and lost some of the the cliques they originally participated in. This means that nodes preserved their regulations, however, some of the co-regulations were lost to minimize the distances

Table 9. Average gain in structural features of Transcriptional Regulatory networks networks.

Measure	Percentage gain per network Size (%)				
	100	200	300	400	500
Diameter	-0.165	-0.193	-0.273	-0.225	-0.122
$\langle c \rangle$	0.02	0.025	0.143	0.071	0.122
$\langle C^L \rangle$	0.001	-0.007	0	0	0
C^G	-0.035	-0.079	-0.096	-0.23	-0.266

Note: Denote the original Feature vector by F and optimized by F' for a sample set of networks. The average gain would be $\left[\sum_i^f (F_i' - F_i)/F_i \right]/f$, where f is the size of the vector.

to their recipients.

Additional analysis of the different re-wired canonical clique motifs is given in Table 10. Only two motifs were significantly affected by the switching: the feed forward loop (FFL), and the Uplinked mutual dyed (UMD). This means that avg(P) can be minimized by re-wiring FFLs and UMDs, which in return would minimize distances between transcription factors and their intended genes. Although an optimal TRN would increase the speed of transcription, other parameters affect gene expression optimality such as minimizing irreversible speedups and creating redundant regulations achieved by motifs such as FFL and UMD. In other words, in order to achieve structural robustness, it will be achieved at the expense of functional robustness and vice versa. Therefore, we conclude that these structures are necessary features for both functional and structural robustness.

In light of this discovery, we additionally conclude that attachment kernels for optimal TRNs should minimize the creation of FFLs and UMDs per simulation steps. This is the opposite effect of what the random models of chapters 3 and 4 have

illustrated, where increased participation of genes in motif structures is necessary.

Could these structures (FFL and UMD) have evolved over time in order to pre-serve bio-functionalities? We will address this question in the next section.

8.6   Insight into the Evolution of Transcriptional Regulatory Networks

Escherichia coli is a specie commonly found in the lower intestine of warm-blooded organisms, generally considered the most studied prokaryotic organism. *E. coli* K-12 is the most studied strain, hence, used as our model organism to form our hypothesis on evolution. Prokaryotes exhibit high levels of gene diversity due to species and strain evolution throughout the course of their emergence to life- over 3 billions years ago [98]. Genomic evolution are mainly due to the processes of mutation, horizontal gene transfer [99], gene duplication [100], Genome reduction [101].
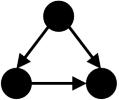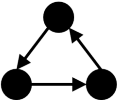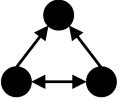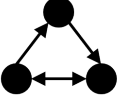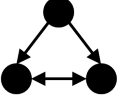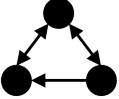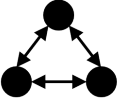
Little is known regarding whether or not acquiring new genes can affect the networks average shortest path. We hypothesize that evolutionary older genes are part of a lower average shortest path compared to genes that were more recently acquired. To test this hypothesis a network of core genome of 56 fully sequenced *E. coli* is used to detect for older genes. Genes that are core, i.e., present in all tested genomes, are known to have existed for a long period of time, while non-core genes might have been recently acquired. A comparison between two E.coli strains estimated their divergence to about 5 million years ago [102].

TRN interactions were downloaded from the latest version of regulonDB [103]- an *E. coli* K-12 Database. Core genome was produced via MicroScope [104] using all 56 *E. coli* isolates from their databases with 50% identity and 80% coverage match. The core genome for the K-12 strain was then obtained and matched to the the genes from regulonDBs interaction list. This list, as well as the entire set of *E. coli*'s interactions were used as separate data-set candidates for optimization using the ILP.

When comparing both sets: *E. coli* vs. *E. coli* core network, there was a significant difference in the number of genes/interactions that were switched, as given in Table 11. In the *E. coli*'s network, there were 2766 (69%) interactions that were switched while the *E. coli*'s core only 1331 (50%). Despite core having 69% of the number of interactions in the non-core, it has 65 more un-switched nodes. In this effect we form a hypothesis that the core genome could have been optimal in terms of the average shortest path, however, it lost its optimality as it evolved over time (i.e., acquired more genes).

In order to arrive at a more conclusive finding regarding the effect of evolution on the average shortest path, more sample TRNs should be considered as well. The same analysis should be employed using core networks at the genus level of *E. coli* along with the genera diverged from the most common ancestor; Salmonella and Shigella, which is thought to have happened about 150 million years ago [105]. To take the set of genes even further back in evolutionary history, we intend to get the core genes of the family level of *E. coli*, Enterobacteriaceae which are thought to have evolved from a common ancestor more than 300 million years ago [106]. Furthermore, for all the above, random samples from *E. coli*'s network of similar number of interactions will be generated for comparison. .

Table 10. Mean percentage of re-wired edges from candidate transcriptional network clique motif structures, due to edge switching from Algorithm 2.

		Candidate TRN sizes				
Motif	Edge	100	200	300	400	500
Feed forward loop	$i \rightarrow j$	0.12	0.19	0.5	0.32	0.54
	$j \rightarrow k$	0.09	0.19	0.45	0.34	0.5
	$i \rightarrow k$	0.05	0.2	0.43	0.39	0.47
Cycle	No change					
Uplinked mutual dyed	$j \rightarrow i$	0.12	0.12	0.61	0.38	0.55
	$j \rightarrow k$	0.09	0.18	0.48	0.16	0.56
	$k \rightarrow j$	0.09	0.12	0.29	0.19	0.48
	$k \rightarrow i$	0.12	0.26	0.62	0.41	0.63
Feed forward and back loop	No change					
Downlinked mutual dyed	No significant change					
Feed back with two mutual dyed	No significant change					
Strongly connected	No change					

Note: nodes i, j, and k considered, are similar to that of Figure 19.

Table 11. Switched vs un-switched (Unchanged) edges in *E. coli*'s complete and core TRN.

	E. coli (all regulonDB)			*E. coli*'s core		
	Total	Unchanged	Switched	Total	Unchanged	Switched
Interactions	4029	1263	2766	2659	1331	1328
Transcription Factors	286	129	157	244	131	113
Genes	2294	993	1301	1524	826	698

CHAPTER 9

CONCLUSIONS AND FUTURE WORKS

The initial premise for the BA-model (i.e., the rich gets richer) is that a nodes degree is the sole contributor to it's evolving linkages within a growing network. Here we support the claim that a rich node indeed gets richer, however, a node's riches is not solely attributed to it's degree. Other factors like nodes closeness and internal community participation can affect the node's evolution as well. One question presented is, how can you grow random networks such that these networks preserve certain aspects of robustness, like it's performance, connectivity and internal clustering? To answer this question, a case on wireless sensor networks was presented, where support vector regressions have selected necessary features to consider. A network growing mechanism will eventually combine two or more of these features.

Attachment kernels are useful in terms of generating random networks with specific structural properties (e.g., degree distributions). These networks can then be used as matching candidates for existing real networks (e.g., Internet and social networks), which helps in giving insight in the dynamics of existing systems. Most growing mechanisms and studies involve undirected networks. Here we provide two different attachment kernel based growing mechanisms for directed networks (Chapters 3 and 4). The first (Ch. 3) uses a preferential attachment scheme based on nodes and in- and out- degree probability distributions, and gives random networks with comparable node-motif participation distributions. The Second (Ch. 4) uses a preferential attachment scheme based on downlink pattern counts probability distributions with comparable in-, out-, cumulative distributions. Both of these algorithms present

comprehensive cases for combining attachment kernels.

The directed transcriptional network growing algorithm of Ch. 4 uses the concept of motif-based preferential attachment, which allows for several new genes and regulatory interactions to be accumulated per step in the network evolution. While many existing algorithms in this area grow undirected networks using the preferential attachment model, or directed networks using the modified preferential attachment scheme with various attachment kernels, they fail to generate networks with high fidelity of motif distributions when contrasted with real-world biological networks. We have proposed using entire transcriptional motifs, which some view as "building blocks of complex networks", as the fundamental unit of network evolution, rather than the accumulation of single genes and regulatory interactions at each potential growth opportunity. Our resulting networks built using this method exhibit higher fidelity to *E. coli* transcriptional networks, both in terms of degree distributions and downlink distributions.

Our algorithm accounts not only for the abundance of downlink motifs, which seem to cover most of the nodes and edges from the *E. coli* transcription regulatory network, but also accounts for two classes of nodes in gene-regulatory networks: genes and transcription factors. One interesting line of future work will be to understand how other transcriptional motifs and types of coupling may contribute to the overall properties of an evolved network model. Another possibility is to consider various centrality measures based on a network renormalized using VMN-based graph transformations. Nevertheless, realistic models of gene-regulatory network evolution will serve to aid future investigations into diverse phenomena, from dynamical signaling over transcriptional-regulatory networks to efforts relating network topology with biological function.

For wireless sensor networks, we observe that degree distribution is not the sole

contributor to optimized routing. SC proves that considering the average shortest path to the sink would be necessary. A network growing mechanism that combines both degree distribution and the average shortest path should be implemented. Comprehensive failure scenarios and ns-2 simulations analysis should be performed for optimal probability distributions among the attachment kernel parameters. Moreover, there are several parameters to consider for attachment kernels such as the density of the hub nodes and the average clustering coefficient. In general, we consider the feature selection process in WSNs completed and ready for the next step (experimenting with attachment kernels).

An example for growing WSNs was presented where we considered an additional feature of a node (additional to it's degree)- it's shortest path to the sink node (or SPS). This feature was coined from Figure 13 due to it's positive Sink Coverage feature. Using this, the sink is considered to be the most influential node in the network followed by it's immediate neighbors, followed by it's second immediate, and so on. Networks were grown using three different mechanism: BA, SPS and a combination of BA and SPS, where for the dual feature case, each feature maintained equal probabilities during each simulation step. SPS gave best results for packet transmission robustness, the dual Combined BA and SPS gave best results in terms of connectivity robustness, while BA gave best results in terms of modularity robustness. Although these insights are not conclusive, it supports our claim that different features should be considered when having robust network generation.

For transcriptional regulatory networks, we consider these networks as robust but not optimal. Meaning, the degree distributions give them resistance towards random nodal/edge deletions, aid in preserving their connectivity post such disturbances, however, degree distributions do not have minimum distance between transmitter and receiver nodes. To address this problem, an integer linear program (ILP) was

93

presented that reduces the average shortests distance to a minimum by rewiring the edges of the network, thereby preserve the degree distributions. The ILP experiences significant difficulties when analyzing larger networks, because the relatively large number of constraints required by this method makes it in-feasible. For example, the largest reported example we are aware of involved a 150 node network with over 10^6 constraints [107], which leads to an excessive convergence time.

Two heuristic algorithms were provided which give solutions (or graphs) near the optimal solution of the ILP. These heuristics can converge within reasonable time for larger networks. Different TRN's were subject to the heuristic, from which we were able to conclude that the only substructures that were significantly affected are the feed forward loop (FFL) and the uplinked mutual dyed (UMD). Other 3-node structures were either not affected at all or slightly reduced, while general topological metrics like the average local clustering remained the same due to the preservation of the degree distributions. Therefore, FFL and UMD are considered relevant features for optimal TRNs. The fact that these two motifs disappear more often than the others may lead to a faster heuristic to find the optimal network. For future works, examining of the switches that are part of these motifs should be prioritized.

Appendix A

AVERAGE SHORTEST PATH LENGTH IN TRANSCRIPTIONAL REGULATORY NETWORKS

The conventional average shortest path formula for a connected undirected graph is given by,

$$\langle d \rangle = \frac{2}{n(n-1)} \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} \tag{A.0.1}$$

for all shortest paths d from nodes i to j. In a transcriptional regulatory network, nodes can be either of two types; transcription factors (TF) and genes making the number of nodes,

$$n = n_t + n_g \tag{A.0.2}$$

where n_t and n_g are the numbers of transcription factors and genes respectively. By substituting Eq. A.0.2 into Eq. A.0.1

$$\langle d \rangle = \frac{2}{n(n-1)} \left[\sum_{i=1}^{n_t} \sum_{j=1}^{n_t} d_{ij} + \sum_{i=1}^{n_t} \sum_{j=n_t+1}^{n_t+n_g} d_{ij} + \right. \\ \left. \sum_{i=n_t+1}^{n_t+n_g} \sum_{j=1}^{n_t} d_{ij} + \sum_{i=n_t+1}^{n_t+n_g} \sum_{j=n_t+1}^{n_t+n_g} d_{ij} \right] \tag{A.0.3}$$

The expression $\sum_{i=1}^{n_t} \sum_{j=1}^{n_t} d_{ij}$ sums the TF-TF shortest paths, which by definition in Section 3, they are considered obsolete. Both the expressions $\sum_{i=n_t+1}^{n_t+n_g} \sum_{j=1}^{n_t} d_{ij}$ and $\sum_{i=n_t+1}^{n_t+n_g} \sum_{j=n_t+1}^{n_t+n_g} d_{ij}$ are equated to zero because they sum the gene-TF and gene-gene shortest paths respectively. These cannot exist because gene nodes have no outgoing edges. Therefore, Eq. A.0.3 can be reduced to,

$$\langle d \rangle = \frac{2}{n(n-1)} \sum_{i=1}^{n_t} \sum_{j=n_t+1}^{n_t+n_g} d_{ij} \tag{A.0.4}$$

95

. The quantity $\frac{2}{n(n-1)}$ is the number of possible pairs in a network, i.e. the number of shortest paths between pairs of nodes in a connected undirected graph. However, in a directed network the number of shortest paths and pairs of nodes are different. Hence, the average shortest path is given by,

$$\langle d \rangle = \frac{1}{|P|} \sum_{i=1}^{n_t} \sum_{j=n_t+1}^{n_t+n_g} d_{ij} \tag{A.0.5}$$

, where $|P|$ is the number of shortest path pairs, such that the source node of each path is a TF and the destination is a gene.

Appendix B

IN- AND OUT-DEGREE DISTRIBUTION

Here we find a solution for the out-degree distribution as given by Eqs 3.3 and 3.4. The derivation to obtain a formula for the in-degree distribution is exactly the same, assuming that the in-degree attachment kernel is independent of a nodes out-degree. Therefore, we restrict ourselves to solving only Eqs 3.3 and 3.4. Note that this derivation follows one provided in detail by [49].

Master equations are given for the out-degree distribution approximated for very large networks (Eqs 3.3 and 3.4):

$$p(K) = np(K-1)mA(K-1) - np(K)mA(K), \text{ and} \tag{B.0.1}$$

$$p(m) = 1 - np(m)mA(m) \tag{B.0.2}$$

Equation B.0.1 can be written as a recursion between out-degrees K and $K-1$:

$$p(K) = \frac{nmA(k-1)}{1 + nmA(K)}p(K-1) \tag{B.0.3}$$

Iterating this recursive relationship, and including Eq. B.0.2, gives

$$p(k) = \frac{1}{nmA(K)}\prod_{i=m}^{k}\left[1 + \frac{1}{nmA(i)}\right]^{-1} \tag{B.0.4}$$

Now, using the fact that $x = e^{\ln x}$, Eq. B.0.4 can be rewritten as

$$p(K) = \frac{1}{nmA(K)}e^{-\sum_{i=m}^{K}\ln[1+1/nmA(i)]} \tag{B.0.5}$$

97

Equation B.0.5 may be further reduced. By assuming the quantity $1/nmA(i)$ is small (i.e., $A(K) > 1/nm$), we may expand $\ln[1 + 1/nmA(i)]$ about $1/nmA(i) = 0$ in a Taylor series: $\ln[1 + 1/nmA(i)] = 1/nmA(i) - [1/nmA(i)]^2/2 + \dots$.

Keeping the leading order term and putting this back into (B.0.5) gives:

$$p(K) = \frac{1}{nmA(K)} e^{-\sum_{i=m}^{K} 1/nmA(i)} \qquad \text{(B.0.6)}$$

Depending on the relationship between A and K, the summation in Eq. B.0.6 may either be exact or further approximated. Because Jeong et al. [26] measured $\gamma = 0.8$ for many real-world networks, we have used this distribution as a choice of synthetic network in the main text. This special case of $nA(K) = K^\gamma/z$ for $1/2 < \gamma < 1$ [with normalization condition $z = \sum K^\gamma p(K)$] was solved along with other cases in Krapivsky et al. [108]; see also Newman [49]:

$$p(K) \sim K^{-\gamma} e^{-zK^{1-\gamma}/m(1-\gamma)}. \qquad \text{(B.0.7)}$$

Appendix C

ABBREVIATIONS, TERMS AND SYMBOLS

Abbreviations and Terms

ER	Erdös - Rényi
E. coli	Escherichia coli
GRN	Gene regulatory network
ILP	Integer Linear Program
NS-2	Network Simulator 2
SF	Scale-free
SW	Small-world
SVR	Support vector regression
TF	Transcription factor
TRN	Transcriptional Regulatory Network
WSN	Wireless Sensor Network
Yeast	Saccharomyces Cerevisiae

Symbols

k	Degree or Degree of a node
γ	Coefficient of the scale-free power law
G	Directed graph
V	Set of nodes forming a graph
E	Set of Edges forming a directed graph G
G'	Undirected graph
L	Set of links forming an undirected graph G'
A	Adjacency matrix storing a directed graph G
M	Adjacency matrix storing an undirected graph G'

d	A shortest path from an arbitrary node i to an arbitrary node j
C^G	Global clustering coefficient
K	The degree vector
DI	Degree Index
HND	Hub nodes density
SC	Sink coverage
ND	Network density
$FFLD$	Feed-forward loop density
BFD	Bifan density
$\langle c \rangle$	Average closeness centrality
$\langle C^L \rangle$	Average local clustering coefficient
ρ	Pearson product-moment correlation coefficient for any two feature vectors
R	Matrix holding ρ for different pairs of F
F	Some feature vector
s	Number of samples or data points for regressor training
X	Feature matrix
w	weight vector (coefficients) for regressor hyperplane
b	bias for regressor hyperplane
O	Output vector for regressor's prediction
δ	Metric for determining optimized networks using ILP
N_{TF}	Number of transcription factor
N_{G}	Number of genes
P	Set of all connected transcription factor and gene pairs
$\mathrm{avg}(P)$	average shortest path between pairs of transcription factors and genes

$M(G)$ Embedded FFL count within G

Z Integer linear program objective parameter

REFERENCES

[1] Réka Albert and Albert-László Barabási. "Statistical mechanics of complex networks". In: *Reviews of Modern Physics* 74 (2002), pp. 47–97.

[2] Qiao Li et al. "Dynamics in small worlds of tree topologies of wireless sensor networks". In: *Journal of Systems Engineering and Electronics* 23 (3 2012), pp. 325–334. DOI: `10.1109/JSEE.2012.00040`.

[3] Chen Bensong, G. N. Rouskas, and R. Dutta. "Clustering Methods for Hierarchical Traffic Grooming in Large-Scale Mesh WDM Networks". In: *IEEE/OSA Journal of Optical Communications and Networking* 2 (8 2010), pp. 502–515. DOI: `10.1364/JOCN.2.000502`.

[4] A. L. Barabási and Z. N. Oltvai. "Network Biology: Understanding the Cell's Functional Organization". In: *Nature Genetics* 5 (2004), pp. 101–114.

[5] M. E. J. Newman. "The Structure and Function of Complex Networks". In: *SIAM Review* 45.2 (2003), pp. 167–256. DOI: `10.1137/S003614450342480`. URL: `http://scitation.aip.org/getabs/servlet/GetabsServlet?prog=normal%5C&id=SIREAD000045000002000167000001%5C&idtype=cvips%5C&gifs=yes`.

[6] S. N. Dorogovtsev and J. F. F. Mendes. *Evolution of Networks: From Biological Nets to the Internet and WWW (Physics)*. New York, NY, USA: Oxford University Press, Inc., 2003. ISBN: 0198515901.

[7] Eric Alm and Adam P Arkin. "Biological networks". In: *Current opinion in structural biology* 13.2 (2003), pp. 193–202. DOI: `10.1016/S0959-440X(03)00031-9`.

[8] Reka Albert, Hawoong Jeong, and Albert-Lazlo Arabasi. "Internet: Diameter of the World-Wide Web". In: *Nature* 401 (1999), pp. 130–131. DOI: `10.1038/43601`.

[9] U. Alon. "Biological networks: the tinkerer as an engineer". In: *Science* 301.5641 (2003), pp. 1866–7+. DOI: `10.1126/science.1089072`. URL: `http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?cmd=Retrieve%5C&%5C#38;db=PubMed%5C&%5C#38;dopt=Citation%5C&%5C#38;list%5C_uids=14512615`.

[10] W. Wang. "Simulating the SARS outbreak in Beijing with limited data". In: *Journal of Theoretical Biology* 227.3 (2004), pp. 369–379. ISSN: 00225193. DOI: `10.1016/j.jtbi.2003.11.014`. URL: `http://dx.doi.org/10.1016/j.jtbi.2003.11.014`.

[11] Lauren Ancel Meyers et al. "Network theory and sars: Predicting outbreak diversity". In: *Journal of Theoretical Biology* 232.1 (2005), pp. 71–81. DOI: `10.1016/j.jtbi.2004.07.026`.

[12] Jeremiah J Faith et al. "Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles". In: *PLoS biology* 5.1 (2007), e8. DOI: `10.1371/journal.pbio.0050008`.

[13] Adam A Margolin et al. "ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context". In: *BMC bioinformatics* 7.Suppl 1 (2006), S7. DOI: `10.1186/1471-2105-7-S1-S7`.

[14] Ilya Shmulevich and Edward R. Dougherty. *Probabilistic Boolean Networks – The Modeling and Control of Gene Regulatory Networks*. SIAM, 2010, pp. I–XIII, 1–267. DOI: `http://dx.doi.org/10.1137/1.9780898717631`.

[15]   J. Feng, J. Jost, and M.P. Qian. *Networks: from biology to theory*. Springer, 2007. DOI: 10.1007/978-1-84628-780-0. URL: http://books.google.com/books?id=qWDl4B0XNL8C.

[16]   S. Ghosh et al. "GaMa : An Evolutionary Algorithmic Approach for the Design of Mesh-Based Radio Access Networks". In: *Local Area Networks* (Nov. 2005), pp. 374–381.

[17]   Hiroaki Kitano. "Biological Robustness". In: *Nature Reviews Genetics* 5 (11 2004), pp. 826–837.

[18]   Hiroaki Kitano. "Towards a theory of biological robustness." In: *Molecular systems biology* 3.1 (Sept. 18, 2007). ISSN: 1744-4292. DOI: 10.1038/msb4100179. URL: http://dx.doi.org/10.1038/msb4100179.

[19]   R. J. Prill, P. A. Iglesias, and A. Levchenko. "Dynamic properties of network motifs contribute to biological network organization". In: *PLoS Biol.* 3.11 (2005), e343. DOI: 10.1371/journal.pbio.0030343.

[20]   Bhanu K. Kamapantula et al. "Quantifying robustness in biological networks using NS-2". In: *Springer-Nanocom* (2015), In Press.

[21]   Bhanu K. Kamapantula et al. "Leveraging the robustness of genetic networks: a case study on bio-inspired wireless sensor network topologies". English. In: *Journal of Ambient Intelligence and Humanized Computing* 5.3 (2014), pp. 323–339. DOI: 10.1007/s12652-013-0180-0.

[22]   Bhanu K. Kamapantula et al. "Performance of wireless sensor topologies inspired by E. coli genetic networks". In: *Pervasive Computing and Communications Workshops, IEEE International Conference on* (2012), pp. 302–307. DOI: http://doi.ieeecomputersociety.org/10.1109/PerComW.2012.6197500.

[23] Preetam Ghosh et al. "Principles of genomic robustness inspire fault-tolerant WSN topologies: A network science based case study". English. In: *PerCom Workshops* (2011), pp. 160–165.

[24] Réka Albert, Hawoong Jeong, and Albert-László Barabási. "Error and Attack Tolerance of Complex Networks". In: *Nature 406* (July 2000), pp. 378–382.

[25] Albert-Laszlo Barabasi and Reka Albert. "Emergence of Scaling in Random Networks". In: *Science* 286.5439 (1999), pp. 509–512. DOI: `10.1126/science.286.5439.509`. eprint: `http://www.sciencemag.org/cgi/reprint/286/5439/509.pdf`. URL: `http://www.sciencemag.org/cgi/content/abstract/286/5439/509`.

[26] A. L. Barabasi et al. "Measuring preferential attachment in evolving networks". In: *Europhysics Letters* 61.61 (), pp. 567–572.

[27] Réka Albert and Albert-László Barabási. "Statistical mechanics of complex networks". In: *Rev. Mod. Phys.* 74 (1 Jan. 2002), pp. 47–97. DOI: `10.1103/RevModPhys.74.47`. URL: `http://link.aps.org/doi/10.1103/RevModPhys.74.47`.

[28] Duncan J. Watts and Steven H. Strogatz. "Collective dynamics of 'small-world' networks". In: *Nature 393* (June 1998), pp. 440–442.

[29] Michael Molloy and Bruce Reed. "A Critical Point for Random Graphs with a Given Degree Sequence". In: *Random Struct. Algorithms* 6.2/3 (Mar. 1995), pp. 161–179. ISSN: 1042-9832. URL: `http://dl.acm.org/citation.cfm?id=259573.259582`.

[30] M. Ángeles Serrano and Marián Boguñá. "Tuning clustering in random networks with arbitrary degree distributions". In: *Phys. Rev. E* 72 (3 Sept. 2005),

p. 036133. DOI: 10.1103/PhysRevE.72.036133. URL: http://link.aps.org/doi/10.1103/PhysRevE.72.036133.

[31]   Michael Mayo et al. "Motif Participation by Genes in E. coli Transcriptional Networks." In: *Front Physiol* 3 (2012), p. 357. ISSN: 1664-042X. URL: http://www.biomedsearch.com/nih/Motif-Participation-by-Genes-in/23055976.html.

[32]   F. Dressler and O.B. Akan. "Bio-inspired networking: from theory to practice". In: *Communications Magazine, IEEE* 48 (11 Nov. 2010). ISSN: 0163-6804. DOI: 10.1109/MCOM.2010.5621985.

[33]   Manfred Eigen and Peter Schuster. *The hypercycle: a principle of natural self-organization*. 1979. URL: https://books.google.com/books/about/The_Hypercycle.html?id=vQ4JAQAAMAAJ.

[34]   Falko Dressler and Ozgur B. Akan. "A Survey on Bio-inspired Networking". In: *Comput. Netw.* 54.6 (Apr. 2010), pp. 881–900. ISSN: 1389-1286. DOI: 10.1016/j.comnet.2009.10.024. URL: http://dx.doi.org/10.1016/j.comnet.2009.10.024.

[35]   Massimo Marchiori Paolo Crucitti Vito Latora and Andrea Rapisarda. "Error and attack tolerance of complex networks". In: *Physica A: Statistical Mechanics and its Applications* 340 (1 2004), pp. 388–394.

[36]   Reuven Cohen et al. "Resilience of the Internet to Random Breakdowns". In: *Phys. Rev. Lett.* 85 (21 Nov. 2000), pp. 4626–4628. DOI: 10.1103/PhysRevLett.85.4626. URL: http://link.aps.org/doi/10.1103/PhysRevLett.85.4626.

[37] Vilmos Ágoston, Péter Csermely, and Sándor Pongor. "Multiple weak hits confuse complex systems: a transcriptional regulatory network as an example". In: *Physical Review E* 71.5 (2005), p. 51909.

[38] S. Arakawa S. Eum and M. Murata. "Toward bio-inspired network robustness - Step 1. Modularity". In: *Bio-Inspired Models of Network, Information and Computing Systems, 2007. Bionetics 2007. 2nd* (Dec. 2007), pp. 84–87.

[39] Shen S. S. Orr et al. "Network motifs in the transcriptional regulation network of Escherichia coli". In: *Nat. Genet.* 31 (May 2002), pp. 64–68.

[40] R. Milo et al. "Network Motifs: Simple Building Blocks of Complex Networks". In: *Science 25* 298.5594 (Oct. 2002), pp. 824–827.

[41] Kang Wu and Christopher V. Rao. "The role of configuration and coupling in autoregulatory gene circuits". In: *Molecular microbiology* 75.2 (2010), pp. 513–527. ISSN: 1365-2958. DOI: 10.1111/j.1365-2958.2009.07011.x. URL: http://dx.doi.org/10.1111/j.1365-2958.2009.07011.x.

[42] S. Mangan and U. Alon. "Structure and function of the feed-forward loop network motif." In: *Proc Natl Acad Sci U S A* 100.21 (Oct. 2003), pp. 11980–11985.

[43] Mark Isalan et al. "Evolvability and hierarchy in rewired bacterial gene networks". In: *Nature* 452.7189 (2008), pp. 840–845. ISSN: 1476-4687. DOI: 10.1038/nature06847. URL: http://dx.doi.org/10.1038/nature06847.

[44] Jeong-Rae Kim, Yeoin Yoon, and Kwang-Hyun Cho. "Coupled feedback loops form dynamic motifs of cellular networks". In: *Biophys J* 94.2 (2008), pp. 359–65. DOI: 10.1529/biophysj.107.105106. URL: http://www.biomedsearch.com/nih/Coupled-feedback-loops-form-dynamic/17951298.html.

[45] Kwon Yung-keun and Cho Kwang-hyun. "Boolean Dynamics of Biological Networks with Multiple Coupled Feedback Loops". In: *Biophys J* 92.8 (2007), pp. 2975–81. DOI: `10.1529/biophysj.106.097097`.

[46] T. Schaffter, D. Marbach, and D. Floreano. "GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods". In: *Bioinformatics* 27.5 (2011 Aug 15), pp. 2263–70. ISSN: 1367-4811. DOI: `10.1093/bioinformatics/btr373`.

[47] M. Cosentino Lagomarsino et al. "Hierarchy and feedback in the evolution of the Escherichia coli transcription network". In: *PNAS* 104.13 (2005), pp. 5516–5520.

[48] Alain Barrat and Romualdo Pastor-Satorras. "Rate equation approach for correlations in growing network models". In: *Phys. Rev. E* 71 (3 Mar. 2005), p. 036127. DOI: `10.1103/PhysRevE.71.036127`. URL: `http://link.aps.org/doi/10.1103/PhysRevE.71.036127`.

[49] Mark Newman. *Networks: An Introduction*. New York, NY, USA: Oxford University Press, Inc., 2010. ISBN: 0199206651, 9780199206650.

[50] Aaron Clauset, Cosma Rohilla Shalizi, and M. E. J. Newman. "Power-Law Distributions in Empirical Data". In: *SIAM Rev.* 51.4 (Nov. 2009), pp. 661–703. ISSN: 0036-1445. DOI: `10.1137/070710111`. URL: `http://dx.doi.org/10.1137/070710111`.

[51] Jacob P. Hoogenboom, Wouter K. den Otter, and Herman L. Offerhaus. "Accurate and unbiased estimation of power-law exponents from single-emitter blinking data". In: *The Journal of Chemical Physics* 125.20, 204713 (2006). DOI: `http://dx.doi.org/10.1063/1.2387165`. URL: `http://scitation.aip.org/content/aip/journal/jcp/125/20/10.1063/1.2387165`.

[52] I. K. Jordan et al. "Conservation and coevolution in the scale-free human gene coexpression network". In: *Mol Biol Evol* 21.11 (2004), pp. 2058–2070.

[53] Victor Kunin, Jose B. Pereira-Leal, and Christos A. Ouzounis. "Functional Evolution of the Yeast Protein Interaction Network". In: *Molecular Biology and Evolution* 21.7 (2004), pp. 1171–1176.

[54] Uri Alon. *An introduction to systems biology : design principles of biological circuits*. Chapman & Hall/CRC mathematical and computational biology series. Boca Raton (Fla.): Chapman & Hall/CRC, 2007. ISBN: 1-58488-642-0. URL: http://opac.inria.fr/record=b1120369.

[55] Ahmed Farouk Abdelzaher et al. "Transcriptional Network growing Models using Motif-based Preferential Attachment". In: *Frontiers in Bioengineering and Biotechnology* 3.157 (2015). ISSN: 2296-4185. DOI: 10.3389/fbioe.2015.00157. URL: http://www.frontiersin.org/systems_biology/10.3389/fbioe.2015.00157/abstract.

[56] C. Genio, Thilo Gross, and Kevin E. Bassler. "All scale-free networks are sparse". In: *Phys Rev Lett* 107 (17 2011).

[57] Fan Chung et al. "Duplication models for biological networks". In: *Journal of computational biology* 10.5 (2003), pp. 677–687. DOI: 10.1089/106652703322539024.

[58] E. Eisenberg and E.Y. Levanon. "Preferential attachment in the protein network evolution". In: *Physical Review Letters* 91.13 (2003), p. 138701. DOI: http://dx.doi.org/10.1103/PhysRevLett.91.138701.

[59] S. Light, P. Kraulis, and A. Elofsson. "Preferential attachment in the evolution of metabolic networks". In: *BMC Genomics* 6.159 (2005), pp. 1–11. DOI: 10.1186/1471-2164-6-159.

[60] D.H. Erwin and E.H. Davidson. "The evolution of hierarchical gene regulatory networks". In: *Nature Reviews Genetics* 10 (2009), pp. 141–148. DOI: `10.1038/nrg2499`.

[61] Michael A. Rowland and Eric J. Deeds. "Crosstalk and the evolution of specificity in two-component signaling". In: *Proceedings of the National Academy of Sciences* 111.15 (2014), pp. 5550–5555. DOI: `10.1073/pnas.1317178111`. URL: `http://www.pnas.org/content/111/15/5550.abstract`.

[62] OpenWetWare. *Biomolecular Breadboards:DNA parts — OpenWetWare.* [Online; accessed 15-August-2015]. 2014. URL: `http://openwetware.org/index.php?title=Biomolecular_Breadboards:DNA_parts&oldid=772520`.

[63] S. Sen, Jongmin Kim, and R.M. Murray. "Designing robustness to temperature in a feedforward loop circuit". In: *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on.* Dec. 2014, pp. 4629–4634. DOI: `10.1109/CDC.2014.7040112`.

[64] W. Deng. *Visualization and Targeted Disruption of Protein Interactions in Living Cells.* 2013. URL: `https://books.google.com/books?id=8kwpoAEACAAJ`.

[65] Eric Bonabeau, Marco Dorigo, and Guy Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems.* New York, NY, USA: Oxford University Press, Inc., 1999. ISBN: 0-19-513159-2.

[66] L.N de Castro and J.I Timmis. *Artificial Immune Systems: A New Computational Intelligence Approach.* Springer, Berlin, Heidelberg, New York, 2002.

[67] Ian F. Akyildiz and Ismail H. Kasimoglu. "Wireless sensor and actor networks: research challenges." In: *Ad Hoc Networks* 2.4 (July 14, 2005), pp. 351–367.

URL: `http://dblp.uni-trier.de/db/journals/adhoc/adhoc2.html#AkyildizK04`.

[68] Barış Atakan and Özgür B. Akan. "Immune System Based Distributed Node and Rate Selection in Wireless Sensor Networks". In: *Proceedings of the 1st International Conference on Bio Inspired Models of Network, Information and Computing Systems*. BIONETICS '06. Cavalese, Italy: ACM, 2006. ISBN: 1-4244-0463-0. DOI: `10.1145/1315843.1315847`. URL: `http://doi.acm.org/10.1145/1315843.1315847`.

[69] Aaron Clauset, Cosma Rohilla Shalizi, and M. E. J. Newman. "Power-Law Distributions in Empirical Data". In: *SIAM Review* 51.4 (2009), pp. 661–703. ISSN: 0036-1445. DOI: `10.1137/070710111`. URL: `http://link.aip.org/link/?SIR/51/661/1`.

[70] Reuven Cohen et al. "Breakdown of the Internet under Intentional Attack". In: *Phys. Rev. Lett.* 86 (16 Apr. 2001), pp. 3682–3685. DOI: `10.1103/PhysRevLett.86.3682`. URL: `http://link.aps.org/doi/10.1103/PhysRevLett.86.3682`.

[71] P. Erdös and A Rényi. "On the Evolution of Random Graphs". In: *Publ. Math. Inst. Hung. Acad. Sci. 5* (1960).

[72] Archana Belle et al. "Quantification of protein half-lives in the budding yeast proteome". In: *Proc. Natl. Acad. Sci.* (2006).

[73] Bhanu K. Kamapantula et al. "Performance of wireless sensor topologies inspired by E. coli genetic networks". In: *Pervasive Computing and Communications Workshops, IEEE International Conference on* (2012), pp. 302–307. DOI: `http://doi.ieeecomputersociety.org/10.1109/PerComW.2012.6197500`.

[74]   *The Network Simulator NS-2.* http://www.isi.edu/nsnam/ns/.

[75]   Chris Savarese, Jan M. Rabaey, and Koen Langendoen. "Robust Positioning Algorithms for Distributed Ad-Hoc Wireless Sensor Networks". In: *Proceedings of the General Track of the Annual Conference on USENIX Annual Technical Conference.* ATEC '02. Berkeley, CA, USA: USENIX Association, 2002, pp. 317–327. ISBN: 1-880446-00-6. URL: http://dl.acm.org/citation.cfm?id=647057.713854.

[76]   T. Feyessa and M. Bikdash. "Measuring nodal contribution to global network robustness". In: *Southeastcon, 2011 Proceedings of IEEE.* Mar. 2011, pp. 131–135. DOI: 10.1109/SECON.2011.5752920.

[77]   K.-. Goh, B. Kahng, and D. Kim. "Graph theoretic analysis of protein interaction networks of eukaryotes". In: *Physica A: Statistical Mechanics and its Applications* 357.3 (2005), pp. 501–512. URL: http://EconPapers.repec.org/RePEc:eee:phsmap:v:357:y:2005:i:3:p:501-512.

[78]   E. Ravasz et al. "Hierarchical organization of modularity in metabolic networks." In: *Science (New York, N.Y.)* 297.5586 (Aug. 30, 2002), pp. 1551–1555. ISSN: 1095-9203. DOI: 10.1126/science.1073374. URL: http://dx.doi.org/10.1126/science.1073374.

[79]   Stanislava Soro and Wendi B. Heinzelman. "Prolonging the Lifetime of Wireless Sensor Networks via Prolonging the Lifetime of Wireless Sensor Networks via". In: *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05) - Workshop 12 - Volume 13.* IPDPS '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 236.2–. ISBN: 0-7695-2312-9. DOI: 10.1109/IPDPS.2005.365. URL: http://dx.doi.org/10.1109/IPDPS.2005.365.

[80] Seema Bandyopadhyay and E. J. Coyle. "An energy efficient hierarchical clustering algorithm for wireless sensor networks". In: *Proceedings of IEEE INFOCOM 2003*. San Francisco, CA, USA, Apr. 2003, pp. 1713–1723. URL: http://www.comsoc.org/confs/ieee-infocom/2003/papers/42%5C_02.PDF.

[81] Vivek Mhatre and Catherine Rosenberg. "Homogeneous vs. heterogeneous clustered sensor networks: A comparative study". In: *In Proceedings of 2004 IEEE International Conference on Communications (ICC 2004*. 2004, pp. 3646–3651.

[82] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan. "An Application-specific Protocol Architecture for Wireless Microsensor Networks". In: *Trans. Wireless. Comm.* 1.4 (Oct. 2002), pp. 660–670. ISSN: 1536-1276. DOI: 10.1109/TWC.2002.804190. URL: http://dx.doi.org/10.1109/TWC.2002.804190.

[83] Qunfeng Dong. "Maximizing System Lifetime in Wireless Sensor Networks". In: *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks*. IPSN '05. Los Angeles, California: IEEE Press, 2005. ISBN: 0-7803-9202-7. URL: http://dl.acm.org/citation.cfm?id=1147685.1147690.

[84] Michael P. H. Stumpf, Carsten Wiuf, and Robert M. May. "Subnets of scale-free networks are not scale-free: Sampling properties of networks". In: *Proceedings of the National Academy of Sciences* 102.12 (2005), pp. 4221–4224.

[85] Peter Flach. *Machine Learning: The Art and Science of Algorithms That Make Sense of Data*. New York, NY, USA: Cambridge University Press, 2012. ISBN: 1107422221, 9781107422223.

[86] Ahmed Abdelzaher et al. "Empirical prediction of packet transmission efficiency in bio-inspired Wireless Sensor Networks." In: *ISDA*. Ed. by Ajith Abraham et al. IEEE, 2012, pp. 705–710. ISBN: 978-1-4673-5117-1. URL: `http://dblp.uni-trier.de/db/conf/isda/isda2012.html#AbdelzaherKGD12`.

[87] Laurent Beauguitte and César Ducruet. "Scale-free and small-world networks in geographical research: A critical examination". In: *17th European Colloquium on Theoretical and Quantitative Geography*. Ed. by Stamatis Kalogirou. Athènes, Greece, Sept. 2011, pp. 663–671. URL: `https://halshs.archives-ouvertes.fr/halshs-00623927`.

[88] M. E. J. Newman. "A measure of betweenness centrality based on random walks". In: *Social Networks* (2005).

[89] Stefan Wuchty and Peter F. Stadler. "Centers of complex networks". In: *Journal of Theoretical Biology* 223.1 (2003), pp. 45–53. DOI: `10.1016/S0022-5193(03)00071-7`. URL: `http://www.sciencedirect.com/science/article/B6WMD-48F5JDP-1/2/1d420f5cd8ffb58769a5d2e06d8276fd`.

[90] Chih-Chung Chang and Chih-Jen Lin. "LIBSVM: A Library for Support Vector Machines". In: *ACM Trans. Intell. Syst. Technol.* 2.3 (May 2011), 27:1–27:27. ISSN: 2157-6904. DOI: `10.1145/1961189.1961199`. URL: `http://doi.acm.org/10.1145/1961189.1961199`.

[91] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. "Optimization by simulated annealing". In: *Science* 220.4598 (1983), pp. 671–680.

[92] Melanie Mitchell. *An Introduction to Genetic Algorithms*. Cambridge, MA, USA: MIT Press, 1998. ISBN: 0262631857.

[93] R.K. Ahuja et al. *Network Flows*. Working paper (Sloan School of Management). Alfred P. Sloan School of Management, Massachusetts Institute of Technology, 1988. URL: https://books.google.com/books?id=WjpkHQAACAAJ.

[94] *IBM ILOG CPLEX Optimizer.* http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/. 2010.

[95] Ahmed F. Abdelzaher et al. "Contribution of canonical feed-forward loop motifs on the fault-tolerance and information transport efficiency of transcriptional regulatory networks". In: *Nano Communication Networks* 6.3 (2015), pp. 133–144. ISSN: 1878-7789.

[96] Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1993. ISBN: 0-13-617549-X.

[97] M. Mayo, A.F. Abdelzaher, and P. Ghosh. "Long-range degree correlations in complex networks". In: *Computational Social Networks* 2.1 (2015), pp. 1–13.

[98] J. L. Payne et al. "Two-phase increase in the maximum size of life over 3.5 billion years reflects biological innovation and environmental opportunity". In: *PNAS* 106.1 (2009), pp. 24–27.

[99] A Toussaint and M Chandler. "Prokaryote genome fluidity: toward a system approach of the mobilome". In: *Methods Mol Biol* (2012), pp. 57–80.

[100] M. H. Serres et al. "Evolution by leaps: gene duplication in bacteria". In: *Biol Direct* (Nov. 23, 2009), pp. 4–46.

[101] Vittorio Boscaro et al. "Polynucleobacter necessarius, a model for genome reduction in both free-living and symbiotic bacteria". In: *PNAS* 110 (46 Nov. 12, 2013), pp. 18590–18595.

[102] Vittorio Boscaro et al. "Further evidence of constrained radiation in the evolution of pathogenic Escherichia coli O157:H7". In: *Infect Genet Evol.* 10 (8 Dec. 2010), pp. 1282–1285.

[103] Socorro Gama-Castro et al. "RegulonDB version 9.0: high-level integration of gene regulation, coexpression, motif clustering and beyond." In: *Nucleic Acids Research* 44.Database-Issue (2016), pp. 133–143. URL: `http://dblp.uni-trier.de/db/journals/nar/nar44.html#Gama-CastroSSLM16`.

[104] Vincent Miele, Simon Penel, and Laurent Duret. "Ultra-fast sequence clustering from similarity networks with SiLiX". In: *BMC Bioinformatics* 12.1 (2011), pp. 1–9. ISSN: 1471-2105. DOI: `10.1186/1471-2105-12-116`. URL: `http://dx.doi.org/10.1186/1471-2105-12-116`.

[105] Howard Ochman and Allan C. Wilson. "Evolution in bacteria: Evidence for a universal substitution rate in cellular genomes". In: *Journal of Molecular Evolution* 26.1 (1987), pp. 74–86. ISSN: 1432-1432. DOI: `10.1007/BF02111283`. URL: `http://dx.doi.org/10.1007/BF02111283`.

[106] David J. Baumler et al. "Inferring ancient metabolism using ancestral core metabolic models of enterobacteria". In: *BMC Systems Biology* 7.1 (2013), pp. 1–17. ISSN: 1752-0509. DOI: `10.1186/1752-0509-7-46`. URL: `http://dx.doi.org/10.1186/1752-0509-7-46`.

[107] Thang N. Dinh and My T. Thai. "Precise Structural Vulnerability Assessment via Mathematical Programming". In: *MILCOM* (Nov. 2011), pp. 1351–1356.

[108] P. L. Krapivsky, S. Redner, and F. Leyvraz. "Connectivity of Growing Random Networks". In: *Physical Review Letters* 85.21 (2000), pp. 4629–4632. DOI: `10.1103/physrevlett.85.4629`. URL: `http://dx.doi.org/10.1103/physrevlett.85.4629`.