UNIVERSITY OF CALIFORNIA,
IRVINE

Alignment-Based Analysis of Dynamics in Complex Networks

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Computer Science

by

Edwin Vargas

Dissertation Committee:
Professor Wayne Hayes, Chair
Professor Amelia C. Regan
Professor German Enciso

2017

Dedication

To my parents and siblings, who share with me this journey of personal development and service to others.

And

To Ms. Emily Smith, whose encouragement, patience, and intelligence motivate me every day.

Table of Contents

List of Figures

List of Tables

Acknowledgements

I am grateful to my advisor and committee chair, Prof. Wayne Hayes, for facilitating the continuation of my academic path in an field of profound interest and professional relevance.

I would also like to thank Dr. Matheus Vianna, collaborator and friend, for his valuable feedback and challenging questions, which helped me fine tune software and analyses. Moreover, I thank my committee members, Prof. German Enciso and Prof. Amelia Regan, for their encouragement and feedback.

Finally, I hereby express my deepest gratitude to Mrs. Holly Byrnes, Computer Science Department Manager at UCI, who quarter after quarter made sure I was able to continue my studies by providing me with suitable teaching assistantship appointments. If not for her direct involvement, I would not have been able to successfully bring this process to a completion.

Curriculum Vitae

Edwin Vargas

Research Interests

Complex Systems: Large, noisy, dynamic, and heterogeneous real-world networks.
Machine Learning: Graphical models; ensemble and deep learners; data representation.
Software Engineering: Formal models; agile development; quality driven development.
Computational Biology: Genome, proteome, and microbiome topology vs function/disease.
Computational Statistics: Optimized learning; Bayesian methodologies.
Computational Chemistry: Protein folding; computational drug discovery and design.

Education

Ph.D., Computer Science **2017**
University of California, Irvine
Dissertation: *Alignment-Based Analysis of Dynamics in Complex Networks*
Advisor: Wayne Hayes

M.S., Software Engineering **2016**
California State University, Fullerton
Capstone: *Formal Concurrent Verification for Scientific Network Analysis*

M.S., Computer Science **2012**
University of California, Irvine
Thesis: *An Integrative Approach to the In Silico Creation of Domain-Shuffling Chimeras*
Advisor: Richard Lathrop

Certificate in Mathematical, Computational, and Systems Biology **2010**
University of California, Irvine

B.S., Mathematics **2004**
Middle Tennessee State University

B.S., Computer Science **2003**
Middle Tennessee State University

Research Experience

Graduate Student Researcher **2013 - 2017**
PI: Wayne Hayes, Department of Computer Science
University of California, Irvine, CA

Graduate Student Researcher **2010-2012**
P53 Research Group, Department of Computer Science
University of California, Irvine, CA

Bioinformatics Grant Specialist **2007-2009**
Molecular Biochemistry Research Group
Texas State University, San Marcos, TX

Research Intern **2003**
SIPHER Program
Vanderbilt University, Nashville, TN

Teaching Experience

Associate Professor **2017**
MiraCosta College, Oceanside, CA

Associate Professor **2016**
Saddleback Community College, Mission Viejo, CA

Teaching Assistant **2011 - 2017**
University of California, Irvine, CA

Teaching Instructional Assistant **2007-2008**
Texas State University, San Marcos, TX

Teaching Assistant **2002**
Middle Tennessee State University, Murfreesboro, TN

Professional Experience

Director of Software Engineering **2006-2007**
Kula Media Group San Marcos, TX

Product Knowledge Engineer **2004-2006**
Edgenet, Inc. Brentwood, TN

Lab & Teaching Assistant **2002-2004**
Middle Tennessee State University Murfreesboro, TN

Corporal **1994-1996**
Escuela de Lanceros Colombian Armed Forces

Articles & Peer-Reviewed Conference Papers

Xiaojie Qi, **Edwin Vargas**, Liza Larsen, Whitney Knapp, G. Wesley Hatfield, Richard Lathrop. Directed DNA Shuffling of Retrovirus and Retrotransposon Integrase Protein Domains. PLOS ONE. May 17th, 2013.

Edwin Vargas. Alignment-Based Analysis of Dynamics in Complex Networks. Journal of Complex Networks. Oxford University Press. *Submitted*.

Edwin Vargas. D-GRAAL: A Fast, Scalable, Dijkstra-based Graph Alignment Algorithm. Proceedings Of The 6th International Conference on Complex Networks and Their Applications. Lyon, France. 2017. *Submitted*.

Edwin Vargas, Matheus Viana, Suzanne Rafelski, Wayne Hayes. Computational Identification of Membrane Events in Mitochondrial Networks. *In preparation*.

Technical Reports

Edwin Vargas. EST Analysis and Annotation of Xiphophorus Maculatus. Technical Report. Texas State University. San Marcos, TX. March of 2009.

Edwin Vargas, Bina Shah, Trione Vincent. Technical Report: Summer Internship Program in Hybrid and Embedded Systems. Technical Report. Vanderbilt University. Nashville, TN. August of 2003.

Conference Presentations

Edwin Vargas. Alignment-Based Analysis of Dynamics in Spatio-Temporal Networks. 1st North American Social Networks Conference (NASN). Washington, DC. July 2017.

Edwin Vargas, X. Qi, G.W. Hatfield, S. Sandmeyer, R. Lathrop. An Integrative Computational Approach to Protein Domain Boundary Assessment and Validation. UCI Center for Complex Biological Systems (CCBS) Annual Retreat. Santa Monica, CA. March 2012.

X. Qi, L. Zicker-Larsen, W. Cornick, **E. Vargas**, S. Hung, R. Lathrop, G. W. Hatfield, S. Sandmeyer. Partitioning integrase activities using directed domain shuffling of HIV, PFV, and Ty3 integrases. 36th annual meeting on Retroviruses. Cold Spring Harbor Laboratory, Cold Spring Harbor, New York. May 2011.

R. Baronio, C. Wassman, F. Salehi, **E. Vargas**, P. Kaiser, R. Lathrop.Computer Guided p53 Reactivation. 1st UC Irvine School Of Medicine Clinical, Basic & Translational Science Festival. Irvine, California. May 2011.

E. Vargas, J. Ovadia, Qing Nie, Arthur D. Lander, and Anne L. Calof. Investigating Control Mechanisms of Neuronal Cell Development and Regeneration in the Olfactory Epithelium. Presented at Systems Biology of Stem Cells Symposium. Irvine, CA. May 2010.

Fellowships & Funding

Miguel Velez - Endowment Fellowship **2015, 2016**
University of California, Irvine, CA

Faculty Mentor Program (FMP) Fellowship Honorable Mention Award **2012**
University of California, Irvine, CA

Biomedical Informatics Training (BIT) Fellowship **2011-2012**
Institute for Genomics and Bioinformatics
University of California, Irvine, CA

N.I.H. Pre-Doctoral Training Fellowship **2009-2011**
Mathematical, Computational, and Systems Biology Program
University of California, Irvine, CA

H.H.M.I. Pre-Doctoral Training Fellowship **2009-2010**
Mathematical, Computational, and Systems Biology Program
University of California, Irvine, CA

N.I.H. Minority Biomedical Research Support Program **Summer 2009**
University of California, Irvine, CA

S.G.A. Scholarship **Fall 2007, Spring 2008**
Texas State University, San Marcos, TX

NSF-CSEM Scholarship **2002-2004**
Middle Tennessee State University, Murfreesboro, TN

Abstract of the Dissertation

Alignment-Based Analysis of Dynamics in Complex Networks

By

Edwin Vargas

Doctor of Philosophy in Computer Science

University of California, Irvine, 2017

Professor Wayne Hayes, Chair

Network analysis and its exploit of graph-theoretical properties such as topology are important mathematical and computational problems, with implications in a variety of fields such as social sciences, economics, and the biological sciences. While the study of complex networks has evolved quite rapidly over the last few years, largely as a result of massive data sets and computational techniques, its study remains largely domain-specific, and focused on static, global network properties. Thus, the ability to achieve node-level granularity in the analysis of rapidly changing network data is currently non-existent.

In order to apply network analysis to rapidly changing complex network systems, such as those representing real-time, physical phenomena, their dynamics must be analyzed in a computationally efficient manner, and predictive mechanisms must exist for their classification and forecasting. This work provides important steps towards the efficient and effective analysis of dynamics in real-world complex networks.

First, we introduce a computationally efficient algorithm for the comparison of net-

works. The development of network alignment as an analysis technique has enabled the efficient mining of complex information from vast sets of data, such as those generated in the field of proteomics. However, network alignment methods have mostly been applied to the analysis of networks representing different entities, such as differing species or corporate structures, under static conditions. Our research explores the application of network alignment to the identification of dynamics in large, potentially fast-changing networks. As a result, we develop a computationally efficient algorithm with potential for distributed and parallel improvements. We compare our algorithm to well established network alignment alternatives, under various network models and complexity properties.

Second, this work presents methodology for the identification and labeling of node/edge-level events in a complex spatio-temporal network system. A process and associated measurements are introduced to effectively identify events, followed by the implementation of an inference method for the labeling of such events.

Finally, our methodology for identification and labeling of events is applied to a biological problem of high consequence for human health: That the identification of fission and fusion events in mitochondrial networks. Lastly, Successes and challenges of this application are discussed.

As stated, the work here presented comprises a series of important steps for the utilization of network alignment in the exploration of dynamics in complex networks, solely using the structural and topological features of the network. Further, the methodology here presented is not domain-specific, thus opening the possibilities for their utilization in the analysis of dynamics in real-world dynamic complex networks.

Chapter 1

Introduction

1.1 Motivation

Network science, defined by the United States National Research Council as "the study of network representations of physical, biological, and social phenomena leading to predictive models of these phenomena" [19], is a highly inter-disciplinary field whose methods are drawn principally from the mathematical areas of graph theory and statistics. Network analysis, and specifically the exploit of network structural properties such as topology, is an important mathematical and computational problem, with implications in a variety of fields such as social sciences [84], economics [15], and the biological sciences [71].

Now, *complex networks* are systems of interconnected entities, represented in terms of pairwise relationships, whose underlying structural principles are irregular, complex, and often changing with time [13]. These systems emerge naturally in most areas of life, and have gained increasing attention with the adoption of ever more powerful data gathering mechanisms, such as sensors, websites, and imaging technologies. Con-

sequently, the ubiquitous presence of complex networks allows for the exploration of relationships, topologies, and patterns, therefore contributing information for control theory to be applied ([95]).

1.2 Challenges in the Analysis of Dynamics in Real-World Complex Networks

In addition to the opportunity to represent, explore, and understand critical real-world phenomena, the study of complex networks also presents computational challenges, largely due to the complexity of the phenomena represented, the size of available data sets, as well as to the consequent search for suitable models and analysis techniques. These challenges are further compounded when considering the impact of time on the various aspects subject of analysis.

Choosing a suitable network model to represent the underlying real-world phenomena in a system of interconnected units can be non-trivial. Adaptation, growth, and other behaviors may impose non-prescriptive pressures on a network's structure, often outside from simple linear transformations on a chosen representational architecture. According to Boccaletti et al [13], a first approach to capture the network's properties of such systems is to model them as graphs whose nodes represent the dynamical units, and whose links stand for the interactions between them. However, on one hand, scientists wrestle with structural issues, such as topology of a complex network representation, revealing the unifying principles at the core of the real networks, and developing models to mimic the growth of a network and reproduce its structural properties. On the other hand, many relevant questions arise when studying complex networks dynamics, such as learning how a large ensemble of dynamical systems that

interact through a complex wiring topology can behave collectively [13].

The analysis of dynamics in complex networks also implies the addition of time as a driver of change. As described later in this work, much of the work that has been carried out in the analysis of complex networks focuses on static properties of a given network. In order to capture the changes over a period of time, suitable extensions must be made to existing methods, or new alternatives must be created. In this work, we evaluate the changes occurring between any two points in a time series of network snapshots via *graph alignment*, a computational technique for the comparison of two networks. To date, the development of network alignment as a comparative analysis technique has enabled the efficient mining of information from dissimilar sets of data, such as those generated in the field of proteomics. Just as sequence alignment has proved to be invaluable in comparative biology, so has network alignment shown promise as a method for comparing species [45]. This work addresses the challenges and opportunities in using network alignment as an analysis tool for the understanding of dynamics in real-world complex networks, such as neuronal and tubular networks encountered in live organisms, or any other physical networks that undergo sometimes fast-occurring changes during a period of time. Our research encompasses a novel, more efficient network alignment algorithm, as well as the application of graph alignment to a broader range of graph classes, such as fast-changing spatio-temporal networks like those observed in living organisms.

A final challenge in the analysis of dynamics in complex networks arises in the context of systems whose network abstraction are extracted from real-world physical systems, such as those encountered in living organisms. While there may exist methods for accurately achieving the desired abstraction, identifying change events across snapshots may be difficult, and perhaps only achieved by expert manual labor at the present. Consequently, automating the process of change event identification in real-world

systems can aid greatly to the analysis of relevant behaviors.

1.3 Dissertation Contributions and Outline

This dissertation contributes to the solution of the challenges described above, as demonstrated within the context of the complex network abstraction for real-world mitochondrial tubular matrix systems [96]. In this work we introduce a novel, more efficient graph alignment algorithm, whose increase in computational performance while preserving tolerable quality results can help in the active alignment of fast-changing network time-series, or in the online analysis of streams of network data. Further, a method and associated metrics is described for the identification of structural changes in spatio-temporal network systems, and subsequently applied to the identification and labeling of biophysical fission and fusion events in budding yeast mitochondrial tubular networks.

The remainder of this dissertation is organized as follows:

In chapter 2, we provide simple yet necessary background on the theory that makes possible our work. This chapter encompasses graph-theory concepts, network models and the generation of in silico network systems, as well as network comparison and alignment.

In chapter 3, we discuss the state of the art for analysis of dynamics in complex networks, describing the progress achieved in a variety of fields such as physics, chemistry, social sciences, and computer science. We then allocate a special section to the analysis of dynamics in biological networks, in order to create the appropriate context for our discussion on mitochondrial networks.

In chapter 4, we introduce and describe in detail our novel algorithm, D-GRAAL, for the efficient computational alignment of networks. Along with the description of our algorithm, we provide comparison results with two well established alternatives.

In chapter 5, we introduce methodology for the identification and labeling of network structural changes in a time series of spatio-temporal network system. In silico experiments are reported to demonstrate the achievement of automated event identification and labeling.

In chapter 6, we apply the methodology introduced in chapter 5 to the problem of automatic detection and labeling of fission/fusion events in budding yeast mitochondrial network systems, which change over time to adapt and respond to cellular and environmental changes.

Finally, chapter 7 provides relevant conclusions on the work done, as well as our perspective on opportunities, challenges, and plans for future research.

Chapter 2

Background

2.1 Graphs

The mathematical field of graph theory is founded on the concepts of *node* and *edge*, introduced by Leonhard Euler in his paper, *Seven Bridges of Königsberg*, published in 1736. A *graph* is a mathematical construct used to represent pairwise relationships (edges) between a given set of entities (vertices or nodes). We here review only the graph theoretical concepts most relevant to our work.

Formally, a graph is an ordered pair G = (V,E), such that V is a set of vertices (or nodes) of G and $E \subseteq V \times V$ is a set of edges of G. The total number of nodes and edges in a given graph, G, are $|V|$ and $|E|$, respectively. Further, each edge has a pair of nodes associated with it, called the edge's *endpoints*. A *self-loop* is an edge that joins a single endpoint to itself ($\forall v \in V : (v, v) \notin E$). Nodes joined by an edge are called *adjacent*, and the reciprocal relationship between them is that of being *neighbors*.

All edges in a graph may or may not have a starting (head) and an ending (tailend) node, therefore setting a direction the each relationship. Consequently, a graph may be *directed* or *undirected* ((u, v) $\in$ E : (u, v) = (v, u)), depending on the directionality (or lack of thereof) on its edges. A graph G(V,E) is *simple* if it is unweighted (no weight function is defined on the edges), undirected, and contains no self-loops or multiple edges. In this dissertation, we concern solely with finite graphs, with no self-loops, that have an associated weight function on their edges. However, our results may extend to other types of graph, as explained when appropriate.

Now, a *subgraph* of G is a graph such that all of its nodes and edges belong to G. In addition, an *induced* subgraph of a graph G on a subset S of nodes of G, is obtained by taking S and all edges of G having both end-points in S. Moreover, a *graphlet* is a small, connected, and induced subgraph of a larger network [70].

A *path* in a graph is a sequence of nodes and edges, such that a node belongs only to the edges before and after it. If no node is repeated, the sequence is a *simple path*. Further, A *path length* is the number of edges in a path, and the smallest number of edges that have to be traversed in a network to get from one node to another is called the *distance* between the two nodes. Moreover, a path through the network that achieves that distance is called the *shortest path* between the nodes, and the *diameter* of a network is the maximum shortest path length.

Finally, an *isomorphism* of graphs G and H is a bijection f between the node sets of G and H such that any two nodes u and v of G are adjacent in G if and only if f (u) and f (v) are adjacent in H.

2.2 Networks

While the terms graph and network are often used interchangeably, we here use *network* when adding a context of meaning to the elements of a given graph. Thus, nodes of a network represent elements of a complex system and two nodes u and v are connected by an edge if they are somehow related. The definition of such relationships may in turn have specific properties and meanings, such that the analysis of the network may extend beyond the mere graph-specific analysis.

An example of two superimposed networks, that of human proteins and that of budding yeast proteins, are shown in a schematic representation in figure 2.1. By successfully mapping the nodes from one network to the other, the scientist is able to infer potential transferable knowledge such as toxicity, cellular processes in which they participate, among many others.

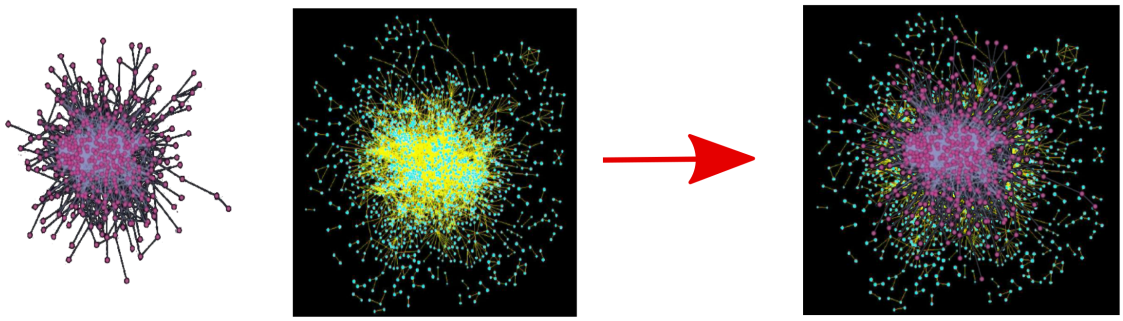


Figure 2.1: Schematic representation of a network alignment

2.3 Topological Network Properties

In order to effectively analyze network abstractions of suitable phenomena, the elements in the network must be evaluated with respect to standardized structural (or topological) metrics. These metrics can be assessed at the node level, at the regional and subgraph level, as well as on the full network. Again, while there are many different metrics that can be used in the evaluation and analysis of nodes, subgraphs, and full networks, we here describe only those directly relevant to the work presented in this document, as outlined below.

2.3.1 Node Properties

The *degree* of a node is the number of edges incident to the node. This is, the number of its neighbors in the network. The *eccentricity* of a node is the maximum shortest path distance from it to any other node in its connected component. Further, the topological *centrality* of a node is a measure of its topological importance in the network. However, there exist various measures by which such importance can be determined.

1. *Degree Centrality:* Nodes with a large degree have a high centrality;
2. *Closeness Centrality:* Nodes with short paths to all other nodes have a high centrality
3. *Betweenness Centrality:* Nodes that occur in the largest number of the shortest paths have a high centrality.
4. *Graphlet Degree Centrality (GDC):* Nodes involved in a higher number of different graphlets have a high centrality [58].

Finally, the *clustering coefficient* of a node is the probability that two neighbors of a node are themselves connected. Clearly, this is a measure of a network's cohesiveness, effectively measuring how densely connected the nodes neighbors are.

2.3.2 Global Properties

A natural extension of basic graph-theoretical concepts and properties at the node level is that of assessing the mean and distributional aspects of different metrics. Thus, considering the shortest path between two given nodes, the average of shortest path lengths over all pairs of nodes in a network is called the *average graph's diameter*. Now, the *distribution of shortest path lengths*, is the distribution of percentages of network node pairs at the shortest path length of size l, for all l.

Moreover, the average of clustering coefficients over all nodes in a network is the *clustering coefficient of the network* and, in turn, the distribution of the average clustering coefficients of all nodes of degree k in a network, is the *clustering spectrum*.

Further, the *density* of a network is the proportion of realized edges in a network, with respect to all the potential edges in a network with N nodes. This is,

$$Density = \frac{2|E|}{N(N-1)}$$

A graphical representation of our density definition is shown in figure 2.2, where a network with 5 nodes and 5 edges (out of 10 possible) has a density of 0.5.

Finally, the concept of node degree naturally extends to a network property, defining the *average degree of a network* as the average of degrees over all nodes in the network, while a *degree distribution*, P(k), is the distribution of percentages of nodes from the network that have degree k, for each value of k.

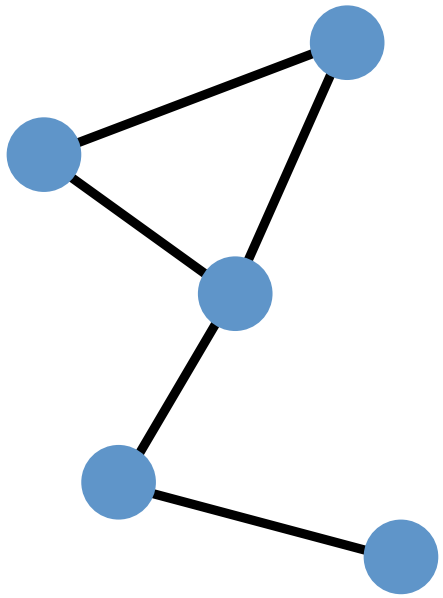


Figure 2.2: Network with 0.5 density

2.3.3 Local Properties

As described, some structural properties can be assessed on the set of two or more nodes, with their corresponding edges. Further, given that topologies can vary greatly while yielding the equivalent global properties, local topological measurements provide additional constraints so that analyses and comparisons can be carried out.

Thus, *motifs* are small subgraphs that occur statistically significantly more often than in randomized networks ([60], [59], [62]). Consequently, network motifs are estimated by counting the number of appearances of all types of n-node subgraphs in the network, as well as in an ensemble of randomized networks with the same number of nodes and edges as the real-world network. A drawback of using motifs is that their assessment with respect to randomized networks fully depends on the random

network model used. Further, selecting an appropriate model is not always obvious.

Further, graphlets are successful at capturing the topological fingerprint of a node's extended neighborhoods, and can provide the basis for a local distributional assessment of the different nodes in a set. Consequently, graphlet degree vectors (GDV) can be measured to assert how many graphlets of a given degree (1-73) is a node part of.

2.4 Network Models

Networks are used to represent the interconnected entities whose phenomena need to be understood. Such entities can be abstractions of either real or theoretical notions, depending on the semantics of the relationships and the network as a whole.

From the work of Watts and Strogatz on small-world networks, appeared in Nature in 1998 [100], and that by Barabsi and Albert on scale-free networks, appeared one year later in Science [7], real-world systems were shown to be adequately represented by network abstractions, therefore setting the path to modern day network modeling, generation, and analysis.

While the number of model alternatives for real-world complex networks is as large as the number of systems that may need such representation, we are here concerned with two well established model principles, used often in the representation of biological systems, as follows:

2.4.1 Erdös- Rènyi Random Model

A *random graph* refers to the aleatory nature of the occurrence of edges in a given graph. Erdös and Rènyi proposed a graph model with N nodes and edges occurring between every pair of nodes with a probability $0 < p < 1$. As prescribed, a Erdös-Rènyi (ER) random network is only one outcome of the many realizations of one such network with N nodes. In practice, ER random networks are used as generalized frameworks for proving the behavior of network properties.

2.4.2 Geometric Random Graph Model

A *geometric random model* [65] is a graph $G(V, E)$ with the set of nodes V distributed uniformly at random in some metric space. An edge exists between two nodes u and v if the distance between them $d(u, v)$ is less than ϵ, for some constant ϵ and some appropriate norm.

2.5 Network Generation Process

A single network may serve very little when analyzing the statistical patterns involved in a phenomena. We define a *network generation process* as the mathematical or computational process by which multiple instances of a network model are realized. In this work, we establish the processes by which a modified geometric random graph series of networks and a ER-based system are generated.

2.6 Network Comparison

A basic analysis technique that can be immediately applied to a set of networks is to compare them. While comparisons can be carried out in a one-to-many cardinality, we are here concerned only with one-to-one network comparisons. Further, considering the many parameters that we could use to compare two candidate networks, such as qualitative assessments on their representation capacity, we here concern only with the mathematical topological comparisons of the graph constructs at hand. First, we define basic parameters for the assessment of similarity between nodes and edges, to then evaluate alternatives to the comparison of larger graph regions, and finally the comparison of full graphs.

2.6.1 Node Similarity

Given two nodes u and v, a basic graph theoretic question is the degree to which u and v are similar or different to each other. In the most basic case, Lorrain and White defined the *structural equivalence* of two nodes as the existance of identical relationship between each of them and any other node to which either of them is associated ([99], [54]).

Further, a non-identical pair of nodes can have a non-zero degree of similarity when node-level properties are compared. Thus, two nodes u and v could be deemed as highly similar according to their degree, eccentricity, clustering coefficient, or any of the measures of node centrality previously defined.

A recent important contribution to the problem of node similarity assessment is that of the use of a Graphlet Degree Vector (GDV) Similarity, by which a count of all possible graphlets in which a node participates is taken as the basis for comparison

with another node. Given that this measure encompasses a highly descriptive notion of the node's extended neighborhood, two nodes would have a high degree of similarity only if they were surrounded by highly similar neighborhoods, as described by the vector [58].

2.6.2 Subgraph Isomorphism Problem

A mapping $f : V_1 \rightarrow V_2$ is an *isomorphism* between G_1 and G_2 iff f is a bijection and $\forall$ u,v $\in$ $V_1, (u, v)$ $\in$ E_1 implies that $(f(u), f(v))$ $\in$ E_2 . Now, the subgraph isomorphism problem is that of assessing if a graph contains a subgraph which is an isomorphic copy of another graph. Whether induced or non-induced subgraphs, a solution to this problem has been proven to be NP-complete ([18], [30]).

2.7 Graph Alignment

A heuristic approximation to the problem of subgraph isomorphism problem, *graph alignment* seeks to find a good mapping between the nodes of two networks, such that the smaller network is fit into the larger as best as possible. However, an alignment may produce mappings that do not correspond to a full embedding of the networks.

While alignments can be carried out between two or more networks at the time, we here concern only with pairwise alignments between two networks. Further, an alignment can be generated by focusing exclusively on a particular area of the networks, or using the totality of the networks, as follows:

2.7.1 Local Alignment

Local alignments are pairwise network alignments that emphasize the mapping of regions in the networks to a high level of similarity, at the expense of the inclusion of other regions. Thus, local alignments often match small subnetworks from one network to one or more subnetworks in the other network. The ambiguity introduced by potential multiple pairings of a given subnetwork, in addition to the inability of local network alignments to identify large subgraphs across similar networks [45, 39, 11], give rise to the need for comprehensive approximations to the comparison of full graphs.

2.7.2 Global Alignment

When an alignment is carried out by using all of the nodes and edges of both networks being compared, this alignment is known as a *global alignment*. Global network alignment provides a unique, one-to-one mapping for every node in one network to exactly one node in the other network, even though this can lead to non-optimal matchings in some local regions.

2.7.3 Alignment Quality Metrics

Given the nature of network alignment as an approximation to the "best" fit between two networks, even if this leads to suboptimal mappings, metrics have been developed to evaluate the quality of such fit. In this work, we use *edge coverage (EC)* and the *Symmetric Substructure Score (S^3)*.

Edge Coverage

Edge Coverage (EC) is the ratio of edges aligned to the number of edges in a source network. Formally, given a mapping, h, between the edges in two networks, EC can be defined as,

$$EC = \frac{|h(E_1, E_2)|}{|E_1|} x100\%$$

.

Created by Kuchaiev et al [45], a high EC indicates that networks G and H are highly topologically similar. In particular, if EC =100%, then the second network, H, contains a subnetwork isomorphic to G. This isomorphic subnetwork, however, is not necessarily induced, since there can exist additional edges in H that do not exist in G among the corresponding sets of nodes.

Symmetric Substructure Score (S^3)

A more recent alignment quality metric, the *Symmetric Substructure Score* (S^3) penalizes both alignments that map denser network regions to sparser ones and alignments that map sparser network regions to denser ones [80]. Consequently, S^3 measures the balance of local topologies in G_1 and G_2 with the global quality of the alignment. Formally, S^3 is defined as,

$$S^3 = \frac{|h(E_1, E_2)|}{|E_1| + |E(G_2[f(V_1)])| - |f(V_1)|} \times 100\%$$

Consequently, the S^3 metric of an alignment is 100% if and only if f is a perfect embedding.

Chapter 3

Related Work

Network analysis has evolved considerably from the earlier, graph-theoretical analysis on static networks pioneered by Girvan [32], Newman [62], and Barabasi ([7], [8]). Today, network representations are ubiquitous in science and engineering. Thus, representation of complex and evolving systems creates the need to characterize the different aspects of such dynamics. Dynamics aspects to be analyzed can be defined at many levels; suitable for analysis include rates of change, nature of the phenomena, direction and consequences, as well as causality factors. Today, while much progress has been made in the area of network dynamics analysis, much of it remains subject-specific, and largely based on a granularity that often masks the topological features that are common to all network abstractions.

In this chapter we provide the reader with commentaries on the state of the art in network dynamics analysis, to the best of our knowledge. Further, after an initial overview of the general state of the field, we focus on the best current approaches to the analysis of dynamics in the biological arena.

3.1 State of the Art in the Analysis of Network Dynamics

3.1.1 Domain-Specificity

Networks are a convenient representation for a wide range of phenomena. However, most of network analysis has developed to address the specific needs of given fields, and therefore such techniques are not immediately transferable across fields.

Further, the analysis of networks and their dynamics has largely been facilitated by our ability to acquire data on a variety of complex networks. Particularly, the social sciences led the creation of techniques that allow for the assessment of change in a variety of contexts. While sometimes qualitative in nature, these studies provide the first steps towards the methodical study of network dynamics.

For example, Orsenigo et al [51] analyze the impact of various industry factors on the structure of pharmaceutical organizations. In this study, the authors map graph-theoretical aspects of the network that change in various ways in response to discountinuities in the technologies they relate to.

Similarly, Powell [67] addresses the interplay between structural properties in life science professional networks. Career and technological events provide context for the understanding of network effects such as cohesiveness, motiff occurrence, and pathway selection.

Further De Menezes et al [23] examines the fluctuations in five different natural and technological networks, measuring the flux behavior of different network elements, and concluding that all of their node-level behaviour obeys a unique scaling law. Such regulating laws, the author argues, can explain the balance between internal

and external factors, such that parameters of these regulating laws can be infered.

Now, in a classic study on epidemiological applications of network analysis, Rothenberg [78] studies the changes in properties of network configurations for individuals at hight risk of HIV infection, as they inhabit their natural environment. Such observational analysis couples the researcher's ability to elicit a longitudinal analysis on fixed-node networks, and evaluate the change in local and global network properties over time. In a different epidemiology study, Read [77] introduces a numerical approach to the empirical study of how dynamic networks influence disease spread. Read asserts that the observed phenomena is suitable for modeling via a random mixing model.

Control theory is also applied to the analysis of networks in many instances. Fries [28], for example, uses a continuous time optimal control problem to study dynamic network traffic assignment. In this sudy, the author contrasts two different approaches on single mode traffic networks, ad frame the problem in a rigorous mathematical optimization setup.

Moreover, in Kuhn's article [47], the concepts of traffic and networks are further applied to computational networks with a variety of rates of change. Algorithmic is developed for the modeling and allocation of distribution of information on changing topologies, and complexity analysis for different cases is carried out.

Further, visualization is among the broadest and more useful network analysis tools. However, the challenges it presents are magnified when dealing with complex networks. In his article, Liao et al [51] introduce ENAVis (Enterprise Network Activities Visualization), a multi-level graph visualization tool, customized for enterprise network visualization and arguably superior the single layer software tools available. The concept of hierarchical visualization starts with the evolution of inter-graph states,

adapts to the visualization of intra-graph clustering, and concludes with the visualization of similarity between individual nodes.

3.1.2 Numerical Methodologies

Now, while network analysis is inextricably tied to the mathematical basis of graph theory, there the application of statistical models has provided the field with valuable tools. For example, Snijders et al [85] introduces a statistical models for evaluating dynamic social networks. Continuous-time Markov models are implemented with parameters estimated from observed data, and tested by methods based on the method of moments. Thus, network evolution is modeled as a sequence of constrained steps in which exogenous and endogenous covariates play a role. Further, in a subsequent publication, Snijder [86] extends the methodology to a more general approach by which both node-based changes, and node-initiated changes can be modeled under the stochastic framework.

Adding a spatial dimension to the complexity of network analysis is a significant challenge. In his article, Benda [10] addresses the spatial and temporal dynamics of river networks, whose hierarchical and spatially distributed branching are modeled as networks with very specific perturbations. Disturbances such as watersheds and basins impact the structural conformation of the network over time.

Another approach to the analysis of networks in a discipline-independent manner is that of creating suitable grammars for the representation of network constructs. Thus, properties can be seemingly imposed on the networks at hand, and simulations and analysis are also readily executed. One such methodology was developed by Shapiro and Mjolsness [82], and has found suitable application in the inference of biological regulatory networks, as well as in the modeling of related processes. Another formal

approach, albeit older, was SHIFT, a programming language for the specification and simulation of dynamic networks introduced by Deshpandel [24] in 1996. Continuous time phases, separated by discrete event transitions can be modeled within a component's behavior. Further, single or multiple components may co-evolve so that dynamics are readily observable.

3.1.3 Community-Level Analyses

Now, while several studies address changes at various levels of granularity, most of the advancement in the analysis of dynamics in complex networks has been achieved by focusing on sets of nodes as the observable units, rather than individual nodes or edges. For example, in [91] the authors provide a framework for identifying densely clustered subsets of dynamic social networks. This is, identifying *communities* that change over time. Using dynamic programing, the algorithms provided approximate solutions to the NP-hard problem of finding the most explanatory community structure. Another example is the article by Rand [76], where an experimental study examining how dynamic social networks maintain cooperation in groups is presented. Time dependencies are explored to evaluate the effects of latency on the selective creation and dissolution of ties (edges) within dynamic communities. Further, time dependencies in multi-mode networks are analyzed by Tang et al [90], utilizing a spectral framework. Thus, actor membership and community identification are addressed. A different community-matching strategy across timepoints is presented by Greene [33].

Now, Lin et al [52], present an algorithm, *Facenet*, for analyzing evolutions of social network data using non-negative matrix factorization. this approach, the author claims, improves on previous methodology that first identifies the communities in a

social network, and progressively compare to determine correspondences. Facenet assumes smoothness of temporal evolution, and regularizes the identification of networks by factorization their matrix equivalents. Later, in a subsequent publication [53], Lin introduces a maximum a posteriori (MAP) estimation approach to be applied on current and historic community data, so as to optimize the identification and monitoring of evolution.

An improvement over earlier community-based analysis methods was introduced in 2009 by Kim et al [41]. In their article, the authors present a particle-and-density evolutionary clustering method to detect communities that are changing over time. Such methodology improves upon the previous evolutionary clustering methods, which usually adopt the temporal smoothness framework. Such framework has a desirable feature of controlling the balance between temporal noise and true concept drift of communities, while constraining the emergence and dissolution of communities, and limiting them to a fan assumed fixed number. Thus, forming, evolving, and dissolving communities are identified.

3.1.4 Online Stream Analysis

Now, most of the analysis in dynamic complex networks is carried out in an offline, observational manner. A major contribution was provided by Leskovec et al [49] when establishing the shrinking diameter phenomena on time-evolving graphs. Further, Sun et al [89] incrementally summarize tensor streams (high-order graph streams) as smaller core tensor streams and projection matrices. However, in order to solve the need for user specified parameters and the lossy compression, Sun et al [88] introduced GraphScope, a method for online detection of communities in dynamic networks in a stream of large and complex social networks. Not only GraphScope

is non-parametric, but it effectively and efficiently identifies actors and communities involved in communication behavior in social networks.

Lastly, Chakrabarti [16] led the advancement in the application of clustering algorithms to stream data sets, by introducing *evolutionaty clustering*, emphasizing the balance between stability of prediction and discovery of change. Thus, evolutionary versions of k-means and agglomerative hierarchical clustering.

3.1.5 Event-Based Community Analysis

Now, changes in communities and graph regions in general are often studied in a generalized manner that does not particularly identify the types of changes occurred, or otherwise characterize them. However, progress has been made in this area over the years. The seminal paper by Samtaney et al [79] described an approach for extracting coherent regions from 2-dimensional and 3-dimensional scalar and vector fields for tracking purposes. To study the evolution of these regions over time, they present certain evolutionary events for objects. Further, event-based methods have also been applied on spatial data [102] and clustered stream data [87]. However, although they use events, they do not deal with evolving graphs.

To address the analysis of events in graph analysis, Asur et al [4] present an event-based characterization of behavioral patterns for temporally varying graphs. In this article, the authors provide an event-based framework for characterizing the evolution of interaction networks. They begin by converting an evolving graph into static snapshot graphs at different time points, and define 4 types of events on communities, across subsequent timeframes, and 4 additional events on individuals, as they relate to communities in the network. Further, when semantic similarity is incorporated into the analysis of events, group splits and merges are defined and evaluated along

with the initial event type set.

As it stands today, event-based network analysis methodologies do not address individual node or edge events, largely due to the lack of appropriate granularity in the involved computational technology. Consequently, our alignment-based approach stands a unique addition to the field, in that single node/edge events can be identified, and subsequently characterized.

3.2 Analysis of Dynamics in Biological Networks

Network analysis has played an important role in the context of the biological sciences. Relationships between biological entities have been characterized in terms of network constructs that represent a variety of real-world interactions, from social, to chemical, to structural and biophysical. Building on the advances of network analysis in other disciplines, and adding in return important methodology and tools, biologists deploy network analysis to understand phenomena in fields such as ecology, cellular, molecular, and neural biology. Below we outline some of the most relevant work as it relates to the study of biological networks and their associated dynamics.

3.2.1 From Static to Dynamics in Network Systems Biology

Just as occurred throughout the evolution of general network analysis, network biology has sustained significant progress and is now capable of addressing the analysis of the dynamics involved in naturally changing systems. Such networks might be constructed from biological entities of the same type or, as Barabasi et al. posit, from diverse elements in the different biological ecosystems [8]. As Hiroaki Kitano states, understanding the system-level behavior of phenomena, in addition to the unit-level

events, can usher breakthrough discoveries that carry great potential [43].

Further, similar to the shift from analysis of static network characteristics to that of dynamics in network systems, systems and computational biologists have now focused on the global inference and analysis of the dynamic interactome. Such an area of research, as pointed out by Przytycka et al. [72] is prompted and ushered by large-scale cellular network information, as well as by other high-throughput experimental data, with the catalyzing effect of rapidly evolving computational and mathematical methods and tools.

3.2.2 Modeling and Tools

Methods and tools developed specifically for the analysis of biological networks have been critical to the empowerment of biological researchers. Mathematical tools such as those developed by Selimkhanov et al. for the inference of network dynamics occurred within a signal-regulated kinase, calcium, and nuclear factor kappa-B pathway [81] are critical to the understanding of multi-level events. Further, as mathematical and computational tools are deployed for the inference and understanding of network structures representing biological entities, as in the case of gene regulatory network inference, fine-tuning of such tools helps in the evolution of the field as a whole, for such methods can be used in a variety of disciplines outside of the life sciences. For example, the use of dynamic Bayesian networks to simulate genetic networks, studied by Dirk Husmeier in [36], and Kim in [42] can also be used for the analysis of stock networks or similar economic entities.

Further, tools and algorithms have been created to implement visualization and manipulation features available for other types of data, yet specializing in the particular needs of the life scientist. One such computational application is that created by

You et al. to help in the overall study of dynamic networks via a host of simulation capabilities [103]. One other set of algorithms is that presented by Kikuchi et al, in which estimation of network characteristics inferred from scarce time-series data, such as that of microarray experiments, can be optimized and extended by simulation methods [40].

Moreover, given the complex nature of biological phenomena, and while a connective theoretical foundation that can capture all such levels remains as an open problem, methods and tools are consistently created to aid in the focused understanding of a particular area. For example, Phair et al combine modeling techniques similar to those mentioned before with in vivo microscopy techniques that help in the elucidation and analysis of network constructs representative of the phenomena implicated in protein binding [66]. Similarly, Dalle Pezze et al demonstrate the utilization of network analysis techniques for the understanding of a particular protein, its processes and interactions [20]. A final example of the utilization of network theory to address the complexity between biological levels is found in the work of Jordi Bascompte, by which he tackles the understanding of structure and dynamics of ecological (animal antagonistic and mutualistic) networks, not previously addressed under pure observational, static-oriented, methodologies [9].

3.2.3 Network Analysis in Genomics

While most of the network analysis aspects of genomics are related to the inference of the edges between genes, often based on experimental data such as microarray assays ([6], [29], [35], [98], among many others), there has been significant progress on the exploration of dynamics within the participating genes. For example, in work carried out by Prill et al., the relationship between dynamic network properties, such as

stability and robustness to small perturbations, was shown to be linked to the relative abundance of network motifs [68]. A step further, Luscombe et al. integrate multi-level data (in this case genomic and transciptional), for the elucidation of genomic relationships. These analyses combined well-known topological measures, local motifs and newly derived statistics [55].

3.2.4 Analysis of Protein-Protein Interaction Networks

At the interactome level, network analysis has both followed and influenced the larger evolution in network science. As stated by Tyson et al. in [94], a new breed of theoretical molecular biologists must reproduce the networks in computers and in mathematical language of dynamical systems.

Thus, starting from analysis of static characteristics such as the nature of hubs in dynamic protein-protein interaction networks [34], to the inception of network alignment as a tool for comparison between the interactome of multiple species ([46], [45], [69], [44], among others), network analysis and graph-theoretical concepts are fundamental to the understanding and utilization of the interactome.

Naturally, the use of network alignment for the interactome analysis has also been coupled with experimental methods, in efforts to enhance their individual capability. One such strategy for aligning protein-protein interaction network with an experimental addition was developed by Kelley et al.[38]. Further, global and local alignment methods have been created to optimize the sensitivity in identifying conserved regions across species [3].

Finally, a valuable source for information on the network aspects of analysis of dynamics in proteomic is found in [92], in which Terentiev provides relevant guidelines

to successfully describe, summarize, and analyze proteomic data.

3.2.5 Network Analysis in Molecular Biology

Stepping down the abstraction ladder, molecular biology represents the interface between the biological sciences and physics. At this level, network analysis has played an important, yet nascent role, specially as it relates to the dynamics associated with molecular structure and behabior. Alexander et al., for example, explores the validity of network-related concepts such as modularity, in the context of molecular networks. In [2], Alexander et al. explore the importance of dynamics in understanding modularity of molecular networks, concluding that a dynamic perspective is essential to grouping molecules into modules and determining their collective function.

Further, as molecular networks and other biophysical constructs interact, regulating and providing feedback onto the behaviors of multi-scale elements. For example, Akhmanova et al. explore the interplay between microtubules plus-end tracking proteins for the development and regulation of such physical networks [1].

3.2.6 Analysis of Neural Networks

Finally in this section, we take a look at the impact of network science on one of the most complex physical systems found in the biological sciences, that of neural networks. To start, we suggest to the reader a very comprehensive review by Vogel et al., dedicated to the analysis of dynamics in neural networks [97].

Further, given the many complex problems the brain must concurrently solve, the comprehensive and connective study of its architecture and function is still evolving. However, progress has been made in specific subareas such as that of olfactory net-

works, in which studies like that of Laurent et al. [48], explore the structural aspects of these physical networks, in combination with theoretical network analysis aspects. Moreover, the addition of experimental techniques to enhance computational and mathematical methods, encountered in other areas of network biology, is also seen in the study of neural networks. For example, Dayan et al. explore the possible combination of non-invasive stimulation techniques to characterize and probe the information obtained from imaging techniques, as well as from graph-theoretical notions of the brain's structure [21].

Finally, as the field evolves, a number of specific tools and methodologies bring about the potential for a revolution in the understanding of the biological neural networks. As such, work by Bullmore et al. [14], Zadeh [104], and De Haan [22] will be extremely influential. Not only we will be able to understand the structure-function relationship in our neural system, but could potentially intervene to alleviate undesirable outcomes such as age-related diseases, or pain [5].

Chapter 4

D-GRAAL: A Fast Dijkstra-based Graph Alignment Algorithm

4.1 Introduction

There exist a variety of algorithms for graph alignment, as outlined in our introductory chapters. Among them, local network alignments target only pairwise alignments [26, 38, 11, 50], thus matching a small subgraph from one network to one or more subgraphs in the other network. The ambiguity introduced by potential multiple pairings of a given subgraph, in addition to the inability of local network alignments to identify large subgraphs across similar networks [45, 39, 11], give rise to the need for comprehensive approximations to the comparison of full graphs.

Consequently, considering the necessity to preserve a balance between alignment of local regions and a global mapping between two networks, we concentrate our attention on the exploration of global network alignment algorithms. Specifically, we describe and evaluate two of the most established graph alignment algorithms, GRAAL, and

MI-GRAAL.

We thus evaluate the strengths of these relevant alignment algorithms, deployed on a suitable environment. Thus, we observe the algorithms GRAAL and MI-GRAAL, paying special attention to their computational efficiency performance, as well as the quality of their alignments under various network complexities.

Our evaluation of these algorithms reveals a loss of computational performance when processing large, complex graphs, a non-desirable feature for the analysis of dynamics in real-time, complex analysis tasks. Consequently, we devise a novel algorithm, D-GRAAL, that outperforms these algorithms in speed while preserving comparable alignment quality.

We now provide the reader with a brief description of the algorithms at hand, as well as a summary of Dijkstra's algorithm, as a basis to describe in detail our graph alignment algorithm. Further, we provide a detailed recount of our comparative results and an assessment on the advantages and challenges of our approach. Finally, we provide concluding remarks regarding the future improvements and potential application problems.

4.2 Background

4.2.1 GRAAL

The first algorithm we evaluate is the Graph Aligner algorithm, GRAAL, [45], which uses a cost function that relies on the graphlet degree vector (GDV) similarity measure, a highly constraining measure of topological similarity between nodes' extended network neighborhoods. Below is a summarized algorithm for the GRAAL algorithm.

Notice that the *power* of a graph is the power of its associated matrix representation. As such, a path of length k in the original matrix will be an edge in the matrix exponentiated to the k power.

- Initialize alignment to empty set
- While $\exists$ nodes not aligned in G_1
- Initialize power to 1
  - Find seed (v_1, v_2)
  - Add (v_1, v_2) in alignment
  - Initialize $size = 1$ and $radius = 1$
  - While $size \neq 0$
    - Get $S_1 = \{v_1$ neighbors in V_1, within $radius\}$
    - Get $S_2 = \{v_2$ neighbors in V_2, within $radius\}$
    - If S_1 and S_2 are non-empty, greedily align them
    - Increase radius
  - If $r \geq 3$ and $power < 3$, raise graphs to next power

4.2.2 MI-GRAAL

Another member of the GRAAL family of global alignment algorithms, MI-GRAAL [46], designed for the alignment of static protein interaction (PPI) networks, allows for the addition of multiple node similarity metrics, while combining the alignment strategies developed in earlier GRAAL algorithms. Below we find a summarized algorithm for MI-GRAAL.

- Place (v_1, v_2) pairs in priority queue based on similarity
- Initialize alignment to empty set
- While $\exists$ nodes not aligned in G_1
  - Find seed (v_1, v_2) in priority queue
  - add (v_1, v_2) to alignment
  - for k in $\{1, ..., min(ecc(v_1), ecc(v_2))\}$
    - Construct k^{th} neighborhood of v_1 in G_1
    - Construct k^{th} neighborhood of v_2 in G_2
    - construct BP, a bipartite graph with only nodes that have aligned neighbors
    - set alignment confidence as cost of nodes
    - Solve the *Maximum Weight Bipartite Matching Problem* for BP
  - If there are unaligned nodes, raise both graphs to next power

4.2.3 Dijkstra's Shortest Path Algorithm

Dutch computer scientist, Dr. Edsger Dijkstra, the recipient of the 1972 A.M Turing Award, created the famed algorithm as a solution to the problem of finding the shortest path between a source node and all other nodes in a graph. Simply expressed, the algorithm can be thought as a greedy algorithm that accounts for a non-negative distance metric attributed to the edge between two nodes, and progressively identifies the shortest path amongst the nodes not yet visited and the pool of nodes and routes encountered so far.

The following is a pseudo-code description of the algorithm:

Shortest-Path(G, source)

- Initialize source path cost (distance) to 0
- Initialize $path = empty$
- Initialize all other vertices to $dist = INF$
- Add all nodes to priority queue
- While queue is not empty
  - Extract node u, with minimum distance
  - for each neighbor, v, of u:
    - $alt = dist(u) + length(u, v)$
    - If $alt < dist[v]$, make u v's predecessor, and update v's distance

4.3 Methods

4.3.1 D-GRAAL Algorithm

Our algorithm, D-GRAAL, was ispired on Dr. Dijskstra's solution to the shortest path algorithm problem, yet applied to the task of finding a mapping between two networks such that maximal similarity across edges is approximated. Our algorithm, D-GRAAL, is akin to Dijkstra's Shortest Path Algorithm, in that minimal difference is sought between two edges belonging to each of the comparing networks. Consequently, one could imagine a resulting graph structure, here named as a *Product Graph*, *PG*, given the similarity between the nodes of two different graphs, *G1* and *G2*. This is, the more similar two nodes, u and v, belonging to graphs *G1* and *G2* respectively, are

found to be, as per the effective similarity metric applied, the shorter the "distance" or "cost" will be. Figure 4.1 provides an illustration of a node in one such graph.

As a consequence of node aggregation process, the largest contiguous set of product nodes will be included in the product graph. We call such subgraphs Locally Connected Components (LCC).

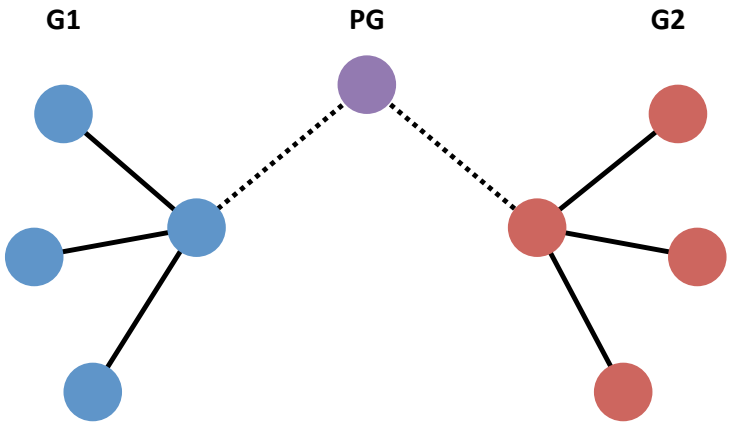


Figure 4.1: Schematic representation of a product node's composition from G1 and G2

The following is a pseudo-code description of the algorithm:

- Given G1 and G2
- For each $(v_1 \in E_1, v_2 \in E_2)$ seed, if not aligned
  - Place in priority queue, Q, with distance set to INF
  - While Q is not empty
    - Pop u from Q
    - for each neighbor pair, v, of u, if not aligned:

* $alt = cost(u) + distance(u, v)$
* If $alt < dist[v]$, update v's distance and push back in Q

Implementation

Our algorithm was implemented in C++, and deployed on a variety of systems. The core of the code is implemented as a class, *D-GRAAL*, which takes as input 2 networks expressed in LEDA format [57]. Further, D-GRAAL also takes a similarity file, in which each row is a node from the first network, followed by a node from the second network, and then by a similarity score. Now, for every pair of nodes between the networks, a search is conducted in an analogous manner to that of Dijkstra's shortest path algorithm to establish what neighboring pair of nodes will yield the least difference in an expanding fashion. Consequently, the process can be thought as if building a *product graph* where each *product node* corresponds to a pair from the comparing networks. At the end of the process, a priority queue that was iteratively populated to contain the possible neighbor pairs with respect to the product node at hand is depleted, and the process will have identified the largest possible set of pairings that approximates the maximum similarity scores across the networks.

4.3.2 Validation

Our algorithm was tested under different architectures, both under Linux and OSX. Comparisons against the relevant algorithms GRAAL and Mi-GRAAL were performed to assess D-GRAAL's alignment quality as well as its computational performance on a variety of data sets, as follows:

We perform alignment comparisons across networks generated from two graph genera-

tion processes (GNP), *Erdos-Renyi Random Graphs* and *Geometric Random Graphs*, outlined in our introductory chapters. Further, we compare our results across a range of network conformations ranging from sparse to highly dense networks, evolved over a number of steps to induce fixed, yet differing, similarity indexes from which quality of alignment can be evaluated. A measure of such similarity change across networks evolved from the same graph, *G0*, is called the network generating process' *Delta Change* or DC.

Thus, we produced two sets of benchmarks as follows:

Erdos-Renyi Random Graph Generation Process

A benchmark set was developed from networks created using an Erdos-Renyi random graph process, such that for each graph, $G0$, networks G1, G2,..., G_k are evolved in such a way that for each G_i and G_{i+1} graphs, there exist a DC percentage in nodes changed. For every network $G0$, a number of subsequent networks were evolved by permutating a *Delta Change* rate of edges, such that no edges repeat across the set of subsequent networks. Figure 4.2 provides a graphical illustration of ER network generation.

A total of 1670 graphs were generated, thus allowing for 1500 alignments across networks of similar characteristics.

Thus, the following network sets were used for comparison of the different alignment algorithms:

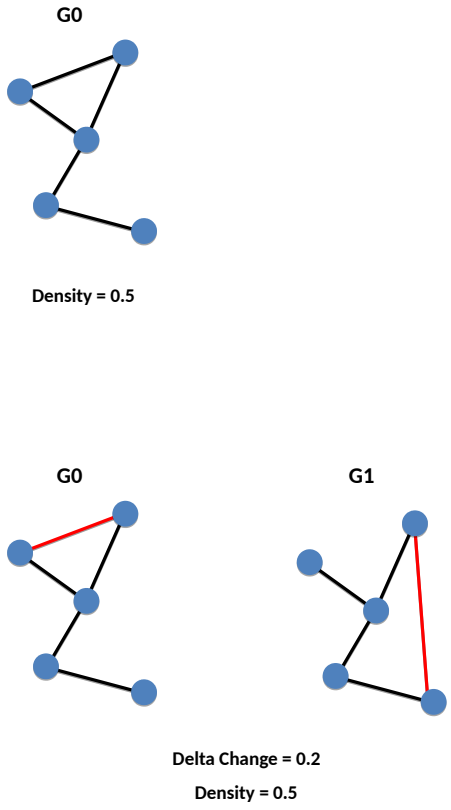


Figure 4.2: Erdos-Renyi Random Graph Generation Illustration

Order/Density	5%	10%	25%	50%
50	-1%	All	All	All
100	All	All	All	All
200	All	All	All	All
400	All	All	All	All
800	All	All	All	All

Table 4.1: Erdos-Renyi GNP. Delta Change Explored (All = [1%, 5%, 10%, 25%])

Geometric Random Graph Generation Process

A geometric graph generation process was created such that for each geometric network $G0$ initially created, a subsequent network $G1$ is created by "merging" *Delta-Change* nodes to their closest neighbor. This is, every edge incident to the node will

now be incident to the node's closest neighbor, and then the node will be removed from the network. The process continues for up to 50 nodes in the series, or until no new nodes can be merged, as it may occur when there are a few nodes to begin.

Figures 4.3, 4.4, and 4.5 provide a graphical illustration of ER network generation.

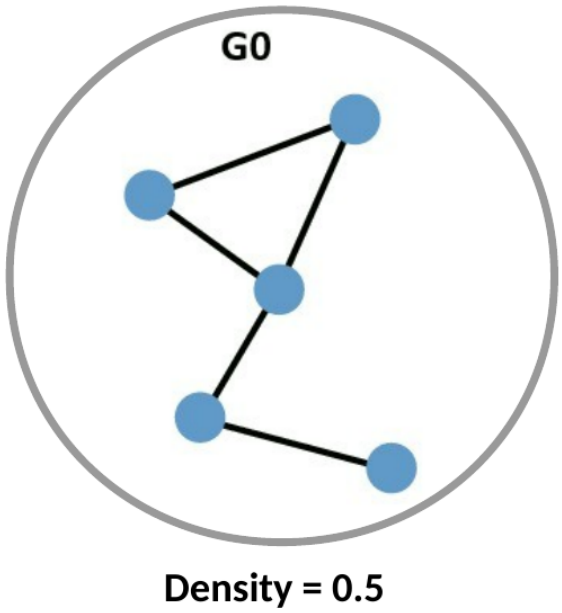


Figure 4.3: Initial geometric network (G0)

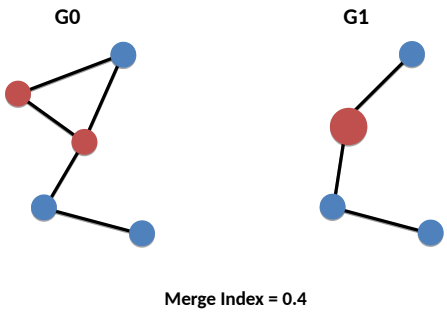


Figure 4.4: Node merge illustration

The following table describes the initial densities and fusion rates applied to our benchmark sets:

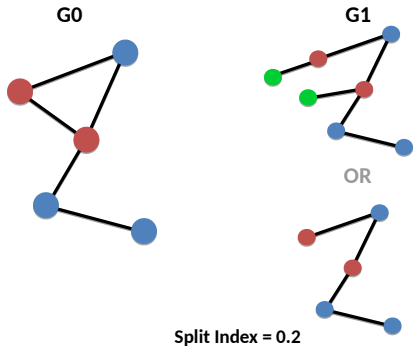


Figure 4.5: Illustration of edge split scenarios

Order/Density	5%	10%	25%	50%
20	-1%	All	All	All
40	All	All	All	All
60	All	All	All	All
80	All	All	All	All
100	All	All	All	All

Table 4.2: Delta Change (All = [1%, 5%, 10%, 25%])

Yeast PPI vs Human PPI: A biological sample

In addition, we compare network performance on the alignment of two well known protein-protein interaction networks; the high-confidence yeast *S. cerevisiae* PPI network [17], and the high-confidence human PPI network [73], henceforth denoted as yeast and human respectively. The yeast network has 16,127 interactions amongst 2,390 proteins and the human network has 41,456 interactions amongst 9,141 proteins. These networks have been used as benchmark sets for comparison alignments in various studies [45][46].

4.4 Results

The results obtained on the different testbeds reveal D-GRAAL's overall preservation of alignment quality with respect to that of the other algorithms, while performance is consistently improved for all networks evaluated, especially as their complexity increases. A detailed breakdown of analysis results is here presented:

4.4.1 Computational Performance

First, we evaluate the runtime computational performance between the different algorithms. To do so, we deployed suitable implementations in the same Linux systems, and proceeded to time the execution of all subsequent networks in a given testbed, by means of the *time* bash function. Further, individual results were aggregated into meaningful statistics that would allow for the comparison of relevant performance.

As a result, we can observe in figures 4.6 and 4.7 how D-GRAAL's overall performance is vastly superior to the other algorithms, especially at larger and more complex network alignments. Such dominance, however, is not dependent on the amount of difference between each of the networks being aligned, as denoted when compared against the Delta Change metric, as displayed in figure 4.8. These results are also observed on the biological alignment between human and yeast protein interaction networks 4.3.

4.4.2 Alignment Quality

Further, we compare D-GRAAL against GRAAL and MI-GRAAL algorithms to evaluate the comparative quality of resulting alignments, as expressed by the metrics Edge

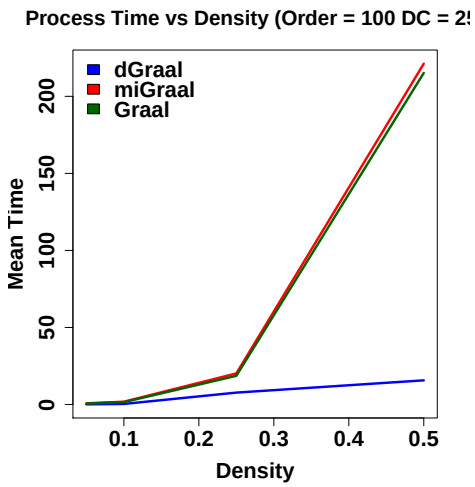


Figure 4.6: Computational Performance with respect to Density on Erdos-Renyi Networks

Table 4.3: Yeast to Human Alignment

Algorithm	Total Time
D-GRAAL	37 : 35
miGRAAL	2 : 40 : 21
GRAAL	1 : 59 : 03

	Nodes	Edges	Density
Yeast	2390	16127	0.005
Human	9141	41456	0.001

Coverave (EC), ICS, and S3.

Edge Coverage

As observed on figures 4.9, 4.10, and 4.11, EC is comparatively at the same level as that of the other algorithms. These results are also encountered when aligning the biological protein interaction networks of yeast and human 4.4. Further, an improvement in EC performance can be observed for larger and more complex networks, providing further evidence for D-GRAAL's comparative advantage on more complex alignments.

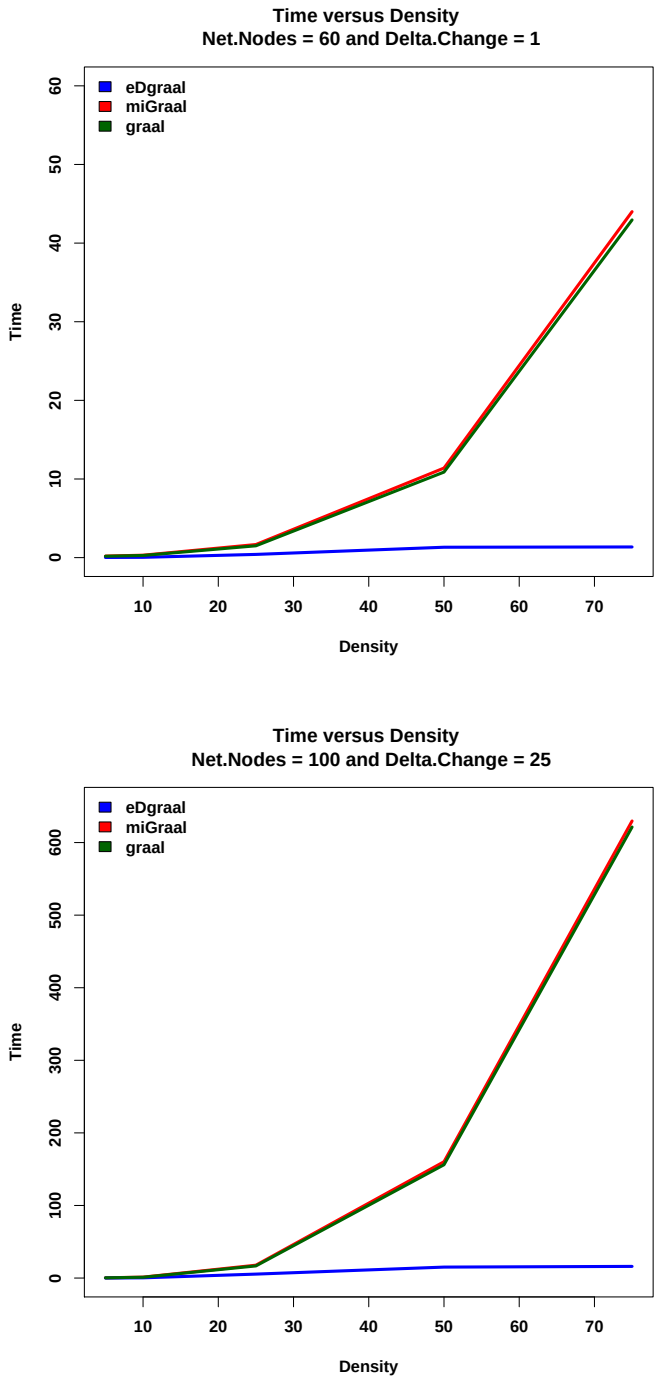


Figure 4.7: Computational Performance with respect to density on Geo Networks

ICS

Simlarly, the ICS measure, described in our introductory chapters, allows us to compare the structural quality of the resulting alignment. As displayed in figures 4.13

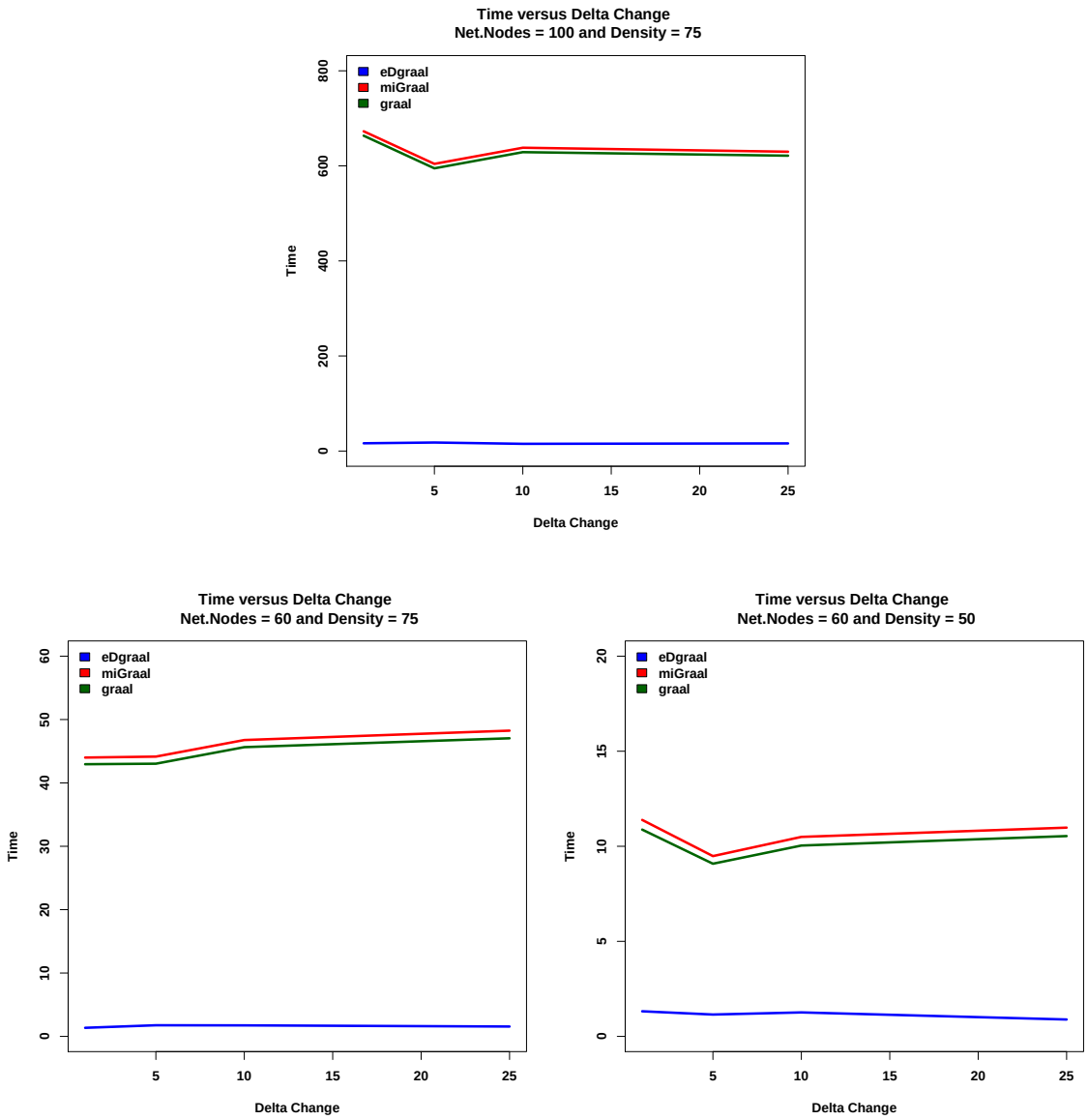


Figure 4.8: Computational Performance with respect to Delta Change

and 4.12, D-GRAAL produces alignments of comparatively the same quality as the other algorithms, with the exception of MI-GRAAL, though seems to improve with respect to every other algorithm when alignments are obtained for larger and more complex networks, as expressed by order and density. Further, the quality of resulting alignments is minimally influenced by the amount of difference between aligned networks, as expressed in the Delta Change metric.

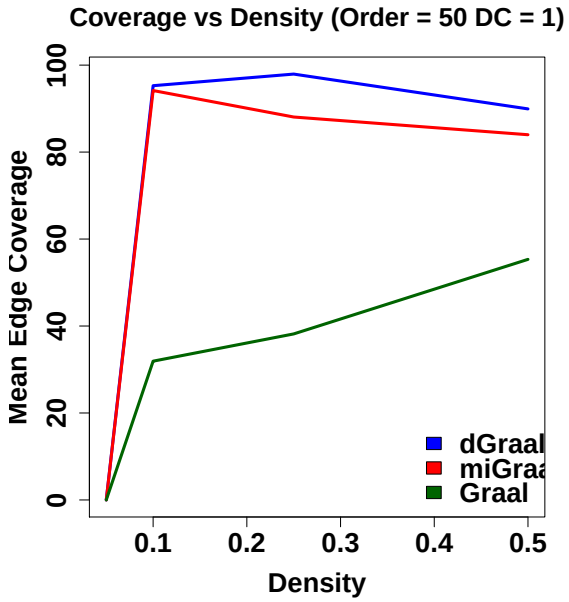


Figure 4.9: Edge Coverage with respect to Density on Erdos-Renyi Networks

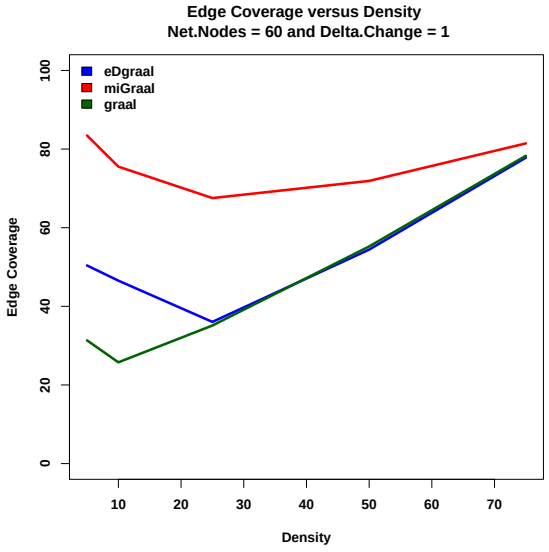


Figure 4.10: Edge Coverage Quality with respect to Density on Geo Networks

S3

Finally, we compare the quality of our resulting alignments in terms of *S3*, the quality metric introduced along with MAGNA, and described in our introductory chapters. As observed in figures 4.14 and 4.15, D-GRAAL produces alignments that while less

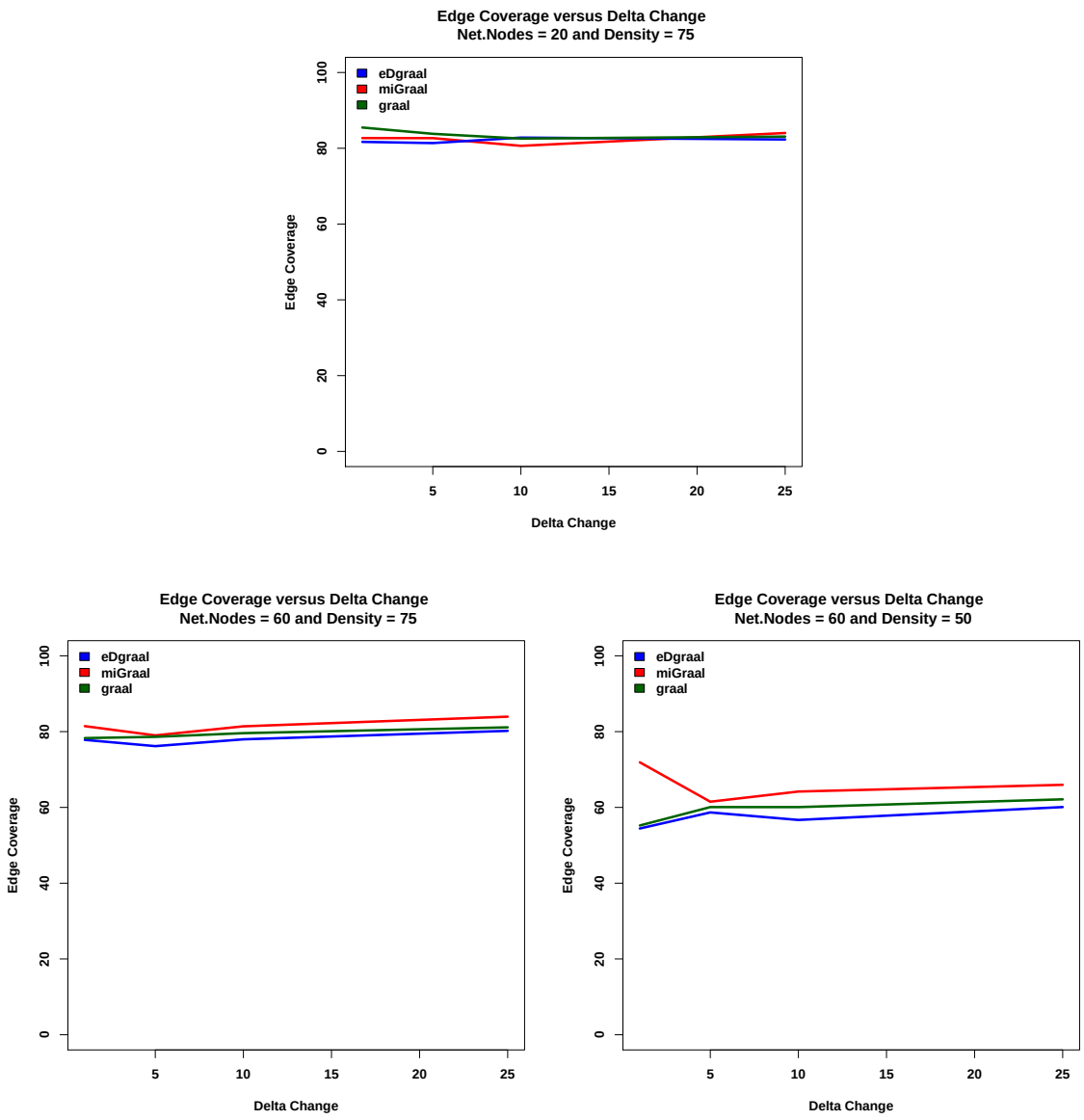


Figure 4.11: Change in EC metric with respect to Delta Value on Geo Networks

favorable for small networks than those obtained with MI-GRAAL, improve to an equal standing as aligned networks become larger and more complex. Further, such behaviors do not seem to be influenced by the amount of difference (Delta Change) between the aligned networks.

Table 4.4: Yeast to Human Alignment

Algorithm	Edge Coverage (N1)
D-GRAAL	22.35
miGRAAL	14.43
GRAAL	5.48769

	Nodes	Edges	Density
Yeast	2390	16127	0.005
Human	9141	41456	0.001

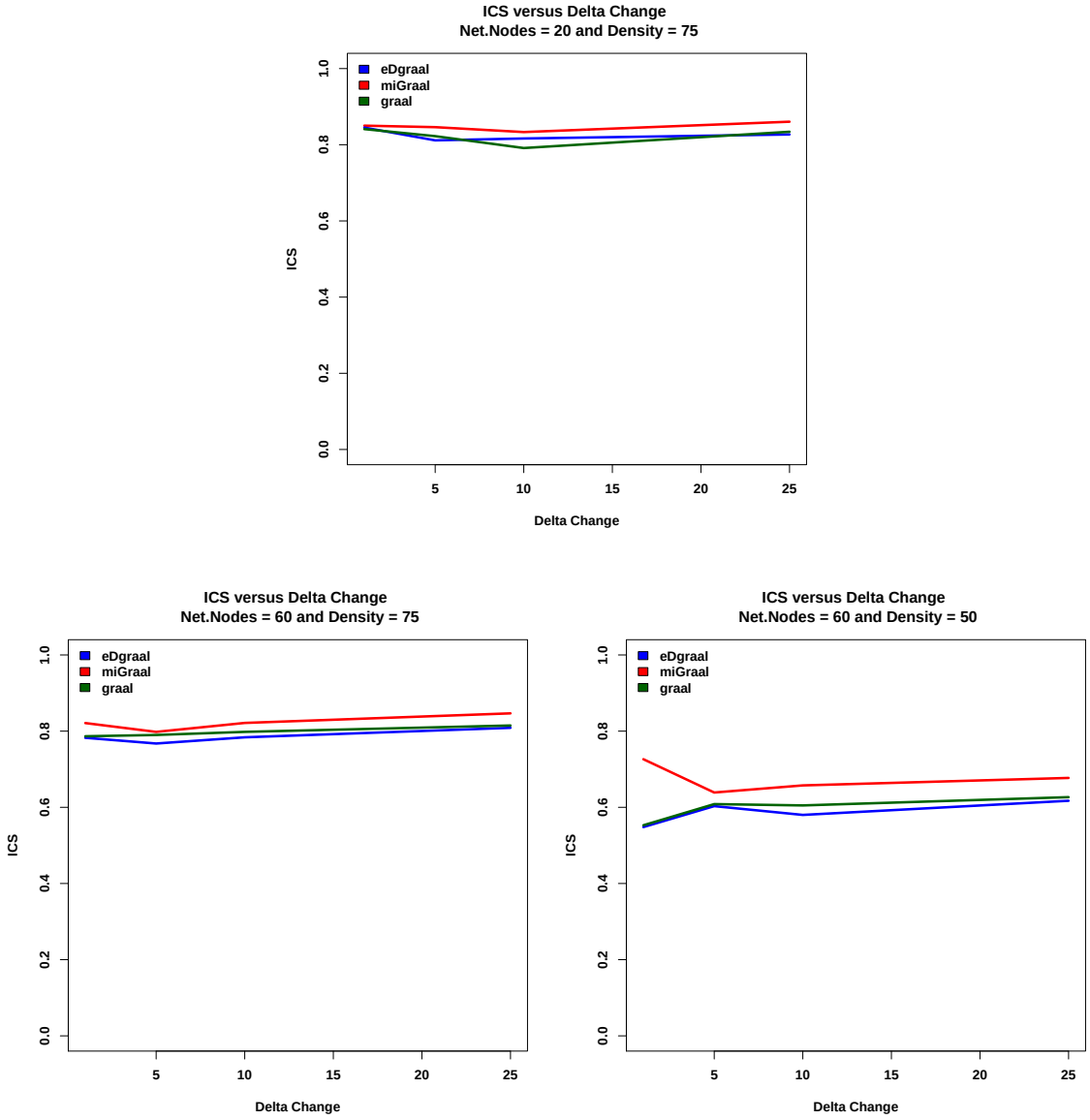


Figure 4.12: Change in ICS metric with respect to Delta Change on Geo Networks

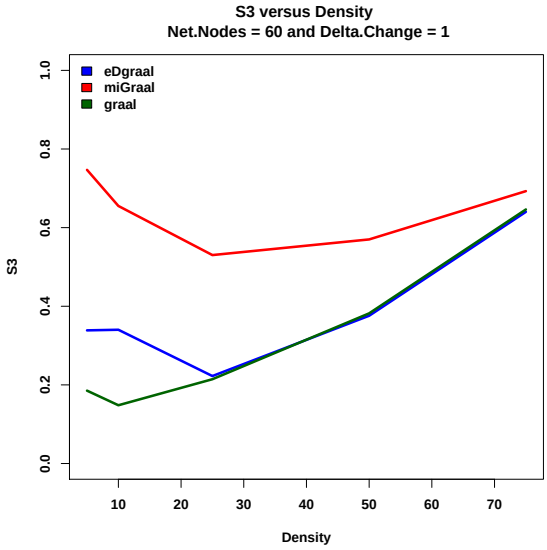


Figure 4.13: Change in S3 metric with respect to Density Change on Geo Networks

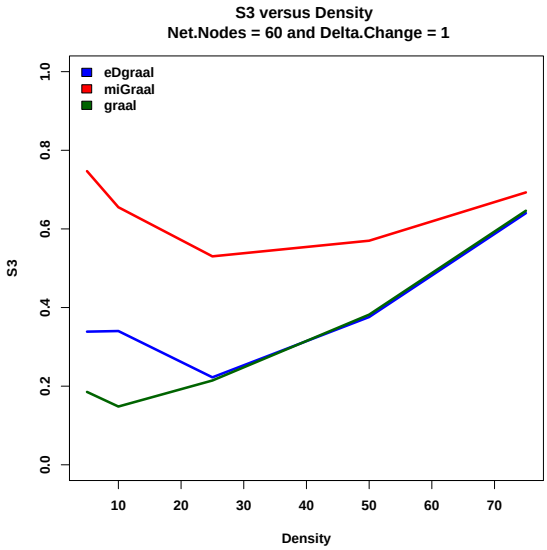


Figure 4.14: Change in S3 quality with respect to Density

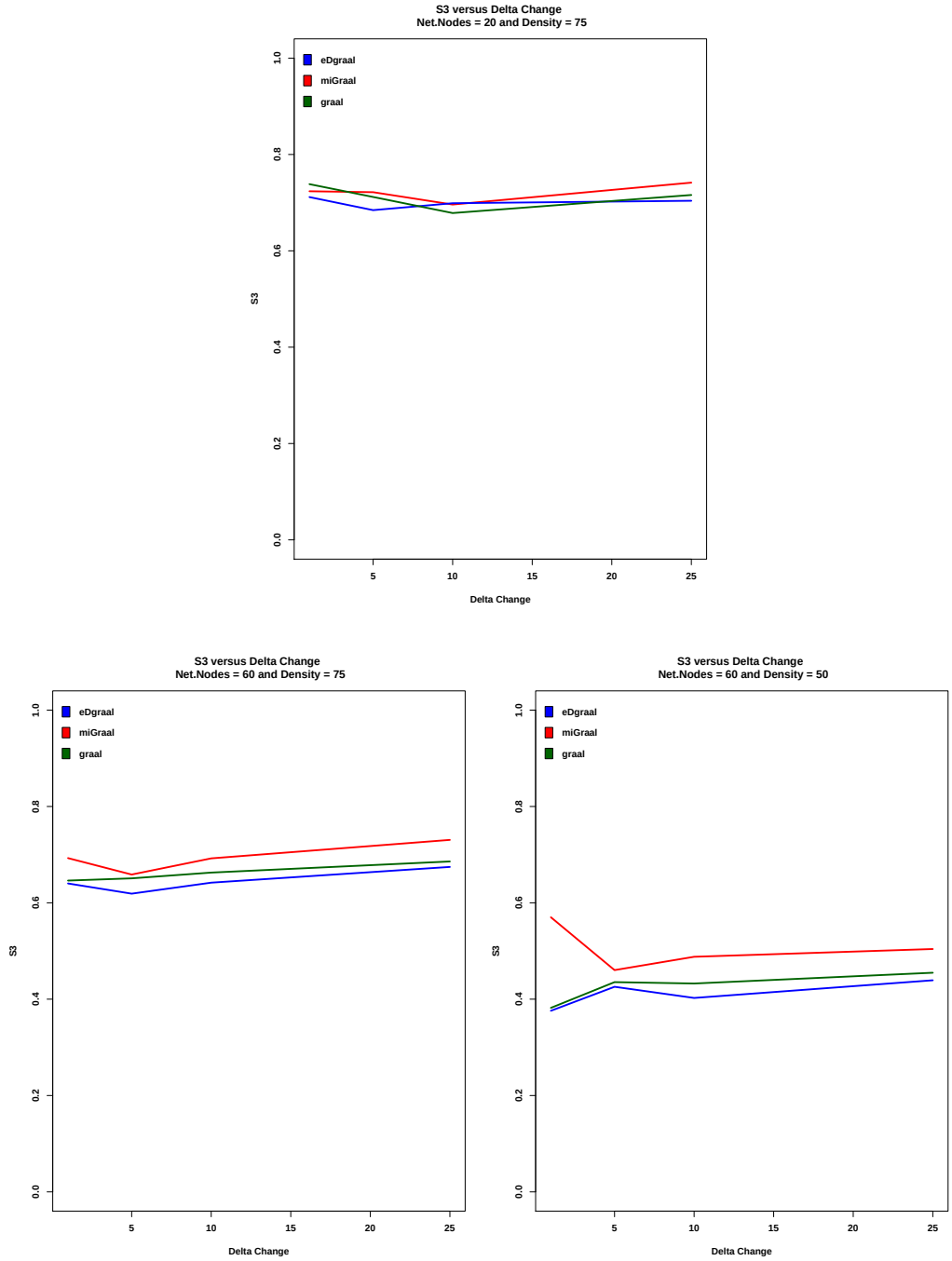


Figure 4.15: Change in s3 metric with respect to delta value

4.5 Conclusions

In conclusion, a novel algorithm was developed, based on Dijkstra's shortest path algorithm, such that its computational speed that of the well established global network alignment algorithms. Not only is D-GRAAL computational speed consistent across network structural properties, but the quality of the resulting alignment improves greatly with network complexity and size, therefore making D-GRAAL a suitable candidate for highly complex alignments, where results must be obtained at a fraction of the time currently spent. This, we believe, effectively places network alignment as a suitable solution for the analysis of real time, physical networks. Further, D-GRAAL exhibits unique design properties that make it suitable for parallelization.

Consequently, we believe D-GRAAL serves as an efficient mechanism for the alignment, and subsequent analysis, of network dynamics in fast-changing, large, and complex networks. The properties here demonstrated, and the increase in usage of network abstraction of every-increasing complexity and size, D-GRAAL will be of great help to the analysis community involved in a variety of fields.

Finally, it is understood that new network alignment algorithms may have been created since our algorithm was developed. Consequently, its comparison with respect to other algorithms is an area open for future research.

Chapter 5

Alignment-Based Analysis of Spatio-Temporal Complex Networks

5.1 Introduction

Spatio-temporal networks (STNs) are spatial networks whose topology and/or attributes change with time. These are encountered in many critical areas of everyday life such as transportation networks, electric power distribution grids, and social networks of mobile users [31], as well as in physical systems such as those encountered in the biological sciences. As seen in the introductory chapters, the analysis of STNs presents particular challenges, and is usually tackled with probabilistic methodologies that do not account for the structural characteristics of the networks at hand.

In this chapter, we introduce methodology and associated metrics for the analysis of STNs, particularly the identification of change mechanisms in a time-series of spatial

networks. While our methods extend to several forms of dynamics analysis, we here concern with the identification and classification of split/merge events, intentionally introduced into synthetically generated network time series. We define a *merge* as the combination of two nodes into a single node, therefore joining all prior edges onto the new node. On the other hand, a *split* occurs when an edge is broken or disappears.

First, we introduce a number of measures for improving node-similarity on STNs. We then describe in detail the following process by which alignment can be utilized to identify split/merge events in said STNs:

1. Generate relevant similarity measures
2. Carry out alignment between subsequent time points
3. Identify events
4. Generate event features
5. Classify events

Finally, we apply our methodology to the labeling of split/merge events in a network generation process for a synthetic STN testbed. We conclude this chapter with a discussion of the results and concluding remarks on achievements and future work on the matter.

5.2 Data

Now, as mentioned in our introduction, we have produced a full benchmark suite, as denoted in the table below:

Order/Density	5%	10%	25%	50%
10	-1%	All	All	All
20	All	All	All	All
40	All	All	All	All
60	All	All	All	All
80	All	All	All	All

Table 5.1: Merge Index Explored (All = [1%, 5%, 10%, 25%])

In this benchmark process we test our inference process on 30 time-points time series for each cross section between the following parameters:

1. Order: Number of nodes in a network
2. Density: Ratio of edges occuring in a network, with respect to the total number of edges that could occur. This equates to the probability for an edge to occur between any two nodes, and relates directly with the complexity of the network
3. Merge Index: Percentage of node pairs that are merged on a given time transition. Similarly, a split index is set, equal to half the number of splits.

5.3 Methods

5.3.1 Spatio-Temporal Topological Measures

A topological network alignment requires that nodes from one network are compared to those in its comparing network, one by one, in order to identify the best possible approximation to their corresponding embeddenness problem. While centrality measures such as Graphlets and degree are usually employed to achieve the highest similarity possible, STNs provide additional information that can be exploited for the assessment of node similarity. Thus, we have created a number of novel node-

similarity metrics, and compare with the use of graphlets as a sole metric, in order to facilitate the evaluation of dynamic network structures belonging to STNs such as those encountered in real physical systems.

1. Graphlet: Described by Oleski et Al [45] as a complex metric relying on the difference in the number of edges at increasing neighborhood depths for a given node. This measure was normalized to the range of similarities between two given networks. Thus, a similarity set will always have a maximum similarity of 1, and a minimum similarity of 0. Two sets of graphlet orbits are shown in figure 5.1.

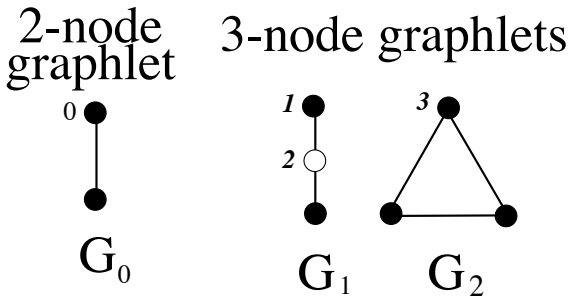


Figure 5.1: Sample graphlet orbits (Olieskii et Al.)

2. Tension: A nodes tension is defined as the sum of lengths for all edges connected to it. Thus, the similarity $T_{i,j}$, between nodes i and j, selected from different networks, is:

$$T_{i,j} = 1 - (DT_{i,j}/maxT)$$

Where DTi,j is the absolute value of the difference in tension between the nodes, and maxT is the maximum value in this set of differences. A sample of tension calculation is shown in figure 5.2.

3. Location: This similarity is based on the Euclidean distance between nodes from different networks, based on their 3D coordinates. Thus, the similarity

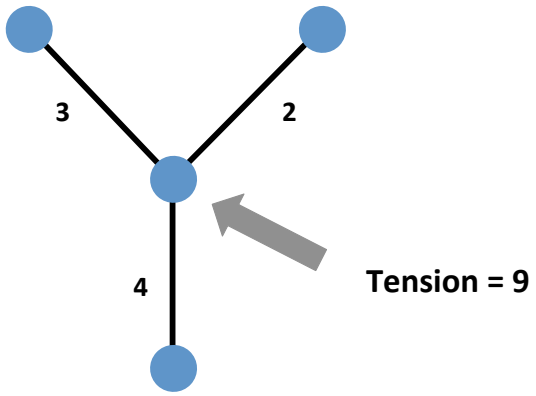


Figure 5.2: Schematic representation of tension similarity metric

$S_{i,j}$, between nodes i and j, selected from different networks, is:

$$S_{i,j} = 1 - (D_{i,j}/maxD)$$

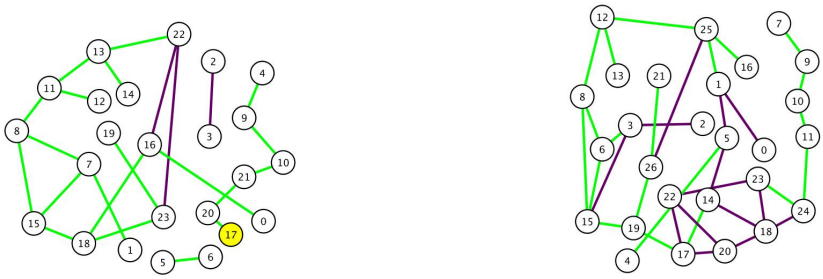
Where Di,j is the aforementioned Euclidean distance between the nodes, and maxD is the maximum value in the set of distances.

In addition, combinations graLoc, graTen, locTen, and graLocTen can be constructed by aggregation of individual measures, and further normalization.

Consequently, for every pair of subsequent time-points an alignment is carried out using one of these 7 similarity measures, in order to obtain the best possible approximation to the embeddedness relationship of the networks at hand. A thorough analysis on the effect of these measures on the quality of alignments is presented in the results section of this chapter.

5.3.2 Identification of Not-Covered Edge Sets

The next step in the process for the use of network alignment for the understanding of dynamics in a STN system, after alignment of two consecutive snapshots, making use of the relevant spatio-temporal node-similarity measures described above, is the identification of changing regions between comparing networks. Thus, we evaluate the alignment in search for sets of edges that are not mapped within the alignment process. We call one such set a not-covered edge set (ES). Having produced a comprehensive list of such differing graph components, which we determine them to be the result of dynamics occurred between the time-points. We must now classify these regions in terms of structurally changing events such as split/merge. A graphical representation of differing regions between two subsequent time-points is displayed in figure 5.3, where purple regions on either graph are deemed to be missing in the other, therefore representing an event:



Timepoint 1 Timepoint 2

Figure 5.3: Not-Covered Edge Sets in two subsequent timepoints

5.3.3 Event Set Consolidation

Event sets can be identified with respect to the earlier or later timepoints in a given transition. Sometimes, however, event sets may be consequences of the same merge or split, and therefore could be counted twice. Thus, we address this by a process of

consolidation in which the centroid of a given set is identified, and compared to that of all other sets identified on the same transition point. If a centroid corresponding to a given set is found to be contained within another set's boundaries, the two are considered as one.

5.3.4 Event Feature Generation

Once an alignment is complete, and not-covered edge sets are identified, we analyze them to extract information valuable to subsequent analysis steps, such as that of the classification of their nature into split/merge labels. To do so, we have designed a number features (covariates) related to the each component, as follows: tension, slope, source graph, node count, persistence, and linkage, are here formalized as described. To do so, we use a running example, shown in figure 5.4, to aid in the visualization of the different event features, labeled as *ES1* and *ES2*:

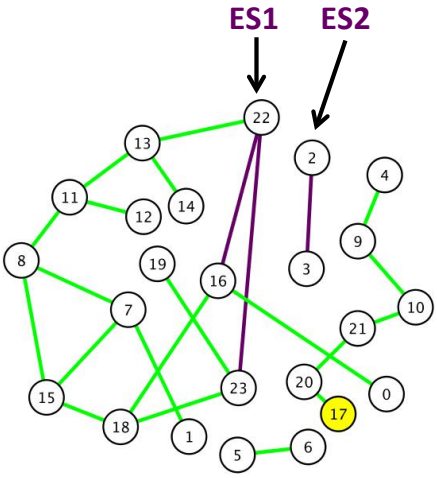


Figure 5.4: Not-covered edge sets (ES) resulting from an alignment

Consequently, we design the following features as suitable indicators of the nature of an edge set, ES_i:

1. Weight: Total component weight, where the cost of each edge is added to that of any other connected edge.

   - $Weight(ES1) = \sum_{i \in ES1} Cost_i$
   - $Weight(ES2) = \sum_{i \in ES2} Cost_i$

2. Source: Source graph index (0 = earlier timepoint, 1 = later timepoint, 2=both timepoints[overlapping centroid]). For example, given that our sets in figure 5.4 are extracted from the same network (timepoint 1 for example), we would have:

   - $Source(ES1) = 1$
   - $Source(ES2) = 1$

3. NCount: Number of nodes in component. Thus, counts for the sets in figure 5.4 are:

   - $NCount(ES1) = 3$
   - $NCount(ES2) = 2$

4. Persistence: Indicator (0/1) value for the complete persistence of component in the next timepoint.

5. Density Change: A real number accounting for the density change between timepoints.

6. Density Change Magnitude: Magnitude of density change between timepoints.

7. Slope: geometrical slope between the bottom-left-most, and top-right-most nodes in the set. Conveys a sense of "direction" for the set at hand.

8. Linkage: Number of graph components to which subject set is attached. Thus, the linkage values for the sets in in figure 5.4 are:

- $Linkage(ES1) = 5$
- $Linkage(ES2) = 0$

5.3.5 Statistical Inference of Split/Merge Events in STNs

Finally, we develop an inference model from which event labels can be assigned to unobserved networks in a given network generation process. This is, we seek to predict the binary label for a component Y, identified as an event in graph transition, i.

Model

Given the binary nature of our response variable of interest, split/merge, we model our data by means of a logistic regression. To this end we define a set of covariates using the features described above, as follows:

$$\text{logit}(p) = \log\left(\frac{p}{1-p}\right) = f(X)$$

and,

$$f(X) = \beta_0 + \beta_1 WEIGHT + \beta_2 SOURCE + \beta_3 NCOUNT + \beta_4 PERS + \beta_5 DCHANGE + \beta_6 MAGN + \beta_7 SLOPE + \beta_8 LINK$$

where p is the probability of event being a merge, for a given set of variables, and $\log\left(\frac{p}{1-p}\right)$ is the log(odds) of a merging event at a given covariate arrangement.

5.3.6 Validation

Now, after predicting our outcome labels, we needed to check how well our inference process works. This was done by a process of 10-fold cross-validation, on nearly 13,000 cases (events) produced synthetically. Further, given our ratio of merge to split events produced (2:1), we have an unbalanced data set, for which $\sim$ 66% of the labels are 1 (merge), and $\sim$ 33% are 0 (split). Thus, we carry out a stratified cross-validation process, by which folds are allocated randomly, with the caveat that their distribution will have roughly the same number of fission and fusion cases.

5.4 Results

5.4.1 Effect of Node-Similarity Metrics on Alignment Quality

As a first step in our analysis process, we determined the overall effect of our similarity metrics on alignment quality. Comparative analysis of edge coverage (EC) across the different node-similarity metrics is seen in figure 5.5. In it, we can observed that the combination between graphlet, location, and tension is conducive to a higher overall alignment quality.

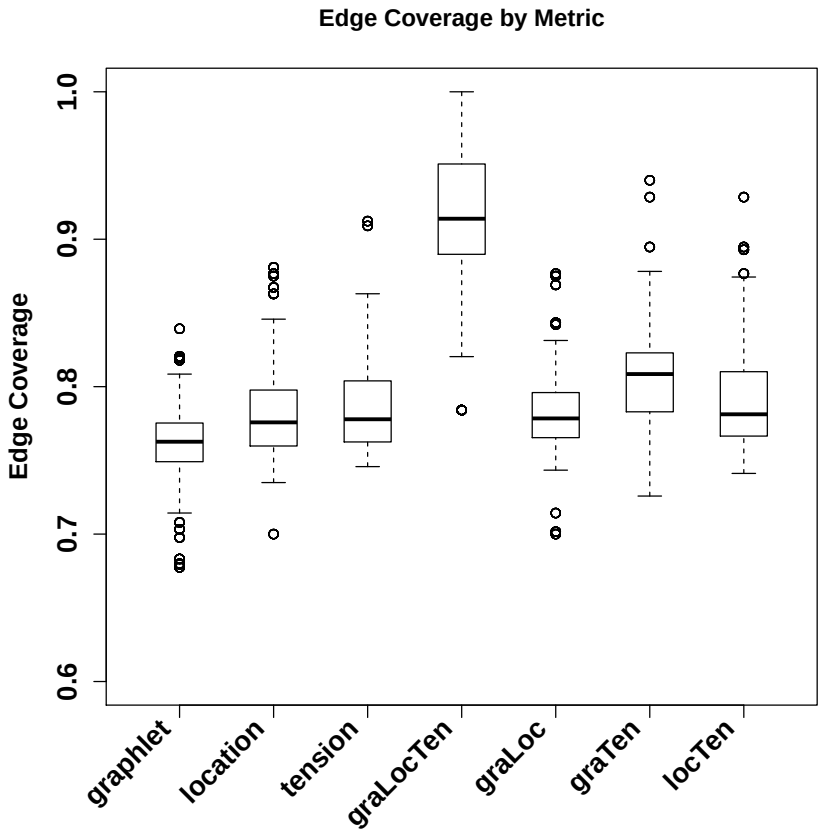


Figure 5.5: Preliminary overall effect of similarity metrics on alignment quality

5.4.2 Inference Model

Preliminary Analysis

First, we have looked at the individual covariates to make sure they conform to our understanding of the phenomena, and that no algorithmic malfunction has been spotted through this stage. In figure 5.6, we plot the response variable to spot visual clusters, as well as two sample covariates, NCount and tension, for all the events in a time series. Further, we verify that no collinearity exist between NCount and tension, which could indicate that the number of nodes in a component would tend to be related to the total tension of it. At this point, no direct trend was spotted.

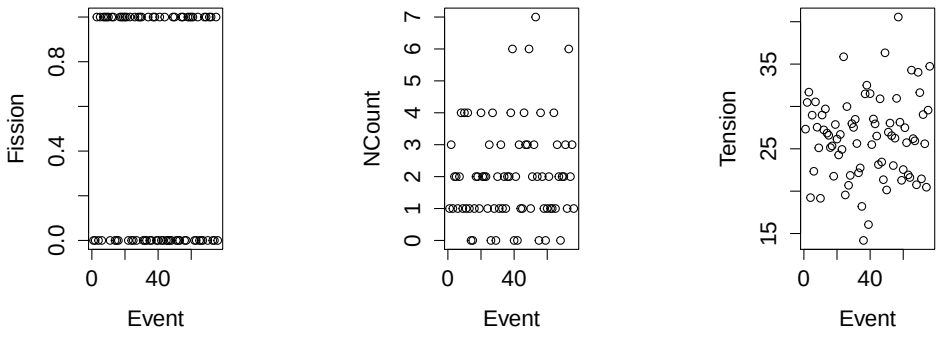


Figure 5.6: Sample Plots to ensure covariate integrity & no collinearity

Model Selection

After initial model elucidation, and fitness evaluation, we carry out a step-wise process in which our initial model is considered a full model, and evaluate the impact of removal of various covariates. Consequently, PERSISTENCE is removed. Thus, our null hypothesis is as follows:

$$Ho : \beta_4 = 0$$

$$Ha : \beta_4 \neq 0$$

A similar setup is enacted for the parameters SLOPE and SOURCE. The computational result of our generalized linear model (glm) function in R offers the solution to this hypothesis test, via its p-value evaluation for significance of a cofactor. Similarly, when we introduce covariates $N1EDGECOV$ and $N2EDGECOV$, to represent the edge coverage obtained during alignment with respect to the two networks.

Final Model Analysis

Now, given the nature of the problem, requiring us to assign labels to an outcome variable corresponding to a potential event, as identified by our algorithmic process,

we execute our inference process on our data set. For each new data point, composed of a vector with the prescribed event-set features. If the odds of labeling our outcome as MERGE (Y = 1) are higher than those of labeling it as SPLIT (Y=0), we predict the outcome to be 1 (merge). Thus, a generalized linear regression based on the binomial outcome is carried out on the data, by means of the R glm functionality. The results are presented in table 6.1.

Table 5.2: Second Binomial Regression Model for Prediction of Event Label

	Estimate	S.E.
Weight	0.000	(0.000)***
Slope	−0.152	(0.416)*
NCount	0.021	(0.002)***
Source		
Prev	−0.096	(0.031)
Post	−0.066	(0.034)
Both	−0.16	(0.034)
linkage	0.133	(0.045)**
N1-EdgeCov	6.227	(0.566)***
N2-EdgeCov	−9.641	(0.645)***
persistence	−0.003	(0.001)*
dChange	−0.3.5	(0.004)***
magnitude	−0.022	(0.002)**
AIC	4267.656	
BIC	4322.853	
Log Likelihood	-2253.324	
Deviance	1385.712	
Num. obs.	2000	

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$

As observed, the general association between the label of an event, and the selected features follows our intuitive understanding of the phenomena. Thus, the p-values assessed given the support set are summarized in figure 6.1 via asterisks in the right-hand side of the table. The more asterisks a covariate has, the more statistically significant it is for the regression at hand. As observed in the table, while it is harder to gauge the persistence of an event over a period of preceding or subsequent time-points, the weight imposed on a given component is indeed indicative of the odds of a changed component to be the result of a fusion event. Similarly, the number of elements in the component (NCount) yields some insight into the nature of the event, increasing the odds of being a fusion. On the other hand, the results lead us to believe that a better notion for "Slope" might be the contrasting of one such measure with that of its nearest component, since the direction of a component, by itself, may not be very relevant to the actual event producing it.

Classification Task

As a main result on the task of labeling merge/split events, our logistic inference model achieves an average accuracy of 92%, with a precision value of 84%, and the apparent full co-locality of labels for give time transitions. Figure 5.7 displays the accuracy and precision scores obtained on the cross-validated data.

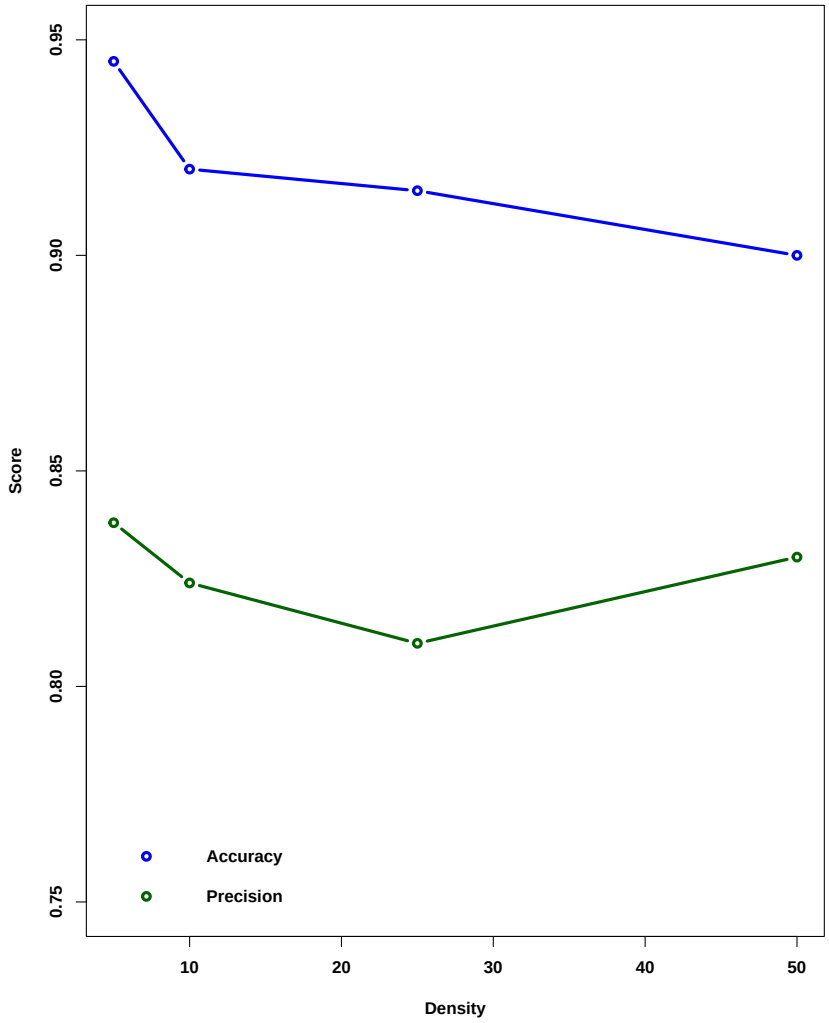


Figure 5.7: Positive Predictive Scores with respect to density change

5.5 Discussion

An automatic process of split/merge event identification and labeling was introduced, tested, and successfully validated.

From the methods above we can draw a number of helpful lessons. First, the methods at hand clearly establish the insignificance of one of our engineered features, *Persistence*, which attempts to indicate the remaining nature of the event at hand, with

respect to a subsequent timepoint. In all the preliminary analyses, as well as in the regressions stated above, such parameter did not play a significant role in deciding a label for an event. A possible explanation for this is that network components, specially in highly dense graphs, are very susceptible to change at each iteration, thus making this variable a poor indicator of the event at hand. i,e: Most of the newly added/subtracted components are somewhat mutated within the next time-frame, at least partially.

A potentially improvement to the problem of persistence, or more generally, to that of time-frame invariance, would be to identify auto-regressive effects beyond the algorithmically generated *Persistence* variable. Thus, the probability of a component at time t would depend on its state at time $t + 1$.

Chapter 6

MitoGraphs: Analysis of Dynamics in Mitochondrial Networks

6.1 Introduction

The main organelle for cellular energy production, the mitochondria is often called the "powerhouse of the cell". Usually depicted as a single organelle (Figure 6.1) the mitochondria can also exist as a system of interconnected tubular networks [75]. Further, mitochondrial networks can be abstracted as mathematical graphs, or networks, whose properties can be quantified and analyzed [96].

Now, mitochondrial networks have been shown to dynamically adjust to the changing cellular environment by growth and contraction of its structure. Such contraction and expansion is chiefly exerted by fusion and fission between adjacent tubular elements [75], [64]. In order to efficiently identify the fission and fusion events occurring in a given mitochondria network, biologists need computational resources that automate such process, currently carried out manually. Given that under the biological

experimental conditions, the mitochondrial network is expected to undergo physical changes in its structural conformation, these changes are reflected in the update of specific graph-theoretical properties such as density. Consequently, leveraging the capabilities achieved through the use of our algorithms, associated metrics,and analysis methods, we here analyze computationally and statistically the dynamics inherent to the biological entity of mitochondrial networks.

This chapter introduces relevant background knowledge regarding the mitochondria organelle, its graph-theoretical abstraction, as well as the methods employed to observe and digitize a living mitochondria's information so that it can be analyzed by our methodology. We then describe the data obtained from our Biology collaborators, and discussed the methods implemented for the analysis and identification of fission and fusion events on the mitochondrial network data. We finally discuss the results obtained, and provide concluding remarks and conclusions regarding the achievements of our research, as well as potential limitations and opportunities for future improvement.

6.2 Background

6.2.1 Mitochondria

The mitochondria (singular *mitochondrio*) are organelles found in the cytoplasm of most eukariotic cells, thought to have originated from prokariotic cells [56]. The mitochondria contain DNA, which is utilized to produce transport RNA (tRNA), ribosomal RNA (rRNA), and proteins involved in electron transport and adenosine triphosphate (ATP) synthesis. Further, the mitochondrial replication works as an independent process from that of cellular division, and has been linked to responses

to cellular needs such as that of growth, arrest, and general energy regulation.

Structurally, a classic view of the the mitochondria, as displayed in figure 6.1, presents a mitochondrio as surrounded by a double membrane, where the layers are separated by an *intermembrane space*. Together, the different layers collaborate in regulating the transport of substances from the cytoplasm to the interior, and vice versa.

Further, the inner membrane occurs as a series of foldings across the perimeter of the organelle, and is further divided into *cristae*, suggested by Thar et al. to influence the propagation of electromagnetic radiation in mitochondria [93]. As the inner membrane folds across the mitochondrion, it surrounds the *matrix*, where (ATP) -the "currency" molecule in most organisms, is created.

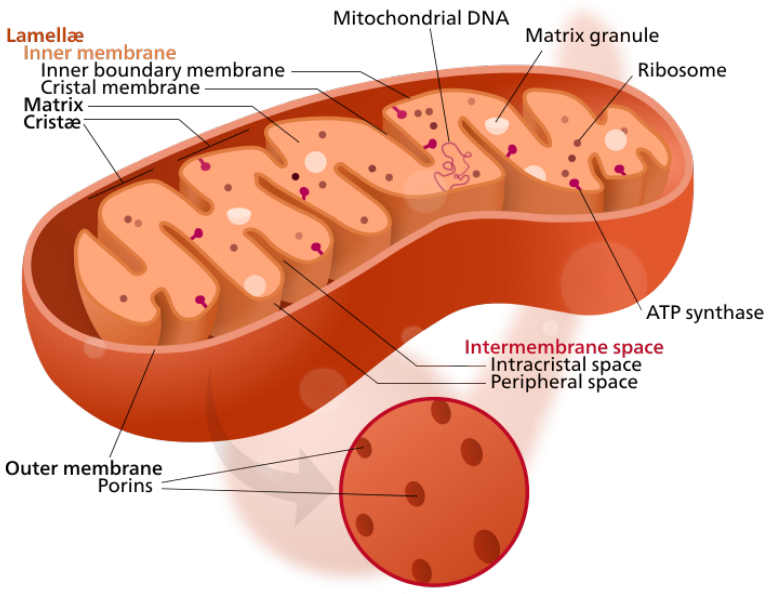


Figure 6.1: Classic representation of a mitochondrion. [101]

With respect to its function, the mitochondria is best known for its direct participation in the process of cellular energy production. However, the implications of its function extend to temperature control (homeostasis), aging, diabetes, and cancer control via programmed cellular death (apoptosis), among others. Consequently, understanding the biophysical processes in the mitochondria is of utmost importance to

the scientific community, specially as it relates to human health outcomes.

Now, as we mention before, the mitochondria does not only exist as separate units dispersed across a cell's cytoplasm. Depending on the organism and the type of cell, a mitochondrion may fuse into complex structures that can be represented as networks by adherence of its tubular elements to neighboring structures, via fusion biophysical processes at different points of its structure. Similarly, in order to adjust to the changing spatial constraints in the inner cell, as well as to possible environmental threats and diseases, the mitochondria may undergo repeated, and even simultaneous fission events in its tubular network. Consequently, the analysis and understanding of the relationship between mitochondrial structure and function is of vital importance. To this effect, an argument is made by Dr. Suzanne Rafelski, our Biological collaborator, for the integrative understanding of mitochondrial morphology [74]. Under this paradigm, organizational aspects such as single or multiplexed mitochondrial occurrence are integrated with structural properties such as size, shape, and position. As a consequence, a geometrical understanding of the organelle can be achieved such that behavior, function, and dynamics can be simulated and subsequently analyzed using computational methods.

6.2.2 Network Abstraction of Mitochondrial Complexes

Adopting an integrative view of the mitochondria is an important step towards the understanding of the organelle at multiple levels of observation, from the nanoscale molecular view of the mitochondrial ultrastructure, to the micron scale microscopical view of tubular networks [74]. Further, having a suitable model for the exploration of geometrical organelle properties in silico constitutes an important step towards a comprehensive understanding of the behavioral aspects of the mitochondrial complexes

and their associated dynamics [96].

Mitograph, an imaging processing method and software application was developed to quantify the volume of mitochondrial networks, and is optimized to perform on budding yeast organelles [96]. To accomplish the task, live samples are cultured and prepared, tagged with fluorescent proteins (FP) and exposed to 3D spinning-disk confocal microscopy, such that a *z-stack* image of the exposed mitochondria complexes are obtained.

Further, Mitograph processes the images obtained by the microscope, applies appropriate mathematical filters, and generates two complementary views of the mitochondrial complex: *surface* and a *skeleton* (See figure 6.2). The surface view displays the complex as a conglomeration of pixels with different densities across the structure, while the skeleton view displays the complex as having constant density al throughout. However, as seen in figure 6.2, both surface and skeleton views represent the mitochondrial complex as a network system where a tubular structure constitutes and edge, and the end points or bifurcations of a tubule constitute a node, as seen in part D) of the caption.

Finally, it is important to note that Mitograph is optimized solely to work on budding yeast *Saccharomyces cerevisiae*, whose mitochondria are three-dimensional (3D) networks of dynamic membrane-bound tubules localized at the cell periphery ([75], [63]). However, Mitograph tests on U2OS cells (human bone osteosarcoma) demonstrated additional challenges in differentiation of tubular conformations that render the images obtained as less than ready to subsequent computational analyses [96].

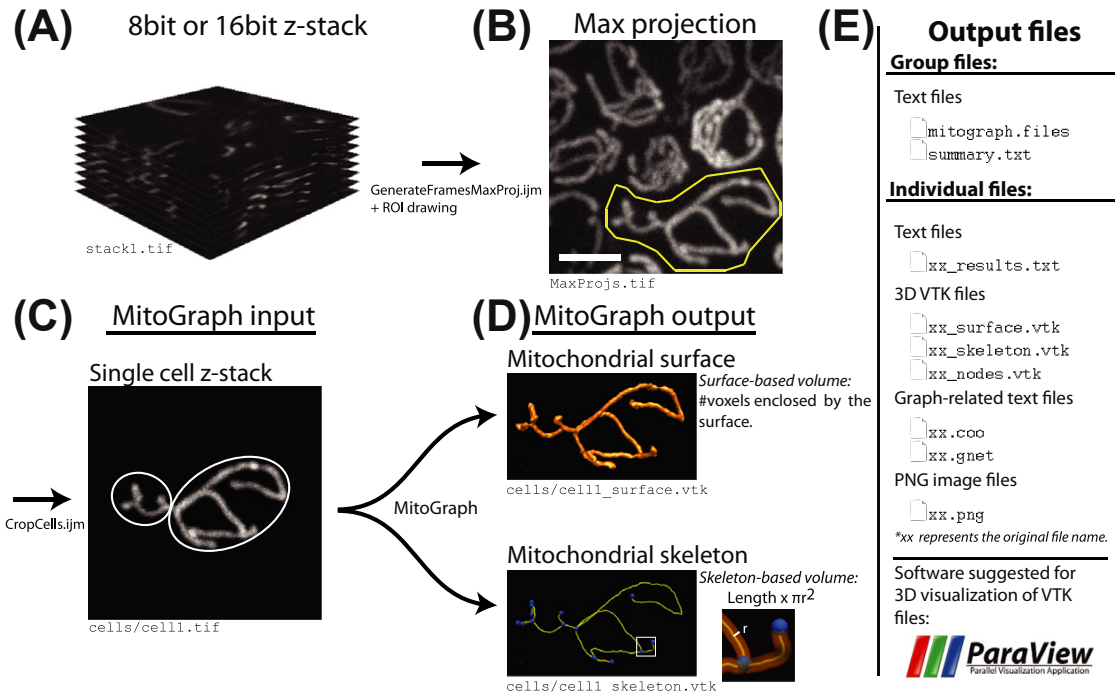


Figure 6.2: Overview of how to use MitoGraph software. Obtained from, and fully described in [96]

.

6.2.3 Mitochondrial Network Dynamics

The relationship between mitochondrial structure and function is known to have important implication for eukariotic cellular behavior, and therefore its impact extends to animal health outcomes. Such structure is known to be regulated in response to cellular developmental needs, as well as to changes in the intra or extra cellular environment ([25], [61], [64]). Now, given that mitochondria's growth mechanisms are independent from that of its host cell, structural and organization corrections are carried out via fission and fusion mechanisms between the composing tubular elements. Thus, the study of dynamics in mitochondria is most often focused on the analysis of fission and fusion events, their patterns, causes, and consequences [37], [74]. Consequently, the full understanding of fission and fusion phenomena helps the potential identification, forecast, and correction of unstable or problematic states ([75], [25], [61]), from which larger biological conclusions can be drawn.

However, the ability to automatically identify fission and fusion events from microscopy is still a manual task. Mostly carried on budding yeast ([75], [83]), the software application, Mitograph, provides an important step in the acquisition of re data and an excellent network abstraction of the organelle, such that the next step in automation can be carried out. An example of such network representation of fission and fusion events are shown in figure 6.3.

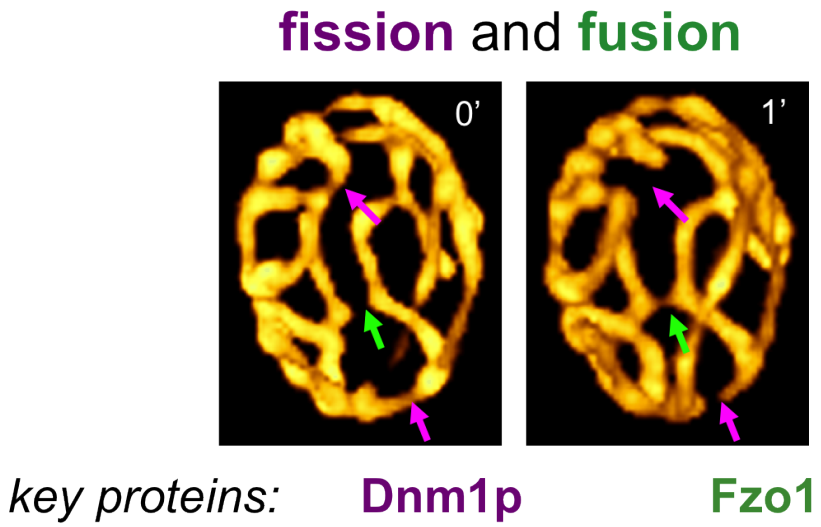


Figure 6.3: Biophysical events at Microscopic level

Consequently, this chapter provides a computational method for the identification of fission and fusion events in budding yeast mitochondria, using images obtained using Mitograph [96].

6.2.4 Approach: Alignment-Based Analysis

We consider the problem of identification of fission and fusion events in mitochondrial imagery to be analogous to the problem presented in chapter 5; That of identifying merge/split events in spatio-temporal networks (STNs). To this effect, the skeleton network abstraction obtained by Mitograph is used. However, a few simplifications are made:

- No self-loops are represented in the network.
- No double edges are represented in the network.

Given that our merge/split (fusion/fission) identification method captures any structural modification occurred between two time-points, these may include the addition of a set of nodes, or the break up of a given edge. Consequently, events occurred as a consequence of changes occurred in self or second loops would be captured and an attempt at labeling would be made.

Thus, the method for event identification and labeling demonstrated in chapter 5 is here applied. In addition, a machine learning approach at labeling events is also applied and compared, using the same set of features identified in chapter 5, with the exception of a more robust method for capturing time dependencies across networks. Lastly, a machine learning clustering method is applied to an unlabeled data set, in order to analyze the effect of using the referenced set of features (covariates).

6.3 Data

Two data sets are used. The first, taken from 4 independent experiments with budding yeast cells, exposed to the green fluorescent protein (GFP) chemical tag, which makes the organelle's tubuli appear as green when photograph by a confocal microscope, as outlined in our earlier section on Mitograph. For each of the experiments, subject cells were imaged at 18 different timepoints, and manual identification of fission/fusion events was carried out. The second dataset is composed of 49 time points, obtained from an immobilized mithocondria organelle, though no labels were obtained for any of the transitions, due to the time implication of the manual process it currently entails.

On both sets, our collaborators prepared the graph representation for each of the timepoint/snapshots, using the software application Mitograph, as per the published protocol [96].

6.4 Methods

6.4.1 Characterization of Density Change in Mitochondrial Networks

Given the results obtained in our simulated network experiment (See chapter 5), an improved method for inclusion of time dependency between networks was created. Considering that a measure of density tendency is a good indicator of mitochondrial behavior, we carry out a statistical time-series analysis of each mitochondria at hand, seeking to obtain a fitting regression formula, such that prediction on unseen values can be rapidly obtained. In turn, such predicted values are fed as a parameter in our logistic regression formula for label identification, or more generally, as an additional cofactor in a predictive machine learning process.

We define ***Density*** as the ratio between the actual number of edges in a network, and the maximum number a network with the same number of nodes could have, or,

$$Density = \frac{2 * Edges}{Nodes * (Nodes - 1)}$$

Further, while a higher number of mitochondria is associated with an increase in metabolic ability by the cell, and therefore with healthier energy outcomes, the effective impact of their network structural density is currently under study [1]. The

characterization of density change in mitochondrial networks is not only important for the biological repercusions of this knowledge, but also for the accurate representation of such networks in silico. To date, there exist no alternative methodologies for quantifying and assessing the impact of mitochondrial network density, thus making our analysis efforts extremely relevant to the interested scientific audience.

Preliminary Analysis

A number of preliminary data surveys are carried out to gain insight into the nature of temporal change in density on a given mitochondrial network. First, as observed in figure 6.4, we plot our original data, and determine a basic linear pattern, which is clearly complemented by some seasonal patterning, to be investigated in a future step.s

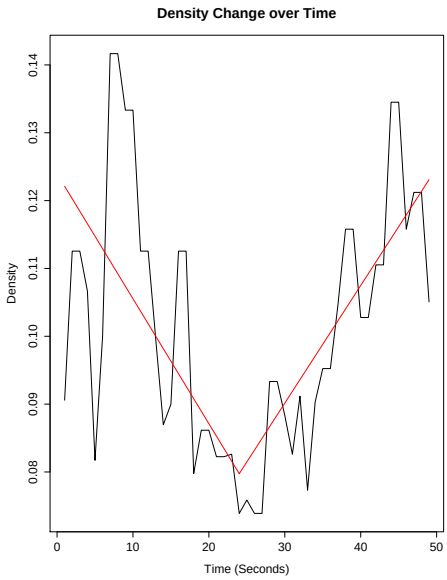


Figure 6.4: Linear fit of Mitochondrial Network Density

Further, our Auto Correlation Function (ACF) and Partial Correlation Function (PCF) plots, displayed in figure 6.5, allow us to determine that we are indeed observing an auto-regressive, AR(1) process, and suggest a potential recurrent behavior

with a period of 10 time units, approximately.

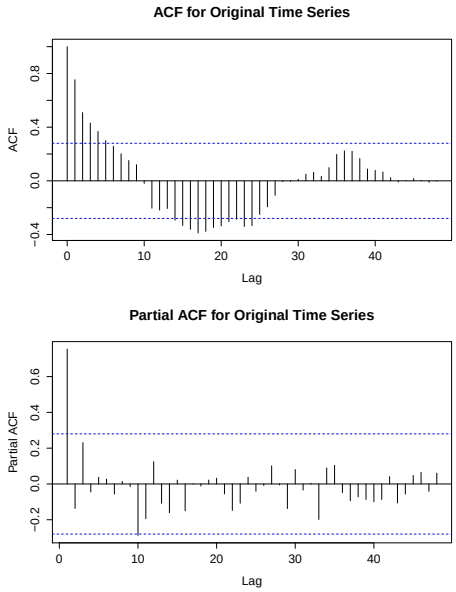


Figure 6.5: Autocorrelation structure of Mitochondrial Network Time Series

Next, we investigate the seasonal nature of the residuals resulting from fitting the preliminary simple linear model,

$$D_t = \beta 1 D_{t-1} + \epsilon_t$$

Where D_t is the observed density at time t. The correlation structure of said residuals is shown in figure 6.6, and gives strength to the notion of periodicity around 11 time units.

In an attempt to characterize the seasonal component of our time series structure, we fit sinusoidal curves to the residuals from our linear model, as follows:

- Sine Component: $sin(2 * \pi * freq * t)$
- Cosine Component: $cos(2 * \pi * freq * t)$
- Amplitude: $sqrt(var(residuals))$

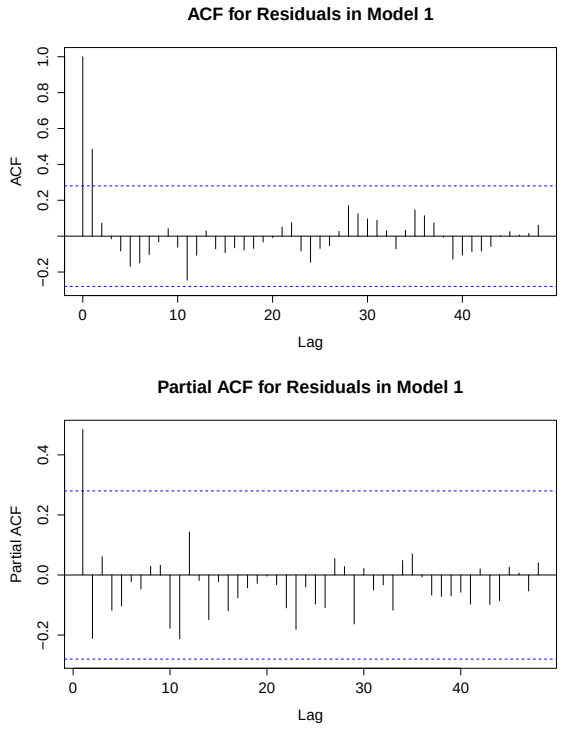


Figure 6.6: Correlation structure resulting from linear fit

Series Characterization

We move forward in our characterization of the time series at hand, by fitting our linear and sinusoidal components onto our time series, seeking to extract all sources of correlated behavior. This is, we fit the model,

$$D_t = \beta 1 D_{t-1} + S_t + C_t + \epsilon_t$$

Where D_t is the observed density at time t, and S and C are sine and cosine components at time t, respectively.

Finally, and moved by our results (discussed later) with the above model, we explored the characterization of our series in terms of an Auto-regressive Moving Average, ARMA(1,11) model. We do this heeding the information observed in our preliminary

analyses, where recurrence in the data seemed to occurred around $q = 11$.

Prediction

As mentioned int our introduction section, one of our goals for undertaking the analysis of mitochondrial density time series data, is to be able to predict unobserved data points. Whether predicting future or missing values, we hope to incorporate such ability into a major effort to identify the driving biological mechanisms for mitochondrial structural changes. Thus, we have prepared two set of validation sets. Given that our prediction problem is that of simple regression of density values, we have stablished a framework of cross-validated runs, using the results from the *predict* R-function, and comparing them to the actual observed values, having the minimum squared predictive error ($*MSPE$) as our validation criterion.

Validation

6.4.2 Statistical Inference of Events in Mitochondrial Networks

We assume the earlier analyses as important steps towards the overarching goal of labeling fission and fusion events occurred within a mitochondrial network. Consequently, we use the best subset of features identified in our statistical analyses on simulated networks, in addition to the predicted density value obtained from a suitable time-series model.

Table 6.1: Second Binomial Regression Model for Prediction of Event Label

	Estimate	S.E.
Weight	0.000	$(0.000)^{***}$
Slope	-0.152	$(0.416)^{*}$
NCount	0.021	$(0.002)^{***}$
Source		
Prev	-0.096	(0.031)
Post	-0.066	(0.034)
Both	-0.16	(0.034)
linkage	0.133	$(0.045)^{**}$
N1-EdgeCov	6.227	$(0.566)^{***}$
N2-EdgeCov	-9.641	$(0.645)^{***}$
persistence	-0.003	$(0.001)^{*}$
dChange	$-0.3.5$	$(0.004)^{***}$
magnitude	-0.022	$(0.002)^{**}$
AIC	4267.656	
BIC	4322.853	
Log Likelihood	-2253.324	
Deviance	1385.712	
Num. obs.	2000	

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$

6.4.3 Machine Learning Inference of Events in Mitochondrial Networks

In addition to our previous logistic regression model, we carry out an inference process using the machine learning mechanism knowns as decision tree, in which the information gain score for each of the covariates is evaluated, and a decision process is therefore implemented ([27], [12]). Further, the results obtained in our labeled dataset are evaluated via cross-validation, in order to gain an understanding for the generalization capacity of our predictive process.

6.4.4 Clustering of Unlabeled Events in Mitochondrial Networks

Finally,a clustering process is carried out on our unlabeled dataset via a K-means algorithm, where K=2 (fission/fusion surrogate sets) and 3 (fission/fussion/other). These experiments aim at evaluating the natural tendency of the data, both in strict and relaxed label frameworks, respectively.

6.5 Results

6.5.1 Time Series Characterization

The process of analyzing our time series data was sequential, and extensive. Thus, we only report the results for models that seem promising at the time of editorial, and for which comparison was carried out.

First, as explained in our methods section, we explore a model with linear and sinusoidal components. The results, shown in table 6.2 reflect the obvious importance of the linear trend, yet fail to incorporate successfully a sine/cosine combination.

	Estimate	Std. Error	t value	Pr(>|t|)
Time	-0.0019	0.0003	-5.80	0.0000***
Linear Component	0.0036	0.0006	6.46	0.0000***
Sine Component	-0.0061	0.0028	-2.17	0.0352*

Table 6.2: Initial Seasonal Pattern fit

Further, figure 6.7 shows the residual pattern of such model, which along with the correlation structure seen in figure 6.8 reflects the existence of additional trends, not capture within the model.

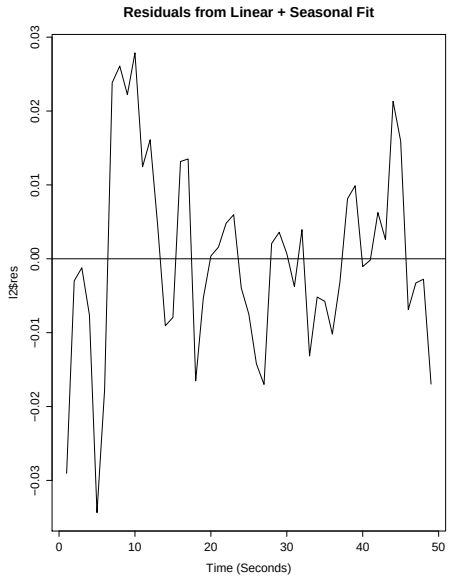


Figure 6.7: Residuals resulting from combined (linear and sinusoidal) fit

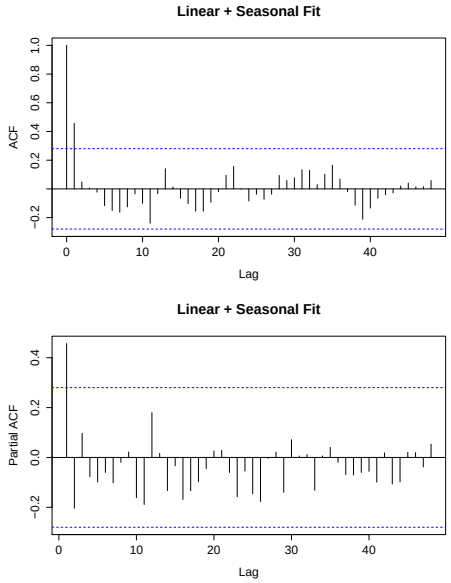


Figure 6.8: Correlation structure resulting from combined (linear and sinusoidal) fit

Consequently, we consider the ARIMA(1,11) model, guided by the continued observation of the residuals from earlier models, which indicate a strong patterning seemingly in terms of a moving average. Thus, such model is created and seems to work better than all other explored models, as seen in figures 6.9 and 6.10, where our residuals seem to contain no further patterning.

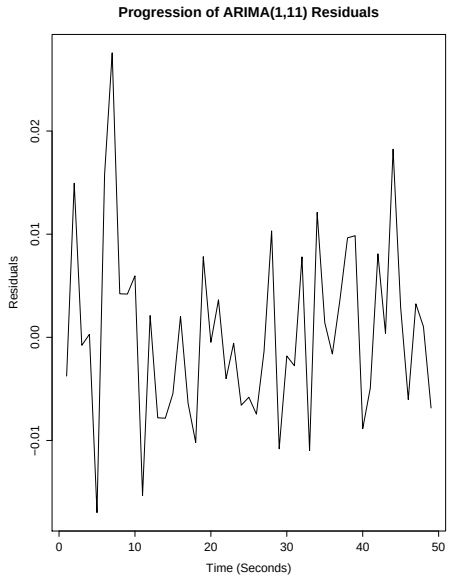


Figure 6.9: Residuals resulting from ARIMA(1,11) fit

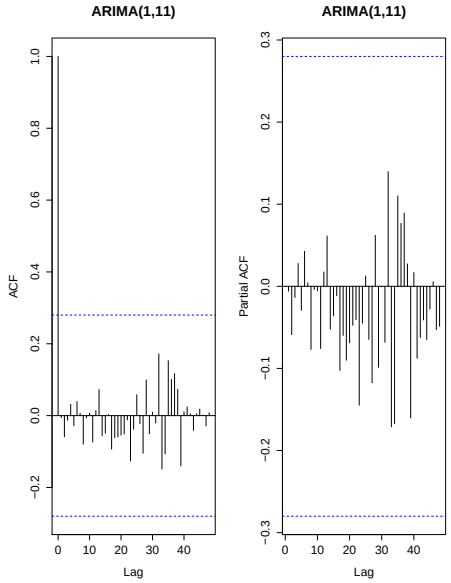


Figure 6.10: Correlation structure resulting from ARIMA(1,11) fit

6.5.2 Inference of Event Labels in Mitochondrial Networks

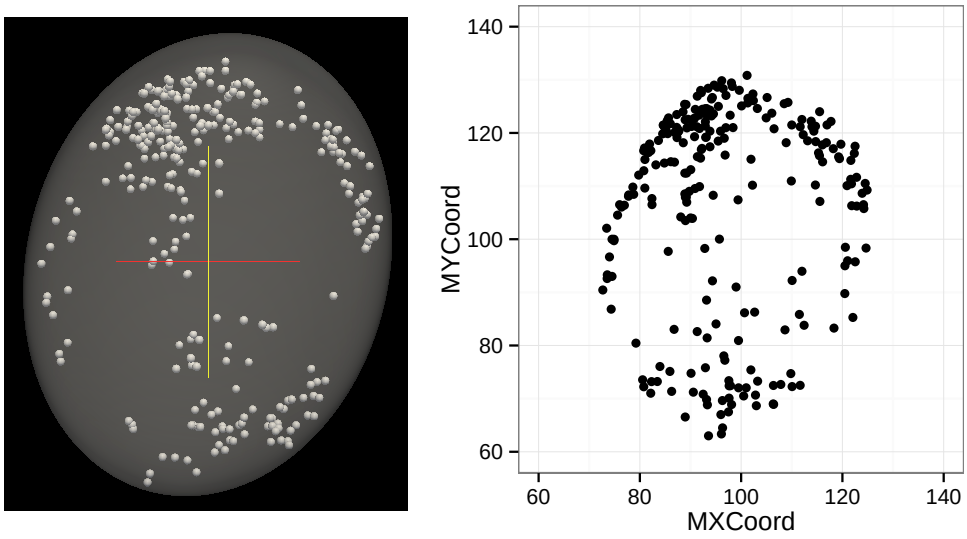
A logistic regression framework was applied to the problem of label identification in 5. The process yielded a cross-validated average of 92% with a precision value of 84% when evaluated on synthetic data. On our labeled biological data we obtained a

cross-validated average of 91% with a precision value of 88%, a relative improvement. Further, a single decision tree yielded a cross-validated average of 90% with a precision value of 83%.

6.5.3 Clustering of Events in Mitochondrial Networks

While expert labeling is yet to evaluate our results on this dataset, a preliminary check carried out on 10% of the available events yielded a 99% accuracy balance on the 2-set clustering assignments. This is, when forced to assign 1 out of 2 labels, our method accurately divides the events in 99% of the cases. On the other hand, a 3-set clustering yields an apparent rate of 70% accuracy, probably due to the assignment of boundary cases to either a fission or a fusion group, though not always to the one an expert would have selected.

Our most intriguing and valuable result at hand, however, is the fact that we are able to recapitulate, with considerable resemblance, the spatial patterning of events occurred in a real mitochondrial process. As displayed in figure 6.11, our collaborator's manually identified events are clustered mostly around the poles of a mitochondrial plane, which is largely recapitulated by our alignment-driven event identification process.



Manual Expert Labeling

Algorithmic+Statistical Process

Figure 6.11: Spatial event identification by manual and automated processes.

6.6 Discussion

An alignment-based analysis of dynamics in mitochondrial networks was carried out, yielding encouraging results, congruent with those obtained previously on simulated STNs. First, the process described in chapter 5 is applied to the microscopy images at hand, so that a suitable network abstraction is obtained. Second, the nature of time dependency on the density of a given mitochondria is analyzed, and a fitting model is used in the extrapolation of values to be used as an additional cofactor in the larger task of even labeling. Then, the augmented feature set was used in classification and a clustering tasks, seeking to evaluated their predictive ability in the context of real mitochondrial data.

The results obtained confirm the suitability of an alignment-based computational approach to the identification and labeling of biophysical events in mitochondrial networks, and set the stage for application to other types of network abstractions, whether in the biological sciences or in other areas of science and engineering.

However, there may be room for improvement at different stages of the overall process. For example, a more powerful machine learning mechanism could be deployed for the solution of the labeling problem. For example, a deep-learning mechanism trained on synthetic data could yield higher accuracy rates, while diminishing any false positive effects from image or process quality.

Chapter 7

Concluding Remarks

7.1 Contributions

This dissertation addresses various challenges related to the analysis of dynamics in real-world complex networks. First, the need for efficient computational analysis techniques that can enable the online evaluation of complex network streams is addressed vi a novel, more computationally efficient network alignment algorithm.

Second, the ability to identify and label node/edge-level events in a dynamic network process is address with relevant methodology, associated metrics, and examples of the incorporation of the results into automated statistical or machine learning prediction processes.

Lastly, our methodology is evaluated on a real-world complex biological network system, with encouraging results, and room for improvement and future research.

7.2 Improvement Opportunities

The improvement opportunities for the work presented are several. First, the computational efficiency of our network alignment algorithm, D-GRAAl, can be enhanced by the development of a distributed or parallel version, such that scalability in computational resources helps in the undertaking of ever more complex, bigger, and rapidly changing data sets.

Further, the methodology presented for identification and labeling of dynamic events can be extended to other types of events, in domain-specific problems. This is currently subject of further research by the author, as we address the problem of dynamic events in molecular dynamics (MD) simulation processes.

Lastly, the application of our methodology to the problem of analysis of mitochondrial dynamics will be further enhanced by future collaborative steps, in which expert feedback can be incorporated for the enhancement and validation of the methodology. Thus, while the inference mechanisms applied are rather simple, more powerful machine learning tools can be deployed at a later time, with the possible use of neural networks and semi-supervised learning techniques.

7.3 Application Opportunities

Given the ubiquitous nature of network representations, and the fast improvement in data gathering capability generated by sensors, web, and other massively deployed technologies, we believe the analysis of large, dynamic complex networks data sets provides a fertile ground for our methodologies to be deployed and improved.

A field of particular interest to the author is that of molecular simulations for drug-

discovery processes. In this realm, massive computational studies are undertaken on a regular basis, both in academic as in industry settings, with the goal of assessing the effects of molecular manipulation on large biological systems. This systems are often represented by network representations. Consequently, not only the processes of analysis at the MD level can be affected by the deployment of our techniques, but also the downstream analysis of effects on actual physical biological systems. This will be the focus of the author's next research efforts, as he embarks on further academic research in the near future.

Bibliography

[1] A. Akhmanova and M. O. Steinmetz. Tracking the ends: a dynamic protein network controls the fate of microtubule tips. *Nature reviews Molecular cell biology*, 9(4):309–322, 2008.

[2] R. P. Alexander, P. M. Kim, T. Emonet, and M. B. Gerstein. Understanding modularity in molecular networks requires dynamics. *Science signaling*, 2(81):pe44, 2009.

[3] F. Amir-Ghiasvand, A. Nowzari-Dalini, and V. Momenzadeh. Pin-align: a new dynamic programming approach to align protein-protein interaction networks. *Computational and mathematical methods in medicine*, 2014, 2014.

[4] S. Asur, S. Parthasarathy, and D. Ucar. An event-based framework for characterizing the evolutionary behavior of interaction graphs. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 3(4):16, 2009.

[5] M. N. Baliki, P. Y. Geha, A. V. Apkarian, and D. R. Chialvo. Beyond feeling: chronic pain hurts the brain, disrupting the default-mode network dynamics. *The Journal of Neuroscience*, 28(6):1398–1403, 2008.

[6] M. Bansal, V. Belcastro, A. Ambesi-Impiombato, and D. Di Bernardo. How to infer gene networks from expression profiles. *Molecular systems biology*, 3(1):78, 2007.

[7] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[8] A.-L. Barabasi and Z. N. Oltvai. Network biology: understanding the cell's functional organization. *Nature reviews genetics*, 5(2):101–113, 2004.

[9] J. Bascompte. Structure and dynamics of ecological networks. *Science*, 329(5993):765–766, 2010.

[10] L. Benda, N. L. Poff, D. Miller, T. Dunne, G. Reeves, G. Pess, and M. Pollock. The network dynamics hypothesis: how channel networks structure riverine habitats. *BioScience*, 54(5):413–427, 2004.

[11] J. Berg and M. Lässig. Local graph alignment and motif search in biological networks. *Proceedings of the National Academy of Sciences of the United States of America*, 101(41):14689–14694, 2004.

[12] C. M. Bishop. Pattern recognition. *Machine Learning*, 128:1–58, 2006.

[13] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang. Complex networks: Structure and dynamics. *Physics reports*, 424(4):175–308, 2006.

[14] E. Bullmore and O. Sporns. Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature reviews. Neuroscience*, 10(3):186, 2009.

[15] R. S. Burt. The network structure of social capital. *Research in organizational behavior*, 22:345–423, 2000.

[16] D. Chakrabarti, R. Kumar, and A. Tomkins. Evolutionary clustering. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 554–560. ACM, 2006.

[17] S. R. Collins, P. Kemmeren, X.-C. Zhao, J. F. Greenblatt, F. Spencer, F. C. Holstege, J. S. Weissman, and N. J. Krogan. Toward a comprehensive atlas of the physical interactome of saccharomyces cerevisiae. *Molecular & Cellular Proteomics*, 6(3):439–450, 2007.

[18] S. A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the third annual ACM symposium on Theory of computing*, pages 151–158. ACM, 1971.

[19] N. R. Council et al. *Network science*. National Academies Press, 2006.

[20] P. Dalle Pezze, A. G. Sonntag, A. Thien, M. T. Prentzell, M. Gödel, S. Fischer, E. Neumann-Haefelin, T. B. Huber, R. Baumeister, D. P. Shanley, et al. A dynamic network model of mtor signaling reveals tsc-independent mtorc2 regulation. *Sci. Signal.*, 5(217):ra25–ra25, 2012.

[21] E. Dayan, N. Censor, E. R. Buch, M. Sandrini, and L. G. Cohen. Noninvasive brain stimulation: from physiology to network dynamics and back. *Nature neuroscience*, 16(7):838–844, 2013.

[22] W. de Haan, Y. A. Pijnenburg, R. L. Strijers, Y. van der Made, W. M. van der Flier, P. Scheltens, and C. J. Stam. Functional neural network analysis in frontotemporal dementia and alzheimer's disease using eeg and graph theory. *BMC neuroscience*, 10(1):101, 2009.

[23] M. A. De Menezes and A.-L. Barabási. Fluctuations in network dynamics. *Physical review letters*, 92(2):028701, 2004.

[24] A. Deshpande, A. Göllü, and P. Varaiya. Shift: A formalism and a programming language for dynamic networks of hybrid automata. In *International Hybrid Systems Workshop*, pages 113–133. Springer, 1996.

[25] M. T. Figge, A. S. Reichert, M. Meyer-Hermann, and H. D. Osiewacz. Deceleration of fusion–fission cycles improves mitochondrial quality control during aging. *PLoS Comput Biol*, 8(6):e1002576, 2012.

[26] J. Flannick, A. Novak, B. S. Srinivasan, H. H. McAdams, and S. Batzoglou. Graemlin: general and robust alignment of multiple large interaction networks. *Genome research*, 16(9):1169–1181, 2006.

[27] J. Friedman, T. Hastie, and R. Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics Springer, Berlin, 2001.

[28] T. L. Friesz, J. Luque, R. L. Tobin, and B.-W. Wie. Dynamic network traffic assignment considered as a continuous time optimal control problem. *Operations Research*, 37(6):893–901, 1989.

[29] T. S. Gardner, D. Di Bernardo, D. Lorenz, and J. J. Collins. Inferring genetic networks and identifying compound mode of action via expression profiling. *Science*, 301(5629):102–105, 2003.

[30] M. R. Gary and D. S. Johnson. Computers and intractability: A guide to the theory of np-completeness, 1979.

[31] B. George and K. Sangho. *Spatio-temporal Networks: Modeling and Algorithms*. Springer Science & Business Media, 2012.

[32] M. Girvan and M. E. Newman. Community structure in social and biological networks. *Proceedings of the national academy of sciences*, 99(12):7821–7826, 2002.

[33] D. Greene, D. Doyle, and P. Cunningham. Tracking the evolution of communities in dynamic social networks. In *Advances in social networks analysis and mining (ASONAM), 2010 international conference on*, pages 176–183. IEEE, 2010.

[34] J.-D. J. Han, N. Bertin, T. Hao, D. S. Goldberg, G. F. Berriz, L. V. Zhang, D. Dupuy, A. J. Walhout, M. E. Cusick, F. P. Roth, et al. Evidence for dynamically organized modularity in the yeast protein–protein interaction network. *Nature*, 430(6995):88–93, 2004.

[35] M. Hecker, S. Lambeck, S. Toepfer, E. Van Someren, and R. Guthke. Gene regulatory network inference: data integration in dynamic modelsa review. *Biosystems*, 96(1):86–103, 2009.

[36] D. Husmeier. Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic bayesian networks. *Bioinformatics*, 19(17):2271–2282, 2003.

[37] V. Jayashankar and S. M. Rafelski. Integrating mitochondrial organization and dynamics with cellular architecture. *Current opinion in cell biology*, 26:34–40, 2014.

[38] B. P. Kelley, R. Sharan, R. M. Karp, T. Sittler, D. E. Root, B. R. Stockwell, and T. Ideker. Conserved pathways within bacteria and yeast as revealed by global protein network alignment. *Proceedings of the National Academy of Sciences*, 100(20):11394–11399, 2003.

[39] B. P. Kelley, B. Yuan, F. Lewitter, R. Sharan, B. R. Stockwell, and T. Ideker. Pathblast: a tool for alignment of protein interaction networks. *Nucleic acids research*, 32(suppl 2):W83–W88, 2004.

[40] S. Kikuchi, D. Tominaga, M. Arita, K. Takahashi, and M. Tomita. Dynamic modeling of genetic networks using genetic algorithm and s-system. *Bioinformatics*, 19(5):643–650, 2003.

[41] M.-S. Kim and J. Han. A particle-and-density based evolutionary clustering method for dynamic networks. *Proceedings of the VLDB Endowment*, 2(1):622–633, 2009.

[42] S. Y. Kim, S. Imoto, and S. Miyano. Inferring gene networks from time series microarray data using dynamic bayesian networks. *Briefings in bioinformatics*, 4(3):228–235, 2003.

[43] H. Kitano. Systems biology: a brief overview. *Science*, 295(5560):1662–1664, 2002.

[44] M. Koyutürk, Y. Kim, U. Topkara, S. Subramaniam, W. Szpankowski, and A. Grama. Pairwise alignment of protein interaction networks. *Journal of Computational Biology*, 13(2):182–199, 2006.

[45] O. Kuchaiev, T. Milenković, V. Memišević, W. Hayes, and N. Pržulj. Topological network alignment uncovers biological function and phylogeny. *Journal of the Royal Society Interface*, page rsif20100063, 2010.

[46] O. Kuchaiev and N. Pržulj. Integrative network alignment reveals large regions of global network similarity in yeast and human. *Bioinformatics*, 27(10):1390–1396, 2011.

[47] F. Kuhn, N. Lynch, and R. Oshman. Distributed computation in dynamic networks. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pages 513–522. ACM, 2010.

[48] G. Laurent. Olfactory network dynamics and the coding of multidimensional signals. *Nature reviews neuroscience*, 3(11):884–895, 2002.

[49] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graph evolution: Densification and shrinking diameters. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):2, 2007.

[50] Z. Liang, M. Xu, M. Teng, and L. Niu. Netalign: a web-based tool for comparison of protein interaction networks. *Bioinformatics*, 22(17):2175–2177, 2006.

[51] Q. Liao, A. Striegel, and N. Chawla. Visualizing graph dynamics and similarity for enterprise network security and management. In *Proceedings of the seventh international symposium on visualization for cyber security*, pages 34–45. ACM, 2010.

[52] Y.-R. Lin, Y. Chi, S. Zhu, H. Sundaram, and B. L. Tseng. Facetnet: a framework for analyzing communities and their evolutions in dynamic networks. In *Proceedings of the 17th international conference on World Wide Web*, pages 685–694. ACM, 2008.

[53] Y.-R. Lin, Y. Chi, S. Zhu, H. Sundaram, and B. L. Tseng. Analyzing communities and their evolutions in dynamic social networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 3(2):8, 2009.

[54] F. Lorrain and H. C. White. Structural equivalence of individuals in social networks. *The Journal of mathematical sociology*, 1(1):49–80, 1971.

[55] N. M. Luscombe, M. M. Babu, H. Yu, M. Snyder, S. A. Teichmann, and M. Gerstein. Genomic analysis of regulatory network dynamics reveals large topological changes. *Nature*, 431(7006):308–312, 2004.

[56] L. Margulis and D. Sagan. *Origins of sex: three billion years of genetic recombination*. Yale University Press, 1990.

[57] K. Mehlhorn and S. Näher. *LEDA: a platform for combinatorial and geometric computing*. Cambridge university press, 1999.

[58] T. Milenković, V. Memišević, A. Bonato, and N. Pržulj. Dominating biological networks. *PloS one*, 6(8):e23016, 2011.

[59] R. Milo, S. Itzkovitz, N. Kashtan, R. Levitt, S. Shen-Orr, I. Ayzenshtat, M. Sheffer, and U. Alon. Superfamilies of evolved and designed networks. *Science*, 303(5663):1538–1542, 2004.

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

[61] P. K. Mouli, G. Twig, and O. S. Shirihai. Frequency and selectivity of mitochondrial fusion are key to its quality maintenance function. *Biophysical journal*, 96(9):3509–3518, 2009.

[62] M. E. Newman. The structure and function of complex networks. *SIAM review*, 45(2):167–256, 2003.

[63] J. Nunnari, W. F. Marshall, A. Straight, A. Murray, J. W. Sedat, and P. Walter. Mitochondrial transmission during mating in saccharomyces cerevisiae is determined by mitochondrial fusion and fission and the intramitochondrial segregation of mitochondrial dna. *Molecular biology of the cell*, 8(7):1233–1242, 1997.

[64] P. K. Patel, O. Shirihai, and K. C. Huang. Optimal dynamics for quality control in spatially distributed mitochondrial networks. *PLoS Comput Biol*, 9(7):e1003108, 2013.

[65] M. Penrose. *Random geometric graphs*. Number 5. Oxford University Press, 2003.

[66] R. D. Phair, P. Scaffidi, C. Elbi, J. Vecerová, A. Dey, K. Ozato, D. T. Brown, G. Hager, M. Bustin, and T. Misteli. Global nature of dynamic protein-chromatin interactions in vivo: three-dimensional genome scanning and dynamic interaction networks of chromatin proteins. *Molecular and Cellular Biology*, 24(14):6393–6402, 2004.

[67] W. W. Powell, D. R. White, K. W. Koput, and J. Owen-Smith. Network dynamics and field evolution: The growth of interorganizational collaboration in the life sciences1. *American journal of sociology*, 110(4):1132–1205, 2005.

[68] R. J. Prill, P. A. Iglesias, and A. Levchenko. Dynamic properties of network motifs contribute to biological network organization. *PLoS Biol*, 3(11):e343, 2005.

[69] N. Pržulj. Protein-protein interactions: Making sense of networks via graph-theoretic modeling. *Bioessays*, 33(2):115–123, 2011.

[70] N. Pržulj, D. G. Corneil, and I. Jurisica. Modeling interactome: scale-free or geometric? *Bioinformatics*, 20(18):3508–3515, 2004.

[71] N. Pržulj and T. Milenkovi. Computational methods for analyzing and modeling biological networks. *Biological Data Mining. Chapman & Hall/CRC*, pages 397–428, 2009.

[72] T. M. Przytycka, M. Singh, and D. K. Slonim. Toward the dynamic interactome: it's about time. *Briefings in bioinformatics*, page bbp057, 2010.

[73] P. Radivojac, K. Peng, W. T. Clark, B. J. Peters, A. Mohan, S. M. Boyle, and S. D. Mooney. An integrated approach to inferring gene–disease associations in humans. *Proteins: Structure, Function, and Bioinformatics*, 72(3):1030–1037, 2008.

[74] S. M. Rafelski. Mitochondrial network morphology: building an integrative, geometrical view. *BMC biology*, 11(1):71, 2013.

[75] S. M. Rafelski, M. P. Viana, Y. Zhang, Y.-H. M. Chan, K. S. Thorn, P. Yam, J. C. Fung, H. Li, L. d. F. Costa, and W. F. Marshall. Mitochondrial network size scaling in budding yeast. *Science*, 338(6108):822–824, 2012.

[76] D. G. Rand, S. Arbesman, and N. A. Christakis. Dynamic social networks promote cooperation in experiments with humans. *Proceedings of the National Academy of Sciences*, 108(48):19193–19198, 2011.

[77] J. M. Read, K. T. Eames, and W. J. Edmunds. Dynamic social networks and the implications for the spread of infectious disease. *Journal of The Royal Society Interface*, 5(26):1001–1007, 2008.

[78] R. B. Rothenberg, J. J. Potterat, D. E. Woodhouse, S. Q. Muth, W. W. Darrow, and A. S. Klovdahl. Social network dynamics and hiv transmission. *Aids*, 12(12):1529–1536, 1998.

[79] R. Samtaney, D. Silver, N. Zabusky, and J. Cao. Visualizing features and tracking their evolution. *Computer*, 27(7):20–27, 1994.

[80] V. Saraph and T. Milenković. Magna: maximizing accuracy in global network alignment. *Bioinformatics*, 30(20):2931–2940, 2014.

[81] J. Selimkhanov, B. Taylor, J. Yao, A. Pilko, J. Albeck, A. Hoffmann, L. Tsimring, and R. Wollman. Accurate information transmission through dynamic biochemical signaling networks. *Science*, 346(6215):1370–1373, 2014.

[82] B. E. Shapiro and E. Mjolsness. Pycellerator: an arrow-based reaction-like modelling language for biological simulations. *Bioinformatics*, 32(4):629–631, 2015.

[83] J. M. Shaw and J. Nunnari. Mitochondrial dynamics and division in budding yeast. *Trends in cell biology*, 12(4):178–184, 2002.

[84] B. Skyrms and R. Pemantle. A dynamic model of social network formation. *Proceedings of the National Academy of Sciences*, 97(16):9340–9346, 2000.

[85] T. A. Snijders. The statistical evaluation of social network dynamics. *Sociological methodology*, 31(1):361–395, 2001.

[86] T. A. Snijders, G. G. Van de Bunt, and C. E. Steglich. Introduction to stochastic actor-based models for network dynamics. *Social networks*, 32(1):44–60, 2010.

[87] M. Spiliopoulou, I. Ntoutsi, Y. Theodoridis, and R. Schult. Monic: modeling and monitoring cluster transitions. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 706–711. ACM, 2006.

[88] J. Sun, C. Faloutsos, S. Papadimitriou, and P. S. Yu. Graphscope: parameter-free mining of large time-evolving graphs. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 687–696. ACM, 2007.

[89] J. Sun, D. Tao, and C. Faloutsos. Beyond streams and graphs: dynamic tensor analysis. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 374–383. ACM, 2006.

[90] L. Tang, H. Liu, J. Zhang, and Z. Nazeri. Community evolution in dynamic multi-mode networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 677–685. ACM, 2008.

[91] C. Tantipathananandh, T. Berger-Wolf, and D. Kempe. A framework for community identification in dynamic social networks. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 717–726. ACM, 2007.

[92] A. Terentiev, N. Moldogazieva, and K. Shaitan. Dynamic proteomics in modeling of the living cell. protein-protein interactions. *Biochemistry (Moscow)*, 74(13):1586–1607, 2009.

[93] R. Thar and M. Kühl. Propagation of electromagnetic radiation in mitochondria? *Journal of theoretical biology*, 230(2):261–270, 2004.

[94] J. J. Tyson, K. Chen, and B. Novak. Network dynamics and cell physiology. *Nature Reviews Molecular Cell Biology*, 2(12):908–916, 2001.

[95] A. van der Schaft. Modeling of physical network systems. *Systems & Control Letters*, 2015.

[96] M. P. Viana, S. Lim, and S. M. Rafelski. Quantifying mitochondrial content in living cells. *Methods in cell biology*, 125:77–93, 2015.

[97] T. P. Vogels, K. Rajan, and L. Abbott. Neural network dynamics. *Annu. Rev. Neurosci.*, 28:357–376, 2005.

[98] Y. Wang, T. Joshi, X.-S. Zhang, D. Xu, and L. Chen. Inferring gene regulatory networks from multiple microarray datasets. *Bioinformatics*, 22(19):2413–2420, 2006.

[99] S. Wasserman and K. Faust. *Social network analysis: Methods and applications*, volume 8. Cambridge university press, 1994.

[100] D. J. Watts and S. H. Strogatz. Collective dynamics of small-worldnetworks. *nature*, 393(6684):440–442, 1998.

[101] Wikipedia. Mitochondrion — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/wiki/Mitochondrion#/media/File:Mitochondrion_mini.svg`, 2013. Accessed: 04-04-2017.

[102] H. Yang, S. Parthasarathy, and S. Mehta. A generalized framework for mining spatio-temporal patterns in scientific data. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 716–721. ACM, 2005.

[103] L. You, A. Hoonlor, and J. Yin. Modeling biological systems using dyneticaa simulator of dynamic networks. *Bioinformatics*, 19(3):435–436, 2003.

[104] L. A. Zadeh. Fuzzy logic, neural networks, and soft computing. *Communications of the ACM*, 37(3):77–85, 1994.