

**Dynamic Mode Decomposition
and Network Motif Evolution**

A Thesis
Presented to the
Faculty of
San Diego State University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in Applied Mathematics
with a Concentration in
Dynamical Systems

by
Robert Simpson
Spring 2021

SAN DIEGO STATE UNIVERSITY

The Undersigned Faculty Committee Approves the
Thesis of Robert Simpson:

Dynamic Mode Decomposition
and Network Motif Evolution

Christopher Curtis, Chair
Department of Mathematics and Statistics

Christopher O'Neill
Department of Mathematics and Statistics

Xialu Liu
Department of Management Information Systems

Approval Date

Copyright © 2021
by
Robert Simpson

DEDICATION

To my family who have been wholly supportive.

ABSTRACT OF THE THESIS

Dynamic Mode Decomposition
and Network Motif Evolution
by
Robert Simpson

Master of Science in Applied Mathematics with a Concentration in Dynamical Systems
San Diego State University, 2021

Complex networks are structurally non-trivial and require a large set of tools to analyze their characteristics. To understand network development, we introduce a set of motifs to categorize local network structure. Examining the Barabási–Albert and Thij models and their respective mechanisms of network growth, we can study temporal motif interaction. We examine how the addition of nodes and edges affect motif composition. This analysis is insightful, but not sufficient to characterize network dynamics. We apply statistical correlation and covariance analysis to the dynamic motif counts to study those motifs which move together. Still we conduct further analysis, looking to characterize the underlying system that drives motif counts. To this end, we use the Dynamic Mode Decomposition algorithm and the Kernel Dynamic Mode Decomposition algorithm to find modes (spatiotemporal coherent structures) of the system. These modes offer a dynamical systems perspective to understand the interactions between motif counts. Finally, we analyze the results from our statistical analysis, DMD, and KDMD to understand motif interaction.

TABLE OF CONTENTS

	PAGE
ABSTRACT	v
LIST OF TABLES	ix
LIST OF FIGURES	x
ACKNOWLEDGMENTS	xvii
CHAPTER	
1 Network Theory	1
1.1 An Introduction	1
1.2 Complex Networks	1
1.3 A Short History	2
1.4 Network Analysis.....	2
1.5 What is Twitter?.....	3
2 Motifs.....	6
2.1 Common Objects of Graph Theory	6
2.2 Non-Simple Cycle Motifs	12
3 Barabási–Albert Model.....	15
4 Twitter and the Thij Model	22
4.1 $\lambda = 0.2, p = 0.2$	25
4.2 $\lambda = 0.8, p = 0.2$	28
4.3 $\lambda = 0.2, p = 0.8$	31
4.4 $\lambda = 0.8, p = 0.8$	34
5 Barabási–Albert Model Motif and Thij <i>T2</i> Event Motif Dynamics	38
5.1 H3	38
5.2 H4	39
5.3 H5	39
5.4 H6	41
5.5 H7	42
5.6 H8	44
5.7 H9	45

5.8	H10	46
5.9	H11	48
5.10	H12	48
5.11	H13	50
5.12	Summary of Preferential Attachment and $T2$ Event Motif Evolution ...	51
6	Twitter Model Specific Motif Evolution	52
6.1	H3	52
6.2	H4	53
6.3	H5	53
6.4	H6	54
6.5	H7	55
6.6	H8	56
6.7	H9	57
6.8	H10	59
6.9	H11	60
6.10	H12	61
6.11	H13	62
6.12	In summary of the $T3$ events.....	63
7	Motif Correlation	64
8	Dynamic Mode Decomposition	77
8.1	The Koopman Operator	77
8.2	Approximating the Koopman Operator via Dynamic Mode Decomposition.....	78
8.3	Kernel Dynamic Mode Decomposition.....	80
8.4	Accuracy Criterion	82
9	Results	83
9.1	Preprocessing Data.....	83
9.2	DMD: Barabási-Albert $m=1$	83
9.3	KDMD: Barabási-Albert $m=1$	88
9.4	DMD: Barabási-Albert $m=2$	90
9.5	KDMD: Barabási-Albert $m=2$	93
9.6	DMD: Thij Model with $\lambda = 0.2$, $p = 0.2$	95

9.7	KDMD: Thij Model with $\lambda = 0.2, p = 0.2$	98
9.8	DMD: Thij Model with $\lambda = 0.2, p = 0.8$	100
9.9	KDMD: Thij Model with $\lambda = 0.2, p = 0.8$	103
9.10	DMD: Thij Model with $\lambda = 0.8, p = 0.2$	105
9.11	KDMD: Thij Model with $\lambda = 0.8, p = 0.2$	108
9.12	DMD: Thij Model with $\lambda = 0.8, p = 0.8$	110
9.13	KDMD: Thij Model with $\lambda = 0.8, p = 0.8$	113
9.14	Accuracy of Methods	115
10	Discussion	119
11	Conclusion	121
	BIBLIOGRAPHY	122

LIST OF TABLES

	PAGE
5.1 The rows denote counts of isomorphisms that can be found in either motif. The H_3 , a four walk, can be found twice in the modified H_3 in event one by starting the walk at either end. In the graph produced again we can only find two H_3 's.....	39
5.2 Motif Counts of the H_4 motif and the possible motifs given a T_2 event.	39
5.3 Motif counts of the possible T_2 events on the H_5 motif.	41
5.4 Motif counts for variations of the T_2 event on the H_6 motif.	42
5.5 Motif counts of the H_7 motif and the possible additions of T_2 event nodes. .	43
5.6 Motif counts graphs formed by possible T_2 events on the H_8 motif.	45
5.7 Motif counts graphs formed by possible T_2 events on the H_9 motif.	45
5.8 Motifs counts of the possible T_2 event on the H_{10} motif.....	47
5.9 Motif counts of the graphs generated by possible T_2 events on the H_{11} motif.	48
5.10 Variations of the T_2 event on the H_{12} motif.....	50
5.11 Motif counts of the possible T_2 events on the H_{13} motif	51
6.1 Motif counts of the possible T_3 events on the H_3 motif.....	53
6.2 Motif counts of the T_3 event on the H_4 motif	53
6.3 Motif counts of the graphs produced by a T_3 event on the H_5 motif.....	54
6.4 Motif counts of graph produced by a T_3 event on the H_6 motif.	55
6.5 Motif counts of the two possible graphs produced by the T_3 event on the H_7 motif	56
6.6 Motif counts of the possible T_3 event on the H_8 motif.....	56
6.7 Motif counts of the possible T_3 event on the H_9 motif.....	58
6.8 Motif counts of the graphs given by a T_3 event on the H_{10} motif	60
6.9 Motif counts of the single graph produced by a T_3 event on the H_{11} motif..	61
6.10 Motif counts of the possible T_3 event on the H_{12} motif.....	61
6.11 Motif counts of the graph given by a T_3 event on the H_{13} motif.....	63

LIST OF FIGURES

	PAGE
1.1 All the circles represent nodes or users in this Twitter network. The red node represents a user in the Twitter network. This user follows (denoted) by the red edges Tesla, Dogecoin, and Grimes, but no others. The red user receives all tweets and retweets from Tesla, Dogecoin, and Grimes. This user may retweet these messages to outside the network bringing in more users to the network. Note the red node and the nodes it's adjacents are isomorphic to, S_3 or $H4$	4
2.1 A homomorphism from G to H	7
2.2 An isomorphism between G and H	8
2.3 Example of subgraphs and induced subgraphs on G	9
2.4 A simple cycle of length 5.....	9
2.5 The bipartite graph $K_{2,3}$	10
2.6 The star S_4	10
2.7 The first four non-simple motifs. This set is comprised of three-walk ($H3$), a star S_3 ($H4$), the star S_3 with two vertices attached ($H5$), and finally an S_3 with two edges added.	13
3.1 This Barabási–Albert model is initialized with $m = 8$ uniformly connected nodes. At each time-step a new node enters and is attached to $k = 1$ nodes in accordance with the preferential attachment mechanism. Only those motif counts which have a node of degree one can increase given $k = 1$	16
3.2 Statistics characterizing the development of the Barabási–Albert model for $k = 1$. Edges and nodes grow linearly, the number of edges being added is k -times per time-step. We also see as Barabási noted, the edge density decreases asymptotically.	17
3.3 The Barabási–Albert model for $k = 1$ at the final time-step.	18
3.4 This Barabási–Albert model is initialized with $m = 5$ uniformly connected nodes. At each time-step a new node enters and is attached to $k = 2$ nodes. We see that different motifs correlate for the $k = 2$ simulation and other motifs like $H5$ can be generated.....	19

3.5	Statistics characterizing the development of the Barabási–Albert model for $k = 2$. Edges and nodes grow linearly, with m edges added each time-step. As Barabási noted, the edge density decreases asymptotically. The histogram at the final time also shows that the vast majority of nodes, approximately 220, have two or three attachments while three vertices have a degree greater than 35.	20
3.6	The Barabási–Albert model for $k = 2$ at the final time-step.	21
4.1	We see the T_1 , T_2 , and T_3 events on a simple graph. In the T_1 case we see a new message node U_3 appears. It has yet to be connected to anything at all. In T_2 we see a new node V_4 . This is a retweet of the message U_1 . Finally in the last plot we see a T_3 event. Here the user V_3 , who has already retweeted U_2 , now retweets message U_1	24
4.2	Here we see that H_8 's lead with H_9 's and H_{10} 's. These motifs are closely correlated with one another throughout the time series. H_7 's and H_3 's have the next highest counts where H_7 movement <i>prima facie</i> appears to correlate with H_8 counts, while H_3 counts steadily increases throughout time.	25
4.3	Compared to the Barabási–Albert model it's obvious neither edge count nor node count grow strictly linear at each time-step. The edge density still tends toward zero, although a T_3 event would amount to an increase in edge density. Finally, we see a final time degree histogram that is similar.	26
4.4	The network for $\lambda = 0.2$, $p = 0.2$ at the final time-step.	27
4.5	For high λ we see many new message nodes appear (T_1 events), but with low p we should see many T_3 events. H_3 motifs are the most prevalent followed by H_8 's and H_4 's.	28
4.6	The edges and nodes travel tightly together with roughly a ratio of one-to-one. Here we see many, many nodes unattached with degree zero. For those that are attached we do see a power law describing degree distribution, but one that is not quite as strong as those found in other simulations.....	29
4.7	The network for $\lambda = 0.8$, $p = 0.2$ at the final time-step.	30
4.8	Small λ decreases the likelihood of new message nodes appearing, but high p means a greater likelihood of T_2 events which we speculate lead to a large count of H_4 's. C_3 's exist around the center where all these H_4 's overlap. Although C_3 's are not prevalent here, a combinatorial effect generates many H_7 's and H_4 's.	31

4.9	Here, we see two nodes with degrees greater than seventy, but an abundance of nodes with only one or two connections. This suggests a couple (or several) large star graphs focused around two heavily connected nodes. This would produce the abundance of $H8$'s in the motif counts.	32
4.10	The network for $\lambda = 0.2, p = 0.8$ at the final time-step.	33
4.11	Here, like the simulation directly in figure 4.9, we see a prominence of $H4$'s. The scales however are separated by several orders of magnitude. Particular to this simulation we have a relatively high count of $H3$'s. We can explain the difference in magnitude due to many $T1$ events introducing many nodes, but the occurrence of $T2$ events is still sufficient to make $H4$ the motif of highest count.	34
4.12	This Twitter simulation produces many more nodes than edges, because of the frequency of $T1$ events. The vast majority of nodes only have degrees of one or two, while we see a single node with 25 connections. This might suggest many small clusters of nodes with a single larger cluster around a single message node.	35
4.13	The network for $\lambda = 0.8, p = 0.8$ at the final time-step.	36
5.1	The possible graphs generated by adding a node to the $H3$ graph and connecting it to an existing node.	38
5.2	The possible graphs generated by adding a node to the $H4$ graph and connecting it to an existing node.	39
5.3	The possible graphs generated by adding a node to the $H5$ graph and connecting it to an existing node.	40
5.4	The possible graphs generated by adding a node to the $H6$ graph and connecting it to an existing node.	41
5.5	The possibly graphs generated by adding a node to the $H7$ graph and connecting it to an existing node.	43
5.6	The possible graphs generated by adding a node to the $H8$ graph and connecting it to an existing node.	44
5.7	The possible graphs generated by adding a node to the $H9$ graph and connecting it to an existing node.	46
5.8	The possible graphs generated by adding a node to the $H10$ graph and attaching it to an existing node.	47
5.9	The possibly graphs generated by adding a node to the $H11$ graph and connecting it to an existing node.	48
5.10	The possibly graphs generated by adding a node to the $H12$ graph and connecting it to an existing node.	49
5.11	The $H13$ graph and possible node attachments up to symmetry.	50

6.1	The possible graphs generated by adding an edge to the $H3$ graph.....	52
6.2	The possible graphs generated by adding an edge to the $H4$ graph. By connecting the two outer edges we find a C_4 and by connecting an outer vertex to an inner vertex we generate a $H5$	53
6.3	The possible graphs generated by adding an edge to the $H5$ graph.....	54
6.4	The possible graphs generated by adding an edge to the $H6$ graph.....	55
6.5	The possible graphs generated by adding an edge to the $H7$ graph.....	55
6.6	The possible graphs generated by adding an edge to the $H8$ graph.....	57
6.7	The possible graphs generated by adding an edge to the $H9$ graph.....	58
6.8	The possible graphs generated by adding an edge to the $H10$ graph.	59
6.9	The possible graphs generated by adding an edge to the $H11$ graph.	60
6.10	The possible graphs generated by adding an edge to the $H12$ graph.	62
6.11	The possible graph generated by a $T3$ event on the $H13$	63
7.1	Barabási–Albert model with eight initial nodes and $m = 1$. The model is capable of only producing certain motifs due to the limita- tions of attaching a single edge and a single node at every time-step. We also see that $H7$'s and $H8$'s correlate together. However, appear- ances of those motifs depend upon the initialization of the graph itself.	64
7.2	Barabási–Albert model with eight initial nodes and $m = 1$. Here we once again see that the $H7$ and $H8$ motifs have a much higher covariance than any other pair of motifs.	65
7.3	Barabási–Albert model with three initial nodes and $m = 2$. Here we see that all motifs are present. They all have fairly high correlation coefficients, but we see that the $H7$'s and $H8$'s are highly correlated as is $H7$ and $H8$ with $H3$	66
7.4	Barabási–Albert model with three initial nodes and $m = 2$. The covariance offers a different perspective than the correlation figure. We now see that their high covariance between $H7$'s and $H8$'s, and we once again see a relationship between $H8$, $H7$, $H3$, $H4$, but many other motifs have a much, much smaller covariances.	67
7.5	$\lambda = 0.2$ and $p = 0.2$. We recall low λ reduces the chance of only adding a new node ($T1$) and low p reduces the chance of adding a new node and a new edge. Thus probabilistically this simulation should be overall adding only edges between existing nodes.	68

7.6	$\lambda = 0.2$ and $p = 0.2$. Here we see once again a strong covariance relationship between $H7$, $H8$, $H9$, $H10$. In this particular graph it's insightful to see these correlate together given these parameters are less likely to produce the induced star subgraph we expect to see for higher p	69
7.7	$\lambda = 0.2$ and $p = 0.8$. Here in the correlation coefficients we see that there is a strong correlation between $H7$, $H8$, but otherwise a much larger spread over the motifs. We also see that no $H6$ or $H13$ motifs appeared throughout this simulation.	70
7.8	$\lambda = 0.2$ and $p = 0.8$. Here the covariance is relatively very small, except for the $H4$ variance. This is suggestive of the high p generating that induced star subgraph.....	71
7.9	$\lambda = 0.8$ and $p = 0.2$. Once again we see a familiar structure in the correlation matrix. Here though we do see strong correlations across a wider selection of motifs: $C3$, $C4$, $C5$, $H3$, $H4$, $H5$, $H7$, $H8$, $H9$, $H10$. It appears the $H6$'s are not easily produced by any of the parameters chosen or the Barabási–Albert model.	72
7.10	$\lambda = 0.8$ and $p = 0.2$. We do see strong covariance around the $H4$ with other motifs. There is possibly multiple induced star subgraphs connected that could drive this phenomena generating many $H4$'s along with other motifs.	73
7.11	$\lambda = 0.8$ and $p = 0.8$. Here we have a higher likelihood of adding a new node or a new edge and node. Here we do not see any new $H6$'s or $H8$'s produced after the initialization of the graph. Any block structures that we saw in earlier correlation matrices are not as apparent, although there is still strong $H7$ - $H8$ correlation.	74
7.12	$\lambda = 0.8$ and $p = 0.8$. The covariances formed by this simulation suggest once again an overlapping of $H4$'s, but there are clearly more nodes attached to the outer edges of this induced subgraph. This could lead to the covariance we see between $H3$ and $H4$	75
9.1	Smooth motif counts for Barabási–Albert model $k = 1$	85
9.2	Mode Eigenvalues for Barabási–Albert model with $k = 1$	86
9.3	Koopman modes by DMD for Barabási–Albert model with $k = 1$	87
9.4	Mode Eigenvalues by KDMD for Barabási–Albert model with $k = 1$	88
9.5	Koopman modes by KDMD for Barabási–Albert model with $k = 1$	89
9.6	Smooth motif counts for Barabási–Albert model $k = 2$	90
9.7	Koopman modes by DMD for Barabási–Albert model with $k = 2$	91
9.8	Mode eigenvalues by DMD for Barabási–Albert model with $k = 2$	92
9.9	Mode eigenvalues by KDMD for Barabási–Albert model with $k = 2$	93

9.10	Koopman modes by KDMD for Barabási–Albert model with $k = 2$	94
9.11	Smooth motif counts for Thij model with $\lambda = 0.2, p = 0.2$	95
9.12	Mode eigenvalues by DMD for Thij model with $\lambda = 0.2, p = 0.2$	96
9.13	Koopman Modes by DMD for Thij model with $\lambda = 0.2, p = 0.2$	97
9.14	Mode eigenvalues by KDMD for Thij model with $\lambda = 0.2, p = 0.2$	98
9.15	Koopman modes by KDMD for Thij model with $\lambda = 0.2, p = 0.2$	99
9.16	Motif counts for Thij model with $\lambda = 0.2, p = 0.8$	100
9.17	Mode eigenvalues by DMD for Thij model with $\lambda = 0.2, p = 0.8$	101
9.18	Koopman modes by DMD for Thij model with $\lambda = 0.2, p = 0.8$	102
9.19	Mode eigenvalues by KDMD for Thij model with $\lambda = 0.2, p = 0.8$	103
9.20	Koopman modes by KDMD for Thij model with $\lambda = 0.2, p = 0.8$	104
9.21	Motif counts for Thij simulation with $\lambda = 0.8, p = 0.2$	105
9.22	Mode eigenvalues by DMD for Thij model with $\lambda = 0.8, p = 0.2$	106
9.23	Koopman modes and phi modes by DMD for Thij model with $\lambda = 0.8, p = 0.2$	107
9.24	Mode eigenvalues by KDMD for Thij model with $\lambda = 0.8, p = 0.2$	108
9.25	Koopman modes and phi modes by KDMD for Thij model with $\lambda = 0.8, p = 0.2$	109
9.26	Motif counts for Thij model with $\lambda = 0.8, p = 0.8$	110
9.27	Mode eigenvalues by DMD for Thij model with $\lambda = 0.8, p = 0.8$	111
9.28	Koopman modes by DMD for Thij model with $\lambda = 0.8, p = 0.8$	112
9.29	Mode eigenvalues by KDMD for Thij model with $\lambda = 0.8, p = 0.8$	113
9.30	Koopman modes and phi modes by KDMD for Thij model with $\lambda = 0.8, p = 0.8$	114
9.31	In orange have the reconstruction error of the DMD results and in cyan the KDMD results. The DMD performs much better with regard to reconstruction error, by several scales of magnitude.	115
9.32	DMD also seems to produce modes with lower mode error. The scale on this figure is linear, so the differences are not as stark in figure 9.31. The KDMD method for $m = 1$ does seem to perform comparable given that all the modes in KDMD have dynamic behavior.	116
9.33	DMD and KDMD reconstruction errors across Thij simulations. DMD here again has very low reconstruction errors compared to KDMD.....	117

- 9.34 DMD and KDMD mode errors across simulations. DMD does seem to produce better results, but KDMD has the advantage of being able to produce more Koopman modes. The KDMD results for $\lambda = 0.2, p = 0.2$ is a noticeable outlier with very low mode error. 117

ACKNOWLEDGMENTS

I would like to thank Chris Curtis for his invaluable insight and mentorship throughout the making of this thesis. I would also like to thank Professor Chris O'Neill and Professor Xialu Liu for taking part as members of the thesis committee. Finally, a thank you to the Professors of the SDSU Department of Mathematics and Statistics who have encouraged my learning and development in applied mathematics these last several years.

This thesis is partially protected against the evil forces of the Montezuma Publishing thesis reviewers by the magic of the Department of Mathematics and Statistics Master's Thesis L^AT_EX Template.

CHAPTER 1

Network Theory

1.1 An Introduction

The social network has taken on new meaning with the advent of the computer, particularly with mobile technology. Social media allows for information (and misinformation) to spread rapidly within and across communities which can be described by network theory, an extension of graph theory. Individuals and those to whom they are connected form nodes within a given network. These nodes and their connections form patterns in the network. These patterns (motifs) offer a way to characterize the local structure of the network making them informational features. These motif counts change over time as users enter and leave the network. The dynamic behavior of these motif counts and how they correlate with one another offers a way to understand how the network changes locally in time. This analysis can extend easily beyond social networks into other domains. In the following section, we first establish some of the fundamentals of network theory.

1.2 Complex Networks

The terms "network" and "graph" are often used interchangeably as networks are represented via vertices and edges, but networks are graphs in specific context. In its most general sense a network is comprised of objects and connections between those objects. These connections may be directed, undirected, or weighted representing any number of different relationships. For this reason networks are found across many fields: physics, biology, economy, chemistry. Some networks are small and simple. Many networks are large containing many nodes and connections and some of these topologically (or structurally) non-trivial. These we refer to as complex networks and they are commonly found in those fields mentioned above.

Complex networks differ from other networks as edges are found between vertices in patterns neither completely random nor regular. Such networks often have degree distributions that are fat-tailed meaning a few nodes are of relatively high degree, while most other nodes are not. These networks are commonly called scale-free networks. The Barabási–Albert model we examine later will fall under this category. These networks also tend to strongly cluster, which correlates with the scale-free property.

1.3 A Short History

Interest in networks has been steadily increasing over the last seven decades after Paul Erdős and Alfréd Rényi worked to develop the theory of random graphs. This seminal theory described a small class of graphs and networks. These graphs are of single scale and their degree distributions are generally Poisson. Their model of random networks, the Erdős-Renyi graph, models networks of small node count and of simple structure well, but this graph does not exhibit the scale-free property that was found across so many networks in practice [7]. Paul Erdős and Alfréd Rényi's work was absolutely critical to the beginnings of network science, but ultimately failed to capture the intricacies of real networks.

Their successors Réka Albert and Albert Barabási recognized that the Erdős–Rényi model was incapable of modeling real-world phenomena because the Erdős–Rényi model does not add in new nodes nor edges through a temporal process. All edges in the Erdős–Rényi model are added at random according to a parameter value, but complex networks are not entirely random. They conform to rules guiding the hand of chance. In three key areas the real networks differed from those of the Erdős–Rényi graph [7]. First, the real networks often had degree distributions that are not explainable according to a Poisson distribution. The tail of the Poisson distribution did not allow for the observed degree distributions. Second, real networks had a sizable largest connected component - a large cluster of nodes inside the network, forming a hub of activity. Finally, the local clustering coefficient of most networks decreases as the node degree decreases, but is independent of overall graph size. For the Erdős–Rényi model the local clustering coefficient is independent of the node degree and the average local clustering coefficient depends on the size of the whole network [7].

1.4 Network Analysis

Network science has many statistical measures to differentiate networks from one another. These measures offer different levels of insight into a network and it's strucuture. One such measure is the notion of centrality. There are several different types of centrality, but each represents a way to denote the most important vertices within a given network. One such measure is degree centrality. Aptly named, degree centrality assigns a weight to each vertex determined by its degree d_i . In chapter 3 and chapter 4, degree centrality is useful as the preferential attachment model's dynamics directly depend on it. The centrality measures are informational about the connectivity of the network, but leave much to be desired in the way of understanding structure.

We also make use of the clustering coefficient in chapters 3 and 4 to characterize graph dynamics. Thij [31] notes that clustering coefficients are a way to understand how a network's density changes over time. He further notes that future study of the proposed Twitter model found should include an analysis of it's temporal clustering coefficients. To define the clustering coefficient we first define the neighborhood of a vertex N_i .

$$N_i = \{v_j : e_{i,j} \in E \vee e_{j,i} \in E\}$$

The local clustering coefficient of a node v_i in the undirected graph G is defined as

$$C_i = \frac{2|p_i|}{k_i(k_i - 1)}$$

where we define p_i

$$p_i = \{e_{jk} : e_j, e_k \in N_i, e_{j,k} \in E\}$$

This can also be calculated by way of the adjacency matrix.

$$C_i = \frac{1}{k_i(k_i - 1)} \sum_{j,k} a_{i,j} a_{j,k} a_{i,k}$$

The average clustering coefficient is as expected.

$$\bar{C} = \frac{1}{|V|} \sum_i C_i$$

The global clustering coefficient is simply

$$C_G = \frac{\text{Total Number of Triangles}}{\text{Total Possible Triangles}}$$

Clustering allows us to understand how dense a network is relative to a complete graph.

1.5 What is Twitter?

Twitter is now, as of 2021, an eminent example of complex networks in social media. As of 2019 there are 330 million monthly active users on Twitter, thirteen years after the platform was established in 2006. However, according to Pew Research, the top ten percent of Twitter users tweet 138 tweets per month, while the bottom ninety percent of Twitter users only tweet twice per month. The top ten percent of users have a median 387 followers while the bottom ninety percent have only 19 median followers. The top ten percent also follow more users, a median of 456 accounts, compared to a median of 74 accounts for the bottom ninety percent [34]. This itself is suggestive of a

Twitter being a scale-free network, but analysis confirms this [5]. Twitter, as a network with vertices as users and edges as signifying one user following another, follows a fat-tailed degree distribution. Moreover, its estimated average clustering coefficient is 256,000 times larger than what is expected for a random graph.

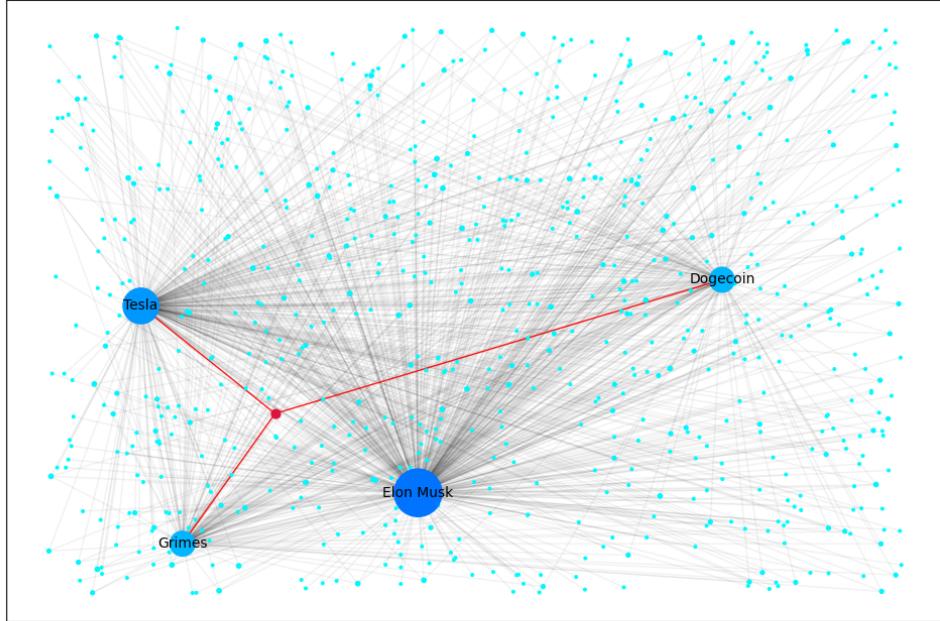


Figure 1.1. All the circles represent nodes or users in this Twitter network. The red node represents a user in the Twitter network. This user follows (denoted) by the red edges Tesla, Dogecoin, and Grimes, but no others. The red user receive all tweets and retweets from Tesla, Dogecoin, and Grimes. This user may retweet these messages to outside the network bringing in more users to the network. Note the red node and the nodes it's adjacents are isomporhic to, S_3 or $H4$.

Twitter's status as a complex network aside, the platform is also notable for its position among social networks not only for its size, but also its influence on culture and politics. During the 2016 election, a survey of thirty million tweets from two million users linked to articles that were found to be spreading false information. Moreover, this information spread based on community structure within an inclusive left-right influencer network. Twitter is clique-y and the communities form around

shared interests. In online communities where exposure to the same memetic theme occurs frequently, facts and rumors may spread easily and quickly [8].

Twitter also has substantial economic impact. Twitter sentiment is known to precede fluctuations in the stock market [11] and Bitcoin prices [30]. However only a handful of Twitter users actually have influence although groups of people could theoretically be enough to influence market prices [28]. Elon Musk is one such person who, as of March 2021, has influenced multiple markets driving Tesla's share price up and down [16], as well as causing jumps in cryptocurrency prices [13] [14].

Twitter's suitability as a subject of network science is clear. One can generate networks from Twitter in a handful of ways. First, there are the user accounts which are linked to one another through followers and follows, in-degrees and out-degrees. One can also construct network representations of network messages as discussed in chapter 4. A user posts a message, which is then retweeted by a portion of their followers, which is then again retweeted by a portion of their own followers, and so on. It is this latter case we will consider in the Thij model.

CHAPTER 2

Motifs

Motifs are the primary object of interest as a way of characterizing the local structure of a network. The dynamic motif count indicates the network is evolving according to certain rules. In the context of static networks, the frequency of motifs has been shown to highlight system properties in biological networks [27] [1].

Motifs are the fundamental components of complex systems. The topological structure of complex networks is tied to the frequency and distribution of their motifs. Initially, the counting of these motifs was treated as a static problem where the frequencies of motifs affect functions on networks [21]. As the significance of motifs becomes more apparent authors have begun examining the emergence of motifs via temporal edges [25]. Our motifs will act as a vector of features characterizing the network at each point in time. Before we examine our specific motifs we must first define common objects of graph theory such as cycles, paths, and stars which will aid us in understanding the motifs we would like to examine.

2.1 Common Objects of Graph Theory

In the pursuit of counting our particular motifs we require some ideas from graph theory. There is no reason some of these objects could not be motifs on their own. We will count three-cycles and four-cycles as motifs themselves, but for this we require the non-simple cycle motifs. Much of the literature on motifs focuses of directed triangle motifs [25] [23], but we the motifs described in this thesis are more varied. Motifs are generally best described by graphs.

Definition 1. Let $G = (V, E)$ be a graph with V being a set of vertices (or nodes), and E , a set of edges. If v is a vertex of G we write $v \in V(G)$. If $u, v \in V(G)$ and there is an edge between them we write $\{u, v\} \in E(G)$

Graphs may be directed or undirected. For directed graphs $\{u, v\} \in E(G)$ is taken to mean there exists an edge from u to v in G . Undirected meaning the edge between u and v to have no notion of direction.

Definition 2. The adjacency matrix for any graph G with $n = |V|$ vertices is a matrix of size $n \times n$. The element a_{ij} is defined to be

$$a_{ij} = \begin{cases} 1 & e_{ij} \in E(G) \\ 0 & \text{otherwise} \end{cases}$$

The adjacency matrix is critical to all of our calculations to come. It renders any graph amenable to the tools of linear algebra.

Definition 3. We call $f : G \rightarrow H$ a homomorphism, if f maps endpoints in $G = (V(G), E(G))$ to endpoints in $H = (V(H), E(H))$. i.e.
 $\forall u, v \in V(G) \quad \{u, v\} \in E(G) \Rightarrow \{f(u), f(v)\} \in E(H)$.

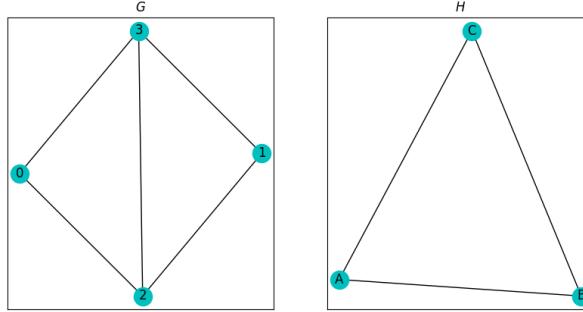


Figure 2.1. A homomorphism from G to H.

For the example in Figure 2.1 we can define a mapping f such that:

$$\begin{aligned} f(0) &= A \\ f(1) &= A \\ f(2) &= B \\ f(3) &= C \end{aligned}$$

f is a homomorphism by the definition presented in definition 3. We now define particular forms of the graph homomorphism.

Definition 4. We call $f : G \rightarrow H$ an isomorphism, if f is a homomorphism and is bijective.

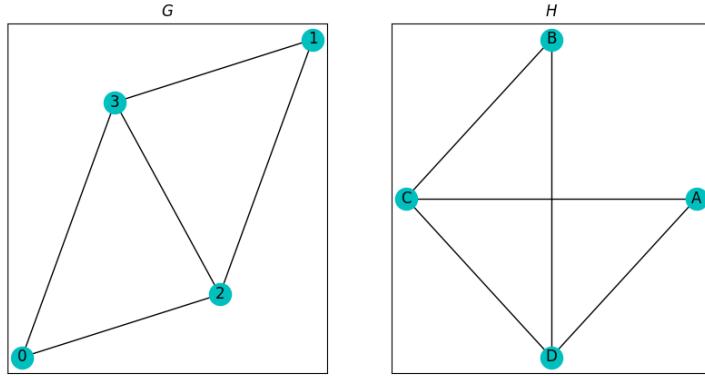


Figure 2.2. An isomorphism between G and H .

We can define an isomorphism f for between G and H in the figure 2.2 such that:

$$\begin{aligned} g(0) &= B \\ g(1) &= A \\ g(2) &= C \\ g(3) &= D \end{aligned}$$

Definition 5. An automorphism is an isomorphism between a graph G and itself. The automorphism is edge-preserving, $\{u, v\} \in G \implies \{f(v), f(u)\} \in G$.

Definition 6. Let $G = (V, E)$ be a graph. We call $G' = (V', E')$ a subgraph if $V' \subseteq V \wedge E' \subseteq E \cap (V' \times V')$. Furthermore we call G' an induced subgraph of G if for the edges $\{u, v\}$ of G' we have $\{u, v\} \in E, u, v \in V'$.

Induced subgraphs are vital to our understanding of how motifs interact as we add edges or nodes to a given graph.

Definition 7. Let $G'' \subset G$ and furthermore let there exist an isomorphism between G'' and G' . We call G'' an appearance of G' . Provided the number of appearances of G' is greater than some N we call G' a motif or pattern.

Definition 8. A walk $W = v_0, e_1, v_1, \dots, v_n$ is a sequence of edges and vertices of G such that for $0 \leq k \leq n - 1$ the edge $e_i = \{v_k, v_{k+1}\}$.

Definition 9. A cycle is a walk whose first and last vertex are the same.

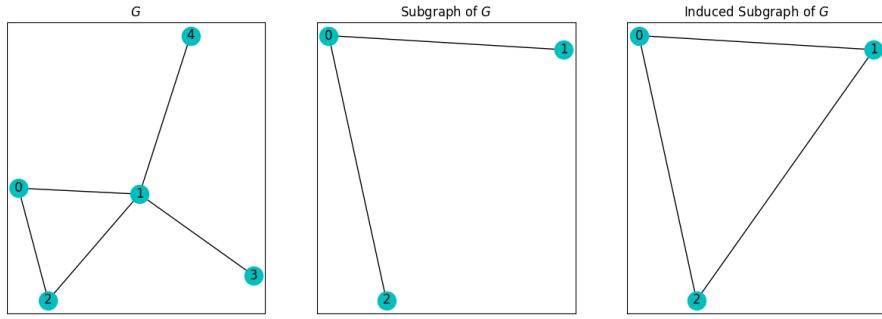


Figure 2.3. Example of subgraphs and induced subgraphs on G .

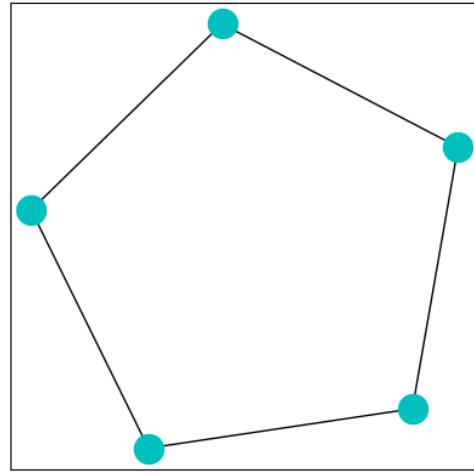


Figure 2.4. A simple cycle of length 5.

We will wish to count cycles of length 3 and 4 which will be used in the counting of more complicated motifs.

Definition 10. *A bipartite graph is a graph whose nodes may be separated into two disjoint sets U and V such that there exists an edge between all vertices in U and all vertices in V .*

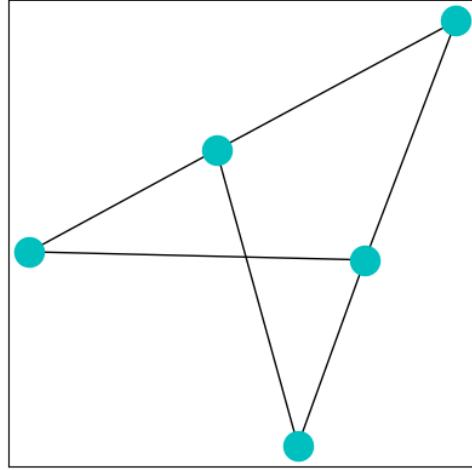


Figure 2.5. The bipartite graph $K_{2,3}$.

Definition 11. A star S_n denotes the complete bipartite graph $K_{i,k}$. In other words a tree with one internal node, but k branches.

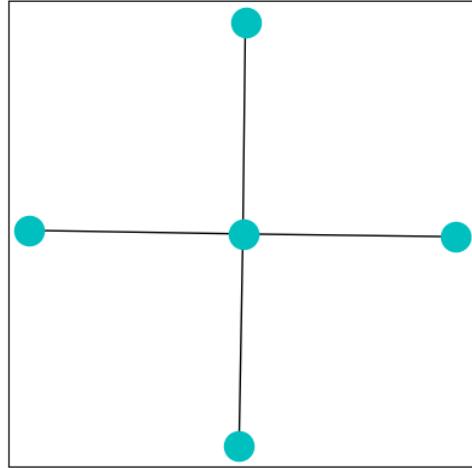


Figure 2.6. The star S_4 .

The $C3$ and $C4$ motif counts are themselves a measurement of clustering the graph. In fact the global clustering coefficient defined in chapter one explicitly makes use of the $C3$ count in its calculation. The notion of a bipartite graph comes in handy

when characterizing some of our motifs, but no explicit bipartite graph is counted as a motif. However, the definition of a star graph S_k is very useful, because some of the motif counts increase exponentially due to the motifs that are generated as a new node or a new edge is added to the S_k graph.

2.2 Non-Simple Cycle Motifs

We wish to consider the following motifs: H_3 through H_{13} . These motifs and the ways in which to count them are described in [4].

Definition 12. *Let f be a homomorphism between graphs G and H . We say that H is a homomorphic image of G provided f is surjective.*

Definition 13. *A graph $H = (V_H, E_H)$ is said to be k -cyclic, for $k > 3$, if it is a homomorphic image of the cycle C_k . The number of different homomorphisms from C_k to H is denoted by $C_k(H)$. H is k -cyclic if and only if $C_k(H) > 0$.*

These motifs are not simple cycles. For the motifs in figures 2.7 we can classify them according to which cycle each graph is a homomorphic image. H_3 , H_4 , H_6 , H_9 , and H_{11} are all six-cyclic. H_5 is the only five-cyclic graph. However H_5 , H_6 , H_7 , H_8 , H_{10} , H_{12} , and H_{13} are all seven-cyclic.

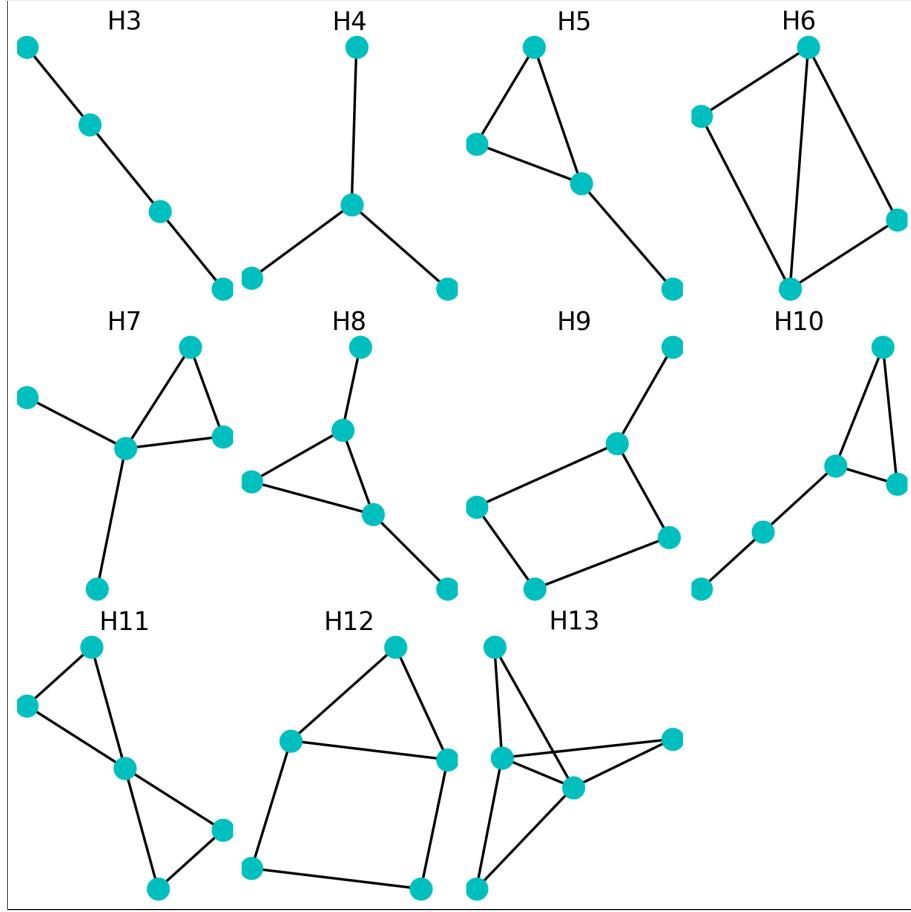


Figure 2.7. The first four non-simple motifs. This set is comprised of three-walk (H_3), a star S_3 (H_4), the star S_3 with two vertices attached (H_5), and finally an S_3 with two edges added.

We will also consider walks of certain sizes as this will be necessary to count appearances at each time. Motif counts can be generated from a network's adjacency matrix in the following ways. Let A be the adjacency matrix of some arbitrary network with greater than four nodes. Let $N_m(A)$ denote the total count of motif m in the network.

When counting the motifs we let E denote the set of edges in the network, e_i, j denotes a particular edge between the i 'th and j 'th nodes. d_i denotes the degree of the i 'th node. A is the adjacency matrix of the graph G . Finally, $a_{i,j}^{(k)}$ denotes the k th power of the matrix element at the i 'th row and j 'th column of matrix A . The formulae for motif counts are as follows:

$$\begin{aligned}
N_G(C3) &= \frac{1}{6} \text{tr}(A^3) \\
N_G(C4) &= \frac{1}{8} \left(\text{tr}(A^4) - 4 \sum_{i=1}^n \binom{d_i}{2} - 2 \sum_{i,j \in E} e_{ij} \right) \\
N_G(C5) &= \frac{1}{2} \sum_{(i,j) \in E} (d_i - 1)(d_j - 1) - 3N_{C3} \\
N_G(H4) &= \sum_{i=1}^n \binom{d_i}{3} \\
N_G(H5) &= \frac{1}{2} \sum_{i=1}^n a_{ii}^{(3)}(d_i - 2) \\
N_G(H6) &= \sum_{(i,j) \in E} \binom{a_{ij}^{(2)}}{2} \\
N_G(H7) &= \frac{1}{2} \sum_{i=1}^n a_{ii}^{(3)} \binom{d_i - 2}{2} \\
N_G(H8) &= \sum_{(i,j) \in E} a_{ij}^{(2)}(d_i - 2)(d_j - 2) - 2N_G(H6) \\
N_G(H9) &= \sum_{i=1}^n (d_i - 2) \sum_{j \neq i} \binom{a_{ij}^{(2)}}{2} \\
N_G(H10) &= \frac{1}{2} \sum_{i=1}^n a_{ii}^{(3)} \sum_{j \neq i} a_{ij}^{(2)} - 3N_G(C3) - 2N_G(H5) - 4N_G(H6) \\
N_G(H11) &= \sum_{i=1}^n \binom{\frac{1}{2} a_{ii}^{(3)}}{2} - 2N_G(H6) \\
N_G(H12) &= \sum_{(i,j) \in E} a_{ij}^{(2)} a_{ij}^{(3)} - 9N_G(C3) - 2N_G(H5) - 4N_G(H6) \\
N_G(H13) &= \sum_{(i,j) \in E} \binom{a_{ij}^{(2)}}{3}
\end{aligned}$$

We want to understand how each of the motif counts affect another given a certain event occurs in any of the simulations. Some motifs contain induced subgraphs of the others and simple changes to a motif may cause a combinatorial effect or multiplicative of new motifs appearing.

CHAPTER 3

Barabási–Albert Model

Réka Albert and Albert Barabási developed a theory of complex networks that encompasses those networks we find common in practice. They contended that ultimately the Erdős–Rényi model failed to account for the growth of networks over time. The addition of new nodes and new edges into the network generated the Netorks' defining characteristics. Edges were also found to obey the preferential attachment mechanism which ascribes probabilities of attachment based upon the relative degree of the nodes in a network. This model was found to simulate scale-free networks well as the model's degree distribution is described by a power-law. A network is scale-free provided the degree distribution of the network follows a power law.

The fraction of nodes P having N nodes attached for large values of k is

$$P(N) \approx k^{-\gamma}, \quad 2 < \gamma < 3$$

We first begin at $t = 0$ by initializing a m number of nodes and randomly distribute a number of edges between them according to a uniform distribution. Then at each time-step $t > 0$, we introduce a new node and a k number of edges between that node and existing nodes. We assign probabilities of attachment via the probability distribution:

$$P(n) = \frac{d_n}{\sum_{i=1}^N d_i}$$

where d_i is the degree of the i 'th node and N is the total number of nodes at $t - 1$. This means that nodes which have a high degree are probabilistically more likely to be attached to be new nodes.

The Barabási–Albert model exhibits a few characteristics that differentiate it from the Erdős–Rényi model. If we consider its diameter, the maximum distance in the network for $m > 1$ and sufficiently large time we can write the diameter

$$\text{diam}(G) = \frac{\ln(n)}{\ln(\ln(N))}$$

The diameter grows slower than $\ln(n)$ meaning the graph's diameter grows slower than that of the Erdős–Rényi model. We can also examine the clustering

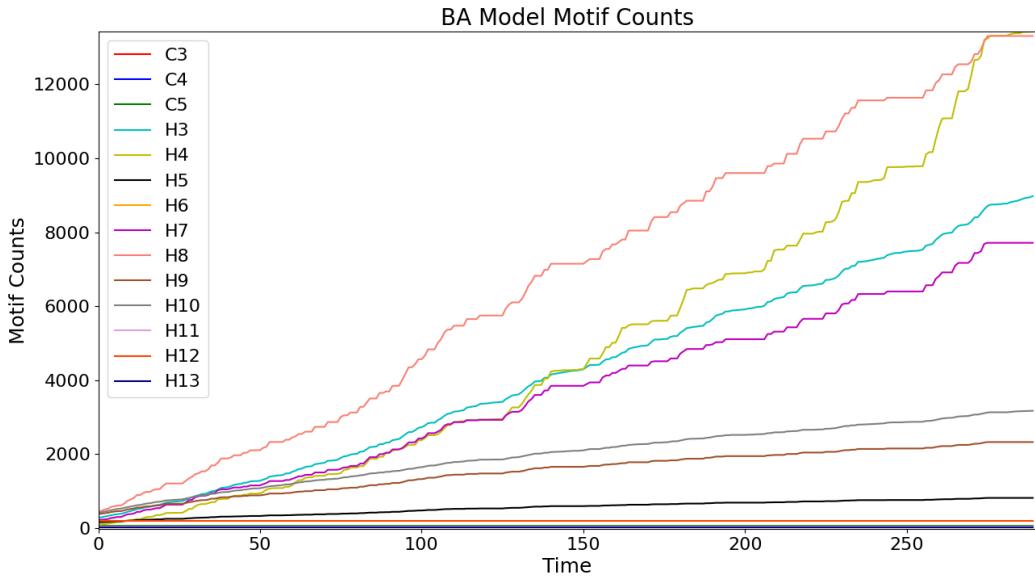


Figure 3.1. This Barabási–Albert model is initialized with $m = 8$ uniformly connected nodes. At each time-step a new node enters and is attached to $k = 1$ nodes in accordance with the preferential attachment mechanism. Only those motif counts which have a node of degree one can increase given $k = 1$.

coefficient of the model. The clustering coefficients for the preferential attachment model grows according to

$$C(G) = \frac{(\ln N)^2}{\ln(N)}$$

This differs from the Erdős–Rényi model by the term $(\ln(N))^2$, which increases the clustering coefficient for large N . The Barabási-Albert network is locally more clustered than a random network.

Motif counts in the Barabási–Albert model are dependent upon the initialization of the nodes and the choice of m . For example, in $m = 1$ the model cannot complete any new cycles. If the intial graph at $t = 0$ does not contain any C_3 or C_4 appearances then any motif which has an induced subgraph isomorphic to C_3 or C_4 cannot appear. The BA model for $k = 1$ can only attach nodes to the existing C_3 or C_4 appearances and in that way generate new $H7$'s, $H8$'s, $H9$'s.

The $k = 2$ BA model is capable of generating those new C_3 and C_4 appearances in the network quite easily. The radically changes the motif composition as we can see in figures BLANK.

The model is a good candidate to compare to the Thij model, given that the Thij model incorporates a preferential attachment mechanism itself, but is slightly more

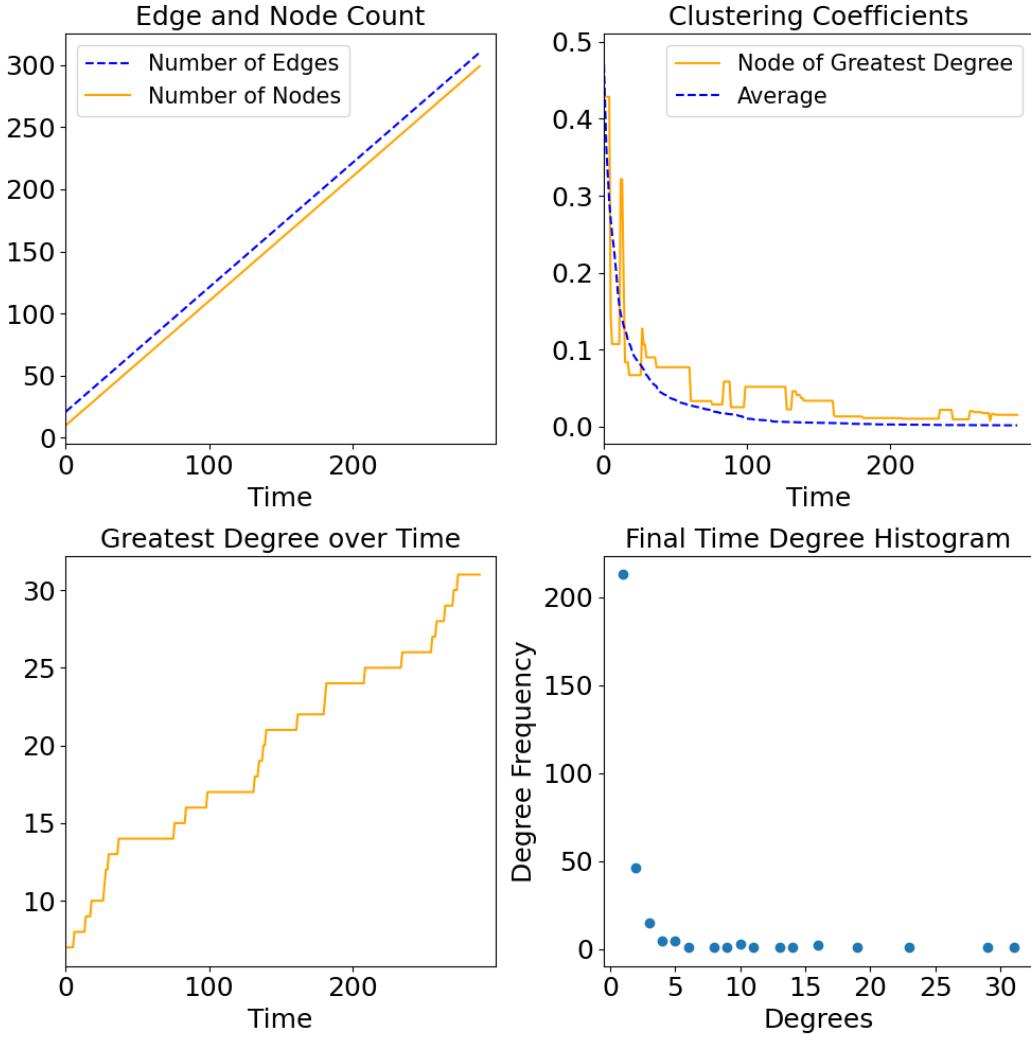


Figure 3.2. Statistics characterizing the development of the Barabási–Albert model for $k = 1$. Edges and nodes grow linearly, the number of edges being added is k -times per time-step. We also see as Barabási noted, the edge density decreases asymptotically.

complex. The Barabási–Albert model represents a simpler, non-trivial model which is widely acknowledged as a useful tool for understanding networks across a variety of disciplines. We should note the limitations to the Barabási–Albert model. The nodes and edges are restricted to growing linear which may fail to capture certain phenomena that appear empirically. The model is also incapable of removing nodes, a point on which Barabási himself has elaborated. The following model does account for these limitations at the cost of increased complexity.

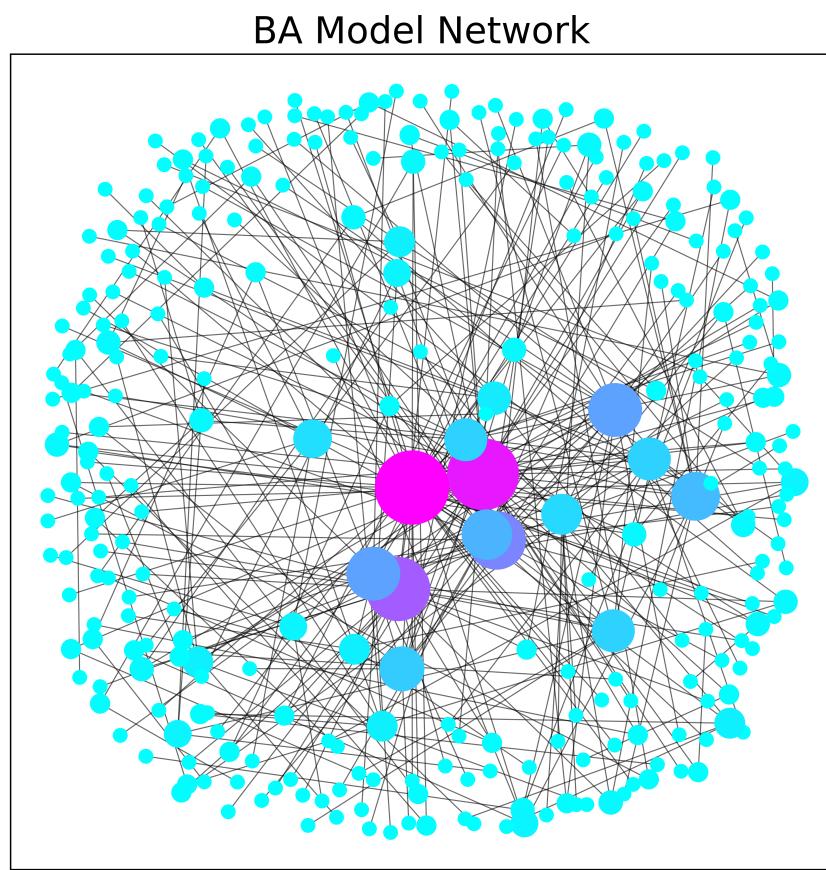


Figure 3.3. The Barabási–Albert model for $k = 1$ at the final time-step.

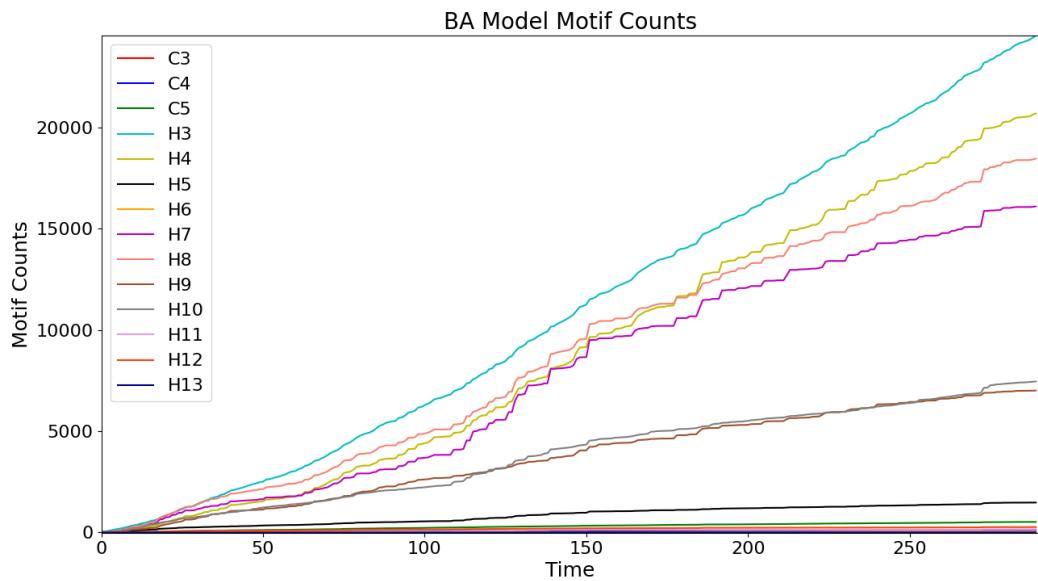


Figure 3.4. This Barabási–Albert model is initialized with $m = 5$ uniformly connected nodes. At each time-step a new node enters and is attached to $k = 2$ nodes. We see that different motifs correlate for the $k = 2$ simulation and other motifs like H_5 can be generated.

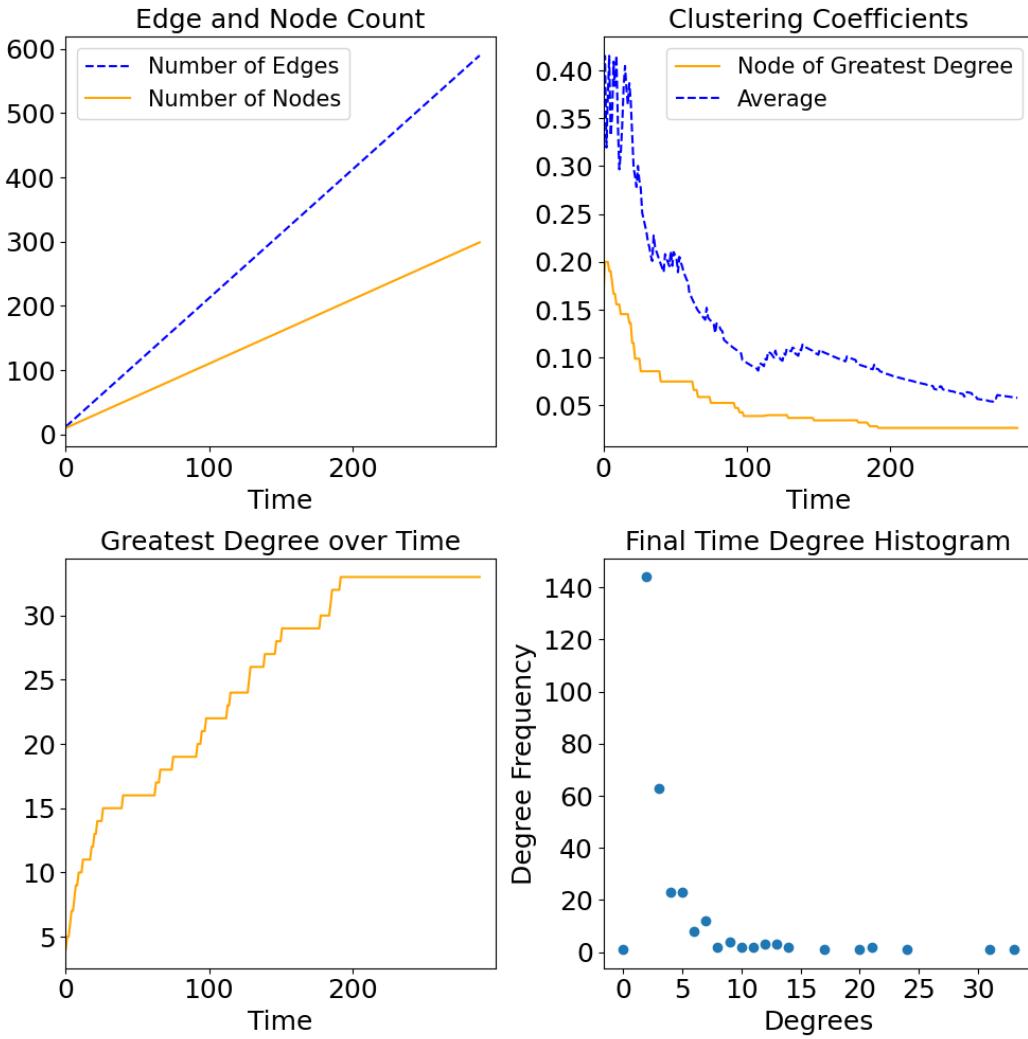


Figure 3.5. Statistics characterizing the development of the Barabási–Albert model for $k = 2$. Edges and nodes grow linearly, with m edges added each time-step. As Barabási noted, the edge density decreases asymptotically. The histogram at the final time also shows that the vast majority of nodes, approximately 220, have two or three attachments while three vertices have a degree greater than 35.

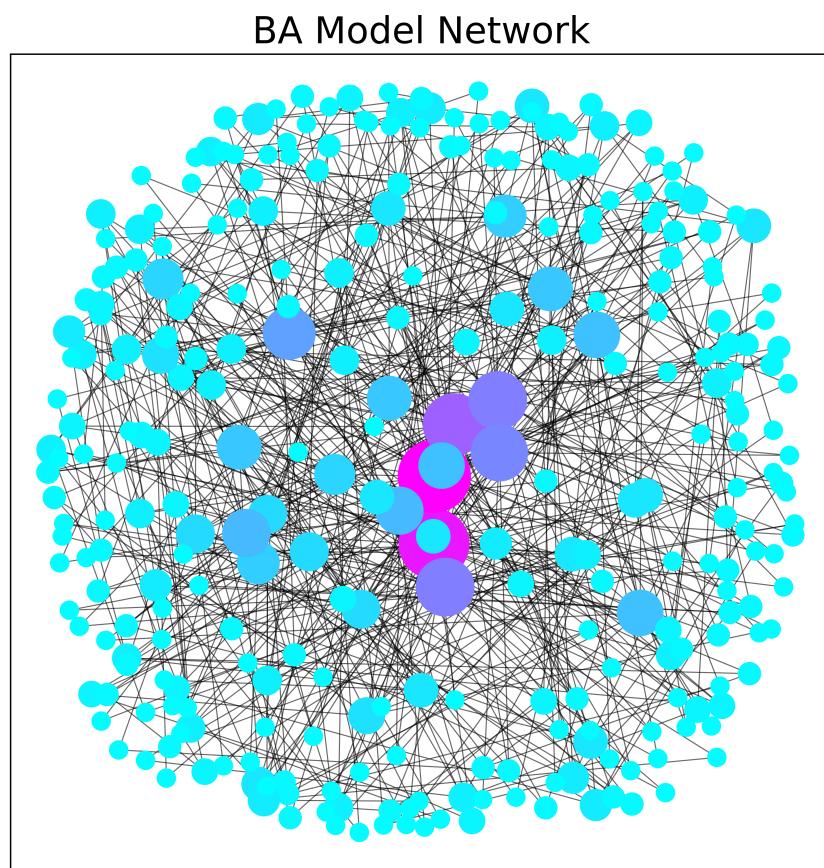


Figure 3.6. The Barabási–Albert model for $k = 2$ at the final time-step.

CHAPTER 4

Twitter and the Thij Model

The Thij model is a particular random graph that seeks to model the development of a Twitter network [31]. Twitter offers a platform where a user may post a message to all of their followers' feeds. Those followers may then ignore, like, reply (comment), or retweet (quote tweet) that message. In the instance they choose to retweet the message, that message is then posted to their respective feed and their set of followers now have the opportunity to then retweet that same message. It is easy to infer that if a message has received a significant number of retweets to a substantial number of followers then the message is more likely to be seen, and thus retweeted again. Intuitively we suspect a preferential attachment mechanism driving popularity of the most popular tweets.

A user can retweet different original messages at different times meaning edges are generated between existing nodes in the network. Accounting for this one can produce a model better suited to Twitter than the Barabási–Albert model described in Chapter 3.

We wish to simulate a network of retweets. There are the original message nodes from users u_i and retweets from users v_i . All the descendants of u_i (all retweets of the original message and retweets of retweets) we loosely call a message tree. v_i may be a simple retweet or a quote tweet meaning v_i may reasonably retweet v_k . We begin with an initial message node from user u_0 at time $t = 0$. There is now the possibility of three events: $T1$, $T2$, and $T3$.

T1: A new message node from user u_t appears.

T2: A user v_t enters the retweet network and retweets an existing user, either a u_i who has posted an original message or v_i with $i \neq t$ who has simply retweeted another user in the network. This user v_t retweets an original message node u_i with probability q and any other node with probability $\frac{1-q}{N}$, N being the total number of nodes in the network.

T3: An existing user v_i retweets another existing user u_i or v_k . Once again the retweeter retweets a message node with probability q and all other message or retweet nodes with probability $\frac{1-q}{N}$.

We must also note that, generally, there will be multiple message nodes in the retweet graph, and thus, we have to decide for any given event which particular message node should be assigned the q probability. Here we introduce a preferential attachment mechanism. At any time t , given a $T2$ or $T3$ event occurring, the probability of a particular message node being chosen is almost the same mechanism described in the Barabási–Albert model. Instead, a message node is chosen by its total descendants and not by the degree of the message node itself. In essence, a message node is selected based upon the number of those who have retweeted the message and all those who have retweeted retweets of that message.

We now assign probabilities of $T1$, $T2$, and $T3$ events. Let $\lambda \geq 0$ and $1 \geq p \geq 0$,

$$\begin{aligned} P(T1) &= \frac{\lambda}{1 + \lambda} \\ P(T2) &= \frac{p}{1 + \lambda} \\ P(T3) &= \frac{1 - p}{1 + \lambda} \end{aligned}$$

Our choices of λ and p will drastically affect the dynamics of the graph, as well as the subgraph motifs that make up its structure. We want to consider a series of cases for different probabilities allowing us to make informed predictions about the development of the graph over time.

The motif counts for various parameters help demonstrate the differences in scale and structure which arise from different probability distributions. Consider $\lambda > 0.5$ and $p > 0.5$ we see an increased chance of adding many new root messages, but still good possibility of introducing a new node with a new edge, or a new edge between existing nodes. $0 < \lambda < 0.5$ and $p > 0.5$ means that adding a new node with an edge is the most likely outcome. The superstar probability q means that these are very likely to attach to a root message node.

The preferential attachment mechanism then encourages an induced subgraph S_k to form with $k \gg 1$. For any $k \geq 3$ S_k itself has many appearances of $H4$ and for every increase in k we see an increase in the probability of yet another increase in k .

Next, consider $\lambda > 0.5$ and $0.5 > p > 0$. Here the $T1$ event should dominate the dynamics with many new message nodes introduced to the overall retweet network. However, these message nodes may only contribute to the overall motif counts provided more than 2 $T2$ events attach nodes to them or they become attached to some other node through a $T3$ event. Last we consider the case when $0.5 > \lambda > 0$ and $0.5 > p > 0$. In this case, it is the $T3$ expected to be the most prevalent. We expect connections,

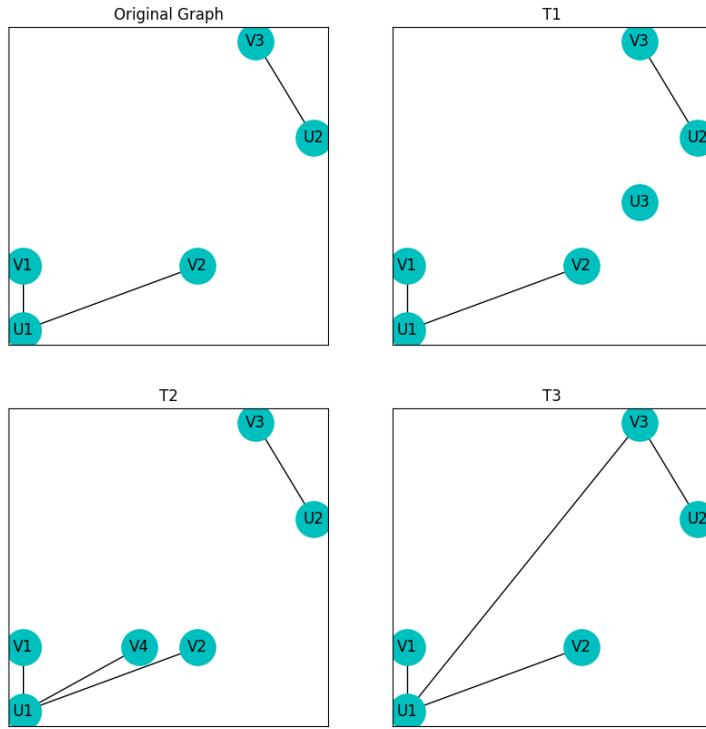


Figure 4.1. We see the T_1 , T_2 , and T_3 events on a simple graph. In the T_1 case we see a new message node U_3 appears. It has yet to be connected to anything at all. In T_2 we see a new node V_4 . This is a retweet of the message U_1 . Finally in the last plot we see a T_3 event. Here the user V_3 , who has already retweeted U_2 , now retweets message U_1 .

governed by the attachment mechanisms, to form between nodes. Given the superstar attachment nodes should overwhelmingly be attached to at least one message node, and may attach to each other thus forming a simple three-walk. This generates many $H7$'s and $H8$'s. If we examine the graph's development for a few different pairs of parameters, we can gain some statistical insight that will inform how each graph is developing.

4.1 $\lambda = 0.2, p = 0.2$

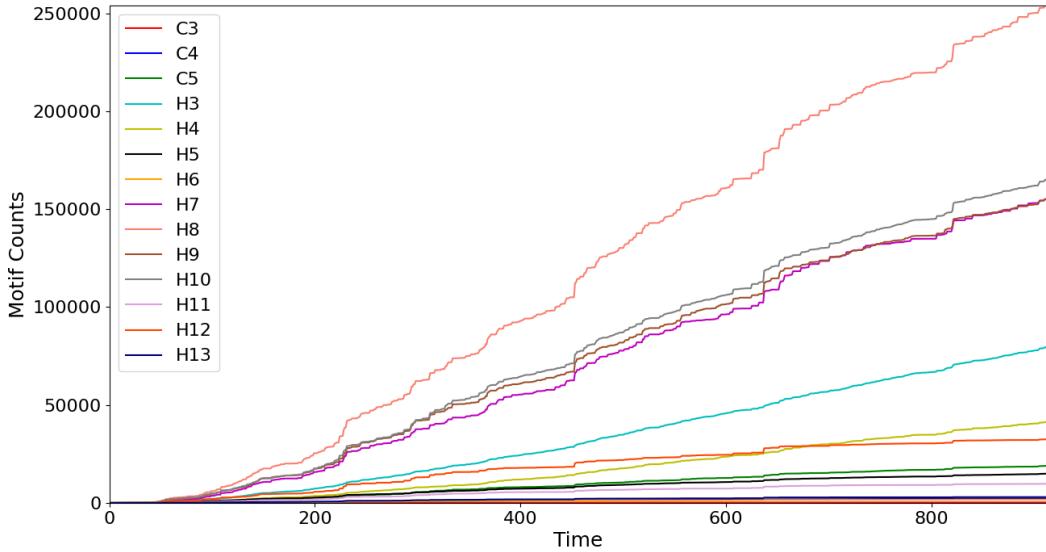


Figure 4.2. Here we see that H_8 's lead with H_9 's and H_{10} 's. These motifs are closely correlated with one another throughout the time series. H_7 's and H_3 's have the next highest counts where H_7 movement *prima facie* appears to correlate with H_8 counts, while H_3 counts steadily increases throughout time.

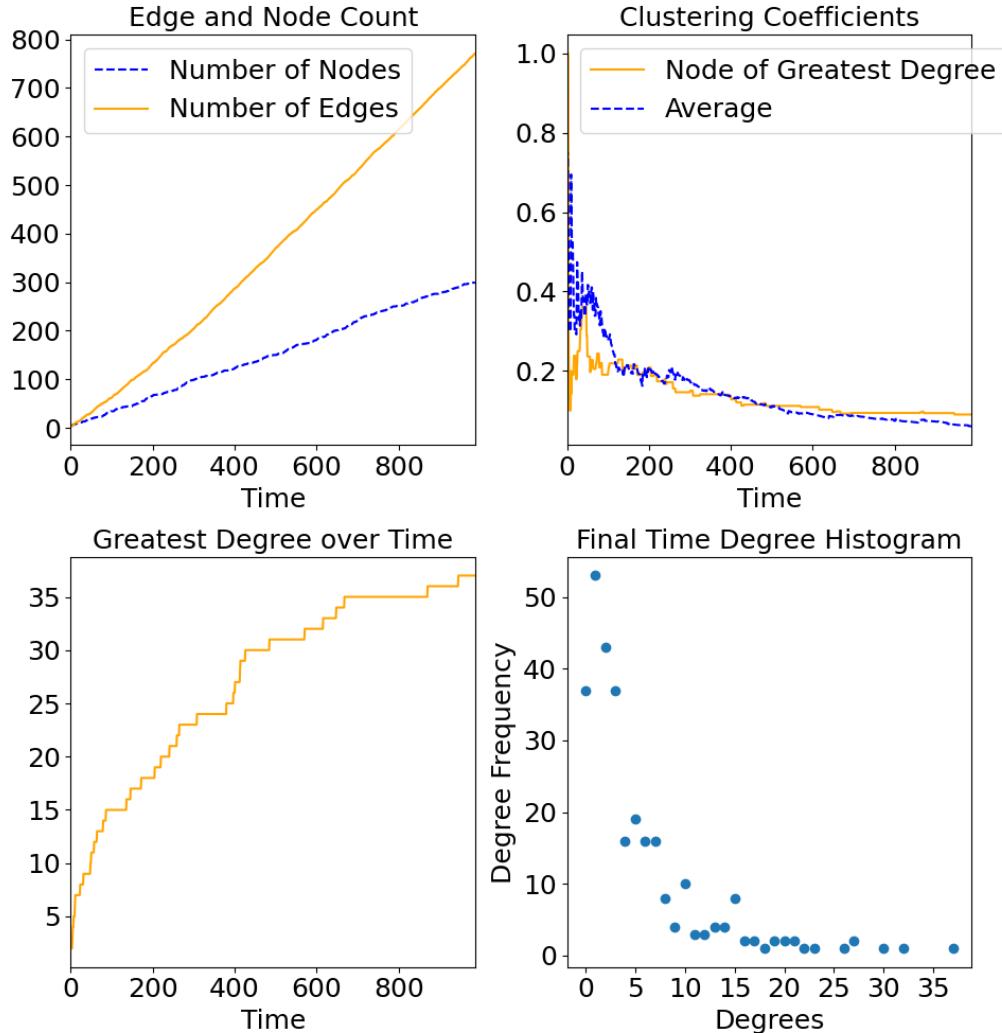


Figure 4.3. Compared to the Barabási–Albert model it's obvious neither edge count nor node count grow strictly linear at each time-step. The edge density still tends toward zero, although a T_3 event would amount to an increase in edge density. Finally, we see a final time degree histogram that is similar.

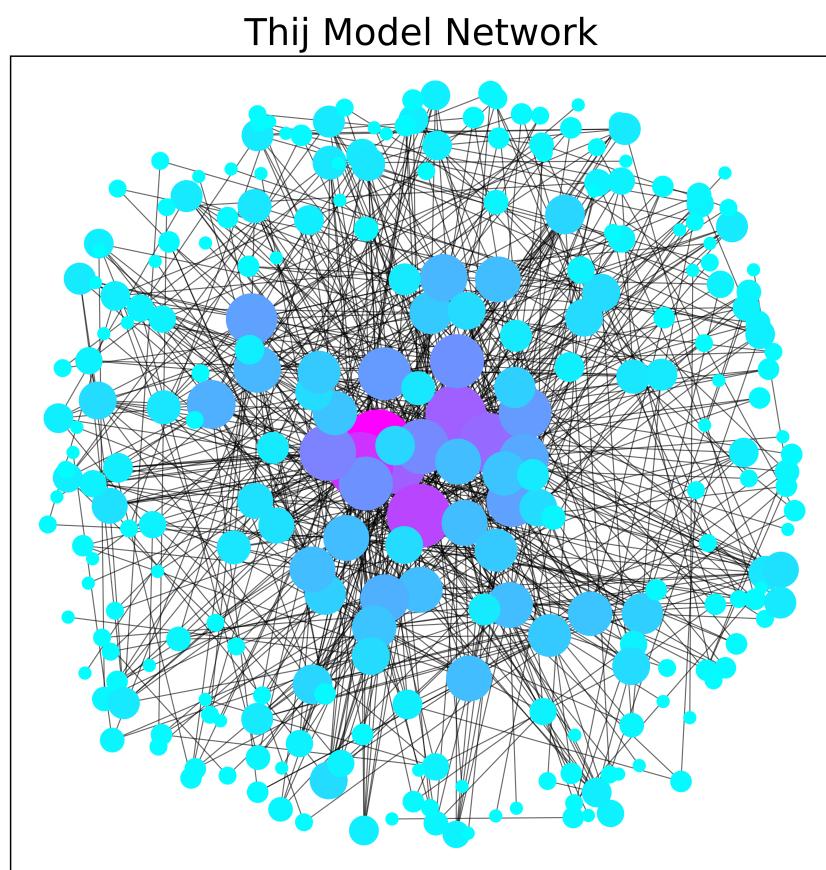


Figure 4.4. The network for $\lambda = 0.2$, $p = 0.2$ at the final time-step.

4.2 $\lambda = 0.8, p = 0.2$

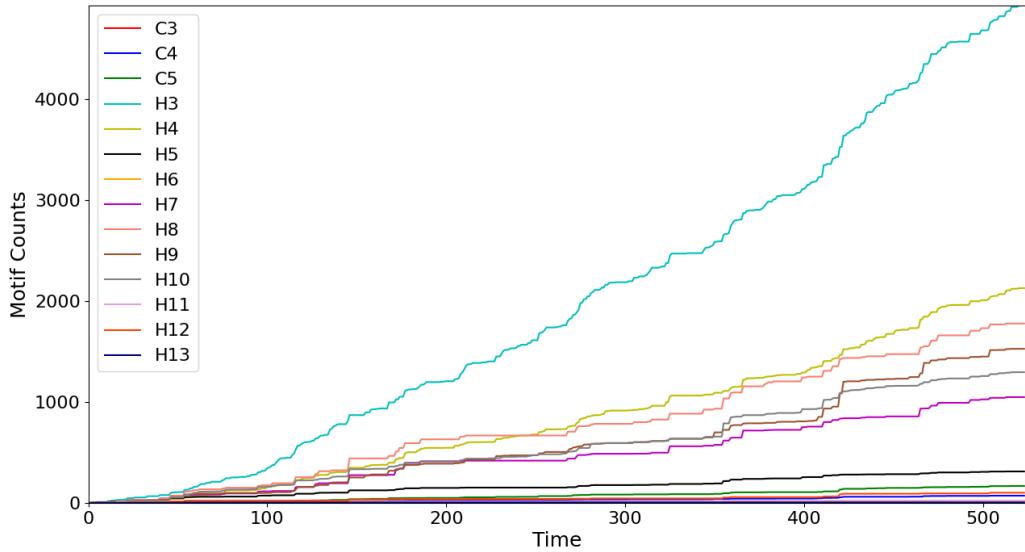


Figure 4.5. For high λ we see many new message nodes appear (T_1 events), but with low p we should see many T_3 events. H_3 motifs are the most prevalent followed by H_8 's and H_4 's.

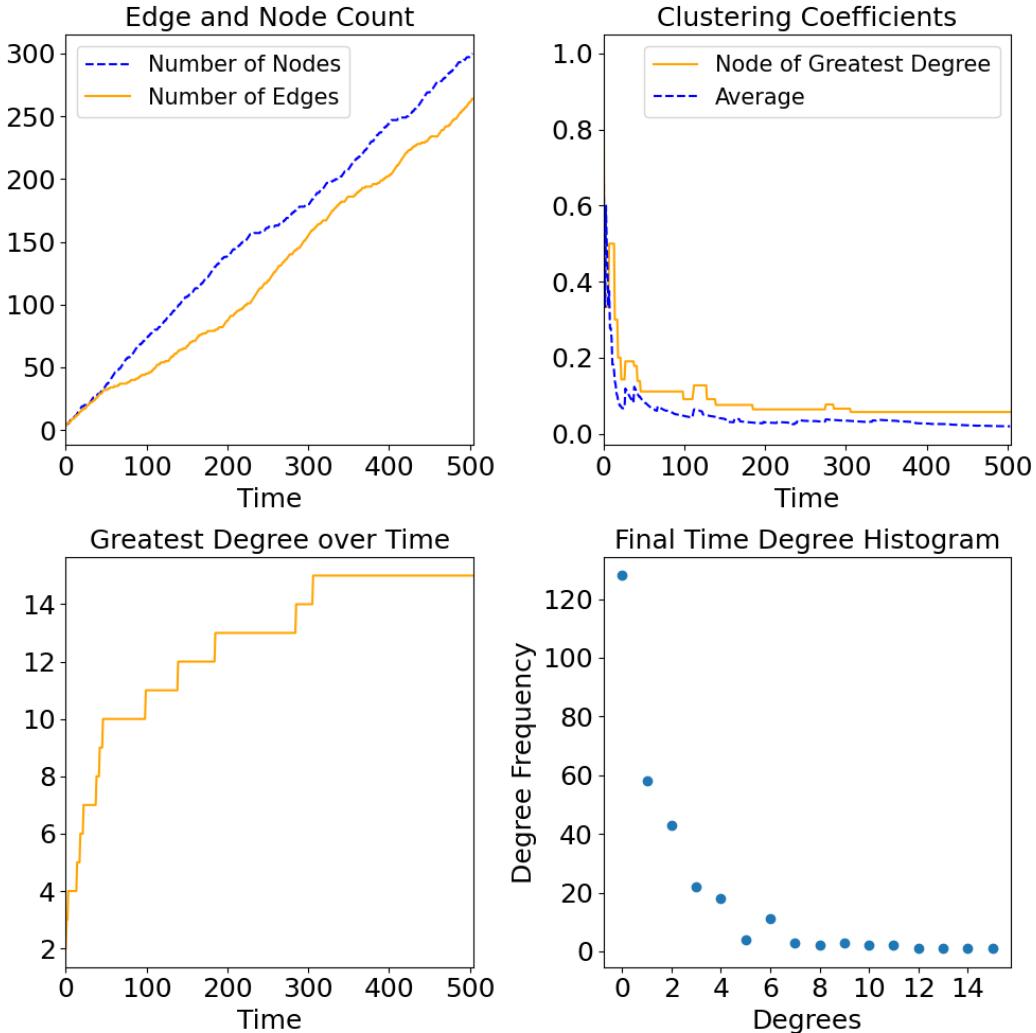


Figure 4.6. The edges and nodes travel tightly together with roughly a ratio of one-to-one. Here we see many, many nodes unattached with degree zero. For those that are attached we do see a power law describing degree distribution, but one that is not quite as strong as those found in other simulations.

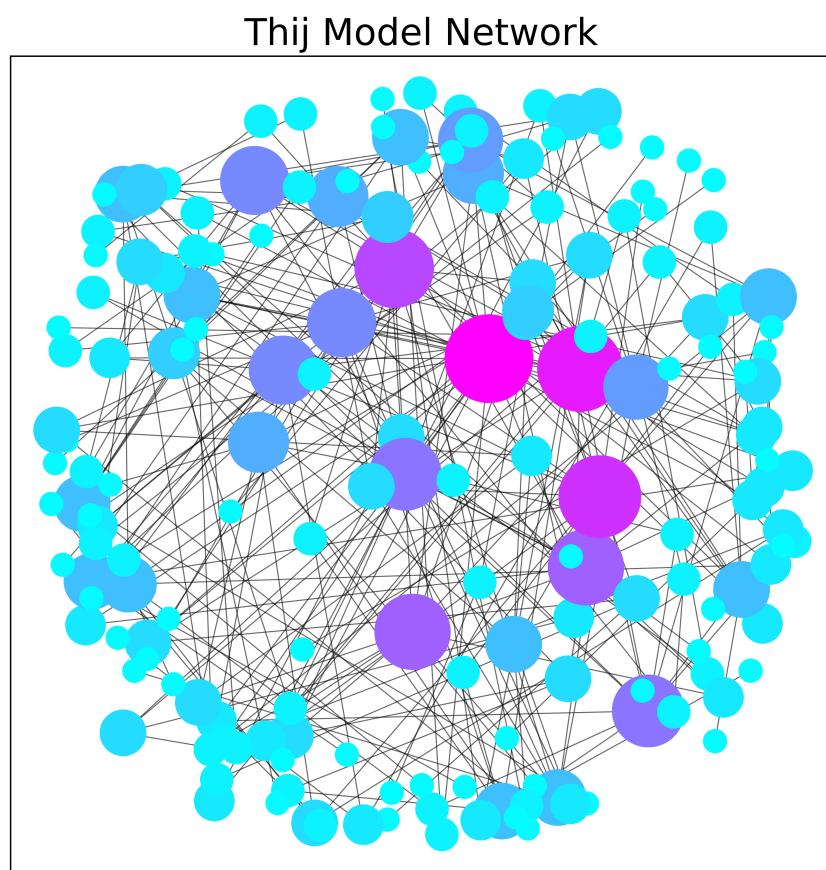


Figure 4.7. The network for $\lambda = 0.8$, $p = 0.2$ at the final time-step.

4.3 $\lambda = 0.2, p = 0.8$

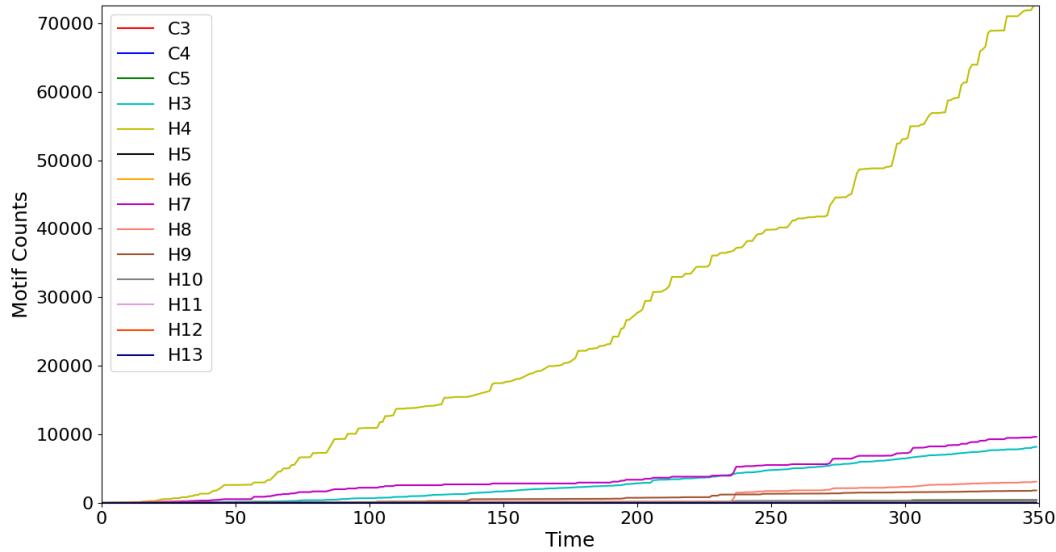


Figure 4.8. Small λ decreases the likelihood of new message nodes appearing, but high p means a greater likelihood of $T2$ events which we speculate lead to a large count of $H4$'s. C_3 's exist around the center where all these $H4$'s overlap. Although C_3 's are not prevalent here, a combinatorial effect generates many $H7$'s and $H4$'s.

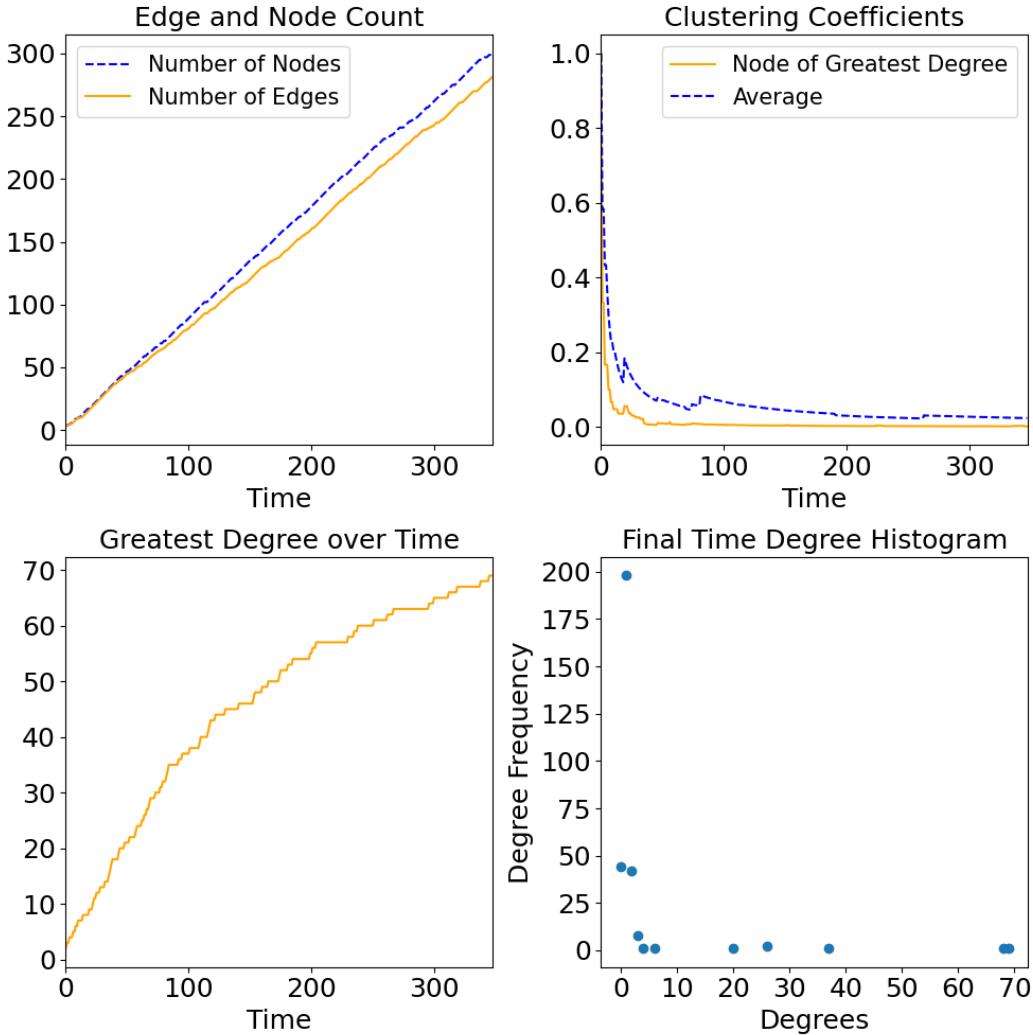


Figure 4.9. Here, we see two nodes with degrees greater than seventy, but an abundance of nodes with only one or two connections. This suggests a couple (or several) large star graphs focused around two heavily connected nodes. This would produce the abundance of H_8 's in the motif counts.

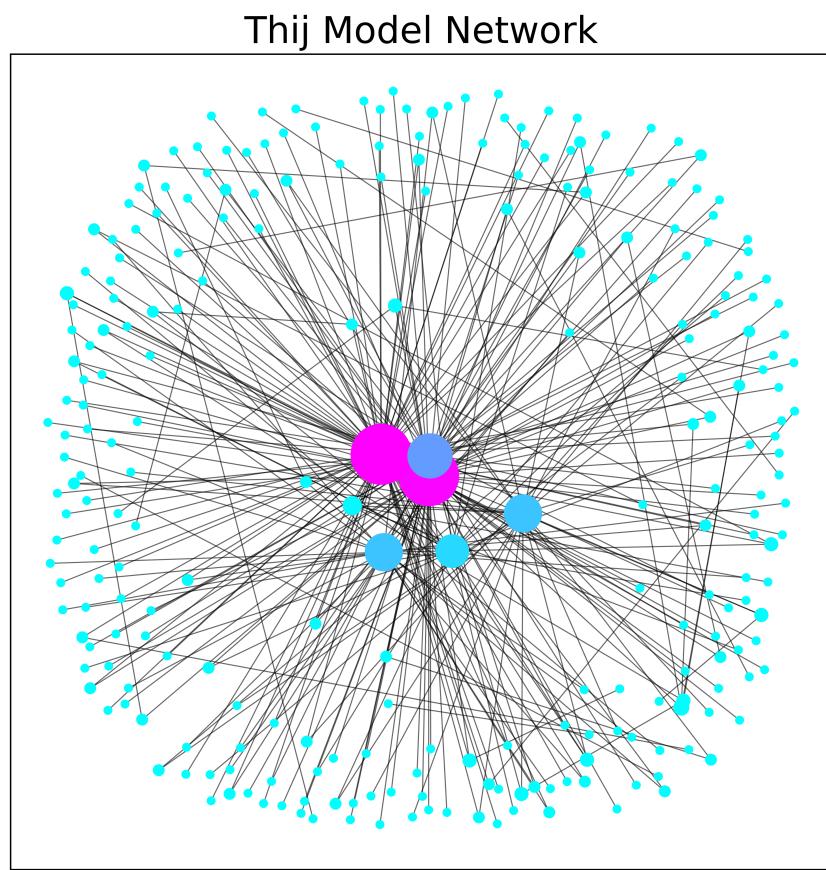


Figure 4.10. The network for $\lambda = 0.2$, $p = 0.8$ at the final time-step.

4.4 $\lambda = 0.8, p = 0.8$

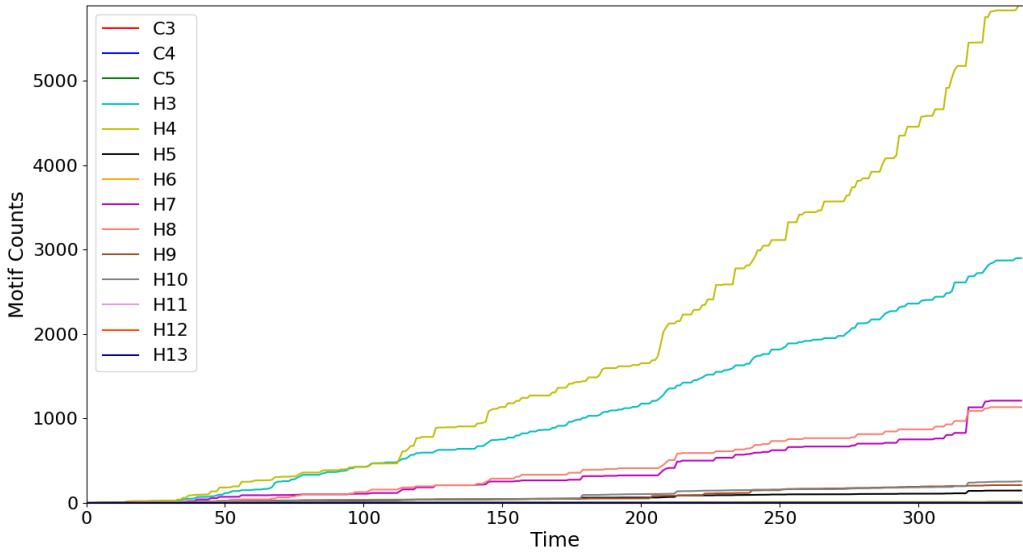


Figure 4.11. Here, like the simulation directly in figure 4.9, we see a prominence of $H4$'s. The scales however are separated by several orders of magnitude. Particular to this simulation we have a relatively high count of $H3$'s. We can explain the difference in magnitude due to many $T1$ events introducing many nodes, but the occurrence of $T2$ events is still sufficient to make $H4$ the motif of highest count.

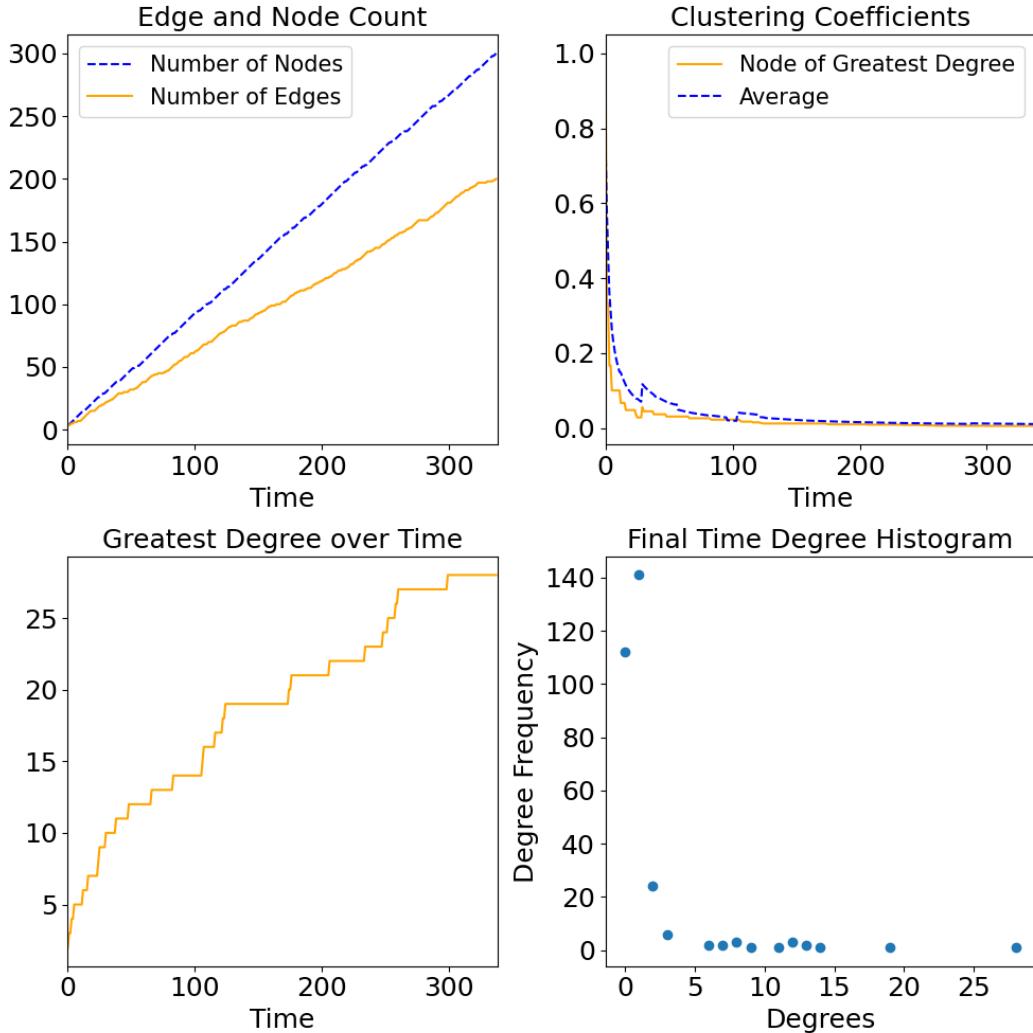


Figure 4.12. This Twitter simulation produces many more nodes than edges, because of the frequency of $T1$ events. The vast majority of nodes only have degrees of one or two, while we see a single node with 25 connections. This might suggest many small clusters of nodes with a single larger cluster around a single message node.

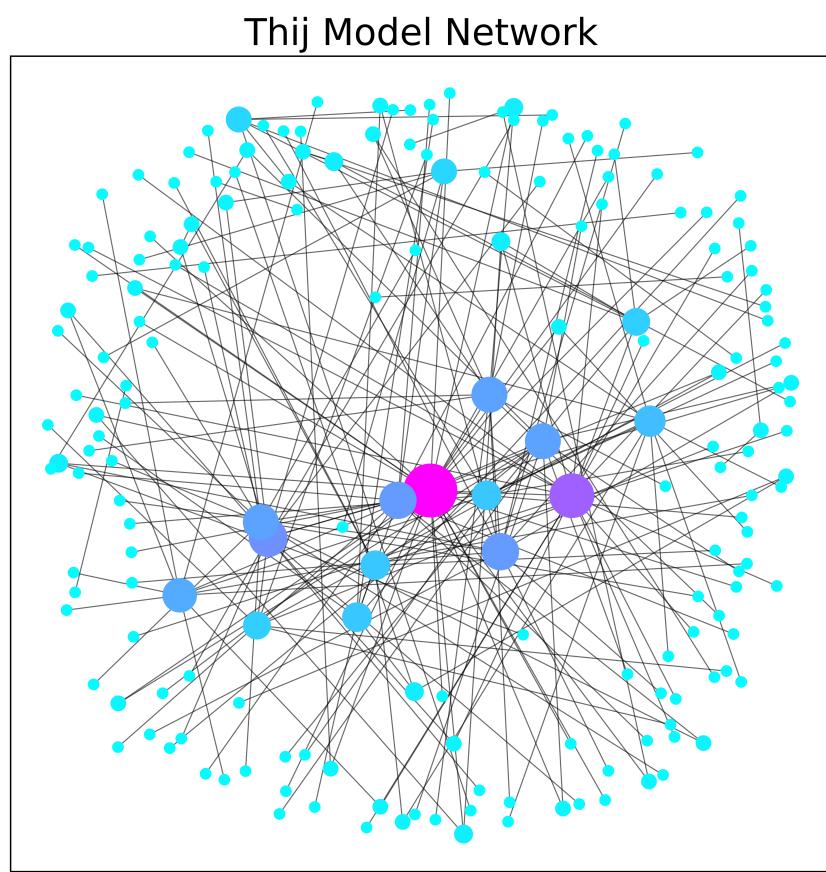


Figure 4.13. The network for $\lambda = 0.8$, $p = 0.8$ at the final time-step.

The dynamics and end state of the network vary across parameter choices. Given greater p values and smaller λ values, the degree distribution will obey stronger power laws. The ratio of edges to vertices for the Twitter model varies much more than the Barabási-Albert model which is to be expected given the probabilistic nature of how many edges may be added in a given turn (0 or 1) compared to the given m of the Barabási–Albert model. This affects not only the strucutre of the network, but the time scale over which the network grows.

Each graph spans over *very* different time frames. Each graph was allowed to grow to a maximum of three-hundred nodes, same as the sampled Barabási–Albert run, before ending the simulation. Ending the simulation with a size threshold makes a comparison of the topology of the networks more feasible. For $\lambda = 0.2$ and $p = 0.8$, the model was able to reach three-hundred nodes very quickly, in approximately three-hundred time-steps, whereas for $\lambda = 0.2$, $p = 0.2$ it took nearly a thousand time-steps. This is consequent of how λ controls how often a new node enters without a new edge and p controls the rate at which new nodes are added without attached edges or if only new edges are introduced.

CHAPTER 5

Barabási–Albert Model Motif and Thij $T2$

Event Motif Dynamics

To gain insight into graph dynamics, we analyze how the composition of motifs change upon the addition of a node to a given motif, and attaching that node to an existing node. The tables below count the isomorphisms of a motif in the present graph, but not automorphisms. For example, in the figure 5.2, the $H4$ motif has six automorphisms, but we only count the identity automorphism as an appearance of $H4$. After adding a node to the root of the star, we count four appearances of $H4$ as the motif $H4$ is isomorphic to four induced subgraphs in the newly generated motif.

5.1 H3

We begin by considering the $H3$ motif. The $H3$ motif is simply a four-path. A preferential attachment mechanism on this motif will most likely connect a new node to one of the center nodes with probability $p = 0.67$, and to an outer node happens with probability of $p = 0.33$. In the Thij model, this change is based upon which node has the superstar quality.

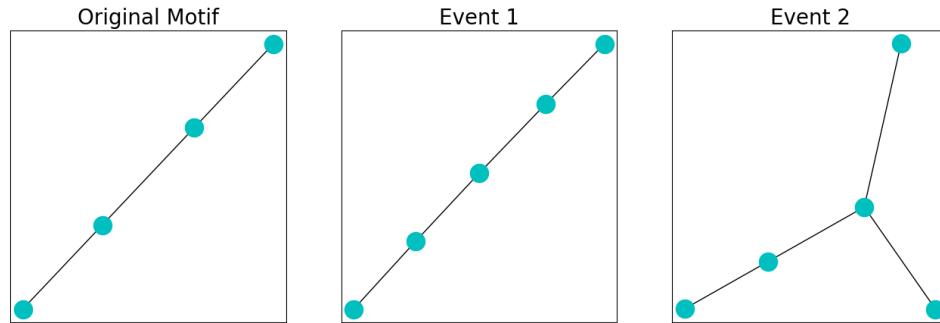


Figure 5.1. The possible graphs generated by adding a node to the $H3$ graph and connecting it to an existing node.

Motif Count	Original Motif	Event 1	Event 2
H3	1	2	2
H4	0	0	1

Table 5.1. The rows denote counts of isomorphisms that can be found in either motif. The H_3 , a four walk, can be found twice in the modified H_3 in event one by starting the walk at either end. In the graph produced again we can only find two H_3 's.

5.2 H4

The H_4 motif is one of the motifs that are of primary interest given that for any star S_k with $k \geq 3$ we will find $\binom{k}{3}$ appearances of the motif. For large clusters of H_4 motifs the H_4 motif count will grow very rapidly in time. Adding a single node and connecting it produces k new H_4 appearances in S_k . The probability for an event one on the single motif is 0.5, and for the event two 0.5. However, a significant difference in the occurrence of event ones over event twos will encourage more event ones in the future.

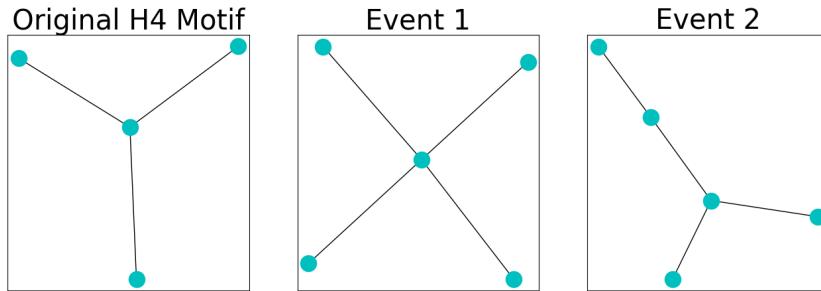


Figure 5.2. The possible graphs generated by adding a node to the H_4 graph and connecting it to an existing node.

Motif Count	Original Motif	Event 1	Event 2
H3	0	0	2
H4	1	4	1
H5	0	0	0

Table 5.2. Motif Counts of the H_4 motif and the possible motifs given a T_2 event.

5.3 H5

H_5 's are of interest due to the relationship they carry to the H_4 motif and the H_7 and H_8 motifs. The C_3 isomorphism in the motif means we find two appearances H_5 in the H_7 and H_8 motifs. We see the H_4 is isomorphic to an induced subgraph of the H_5 . The probabilities, given by a preferential attachment mechanism, of the events in figure 5.3 event one, $p = 0.375$, for event two, $p = 0.125$, and event three, $p = 0.5$. The last event only has the highest probability due to symmetry.

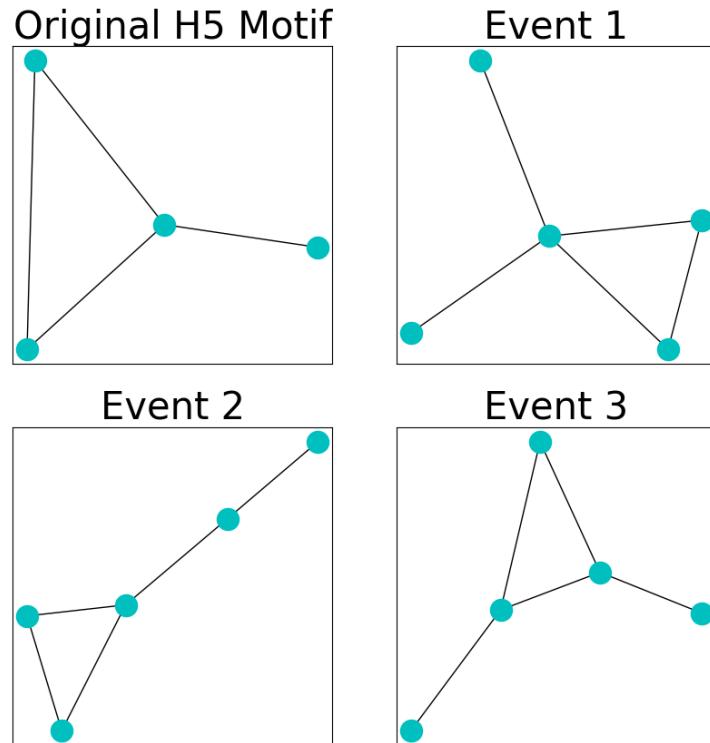


Figure 5.3. The possible graphs generated by adding a node to the H_5 graph and connecting it to an existing node

Motif Count	Original Motif	Event 1	Event 2	Event 3
H3	2	4	4	5
H4	1	4	1	2
H5	1	2	1	2
H6	0	0	0	0
H7	0	1	0	0
H8	0	0	0	1
H9	0	0	1	0

Table 5.3. Motif counts of the possible T_2 events on the H_5 motif.

5.4 H6

The H_6 motif is formed starting with an H_4 and adding two edges between the three outer nodes. It is almost a complete four-node graph. The H_6 motif count is not relatively high when compared to other motifs in the Thij model because it would require exact T_3 events to generate them. Moreover, this T_3 event has to occur between what are likely non-root nodes. However, in the preferential attachment model for $m > 2$ it is more likely to see many H_6 appearances because of the manner in which the preferential attachment model adds multiple edges from a single new node.

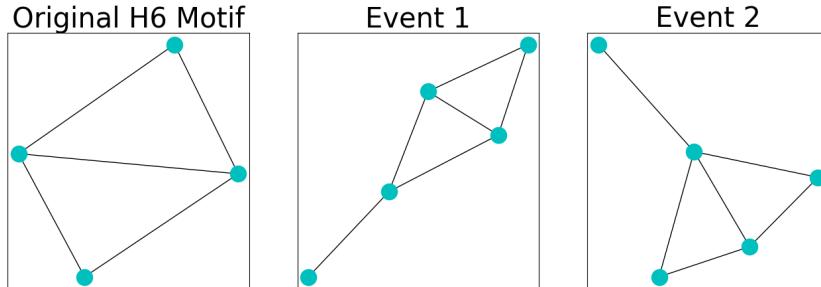


Figure 5.4. The possible graphs generated by adding a node to the H_6 graph and connecting it to an existing node

Motif Count	Original Motif	Event 1	Event 2
H3	6	10	10
H4	2	3	5
H5	4	5	6
H6	1	1	1
H7	0	0	2
H8	0	2	2
H9	0	1	1
H10	0	2	0

Table 5.4. Motif counts for variations of the $T2$ event on the $H6$ motif.

5.5 H7

$H7$ motifs feature prominently given certain parameters in the Thij model. This is another consequence of S_k induced subgraphs in the networks. Connecting any two of the outer edges of S_k will generate $\binom{k-2}{2}$ $H7$ motifs. If we generalize it and assume a network has formed with nodes attached across to all three vertices of a C_3 the $H7$ count is the sum of $\binom{d_i-2}{2}$ for $i = 1, 2, 3$. Adding a node to any one of those vertices, assuming $d_i \geq 4$, will generate $d_i - 2$ new $H7$ appearances.

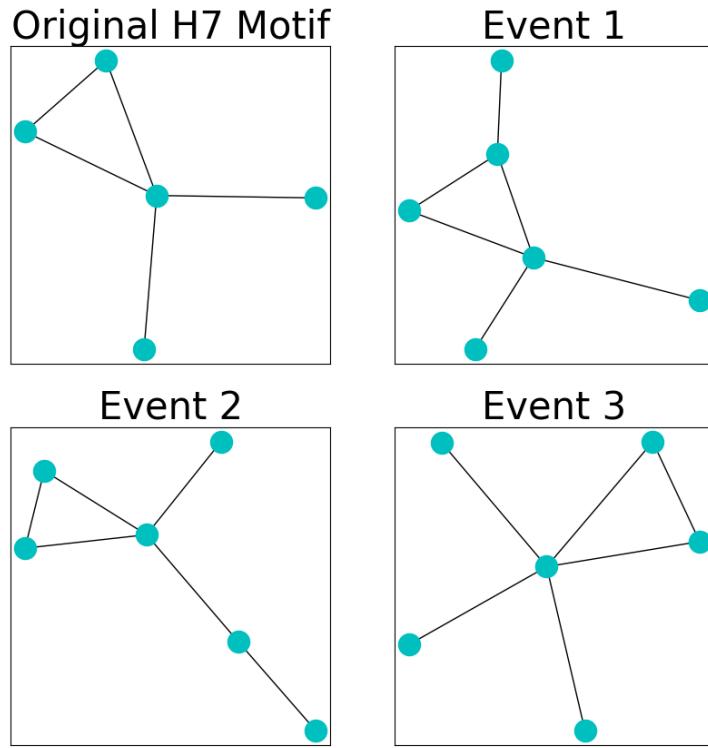


Figure 5.5. The possibly graphs generated by adding a node to the $H7$ graph and connecting it to an existing node

Motif Count	Original Motif	Event 1	Event 2	Event 3
H3	4	8	7	6
H4	4	5	4	10
H5	2	3	2	3
H6	0	0	0	0
H7	1	1	1	3
H8	0	2	0	0
H9	0	0	0	0
H10	0	0	1	0

Table 5.5. Motif counts of the $H7$ motif and the possible additions of $T2$ event nodes.

5.6 H8

The $H8$ motif is another we expect to appear fairly often. The $H8$ is two $H4$'s sharing an edge and a node. We can also characterize it as C_3 with two nodes attached to distinct vertices on the C_3 . For the $H8$ it is relatively easy to characterize growth as nodes connect to the vertices of the C_3 . Given a C_3 with at-least one node attached to each vertex we have $(d_i - 2)(d_j - 2)(d_k - 2)$ $H8$'s. The number of new $H8$'s by connecting a node to vertex v_i is given by $(d_j - 2)(d_k - 2)$.

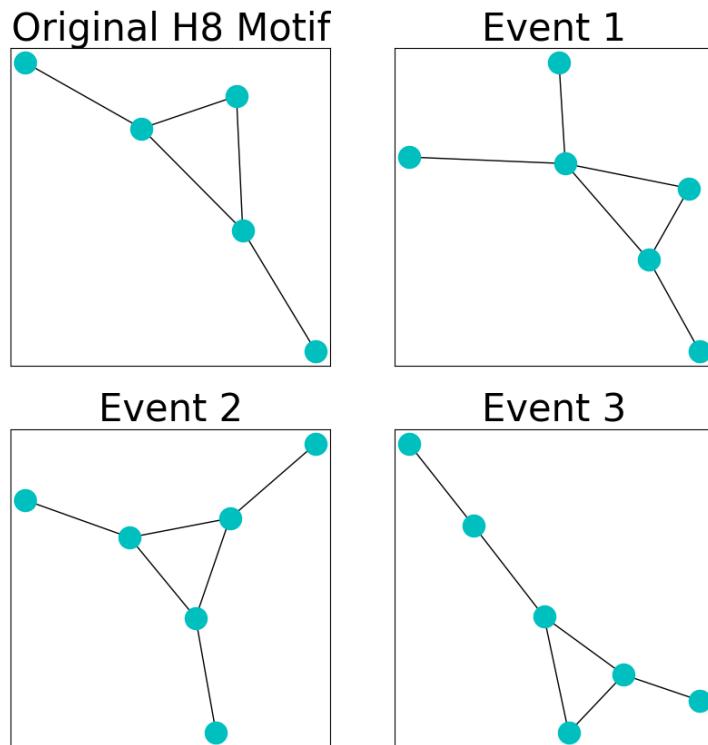


Figure 5.6. The possible graphs generated by adding a node to the $H8$ graph and connecting it to an existing node

Motif Count	Original Motif	Event 1	Event 2	Event 3
H3	5	8	9	7
H4	2	5	3	2
H5	2	3	3	2
H6	0	0	0	0
H7	0	1	0	0
H8	1	2	3	1
H9	0	0	0	0
H10	0	0	0	1

Table 5.6. Motif counts graphs formed by possible T_2 events on the H_8 motif.

5.7 H9

The H_9 motifs do not form around stars in the same way we might expect H_7 's and H_8 's to do so. The H_9 motif could develop in a way similar to the H_5 . This is because the H_9 has an induced subgraph isomorphic to the C_4 . The H_9 is produced by attaching a node to any one of those vertices in the induced subgraph. Given a C_4 and attachment of new nodes to any of its four vertices, the count of H_9 's is simply the sum of the degrees of each vertex minus 2. The growth is additive, not combinatorial in the manner of H_7 's or H_8 's.

Motif Count	Original Motif	Event 1	Event 2	Event 3	Event 4
H3	6	8	8	9	8
H4	1	1	4	2	2
H5	0	0	0	0	0
H6	0	0	0	0	0
H7	0	0	0	0	0
H8	0	0	0	0	0
H9	1	1	2	2	2

Table 5.7. Motif counts graphs formed by possible T_2 events on the H_9 motif.

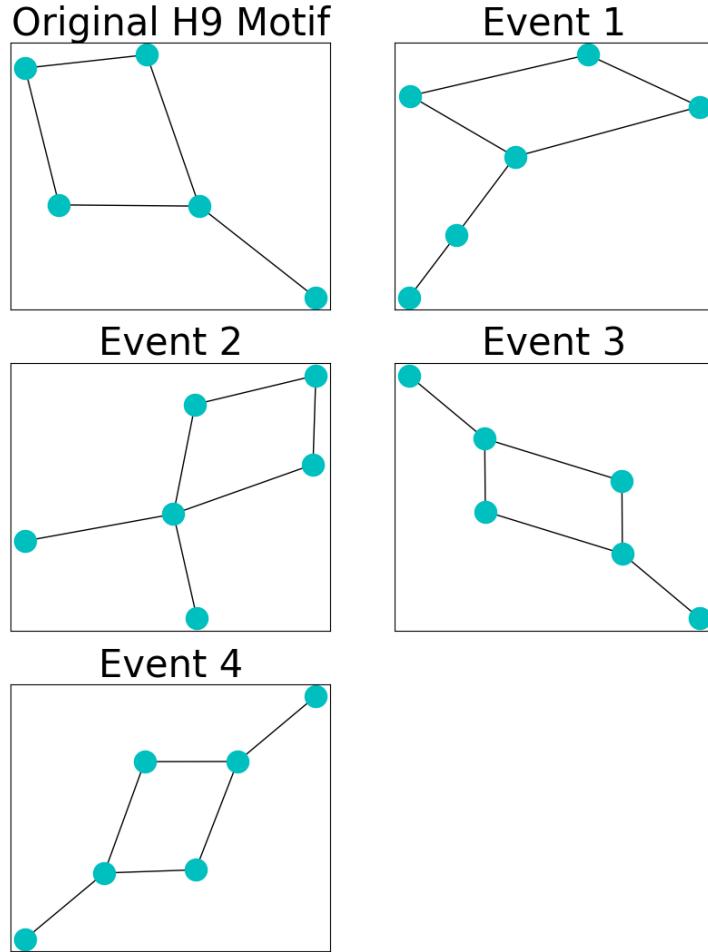


Figure 5.7. The possible graphs generated by adding a node to the $H9$ graph and connecting it to an existing node

5.8 H10

The $H10$ is the $H9$ with an extra vertex and an edge between that vertex to the single vertex of degree one in the $H9$. Many $H10$'s could be generated from a single $H9$ if vertices attach to the single vertex of degree one in the $H9$ motif. A star graph would form with that vertex at the center. The growth for the $H10$ in the preferential mechanism with $k = 1$ is additive. For $k > 1$, it is possible clusters of triangles to see the $H10$ motif count to grow faster than what one would see, adding a single node and edge at each time step.

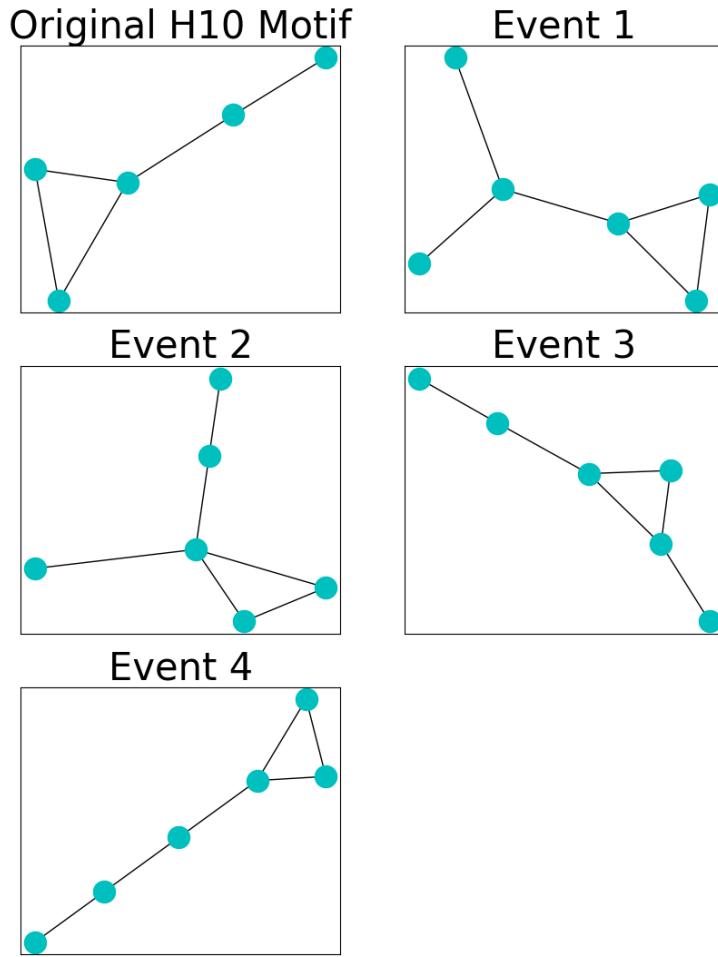


Figure 5.8. The possible graphs generated by adding a node to the H_{10} graph and attaching it to an existing node.

Motif Count	Original Motif	Event 1	Event 2	Event 3	Event 4
H3	4	6	7	7	5
H4	1	2	4	2	1
H5	1	1	2	2	1
H6	0	0	0	0	5
H7	0	0	1	0	0
H8	0	0	0	1	0
H9	0	0	0	0	0
H10	1	2	1	1	1

Table 5.8. Motifs counts of the possible T_2 event on the H_{10} motif.

5.9 H11

The H_{11} motif, shaped like a bow-tie, is formed by two three-walks that share a common vertex. This motif does need $k \geq 2$ in the Barabási–Albert Model or a T_3 event to form an edge between two existing nodes. The H_{11} motif does not appear commonly without those necessary criteria.

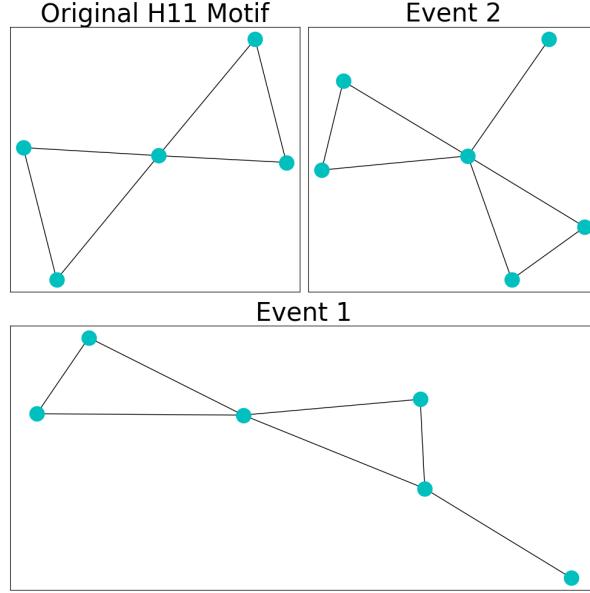


Figure 5.9. The possibly graphs generated by adding a node to the H_{11} graph and connecting it to an existing node.

Motif Count	Original Motif	Event 1	Event 2
H3	8	12	12
H4	4	5	10
H5	0	5	6
H6	0	0	0
H7	2	2	6
H8	0	2	0
H9	0	0	0
H10	4	5	4
H11	1	1	1

Table 5.9. Motif counts of the graphs generated by possible T^2 events on the H_{11} motif.

5.10 H12

H_{12} , shaped like a house, contains five nodes, six edges, with an induced subgraph isomorphic to C_4 and a single vertex attached to two vertices they connected. The H_{12} , like other motifs, contains an induced subgraph isomorphic to C_4 is not a priori expected to have a relatively large count given the preferential attachment mechanism.

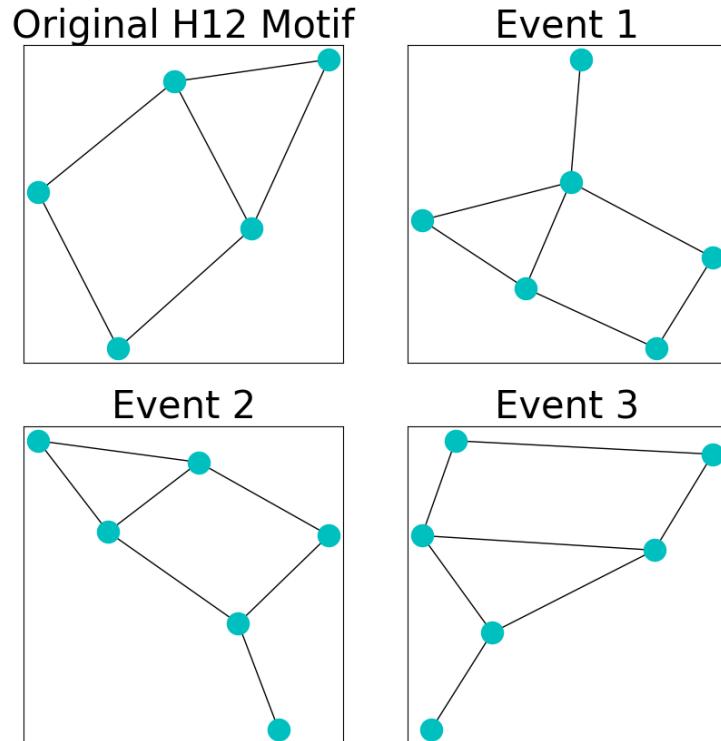


Figure 5.10. The possibly graphs generated by adding a node to the H_{12} graph and connecting it to an existing node.

Motif Count	Original Motif	Event 1	Event 2	Event 3
H3	10	14	13	14
H4	2	5	3	3
H5	2	3	2	3
H6	0	0	0	0
H7	0	1	0	0
H8	1	2	1	3
H9	2	3	3	2
H10	2	2	3	2
H11	0	0	0	0
H12	1	1	1	1

Table 5.10. Variations of the T_2 event on the H_{12} motif

5.11 H13

The H_{13} could plausibly form in the $k \geq 2$ case for the Barabási–Albert model, but it would require a new node being consistently attached to the same two nodes repeatedly. If this process of generating new H_{13} 's, the increase in the motif counts is additive. Thus for the Barabási–Albert model of $0 < k < 3$, the presence of an H_{13} is sensitive to the initial graph and its early development when there might still be a 'more uniform' probability of attachment.

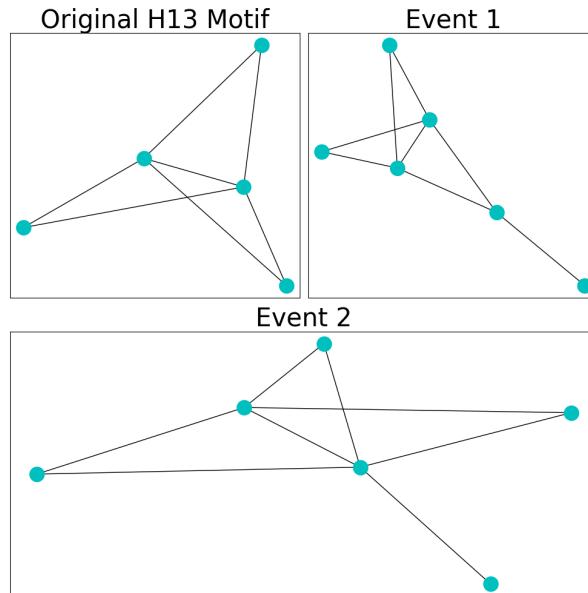


Figure 5.11. The H_{13} graph and possible node attachments up to symmetry.

Motif Count	Original Motif	Event 1	Event 2
H3	18	24	24
H4	8	14	9
H5	12	15	13
H6	3	3	3
H7	6	12	6
H8	6	12	10
H9	6	9	8
H10	0	0	4
H11	0	0	0
H12	0	0	0
H13	1	1	1

Table 5.11. Motif counts of the possible $T2$ events on the $H13$ motif

5.12 Summary of Preferential Attachment and $T2$ Event Motif Evolution

Motif development in the Barabási–Albert model is dependent on how the model is initialized and the choice of k . If we only add $k = 1$ edges for every new node this limits the types of motifs that can appear. There will be no new $H6$'s, $H11$'s, $H12$'s, or $H13$'s generated. There is no possible way for them to form as there is no node of degree one in those motifs. Only upon taking $k > 1$ could those motif counts change over time. Some graphs still are more likely for $k = 2$ than others like the $H4$'s, $H7$'s, and $H8$'s.

This analysis also applies to the $T2$ event in the Thij model. The occurrence is similar to the preferential attachment model with $k = 1$ as it is simply the addition of a node which is then connected to a single existing node. The $T2$ event, unlike the BA model, selects a message tree and then uses a superstar attachment mechanism to attach to a node with probability $q = 0.9$. Even when the network is relatively small nodes will still overwhelmingly attach to the root message node. This mechanism is much more probable to generate $H4$'s as seen above in chapter 3. If the root message node is the vertex of a C_3 then we may see $H7$ and $H8$ counts rapidly increase over time, correlating with the $H4$ count.

CHAPTER 6

Twitter Model Specific Motif Evolution

In chapter 3, we specified three possible events in the Thij model: a new root node, a new node and an attached edge, or a new edge between existing nodes (T_1 , T_2 , T_3 respectively). The only event specific to the Thij model are T_1 and T_3 events. T_1 events add a new message node, but they don't immediately affect any change in the motif counts. They could potentially with the right T_3 event or a series of T_2 events. The T_3 event we must consider, because a T_3 event can change the composition of the network in ways T_2 events cannot. Given that a T_3 event is occurring it follows first, the preferential-attachment mechanism for tree selection, then follows superstar probability for source node selection, and then finally uniform probability for target node selection. This makes it difficult to discuss the associated probabilities of a potential T_3 event. We may suspect a root message node within a graph based on its degree, but may not know without prior knowledge of the data. We can examine how each T_3 event will affect a given motif graph provided the T_3 event occurs between nodes in the motif.

6.1 H3

The H_3 motif only has two possible events. We see either an H_5 form by connecting an outer vertex to the opposite inner or the outer two vertices are connected forming a four-cycle. Future development of this new H_5 in the context of T_3 events is discussed in section 6.3.

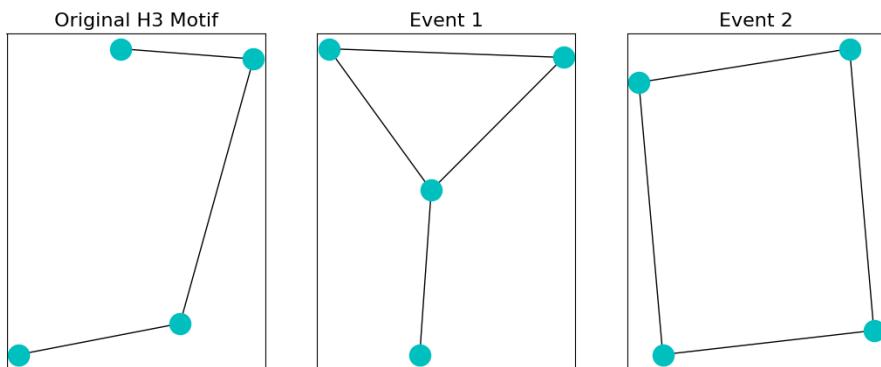


Figure 6.1. The possible graphs generated by adding an edge to the H_3 graph.

Motif Count	Original Motif	Event 1	Event 2
H3	1	2	4
H4	0	1	0
H5	0	1	0

Table 6.1. Motif counts of the possible T_3 events on the H_3 motif

6.2 H4

Due to the three symmetry of the H_4 , adding an edge between any two unconnected nodes creates a single H_5 . Therefore given some balance between T_2 and T_3 events we could see correlation between these two motifs.

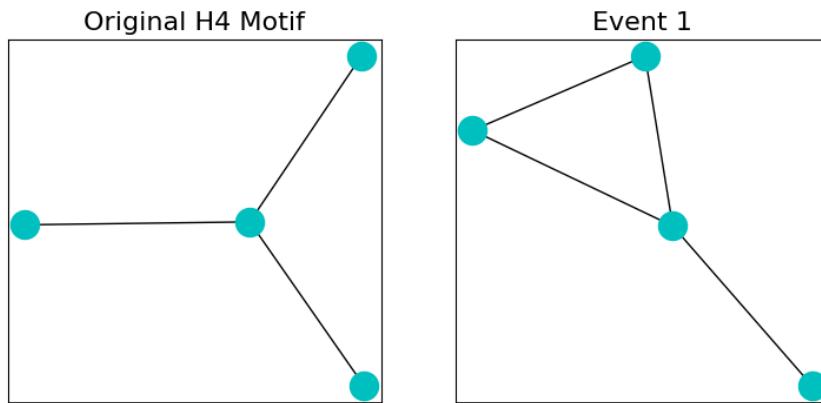


Figure 6.2. The possible graphs generated by adding an edge to the H_4 graph. By connecting the two outer edges we find a C_4 and by connecting an outer vertex to an inner vertex we generate a H_5 .

Motif Count	Original Motif	Event 1
H3	0	2
H4	1	1
H5	0	1

Table 6.2. Motif counts of the T_3 event on the H_4 motif

6.3 H5

The H_5 motif is symmetric. Connecting any two unconnected vertices of the H_5 produces an H_6 . A H_4 with a $T3$ event creates an H_5 graph. A $T3$ event occurs again on the same graph and a H_6 graph is produced.

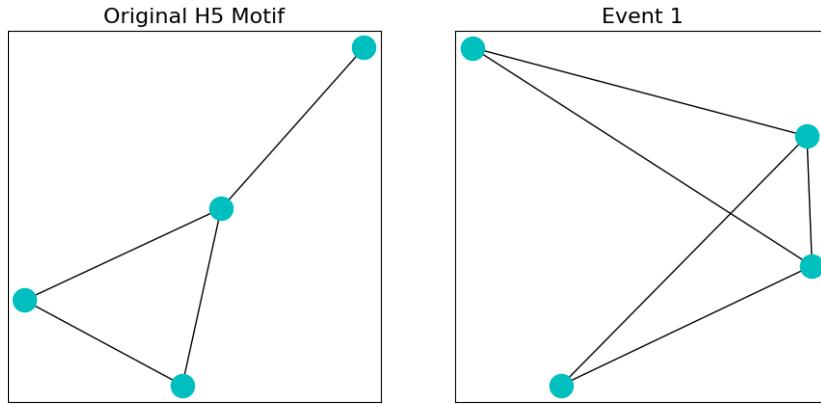


Figure 6.3. The possible graphs generated by adding an edge to the H_5 graph.

Motif Count	Original Motif	Event 1
H3	2	6
H4	1	2
H5	1	4
H6	0	1

Table 6.3. Motif counts of the graphs produced by a $T3$ event on the H_5 motif.

6.4 H6

Adding an edge to H_6 generates a complete graph of four nodes. This event is unlikely given a large graph, but not impossible. If a vertex of the H_6 is a message node then an edge could be added within the H_6 graph. However, an edge added between the H_6 appearance and the appearance of another motif is more likely.

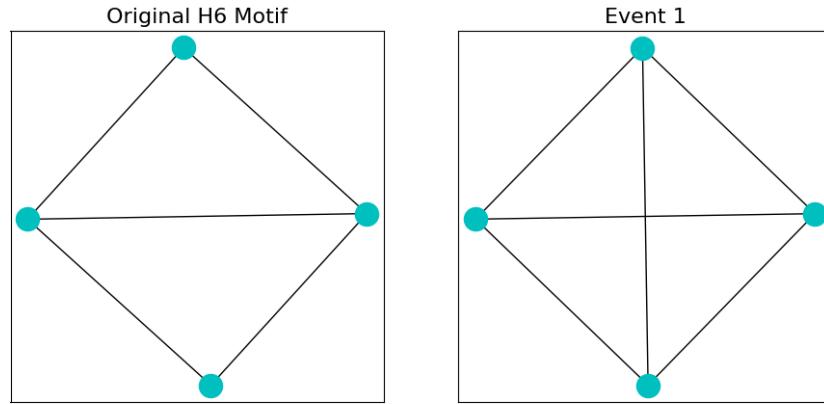


Figure 6.4. The possible graphs generated by adding an edge to the H_6 graph.

Motif Count	Original Motif	Event 1
H3	6	12
H4	2	4
H5	4	12
H6	1	6

Table 6.4. Motif counts of graph produced by a T_3 event on the H_6 motif.

6.5 H7

Adding edges, the H_7 motif gives only two non-isomorphic graphs as a consequence of the motif's symmetry. Given an event two in figure 6.5 the graph becomes an H_{11} . Both events do show that one could see a high H_7 count given enough clustering around a message node.

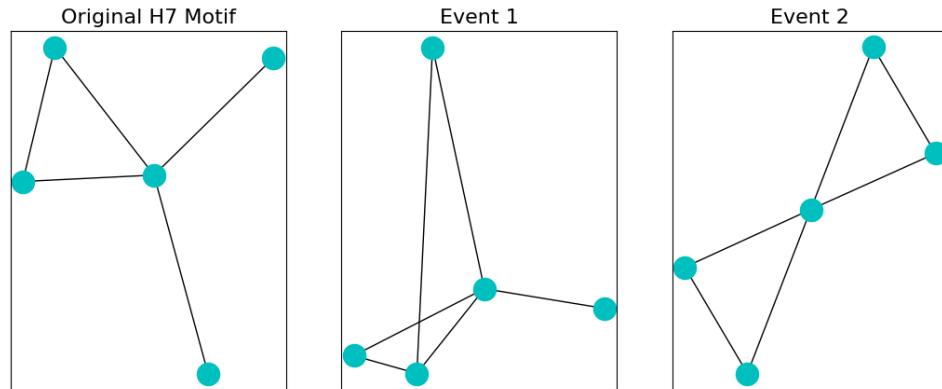


Figure 6.5. The possible graphs generated by adding an edge to the H_7 graph.

Motif Count	Original Motif	Event 1	Event 2
H3	4	10	8
H4	4	5	4
H5	2	6	4
H6	0	1	0
H7	1	2	2
H8	0	2	0
H9	0	1	0
H10	0	0	4
H11	0	0	1

Table 6.5. Motif counts of the two possible graphs produced by the T_3 event on the H_7 motif

6.6 H8

The H_8 motif here generates three different motifs depending upon the nodes which become connected. The H_8 motif is similar to the H_7 motif as clusters of triangles can contain many H_8 appearances.

Motif Count	Original Motif	Event 1	Event 2	Event 3
H3	5	10	10	10
H4	2	5	3	2
H5	2	6	5	2
H6	0	1	1	0
H7	0	2	0	0
H8	1	2	2	1
H9	0	1	1	2
H10	0	0	2	2
H11	0	0	1	0
H12	0	0	0	0

Table 6.6. Motif counts of the possible T_3 event on the H_8 motif

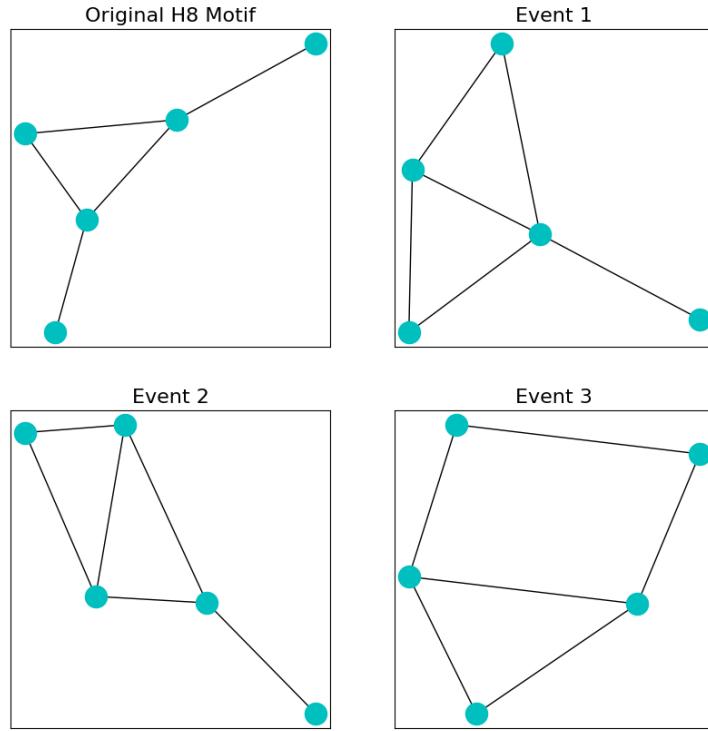


Figure 6.6. The possible graphs generated by adding an edge to the H_8 graph.

6.7 H9

An H_9 motif with an edge added anywhere does not produce a combinatorial jump in counts as other motifs might. The H_9 had an induced subgraph, isomorphic to the C_4 which means the graph requires more events or the right initialization to generate itself. Given a high $T3$ probability there relatively high H_9 counts as seen in figures 9.11 and 9.21.

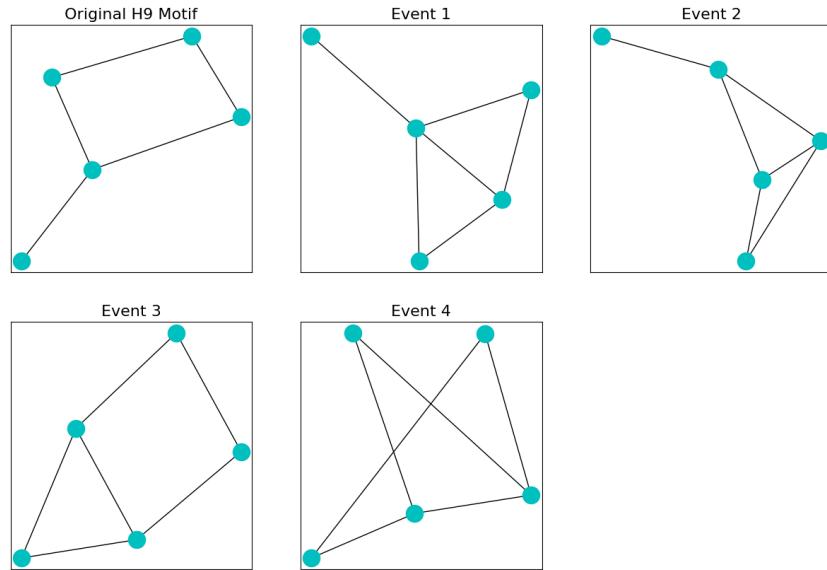


Figure 6.7. The possible graphs generated by adding an edge to the $H9$ graph.

Motif Count	Original Motif	Event 1	Event 2	Event 3	Event 4
H3	6	10	10	10	10
H4	1	5	3	2	2
H5	0	6	5	2	2
H6	0	1	1	0	0
H7	1	2	1	0	0
H8	0	2	2	1	1
H9	1	1	2	2	2
H10	0	0	2	2	2
H11	0	0	0	0	0
H12	0	0	0	1	1

Table 6.7. Motif counts of the possible $T3$ event on the $H9$ motif

6.8 H10

The H_{10} motif features in those Thij models with $p < 0.5$ line in figures 9.11 and 9.26, given that there is enough likelihood a $T3$ event occurs acting on an $H9$. For events two, three, and four below in Figure 6.8, attaching a node to the H_{10} motif produces a handful more of H_{10} 's.

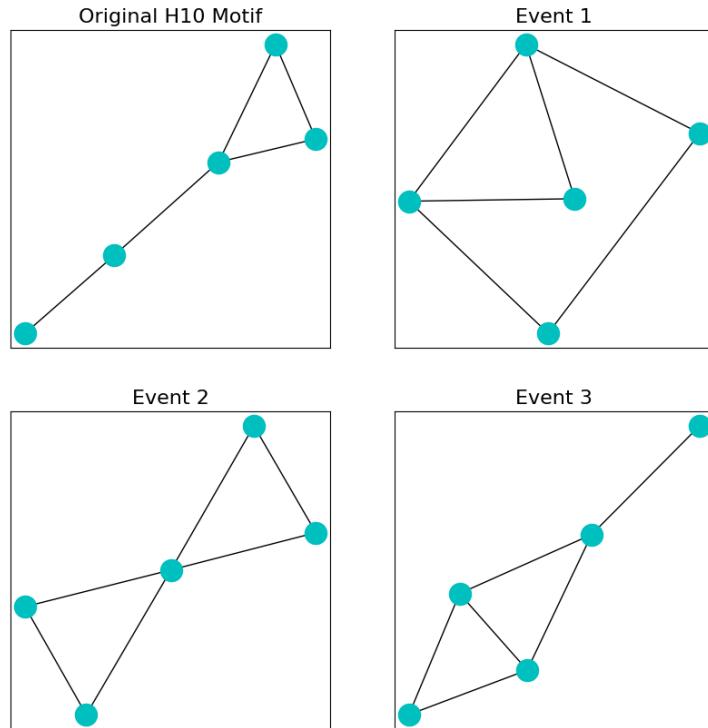


Figure 6.8. The possible graphs generated by adding an edge to the H_{10} graph.

Motif Count	Original Motif	Event 1	Event 2	Event 3
H3	4	10	8	10
H4	1	2	4	3
H5	1	2	4	5
H6	0	0	0	1
H7	0	0	2	0
H8	0	1	0	2
H9	0	2	0	1
H10	1	2	4	2
H11	0	0	1	0
H12	0	1	0	0

Table 6.8. Motif counts of the graphs given by a T_3 event on the H_{10} motif

6.9 H11

H_{11} 's do not appear prominently in any of the simulations because they are isomorphic to two C_3 's connect at a single vertex. To produce an H_{11} from another one would need to add a whole new C_3 and to connect to one of the H_{11} vertices . Connecting at its center would increase the H_{11} count by a total of the three cycles that make it up, but in our Thij model we would need exactly the right T_2 events and an exact T_3 event.

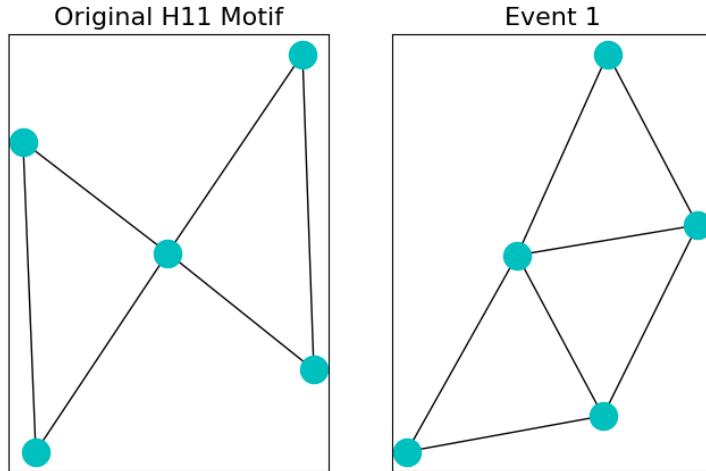


Figure 6.9. The possible graphs generated by adding an edge to the H_{11} graph.

Motif Count	Original Motif	Event 1
H3	8	17
H4	4	6
H5	4	10
H6	0	2
H7	2	3
H8	0	5
H9	0	4
H10	4	6
H11	1	1
H12	0	2
H13	0	0

Table 6.9. Motif counts of the single graph produced by a $T3$ event on the $H11$ motif

6.10 H12

As we discussed in section 5.10, the $H12$ motifs are not commonly found in the network simulations because they require a four-walk. In the Thij model, assuming the presence of an $H12$, one could generate more $H12$'s by a $T2$ event to a vertex of the four-cycle and then a $T3$ event to follow. In table 6.10 we see even a single $T3$ would suffice to produce two or even four new $H12$'s.

Motif Count	Original Motif	Event 1	Event 2
H3	10	17	18
H4	2	6	4
H5	2	10	6
H6	0	2	1
H7	0	3	0
H8	1	5	4
H9	2	4	8
H10	2	6	6
H11	0	1	0
H12	1	2	4
H13	0	0	0

Table 6.10. Motif counts of the possible $T3$ event on the $H12$ motif.

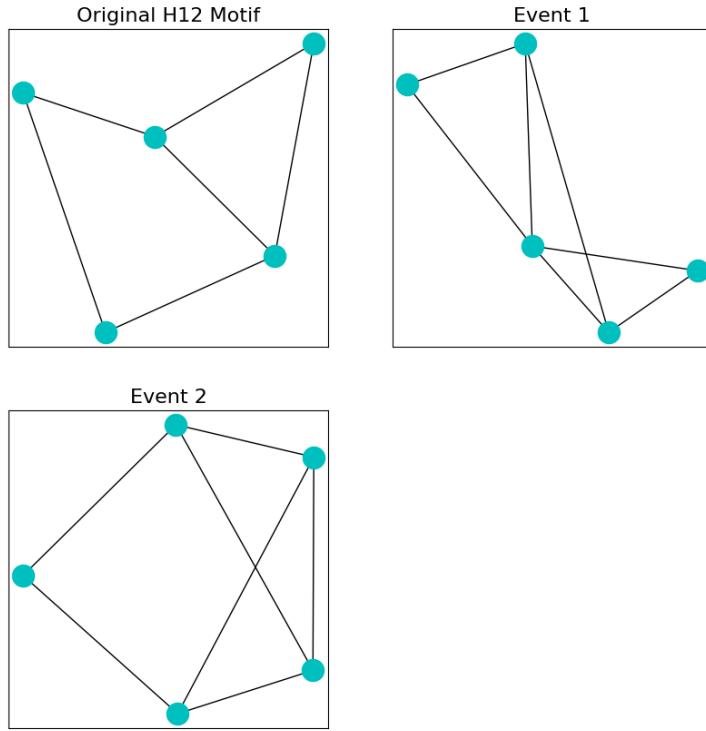


Figure 6.10. The possible graphs generated by adding an edge to the $H12$ graph.

6.11 $H13$

The $H13$ does not frequently appear in any of the simulations. $H13$'s are not readily generated from a $H13$ motif, but require a $T2$ event and a $T3$ event to connect the newly introduced $T2$ node such that it forms a C_4 at one of the nodes of degree four in the $H13$. Contrast this to an $H7$ or an $H8$ which upon a $T3$ event alone can generate new $H7$'s or $H8$'s. The $H13$ requires events to occur, which are not likely given the attachment mechanism.

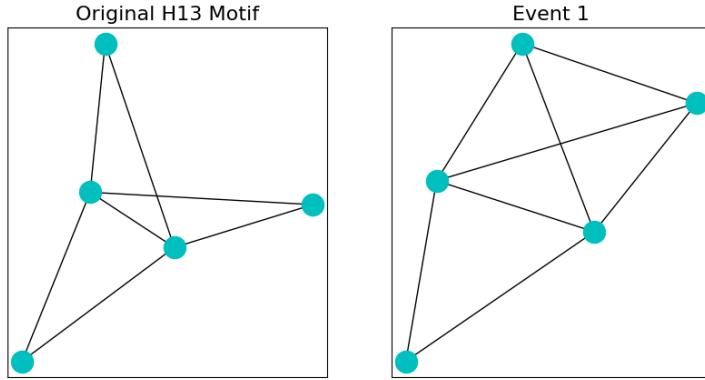


Figure 6.11. The possible graph generated by a T_3 event on the $H13$.

Motif Count	Original Motif	Event 1
H3	18	28
H4	8	10
H5	12	22
H6	3	8
H7	6	8
H8	6	14
H9	6	12
H10	0	12
H11	0	2
H12	0	6
H13	1	1

Table 6.11. Motif counts of the graph given by a T_3 event on the $H13$ motif.

6.12 In summary of the T_3 events

The T_3 event is a much more complex animal than the T_2 . For any of the motifs $H3-H13$, we see how T_3 events change the motif counts, but T_3 's can add edges *between* different motifs as well as on motifs. If two motifs are disjoint a T_3 bridges them and functions as one or more T_2 events occurring in a single time-step. This is still a simple case, but for the eleven motifs considered there are fifty-five different possibilities.

The T_3 ultimately opens up several possibilities for the model. C_3 's and C_4 's are more likely to form around root nodes. We may ultimately see more $H6$'s through $H13$'s. For $p = 0.2$, the Thij model exhibits significantly more clustering throughout time. Shedding light on the impact of the p parameter is for the tools of statistical analysis.

CHAPTER 7

Motif Correlation

Given the analysis in chapters 4 and 5, a $T2$ or $T3$ event applied to a certain motif will generate a given amount of new motifs. We expect some motifs like $H4$, $H7$, and $H8$ to correlate together. We compute correlation and covariance matrices for the time-series generated by motif counts. There is an order of magnitude difference in covariances between motifs although all are strongly correlated. All covariance matrices below are described on a log scale. We begin with a simulation of the Barabási–Albert model resulting in the heat maps in figures 7.1 and 7.2:

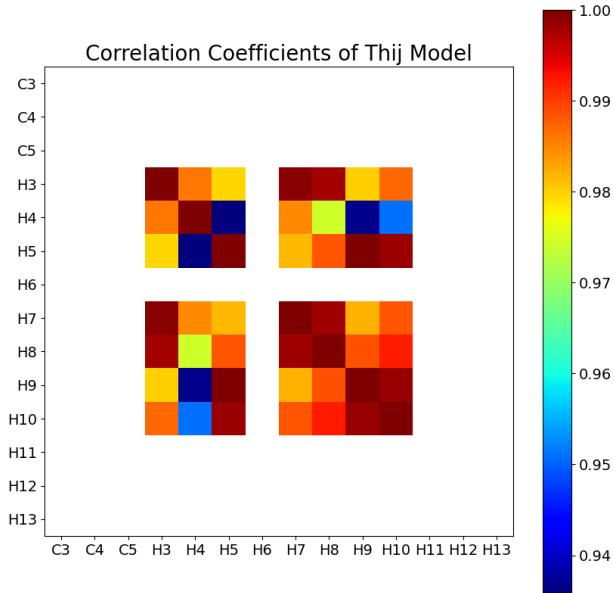


Figure 7.1. Barabási–Albert model with eight initial nodes and $m = 1$. The model is capable of only producing certain motifs due to the limitations of attaching a single edge and a single node at every time-step. We also see that $H7$'s and $H8$'s correlate together. However, appearances of those motifs depend upon the initialization of the graph itself.

The Barabási–Albert model, for $m = 1$, is only capable of generating certain new motifs after the initialization as it can only add one edge at a time. Some motifs would

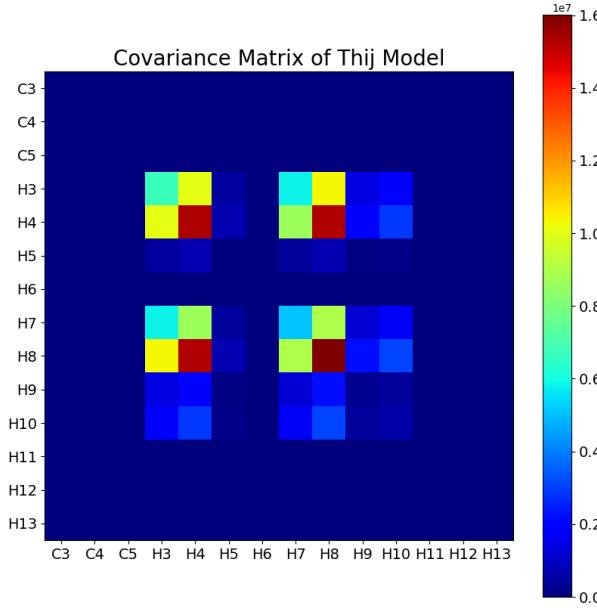


Figure 7.2. Barabási–Albert model with eight initial nodes and $m = 1$. Here we once again see that the H_7 and H_8 motifs have a much higher covariance than any other pair of motifs.

require the introduction of an event which can add edges between existing nodes. The only motifs that increase in size after the initialization of the $m = 1$ Barabási–Albert model are $H_2, H_4, H_5, H_7, H_8, H_9$, and H_{10} . There is an initialized cluster of nodes at the center of the graph and new nodes are introduced by the preferential attachment mechanism. However we see in figures 7.3 and 7.4 that we can generate all motifs provided $m > 2$.

In figures 7.3 and 7.4, we have a Barabási–Albert simulation that more so resembles our Thij model for certain parameters. For the Barabási–Albert model with $m = 2$, all motifs can now be generated via an attachment event. This allows for the formation of C_3 and C_4 subgraphs that are necessary for the the motifs that also have C_3 and c_4 subgraphs. We also see that some of our a-priori speculation in chapter 5 is supported by the correlation and covariance matrices.

We recall that for relatively high p we are more likely to attach a new node and a new edge. For relatively low p we attach a new edge between existing nodes. For relatively high λ we are likely to introduce a new message node without any attachments. In chapter 4, we discussed that for high p we introduce a new node and

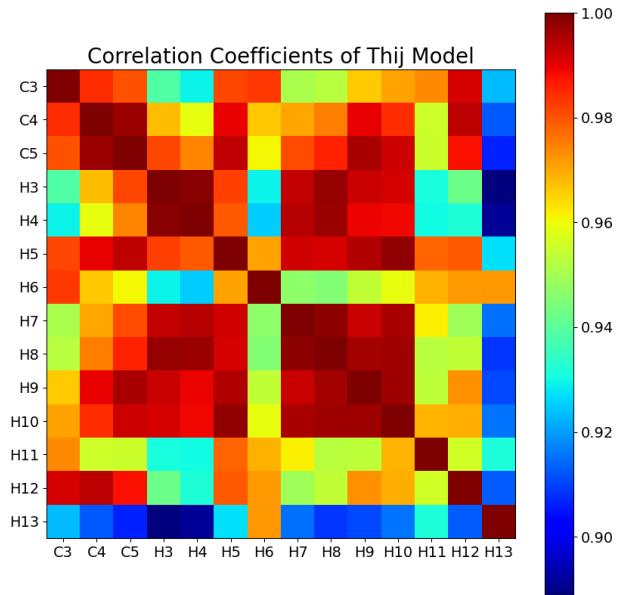


Figure 7.3. Barabási–Albert model with three initial nodes and $m = 2$. Here we see that all motifs are present. They all have fairly high correlation coefficients, but we see that the H7's and H8's are highly correlated as is H7 and H8 with H3.

attach it with a the superstar mechanism after selecting a message subgraph with the preferential attachment mechanism.

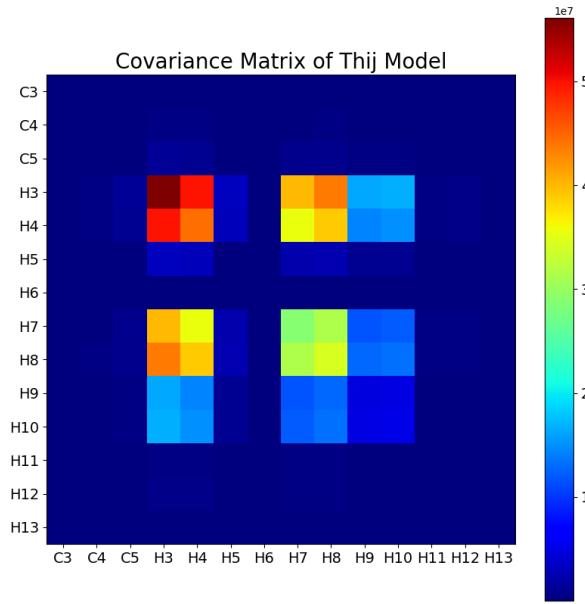


Figure 7.4. Barabási–Albert model with three initial nodes and $m = 2$. The covariance offers a different perspective than the correlation figure. We now see that their high covariance between $H7$'s and $H8$'s, and we once again see a relationship between $H8$, $H7$, $H3$, $H4$, but many other motifs have a much, much smaller covariances.

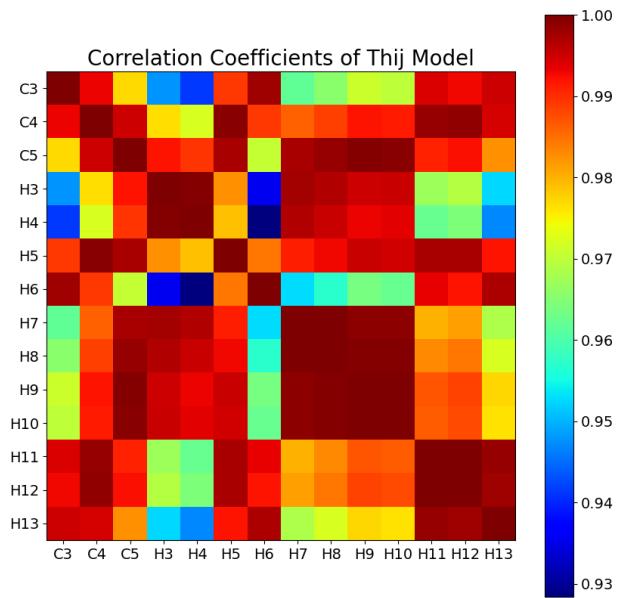


Figure 7.5. $\lambda = 0.2$ and $p = 0.2$. We recall low λ reduces the chance of only adding a new node (T1) and low p reduces the chance of adding a new node and a new edge. Thus probabilistically this simulation should be overall adding only edges between existing nodes.

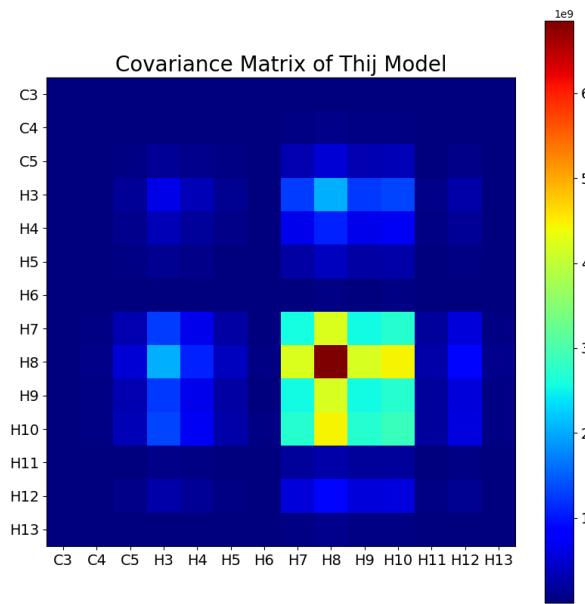


Figure 7.6. $\lambda = 0.2$ and $p = 0.2$. Here we see once again a strong covariance relationship between H_7 , H_8 , H_9 , H_{10} . In this particular graph it's insightful to see these correlate together given these parameters are less likely to produce the induced star subgraph we expect to see for higher p .

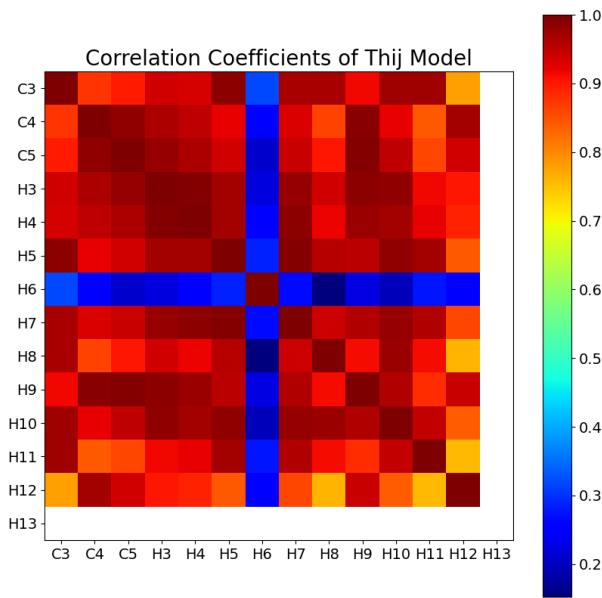


Figure 7.7. $\lambda = 0.2$ and $p = 0.8$. Here in the correlation coefficients we see that there is a strong correlation between H_7 , H_8 , but otherwise a much larger spread over the motifs. We also see that no H_6 or H_{13} motifs appeared throughout this simulation.

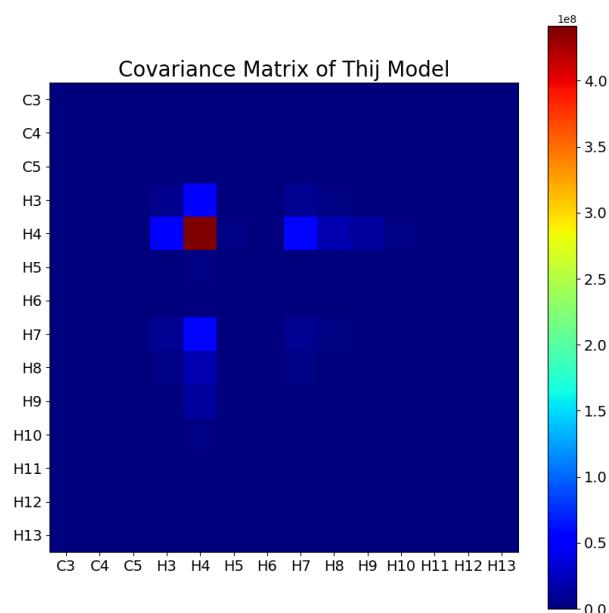


Figure 7.8. $\lambda = 0.2$ and $p = 0.8$. Here the covariance is relatively very small, except for the H4 variance. This is suggestive of the high p generating that induced star subgraph.

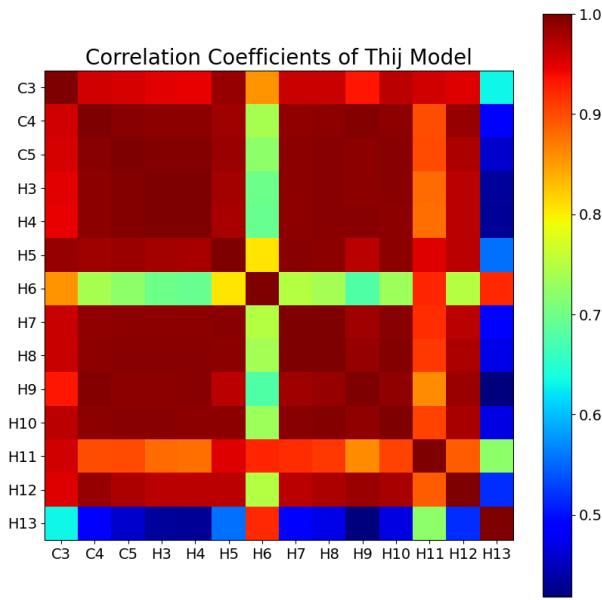


Figure 7.9. $\lambda = 0.8$ and $p = 0.2$. Once again we see a familiar structure in the correlation matrix. Here though we do see strong correlations across a wider selection of motifs: C_3 , C_4 , C_5 , H_3 , H_4 , H_5 , H_7 , H_8 , H_9 , H_{10} . It appears the H_6 's are not easily produced by any of the parameters chosen or the Barabási–Albert model.

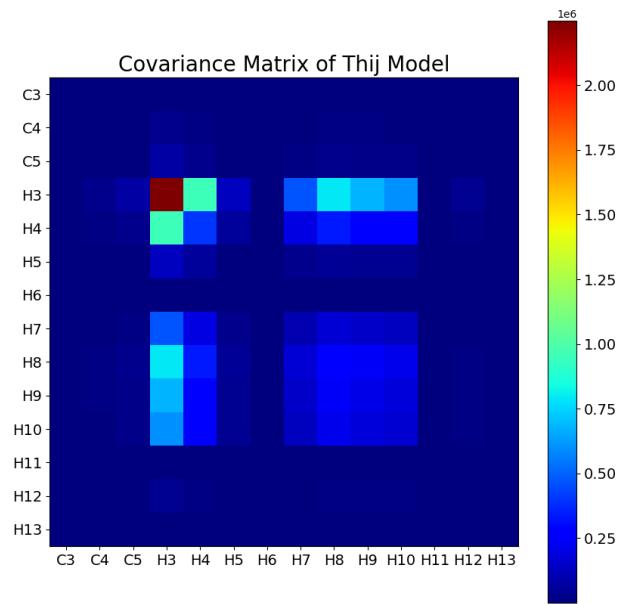


Figure 7.10. $\lambda = 0.8$ and $p = 0.2$. We do see strong covariance around the H_4 with other motifs. There is possibly multiple induced star subgraphs connected that could drive this phenomena generating many H_4 's along with other motifs.

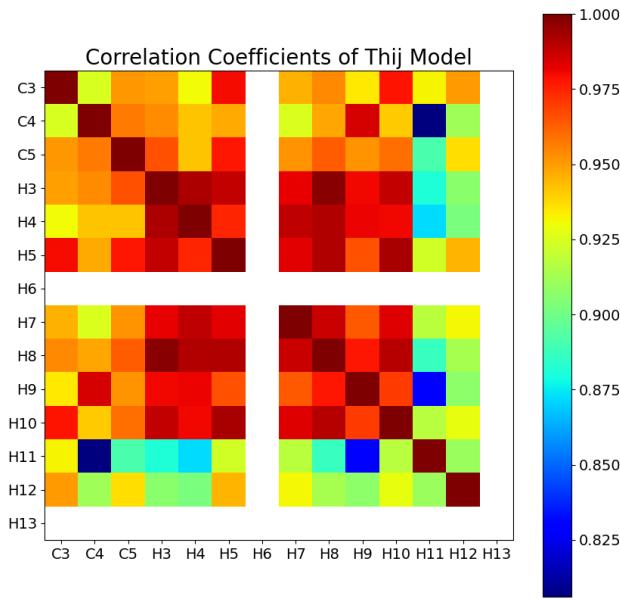


Figure 7.11. $\lambda = 0.8$ and $p = 0.8$. Here we have a higher likelihood of adding a new node or a new edge and node. Here we do not see any new H_6 's or H_8 's produced after the initialization of the graph. Any block structures that we saw in earlier correlation matrices are not as apparent, although there is still strong H_7 - H_8 correlation.

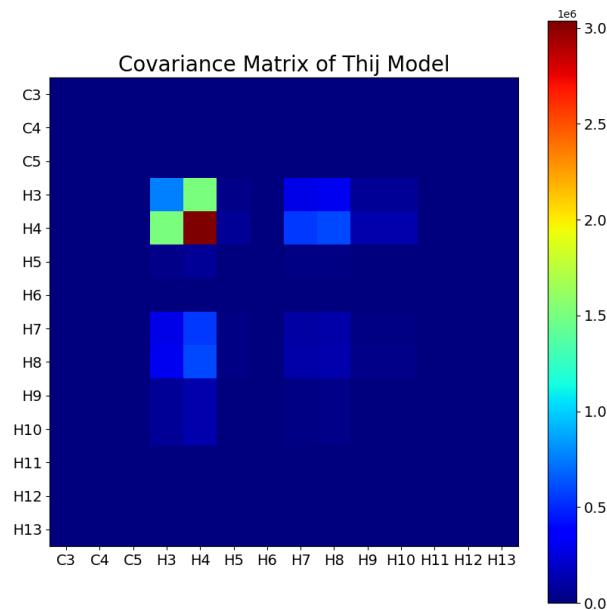


Figure 7.12. $\lambda = 0.8$ and $p = 0.8$. The covariances formed by this simulation suggest once again an overlapping of H_4 's, but there are clearly more nodes attached to the outer edges of this induced subgraph. This could lead to the covariance we see between H_3 and H_4

In figures 7.11 and 7.12, we see clearly evidence of the S_k , $k >> 1$ induced subgraph for high values of p . We see relatively high covariance between the $H7's$ and $H8's$ and with $H3$ and $H3$. $H13$ and $H6$ counts don't change at all because the probabilistically of $T2$ attachment through the superstar mechanism is high relatively high. In fact, this mechanism is what encourages the growth of those motifs. We expect to grow in a combinatorial manner around S_k induced subgraphs.

CHAPTER 8

Dynamic Mode Decomposition

Another way to characterize the development of the motifs, we characterize the evolution of the motifs as a dynamical system. We do this through dynamic mode decomposition to describe the dynamics via approximations to the modes and eigenvalues of the Koopman operator. The DMD algorithm produces spatiotemporal coherent structures (modes) which have associated temporal behavior: growth, decay, and oscillation.

8.1 The Koopman Operator

Suppose we have some continuous, finite-dimensional, non-linear dynamical system

$$\frac{dy}{dt} = f(y) \quad y(0) = x \in \mathbb{R}^N$$

with $N >> 1$. $y(t)$ is the state of the dynamical system at time t . Sampling the dynamical system every Δt we get the discrete time-series

$$y_{k+1} = F(y_k).$$

with $y_k = y(t_k) = y(k\Delta t)$. We would like a coordinate transformation render the dynamics of the non-linear system much simpler. In this new coordinate system the dynamics could be described linearly. We seek some ϕ such that $z = \phi(y)$ where the dynamics are much easier to evaluate in the z -coordinates. In this context, we seek $y_{k+1} = Ky_k$.

We define a Hilbert space of observables, that is

$$L_2(O) = L_2(\mathbb{R}^N, \mathbb{R}, \mu),$$

with an associated norm

$$\int_{\mathbb{R}^N} |g(x)|^2 d\mu(x) < \infty,$$

where μ is some appropriately chosen measure. The Koopman operator K is a mapping between the Hilbert space of observables unto itself,

$$K : L_2(O) \rightarrow L_2(O)$$

such that

$$Kg(x) = g(F(x)) = g(x_{k+1}) \quad g \in L_2(O)$$

For an eigenfunction φ of K and their associated eigenvalue μ we have that

$$K(\varphi(x_k)) = \mu\varphi(x_k) = \varphi(x_{k+1})$$

A vector of observables g can be written in terms of a basis of eigenvectors

$$g(x_k) = \sum_{j=1}^{\infty} \varphi_j(x_k) \xi_j$$

which implies we can evolve the system like so

$$g(x_{k+1}) = \sum_{j=1}^{\infty} \mu_j \varphi_j(x_k) \xi_j.$$

However, finding the exact eigenfunctions, modes, and eigenvalues of K analytically is difficult for any meaningful problem, and thus we have to resort to numerics to estimate them via the Dynamic Mode Decomposition.

8.2 Approximating the Koopman Operator via Dynamic Mode Decomposition

Similar to the previous section we consider now sampled-data collected from a non-linear dynamical system of the form:

$$\frac{dx}{dt} = f(x, t; \mu)$$

where $x(t) \in \mathbb{R}^n$ is a vector representing the state of the dynamical system at time t and μ are parameter values. Our data is sampled at discrete points in time every Δt . Every sampling can be written as $x_k = x(k\Delta t)$. This discrete time-flow map we denote $X_{k+1} = F(x_k)$. We can then extend our collected data:

$$y_k = g(x_k)$$

where $g(k)$ is the set of observables formed from the state space. Here we now use the Koopman operator theory to introduce a linear, infinite-dimensional operator A :

$$Ag = Ag(x_k) = g(x_{k+1}),$$

exactly as we described in section 8.1. A advances measurements along the flow F by Δt . We calculate the Koopman operator of our dynamical system in the following manner [15]. Beginning with our data snapshot matrix composed of the relevant observables we write

$$G = [g(x_0), g(x_1), g(x_2), \dots, g(x_n)] = [g_1, g_2, \dots, g_n],$$

which we can then break into two matrices:

$$\begin{aligned} G_+ &= [g_1, g_2, g_3, \dots, g_n] \\ G_- &= [g_0, g_1, g_2, \dots, g_{n-1}] \end{aligned}$$

The matrix G_+ is simply our matrix G_- taken forward one step in time. The particular Koopman operator A we wish to find, satisfies:

$$G_+ = AG_-$$

Subtracting AG_- from both sides, we have

$$G_+ - AG_- = 0$$

We would then like to find that

$$\tilde{A} = \arg \min_A \|G_+ - AG_-\|_F$$

where $\|\cdot\|_F$ denotes the Frobenius norm. We still have the matter of finding the \tilde{A} that minimizes the norm, which is our best approximation to A . We could use a least-squares fit or solve using Moore pseudo-inverse caculating $G_+G_-^\dagger$, but we instead make use of the r-rank singular value decomposition in the following way.

$$\begin{aligned} G_+ &= \tilde{A}G_- \\ &= \tilde{A}U_rS_rV_r^T \quad \text{by singular value decomposition} \\ G_+V_rS_r^{-1}U_r^T &= \tilde{A} \end{aligned}$$

Taking the r -rank of the SVD limits the number of modes and eigenvalues to a few that satisfy a threshold tolerance giving us a low-rank model that still captures

most of the information. The columns of U are the Proper Orthogonal Decomposition modes of the data. The associated singular values are normalized by their total. The singular values are weights to each mode and can be used to discard those modes which don't contribute a substantial amount of energy in the decomposition.

Finally, we use an eigendecomposition on the left-hand side to find the eigenvalues and associated modes of A , our Koopman modes. Once we have the Koopman modes and eigenvalues, we can begin to look at which DMD modes "contribute the most" throughout the snapshot matrix and the associated eigenvalues which describe their temporal behavior: growth, decay, or oscillation. We have not yet discussed how to choose g , how we build our dictionary of observables. If we choose g to be the identity mapping, then the method reduces to the standard DMD algorithm, where $g(x_i) = x_i$. Choosing any other function of g in a meaningful way requires some a-priori knowledge of the system. A mapping g can extend the dictionary of observables beyond the state space using some appropriate basis, usually a Taylor or Fourier basis. This is aptly called Extended Dynamic Mode Decomposition (EDMD) [22]. A quick example of such a choice might be:

$$Y = \{y_1, y_2, y_3, \dots, y_n\}, \quad y_i \in \mathbb{R}^N$$

with

$$y_i = y_{i,1}, y_{i,2}, \dots, y_{i,N}$$

then we might take g such that:

$$G(Y) = \{g(y_1), g(y_2), g(y_3), g(y_4), \dots, g(y_n)\}$$

$$g(y_i) = y_{i,1}, y_{i,2}, \dots, y_{i,N}, y_{i,1}^2, y_{i,1}y_{i,2}, \dots, y_{i,N}^2$$

This example of g is one of many possible functions.

8.3 Kernel Dynamic Mode Decomposition

EDMD can quickly generate a large set of observables and the computational complexity of the problem increases rapidly. It suffers from the dimensionality problem and works well provided many more snapshots than state-space variables. If one is analyzing for instance two-dimensional fluid flows, the flattened matrix-snapshot can easily have thousand of entries. Extending the state-space by building a dictionary with nonlinear combinations of the state space will generate very large matrices [33]. For

that reason one applies the kernel trick. One can use the kernel function to "collapse" the information of many nonlinear terms through a kernel function. In our analysis our vectors are relatively small and thus not a conventional use of KDMD. KDMD still offers a way to analyze nonlinear terms through the kernel trick and as we will see in chapters 9, can produce Koopman modes with relatively low mode error. These modes, but not their associated eigenfunctions, are similar in structure to the one's produced by DMD, but not identical [22]. We introduce the notion of the kernel.

Definition 14. *We define the Kernel function*

$k : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$, $k(x, \hat{x}) = \langle \phi(x), \phi(\hat{x}) \rangle$. That is k maps a pair of vectors to an inner product of observables of the data.

There are a variety of different kernels that we may choose. Some include:

Sigmoid:

$$f(x, x') = \tanh(x^T x' + a)$$

Radial:

$$f(x, x') = e^{-a|x-x'|^2}$$

Polynomial:

$$f(x, x') = (1 + x^T x')^p$$

This allows us to store a large amount of information in a single value. We generate matrices Φ^+ and Φ^- from the polynomial basis function. Our data snapshots are as follows:

$$Y = \{y_1, y_2, y_3, \dots, y_n\}$$

$$k(y, \hat{y}) = (1 + y^T \hat{y})^p$$

We then construct the observable matrices $Y_-^T Y_+ = \Phi^+$ and $Y_-^T Y_- = \Phi^-$

$$\Phi_{i,j}^+ = k(Y_i^+, Y_j^-)$$

$$\Phi_{i,j}^- = k(Y_i^-, Y_j^-)$$

Then for every element in $\Phi_{i,j}^+$ and $\Phi_{i,j}^-$, we have a kernel between two snapshots in time. And the dimensionality of the extended observables is not constrained by the total number of snapshots. For KDMD, we finally get our approximations to the Koopman modes in the familiar way, by applying the SVD and my matrix multiplication.

$$\Phi^+ = \bar{A}\Phi^-$$

8.4 Accuracy Criterion

For the DMD and KDMD algorithms, we evaluate how well the methods performed. We consider two different measures of accuracy. First we define the one-step reconstruction error r . We define the matrix \tilde{G} where the j 'th column vectors of \tilde{G} are the one-step reconstructions of the state space at time $j + 1$ for $j = 0, 1, \dots, n - 1$.

$$\begin{aligned}\tilde{G}_j &= \sum_i \mu_i b_{i,j} \xi_i \\ r &= \frac{\|G_+ - \tilde{G}\|_F}{\|G_+\|_F}\end{aligned}$$

where $\|\cdot\|_F$ denotes the Frobenius norm.

We also wish to know if the approximated modes and eigenvalues are good approximations to Koopman modes. Let ξ_j and λ be an approximated Koopman mode for the dynamical system F and its corresponding eigenvalue. Provided that ξ_j and λ_j are a true eigenpair it follows:

$$\xi_j \circ F = \lambda_j \xi_j$$

Now, letting $\|\cdot\|$ denotes the $L2$ norm, we would like to calculate

$$\frac{\|\xi \circ F - \lambda \xi\|}{\|\xi\|}$$

However, if we know F then Dynamic Mode Decomposition wouldn't be a useful tool. We must then estimate F using a finite number of data points. We define two data points $x_k, x_{k+1} \in X$. We can now write

$$\tilde{r}_j = \frac{\sum_k |\varphi_j(x_{k+1}) - \lambda_j \varphi_j(x_k)|}{\sum_k |\varphi_j(x_k)|}$$

We are measuring how much each eigenfunction ϕ_j behaves like a Koopman eigenfunction. We can use this to evaluate each mode individually. This equation gives the mode error for the j 'th mode as it determines how faithfully the Dynamic Mode Decomposition produces a set of modes similar to those of the Koopman Operator [35].

CHAPTER 9

Results

9.1 Preprocessing Data

Dynamic Mode Decomposition is sensitive to noise in the data. We implement several techniques to better the errors of the DMD modes. First, we modify the data via a Poisson process such that the events do not occur at each snapshot, but the occurrence of events occurs relative to a δt . To illustrate, suppose we have

$$X = [x_1, x_2, x_3, \dots, x_n]$$

where x_i describes the motif counts at $i\Delta t = t$. We then allow

$$A = \frac{1}{2} \left(\frac{1}{\lambda} + \psi(\lambda) \right)$$

$$\delta t = \min(A)$$

$\psi(\lambda)$ is a vector of samples drawn from the exponential distribution with the same number of time samples and λ is the exponent of the distribution. We then distribute events in the following way. Let \hat{A} denote a matrix that initially only contains the value x_1 . For $k = 1, 2, 3, \dots$ we calculate $k\delta t$. While $k\delta t < A_i$ we append x_i to a \hat{A} . Thus we generate a new matrix \hat{A} that now contains $n = \sum_{i=1}^N A_i \bmod (\delta t)$ snapshots where the interval of times between events are distributed by a Poisson process. We smooth the data with a window-average. This smoothing will help the DMD and KDMD generate better fits. A last consideration is to center the data about the mean.

9.2 DMD: Barabási–Albert m=1

We now reach the results of the DMD algorithm applied to the motif counts of the simulated networks. We will first examine DMD applied to the Barabási–Albert motif counts. Thereafter, we examine several parameter choices for the Thij model. For all DMD and KDMD results, the Koopman modes and eigenvalues are as discussed in chapter 8. The phi modes are simply the weighting of each Koopman mode at each time step. These tell us how much energy the respective Koopman mode contributes at a given time step.

We can now consider DMD results of the Barabási–Albert motif counts. We show in this section the $m = 1$ and $m = 2$ case to observe differences and similarities between the approximated Koopman modes and eigenvalues.

For the $m = 1$ case, we know that a new node and edge are added at every time step. Moreover from our correlation and covariance heat maps we know the only motifs that change in the $m = 1$ case are as such:

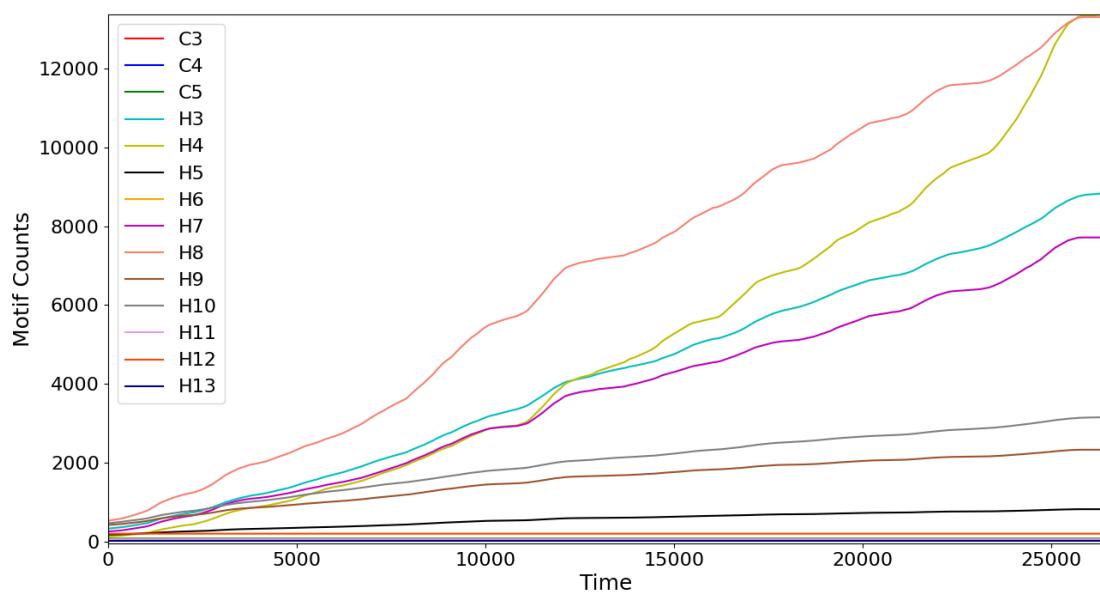


Figure 9.1. Smooth motif counts for Barabási–Albert model $k = 1$.

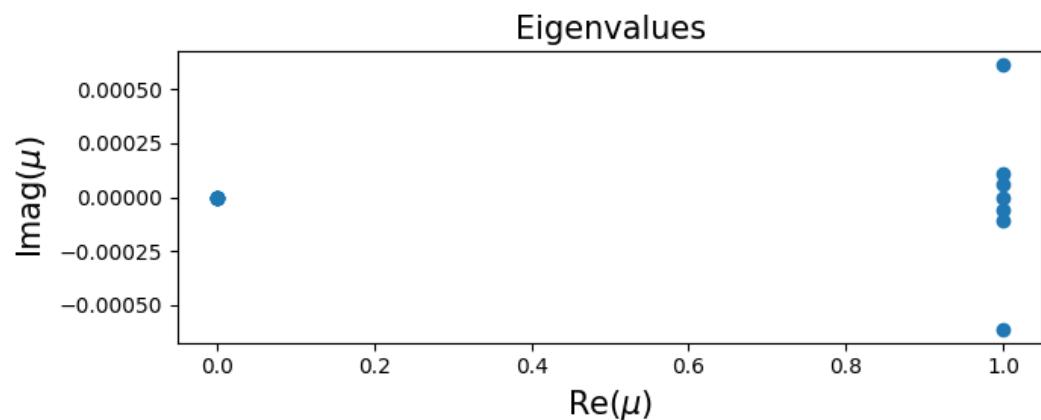


Figure 9.2. Mode Eigenvalues for Barabási–Albert model with $k = 1$.

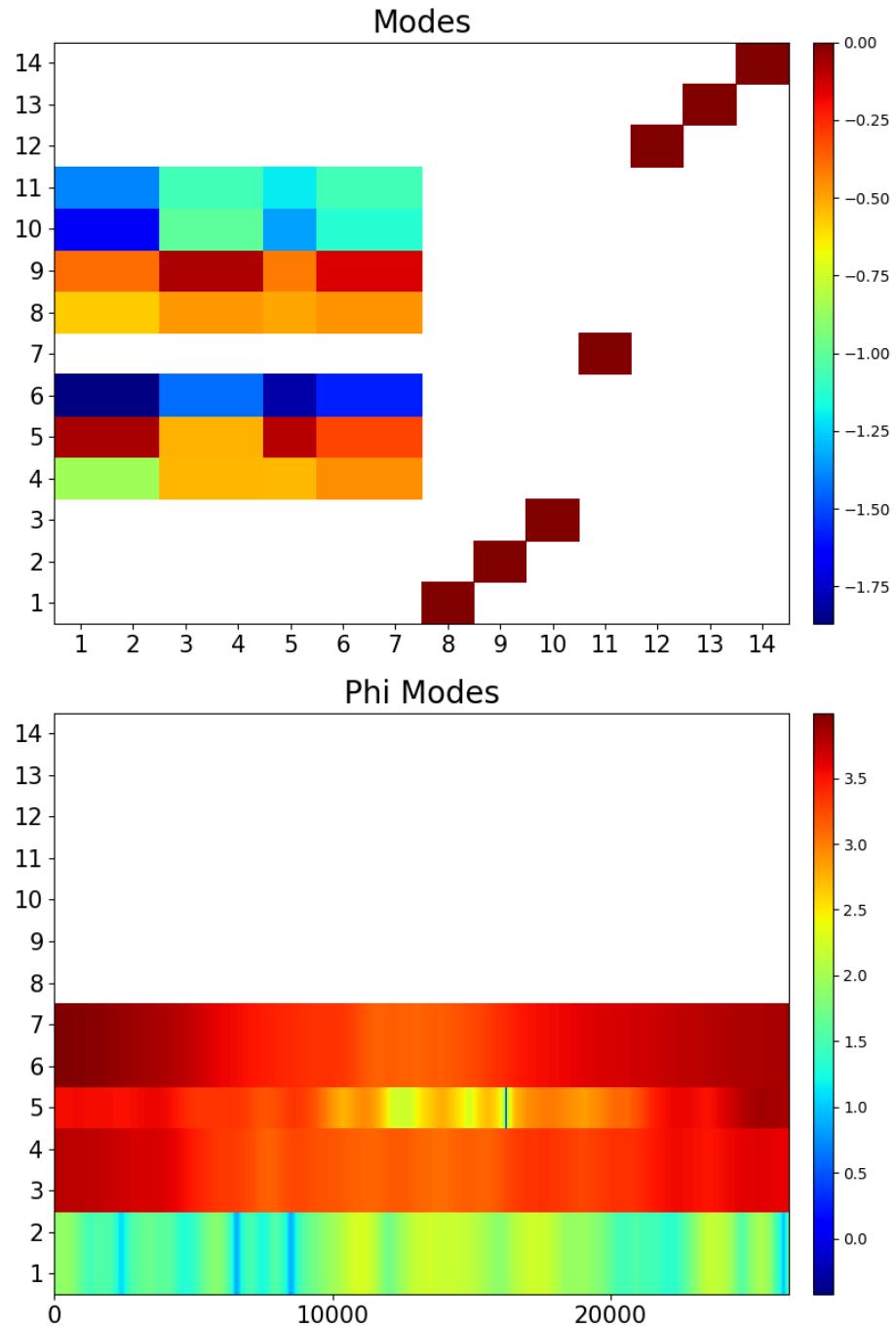


Figure 9.3. Koopman modes by DMD for Barabási–Albert model with $k = 1$.

9.3 KDMD: Barabási-Albert m=1

We now examine the same simulations through the KDMD method.

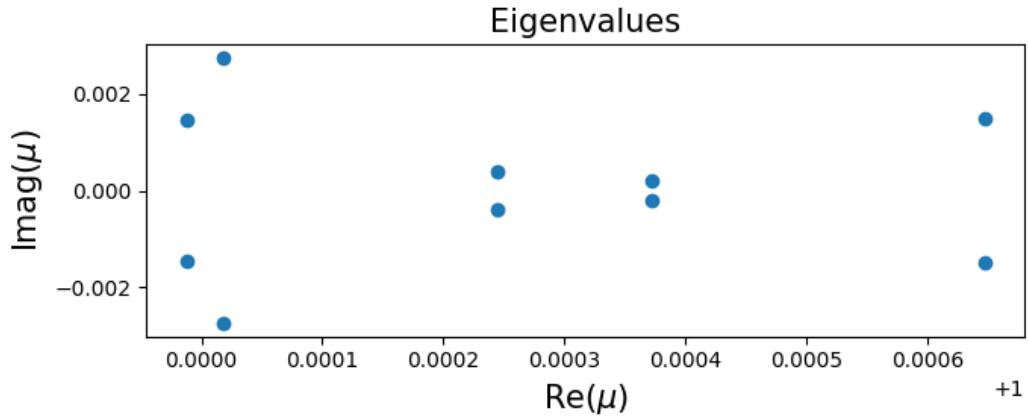


Figure 9.4. Mode Eigenvalues by KDMD for Barabási–Albert model with $k = 1$.

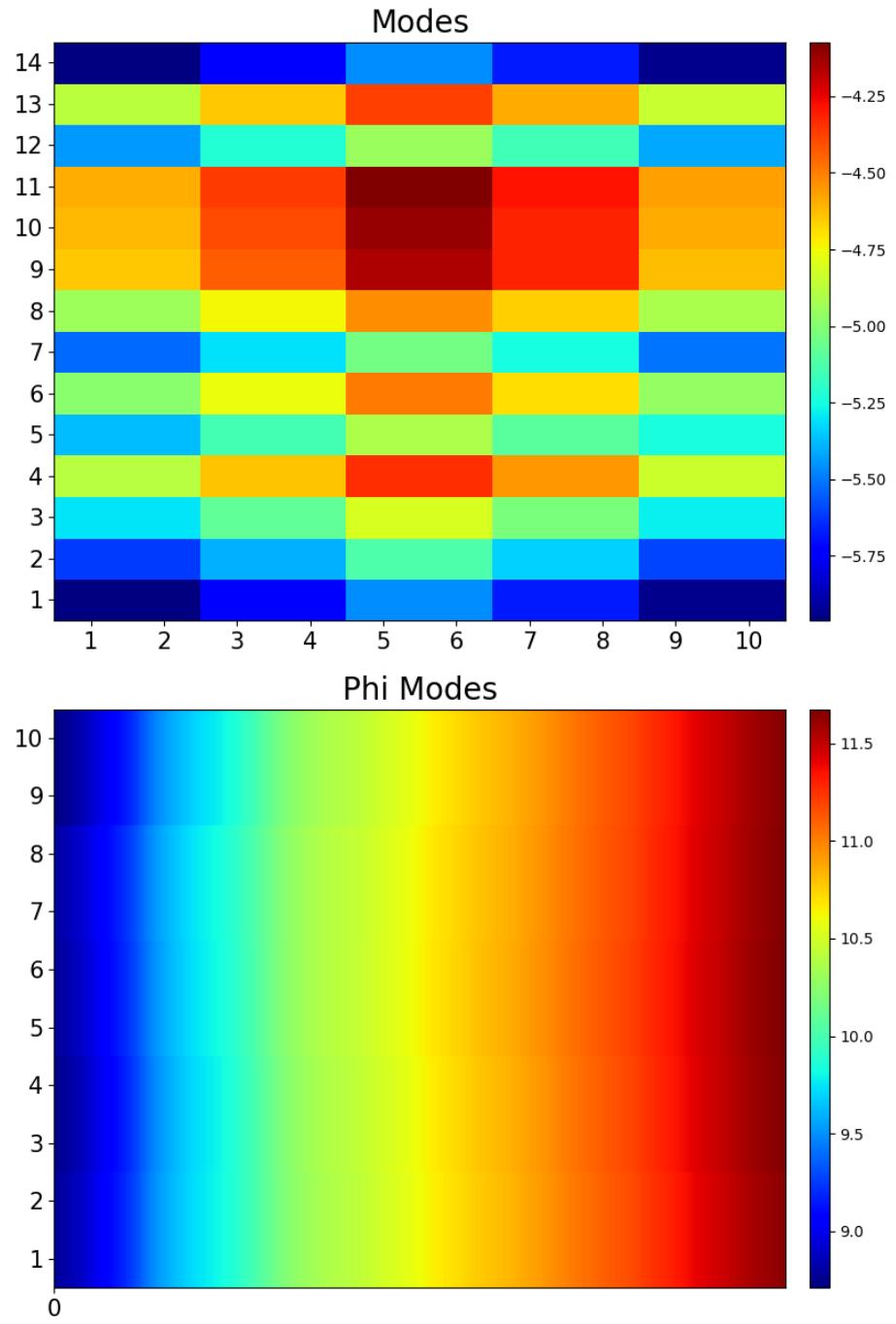


Figure 9.5. Koopman modes by KDMD for Barabási–Albert model with $k = 1$.

9.4 DMD: Barabási–Albert m=2

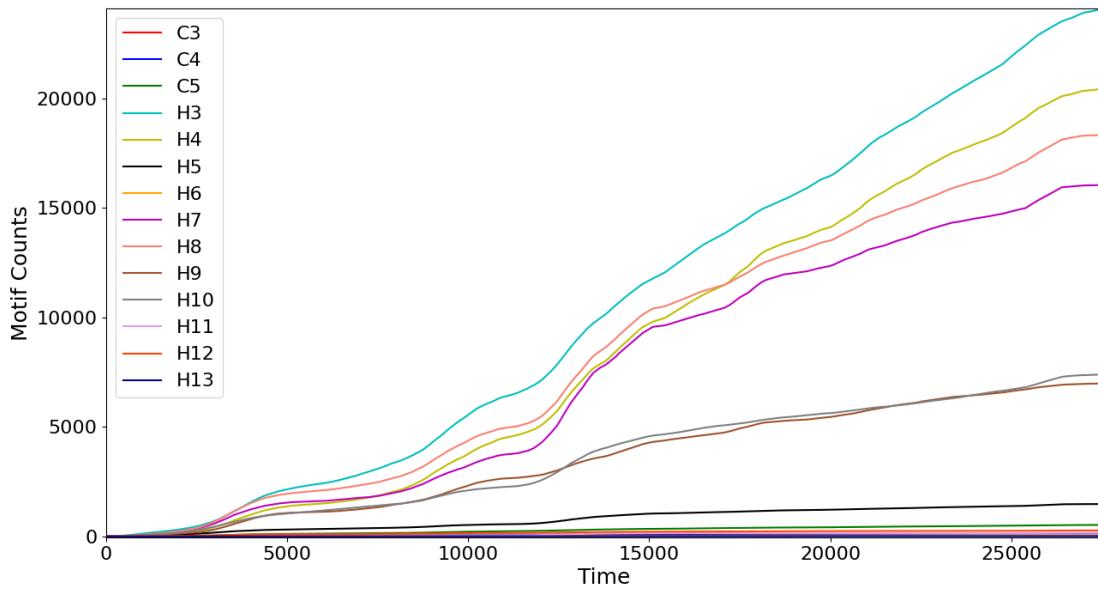


Figure 9.6. Smooth motif counts for Barabási–Albert model $k = 2$.

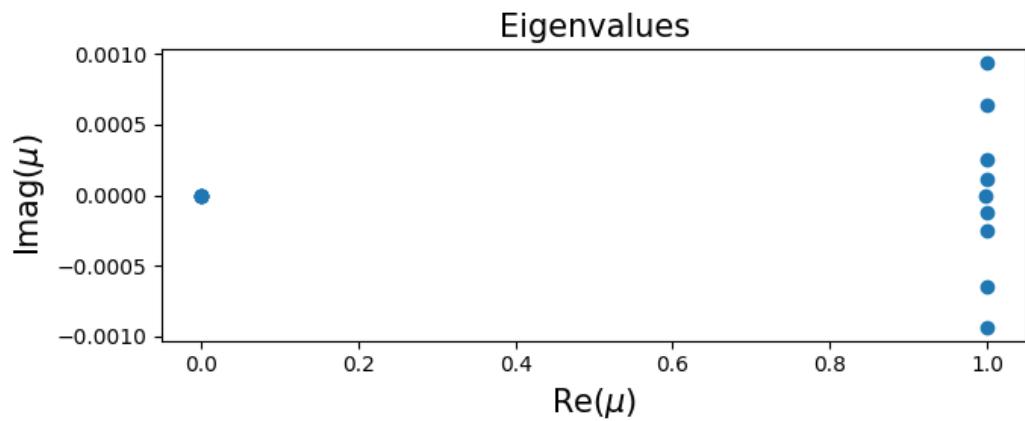


Figure 9.7. Koopman modes by DMD for Barabási–Albert model with $k = 2$.

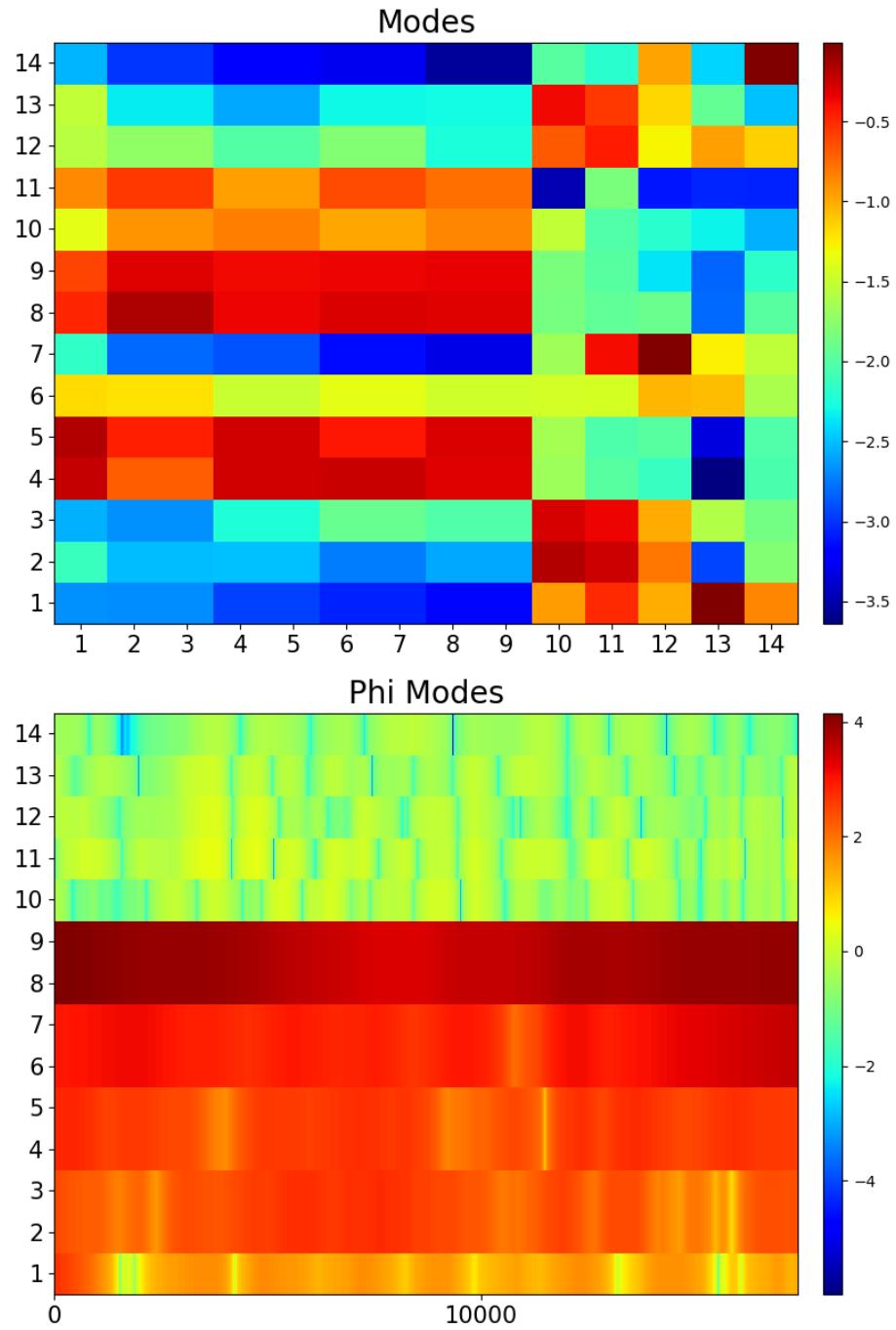


Figure 9.8. Mode eigenvalues by DMD for Barabási–Albert model with $k = 2$.

9.5 KDMD: Barabási–Albert m=2

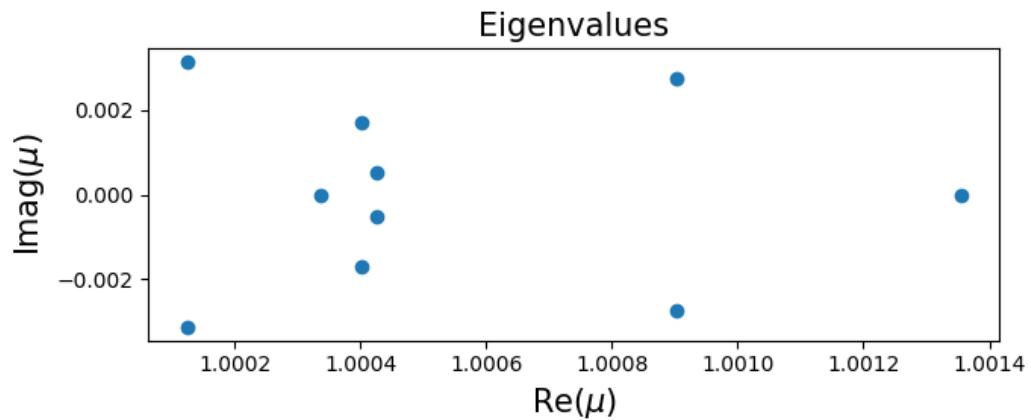


Figure 9.9. Mode eigenvalues by KDMD for Barabási–Albert model with $k = 2$.

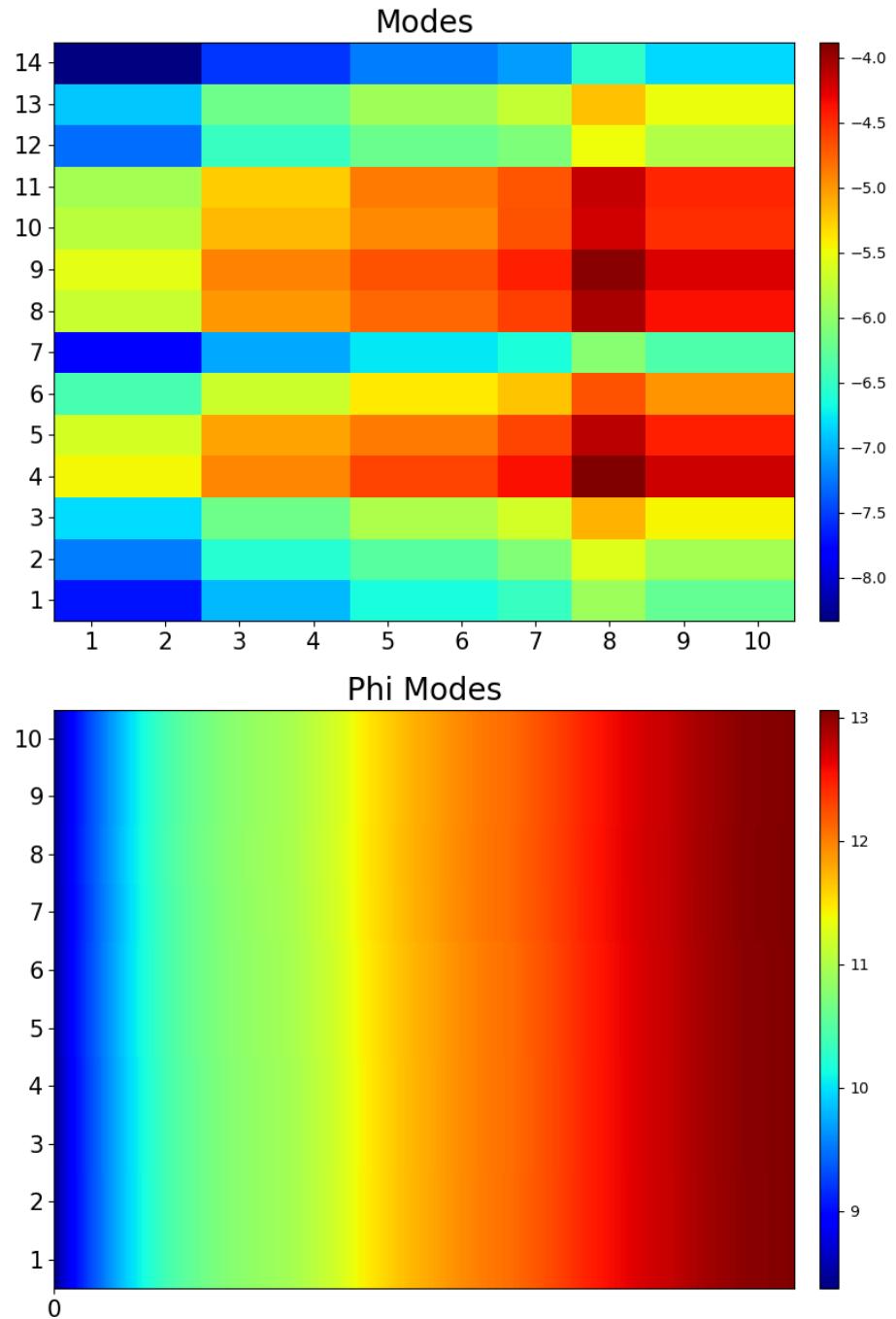


Figure 9.10. Koopman modes by KDMD for Barabási–Albert model with $k = 2$.

9.6 DMD: Thij Model with $\lambda = 0.2$, $p = 0.2$

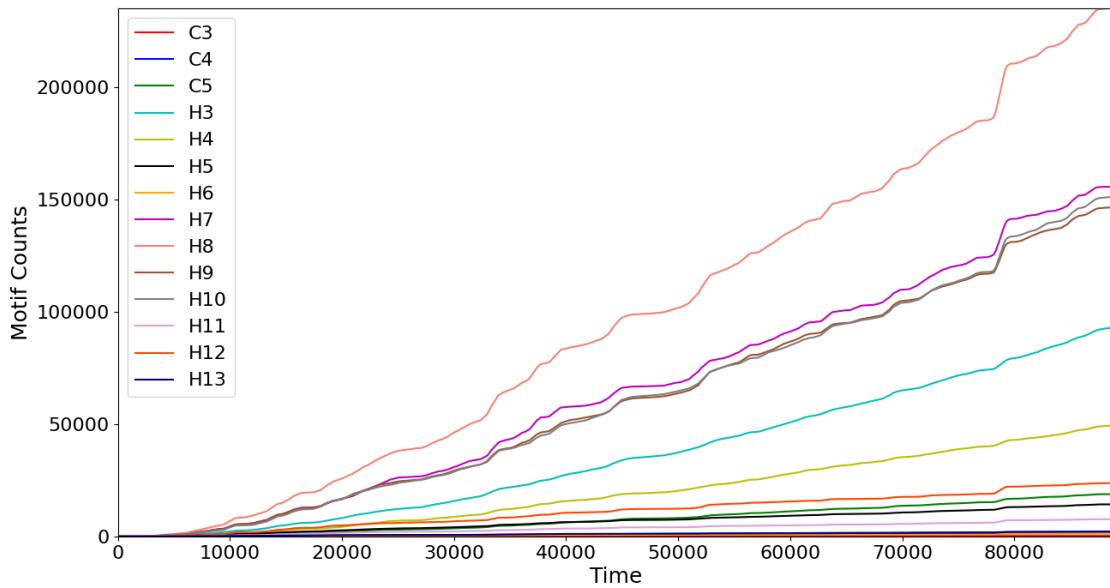


Figure 9.11. Smooth motif counts for Thij model with $\lambda = 0.2$, $p = 0.2$.

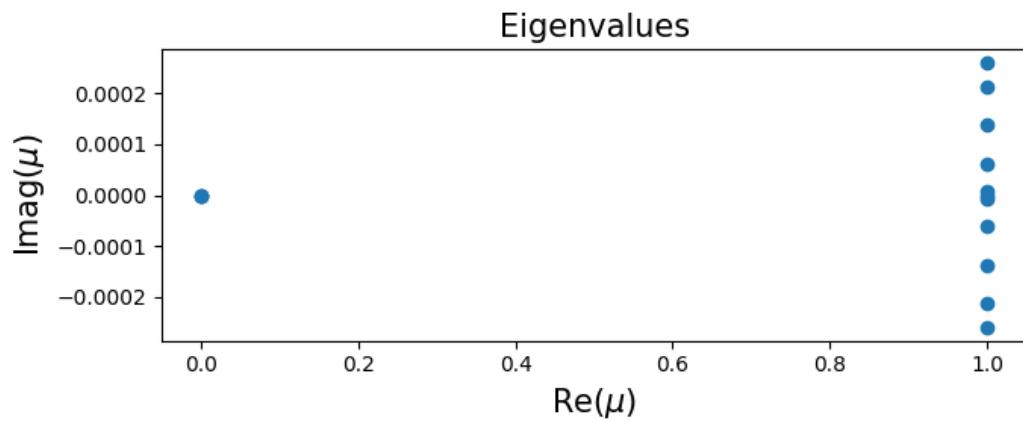


Figure 9.12. Mode eigenvalues by DMD for Thij model with $\lambda = 0.2$, $p = 0.2$.

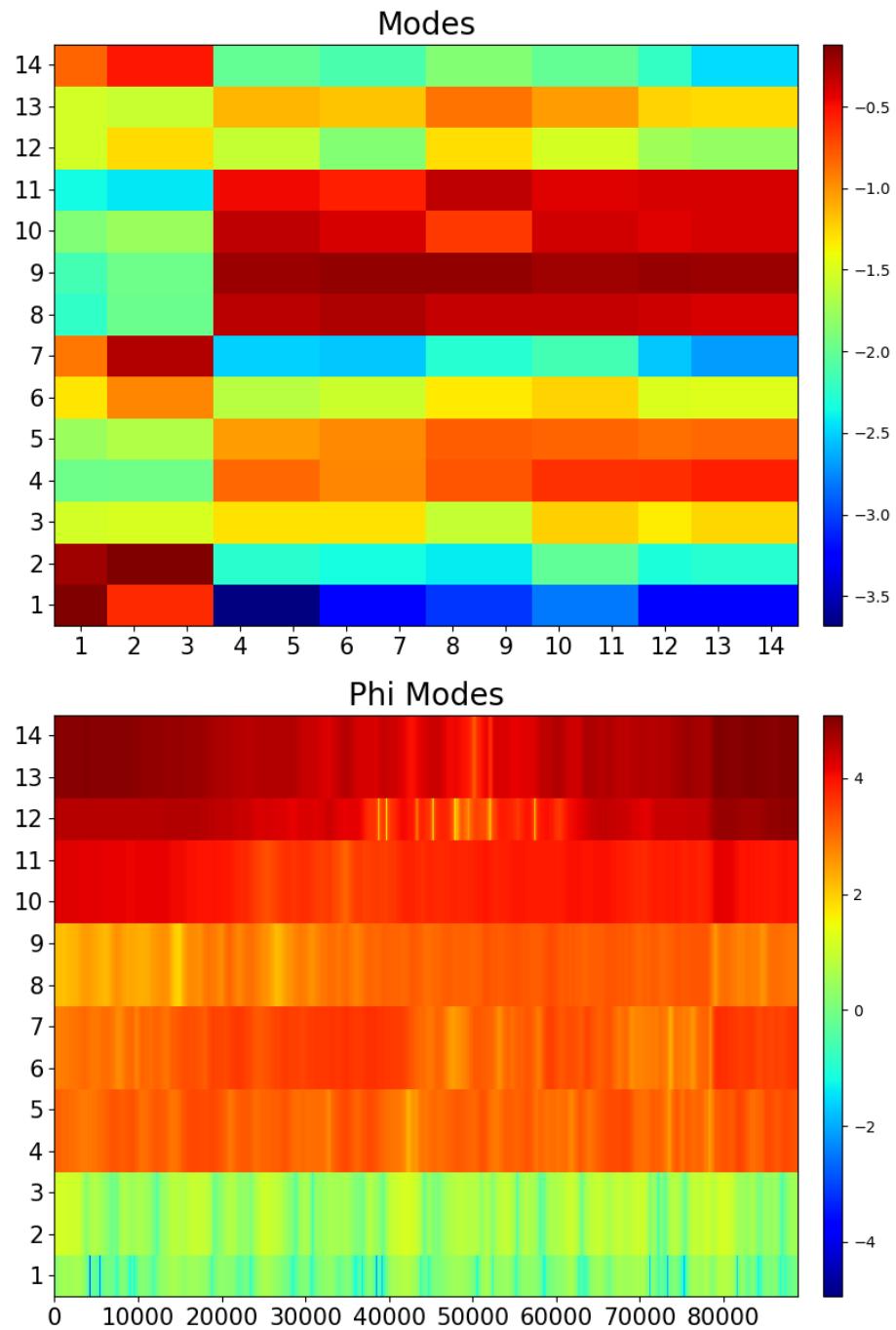


Figure 9.13. Koopman Modes by DMD for Thij model with $\lambda = 0.2$, $p = 0.2$.

9.7 KMD: Thij Model with $\lambda = 0.2, p = 0.2$

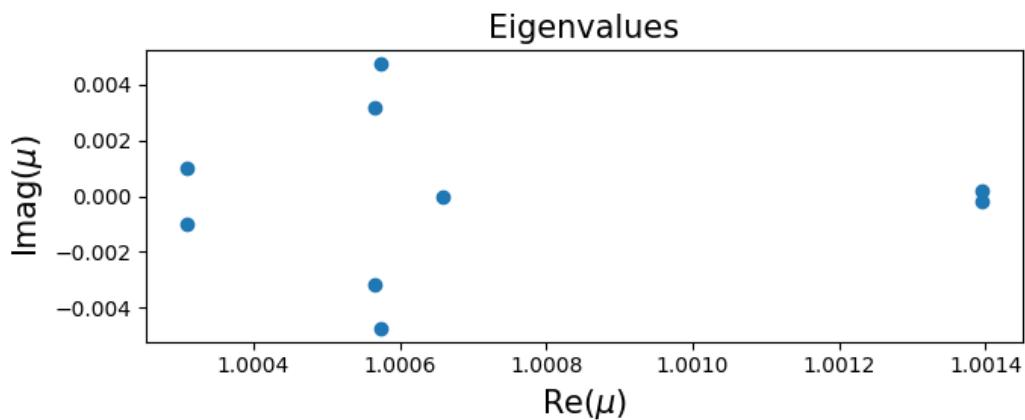


Figure 9.14. Mode eigenvalues by KMD for Thij model with $\lambda = 0.2, p = 0.2$.

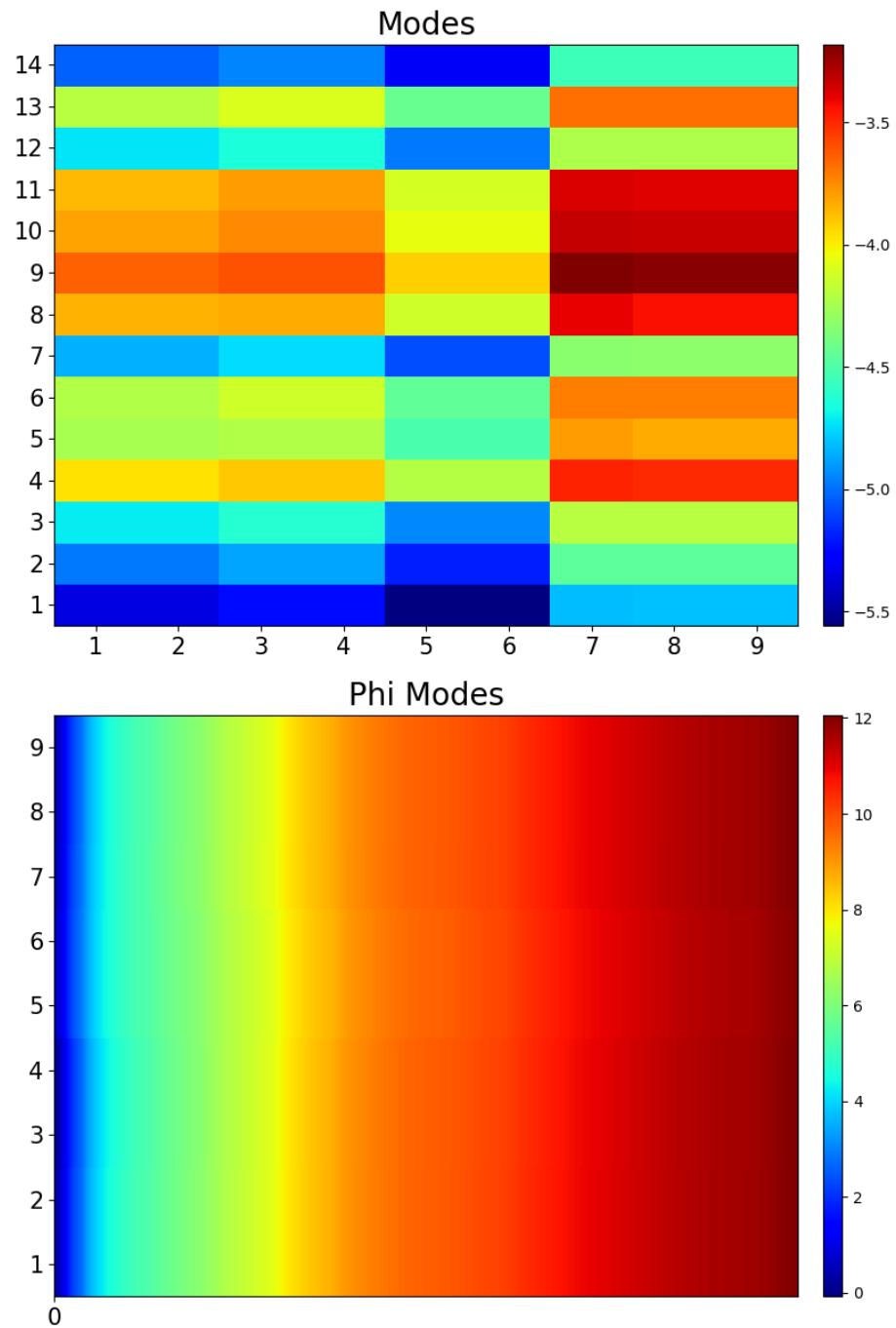


Figure 9.15. Koopman modes by KDMD for Thij model with $\lambda = 0.2$, $p = 0.2$.

9.8 DMD: Thij Model with $\lambda = 0.2$, $p = 0.8$

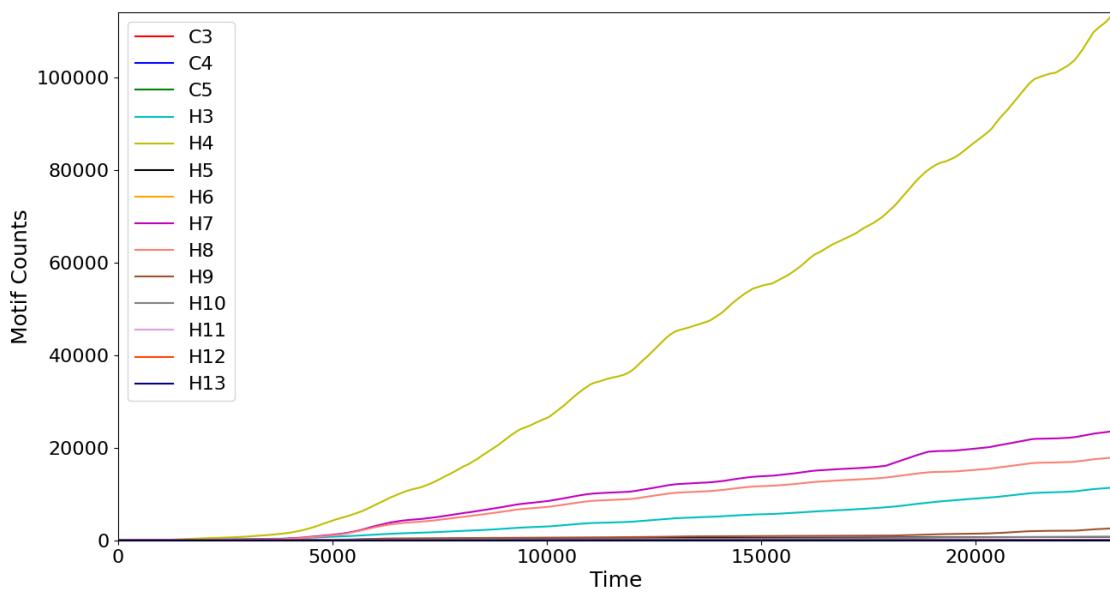


Figure 9.16. Motif counts for Thij model with $\lambda = 0.2$, $p = 0.8$.

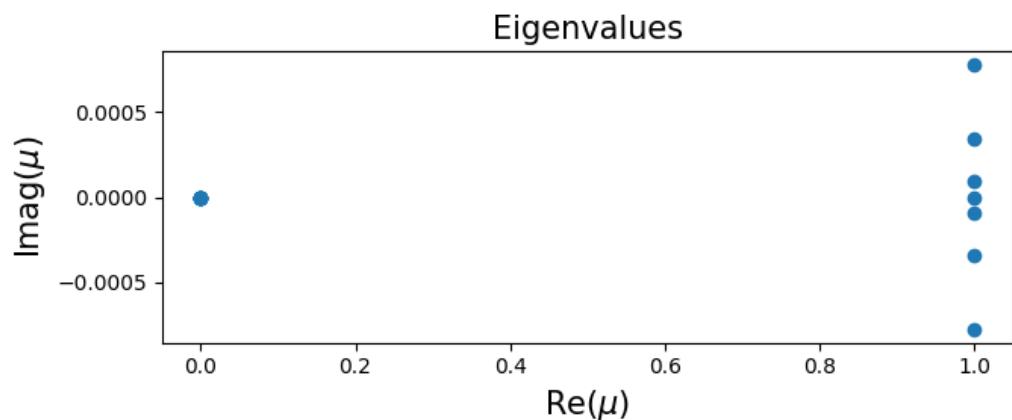


Figure 9.17. Mode eigenvalues by DMD for Thij model with $\lambda = 0.2$, $p = 0.8$.

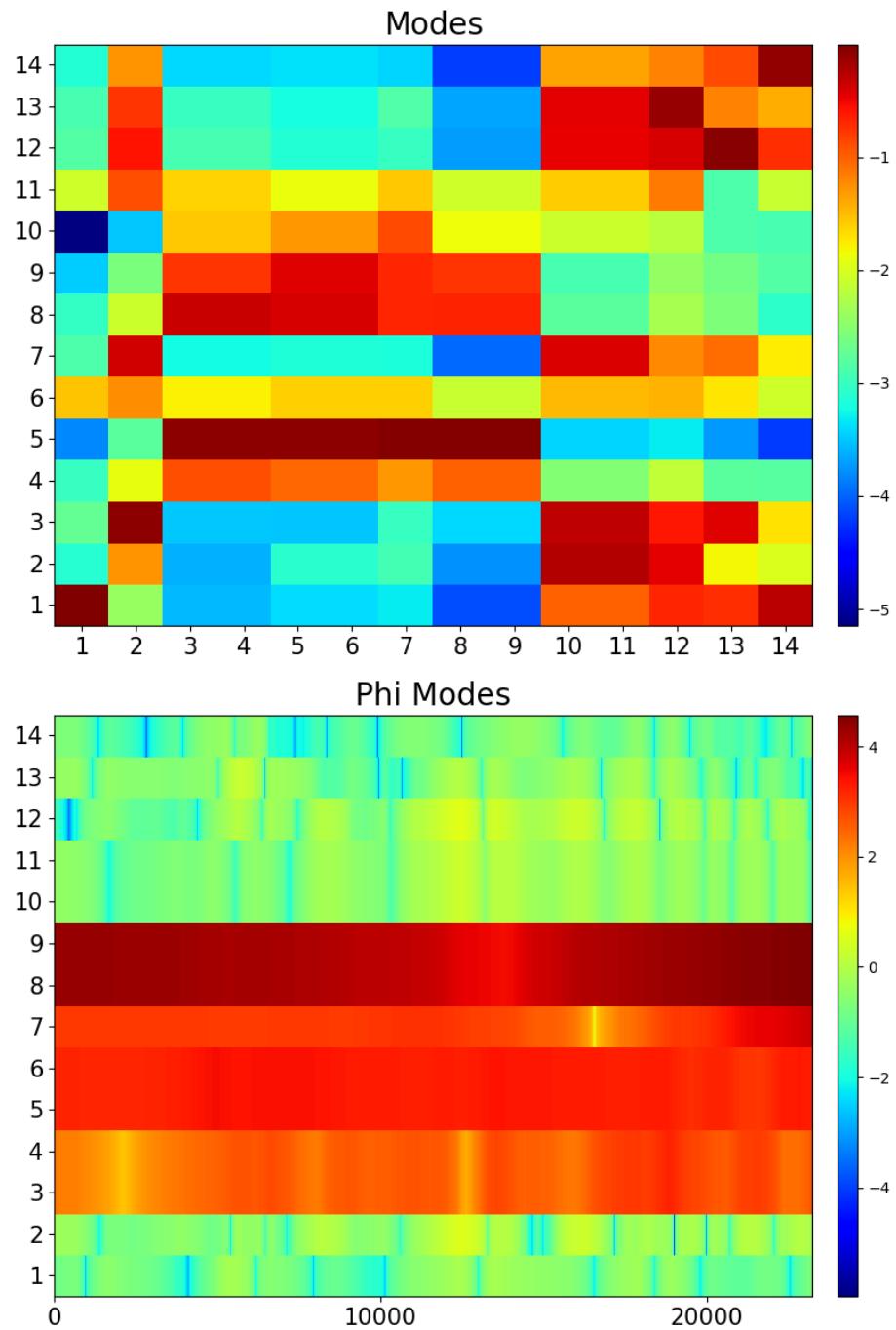


Figure 9.18. Koopman modes by DMD for Thij model with $\lambda = 0.2$, $p = 0.8$.

9.9 KMD: Thij Model with $\lambda = 0.2, p = 0.8$

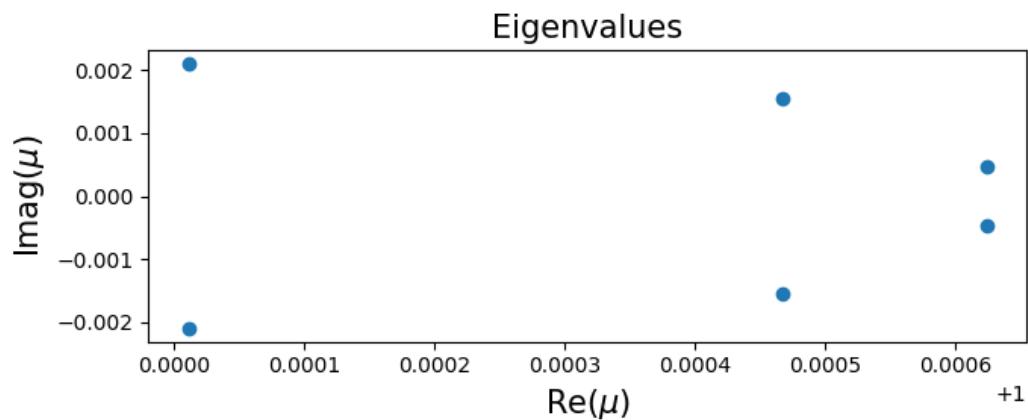


Figure 9.19. Mode eigenvalues by KMD for Thij model with $\lambda = 0.2, p = 0.8$.

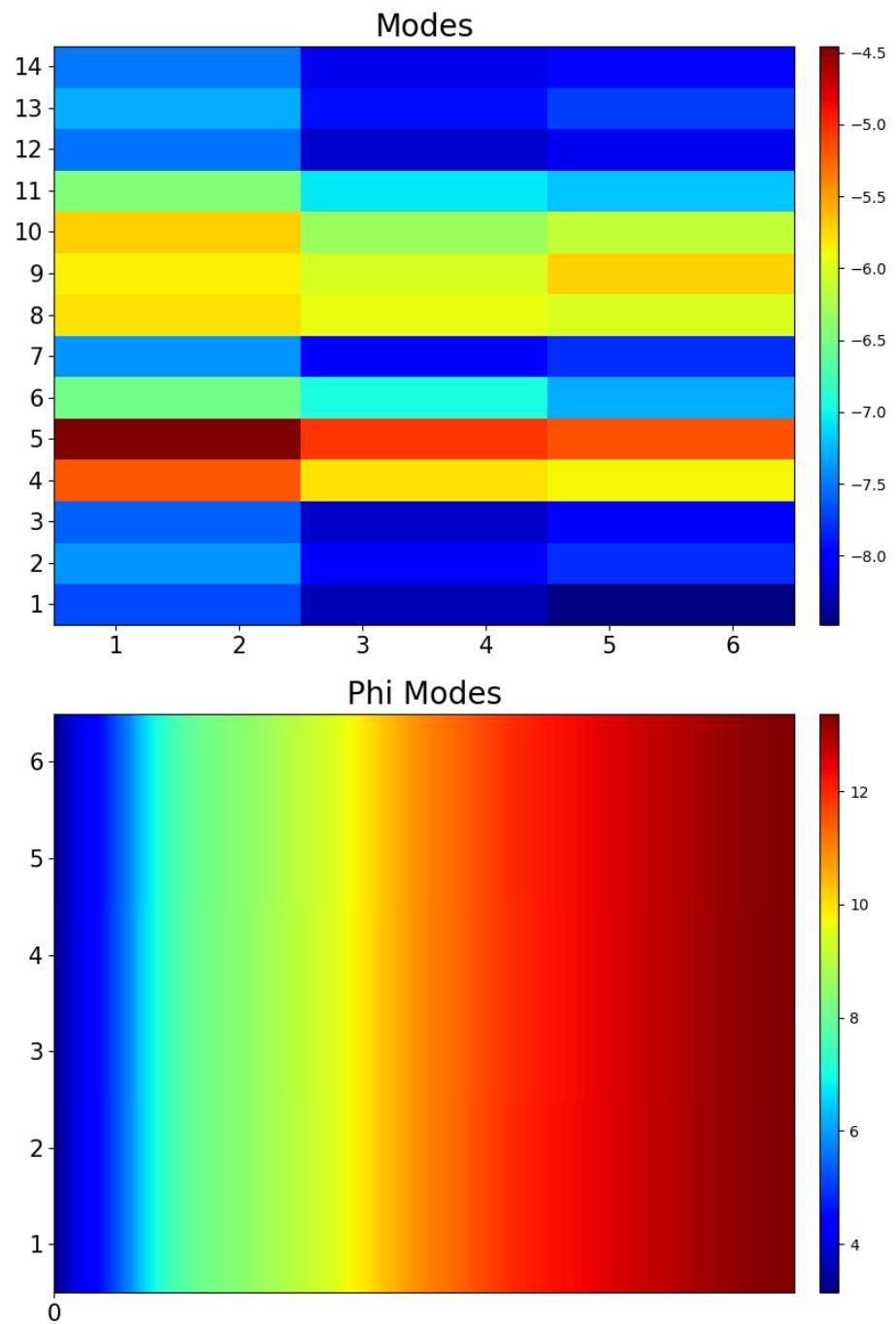


Figure 9.20. Koopman modes by KDMD for Thij model with $\lambda = 0.2$, $p = 0.8$.

9.10 DMD: Thij Model with $\lambda = 0.8$, $p = 0.2$

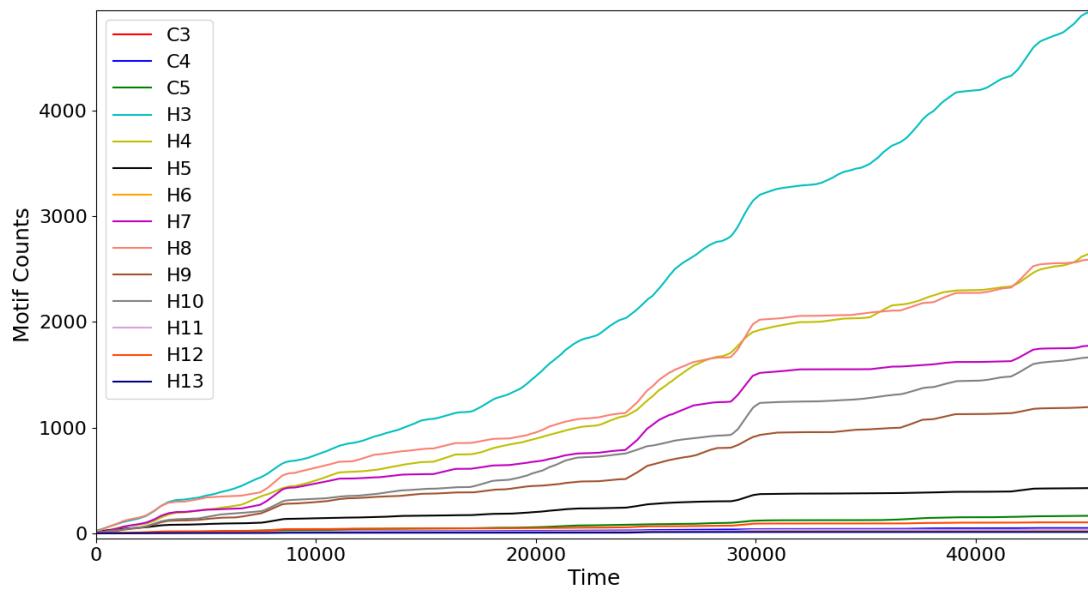


Figure 9.21. Motif counts for Thij simulation with $\lambda = 0.8$, $p = 0.2$

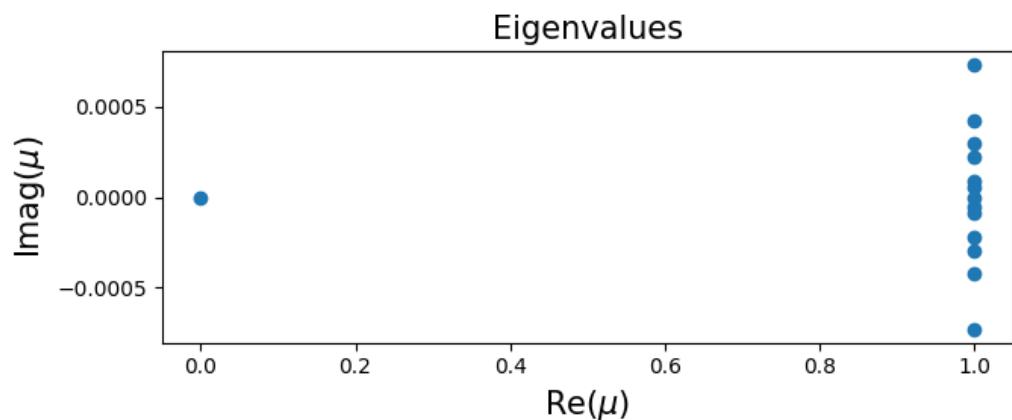


Figure 9.22. Mode eigenvalues by DMD for Thij model with $\lambda = 0.8$, $p = 0.2$.

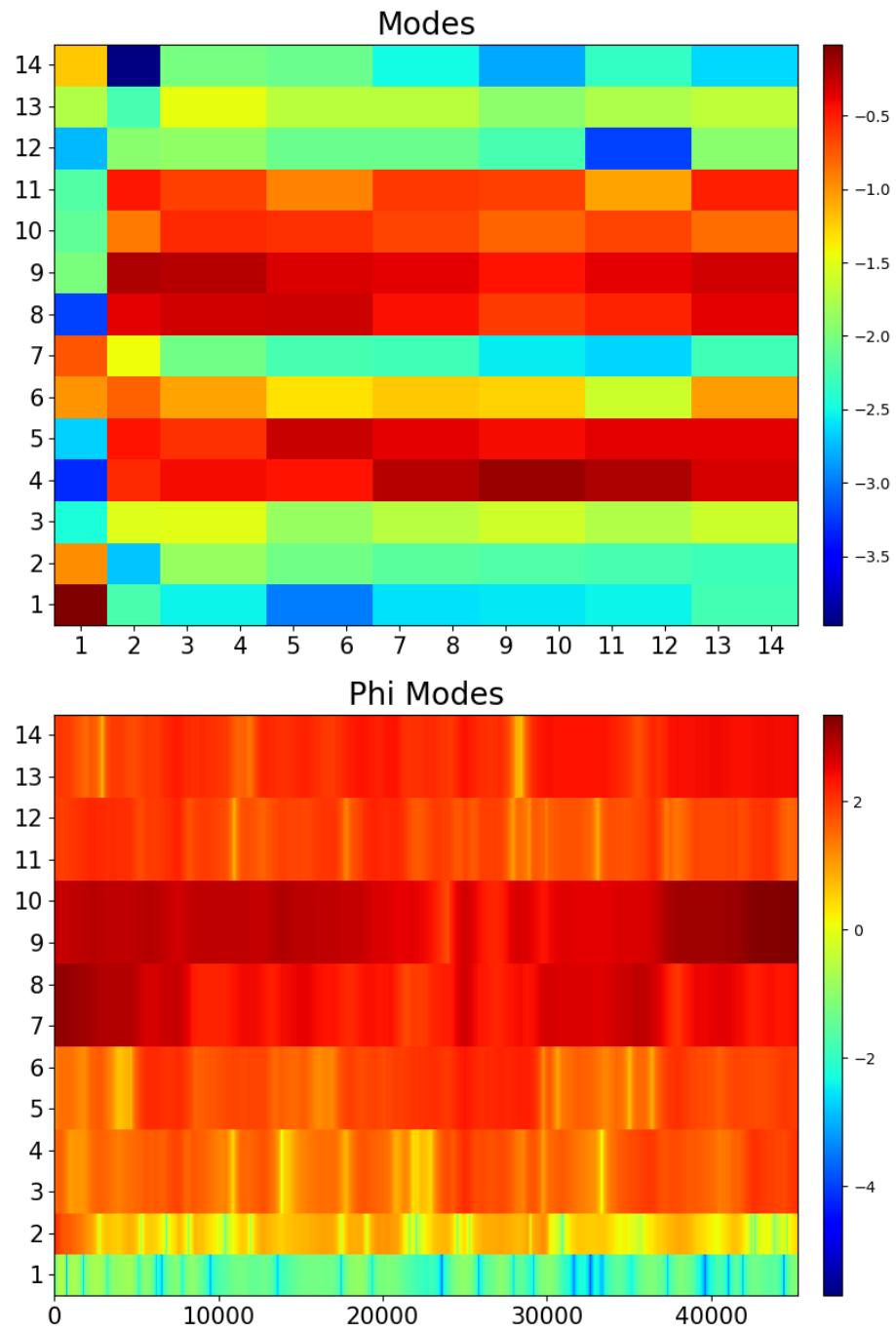


Figure 9.23. Koopman modes and phi modes by DMD for Thij model with $\lambda = 0.8$, $p = 0.2$.

9.11 KDMD: Thij Model with $\lambda = 0.8, p = 0.2$

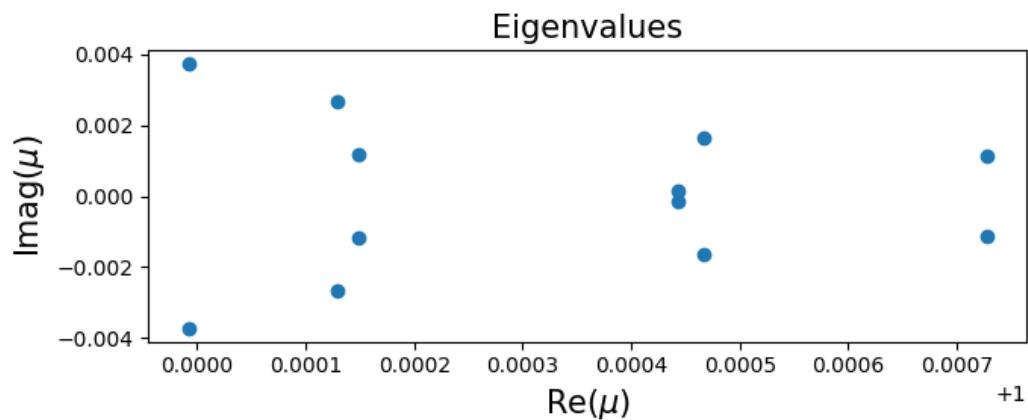


Figure 9.24. Mode eigenvalues by KDMD for Thij model with $\lambda = 0.8, p = 0.2$.

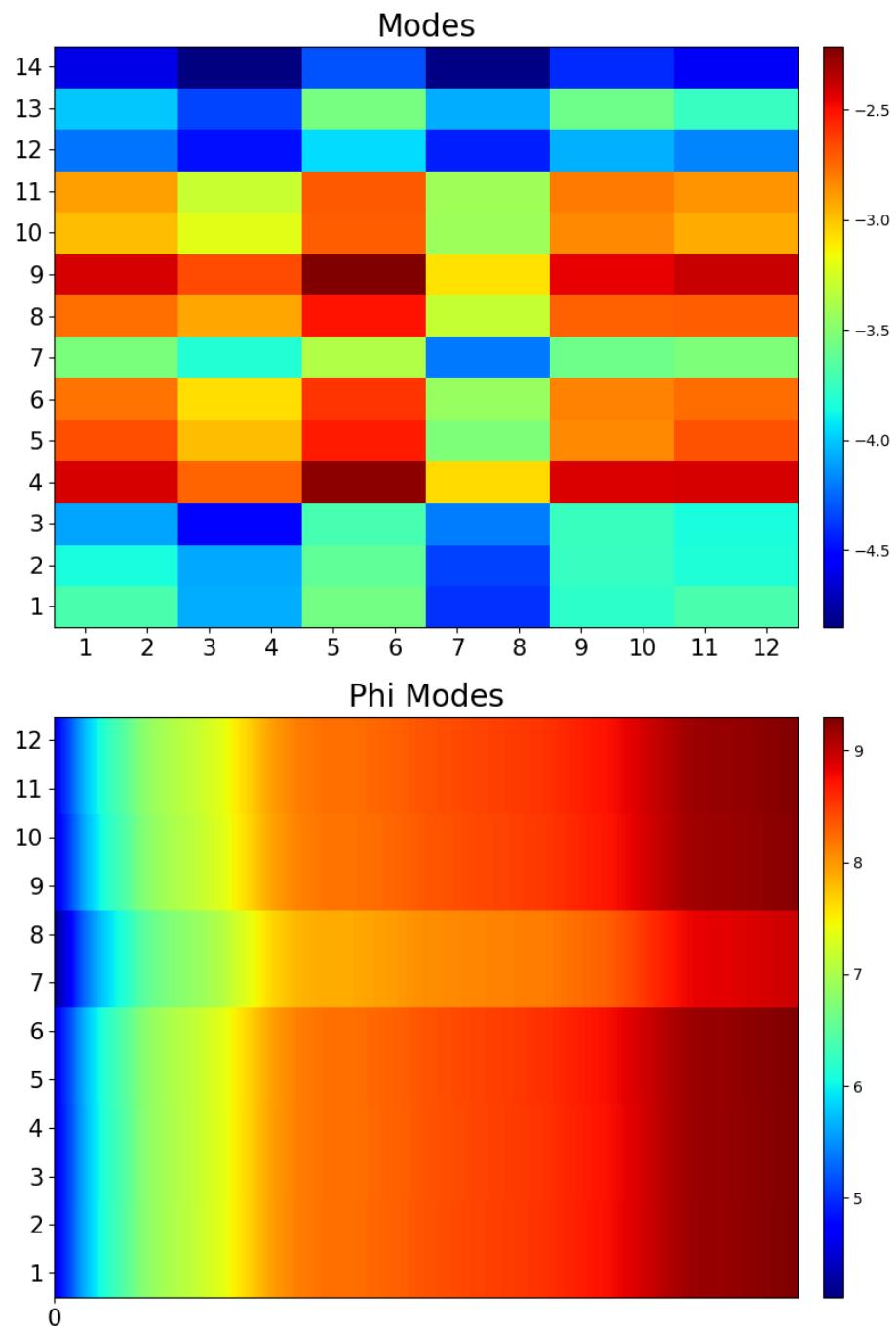


Figure 9.25. Koopman modes and phi modes by KDMD for Thij model with $\lambda = 0.8$, $p = 0.2$.

9.12 DMD: Thij Model with $\lambda = 0.8$, $p = 0.8$

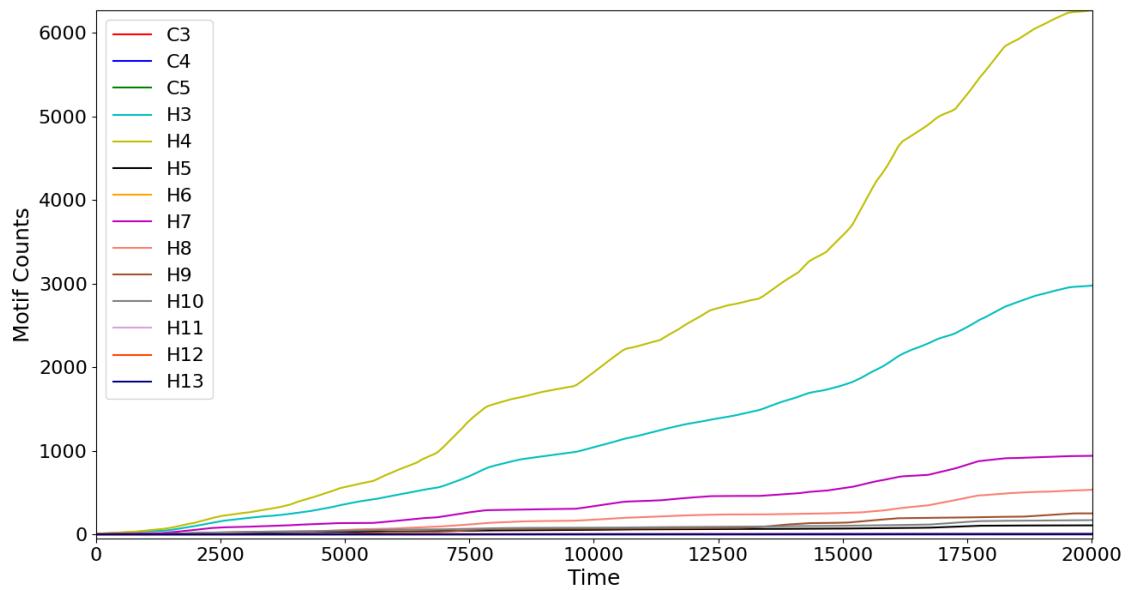


Figure 9.26. Motif counts for Thij model with $\lambda = 0.8$, $p = 0.8$.

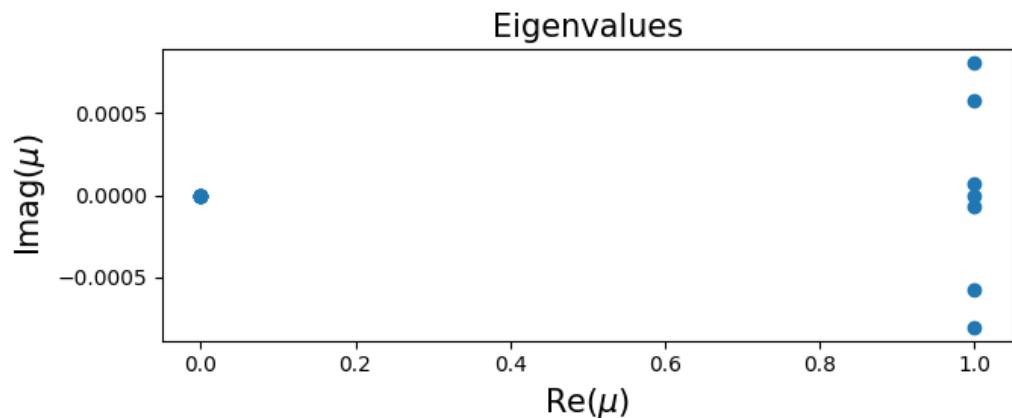


Figure 9.27. Mode eigenvalues by DMD for Thij model with $\lambda = 0.8$, $p = 0.8$.

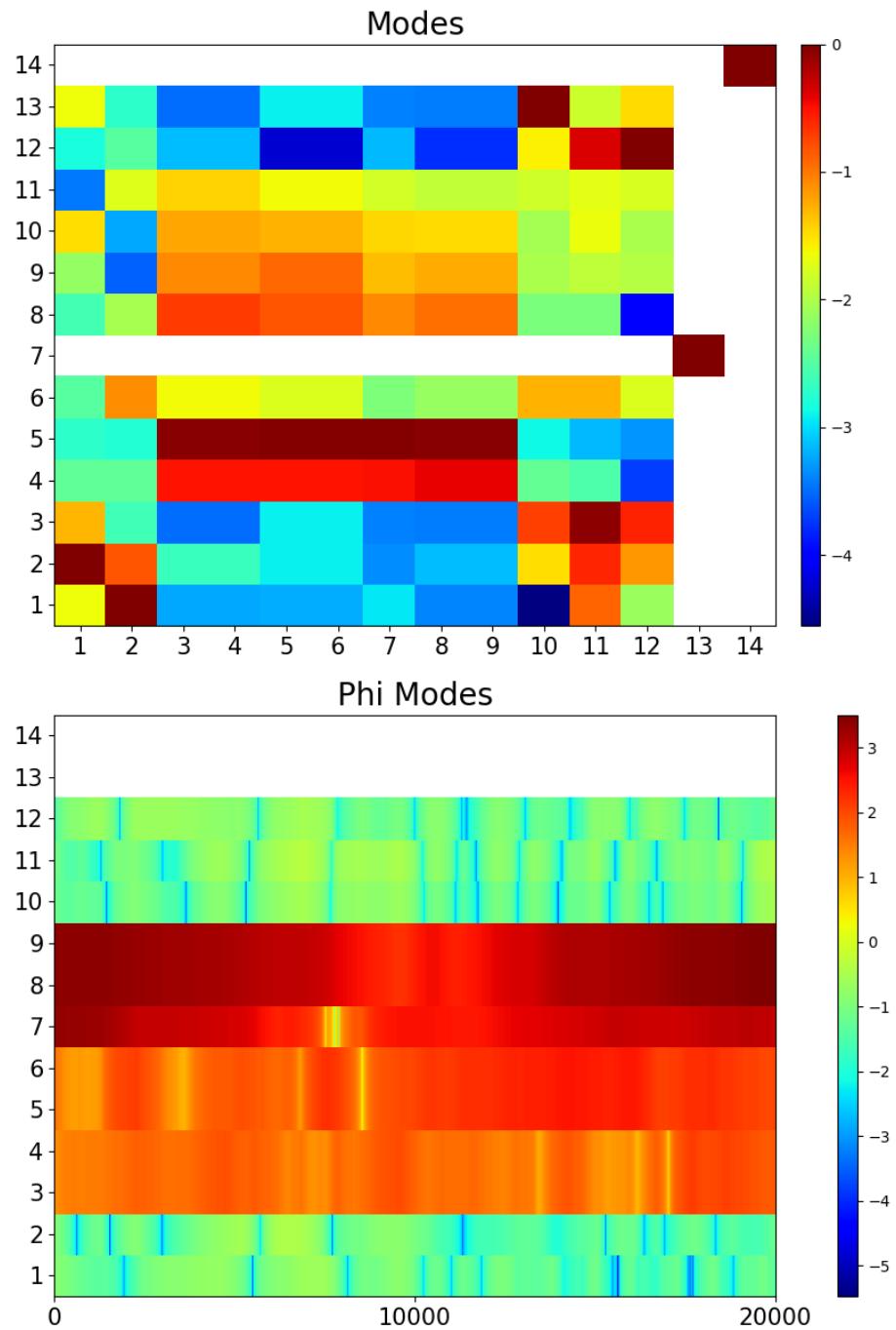


Figure 9.28. Koopman modes by DMD for Thij model with $\lambda = 0.8$, $p = 0.8$.

9.13 KDMD: Thij Model with $\lambda = 0.8, p = 0.8$

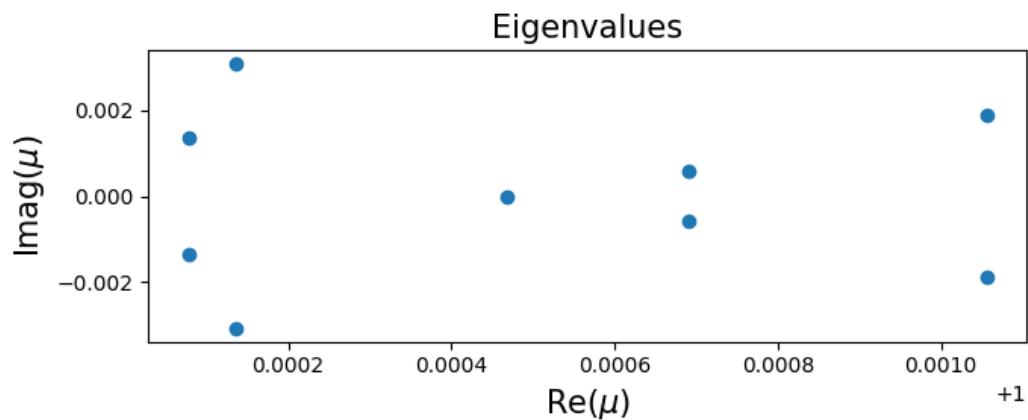


Figure 9.29. Mode eigenvalues by KDMD for Thij model with $\lambda = 0.8, p = 0.8$.

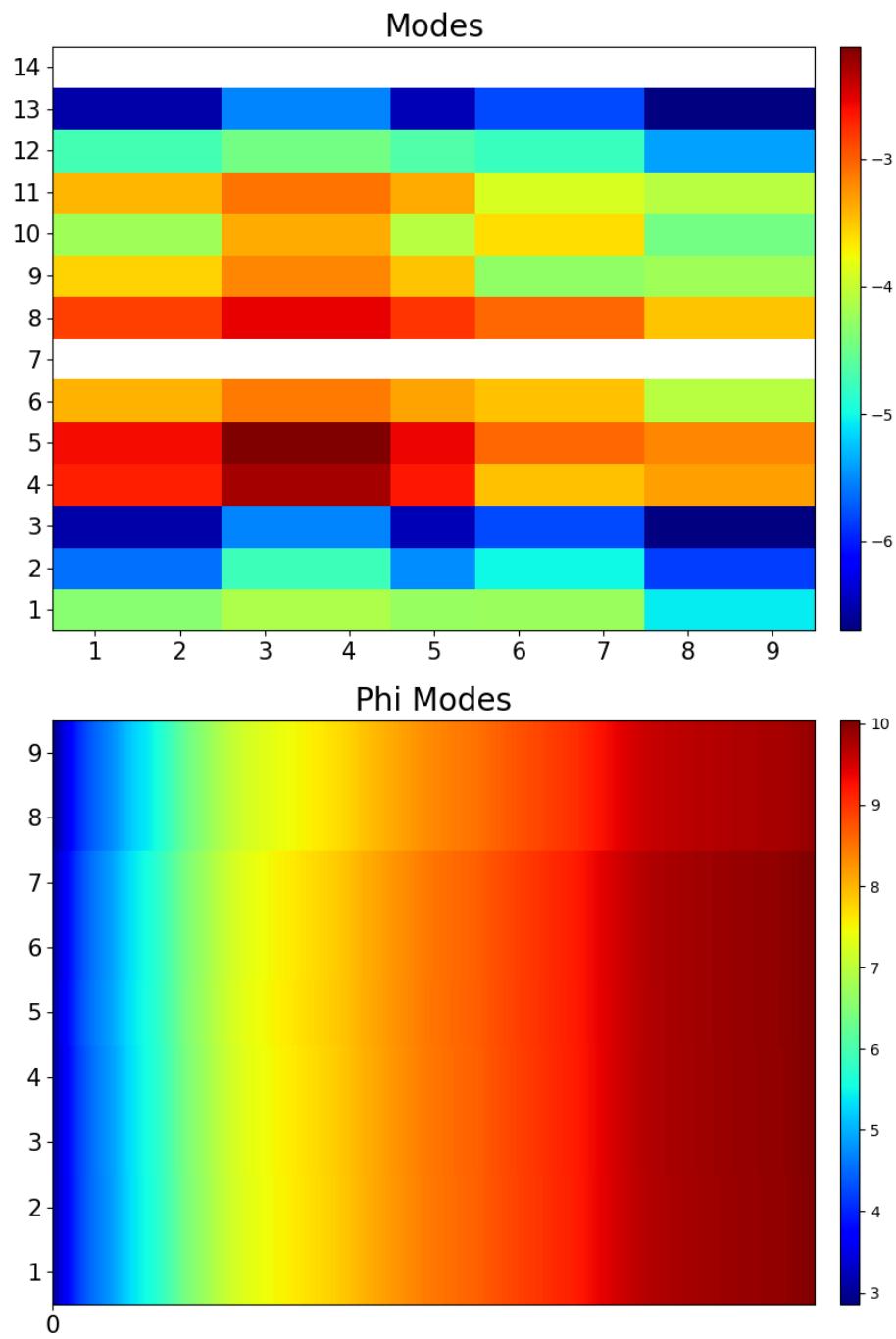


Figure 9.30. Koopman modes and phi modes by KDMD for Thij model with $\lambda = 0.8$, $p = 0.8$.

9.14 Accuracy of Methods

We want to know how DMD handles different parameter simulations of the Thij model and how it compares to KDMD. We also need to evaluate how the two accuracy criterion described in section 8.2 perform across simulations and methods. We recall that we use the one-step reconstruction error, a measure of how well the modes and eigenvalues replicate the data. We also define the mode error describing how well each DMD mode behaves like a Koopman mode. We begin with the reconstruction error of the Barabási–Albert models.

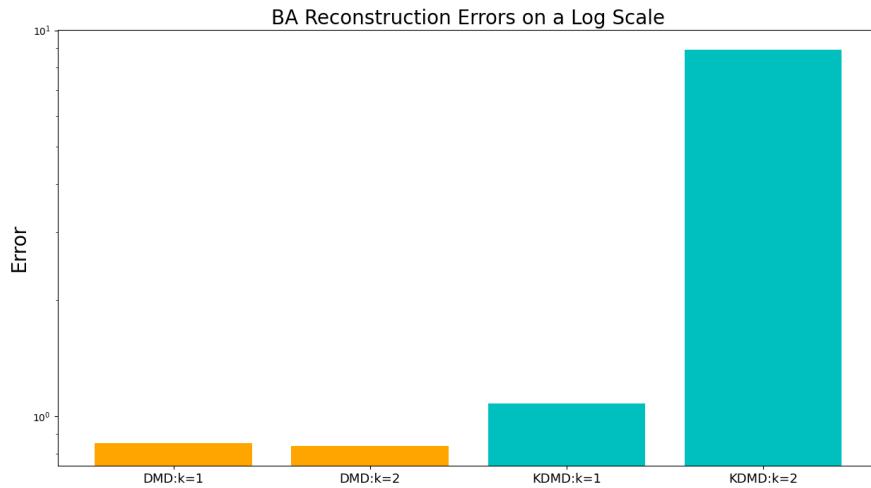


Figure 9.31. In orange have the reconstruction error of the DMD results and in cyan the KDMD results. The DMD performs much better with regard to reconstruction error, by several scales of magnitude.

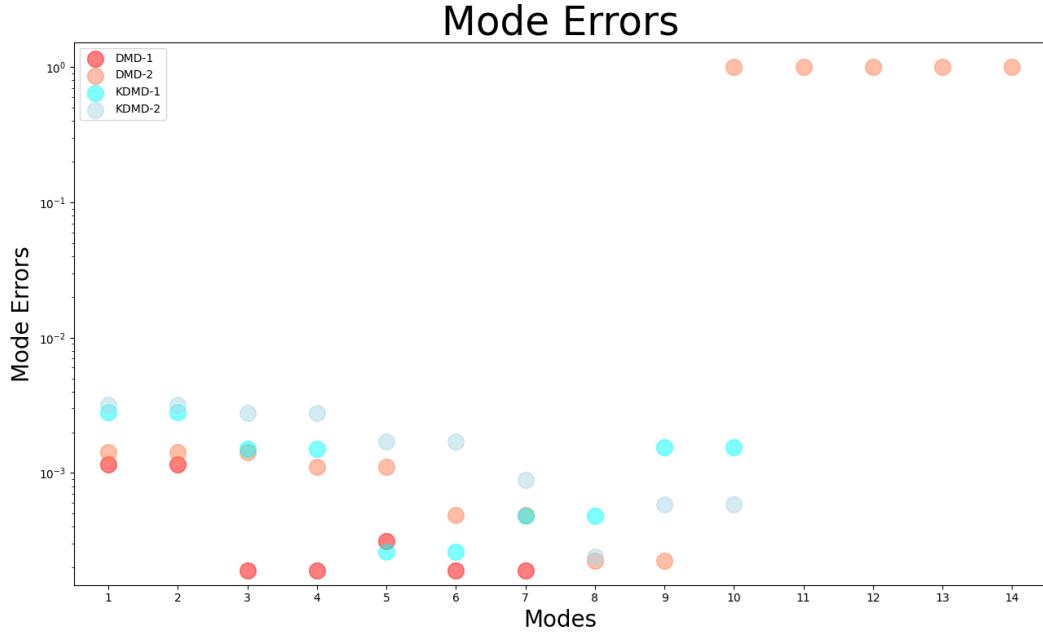


Figure 9.32. DMD also seems to produce modes with lower mode error. The scale on this figure is linear, so the differences are not as stark in figure 9.31. The KMD method for $m = 1$ does seem to perform comparable given that all the modes in KMD have dynamic behavior.

Now we consider the performance of DMD and KMD on the more complex Thij model.

Now we can evaluate the respective mode errors.

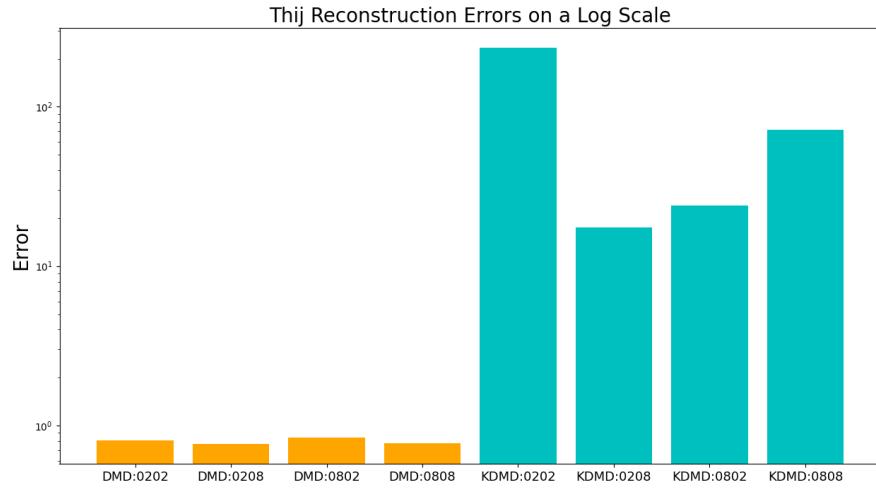


Figure 9.33. DMD and KDMD reconstruction errors across Thij simulations. DMD here again has very low reconstruction errors compared to KDMD.



Figure 9.34. DMD and KDMD mode errors across simulations. DMD does seem to produce better results, but KDMD has the advantage of being able to produce more Koopman modes. The KDMD results for $\lambda = 0.2$, $p = 0.2$ is a noticeable outlier with very low mode error.

Examining these mode errors, we can see the modes from the standard DMD algorithm are just as accurate or more accurate than the KDMD modes. DMD also performs much better in terms of one-step reconstruction error over KDMD, which is not unexpected since DMD is a least-squares fit to the data. Overall the KDMD modes look to be good Koopman modes. However, examining the phi modes above we see that in DMD we can pick out modes which effectively contribute more to the overall energy at each time step [17]. We could use to select a low rank representation of the data, reconstructing it with only a few nodes. However, reconstruction is not the focus of this thesis.

CHAPTER 10

Discussion

From the results in the previous chapter, we are able to sift out differences in dynamic behavior by applying DMD to the motif counts. DMD produces modes that have associated temporal patterns: growth, decay, and oscillation. For many of these simulations, the real part of the approximated Koopman eigenvalues clusters around one or zero. We only see zero eigenvalues given that a mode does not exhibit any temporal behavior such as in the $k = 1$ Barabási–Albert simulation or the $p = 0.8$ Thij simulations. Otherwise our observed eigenvalues tend to cluster in tight groups at the edge of the unit circle.

Examining the modes above we see block structure in the heatmaps in a pattern similar to those generated in the correlation and covariance matrices in chapter 7. If we look to figure 9.18 and 9.20 we can pick out in the former the strongest DMD modes reflect an association between $H3$, $H4$, $H7$, $H8$. In the latter simulation, KDMD produces modes of equal amplitude throughout time but we can once again see the association between the same motifs.

For certain simulations, we have a readily available a-priori explanation of global network dynamics, the formation of large induced star subgraphs. The preferential attachment mechanism effectively acts as a positive feedback loop - the additional attachment of nodes to a node u increases the likelihood of the future attachment to node u . Even without such a tale DMD and KDMD are capable of extracting out underlying structure from those simulations, for example a Thij model with $\lambda = 0.8$, $p = 0.2$ or $\lambda = 0.8$, $p = 0.8$.

This tale of induced star subgraphs on the network is most apt when there is a strong preferential attachment mechanism, but not so for $T3$ events. It's true that for any of the given simulations examined, the $H4$ motif was present if not outright central to the formation of the network's temporal development. The development of other motifs alongside the $H4$ development depends on the initialization of the graph, especially in the Barabási–Albert model which cannot add in edges between any two nodes. Even for the Barabási–Albert simulations in chapter 5 we see that $H4$ is preparing to surpass in either the $m = 2$ or $m = 1$ simulations. One might speculate, asymptotically, given a preferential attachment mechanism a network, $H4$ will come to be the dominant motif.

This is further motivated by our analysis of the correlation and covariance matrices of the given simulations. The $H7$'s and $H8$'s generally have relatively high covariances with one another and relatively high variances. In the analysis of their development, adding a node and an edge to either of the motifs generates a host of new ones either combinatorially or multiplicatively respectively.

CHAPTER 11

Conclusion

The motif counts make valuable features to characterize the local structure of the network over time. Their dynamical behavior can be studied using traditional statistical methods as we have done above using correlation and covariance statistics. However DMD also provides a way to extract the spatiotemporal coherent structures associated with the motif counts. It is a different way of understanding how attaching nodes and edges to a graph, generates those new motifs.

The effects of adding nodes and edges can be understood through graph theory. As the attachment mechanisms become more complex the analysis becomes much more difficult. The connection of disjoint graphs can generate many new motifs quite suddenly. In the simplest case however the induced star subgraph offers a good explanation of why motifs correlate the way they do and could offer a way to interpret the DMD modes in the future. The study of motifs is young and better understanding of the dynamic interactions between motifs would greatly improve the ability to interpret the modes as physical phenomena.

The models above can be made more complex by introducing other mechanisms, such as an edge deletion mechanism. That mechanism would cause the deletion of certain motifs and could potentially introduce oscillatory behavior into the motif counts. These DMD and KMD modes may offer context to the modes we see above in chapter 8.

BIBLIOGRAPHY

- [1] I. ALBERT AND R. ALBERT, *Conserved network motifs allow protein–protein interaction prediction*, Bioinformatics, 20 (2004), pp. 3346–3352.
- [2] R. ALBERT AND A.-L. BARABÁSI, *Statistical mechanics of complex networks*, Reviews of Modern Physics, 74 (2002), p. 47–97.
- [3] J. ALLEN, B. HOWLAND, M. MOBIUS, D. ROTHSCHILD, AND D. J. WATTS, *Evaluating the fake news problem at the scale of the information ecosystem*, Science Advances, 6 (2020), p. eaay3539.
- [4] N. ALON, R. YUSTER, AND U. ZWICK, *Finding and counting given length cycles*, in Algorithms — ESA '94, J. van Leeuwen, ed., Berlin, Heidelberg, 1994, Springer Berlin Heidelberg, pp. 354–364.
- [5] S. APARICIO, J. VILLAZÓN-TERRAZAS, AND G. ÁLVAREZ, *A model for scale-free networks: Application to twitter*, Entropy, 17 (2015), pp. 5848–5867.
- [6] A.-L. BARABÁSI AND R. ALBERT, *Emergence of scaling in random networks*, Science, 286 (1999), p. 509–512.
- [7] A.-L. BARABÁSI AND M. PÓSFAI, *Network science*, Cambridge University Press, Cambridge, 2016.
- [8] A. BESSI, F. PETRONI, M. DEL VICARIO, F. ZOLLO, A. ANAGNOSTOPOULOS, A. SCALA, G. CALDARELLI, AND W. QUATTROCIOCCHI, *Homophily and polarization in the age of misinformation*, The European Physical Journal Special Topics, 225 (2016).
- [9] S. BHAMIDI, J. M. STEELE, AND T. ZAMAN, *Twitter event networks and the superstar model*, The Annals of Applied Probability, 25 (2015).
- [10] G. BLANK, *The digital divide among twitter users and its implications for social research*, Social Science Computer Review, 35 (2017), pp. 679–697.
- [11] J. BOLLEN, H. MAO, AND X. ZENG, *Twitter mood predicts the stock market*, Journal of Computational Science, 2 (2011), pp. 1–8.
- [12] A. BOVET AND H. A. MAKSE, *Influence of fake news in twitter during the 2016 us presidential election*, Nature communications, 10 (2019), pp. 1–14.
- [13] R. BROWNE, *Bitcoin spikes 20% after elon musk adds #bitcoin to his twitter bio*. "<https://www.cnbc.com/2021/01/29/bitcoin-spikes-20percent-after-elon-musk-adds-bitcoin-to-his-twitter-bio.html>", March 2020. [Online; accessed 15-March-2021].
- [14] R. BROWNE, *Tweets from elon musk and other celebrities send dogecoin to a record high*. "<https://www.cnbc.com/2021/02/08/>

- [tweets-from-elon-musk-and-celebrities-send-dogecoin-to-a-record-high.html](https://www.cnbc.com/2020/05/01/tesla-ceo-elon-musk-says-stock-price-is-too-high-shares-fall.html)", February 2020. [Online; accessed 19-March-2021].
- [15] S. L. BRUNTON, M. BUDIŠIĆ, E. KAISER, AND J. N. KUTZ, *Modern koopman theory for dynamical systems*, 2021.
 - [16] J. BURSZTYNSKY, *Tesla shares tank after elon musk tweets the stock price is ‘too high’*. "<https://www.cnbc.com/2020/05/01/tesla-ceo-elon-musk-says-stock-price-is-too-high-shares-fall.html>", March 2020. [Online; accessed 15-March-2021].
 - [17] C. CURTIS, R. CARRETERO-GONZALEZ, AND M. POLIMENO, *Characterizing coherent structures in bose-einstein condensates through dynamic-mode decomposition*, Physical Review E, 99 (2019).
 - [18] E. FERRARA, *# covid-19 on twitter: Bots, conspiracies, and social media activism*, arXiv preprint arXiv:2004.09531, (2020).
 - [19] V. R. K. GARIMELLA AND I. WEBER, *A long-term analysis of polarization on twitter*, in Proceedings of the International AAAI Conference on Web and Social Media, vol. 11, 2017.
 - [20] J. H. TU, C. W. ROWLEY, D. M. LUCHTENBURG, S. L. BRUNTON, AND J. NATHAN KUTZ, *On dynamic mode decomposition: Theory and applications*, Journal of Computational Dynamics, 1 (2014), p. 391–421.
 - [21] A. JAZAYERI AND C. C. YANG, *Motif discovery algorithms in static and temporal networks: A survey*, Journal of Complex Networks, 8 (2020). cnaa031.
 - [22] J. N. KUTZ, S. L. BRUNTON, B. W. BRUNTON, AND J. L. PROCTOR, *Dynamic Mode Decomposition*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2016.
 - [23] R. MILO, S. SHEN-ORR, S. ITZKOVITZ, N. KASHTAN, D. CHKLOVSKII, AND U. ALON, *Network motifs: Simple building blocks of complex networks*, Science, 298 (2002), pp. 824–827.
 - [24] M. MOSLEH, G. PENNYCOOK, A. A. ARECHAR, AND D. G. RAND, *Cognitive reflection correlates with behavior on twitter*, Nature communications, 12 (2021), pp. 1–10.
 - [25] A. PARANJAPE, A. R. BENSON, AND J. LESKOVEC, *Motifs in temporal networks*, in Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, WSDM ’17, New York, NY, USA, 2017, Association for Computing Machinery, p. 601–610.
 - [26] F. PIERRI, C. PICCARDI, AND S. CERI, *Topology comparison of twitter diffusion networks reliably reveals disinformation news*, (2019).
 - [27] R. J. PRILL, P. A. IGLESIAS, AND A. LEVCHENKO, *Dynamic properties of network motifs contribute to biological network organization*, PLOS Biology, 3 (2005), p. null.

- [28] Y. RUAN, A. DURRESI, AND L. ALFANTOUKH, *Using twitter trust network for stock market analysis*, Knowledge-Based Systems, 145 (2018), pp. 207–218.
- [29] C. SHAO, G. L. CIAMPAGLIA, O. VAROL, K.-C. YANG, A. FLAMMINI, AND F. MENCZER, *The spread of low-credibility content by social bots*, Nature Communications, 9 (2018).
- [30] D. SHEN, A. URQUHART, AND P. WANG, *Does twitter predict bitcoin?*, Economics Letters, 174 (2019), pp. 118–122.
- [31] M. TEN THIJ, T. OUBOTER, D. WORM, N. LITVAK, H. BERG, AND S. BHULAI, *Modelling of trends in twitter using retweet graph dynamics*, 01 2015.
- [32] M. O. WILLIAMS, I. G. KEVREKIDIS, AND C. W. ROWLEY, *A data-driven approximation of the koopman operator: Extending dynamic mode decomposition*, Journal of Nonlinear Science, 25 (2015), p. 1307–1346.
- [33] M. O. WILLIAMS, C. W. ROWLEY, AND I. G. KEVREKIDIS, *A kernel-based approach to data-driven koopman spectral analysis*, 2015.
- [34] S. WOJCIK AND A. HUGHES, *Sizing up twitter users*, PEW research center, 24 (2019).
- [35] H. ZHANG, S. T. M. DAWSON, C. W. ROWLEY, E. A. DEEM, AND L. N. CATTAFFESTA, *Evaluating the accuracy of the dynamic mode decomposition*, 2017.