Modeling and Analysis of Real World Networks using Kronecker Graphs[*]

Adithya Rao
adithyar@stanford.edu

Sandeep Sripada
sss@cs.stanford.edu

Gautam Kumar Parai
gkparai@cs.stanford.edu

ABSTRACT

It has been observed that self-similarity is an emergent property of many real world networks such as WWW, e-mail and biological networks. These networks show properties such as heavy tails for the in- and out-degree distribution, heavy tails for the eigenvalues and eigenvectors, small diameters, and densification and shrinking diameters over time. Recently, Kronecker Graphs have been shown to elegantly model these networks, while being mathematically tractable [3]. In this project, we explored the property of self-similarity exhibited by real world networks like Twitter and popular real world datasets like Memetracker. The goal of the paper is to analyze fractal and self-similar properties of real world and Kronecker graphs by applying methods such as Box counting, Bifurcation ratio using Horton-Strahler index, Hurst exponent.

General Terms

Self-similar networks, fractality, scale invariance, modularity, Kronecker graphs

1. INTRODUCTION

Structures of networks which look the same on all length scales are recognized in numerous examples in nature, from snowflakes and trees to phase transitions in critical phenomena [10] [9]. Recently new forms of topological fractality have been observed in complex networks [7] like WWW, e-mail and biological networks. We start by looking at work that analyzed self-similar properties on networks like metabolic networks in organisms [6], email networks and their comparison to river networks [1], complex networks and their growth [7] [8].

In some of the early work in this area, Ravasz et al. [6] analyzed the metabolic networks of 43 distinct organisms, by calculating the average clustering coefficient for each organism. They proposed a simple heuristic model of metabolic organization, referred to as a "hierarchical" network, which showed both scale-free nature and modularity. Somewhat surprisingly, such self-similarity was also observed in human network interactions, and was studied by Guimera et al. [1]. They analyzed an email network and determined that the network self-organizes into a self-similar structure. The results on the email network clearly showed that the community structure and the branching resembled those of river networks.

The fact that this property of self-similarity was observed in various real world networks, led to further studies to understand the underlying mechanism in the formation and evolution of such networks. In [7], Song et al. analyzed several real-world networks and computed their 'fractal dimension' using box counting and cluster growing methods. The authors conjectured that the renormalized network generates a new probability distribution of links invariant under renormalization, and demonstrated its validity by showing a data collapse of all distributions for the WWW. Extending this work in [8], the authors showed that the architecture of fractal networks is mainly because of the strong repulsion (disassortativity) between hubs on all length scales, leading to a robust modular network with fractal topology. They also proposed that network growth dynamics could be modeled as the inverse of the renormalization procedure.

In [3], Leskovec et al. introduced Kronecker Graphs, a generative network model which obeyed not only all the main static network patterns that had been observed in real networks, but also temporal evolution patterns. In this paper, the authors also presented KRONFIT, a fast and scalable algorithm for fitting Kronecker graphs by using the maximum likelihood principle. The algorithm was then used to fit the stochastic Kronecker graph model to some real world graphs, such as Internet Autonomous Systems graph and Epinion trust graphs. Properties of the Kronecker model such as connectivity and diameter were rigorously analyzed in the deterministic case, and empirically shown in the general stochastic case (also in [4]). In [5], Mahdian et al. studied the basic properties of stochastic Kronecker products. Through a series of theorems, the authors showed a phase transition for the emergence of the giant component and another phase transition for connectivity. They also showed that Kronecker Graphs do not admit short decentralized routing algorithms based on local information alone, unless the path is deterministic.

[*] CS 224W: Final Project

In the next few sections we explore different metrics for analyzing the self-similar and fractal properties of real world networks as well as their corresponding Kronecker Graphs. The rest of the paper is organized as follows: section 2 describes the datasets we analyzed, section 3 looks at the properties of Kronecker Graphs with different initiator matrices, section 4 analyzes the Twitter Graph, section 6 looks at other metrics for evaluating fractal and self-similar properties such as the Box Counting method [7], Bifurcation ratio technique [1], and Hurst exponent for a time series of data flowing in the Memetracker network. Finally section **??** describes the conclusions we draw from the analysis.

2. DATA DESCRIPTION

Twitter: a microblogging service, commands more than 190 million users as of June 2010 and is growing fast. In this project we use this dataset and model the Twitter graph as a Kronecker Graph and compare the properties of the synthetic and the real graphs.

Others: We also use the graphs from the SNAP library such as the p2p-Gnutella, Wiki-Vote, Epinions, Memetracker, AS Skitter and Email-Enron to analyze other properties of real world networks.

3. PROPERTIES OF KRONECKER GRAPHS

In [3], it is shown that Kronecker Graph is constructed by repeatedly taking the Kronecker product of an initiator matrix. The Kronecker graph model is thus based on a recursive construction of self similar graphs. The major advantage of Kronecker graphs over other network models is that it is possible to prove analytical results about many properties related to the real-world network unlike previous models which target specific properties. The entries in the initiator matrix can take values between 0 and 1, to generate a stochastic model rather than a deterministic one. In this case, the stochastic adjacency matrix encodes the probability of the particular edge appearing in the graph. The diagonal elements denote the probability with which edges within sets are formed and the off-diagonal elements denote the probability with which edges across sets are formed. A natural interpretations of the generative process is that networks are hierarchically organized into communities (clusters), which grow recursively, creating miniature copies of themselves.

In [3], the authors address the issue of automatically estimating the Kronecker initiator graph parameters. The approach to the problem of estimating Stochastic Kronecker initiator matrix is by dening the likelihood over the individual entries of the graph adjacency matrix. It was also shown that the KRONFIT algorithm efficiently performs this fitting in linear time, and the corresponding results for AS-Routeviews and Epinions were presented.

Here, we create Kronecker Graph models using various values for initial parameters in the initiator matrix, and observe the properties of the generated graphs. We evaluated the graphs starting with initiator matrices as:

- Fix the diagonal elements of the initiator matrix, (0.9 and 0.1) and vary the off-diagonal entries.
- Fix the off-diagonal values to (0.6, 0.7) and changed a single value of the diagonal of the initiator matrix.
- Use different off-diagonal entries instead of identical ones, and observed the plots.

Table 1: Values of Initiator Matrices

SNo	Matrix
1	[0.9 0.1; 0.1 0.1]
2	[0.9 0.2; 0.2 0.1]
3	[0.9 0.3; 0.3 0.1]
4	[0.9 0.4; 0.4 0.1]
5	[0.9 0.5; 0.5 0.1]
6	[0.9 0.6; 0.6 0.1]
7	[0.9 0.7; 0.7 0.1]

SNo	Matrix
1	[0.9 0.6; 0.6 0.1]
2	[0.9 0.6; 0.6 0.2]
3	[0.9 0.6; 0.6 0.3]
4	[0.9 0.7; 0.7 0.1]
5	[0.8 0.7; 0.7 0.4]

SNo	Matrix
1	[0.8 0.3; 0.4 0.4]
2	[0.8 0.4; 0.3 0.4]
3	[0.8 0.5; 0.4 0.4]

Table 1 shows the different values of the initiator matrices used. Figures 1, 2 and 3 shows a sample of plots that we generated with the various matrices. More plots can be found here: Additional Plots.

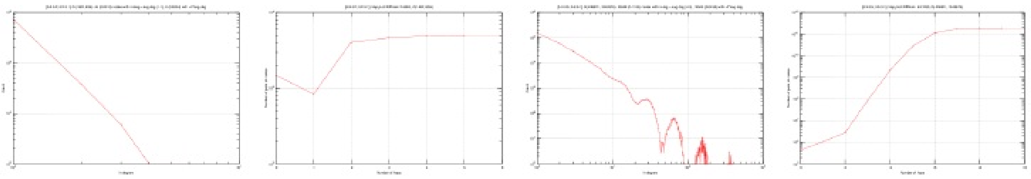
Observations relating to Table 1: From the plots, we observe that as we increase the off-diagonal entries of the initiator matrix, the neighborhoods of nodes expand faster. This is because as the off- diagonal entries increase, the number of edges between communities increase thereby increasing the network diameter and degree distribution. As we fix the off-diagonal entries and vary only one of the diagonal entries, we observe that [0.9 0.6; 0.6 0.2] seems to behave differently compared to the others. The hop plot and the degree distribution are smoother, hop plots has a higher number of hops. Also [0.8 0.7; 0.7 0.4], does not seem to behave as expected. This is also validated by the results in Table 4 of [3] where the estimated parameters are closer to [0.9 0.6; 0.6 0.2]. This suggests that there are certain values of initiator matrices that model real life networks better than others.

4. PROPERTIES OF TWITTER GRAPH

The dataset was obtained from [2] who crawled the entire Twitter site and obtained 41.7 million user profiles and 1.47 billion social relations. They also performed a quantitative study on the entire Twittersphere and information diffusion on it. Here we use this same dataset and model the Twitter graph as a Kronecker Graph. Fig 1 shows a snapshot of the Twitter graph and its corresponding Kronecker Graph.

Table 2: Kronfit parameters

Nodes	Edges	MLE Params	LL
262144	5,719,211	[0.9999 0.5453;0.5871 0.2174]	-4.29E^7
300000	6,601,661	[0.9999 0.5518;0.5852 0.2526]	-5.13E^7
524288	13,156,554	[0.9999 0.542;0.5832 0.2531]	-1.15E^8
600000	15,306,431	[0.9999 0.5542;0.5785 0.2534]	-1.4E^8
900000	25,612,280	[0.9999 0.534;0.5455 0.2732]	-2.4E^8



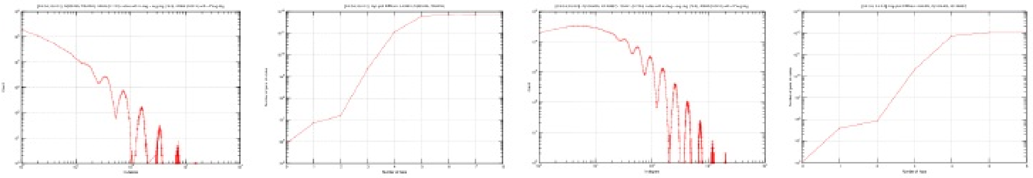
(a) Degree Distribution (in-degree)

(b) Hop plot

(c) Degree Distribution (in-degree)

(d) Hop plot

Figure 1: Plots for fixed diagonal elements of 0.9, 0.1



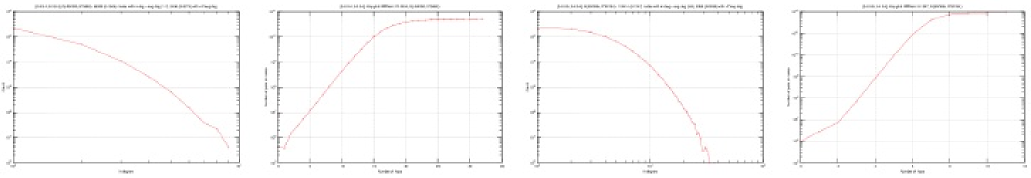
(a) Degree Distribution (in-degree)

(b) Hop plot

(c) Degree Distribution (in-degree)

(d) Hop plot

Figure 2: Plots for fixed off-diagonal elements of 0.6, 0.7



(a) Degree Distribution (in-degree)

(b) Hop plot

(c) Degree Distribution (in-degree)

(d) Hop plot

Figure 3: Plots for different diagonal elements

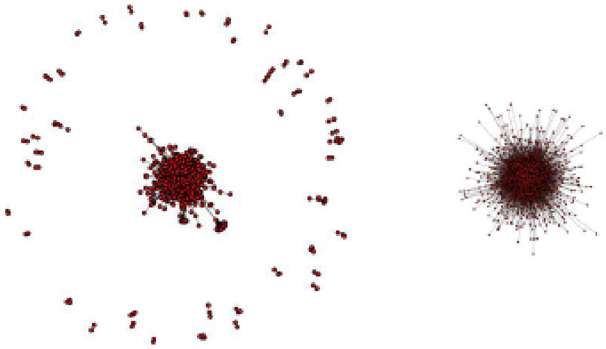


Figure 1: Snapshot of Twitter graph

Since the Twitter graph had a large number of nodes, we reduced the graph size by selecting nodes using the time of creation as the selection parameter. For example, we considered the snapshot of twitter graph at the time when the first 100,000 nodes were present in the graph, and estimated the KRONFIT parameters for this smaller graph (to reduce computational costs). We performed this for different values of nodes, as shown in Table 2. Gradient descent was done with 100 iterations.

We observed that the KronFit parameters remain almost constant across different times. If the Twitter graph were not self similar, then we would get different parameters for KronFit, as the graph changed over time. The near-constant estimated KronFit parameters strongly suggest that the Twitter graph shows the self-similar property, and may therefore be modeled as a Kronecker Graph.

Also, comparing with the results in Table 4 of [3], we see that the estimated parameters from the Twitter graph seem to closely resemble BLOG-NAT05-6M, BLOG-NAT06ALL and somewhat similar to the parameters of EPINIONS, FLICKR.

Using the estimated parameters above, we generate synthetic Kronecker graphs with the same number of nodes, and compare the properties of the real and synthetic graphs. More specifically, We used two sizes of the real Twitter graphs, having 2^{18} and 2^{19} nodes respectively. We ran Kronfit on these reduced graphs and estimated their initiator matrix parameters. We then generated synthetic Kronecker Graphs of the same size, and compared the properties of the real and synthetic graphs.

Our evaluation considers the following static properties:

- Network structure:
  - Degree distribution
  - Small world property
- Hop-plot: Such a plot (number of reachable pairs $g(h)$ within h hops vs hops h) gives us a sense of how quickly nodes neighborhoods expand with the number of hops.
- Scree plot: Plot of the eigenvalues (or singular values) of the graph adjacency matrix, versus their rank, using the logarithmic scale.
- Network Value: Plot of the sorted principal eigenvector components versus rank.

From the plots in Figure 2, we find that in both cases, the Kronecker graph mimics many of the real world graph properties. We observed some differences in the clustering coefficient and the strongly connected components.

5. SAMPLING AND PROPERTIES ACROSS SCALES

In this task we observed various properties across different scales for different networks.

- Robustness
- Information cascades

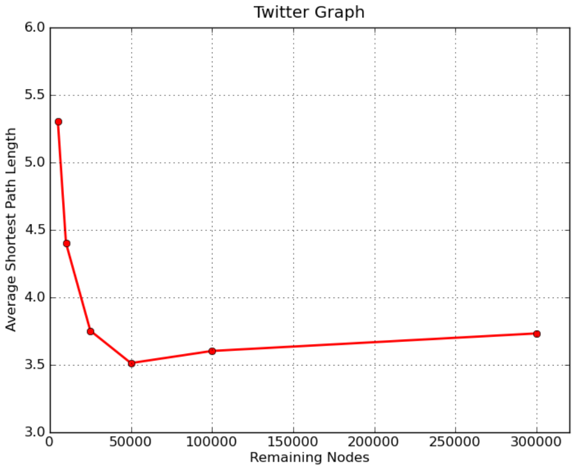
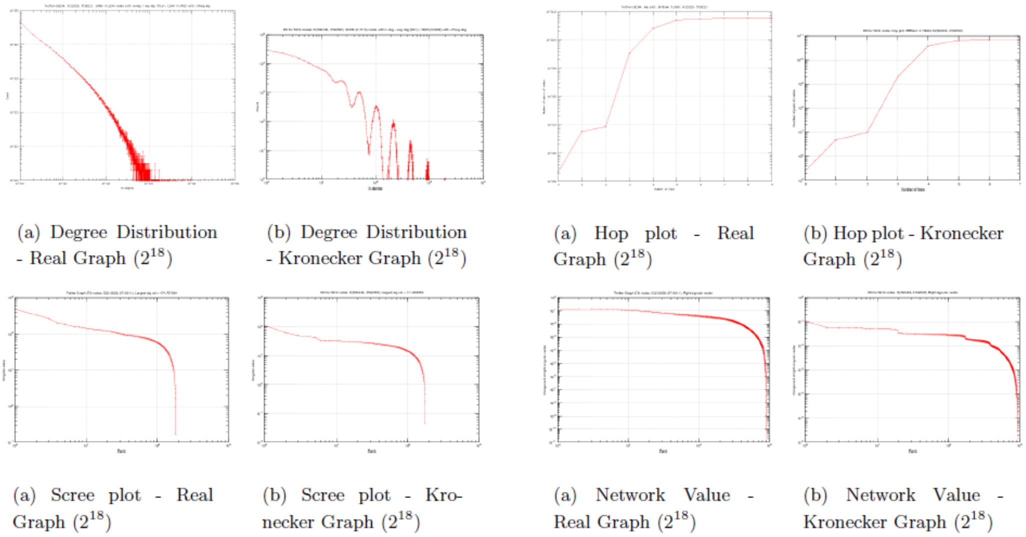


Figure 3: Robustness of Twitter graph

Robustness: We wanted to observe how vulnerable the network is by removing nodes in progressively larger subsets of the graph, and comparing the average path lengths. We expect that the graphs would have similar behavior across all scales. We ran robustness experiments on the Twitter graph, Autonomous systems (Skitter) network, Google web graph. Figure **??** shows the robustness plot for the Twitter graph.

Since, the graphs were huge, we randomly sampled node pairs in the graphs and computed the average shortest path lengths. We computed these for slices upto 0.3 million node sized graphs. We observed that the Twitter graph was remarkably robust with respect to random node deletion, with average shortest paths not changing very much up until only 50000 nodes remain in the network.

Information cascades: We modeled information cascades in the Twitter graph and the corresponding Kronecker Graph. For slices of the graphs ranging from 2^{10} to 2^{16}, all the nodes are initially assigned a state A. An initial set of starter nodes is then assigned the new state B. The payoffs for the edges



(a) Degree Distribution - Real Graph (2^{18})

(b) Degree Distribution - Kronecker Graph (2^{18})

(a) Hop plot - Real Graph (2^{18})

(b) Hop plot - Kronecker Graph (2^{18})

(a) Scree plot - Real Graph (2^{18})

(b) Scree plot - Kronecker Graph (2^{18})

(a) Network Value - Real Graph (2^{18})

(b) Network Value - Kronecker Graph (2^{18})

Figure 2: Comparison of Twitter & Kronecker

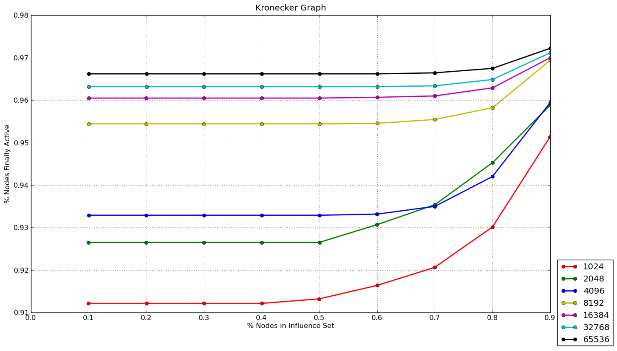


Figure 4: Information cascades at various sizes of Kronecker graphs

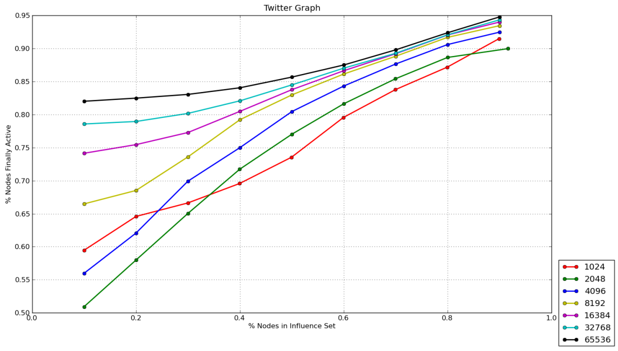


Figure 5: Information cascades at various sizes of Twitter graph

are 2 for $A-A$, 3 for $B-B$ and 0 for $A-B$. Thus for a particular node, the payoff for converting to the new state B, is the sum of payoffs of incoming edges from its neighbors in state B. Similarly, the payoff to remain in state A is the sum of payoffs of the incoming edges from its neighbors in state A. If the payoff it gets by converting to B is greater than the payoff it gets by remaining in A, the node assumes the new state B. Thus in each iteration, we count the new number of nodes converted to state B, and stop the iterations when no new nodes are converted to the new state.

We perform this information cascade for different sizes of initial starter sets for both the Twitter graph as well as the corresponding Kronecker graphs. Figures 4, 5 show the plots for both, with the size of the initial set (in terms of percentage of total nodes) on the X-axis and the final percentage of nodes converted on the Y-axis. From the plots, we see that as the size of the graph increases, the curves for the information cascades of the Twitter graph tend to those of the corresponding Kronecker graph. This again reinforces the notion that the Twitter graph can be approximated by a Kronecker graph.

6. METRICS

6.1 Box Counting Method

The box covering method is central to the understanding of the scale-invariant properties of networks. To calculate this dimension for a fractal set S, this fractal is assumed to be lying on an evenly-spaced grid, and count how many boxes are required to cover the set. The box-counting dimension is then calculated by seeing how this number changes as we make the grid finer.

The authors in [7] investigated the concept of renormalization as a mechanism for the growth of fractal and non-fractal modular networks. In this method, for each size l_B, boxes are chosen randomly until the network is covered, and each box consists of nodes separated by a distance $l < l_B$. Then each box is replaced by a node, in a process known as renormalization. The renormalized nodes are connected if there

Table 3: Box counting plot slopes

Graph	Slope
Email	1.63
WikiVote	1.64
Twitter	1.38
Kronecker	1.58

is at least one link between the unrenormalized boxes. This procedure is repeated until the network collapses to one node. In each iteration, the network is covered with N_B boxes of linear size l_B. The fractal dimension or box dimension d_B is then given by: $N_B \sim l_B^{-d_B}$. This renormalization was applied on the WWW network in [3], and it was shown that the degree distribution remains invariant under renormalization, suggesting the self similarity of the network.

It is clear that the adjacency matrix of a graph captures the node and edge relationships of a graph. Thus an adjacency matrix offers a ready and complete representation of a graph as a set S in an evenly spaced grid. In our approach, instead of applying box-counting and renormalization on the nodes as given in [7], we instead apply the method to the adjacency matrix of the graph. The graph can thus be thought as a 2D black and white image whose fractal dimension needs to be verified.

We ran the box counting method on the adjacency matrix of different kinds of networks such as peer-to-peer networks (p2p-Gnutella), social networks (Twitter) and communication networks (Email-Enron). We also performed the procedure on Kronecker Graphs of different parameters.

Fig 6shows the log-log plots obtained. Table 3 shows the slopes for the respective graphs.

6.2 Horton-Strahler (HS) Index and Bifurcation Ratio

The HS-index for self-similarity is inspired from [1] where the authors have used it to determine the self-similar structure of river networks. We first detect the communities in the graph using the Girvan-Newman community detection algorithm and construct the corresponding tree structure. We then apply the recursive algorithm proposed in [1] for computing the HS index. Once the HS index has been computed, we computed the Bifurcation ratios as the ratio of the number of edges at level i to that at level $i+1$.

$$i = \begin{cases} i_1 + 1 & \text{if } i_1 = i_2 \\ max(i_1, i_2) & \text{otherwise} \end{cases}$$

$$B_i = \frac{N_i}{N_{i+1}}$$

If the Bifurcation ratios remain approximately the same at all levels i.e. $B_i \sim B$ then the structure is topologically self-similar since, the overall tree consists of B subtrees each of which in turn consists of B subtrees. This recursive definition naturally gives rise to topologically self-similar structure. For river networks, the authors in [1] found $3 < B < 5$. Since, the Girvan-Newman algorithm is expensive, we ran the algorithm on slices of the original network. From our experiments, we found that the Twitter graph as well as the Synthetic Kronecker graphs exhibit topological self-similarity.

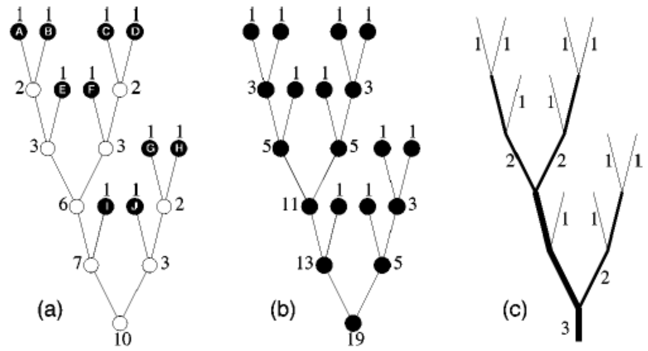


Figure 7: Calculation of HS-Index (from [1])

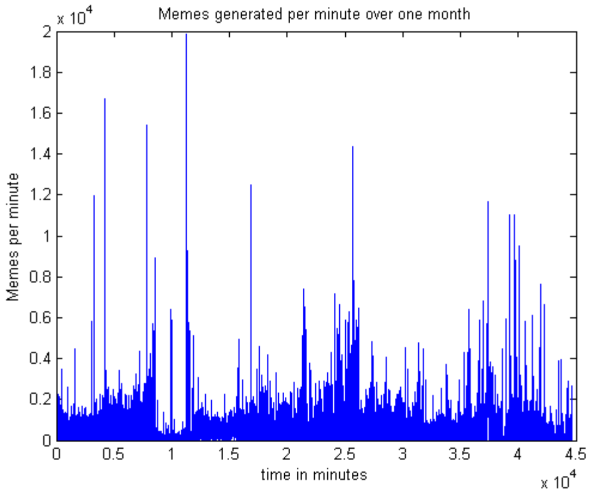


Figure 8: Time series of Memetracker data (Aug 2008)

6.3 Hurst Exponent

The Hurst exponent H is used as a measure of autocorrelation of a given time series. Unlike the previous two metrics, the Hurst exponent measures a form of self-similarity called statistical self-similarity and not topological self-similarity.

A value of $0 < H < 0.5$ indicates negative autocorrelation, whereas a value of $0.5 < H < 1$ indicates positive autocorrelation. A Hurst exponent in the range $0.5 < H < 1$, indicates that the given time series data shows extended temporal correlations, or long-range dependence (LRD). When viewed within some range of time scales, the data appears to be fractal-like or self-similar. In other words, a segment of the data measured at some time scale looks or behaves (statistically) just like an appropriately scaled version of the data measured over a different time scale.

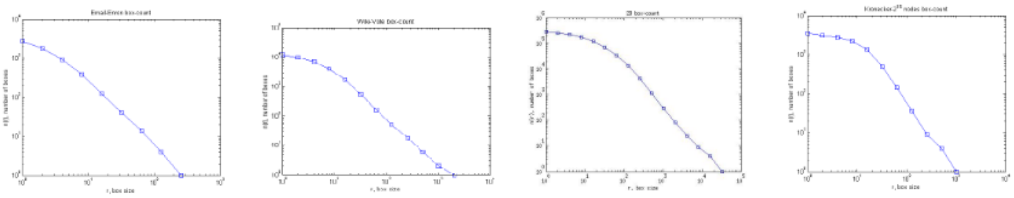


Figure 6: Box counting plots for different graphs

It was previously shown in [11] that network traffic over ethernet computer networks is statistically self-similar, with a Hurst exponent in the range $[0.85, 0.90]$. Here we examine the flow of information in the blogspace using MemeTracker data from one month. We count the number of memes generated in each minute in the month of August 2008. Fig 8 shows the time series obtained. The Hurst exponent for this data was calculated to be **0.68**, which indeed indicates that the data is statistically self-similar.

7. CONCLUSIONS & FUTURE WORK

After the experiments using various initiator matrices, we saw that some of the values better model a real-world network than others. By looking at various network properties it may be ascertained whether a Kronecker graph models a network realistically. Also, the KronFit estimates across various scales of the Twitter graph are very close to each other indicating self-similarity. By comparing the network properties of the real Twitter graph with the corresponding Kronecker graphs we observed that Kronecker graphs closely model Twitter graphs. This work could be used in examining network evolution for the Twitter graph using a mathematically tractable Kronecker graphs. We also ran experiments on robustness and information cascades for Twitter and observed consistent self-similar network properties.

We analyzed fractal properties of various real-world networks using Box-counting, Bifurcation ratio calculated from HS-Index, and Hurst exponent. We observed that the fractal dimensions (d_B) calculated using Box counting for several real-world networks such as p2p-Gnutella, Enron and Twitter lie in the range $\sim [1.3, 1.8]$. Kronecker graphs also showed similar values for fractal dimension. In the case of Bifurcation ratio calculation using HS-Index, efficient computation of communities would allow us to calculate the Bifurcation ratio values for larger networks. Also the range in which the Bifurcation ratio values fall are consistent with other self-similar networks such as River networks. Apart from these topological self-similarity metrics, the Hurst exponent calculated for Memetracker data shows that information flowing through the network shows statistical self-similarity over time. An extension to this work would be to apply these metrics to various other networks and understand their behavior in more detail.

8. REFERENCES

[1] R. Guimera, L. Danon, A. Diaz-Guilera, F. Giralt, and A. A. Self-similar community structure in a network of human interactions. *Physical Review E 68, 065103(R)*, 2003.

[2] H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media? *Proc. WWW*, 2010.

[3] J. Leskovec, D. Chakrabarti, J. Kleinberg, C. Faloutsos, and G. Z. Kronecker graphs: An approach to modeling networks. *Journal of Machine Learning Research (JMLR)*, pages 985–1042, 11 Feb, 2010.

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

[5] M. Mahdian and Y. Xu. Stochastic kronecker graphs. *WAW 2007, LNCS 4863*, pages 179–186, 2007.

[6] E. Ravasz, A. L. Somera, D. A. Mongru, Z. N. Oltvai, and A. Barabasi. Hierarchical organization of modularity in metabolic networks. *Science 297*, pages 1551–1555, 2002.

[7] C. Song, S. Havlin, and H. Makse. Self-similarity of complex networks. *Nature 433*, pages 392–395, 2005.

[8] C. Song, S. Havlin, and H. Makse. Origins of fractality in the growth of complex networks. *Nature*, April 2006.

[9] H. E. Stanley. Introduction to phase transitions and critical phenomena. *Oxford Univ. Press, Oxford,*, 1971.

[10] T. Vicsek. Fractal growth phenomena. *2nd edn. Part IV (World Scientific, Singapore)*, 1992.

[11] W. Willinger, M. Taqqu, S. S., and D. Wilson. Self-similarity through high-variability: Statistical analysis of ethernet lan traffic at the source level. *IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 5, NO. 1*, FEBRUARY 1997.