Self-Similarity in Social Network Dynamics

QINGYUN LIU and XIAOHAN ZHAO, University of California, Santa Barbara
WALTER WILLINGER, Niksun
XIAO WANG, Renren Inc.
BEN Y. ZHAO and HAITAO ZHENG, University of California, Santa Barbara

Analyzing and modeling social network dynamics are key to accurately predicting resource needs and system behavior in online social networks. The presence of statistical scaling properties, that is, self-similarity, is critical for determining how to model network dynamics. In this work, we study the role that self-similarity scaling plays in a social network edge creation (that is, links created between users) process, through analysis of two detailed, time-stamped traces, a 199 million edge trace over 2 years in the Renren social network, and 876K interactions in a 4-year trace of Facebook. Using wavelet-based analysis, we find that the edge creation process in both networks is consistent with self-similarity scaling, once we account for periodic user activity that makes edge creation process non-stationary. Using these findings, we build a complete model of social network dynamics that combines temporal and spatial components. Specifically, the temporal behavior of our model reflects self-similar scaling properties, and accounts for certain deterministic non-stationary features. The spatial side accounts for observed long-term graph properties, such as graph distance shrinkage and local declustering. We validate our model against network dynamics in Renren and Facebook datasets, and show that it succeeds in producing desired properties in both temporal patterns and graph structural features.

CCS Concepts: • **Networks** → **Network simulations**; **Network measurement**; **Network dynamics;**

Additional Key Words and Phrases: Self-similarity, social network dynamics, social network measurement, social network modeling

ACM Reference Format:
Qingyun Liu, Xiaohan Zhao, Walter Willinger, Xiao Wang, Ben Y. Zhao, and Haitao Zheng. 2016. Self-similarity in social network dynamics. ACM Trans. Model. Perform. Eval. Comput. Syst. 2, 1, Article 5 (October 2016), 26 pages.
DOI: http://dx.doi.org/10.1145/2994142

5

1. INTRODUCTION

Studying the dynamics of social networks, that is, network evolution including detailed timings of when nodes (the abstract notion of users) arrive and edges (relationship built between a pair of users, for example, becoming friends with each other) are created, is important for many social network applications, including system design, resource allocation, anomaly detection, and demand forecasting. However, despite

This project was supported by NSF grants CNS-1527939 and IIS-1321083. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of any funding agencies.
Authors' addresses: Q. Liu, X. Zhao, B. Y. Zhao, and H. Zheng, Computer Science Department, University of California, Santa Barbara, CA 93106, USA; emails: {qingyun_liu, xiaohanzhao, ravenben, htzheng}@cs.ucsb.edu; W. Willinger, Niksun, 100 Nassau Park Blvd, Princeton, NJ 08540, USA; email: wwillinger@niksun.com; X. Wang, Renren Inc., 1/F Great Creativity Information Industry Garden, North Building Suite, 18 Jiuxianqiao Middle Road, Chaoyang District, Beijing, 100000, China; email: xiao.wang@renren-inc.com.

DOI: http://dx.doi.org/10.1145/2994142

recent progress of analyzing and modeling online social networks (OSNs) [Schneider et al. 2009; Wilson et al. 2009; Jiang et al. 2010; Sarkar et al. 2012; De Meo et al. 2014; Cho et al. 2016; Bi and Cho 2016], their network dynamics are still poorly understood. Current methods often study them via *static* snapshots, which capture network dynamics only at discrete points in time and lack time information about events that occur between snapshots. Similarly, current models of network dynamics are typically randomized, generative graph models that produce sequences of events leading to an observed network structure [Karger and Ruhl 2002; Leskovec et al. 2005; Akoglu et al. 2008; Akoglu and Faloutsos 2009; Navlakha et al. 2015; Xia and Hu 2015]. Focusing primarily on producing a graph with some desired structural properties, they do not model or match the sequence of dynamic events that lead to that structure.

Our work seeks to address this need by studying detailed dynamics in "time-stamped" traces of network growth. While most/all existing work analyze and model dynamics using *logical clocks*, we examine the relationship between network dynamics and real *physical clock* time. Specifically, the use of physical time allows us to tackle two significant challenges in the modeling of network dynamics. First, physical time allows us to determine if social network dynamics exhibit *self-similarity*, an invariance of behavior at different time scales. Self-similarity is a fundamental statistical property, that, if discovered, defines hard limits on how such dynamics can be modeled using traditional means, for example, Poisson. Its detection in contexts such as network traffic and web traffic has led to significant shifts in how such datasets were analyzed and modeled.

Second, analysis of a physical time trace allows us to build a model of OSN dynamics that captures not only structural properties of the network but also the sequence of dynamic events leading to that structure. This type of dynamic graph model would address several practical OSN problems. First, the research community has repeatedly expressed a need for real dynamic graph traces. Using a real trace for calibration, our model can generate "realistic" dynamic graphs with a complete list of time-stamped network events. Next, our model can be used to perform "interpolation," that is, construct complete dynamic graph traces that approximate the continuous network evolution between successive static snapshots of OSNs. Finally, our model can be used to detect abnormal events (attacks or changes in user behavior) in real networks, that is, events that disrupt expected network dynamics.

In this work, we perform an empirical study of network dynamics by examining network events over multiple years. For this, our work relies on two detailed, time-stamped traces of social networks, the *Renren* dataset [Zhao et al. 2012] (complete, time-stamped trace of 199 million social links over 2 years) and the *Facebook wall post* dataset [Kunegis 2013] (876K wall posts between users over 4 years in a Facebook regional network). To the best of our knowledge, these are the only datasets available today with sufficient granularity and event frequency to provide accurate analysis on network dynamics and self-similarity.

Self-Similarity-Based Network Analysis. Self-similarity refers to the invariance behavior of a time series under rescalings, that is, the relative variance or volatility of traffic traces stays similar across different time scales.[1] Successful detection of self-similar properties is a very meaningful result (for network modeling), because it defines fundamental limits on how such datasets can be modeled using traditional means. Due to its very different statistical properties, for example, significantly higher burstiness, *self-similar* traffic cannot be easily captured or modeled by popular traffic models. In recent years, self-similarity has been found and has led to changes in data modeling in a

[1]Self-similarity can be used to describe scale invariance of certain properties of an object in space and/or time. In this article, we adopt the *temporal* meaning, that is, self-similarity along the time dimension.

variety of contexts, including local network traffic, wide-area network traffic, file system accesses, disk-level I/O (input/output), messaging and email communications, and web traffic requests [Leland et al. 1994; Paxson and Floyd 1995; Crovella and Bestavros 1997; Gribble et al. 1998; Riska and Riedel 2006; Eisler et al. 2008; Rybski et al. 2009; Deng et al. 2012]. In each case, the discovery of self-similar scaling properties led to a noteworthy shift in how such datasets were analyzed and modeled.

It is challenging to detect and quantify self-similar scaling properties in real network traces in a statistically rigorous manner. This is partially due to the likely presence of patterns (e.g., deterministic trends and diurnal or weekly cycles) that introduce non-stationarity. The edge creation process may be consistent with self-similar scaling over time scales ranging from seconds to hours. But patterns like diurnal or weekly user cycles likely dominate over larger time scales like days and weeks and need to be accounted for before any self-similarity analysis. Intuitively, we seek to not only detect self-similar scaling properties in edge creation process but also determine time scales where self-similarity is visible and can be quantified. Thus, we use a range of techniques including R/S (rescaled range) analysis, the variance fitting method, and a wavelet-based method. And our analysis focuses on edge creation, mainly because an exploratory analysis of the Renren data revealed no particular structure underlying the observed node creation events.

A Model of Social Network Edge Dynamics. We incorporate the findings from our self-similarity analysis into a complete evolutionary network model, including a *temporal* component that determines "when" new edge creations occur in time and a *spatial* component that specifies "where" these new edges form. Together, this model produces a sequence of time-stamped events that uniquely define the formation and evolution of a social network or graph in time and space. By tuning a small number of parameters, our model can be calibrated to "fit" traces of measured graph dynamics exhibiting self-similar properties. We validate the model by comparing the model-generated edge creations to that of the real data (Renren and Facebook). Our results on both datasets show that the synthetic edge creation matches both the self-similar scaling behavior and the diurnal patterns exhibited by the real data. Furthermore, successive snapshots of the graph structure generated by our model match the corresponding snapshots of the original data on a variety of metrics, including average path length and average clustering coefficient.

Key contributions in our work are as follows:

—We find that Renren's edge creation process is non-stationary over long-term periods. Even after removing the impact of node arrivals, traditional R/S and variance methods still produce inconclusive results on self-similar scaling. Thus, the two methods are unsuitable for measuring self-similarity in real traces in social networks (Section 3).
—By applying the more robust wavelet-based method for examining self-similarity, we find the edge creation process in Renren does exhibit properties consistent with self-similarity over time scales ranging from seconds to hours. We find the wavelet-based method to be highly robust detecting self-similarity in the presence of non-stationary trends (Section 4).
—We cross-validate our observations by repeating the above analyses on the Facebook wall post dataset and confirm that it exhibits similar self-similarity properties observed from the Renren dataset (Section 5).
—We propose a detailed model of social network dynamics that captures both the temporal properties of graph dynamics, in terms of self-similar scaling and deterministic non-stationary periodic patterns like diurnal or weekly cycles of user activity, and

its spatial properties, including long-term graph distance shrinkage and reduction in local clustering (Section 6).
—We validate our model by showing that it produces dynamic traces that match key properties of the original Renren and Facebook datasets, both temporally and spatially. Thus, by providing a practical method for generating realistic traces of time-stamped network events, our model fills an existing void in the research community (Section 7).

To the best of our knowledge, our work is the first to empirically study the presence of self-similarity in the time dynamics of OSNs. Our findings highlight that, instead of traditional Poisson models, the dynamics of real-world networks such as a Renren social graph can often be adequately captured by a combination of a non-stationary component, for example, long-term deterministic trends, and a stationary component, for example, a self-similar process. We believe that our model is the first to explicitly account for both temporal and spatial features in network dynamics and addresses an urgent need for accurate models of graph dynamics.

2. BACKGROUND AND DATASETS

In this section, we introduce briefly the notion of self-similarity and describe the Renren and Facebook dataset used in our study.

Self-similarity. For a time process, self-similarity refers to an invariance behavior, where certain statistical properties are similar under appropriately rescaled versions of the process [Beran 1994; Leland et al. 1994; Cox 1984]. Self-similarity has been observed in a variety of contexts in computing systems and networks, including web traffic [Crovella and Bestavros 1997], file system accesses [Gribble et al. 1998], and traffic in both wide-area networks [Paxson and Floyd 1995] and local Ethernet networks [Leland et al. 1994]. For self-similar traffic, the aggregation of many bursty sources remains bursty across a wide range of time scales. This behavior differs considerably from conventional Poisson processes that tend to produce traffic that smoothes out when observed over large time scales. While self-similarity can also be associated with geometry and describe the invariance in hierarchical structures [Song et al. 2005], this work focuses on the temporal domain.[2]

To formally define self-similarity, let $X = \{X_i : i = 1, 2, \ldots\}$ be a *covariance stationary* stochastic process whose autocorrelation function $r(k) \propto k^{-\beta}$ $(0 < \beta < 1)$ as $k \to \infty$. For each integer m $(m > 0)$, we form a new process $X^{(m)}$ representing the averaged values of X over disjoint blocks of size m. That is, the jth element of $X^{(m)}$ is

$$X_j^{(m)} = \frac{1}{m}\left(X_{(j-1)m+1} + X_{(j-1)m+2} + \cdots + X_{jm}\right), j = 1, 2, \ldots \quad (1)$$

If X is self-similar, then $r^{(m)}(k)$, the autocorrelation function of $X^{(m)}$, should satisfy [Gribble et al. 1998; Leland et al. 1994]:

$$r^{(m)}(k) = r(k), \quad \text{or} \quad r^{(m)}(k) \to r(k), \ m \to \infty. \quad (2)$$

An effective and commonly used metric to detect the existence or quantify the degree of self-similarity is the *Hurst parameter* (H), measureable in multiple ways [Abry and Veitch 1998; Leland et al. 1994]. Intuitively, H helps to capture the "burstiness" of a covariance stationary process, where a higher H corresponds to a process with more pronounced "bursts," that is, large observations have a tendency to be followed by large observations and small observations by small ones. Formally, $H = 1 - \beta/2$, where β

[2]Throughout the article, we refer to *temporal* self-similarity as self-similarity.

Table I. Statistics of the Two OSN Datasets, with the Start/End Date of the Traces, the Granularity of Time Stamps in the Traces, the Total Count of Nodes That Have Been Involved in Edge Creation, and the Total Count of Edges That Have Been Newly Created in the Traces

Graph	Trace Start Date	Trace End Date	Granularity	# of Nodes	# of Edges
Renren (Non-sampled) [Zhao et al. 2012]	11/21/05	12/31/07	Seconds	10,572,832	199,564,006
Facebook (New Orleans) [Kunegis 2013]	09/14/04	01/22/09	Seconds	46,952	876,993

is defined by the process X's autocorrelation function $r(k) \propto k^{-\beta}$. A process exhibits self-similarity if H falls in the range of $(0.5, 1)$.

Ideally, the finite-dimensional distributions of a self-similar process should stay invariant across all time scales. In reality, this property often exists at smaller time scales but breaks down at large time scales due to non-stationary patterns and finite datasets [Garrett and Willinger 1994; Gribble et al. 1998]. For example, diurnal user activity breaks stationarity and interferes with self-similarity at time scales larger than a few hours. Thus, analyzing for self-similarity requires determining the range of time scales over which it is visible [Abry and Veitch 1998; Garrett and Willinger 1994; Gribble et al. 1998].

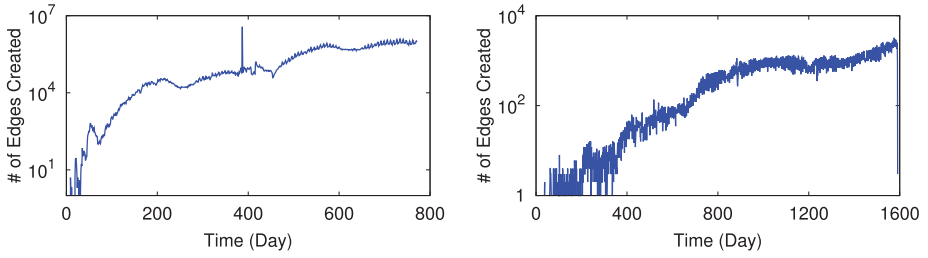
Datasets. An OSN is an online platform to build social relations among people who share similar interests, opinions, or have real-life connections.[3] While many have diverse features, they typically share features that allow individuals to construct a page or profile and build connections with other users, for example, by friending others. When modeling OSNs, an individual user is usually regarded as a "node," while the relationship between a pair of users is regarded as an "edge" or a "link."

Our analysis is based on the following OSNs: Facebook and Renren, where our work is the first to empirically study the presence of self-similarity in the time dynamics of OSNs. Facebook is the world's most popular online social network with over 1.5 billion users,[4] while Renren is the Chinese version of Facebook, the largest and oldest OSN in China with more than 220 million users [Jiang et al. 2010]. For both sites, a registered user can create his or her profile, add other users as "friends," and post messages on others' wall (called "wall posts"), an area on each user's own profile where others (usually friends) can make comments.

We show the summarized statistics of the two datasets in Table I.The first and primary is an anonymized dataset from Renren [Zhao et al. 2012], with a detailed time-stamped (down to the second) trace of the creation of all nodes (10,572,832) and all edges (199,564,006) over a 25-month period from November 21, 2005 (the launch of Renren), to December 31, 2007. Here an edge is created when two users become friends. To the best of our knowledge, this is one of the largest time-stamped datasets on social network evolution studied to date.

Figure 1(b) plots the daily edge growth of the Renren social network, where data points represent the number of *new* edges created on each day. This plot shows that the dataset covers both the initial explosive growth (from day 1 to around day 200) and the stabilized evolution of the Renren network [Zhao et al. 2012]. Note the unusually large spike on day 386 (December 12, 2006). This is the result of a merge event: Renren merged with *5Q*, its largest Chinese competitor at that time. The network doubled in

[3]https://en.wikipedia.org/wiki/Social_networking_service.
[4]https://en.wikipedia.org/wiki/Facebook.



(a) Renren (b) Facebook

Fig. 1. Daily edge growth in both Renren and Facebook datasets.

size in a single day, growing from 624K users and 8.2M links to 1.3M users with 11.2M links. Given it is a one-time event, we exclude it from our analysis and focus our study on continuous data segments before or after the merge.

The second dataset is the Facebook wall post dataset[5] [Kunegis 2013]. It contains wall posts produced by users from the Facebook New Orleans regional network, that is, 46, 952 users and 876, 993 posts created over a 4-year period from September 14, 2004, to January 22, 2009. Each post is also time stamped to the granularity of a second. Like in Wilson et al. [2009], we consider each wall post as an edge representing an interaction between two users. Figure 1(b) plots the daily edge growth of the Facebook social network. Like Renren, this dataset also covers periods where edge creation events increase significantly at the beginning and then stabilize (around day 750). Compared to Renren, this dataset is much more sparse.

3. PRELIMINARY ANALYSIS

Our goal is to determine if Renren and Facebook's network evolution display any property consistent with self-similarity and, if so, over what range of time scales. For clarity we first describe our analysis for Renren, which we repeat on the Facebook dataset in Section 5. Our analysis focuses on the edge creation process, since initial analysis showed no particular structure underlying the observed node creation events. The key challenge we face is how to identify and isolate the impact of non-stationary patterns in the edge creation data. As a first step, we limit the impact of new node arrivals on edge creation by focusing our analysis on edges created between members of a fixed user population. We remove this restriction and extend our analysis for all edge creation events in Section 4.3.

Next, we start by briefly describing how we sample the original dataset by removing certain node arrival and other obvious non-stationary events. We then discuss the methods for detecting self-similarity, our initial analytical findings, and key insights.

3.1. Experiment Setup

Data Sampling. We begin our analysis with a conservatively sampled subset of our data to remove obvious non-stationary factors that may impede any direct analysis of self-similar scaling property. Specifically, we limit our sample to include only existing users as of December 1, 2007, and study all edge creation events between them during December 2007, that is, days 741 to 771. This sampling eliminates three factors. First, by studying only edges created between members of a fixed user population, we minimize the impact of new node arrivals. Second, this month avoids the abnormal

[5]http://konect.uni-koblenz.de/networks/facebook-wosn-wall.

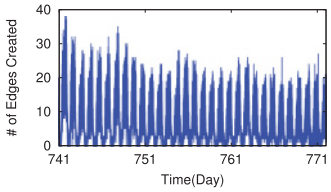


Fig. 2. Edge growth in sampled dataset of Renren, in terms of the number of new edges created per second. It shows a clear diurnal pattern.

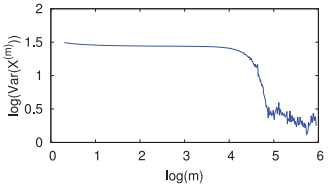


Fig. 3. Variance analysis of sampled dataset of Renren: The slope changes greatly when $m > 10^4$ seconds (≈3 hours), preventing direct analysis on self-similarity.

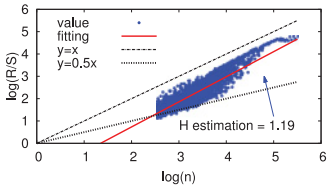


Fig. 4. R/S analysis of sampled dataset of Renren: H estimation is beyond range of self-similarity, and data shape changes significantly for $n > 10^4$ seconds (≈3 hours).

expansion of new edges around day 386 as a result of the one-time merge event of two social networks (Renren and 5Q). Finally, this time period is sufficiently late in the history of Renren that it avoids the initial exponential network growth experienced by most social networks [Zhao et al. 2012]. This data sample represents a stable growth period in Renren, which contains 18,714,712 edges created between 6,219,531 existing users. In the following, we refer to this sampled dataset as "sampled dataset of Renren" to differ from the entire dataset without sampling as "full Renren."

Estimating *H*. The two most popular (and simple) methods to estimate H are *variance analysis* and *R/S analysis* [Garrett and Willinger 1994; Gribble et al. 1998; Leland et al. 1994]. Our initial analysis efforts consist of applying these two methods in addition to directly visualizing the raw data.

Variance fitting method [Leland et al. 1994; Paxson and Floyd 1995] analyzes the decaying behavior of variances of the aggregated processes $X^{(m)}$ introduced earlier, with m the block size. From Equation (2) in Section 2, a self-similar process X satisfies $\log(Var(X^{(m)})) \propto -\beta \log(m)$ when $m \to \infty$, where $\beta = 2(1 - H)$. Thus by linearly fitting the plot of $\log(Var(X^{(m)}))$ versus $\log(m)$, this method can estimate β and then $H = 1 - \beta/2$.

R/S analysis computes H by measuring how apparent the variability of a time series changes with the length of the time-period being considered, which can be formally captured by the R/S statistic [Gribble et al. 1998; Leland et al. 1994]. To compute H, it divides the process X into blocks of size n and computes the corresponding R/S statistic $R(n)/S(n)$. Because $E[R(n)/S(n)] \propto n^H$ [Gribble et al. 1998] for self-similar processes, H is estimated using the slope of $\log(E[R(n)/S(n)])$ versus $\log(n)$.

3.2. Measurement Results

We now present the results using three heuristics: *visualization of raw data*, *variance analysis*, and *R/S analysis*.

A Long-Term Diurnal Pattern. Figure 2 visualizes the edge creation process by plotting the number of new edges created in each second over the one month (days 741–771). We can clearly observe a diurnal pattern in the edge creation process. This non-stationary behavior precludes any direct analysis of self-similarity. We confirm this from the results of the variance and R/S analysis. Figure 3 plots the values of $\log(Var(X^{(m)}))$ against $\log(m)$. The curve maintains a linear shape until m reaches 10^4 seconds (≈3 hours), and then its slope changes significantly. Similarly, Figure 4 plots in log-log scale individual R/S statistics as a function of the block size n (in seconds). The red straight line shows the best linear fit and its slope results in an H-estimate of $H = 1.19$, clearly outside the allowed range of $(0.5 < H < 1)$.

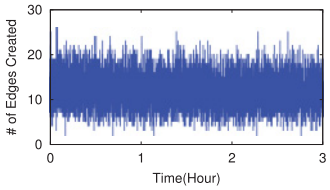


Fig. 5. An example of edge growth of a randomly chosen 3-hour segment in the sampled dataset of Renren. It is highly bursty, appears stationary and suggests further exploration for self-similar scaling behavior.

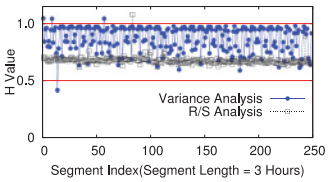


Fig. 6. Estimates of H by both Variance and R/S analysis on disjoint 3-hour segments in the sampled dataset of Renren, where 98%+ of H-estimates fall within (0.5,1).

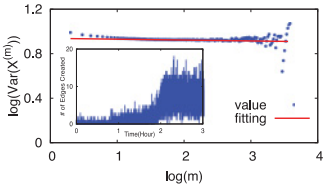


Fig. 7. An example of poor line fitting in variance analysis, which has poor $R^2 = 0.0458$. This is also confirmed by the inset that displays the raw edge growth during the corresponding time period and shows a clearly non-stationary event.

The appearance of such a pronounced diurnal pattern has a direct impact on subsequent efforts to model our dataset. It suggests that models should include a component that accounts for this expected user-generated periodic behavior.

Self-Similar Fluctuations. An interesting observation from Figure 2 is that the fluctuations on top of the diurnal component display a bursty behavior. Similarly, Figures 3 and 4 show that the curve only starts to lose its line shape when m or n exceeds 10^4 seconds ($\approx$3 hours). Figure 5 shows the edge creation events of a randomly chosen 3-hour segment (6pm–9pm, December 16, 2007). It is highly bursty and appears stationary and could therefore exhibit self-similar scaling behavior. Together, these observations suggest that over time scales not significantly impacted by the observed diurnal patterns (i.e., a few hours and below), the edge creation process may be consistent with self-similar scaling behavior.

We confirm this intuition by performing variance and R/S analysis on each 3-hour log segment and computing its H value. Figure 6 plots the results over the entire month as 248 disjoint 3-hour segments. H-estimates based on the variance analysis method vary across segments, with a mean of 0.89 and variance of 0.01, while R/S analysis remains stable, with mean of 0.68 and variance 0.001. For both methods, an overwhelming majority of segments (98.4% for variance, 99.5% for R/S) estimates H within ($0.5 < H < 1$). These results suggest that the Renren edge creation process exhibits self-similarity over time scales ranging from seconds to hours.

3.3. The Reliability of our *H* Estimates

In our analysis, we encountered potential issues regarding the reliability of H-estimates using the variance and R/S analysis methods. For some segments, the methods produced poorly-fitting linear regression lines, which in turn resulted in highly questionable estimates of H. Figure 7 shows an example of such a "problematic" segment (6–9am, December 6, 2007), where the line fitting is poor via variance analysis. We also plot as an inset in the figure the raw edge growth during the time period, which shows a clearly non-stationary event. We further study these events in Section 4.2.

To quantify the impact of such poor data fitting on the obtained H-estimates, we compute the coefficient of determination R^2 for each segment. R^2 measures how well the observed data points are represented by a straight line. Like Gribble et al. [1998], we use the criterion of $R^2 > 0.9$ to indicate that the fitting is sufficiently good to provide a reliable H-estimate. Of all segments, 38.3% have unreliable H-estimates by R/S analysis vs. 71.0% by variance analysis. Prior studies have reported similar reliability issues [Karagiannis et al. 2002; Taqqu et al. 1995].

3.4. Summary of Observations

Our initial analysis led to three main findings. *First*, the Renren edge creation displays a typical diurnal pattern in user activity that makes the process inherently non-stationary, preventing a direct analysis of self-similarity. This suggests that any accurate model of Renren's edge creation process must include a component that explicitly accounts for this periodic behavior. *Second*, local fluctuations on top of the periodic component display behavior that indicates potential self-similarity. *Finally*, we find that two commonly used methods, that is, variance and R/S analysis methods, cannot provide reliable H-estimates for real data that displays non-stationary patterns.

Thus, our next step is to avoid most of the encountered problems by applying a more rigorous method for systematically analyzing data with potential scaling behavior that has strong robustness properties with respect to underlying non-stationary patterns and results in H-estimates with known statistical properties (e.g., confidence intervals).

4. WAVELET-BASED ANALYSIS

Following our initial analysis, in this section we apply a more rigorous wavelet-based method to systematically study potential self-similar scaling behavior exhibited by our dataset. This method has strong robustness against underlying non-stationary patterns and can provide H-estimates with confidence intervals. To this end, we first briefly introduce the wavelet method and then present our findings.

4.1. The Wavelet Method

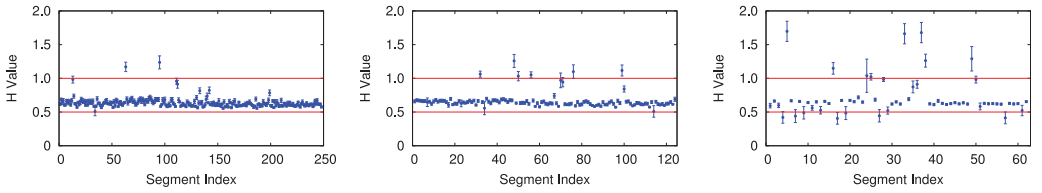
Estimation errors of the variance and R/S analysis methods can be attributed to their "eyeballing" approach when attempting to identify self-similarity in highly variable data. In contrast, the wavelet-based method offers a principled and rigorous analysis of a given dataset's scaling property by isolating characteristics of data via a combined scale-time presentation. In turn, it provides a more reliable self-similarity analysis [Abry and Veitch 1998].

In short, wavelet-based analysis represents a process X by a sequence of subspaces $\{W_j\}_{j\in Z}$ where W_j is at a finer scale than W_{j-1} ($W_j \subset W_{j-1}$). This way, it can reveal detailed properties of X at different time scales. If X is self-similar, then its projection on the W_j subspace Γ_j, satisfies $E[\Gamma_j] \sim |2^{-j}v_0|^{1-2H}$. Here $2^{-j}v_0$ represents the reference frequency of the jth subspace W_j while v_0 is the reference frequency of the root subspace W_0. One can estimate the Hurst parameter H by plotting $E[\Gamma_j]$ vs. j on a log-log scale and applying linear regression.

We estimate H using the wavelet software developed [Abry and Veitch 1998] for self-similarity analysis. By carefully choosing the number of vanishing moments N that controls v_0, the tool can systematically detect and remove the impact of various types of deterministic trends in the dataset. Furthermore, it also relies on known theoretical properties of the resulting H-estimate to provide confidence interval (CI) for H. In the analysis of our dataset, we choose the value of N that produces both a good fit and the smallest confidence interval.

4.2. Measurement Results

We seek to confirm and substantiate our preliminary results that show properties consistent with self-similar scaling in our Renren dataset. We divide our sampled dataset into disjointed segments of lengths between 3 and 12 hours and apply the wavelet-based analysis to each segment. In our analysis, we refer to a segment as "abnormal" if its H-estimate (including its 95% confidence interval) does not completely fall within the self-similar range (0.5, 1). Our analysis leads to two key findings.



(a) Segment Length = 3 hours (b) Segment Length = 6 hours (c) Segment Length = 12 hours

Fig. 8. Wavelet analysis on data segments with different segment lengths (sampled dataset of Renren).

Table II. Statistics of Wavelet Analysis on 3-Hour Segments with Start Time Shifts (Sampled Dataset of Renren)

Start Time Shift	Normal Segments		Abnormal Segment Portion
	H mean	H variance	
0 hour	0.631	0.002	2.02%
1 hour	0.633	0.002	2.43%
2 hours	0.629	0.002	2.02%

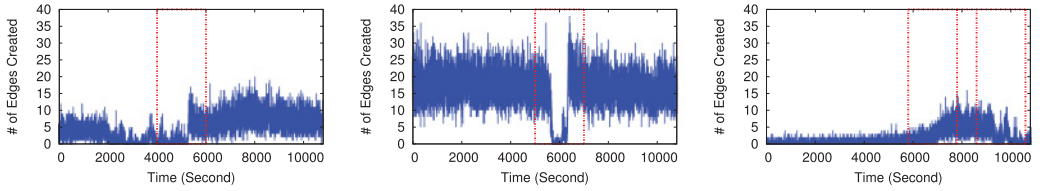
Self-Similarity Over Time Scales from Seconds to Hours. Our analysis confirms that over time scales ranging from seconds to a few hours, Renren's edge creation process exhibit properties consistent with self-similar scaling. As an example, Figure 8(a) shows the H-estimates with their 95% confidence interval for all 248 3-hour-long segments. Only 5 segments are abnormal, while the rest (98%) consistently produce H-estimates within $(0.5, 1)$ and tightly clustered around $H = 0.63$.

To examine the robustness of our results, we check different segment compositions by shifting the start time of each segment by 0, 1, and 2 hours separately. From the summarized results in Table II, we notice the stability in the mean (0.63) and variance (0.002) of H-estimates for normal segments and also the portion of segments deemed abnormal ($2.02\% \sim 2.43\%$) These results provide further evidence that Renren's edge creation process behaves properties consistent with self-similar scaling over time scales from seconds to a few hours.

Scaling Behavior over Larger Time Scales. We observe that the number of abnormal segments increases as the segment size increases. Figures 8(b) and (c) plots the H-estimates across all segments for segment lengths of 6 and 12 hours. The ratio of abnormal segments increases to 8.1% for 6-hour segments, and up to 32.3% for 12-hour ones. It confirms our earlier conclusion that the properties consistent with self-similar scaling weaken in Renren's edge creation process, when viewed over larger time scales. This phenomenon is perhaps due to the presence of harder-to-account-for non-stationary patterns, such as heteroscedasticonfidence intervalty (i.e., edge creation in Renren is more variable during peak hours than during low hours).

Patterns of Abnormal Segments. We also wish to understand patterns and potential causes for the observed abnormal segments. We find that these abnormal segments are randomly distributed across days, and within a day, around 60% of them appear during 6–9pm, when Renren users are most active (the number of edges created account for 23% of the whole day).

We also find that abnormal segments are caused by sudden changes in the edge growth process. Based on the edge growth patterns, we are able to classify abnormal segments into three types, all shown in Figure 9. These include *level shift*, where the volume of edge growth suddenly increases (or decreases); *momentary drop,* where the growth experiences a short period of extremely low activity; and *ramp up / down*, where



(a) Level shift (b) Momentary outage (c) Ramp up/down

Fig. 9. Examples of three types of abnormal segments, where the red dot boxes show the unusual edge creation events (sampled dataset of Renren).

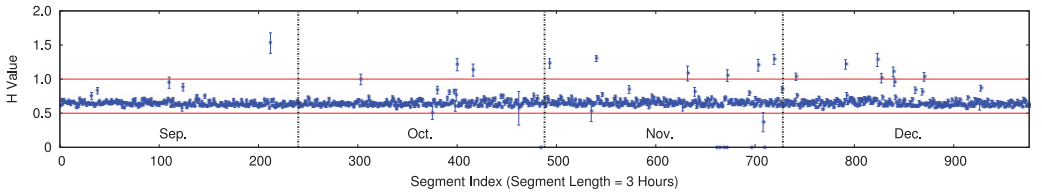


Fig. 10. The H-estimates of all the disjoint 3-hour segments between September and December 2007 of the Renren dataset, after performing wavelet analysis on the entire dataset without sampling (full Renren). The results align with those with sampling (labeled as "sampled dataset of Renren" in caption).

the edge activity quickly increases or decreases in the segment. Our collaborators at Renren have confirmed that while per-hour identification is difficult, it is possible that at least some of these abnormal events match changes to the site and its features. Intuitively, level shifts and momentary drops might be caused by new features or localized failures, and ramp up/down events might correspond to ad promotions to increase user membership. We are working with Renren to further confirm this.

4.3. Analysis without Sampling

Finally, we expand our analysis to consider the full, unsampled dataset. This is to examine whether the observed property consistent with self-similar scaling on the sampled data still present after including new nodes with rapid (and non-stationary) edge growth.

We first consider the complete dataset from the month of December 2007. Interestingly, 97% of the 3-hour segments produce H-estimates within the self-similar range, with mean $H = 0.65$. We show detailed H-estimates in Figure 10, which are highly consistent with our prior analysis on the sampled dataset (Figure 8(a)). The only minor difference is two additional abnormal segments, possibly caused by non-stationary edge growth of the new nodes.

Next, we examine all edge events in the year of 2007. Again, we get consistent results: H-estimates of 97% of the 3-hour segments fall into the self-similar range, with mean $H = 0.64$. Figure 10 shows H-estimates for September–December 2007 (due to the space limit), which are representative of all other months. Together, these results suggest, with high possibility, that the same self-similar property is present consistently throughout time. These results also confirm the high reliability of the wavelet method in self-similarity detection.

4.4. Summary

We apply the more reliable and accurate wavelet method to detect self-similarity at different time scales in Renren's edge creation process. The outcomes confirm prior observations from R/S and variance results, with high confidence, that the property

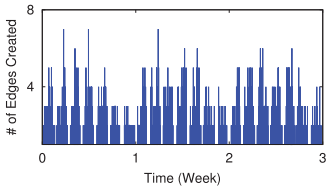


Fig. 11. An example of edge growth of a randomly chosen 3-week data (Facebook).

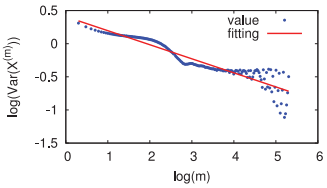


Fig. 12. Variance analysis on the entire data: doutable fitting with curves around 10^3 units (Facebook).

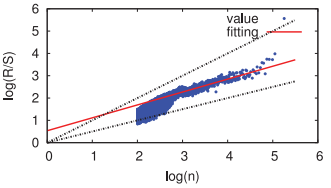


Fig. 13. R/S analysis on the entire data: doutable fitting since the shape changes greatly after 10^3 units (Facebook).

consistent with self-similar scaling lasts to several hours. This property also holds for our full, unsampled dataset (after the network merge).

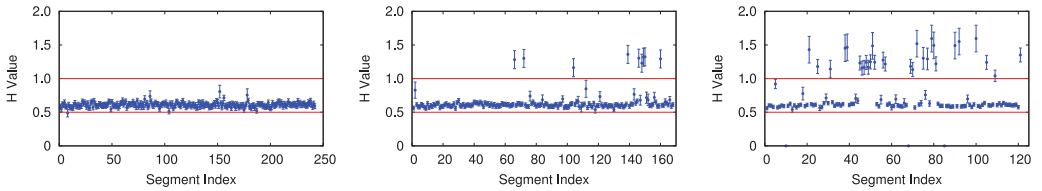
5. VALIDATION VIA FACEBOOK DATASET

One reasonable question is whether our results are strongly biased by our choice of dataset, that is, property consistent with self-similarity is only present in Renren network. Here, we validate our findings using the Facebook wall post dataset [Kunegis 2013]. Recall that for self-similar property to be detectable, a dataset must cover temporal events in fine granularity and have sufficient event frequency to provide meaningful statistics. To our knowledge, the Facebook wall post dataset [Kunegis 2013] is the *only* dataset aside from our Renren dataset that meets these requirements.

As Figure 1(b) shows, like Renren, the number of edge creations in the Facebook dataset increases significantly at the beginning and stabilizes around day 750. To eliminate the impact of this obvious non-stationary increasing trend, we focus on the edge creation process after day 750 (for a total duration of 841 days). Compared to Renren, this dataset is much more sparse, and per-second level analysis does not show any meaningful statistics (only 1.15% of non-zero data points). Thus, we enlarge the time unit for analysis to 120 seconds, where the resulting ratio of non-zero data points (61.18%) is comparable to that of Renren (61.46%).

Following analysis in Sections 3 and 4, we start by visulizing the raw data in the Facebook dataset and then apply the variance, R/S, and wavelet anlaysis methods to see whether any property consistent with self-similar scaling exists across the whole time range. Figure 11 shows a random sample of three successive weeks (from day 762 to day 782) in the edge creation process, which displays a clear weekly pattern. Similarly, this obvious non-stationary behavior precludes any direct analysis on self-similarity. We also confirm this using the R/S and variance analysis methods, where for three successive weeks the estimated H values are 0.5779 and 0.8940, respectively. Although these H-estimates are within the self-similar range (0.5, 1), Figures 12 and 13 show that the two methods have poor data fitting, resulting in unreliable estimations on H. On the other hand, the wavelet analysis produces an H value of 1.11, indicating that there is no property consistent with self-similarity across the entire time range. Together, all these results suggest that over the time scale up to years, we cannot reliably detect self-similarity properties in the Facebook dataset.

Next, we explore whether the dataset displays self-similar scaling properties on shorter time scales. We split the entire dataset into fixed size segments of lengths varying between 1 and 7 days and apply the wavelet analysis to each segment. Figures 14(a)–(c) plot the H-estimates with their 95% CI at segment length of 3.5 days, 5 days, and 7 days, respectively. We obtain two key observations. *First*, we observe strong self-similarity properties over the time scale between minutes and days. Figure 14(a) shows that 98.35% of 3.5-day segments have H values with 95% CI falling into range



(a) Segment length = 3.5 days (b) Segment length = 5 days (c) Segment length = 7 days

Fig. 14. Wavelet analysis on data segments with different segment lengths (Facebook).

Table III. Statistics of Wavelet Analysis on 3.5-Day Segments with Start Time Shifts (Facebook)

Start Time Shift	Normal Segments		Abnormal Segment Portion
	H mean	H variance	
0 day	0.612	0.001	1.65%
1 day	0.611	0.001	2.07%
2 days	0.611	0.001	2.07%
3 days	0.613	0.001	3.32%

$(0.5, 1)$, centered around $H = 0.61$. By shifting start times of segments, the consistent results in Table III further confirm this observation.

Second, the portion of abnormal segments (whose H-estimates are from $(0.5, 1)$) increases with segment length, that is, 1.65% for 3.5 days, 5.95% for 5 days, and 26.45% for 7 days. A detailed analysis on the dataset shows that this is mostly caused by a weekly pattern (as shown in Figure 11) of user activities that dominates at larger time scales.

In summary, our results on the Facebook dataset align very well with our observations from the Renren dataset. Due to the existence of non-stationary patterns introduced by human behaviors, for example, diurnal or weekly user activities, properties consistent with self-similar scaling exist but only hold over certain time ranges and gradually weaken at larger time scales.

6. A MODEL OF NETWORK DYNAMICS

Motivated by our self-similarity analysis of Renren and Facebook's edge creation process, we next seek to build a complete model of social network dynamics. Our proposed model includes two components: a *temporal* component that produces a sequence of time-stamped events defining *when* and *how many* new edges are formed in a given time interval and a *spatial* component defining *where* in the graph these new edge creations take place (i.e., which nodes are involved). Ideally, the model should produce synthetic dynamic graphs whose edge creation will display deterministic non-stationary periodic patterns (e.g., diurnal or weekly user activities) and properties consistent with self-similarity and whose graph structural changes match those observed from the original data and account for key spatial properties, for example, graph densification, path shrinkage, and local declustering [Zhao et al. 2012]. Next, we explain the model in detail and provide validation in Section 7.

6.1. The Temporal Component

Our analysis in Sections 3, 4, and 5 shows that, for both Renren and Facebook datasets, the edge creation process displays a combination of deterministic non-stationary periodic patterns, that is, diurnal or weekly user activities and properties consistent with self-similarity. These observations motivate designing the temporal component of our model as a combination of two sub-modules: a *non-stationary* module that captures

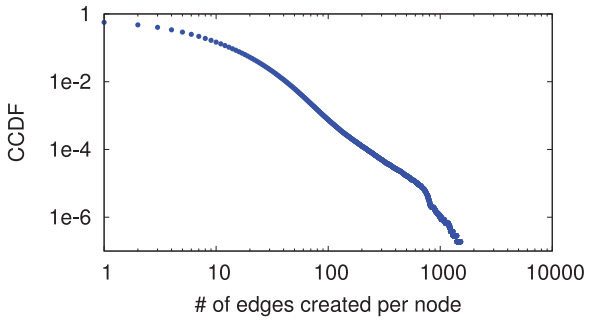


Fig. 15. CCDF of the number of edges created per user in December 2007 in the Renren dataset.

the predictable cycles in user activities, for example, daily or weekly cycles, and a *self-similar* module that parsimoniously accounts for the inherent burstiness in user edge creations over certain time scales, for example, from seconds to a few hours.

The Self-Similar Module. Prior work has demonstrated two effective methods for producing self-similar traffic. The first method aggregates many ON/OFF processes and, under certain conditions, the superposition process displays a self-similar scaling [Willinger et al. 1997; Gribble et al. 1998]. In particular, this construction requires statistical knowledge of the ON and OFF periods and assumes that either of those periods are modeled by a heavy-tailed distributions. The second method is based on the $M|G|\infty$ queuing model [Cox 1984; Willinger et al. 1998]. Here, each source arrives according to a Poisson process, and the distribution of its active time is assumed to be heavy-tailed, for example, the Pareto distribution. During its active time, each source is assumed to operate at a constant rate. Then the resulting count process $\{N_t, t = 0, 1, 2, \dots\}$, where N_t is the number of active sources at time t, is self-similar. In other words, by multiplexing sources with Poisson arrivals and heavy-tailed active times, one can produce a self-similar process.

Examining our two datasets in more detail shows that the $M|G|\infty$-based method provides an intuitive way and a good fit for modeling edge creation. For one, we observe that, over time, the number of edges created per user follows a heavy-tailed distribution. For example, Figure 15 plots the distribution of the number of edges created per Renren user during December 2007, which can be approximated as a heavy-tailed pattern. Moreover, assuming each user creates edges at a constant rate, the active time of a user is directly proportional to the number of edges that user created. This in turn implies that each user's active time also follows a heavy-tailed distribution, consistent with the $M|G|\infty$-based construction of self-similar processes.

Based on this intuition, we build the self-similar module based on a standard $M|G|\infty$ process [Cox 1984]. Users arrive according to a Poisson process with rate λ. On arrival, each user independently starts its active time duration T_i (seconds) chosen from a Pareto distribution,

$$P(X > x) = \left(\frac{x_m}{x}\right)^{\alpha}, \quad x \geq x_m, \quad 1 < \alpha < 2.$$

Assuming that each user creates edges at a constant rate γ/s, we can calculate the total expected number of edges created by user i by $T_i \cdot \gamma$. Since an edge creation involves two users, we derive the number of edges S_t created at time t from the number of active users N_t: $S_t = \gamma \cdot N_t/2$. The time series $\{S_t, t = 0s, 1s, 2s, \dots\}$ defines the self-similar module of the temporal component of our model.

The Non-Stationary Module. We extract the deterministic non-stationary periodic component by subtracting the self-similar component from the original edge creation process. Suppose the number of original edge creation is O_t at time t. Then the subtraction produces a process $\{U_t = O_t - S_t, t = 0s, 1s, 2s, \dots\}$. Next we apply a sliding window over U_t to obtain a smooth deterministic process and then fit it with a periodic function, that is, Sine, to produce D_t, the non-stationary module of the temporal component of our model.

Integrating the Two Modules. We combine S_t and D_t and obtain our targeted edge creation process E_t: $\{E_t = S_t + D_t, t = 0s, 1s, 2s, \dots\}$. Since the non-stationary periodic component D_t may generate negative values, we set a minimum for the sum to be 0. Note that we designed this temporal component to describe new edge creations aggregated across all the users. Importantly, this temporal component only generates timestamps of new edges (in terms of the total number of edges created in each second) but does not associate any of these new edges to specific users. In other words, the temporal component will produce the total number of edges created in each second, but will not predict which nodes created these edges. This is because we design the temporal component to specifically capture the edge dynamics aggregated across all the users, that is, property consistent with self-similar scaling and deterministic non-stationary periodic user patterns. The actual distribution or mapping of edge events across users is performed by the spatial component of our model, which we will describe in Section 6.2.

6.2. The Spatial Component

To determine where each new edge is created as part of the overall network evolution process, we first highlight two key observations made by our prior analysis on the Renren network [Zhao et al. 2012]. *First*, after an initial bursty growth phase, new edge creation was dominated by existing nodes (>80%). This empirical result diverges from generative models, which assume that new node arrivals drive edge creation regardless of network size. *Second*, we observe three structural properties over time: graph densification, distance shrinkage, and high but decreasing clustering coefficient (CC). Existing graph models [Akoglu and Faloutsos 2009; Akoglu et al. 2008; Bonato et al. 2009; Leskovec et al. 2005] capture only a subset of them.

Intuition. We consider a stable social network in a state of ongoing growth.[6] After a fast initial period of explosive growth, the arrival rate of new users becomes relatively small compared to existing users. At this point, continuous friend discovery between existing users dwarfs the initial bursts of edge creations triggered by new user arrivals. Therefore, in our model, we use interarrival gaps between new users as iterations to drive the formation of new edges between existing users.

With these in mind, our model will focus on the creation of edges between existing users following the arrival of each new user. Specifically, we assume a new user u_i creates an edge before the arrival of the next user u_{i+1}, and after this edge creation u_i immediately becomes an "existing user." We hypothesize that existing users are often introduced to groups of friends, either discovering the presence of an offline friend (and other mutual friends) or creating new groups of friends via common interests or social applications. To capture this intuition, in each iteration, our model selects two existing nodes u and v at random and connects u repeatedly to multiple users in v's neighborhood. Here v can be an existing friend of u or a previously unknown "stranger." The continuous formation of random connections between existing users

[6]Note that our model explicitly targets the ongoing growth phase of a social network. We leave the measurement, analysis, and modeling of a network in decline for future work.

shrinks average path lengths and lowers clustering coefficient by building shortcuts between nodes, while connecting friends of friends slows the rate of declustering.

Model Details. The spatial component is strongly dependent on the temporal component to determine the maximum number of edges created in any iteration (that is, between two node arrivals). Let $F(n)$ represent the number of edges in the network when the network contains n nodes. Then $F(i+1)-F(i)$ represents the total number of edges created between the arrivals of u_i and u_{i+1}. With the knowledge of node arrival time statistics, that is, t_i and t_{i+1}, we can estimate the total number of edges k created between t_i and t_{i+1} as $k = F(i+1) - F(i) = \sum_{t=t_i}^{t_{i+1}} E_t$.

Specifically, our proposed edge formation process is defined as follows. We drive the process using a parameter p, which defines the probability a node is selected in the recursive edge creation process between existing nodes.

---

(1) When a new node u_i joins the network, $k = F(i+1) - F(i)$.
(2) **Edge creation by the new node:** The new node u_i randomly select an existing node u_j to connect. Set $k = k - 1$. Now u_i becomes an existing node.
(3) **Edge creation between existing nodes:** Randomly select two existing nodes u and v. If they are not connected, then connect them and set $k = k - 1$. Then u starts steps (a)–(c) to connect neighbors of v and repeat them until all the required edges have been created (i.e., $k = 0$) or there are no more nodes to connect. Each time an edge is created, set $k = k - 1$.
   (a) Generate a random number x following the geometric distribution with mean $(1-p)^{-1}$.
   (b) Randomly select neighbors of v that do not connect u until reaching any of the three situations:
      i. x neighbors are selected;
      ii. no more edges need to be created, that is, $k = 0$;
      iii. all available neighbors of v are selected. Let $R = \{r_1, r_2, \ldots\}$ be the set of selected nodes.
   (c) For each node $r_i \in R$, u connects r_i and repeats steps (a) and (b) on r_i.
(4) If more edges need to be created ($k \neq 0$), then repeat step (3).

Comparison to Existing Models. The existing model most similar to our new model is the Forest Fire model [Leskovec et al. 2005], which simulates network growth by creating edges between each new node to a set of existing nodes. A new node joining the network randomly connects to an existing node and some of its neighbors; this repeats across the network, like a fire burning through a forest. This "burning process" and our recursive edge creation process between existing nodes both act to produce a high clustering coefficient by recursively connecting to neighbors of neighbors.

Three key differences separate our model from Forest Fire. *First*, our model captures the observation that existing nodes drive edge creation in a stable growth network. *Second*, our model produces decreasing clustering coefficient by connecting pairs of random existing nodes. Forest Fire does not capture this property because it always forms close triangles in each node's neighborhood, leading to relatively high clustering coefficient unlikely to decrease over time. *Third*, our model can be accurately calibrated to the observed dynamics of an existing network trace by incorporating the network growth function from the temporal model. This additional flexibility makes it more attractive for generating realistic dynamic network traces.

7. MODEL VALIDATION

Having described our proposed model for network edge dynamics in Section 6, we next validate the proposed network dynamic model. We calibrate the model using real

data and use it to generate synthetic dynamic graphs and then compare these synthetic graphs to the original data in terms of both temporal and spatial properties. Since the temporal and spatial components are complementary and operate at different scales, we validate them sequentially to examine their contributions to network evolution. Because the output of the temporal component is used as an input to the spatial component, the validation on the spatial component also serves as validation of the complete model with both components. Our validation results on the Renren and Facebook datasets lead to the same observations. For brievity, we present the Renren results in detail in Sections 7.1 and 7.2 and summarize the Facebook results briefly in Section 7.3.

7.1. Validating the Temporal Component

Our validation is first based on the Renren dataset for the month of December 2007, the same datasets used in our self-similarity analysis (Section 3 and Section 4). We leave the validation of the Facebook dataset to Section 7.3. To validate our model, we first describe how we calibrate the model using the Renren dataset. As explained in Section 6.1, the temporal component consists of two sub-modules: a self-similar module (i.e., stationary stochastic process) and a non-stationary module (i.e., non-stationary deterministic function).

Calibrating the Self-Similar Module. We construct the self-similar module according to the $M|G|\infty$ model described earlier. That is, nodes arrive according to a Poisson process with rate λ, and the length of each node's active time is chosen independently from a Pareto distribution with parameters α and x_m. Consider the Renren edge creation data collected in December 2007 where $7,246,621$ nodes have created edges. We estimate the corresponding value of rate λ in the Poisson process of this period by the average active node count per second, that is, $\lambda \approx 2.7$/s. To derive the active time (in seconds) statistics, we leverage a proven relationship between H and α [Cox 1984; Leland et al. 1994]: $H = (3 - \alpha)/2$. Since our measured H-estimate for the December 2007 data is around 0.65, we set $\alpha = 1.7$. Finally, assuming a node creates edges at a constant rate of 1/s, the average number of edges created per node is then equal to the average active time across all the nodes, which can be calculated as the mean of the Pareto distribution, that is, $x_m * \alpha/(\alpha - 1)$. By measuring the average edges created per node in December 2007, we get $x_m \approx 3.2$.

Using the $M|G|\infty$-based method with $\lambda = 2.7$/s, $\alpha = 1.7$ and $x_m = 3.2$, we generate a synthetic trace that represents the edge creation process contributed by the self-similar module. Figure 16 plots a randomly chosen 3-hour segment in the synthetic trace, which displays burstiness similar to the original data. By applying the earlier-described R/S, variance and wavelet analysis methods, we get H-estimates 0.68, 0.63 (both with a good line fitting) and 0.69 (with the 95% confidence interval 0.0099), respectively. The graphical fitting in Figure 18 and Figure 17 show that both R/S method and variance analysis have good fit. All these validate that the resulting trace is indeed consistent with the designed-for self-similar scaling behavior (i.e., $H = 0.65$).

Calibrating the Non-Stationary Module. To calibrate the non-stationary module, we first subtract the synthetic trace generated by the self-similar module from the original edge creation data. We then apply a sliding window (with a window size of 1 hour and a step size of 1 second) to smooth the subtraction result over time. One sample of the smoothed data (for December 2007) is shown by the blue curve in Figure 19, displaying a daily pattern with almost 0 mean. The blue curve is well fitted by the sine function: $9.70 \sin(7.27 \cdot 10^{-5} t + 3.56) - 0.003$, shown as the red curve.

Validation Results. We sum the synthetic traces produced by the above two sub-modules to build a single synthetic edge creation trace and then compare this combined

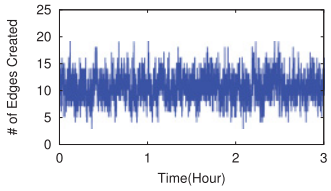


Fig. 16. An example of edge growth of a randomly chosen 3-hour segment in the synthetic self-similar module (Renren).

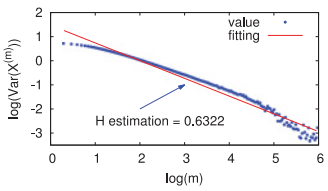


Fig. 17. Variance analysis of synthetic self-similar module: H estimation = 0.67 and in good linear fitting (Renren).

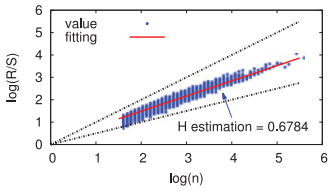


Fig. 18. R/S analysis of synthetic self-similar module: H estimation = 0.63 and in good linear fitting (Renren).

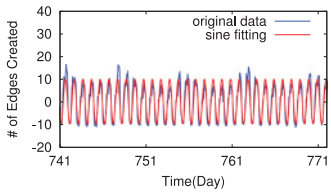


Fig. 19. The synthetic non-stationary module (red curve) well captured the smoothed diurnal pattern in the original dataset (blue curve) (Renren).

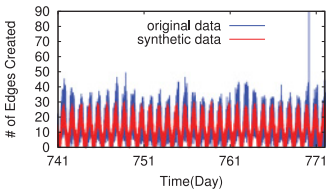


Fig. 20. Synthetic trace by our temporal component (red) vs. original edge creation process (blue) (Renren).

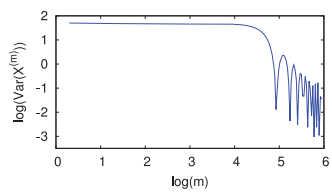


Fig. 21. Variance analysis of the entire synthetic trace: Like the original data, slope also changes for $m > 10^4$ seconds (≈3 hours) (Renren).

trace to the original data. Repeating the process *5 times* produces very consistent outcomes, for example the total edge counts are similar, with an average ratio between the synthetic and the original trace of 1.007 and variance $<10^{-6}$. Figure 20 plots a sample of one synthetic trace together with the original trace (for December 2007) and illustrates that the synthetic data displays diurnal patterns similar to the original data.

We further compare the synthetic and original traces by performing on the synthetic trace the same self-similarity analysis that we applied in Section 3 and Section 4 on the original trace. Figures 21 (variance analysis) and 22 (R/S analysis) demonstrate that the synthetic trace exhibits the very same issues that plagued our preliminary analysis of the original data; for example, scaling behavior changes drastically for time scales larger than a few hours, and H estimation is outside the theoretical range (0.5, 1.0), and thus non-stationary diurnal patterns prevent a direct scaling analysis of the data.

Next we apply the wavelet-based analysis method to examine the self-similar nature of the synthetic trace over 3-hour segments. Figure 23 plots the resulting H-estimates for each segment with 95% CI. We see that the H-estimates for the synthetic trace also fall consistently between (0.5, 1) with an exception of 4.03%, which closely matches the 3% exception seen from the original data. The average H value for the synthetic trace is around 0.75, again similar to that of the original trace (mean $H = 0.65$) as shown in Figure 8(a). Finally, we evaluate the robustness of our results by shifting the starting time of each segment by 0, 1, and 2 hours separately and find both the abnormal segment ratios and H estimates remain stable (we omit the results for brevity). Thus we conclude that the original trace and the synthetic traces are qualitatively and quantitatively similar.

Together, these results demonstrate that the temporal component of our model can accurately capture the diurnal patterns and self-similar scaling behavior displayed by the original Renren data. Furthermore, the contributions of the two sub-modules

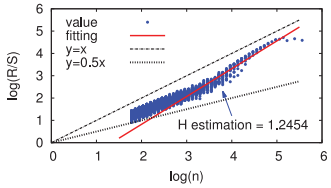


Fig. 22. R/S analysis of the entire synthetic trace: Like the original data, H-estimate is beyond the self-similar range, and data shape changes $n > 10^4$ seconds ($\approx$3 hours) (Renren).

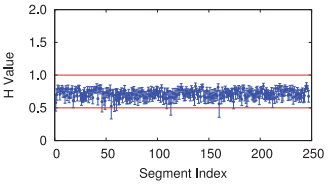


Fig. 23. Wavelet analysis on 3-hour segments of synthetic trace. Like the original data, the vast majority of segments have estimated H within (0.5,1) (Renren).

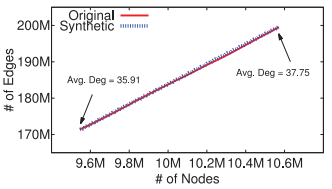


Fig. 24. Network growth of the synthetic trace generated by the temporal component vs. the original data (Renren).

illustrate why and how the presence of deterministic non-stationary periodic trends like diurnal user activity patterns impacts any direct scaling analysis of such non-stationary data.

Connecting the Temporal and Spatial Components. Recall that the spatial component of our model uses the temporal component to compute the number of edges created between each pair of node arrivals. As a result, we need to be able to accurately estimate the arrival time of each node. From our exploratory analysis, we noticed no specific properties of the node arrival process other than that it is largely consistent with a Poisson process with rate λ_{new}, where λ_{new} is estimated as the average number of new node arrivals per second.[7] Figure 24 shows that our solution can accurately predict the network edge growth in December 2007.

7.2. Validating the Spatial Component

Next, we validate our spatial component. Ideally, we would calibrate the model using the entire Renren dataset (from November 21, 2005, to December 31, 2007) and produce synthetic traces for the entire 25-month period. However, using the entire dataset is impractical for two reasons. First, due to the size of the network at the end of the 25-month period (i.e., 10.6M nodes and 199M edges), the calibration process would be computationally prohibitive. Second, the merge event on December 12, 2006, introduced significant changes to the network, impacting any analysis of the network's dynamics.

As a viable practical alternative, we use two subsets of the Renren data for validation. The first segment (referred to as *2006 Original*) covers the period from the launch of the network (November 21, 2005) until right before the merge event (December 11, 2006). The corresponding last snapshot of the graph includes 624K nodes and 8M edges. This represents the "early" period of the network. The second segment (*2007 Original*) covers the first 2 months of 2007, with the snapshot on December 31, 2006, as the initial graph, and its last snapshot has 1.75M nodes and 18M edges. This represents the "stable growth" period of the network. Table IV summarizes the observed network statistics for the two segments.

Spatial Component Calibration. We calibrate the component for the two segments separately. As discussed in Section 6.2, the spatial component has two parameters: *network edge growth function* $F(n)$ and *node selection probability* p. For the 2007 segment, we derive $F(n)$ from the temporal component. For the 2006 segment, however, we have

[7]This is analogous to the observation in Paxson and Floyd [1995] that while packet arrivals in network traffic appear better modeled using self-similar processes, Poisson effectively captures user session arrivals.

Table IV. Statistics of the Original Graph and the Synthetic Graph Generated by Our Spatial Component for Renren Dataset. The 2006 Graphs Are Built before December 12, 2006; the 2007 Graphs Are Built for January to February, 2007

Graph	# of Nodes	# of Edges	Avg. Deg	Avg. path	Avg. CC
2006 Original	624,364	8,258,266	26.45	4.16	0.159
2006 Synthetic	624,364	8,721,927	27.93	4.46	0.183
2007 Original	1,751,146	18,203,520	20.79	4.87	0.156
2007 Synthetic	1,751,146	18,305,972	20.9	4.84	0.161

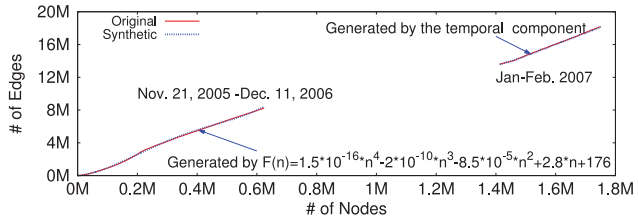


Fig. 25. Fitting of network growth with the network edge growth function $F(n)$ (Renren).

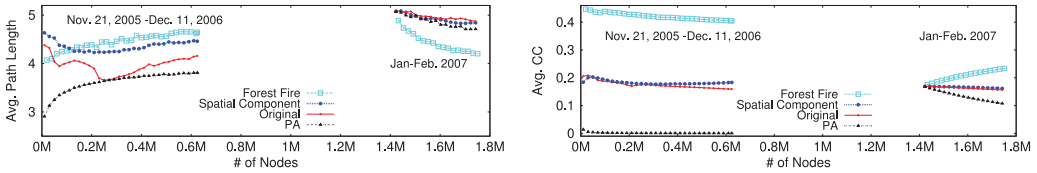
to manually fit the network growth by a polynomial function. This is because our measurement shows that in 2006 the network is not stable and large enough to display significant temporal patterns. Figure 25 shows the $F(n)$ estimation results for both segments, which closely match the original data.

Next, we follow the methodology by Sala et al. [2010] to determine p. We generate a series of synthetic graphs with p varying between $(0.1, 0.9)$ and choose the best p value that produces graphs with network distance and clustering coefficient most similar to the original data. The resulting p values differ for the two segments: 0.7 for the 2006 segment and 0.5 for the 2007 segment.

Validation Results. Using the calibrated component, we generate synthetic dynamic graphs for the two data segments. As shown in Table IV, the synthetic graphs statistically match the original graphs in the corresponding last snapshot, in terms of average degree, average path length, and average CC. The emphasis of our validation is to understand whether synthetic graphs display the three dynamic properties observed from the Renren social network [Zhao et al. 2012]: graph densification, average path length shrinkage, and decreasing clustering coefficient. Using the network growth function $F(n)$, Figure 25 confirms that the synthetic graphs can accurately capture the densification property. Thus, in the following, we focus on evaluating dynamics of average path length and average clustering coefficient in synthetic graphs. As a reference, we also include the results using the Preferential Attachment model [Barabási and Albert 1999], which is the most popular static graph model, and the Forest Fire model [Leskovec et al. 2005].[8] We repeated our experiments *5 times* for all three models and obtained consistent results, with the variance across all runs at least three orders of magnitude smaller than the average value. Thus, for brevity, we only show the result for a single run.

Average Path Length Evolution: Figure 26(a) plots the average path length over time using our spatial component, the Preferential Attachment model, the Forest Fire

[8]Following a similar procedure described by Sala et al. [2010], we modify the Forest Fire model to produce undirected graphs by creating undirected edges and allowing the "burning" process to proceed in both directions of an edge. To calibrate the model, that is, determining the burning probability p, we sample values between $(0, 1)$ to find the best fit p where the corresponding synthetic graphs match the original graph the most in terms of network distance and clustering coefficient.



(a) Avg. path length (b) Avg. clustering coefficient

Fig. 26. Graph dynamic properties on generated synthetic graphs and the original Renren graph. All include two time periods from the very beginning to December 11, 2006, and in January to February, 2007 (to avoid the one-time merge event in Renren with another OSN). (Original: Renren graph; Spatial Component: graph generated by our spatial component; PA: graph generated by the preferential attachment model; Forest Fire: graph generated by the Forest Fire model).

model, and the original data. For the 2006 segment, our spatial component displays the most similar pattern to the original data, where the path length decreases first and then increases slightly, while the Preferential Attachment and Forest Fire models produce increasing path length. For the 2007 segment, while all four graphs display a decreasing pattern over time, our spatial component is the closest to the original graph. In this segment, behaviors of the Preferential Attachment and Forest Fire models change because the snapshot of the original data on December 31, 2006, is used as the initial graph, removing the long-term impact of preferential attachment [Zhao et al. 2012] that produces increasing average path length over time.

Average Clustering Coefficient Evolution: Figure 26(b) plots the results for the average clustering coefficient from the three models and the original data. For the 2006 segment, only our spatial component behaves similarly to the original data, with an average clustering coefficient in (0.15, 0.22), while that of the Preferential Attachment model stays closely to 0, and the Forest Fire model remains above 0.4. For the 2007 segment, again our spatial component produces nearly identical value of the original data, while the results of the Preferential Attachment and Forest Fire model deviate largely. Together, these results confirm three key findings. *First*, our spatial component can accurately capture the significant local connectivity and the slowly decreasing clustering coefficient. *Second*, the Preferential Attachment model is unable to maintain high clustering coefficient over time, even when growing from a highly clustered graph. *Finally*, as indicated by our earlier analysis, the Forest Fire model produces relatively high clustering coefficient, unable to capture the key properties of Renren such as decreasing clustering coefficient.

Summary of Results. Our validation confirms that the spatial component can accurately capture key dynamic features observed in Renren dataset. Since our 2007 synthetic trace takes input from the temporal component of our model, the spatial component validation also provides an overall validation of our proposed model.

7.3. Facebook Results

We now briefly summarize how we validate our model using the Facebook dataset, since the methodology is very similar to what is applied to the Renren dataset. Like Renren, our results on Facebook datasets also strongly validate the effectiveness of our model.

First, we validate the temporal component by calibrating the self-similar module and the non-stationary module and then produce a synthetic edge creation trace (repeated 5 times). The total edge count matches the original data, that is, the average ratio between the original and synthetic traces, is 1.05 with variance $<10^{-5}$. The wavelet-based analysis on the synthetic traces leads to results consistent with that of the original trace. Specifically, for 3-day segments, H estimates with 95% CI fall between

Table V. Statistics of the Original Facebook Graph, the Synthetic Graph Generated by Our Spatial Component, by the Forest Fire Model, and by the Preferential Attachment Model

Graph	Avg. Degree	Avg. path length	Avg. CC	Final avg. Degree	Final avg. path length	Final CC	Final # of Edges
Facebook Original	32.293	5.760	0.101	37.357	5.630	0.108	876,993
Synthetic (our model)	32.508	3.650	0.122	37.545	3.645	0.118	881,415
Forest Fire	70.273	3.836	0.469	69.509	4.030	0.446	1,631,792
Preferential Attachment	35.960	2.890	0.012	35.986	2.996	0.006	844,812

Path length and CC do not consider multiple edges between node pairs. All standard deviations are less than 4%. Columns 2–4 refer to averaged results for intermediate graph snapshots, and columns 5–8 refer to the final graph snapshot (Facebook).

$(0.5, 1)$ with an exception of $<1\%$. The exception ratio (the portion of abnormal segments whose H estimates are of $(0.5,1)$) grows to 25% for 15-day segments and 100% for 20-day segments.

Second, the spatial component for Facebook differs slightly from that of Renren because Facebook wall posts can lead to multiple edges between a node pair (while Renren only has one per node pair). Thus we modify our model, as well as the Forest Fire and Preferential Attachment models, to allow duplicated edges. We grow the three models from 0 node to the total number of nodes $46,952$ in the Facebook dataset. We compare the synthetic traces generated by the three models and the original data (see Table V). Again, our results show that our model can accurately capture the growth of the Facebook trace. Its average node degree and clustering coefficient, for both intermediate and final snapshots, are almost identical to the original data, while the Forest Fire and Preferential Attachment models produce large deviations.

7.4. Summary

Our results on the Renren and Facebook datasets consistently show that our model can successfully capture both the temporal properties of graph dynamics, in terms of self-similar scaling and deterministic non-stationary trends in terms of periodic patterns, and its spatial properties observed, including long-term graph distance shrinkage and reduction in local clustering.

8. RELATED WORK

8.1. Self-Similarity Measurements and Models

Self-similarity describes the phenomenon where a property is preserved with respect to scaling in space and/or time. If an object is self-similar, then its parts, when magnified, resemble the shape of the whole [Park and Willinger 2000]. Previous works have studied *structural self-similarity* in networks [Guimera et al. 2003; Song et al. 2005], that is, the scale-invariance properties of physical structures of a graph (e.g., node degree or community size distribution) under coarse graining of vertices. Our work differs by studying self-similar scaling properties on time dynamics, that is, "temporal" self-similarity, which has not been studied in social networks.

Self-Similarity Measurements. Temporal self-similarity describes the scaling properties of certain statistics (e.g., variance, R/S, wavelet coefficients, finite-dimensional distributions) of a time series when computed at different time scales [Park and Willinger 2000]. It has been detected in diverse contexts such as ecology, life sciences, and stock markets [Eisler et al. 2008] and was first introduced to network traffic for the purpose of modeling the bursty characteristics observed in Ethernet LAN (local area network) traffic [Leland and Wilson 1991; Leland et al. 1994]. Later studies show self-similarity has also been observed in other network traffic scenarios,

including wide-area traffic [Paxson and Floyd 1995], world wide web traffic [Crovella and Bestavros 1997], disk-level I/O [Riska and Riedel 2006], HTTP traffic traces [Deng et al. 2012], variable-bit-rate video [Beran et al. 1995; Garrett and Willinger 1994], blog posts [Goetz et al. 2009], messages [Rybski et al. 2009], and emails [Eisler et al. 2008] in communication networks. Note that these empirical studies show that, in practice, self-similar property is typically observed over a finite range of time scales [Abry and Veitch 1998; Garrett and Willinger 1994; Gribble et al. 1998] and is difficult to discern at both very small and very large time scales.

Self-Similarity Models. Generally speaking, there are two classes of self-similar models. The first are purely mathematical models, for example, fractional Gaussian noise [Mandelbrot and van Ness 1968], fractional Brownian motion [Mandelbrot and van Ness 1968], fractional autoregressive integrated moving average (ARIMA) processes [Hosking 1981], and b-model [Wang et al. 2002]. They are strictly descriptive and cannot explain the root cause underlying the formation of self-similarity. The second class seeks to provide physical reasons behind self-similarity. Inspired by the renewal reward process in economics [Taqqu and Levy Taqqu and Levy], the superposition of many ON/OFF sources [Willinger et al. 1997; Gribble et al. 1998] captures the observed self-similar nature of Ethernet LAN traffic if the durations of the ON- or OFF-periods have a heavy-tailed distribution. The $M|G|\infty$ queuing model [Cox 1984; Paxson and Floyd 1995; Park and Willinger 2000], where sources arrive according to a Poisson process and each source is active for a duration that is described by a heavy-tailed distribution, can also successfully explain self-similar phenomena.

8.2. Graph Models

In general, graph models can be classified as static graph models or dynamic graph models.

Static Models. We further classify static models into three sets. One set includes feature-driven models designed to capture one or more static graph features, for example, small-world [Watts and Strogatz 1998], power-law degree distribution [Barabási and Albert 1999; Holme and Kim 2002], and high clustering coefficients [Holme and Kim 2002]. A second set includes intent-driven models that try to explain the underlying process of graph formation. Nearest-neighbor models [Vázquez 2003; Davidsen et al. 2002; Toivonen et al. 2006], random-walk models [Blum et al. 2006; Vázquez 2003], and copying models [Kumar et al. 2000; Vázquez 2003] belong to this set. Finally, a third set of models generates graphs based on graph structural statistics instead of graph features. Kronecker graphs [Leskovec and Faloutsos 2007] apply Kronecker multiplication to generate graphs similar to real graphs. The dK-series model [Mahadevan et al. 2006] uses subgraph degree distributions to capture increasingly detailed representations of graph structures. Finally, Sala et al. [2010] proposes a general technique to produce "realistic" synthetic graphs by calibrating graph models using real graphs.

Dynamic Models. In contrast, dynamic models aim to capture dynamic features of graphs. Barabâsi et al. [2002] modifies a preferential attachment model to capture graph densification. Leskovec et al. [2005] proposes a Forest Fire model to capture both graph densification and diameter shrinking properties in networks. Later models [McGlohon et al. 2008; Xia and Hu 2015] capture similar properties. The dynamic copying model captures the property of decreasing clustering coefficients but not the power-law degree distribution [Bonato et al. 2009]. Based on graph structure statistics, Akoglu et al. [2008] proposes a three-dimensional Kronecker model. Akoglu and Faloutsos [2009] is a model based on random typing statistics to capture several graph dynamic features. Unlike our work, Akoglu and Faloutsos [2009] is not modeled after

empirical data of graph dynamics. Leskovec et al. [2008] designs a model of network evolution but focuses on reproducing desired structural properties in the *final* snapshot. Finally, Navlakha et al. [2015] tries to include capturing the network harshness into the model, that is, how likely a node will be lost, but also cares about the final sturctural statistics only.

9. CONCLUSION

Starting from the exploration of self-similarity properties, which is critical in determining how to model network dynamics, our work takes a concrete step towards studying the detailed dynamics of social networks. We focus on "time-stamped" traces of network growth, that is, a network includes detailed timings of when nodes arrive and edges are created. By performing empirical studies of network dynamics over two detailed, time-stamped traces of social networks over multiple years, that is, the Renren dataset and Facebook wall post dataset, we have detected that the edge creation process in both networks does have properties consistent with self-similar scaling. We have also quantified that such properties hold from seconds to hours and gradually weaken at larger time scales due to the existence of non-stationary patterns introduced by human behaviors, for example, diurnal or weekly user activities.

Specifically, we find that the edge creation process in the two OSNs is non-stationary over long-term periods, and the two traditional techniques for self-similarity detection, that is, R/S and variance methods, produce inconclusive results and are unsuitable for measuring self-similarity in real traces in social networks. By applying the more robust wavelet-based method against underlying non-stationary trends, we find the edge creation process in both network traces does exhibit properties consistent with self-similarity over time scales ranging from seconds to hours.

We leverage this new result to propose a comprehensive model of graph dynamics, including a temporal component that defines when and how many new edges are formed across all the users, and a spatial component that defines where in the graph new edges form. Our temporal component captures the coexistence of long-term non-stationary periodic structure, for example, diurnal or weekly patterns, and properties consistent with self-similarity at shorter time scales, while our spatial component is a dynamic graph model that simulates edge creation process driven primarily by existing users and captures graph densification, shrinking network diameter, and decreasing local clustering.

Our detailed validation efforts on both datasets consistently show that our model accurately captures both the temporal properties of graph dynamics, in terms of self-similar scaling and certain deterministic non-stationary features, and also many key dynamic structural properties of the two social graphs over time.

By providing such a practical method for generating a realistic sequence of time-stamped events that uniquely define the formation and evolution of a social network in time and space, our model fills an existing void in network dynamics, and addresses an urgent need for accurate models that account for both temporal and spatial features in network dynamics.

ACKNOWLEDGMENTS

The authors thank the anonymous reviewers for their helpful comments.

REFERENCES

Patrice Abry and Darryl Veitch. 1998. Wavelet analysis of long-range-dependent traffic. *IEEE TOIT* 44, 1 (1998), 2–15.

L. Akoglu and C. Faloutsos. 2009. RTG: A recursive realistic graph generator using random typing. In *ECML PKDD*.

L. Akoglu, M. McGlohon, and C. Faloutsos. 2008. RTM: Laws and a recursive generator for weighted time-evolving graphs. In *Proceedings of ICDM*.

A. L. Barabási and R. Albert. 1999. Emergence of scaling in random networks. *Science* 286, 5439 (1999), 509–512.

A. L. Barabâsi, H. Jeong, Z. Néda, E. Ravasz, A. Schubert, and T. Vicsek. 2002. Evolution of the social network of scientific collaborations. *Physica A: Statistical Mechanics and Its Applications* 311, 3–4 (2002), 590–614.

J. Beran. 1994. *Statistics for Long-memory Processes*. Chapman & Hall/CRC.

Jan Beran, Robert Sherman, Murad S. Taqqu, and Walter Willinger. 1995. Long-range dependence in variable-bit-rate video traffic. *IEEE Transactions on Communications* 43, 2/3/4 (1995), 1566–1579.

Bin Bi and Junghoo Cho. 2016. Modeling a retweet network via an adaptive Bayesian approach. In *Proceedings of WWW*.

A. Blum, T. H. H. Chan, and M. R. Rwebangira. 2006. A random-surfer web-graph model. In *Proceedings of WAEE and WAAC*, Vol. 123.

Anthony Bonato, Noor Hadi, Paul Horn, Paweł Prałat, and Changping Wang. 2009. A dynamic model for on-line social networks. *Algorithms and Models for the Web-Graph* 5427 (2009), 127–142.

Yoon-Sik Cho, Greg Ver Steeg, Emilio Ferrara, and Aram Galstyan. 2016. Latent space model for multi-modal social data. In *Proceedings of WWW*.

D. Cox. 1984. Long range dependence: A review. In *Statistics: An Appraisal,* H. A. David and H. T. David (Eds.). Iowa State University Press, AMES, IA, 55–74.

Mark E. Crovella and Azer Bestavros. 1997. Self-similarity in world wide web traffic: Evidence and possible causes. *IEEE/ACM TON* (1997).

J. Davidsen, H. Ebel, and S. Bornholdt. 2002. Emergence of a small world from local interactions: Modeling acquaintance networks. *Physical Review Letters* 88, 12 (2002), 128701.

Pasquale De Meo, Emilio Ferrara, Giacomo Fiumara, and Alessandro Provetti. 2014. On facebook, most ties are weak. *Commun. ACM* 57, 11 (2014), 78–84.

Yuhui Deng, Xiaohua Meng, and Jipeng Zhou. 2012. Self-similarity: Behind workload reshaping and prediction. *Future Generation Computer Systems* 28, 2 (2012), 350–357.

Zoltán Eisler, Imre Bartos, and János Kertész. 2008. Fluctuation scaling in complex systems: Taylor's law and beyond 1. *Advances in Physics* 57, 1 (2008), 89–142.

Mark W. Garrett and Walter Willinger. 1994. Analysis, modeling and generation of self-similar VBR video traffic. In *Proceedings of SIGCOMM*.

Michaela Goetz, Jure Leskovec, Mary McGlohon, and Christos Faloutsos. 2009. Modeling blog dynamics. In *ICWSM*.

Steven D. Gribble, Gurmeet Singh Manku, Drew Roselli, Eric A. Brewer, Timothy J. Gibson, and Ethan L. Miller. 1998. Self-similarity in file systems. In *Proceedings of SIGMETRICS*.

Roger Guimera, Leon Danon, Albert Diaz-Guilera, Francesc Giralt, and Alex Arenas. 2003. Self-similar community structure in a network of human interactions. *Physical Review E* 68, 6 (2003), 065103.

P. Holme and B. J. Kim. 2002. Growing scale-free networks with tunable clustering. *Physical Review E* 65, 2 (2002), 026107.

Jonathan R. M. Hosking. 1981. Fractional differencing. *Biometrika* 68, 1 (1981), 165–176.

Jing Jiang, Christo Wilson, Xiao Wang, Peng Huang, Wenpeng Sha, Yafei Dai, and Ben Y. Zhao. 2010. Understanding latent interactions in online social networks. In *Proceedings of IMC*.

Thomas Karagiannis, Michalis Faloutsos, and Rudolf H. Riedi. 2002. Long-range dependence: Now you see it, now you don't! In *IEEE Globecom*.

David Karger and Matthias Ruhl. 2002. Find nearest neighbors in growth-restricted metrics. In *Proceedings of STOC*.

Ravi Kumar and others. 2000. Stochastic models for the web graph. In *Proceedings of FOCS*.

Jérôme Kunegis. 2013. Konect: The Koblenz network collection. In *Proceedings of WWW Companion*.

Will E. Leland, Murad S. Taqqu, Walter Willinger, and Daniel V. Wilson. 1994. On the self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Transactions on Networking* 2, 1 (1994), 1–15.

Will E. Leland and Daniel V. Wilson. 1991. High time-resolution measurement and analysis of LAN traffic: Implications for LAN interconnection. In *Proceedings of INFOCOM*.

J. Leskovec, L. Backstrom, R. Kumar, and A. Tomkins. 2008. Microscopic evolution of social networks. In *Proceedings of KDD*.

J. Leskovec and C. Faloutsos. 2007. Scalable modeling of real graphs using Kronecker multiplication. In *Proceedings of ICML*.

Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. 2005. Graphs over time: Densification laws, shrinking diameters and possible explanations. In *Proceedings of KDD*.

P. Mahadevan, D. Krioukov, K. Fall, and A. Vahdat. 2006. Systematic topology analysis and generation using degree correlations. In *Proceedings of SIGCOMM*.

B. B. Mandelbrot and J. W. van Ness. 1968. Fractional Brownian motions, fractional noises and applications. *SIAM Rev.* 10 (1968), 422–437.

Mary McGlohon, Leman Akoglu, and Christos Faloutsos. 2008. Weighted graphs and disconnected components: Patterns and a generator. In *Proceedings of KDD*.

Saket Navlakha, Christos Faloutsos, and Ziv Bar-Joseph. 2015. MassExodus: Modeling evolving networks in harsh environments. *Data Min. Knowl. Discov.* 29, 5 (2015), 1211–1232.

Kihong Park and Walter Willinger. 2000. *Self-similar Network Traffic and Performance Evaluation*. Wiley Online Library.

Vern Paxson and Sally Floyd. 1995. Wide area traffic: the failure of Poisson modeling. *IEEE/ACM Transactions on Networking (ToN)* 3, 3 (1995), 226–244.

Alma Riska and Erik Riedel. 2006. Long-range dependence at the disk drive level. In *Proceedings of QEST*.

Diego Rybski, Sergey V. Buldyrev, Shlomo Havlin, Fredrik Liljeros, and Hernán A. Makse. 2009. Scaling laws of human interaction activity. *Proc. Natl. Acad. Sci. U.S.A.* 106, 31 (2009), 12640–12645.

Alessandra Sala, Lili Cao, Christo Wilson, Robert Zablit, Haitao Zheng, and Ben Y. Zhao. 2010. Measurement-calibrated graph models for social network experiments. In *Proceedings of WWW*.

Purnamrita Sarkar, Deepayan Chakrabarti, and Michael Jordan. 2012. Nonparametric link prediction in dynamic networks. In *Proceedings of ICML*.

Fabian Schneider, Anja Feldmann, Balachander Krishnamurthy, and Walter Willinger. 2009. Understanding online social network usage from a network perspective. In *Proceedings of IMC*.

Chaoming Song, Shlomo Havlin, and Hernan A. Makse. 2005. Self-similarity of complex networks. *Nature* 433, 7024 (2005), 392–395.

M. S. Taqqu and J. Levy. Using renewal processes to generate long-range dependence and high variability. *Depend. Probab. Stat.*, 73–89.

Murad S. Taqqu, Vadim Teverovsky, and Walter Willinger. 1995. Estimators for long-range dependence: An empirical study. *Fractals* 3, 4 (1995), 785–798.

Riitta Toivonen and others. 2006. A model for social networks. *Physica A* 371, 2 (2006), 851–860.

A. Vázquez. 2003. Growing network with local rules: Preferential attachment, clustering hierarchy, and degree correlations. *Phys. Rev. E* 67, 5 (2003), 056104.

Mengzhi Wang, Tara Madhyastha, Ngai Hang Chan, Spiros Papadimitriou, and Christos Faloutsos. 2002. Data mining meets performance evaluation: Fast algorithms for modeling bursty traffic. In *Proceedings of ICDE*.

D. J. Watts and S. H. Strogatz. 1998. Collective dynamics of 'small-world' networks. *Nature* 393, 6684 (1998), 440–442.

Walter Willinger, Vern Paxson, and Murad S. Taqqu. 1998. Self-similarity and heavy tails: Structural modeling of network traffic. *A Practical Guide to Heavy Tails: Statistical Techniques and Applications* 23 (1998), 27–53.

Walter Willinger, Murad S. Taqqu, Robert Sherman, and Daniel V. Wilson. 1997. Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level. *IEEE/ACM Transactions on Networking (ToN)* 5, 1 (1997), 71–86.

Christo Wilson, Bryce Boe, Alessandra Sala, Krishna P. N. Puttaswamy, and Ben Y. Zhao. 2009. User interactions in social networks and their implications. In *Proceedings of EuroSys*.

Hongke Xia and Xiang Hu. 2015. FBM: A flexible random walk based generative model for social network. *Open Cybernetics & Systemics Journal* 9, 1 (2015), 280–287.

Xiaohan Zhao, Alessandra Sala, Christo Wilson, Xiao Wang, Sabrina Gaito, Haitao Zheng, and Ben Y. Zhao. 2012. Multi-scale dynamics in a massive online social network. In *Proceedings of IMC*.

Received January 2016; revised May 2016; accepted August 2016