Hypersparse Neural Network Analysis of Large-Scale Internet Traffic

Jeremy Kepner[1], Kenjiro Cho[2], KC Claffy[3], Vijay Gadepally[1], Peter Michaleas[1], Lauren Milechin[4]
[1]MIT Lincoln Laboratory Supercomputing Center, [2]Research Laboratory, Internet Initiative Japan, Inc.,
[3]UCSD Center for Applied Internet Data Analysis, [4]MIT Dept. of Earth, Atmospheric, & Planetary Sciences

***Abstract*—The Internet is transforming our society, necessitating a quantitative understanding of Internet traffic. Our team collects and curates the largest publicly available Internet traffic data containing 50 billion packets. Utilizing a novel hypersparse neural network analysis of "video" streams of this traffic using 10,000 processors in the MIT SuperCloud reveals a new phenomena: the importance of otherwise unseen leaf nodes and isolated links in Internet traffic. Our neural network approach further shows that a two-parameter modified Zipf-Mandelbrot distribution accurately describes a wide variety of source/destination statistics on moving sample windows ranging from 100,000 to 100,000,000 packets over collections that span years and continents. The inferred model parameters distinguish different network streams and the model leaf parameter strongly correlates with the fraction of the traffic in different underlying network topologies. The hypersparse neural network pipeline is highly adaptable and different network statistics and training models can be incorporated with simple changes to the image filter functions.**

***Index Terms*—Internet modeling, packet capture, neural networks, power-law networks, hypersparse matrices**

I. Introduction

Our civilization is now dependent on the Internet, necessitating a scientific understanding of this virtual universe [1], [2], that is made more urgent by the rising influence of adversarial Internet robots (botnets) on society [3], [4]. The two largest efforts to capture, curate, and share Internet packet traffic data for scientific analysis are led by our team via the Widely Integrated Distributed Environment (WIDE) project [5] and the Center for Applied Internet Data Analysis (CAIDA) [6]. These data have supported a variety of research projects resulting in hundreds of peer-reviewed publications [7], ranging from characterizing the global state of Internet traffic, to specific studies of the prevalence of peer-to-peer filesharing, to testing prototype software designed to stop the spread of Internet worms.

The stochastic network structure of Internet traffic is a core property of great interest to Internet stakeholders [2] and network scientists [8]. Of particular interest is the probability distribution $p(d)$ where d is the degree (or count) of one of several network quantities depicted in Figure 1: source packets, source fan-out, packets over a unique source-destination pair (or link), destination fan-in, and destination packets. Amongst the earliest and most widely cited results of virtual Internet topology analysis has been the observation of the power-law relationship

$$p(d) \propto 1/d^{\alpha} \tag{1}$$

with a model exponent $1 < \alpha < 3$ for large values of d [9]–[11]. [Note: in our work network topology refers to the graph theoretic virtual topology of sources and destinations and not the underlying physical topology of the Internet.] These early observations demonstrated the importance of a few supernodes in the Internet (see Figure 2) [12]. Measurements of power-laws in Internet data stimulated investigations into a wide range of network phenomena in many domains and lay the

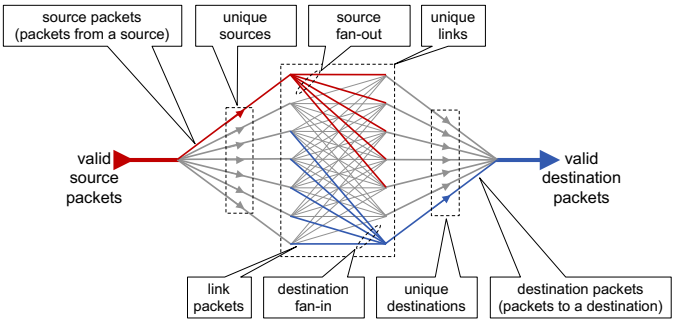

Fig. 1. **Streaming network traffic quantities.** Internet traffic streams of N_V valid packets are divided into a variety of quantities for analysis: source packets, source fan-out, unique source-destination pair packets (or links), destination fan-in, and destination packets.

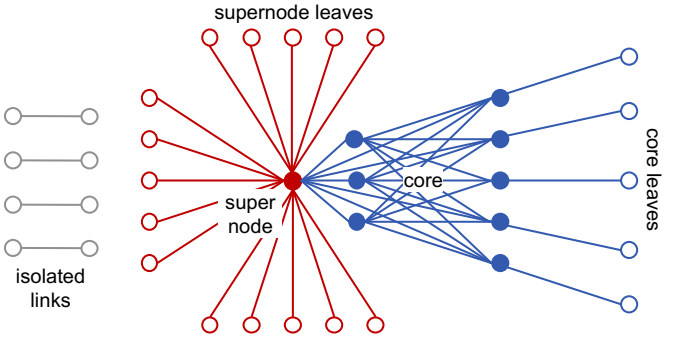

Fig. 2. **Traffic network topologies.** Internet traffic forms networks consisting of a variety of topologies: isolated links, supernode leaves connected to a supernode, densely connected core(s) with corresponding core leaves.

This material is based in part upon work supported by the NSF under grants DMS-1312831, CCF-1533644, and CNS-1513283, DHS cooperative agreement FA8750-18-2-0049, and ASD(R&E) under contract FA8702-15-D-0001. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF, DHS, or ASD(R&E).

foundation for the field of network science [8].

Classification of Internet phenomena is often based on data obtained from crawling the network from a number of starting points [13]. These webcrawls naturally sample the supernodes of the network [12] and their resulting $p(d)$ are accurately fit at large values of d by single-parameter power-law models. Unfortunately, these models have impractically large deviations for other values of d (see [14] figures 8H/9W/9X, [15] figure 4B, [16] figure 3A, and [17] figure 21) and are not usable for modeling Internet traffic in real-world settings. Characterizing a network by a single power-law exponent provides one view of Internet phenomena, but more accurate and complex models are required to understand the diverse topologies seen in streaming samples of the Internet.

Improving Internet model accuracy while also increasing model complexity requires overcoming a number of challenges, including acquisition of larger, rigorously collected data sets [18], [19]; the enormous computational cost of processing large network traffic graphs [20]–[22]; careful filtering, binning, and normalization of the data; and inferring nonlinear models to the data [8], [14]. This paper presents approaches for overcoming these challenges to improved model accuracy by employing a novel hypersparse neural network analysis of "video" stream representations of Internet traffic. Furthermore, the hypersparse neural network pipeline is highly adaptable and different network statistics and training models can be incorporated with simple changes to the image filter functions.

II. Streaming Internet Data

The two largest efforts to capture, curate, and share Internet packet traffic data for scientific analysis are led by our team via the WIDE and CAIDA efforts. This paper analyzes, for the first time, the very largest collections in our corpora containing 49.6 billion packets (see Table I).

A. MAWI Internet Traffic Collection

The WIDE project is a research consortium in Japan established in 1988 [5]. The members of the project include network engineers, researchers, university students, and industrial partners. The focus of WIDE is on the empirical study of the large-scale internet. WIDE operates an internet testbed for both commercial traffic and for conducting research experiments. These data have enabled quantitative analysis of Internet traffic spanning years illustrating trends such as, the emergence of residential usage, peer-to-peer networks, probe scanning, and botnets [23]–[25]. The Tokyo datasets are publicly available packet traces provided by the WIDE project (a.k.a. the MAWI traces). The traces are collected from a 1 Gbps academic backbone connection in Japan. The 2015 and 2017 datasets are 48-hour-long traces captured during December 2-3 2015 and April 12-13 2017 in JST. The IP addresses appearing in the traces are anonymized using a prefix-preserving method [26].

B. CAIDA Internet Traffic Collection

CAIDA collects several different data types at geographically and topologically diverse locations, and makes this data available to the research community to the extent possible while preserving the privacy of individuals and organizations who donate data or network access [6] [27]. CAIDA has (and had) monitoring locations in Internet Service Providers (ISPs) in the United States. CAIDA's passive traces dataset contains traces collected from high-speed monitors on a commercial backbone link. The data collection started in April 2008 and is ongoing. These data are useful for research on the characteristics of Internet traffic, including application breakdown (based on TCP/IP ports), security events, geographic and topological distribution, flow volume and duration. For an overview of all traces see the trace statistics page [28]. Collectively, our consortium has enabled scientific analysis of Internet traffic resulting in hundreds of peer-reviewed publications with over 30,000 citations [7].

TABLE I
PACKET CAPTURE DATA.

Large-scale network traffic packet data sets containing 49.6 billion packets collected at different locations, times, and durations over two years. All source data can be found at the websites https://mawi.wide.ad.jp (/mawi/ditl/ditl2015/ and /mawi/ditl/ditl2017/) and https://www.caida.org (/datasets/passive-2016/equinix-chicago/). This work used the CAIDA UCSD Anonymized Internet Traces - 2016 January 21, February 18, March 17, and April 06.

Location	Date	Duration	Bandwidth	Packets
Tokyo	2015 Dec 02	2 days	10^9 bits/sec	17.0×10^9
Tokyo	2017 Apr 12	2 days	10^9 bits/sec	16.8×10^9
Chicago A	2016 Jan 21	1 hour	10^{10} bits/sec	2.0×10^9
Chicago A	2016 Feb 18	1 hour	10^{10} bits/sec	2.0×10^9
Chicago A	2016 Mar 17	1 hour	10^{10} bits/sec	1.8×10^9
Chicago A	2016 Apr 06	1 hour	10^{10} bits/sec	1.8×10^9
Chicago B	2016 Jan 21	1 hour	10^{10} bits/sec	2.3×10^9
Chicago B	2016 Feb 18	1 hour	10^{10} bits/sec	1.7×10^9
Chicago B	2016 Mar 17	1 hour	10^{10} bits/sec	2.0×10^9
Chicago B	2016 Apr 06	1 hour	10^{10} bits/sec	2.1×10^9

The traffic traces used in this paper are anonymized using CryptoPan prefix-preserving anonymization. The anonymization key changes annually and is the same for all traces recorded during the same calendar year. During capture packets are truncated at a snap length selected to avoid excessive packet loss due to disk I/O overload. The snap length has historically varied from 64 to 96 bytes. In addition, payload is removed from all packets: only header information up to layer 4 (transport layer) remains. Endace network cards used to record these traces provide timestamps with nanosecond precision. However, the anonymized traces are stored in pcap format with timestamps truncated to microseconds. Starting with the 2010 traces the original nanosecond timestamps are provided as separate ascii files alongside the packet capture files.

III. Approach

This work overcomes obstacles to improved model accuracy by employing a novel hypersparse neural network analysis of "video" stream representations of the Internet traffic (Figure 3). Utilizing recent innovations in interactive supercomputing [29], [30], matrix-based graph theory [31],

978-1-7281-5020-8/19/$31.00 ©2019 IEEE

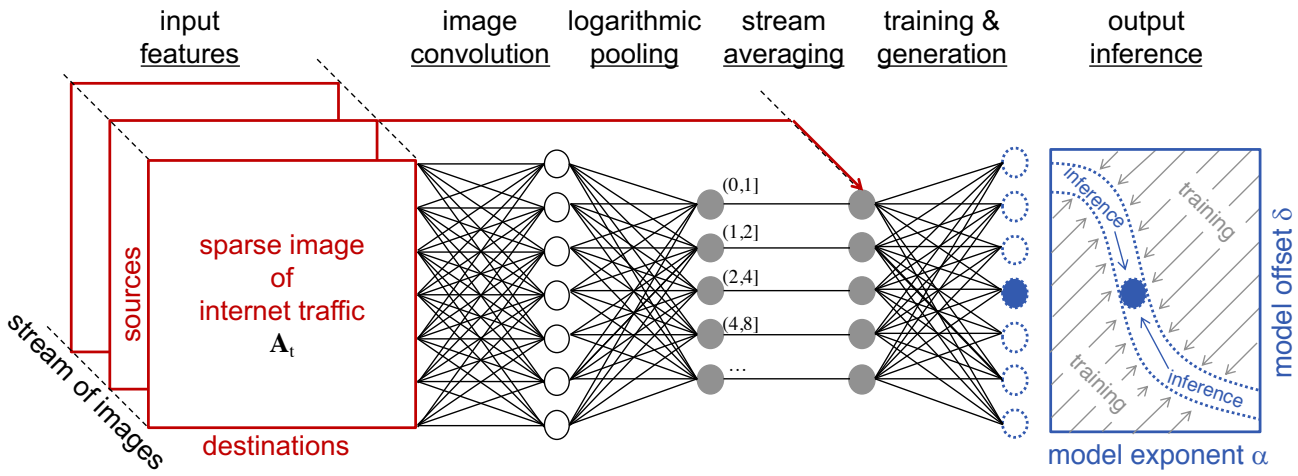

Fig. 3. **Hypersparse neural network pipeline.** Traffic streams are turned into hypersparse images (stored as associative arrays). Network quantities are extracted from the images via convolution with different filters. The resulting network quantities are logarithmically pooled (binned) and averaged over the streaming "video" of the images. The averaged data is used to train and then generate network model weights that are used to infer the classification of the network within the model parameter space.

TABLE II
AGGREGATE NETWORK PROPERTIES

Formulas for computing aggregates from a sparse network image $\mathbf{A}_t$ at time t in both summation and matrix notation. $\mathbf{1}$ is a column vector of all 1's, $^{\mathsf{T}}$ is the transpose operation, and $|\ |_0$ is the zero-norm that sets each nonzero value of its argument to 1 [35].

Aggregate Property	Summation Notation	Matrix Notation
Valid packets N_V	$\sum_i \sum_j \mathbf{A}_t(i,j)$	$\mathbf{1}^{\mathsf{T}}\mathbf{A}_t\mathbf{1}$
Unique links	$\sum_i \sum_j \lvert\mathbf{A}_t(i,j)\rvert_0$	$\mathbf{1}^{\mathsf{T}}\lvert\mathbf{A}_t\rvert_0\mathbf{1}$
Unique sources	$\sum_i \lvert\sum_j \mathbf{A}_t(i,j)\rvert_0$	$\mathbf{1}^{\mathsf{T}}\lvert\mathbf{A}_t\mathbf{1}\rvert_0$
Unique destinations	$\sum_j \lvert\sum_i \mathbf{A}_t(i,j)\rvert_0$	$\lvert\mathbf{1}^{\mathsf{T}}\mathbf{A}_t\rvert_0\mathbf{1}$

TABLE III
NEURAL NETWORK IMAGE CONVOLUTION FILTERS

Different network quantities are extracted from a sparse traffic image $\mathbf{A}_t$ at time t via convolution with different filters. Formulas for the filters are given in both summation and matrix notation. $\mathbf{1}$ is a column vector of all 1's, $^{\mathsf{T}}$ is the transpose operation, and $|\ |_0$ is the zero-norm that sets each nonzero value of its argument to 1 [35].

Network Quantity	Summation Notation	Matrix Notation
Source packets from i	$\sum_j \mathbf{A}_t(i,j)$	$\mathbf{A}_t\ \mathbf{1}$
Source fan-out from i	$\sum_j \lvert\mathbf{A}_t(i,j)\rvert_0$	$\lvert\mathbf{A}_t\rvert_0\mathbf{1}$
Link packets from i to j	$\mathbf{A}_t(i,j)$	$\mathbf{A}_t$
Destination fan-in to j	$\sum_i \lvert\mathbf{A}_t(i,j)\rvert_0$	$\mathbf{1}^{\mathsf{T}}\ \mathbf{A}_t$
Destination packets to j	$\sum_i \mathbf{A}_t(i,j)$	$\mathbf{1}^{\mathsf{T}}\lvert\mathbf{A}_t\rvert_0$

[32], and big data mathematics [33], we have developed a scalable neural network Internet traffic processing pipeline that runs efficiently on more than 10,000 processors in the MIT SuperCloud [34]. This neural network pipeline allows us, for the first time, to process our largest traffic collections as network traffic graphs.

The hypersparse neural network pipeline depicted in Figure 3 begins with the construction of sparse images of network traffic data. These images are then convolved with a filter corresponding to the specific network quantity being analyzed: source packets, source fan-out, links, destination fan-in, and destination packets. The resulting network quantities are logarithmically pooled (binned) and averaged over the streaming "video" of the images. The averaged data is used to train and then generate network model weights that are used to infer the classification of the network within the model parameter space.

A. Image Convolution

Origin-destination traffic matrices or images are one of the most generally useful representations of Internet traffic [18], [22]. These matrices can be used to compute a wide range of network statistics useful in the analysis, monitoring, and control of the Internet. Such analysis include the temporal fluctuations of the supernodes [18] and inferring the presence of unobserved traffic [19] [36]. Not all packets have both source and destination Internet protocol version 4 (IPv4) addresses. To reduce statistical fluctuations the streaming data have been partitioned so that for any chosen time window all data sets have the same number of valid IPv4 packets. At a given time t, N_V consecutive valid packets are aggregated from the traffic into a sparse matrix $\mathbf{A}_t$, where $\mathbf{A}_t(i,j)$ is the number of valid packets between the source i and destination j [37]. The sum of all the entries in $\mathbf{A}_t$ is equal to N_V

$$\sum_{i,j} \mathbf{A}_t(i,j) = N_V \tag{2}$$

All the network quantities depicted in Figure 1 can be readily computed from $\mathbf{A}_t$ using the formulas listed in Table II and Table III.

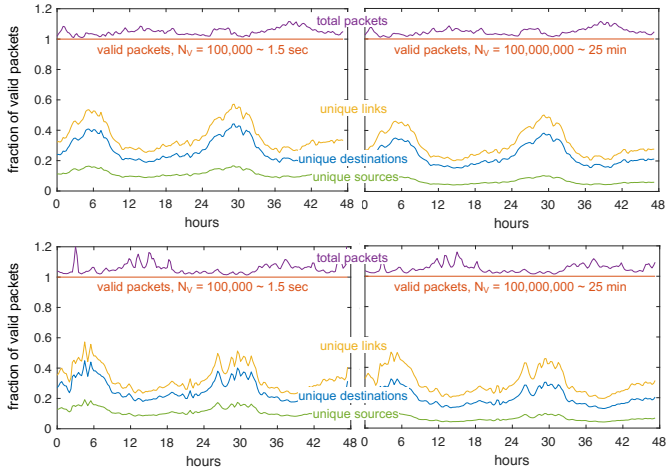

Fig. 4. **Valid packets.** Analyzing packet windows with the same numbers of valid packets produces consistent fractions of the aggregate numbers of unique links, unique destinations, and unique sources over a wide range of packet sizes for the Tokyo 2015 (top) and Tokyo 2017 (bottom) data sets. The plots show these fractions for moving packet windows of with N_V = 100,000 packets (left) and N_V = 100,000,000 packets (right). The packet windows correspond to time windows of approximately 1.5 seconds and 25 minutes.

An essential step for increasing the accuracy of the statistical measures of Internet traffic is using windows with the same number of valid packets N_V. For this analysis, a valid packet is defined as TCP over IPv4, which includes more than 95% of the data in the collection and eliminates a small amount of data that uses other protocols or contains anomalies. Using packet windows with the same number of valid packets produces aggregates that are consistent over a wide range from $N_V = 100{,}000$ to $N_V = 100{,}000{,}000$ (Figure 4).

B. Logarithmic Pooling

A network quantity d is computed via convolution with the image $\mathbf{A}_t$ using a filter selected from Table III. The corresponding histogram of the network quantity is denoted by $n_t(d)$, with corresponding probability

$$p_t(d) = n_t(d) / \sum_d n_t(d) \tag{3}$$

and cumulative probability

$$P_t(d) = \sum_{i=1,d} p_t(d) \tag{4}$$

Because of the relatively large values of d observed due to a single supernode, the measured probability at large d often exhibits large fluctuations. However, the cumulative probability lacks sufficient detail to see variations around specific values of d, so it is typical to pool the differential cumulative probability with logarithmic bins in d

$$D_t(d_i) = P_t(d_i) - P_t(d_{i-1}) \tag{5}$$

where $d_i = 2^i$ [14]. All computed probability distributions use the same binary logarithmic pooling (binning) to allow for consistent statistical comparison across data sets (Eq. 4) [8], [14]. The corresponding mean and standard deviation of $D_t(d_i)$ over many different consecutive values of t for a given data set are denoted $D(d_i)$ and $\sigma(d_i)$.

C. Modified Zipf-Mandelbrot Model

Measurements of $D(d_i)$ can reveal many properties of network traffic, such as the number of nodes with only one connection $D(d = 1)$ and the size of the supernode

$$d_{\max} = \text{argmax}(D(d) > 0) \tag{6}$$

Effective classification of a network with a low parameter model allows these and many other properties to be summarized and computed efficiently. In the standard Zipf-Mandelbrot model typically used in linguistic contexts, d is a ranking with $d = 1$ corresponding to the most popular value [38]–[40]. To accurately classify the network data using the full range of d, the Zipf-Mandelbrot model is modified so that d is a measured network quantity instead of a rank index

$$p(d; \alpha, \delta) \propto 1/(d + \delta)^\alpha \tag{7}$$

The inclusion of a second model offset parameter δ allows the model to accurately fit small values of d, in particular $d = 1$, which has the highest observed probability in these streaming data. The model exponent α has a larger impact on the model at large values of d while the model offset δ has a larger impact on the model at small values of d and in particular at $d = 1$.

The unnormalized modified Zipf-Mandelbrot model is denoted

$$\rho(d; \alpha, \delta) = \frac{1}{(d + \delta)^\alpha} \tag{8}$$

with correspond gradient

$$\partial_\delta \rho(d; \alpha, \delta) = \frac{-\alpha}{(d + \delta)^{\alpha+1}} = -\alpha \rho(d; \alpha + 1, \delta) \tag{9}$$

The normalized model probability is given by

$$p(d; \alpha, \delta) = \frac{\rho(d; \alpha, \delta)}{\sum_{d=1}^{d_{\max}} \rho(d; \alpha, \delta)} \tag{10}$$

where d_{max} is the largest value of the network quantity d. The cumulative model probability is the sum

$$P(d_i; \alpha, \delta) = \sum_{d=1}^{d_i} p(d; \alpha, \delta) \tag{11}$$

The corresponding differential cumulative model probability is

$$D(d_i; \alpha, \delta) = P(d_i; \alpha, \delta) - P(d_{i-1}; \alpha, \delta) \tag{12}$$

where $d_i = 2^i$.

978-1-7281-5020-8/19/$31.00 ©2019 IEEE

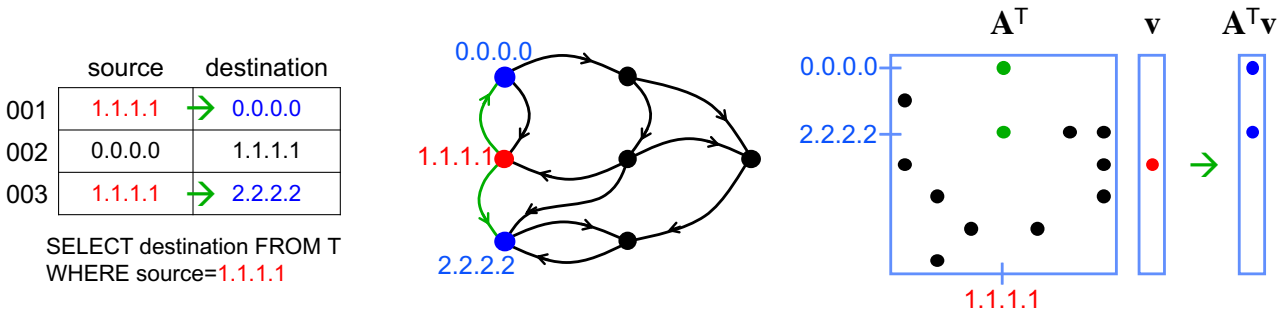

Fig. 5. **Associative Arrays.** Hypersparse network data are naturally represented as associative arrays that uniquely label each row and and column. Associative arrays further allow the data to be readily manipulated using relational database operations, graph algorithms, and matrix mathematics. (left) Tabular representation of raw network traffic and corresponding database query to find all records beginning with source 1.1.1.1. (middle) Network graph highlighting nearest neighbors of source node 1.1.1.1. (right) Corresponding associative array representation of the network graph illustrating how the neighbors of source node 1.1.1.1 are computed with matrix vector multiplication.

D. Training and Weight Generation

Classifying the logarithmically pooled data in terms of model parameters α and δ begins with training a set of candidate model weights that can then be used to infer the model parameters. Initially, a set of candidate exponent values is selected, typically $\alpha = 0.10, 0.11, \ldots, 3.99, 4.00$. For each value of α, a value of δ is trained that exactly matches the model with the data at $D(1)$. Training the value of δ corresponding to a give $D(1)$ is done using the gradient based Newton's method as follows. Setting the measured value of $D(1)$ equal to the model value $D(1;\alpha,\delta)$ gives

$$D(1) = D(1;\alpha,\delta) = \frac{1}{(1+\delta)^{\alpha}\sum_{d=1}^{d_{\max}}\rho(d;\alpha,\delta)} \tag{13}$$

Newton's method works on functions of the form $f(\delta) = 0$. Rewriting the above expression produces

$$f(\delta) = D(1)(1+\delta)^{\alpha}\sum_{d=1}^{d_{\max}}\rho(d;\alpha,\delta) - 1 = 0 \tag{14}$$

For given value of α, δ can be trained using the following iterative gradient based equation

$$\delta \rightarrow \delta - \frac{f(\delta)}{\partial_{\delta} f(\delta)} \tag{15}$$

where the gradient is

$$\partial_{\delta} f(\delta) = \alpha D(1)(1+\delta)^{\alpha} \left[(1+\delta)^{-1}\sum_{d=1}^{d_{\max}}\rho(d;\alpha,\delta) - \sum_{d=1}^{d_{\max}}\rho(d;\alpha+1,\delta)\right] \tag{16}$$

Using a starting value of $\delta = 1$ and bounds of $0 < \delta < 10$, Newton's method can be iterated until the differences in successive values of δ fall below a specified error (typically 0.001) and is usually achieved in less than five iterations.

E. Parameter Inference

The inferred α (and corresponding δ) is chosen by minimizing the $|\ |^{1/2}$ metric over logarithmic differences between the weights of candidate models $D(d_i;\alpha,\delta)$ and the data

$$\text{argmin}_{\alpha}\sum_{d_i}|\log(D(d_i)) - \log(D(d_i;\alpha,\delta))|^{1/2} \tag{17}$$

The $|\ |^{1/2}$ metric (or $|\ |_p$-norm with $p = 1/2$) favors maximizing error sparsity over minimizing outliers [41]–[43] [35], [44]–[46]. Several authors have shown recently that it is possible to reconstruct a nearly sparse signal from fewer linear measurements than would be expected from traditional sampling theory. Furthermore, by replacing the $|\ |_1$ norm with the $|\ |^p$ with $p < 1$, reconstruction is possible with substantially fewer measurements.

Using logarithmic values more evenly weights their contribution to the inferred model and more accurately reflects the number of packets used to compute each value of $D(d_i)$. Lower accuracy data points are avoided by limiting the training and inference procedure to data points where the value is greater than the standard deviation: $D(d_i) > \sigma(d_i)$.

F. Memory and Computation Requirements

Processing 49.6 billion Internet packets with a variety of algorithms presents numerous computational challenges. Dividing the data set into combinable units of approximately 100,000 consecutive packets made the analysis amenable to processing on a massively parallel supercomputer. The detailed architecture of the parallel processing system and its corresponding performance are described in [34]. The resulting processing pipeline was able to efficiently use over 10,000 processors on the MIT SuperCloud and was essential to this first-ever complete analysis of these data.

A key element of our analysis is the use of novel hypersparse matrix mathematics in concert with the MIT SuperCloud to process very large network traffic matrices (Figure 5). Construction and analysis of network traffic matrices of the entire Internet address space have been considered impractical

978-1-7281-5020-8/19/$31.00 ©2019 IEEE

for its massive size [22]. Internet Protocol version 4 (IPv4) has 2^{32} unique addresses, but at any given collection point only a fraction of these addresses will be observed. Exploiting this property to save memory can be accomplished by extending traditional sparse matrices so that new rows and columns can be added dynamically. The algebra of associative arrays [33] and its corresponding implementation in the Dynamic Distributed Dimensional Data Model (D4M) software library (d4m.mit.edu) allows the row and columns of a sparse matrix to be any sortable value, in this case character string representations of the Internet addresses (Figure 5). Associative arrays extend sparse matrices to have database table properties with dynamically insertable and removable rows and columns that adjust as new data is added or subtracted to the matrix. Using these properties, the memory requirements of forming network traffic matrices can be reduced at the cost of increasing the required computation necessary to resort the rows and columns.

A hypersparse associative array representing an image of traffic $\mathbf{A}_t$ with $N_V = 100{,}000{,}000$ typically requires 2 Gigabytes of memory. Complete analysis of the statistics and topologies of $\mathbf{A}_t$ typically takes 10 minutes on a single MIT SuperCloud Intel Knights Landing processor core. Using increments of 100,000 packets means that this analysis is repeated over 500,000 times to process all 49.6 billion packets. Using 10,000 processors on the MIT SuperCloud shortens the run time of these analysis to approximately eight hours. The results presented here are the product of an interactive discovery process that required hundreds of such runs that would not have been possible without the MIT SuperCloud. Fortunately, the utilization of these results by Internet stakeholders can be significantly accelerated by creating optimized embedded hypersparse neural network implementations that only compute the desired statistics and are not required to support an interactive analytics discovery process [47], [48].

IV. Results

A. Daily Variations

Diurnal variations in supernode network traffic are well known [18]. The Tokyo packet data were collected over a period spanning two days, and allow the daily variations in packet traffic to be observed. The precision and accuracy of our measurements allows these variations to be observed across a wide range of nodes. Figure 6 shows the fraction of source fan-outs in each of various bin ranges. The fluctuations show the network evolving between two envelopes occurring between noon and midnight that are shown in Figure 7.

B. Inferred Modified Zipf-Mandelbrot Distributions

Figure 8A shows five representative inferred models out of the 350 performed on 10 datasets, 5 network quantities, and 7 valid packet windows: $N_V = 10^5$, $3{\times}10^5$, 10^6, $3{\times}10^6$, 10^7, $3{\times}10^7$, 10^8. The inferred models are valid over the entire range of d and provide parameter estimates with precisions of 0.01. In every case, the high value of $p(d = 1)$ is indicative of a large contribution from a combination of supernode

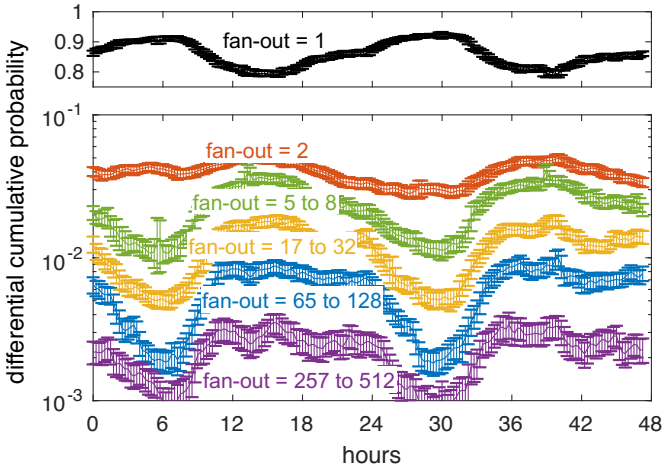

Fig. 6. **Daily variation in Internet traffic.** The fraction of source nodes with a given range of fan-out are shown as a function of time for the Tokyo 2015 data. The $p(d = 1)$ value is plotted on a separate linear scale because of the larger magnitude relative to the other points. Each point is the mean of many neighboring points in time and the error bars are the measured ± 1-σ. The daily variation of the distributions oscillate between extremes corresponding to approximately local noon and midnight.

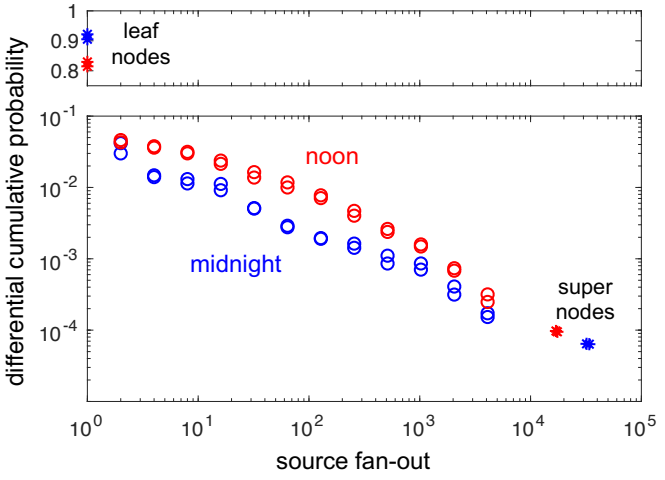

Fig. 7. **Daily limits in Internet traffic.** The fraction of source nodes versus fan-out are shown for two noons and two midnights for the Tokyo 2015 data. The overlap among the noons and the midnights shows the relative day-to-day consistency in these data and show the limits of the two extremes in daily variation. During the day, there is more traffic among nodes with intermediate fan-out. At night the traffic is more dominated by leaf nodes and the supernode.

leaves, core leaves, and isolated links (Figure 2). The breadth and accuracy of these data allow detailed comparison of the inferred models. Figure 1B shows the inferred model offset δ versus the model exponent α for all 350 fits. The different collection locations are clearly distinguishable in this model parameter space. The Tokyo collections have smaller offsets and are more tightly clustered than the Chicago collections. Chicago B has a consistently smaller source and link packet model offset than Chicago A. All the collections have source, link, and destination packet model exponents in the relatively

978-1-7281-5020-8/19/$31.00 ©2019 IEEE

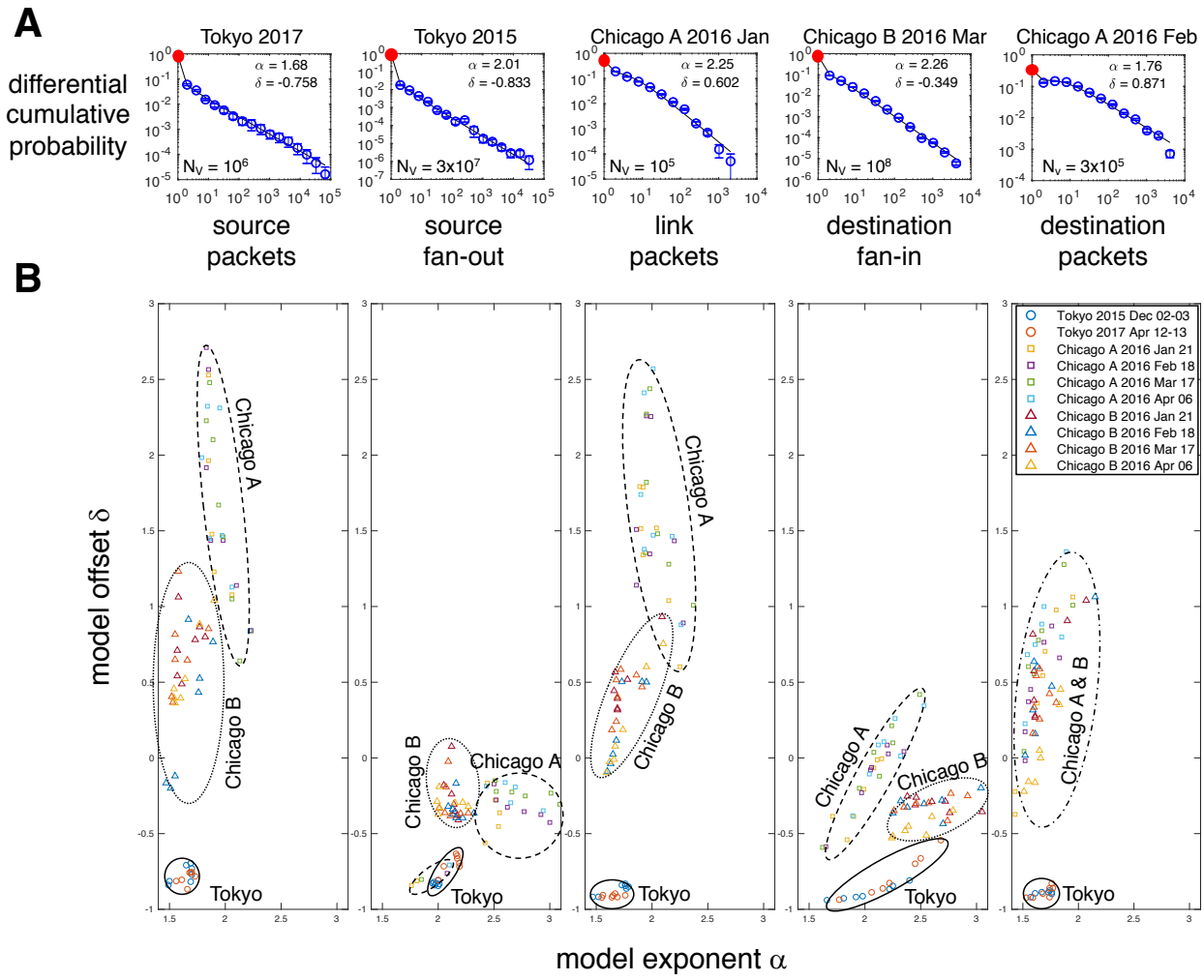

Fig. 8. **Measured network traffic distributions and inferred models.** (A) A selection of 5 of the 350 measured differential cumulative probabilities spanning different locations, dates, and packet windows. Blue circles are measured data with ± 1-σ error bars. Black lines are the best-fit modified Zipf-Mandelbrot models with parameters α and δ. Red dots highlight the large contribution of leaf nodes and isolated links. (B) Inferred model parameters for all 350 measured probability distributions reveal the underlying structural differences among the data collected in Tokyo, Chicago A, and Chicago B.

narrow $1.5 < \alpha < 2$ range. The source fan-out and destination fan-in model exponents are in the broader $1.5 < \alpha < 2.5$ range and are consistent with the prior literature [14]. These results represent an entirely new approach to characterizing Internet traffic that allows the data to be projected into a low-dimensional space and enables accurate comparisons among packet collections with different locations, dates, durations, and sizes.

C. Measured Network Topologies

Figure 2 depicts the major topological structures in the network traffic: isolated links, supernode leaves, core, and core leaves. Formulas for computing these topologies from $\mathbf{A}_t$ are given in Appendix A. Figure 9 shows the average relative fractions of sources, total packets, total links, and number of destinations in each of the five topologies for the ten data sets, and seven valid packet windows: $N_V = 10^5$, $3{\times}10^5$, 10^6, $3{\times}10^6$, 10^7, $3{\times}10^7$, 10^8. The four projections in Figure 9 were chosen to highlight the differences in the collection locations. The distinct regions in the various projections shown in Figure 9 indicate that underlying topological differences are present in the data. The Tokyo collections have much larger supernode leaf components than the Chicago collections. The Chicago collections have much larger core and core leaves components than the Tokyo collections. Chicago A consistently has fewer isolated links than Chicago B. Comparing the modified Zipf-Mandelbrot model parameters in Figure 8B and underlying topologies in Figure 9 suggests that the inferred model parameters are a more compact way to distinguish the network traffic.

Figures 8B and 9 indicate that different collection points produce different inferred model parameters α and δ, and that these collection points also have different underlying topologies. Figure 10 connects the inferred models and topology

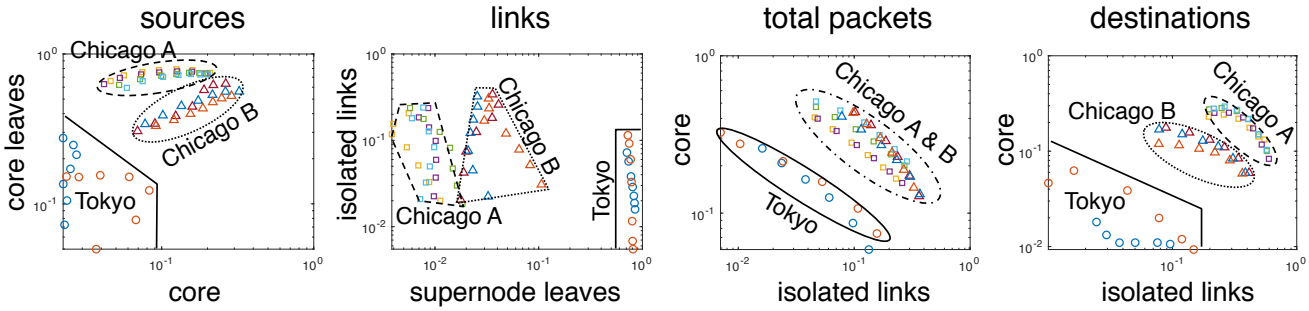

Fig. 9. **Distribution of traffic among network topologies.** A selection of four projections showing the fraction of data in various underlying topologies using the same legend as Figure 8B. Horizontal and vertical axis are the corresponding fraction of the sources, links, total packets and destinations that are in various topologies for each location, time, and seven packet windows ($N_V = 10^5, \ldots, 10^8$). These data reveal the differences in the network traffic topologies in the data collected in Tokyo (dominated by supernode leaves), Chicago A (dominated by core leaves), and Chicago B (between Tokyo and Chicago A).

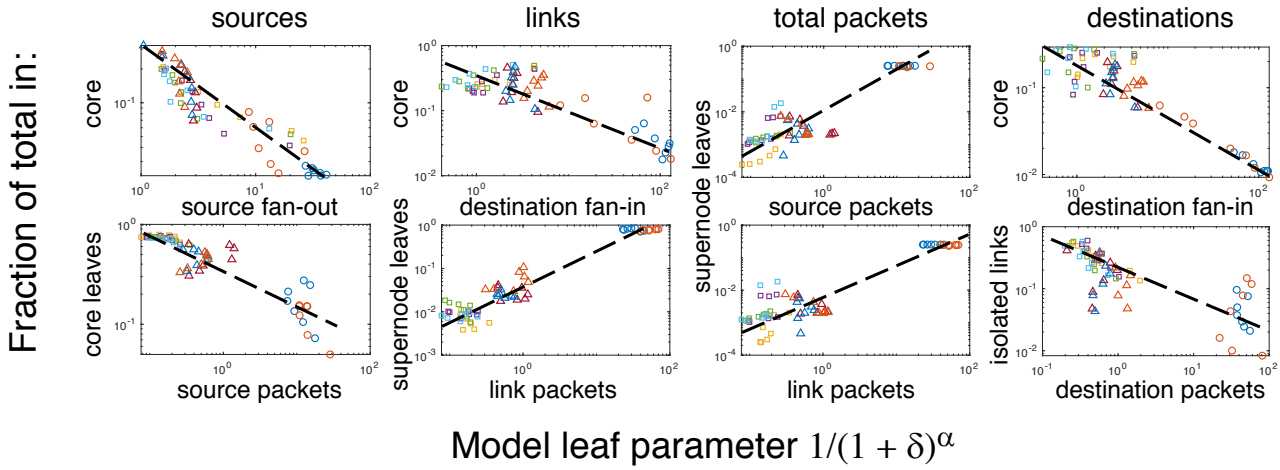

Fig. 10. **Topology versus model leaf parameter.** Network topology is highly correlated with the modified Zipf-Mandelbrot model leaf parameter $1/(1+\delta)^{\alpha}$. A selection of eight projections showing the fraction of sources, links, total packets, and destinations in various underlying topologies using the same legend as Figure 8B. Vertical axis are the corresponding fraction of the sources, links, total packets and destinations that are in various topologies. Horizontal axis is the value of the model parameter taken from either the source packet, source fan-out, link packet, destination fan-in and destination packet fits. Data points are for each location, time, and seven packet windows ($N_V = 10^5, \ldots, 10^8$).

observations by plotting the topology fraction as a function of the model leaf parameter $1/(1+\delta)^{\alpha}$ which corresponds to the relative strength of leaf nodes and isolated links $p(d=1)$

$$1/(1+\delta)^{\alpha} \propto p(d=1;\alpha,\delta) \tag{18}$$

The correlations revealed in Figure 10 suggest that the model leaf parameter strongly correlates with the fraction of the traffic in different underlying network topologies and is a potentially new and beneficial way to characterize networks. Figure 10 indicates that the fraction of sources, links, and destinations in the core shrinks as the relative importance of the leaf parameter in the source fan-out and destination fan-in increases. In other words, more source and destination leaves means a smaller core. Likewise, the fraction of links and total packets in the supernode leaves grows as the leaf parameter in the link packets and source packets increases. Interestingly, the fraction of sources in the core leaves and isolated links decreases as the leaf parameter in the source and destination packets increases indicating a shift of sources away from the core leaves and isolated links into supernode leaves. Thus, the modified Zipf-Mandelbrot model and its leaf parameter provide a direct connection with the network topology, underscoring the value of having accurate model fits across the entire range of values and in particular for $d = 1$.

V. Conclusion

Measurements of internet traffic are useful for informing policy, identifying and preventing outages, defeating attacks, planning for future loads, and protecting the domain name system [49]. On a given day, millions of IPs are engaged in scanning behavior. Our improved models can aid cybersecurity analysts in determining which of these IPs are nefarious [50], the distribution of attacks in particular critical sectors [51], identifying spamming behavior [52], how to vacinate against computer viruses [53], obscuring web sources [54], identifying

significant flow aggregates in traffic [55], and sources of rumors [56].

The results presented here have a number of potential practical applications for Internet stakeholders. The methods presented of collecting, filtering, computing, and binning the data to produce accurate measurements of a variety of network quantities are generally applicable to Internet measurement and have the potential to produce more accurate measures of these quantities. The accurate fits of the two parameter modified Zipf-Mandelbrot distribution offer all the usual benefits of low parameter models: measuring parameters with far less data, accurate predictions of network quantities based on a few parameters, observing changes in the underlying distribution, and using modeled distributions to detect anomalies in the data.

From a scientific perspective, improved knowledge of how Internet traffic flows can inform our understanding of how economics, topology, and demand shape the Internet over time. As with all scientific disciplines, the ability of theoreticians to develop and test theories of the Internet and network phenomena is bounded by the scale and accuracy of measured phenomena [57]–[60]. In contrast to previous network models that have principally been based on data obtained from network crawls from a variety of start points on the network, our network traffic data are collected from observations of network streams. Both viewpoints provide important network observations. Observations of a network stream provide complementary data on network dynamics and highlight the contribution of leaves and isolated edges, which are less sampled in network crawls.

The aggregated data set our teams have collected provide a unique window into these questions. The hypersparse neural network pipeline is a novel approach for inferring power-law models and have potential applications to power-law networks in diverse domains. The inferred model parameters present new opportunities to connect the distributions to underlying theoretical models of networks. That the inferred model parameters distinguish the different collection points and are reflective of different network topologies in the data at these points suggests a deeper underlying connection between the models and the network topologies.

ACKNOWLEDGMENT

The authors wish to acknowledge the following individuals for their contributions and support: Shohei Araki, William Arcand, David Bestor, William Bergeron, Bob Bond, Paul Burkhardt, Chansup Byun, Cary Conrad, Alan Edelman, Sterling Foster, Bo Hu, Matthew Hubbell, Micheal Houle, Micheal Jones, Anne Klein, Charles Leiserson, Dave Martinez, Mimi McClure, Julie Mullen, Steve Pritchard, Andrew Prout, Albert Reuther, Antonio Rosa, Victor Roytburd, Siddharth Samsi, Koichi Suzuki, Kenji Takahashi, Michael Wright, Charles Yee, and Michitoshi Yoshida.

REFERENCES

[1] M. Hilbert and P. López, "The world's technological capacity to store, communicate, and compute information," *Science*, p. 1200970, 2011.

[2] B. Li, J. Springer, G. Bebis, and M. H. Gunes, "A survey of network flow applications," *Journal of Network and Computer Applications*, vol. 36, no. 2, pp. 567–581, 2013.

[3] H. Allcott and M. Gentzkow, "Social media and fake news in the 2016 election," *Journal of Economic Perspectives*, vol. 31, no. 2, pp. 211–36, 2017.

[4] "https://www.neosit.com/files/neos_distil_bad_bot_report_2018.pdf."

[5] K. Cho, K. Mitsuya, and A. Kato, "Traffic data repository at the wide project," in *Proceedings of USENIX 2000 Annual Technical Conference: FREENIX Track*, pp. 263–270, 2000.

[6] K. Claffy, "Internet tomography," *Nature, Web Matter*, 1999.

[7] "http://www.caida.org/data/publications/."

[8] A.-L. Barabási *et al.*, *Network science*. Cambridge university press, 2016.

[9] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[10] R. Albert, H. Jeong, and A.-L. Barabási, "Internet: Diameter of the world-wide web," *Nature*, vol. 401, no. 6749, p. 130, 1999.

[11] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graphs over time: densification laws, shrinking diameters and possible explanations," in *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pp. 177–187, ACM, 2005.

[12] J. Cao, Y. Jin, A. Chen, T. Bu, and Z.-L. Zhang, "Identifying high cardinality internet hosts," in *INFOCOM 2009, IEEE*, pp. 810–818, IEEE, 2009.

[13] C. Olston, M. Najork, *et al.*, "Web crawling," *Foundations and Trends® in Information Retrieval*, vol. 4, no. 3, pp. 175–246, 2010.

[14] A. Clauset, C. R. Shalizi, and M. E. Newman, "Power-law distributions in empirical data," *SIAM review*, vol. 51, no. 4, pp. 661–703, 2009.

[15] A. Mahanti, N. Carlsson, A. Mahanti, M. Arlitt, and C. Williamson, "A tale of the tails: Power-laws in internet measurements," *IEEE Network*, vol. 27, no. 1, pp. 59–64, 2013.

[16] M. Kitsak, A. Elmokashfi, S. Havlin, and D. Krioukov, "Long-range correlations and memory in the dynamics of internet interdomain routing," *PloS one*, vol. 10, no. 11, p. e0141481, 2015.

[17] M. Lischke and B. Fabian, "Analyzing the bitcoin network: The first four years," *Future Internet*, vol. 8, no. 1, p. 7, 2016.

[18] A. Soule, A. Nucci, R. Cruz, E. Leonardi, and N. Taft, "How to identify and estimate the largest traffic matrix elements in a dynamic environment," in *ACM SIGMETRICS Performance Evaluation Review*, vol. 32, pp. 73–84, ACM, 2004.

[19] Y. Zhang, M. Roughan, C. Lund, and D. L. Donoho, "Estimating point-to-point and point-to-multipoint traffic matrices: an information-theoretic approach," *IEEE/ACM Transactions on Networking (TON)*, vol. 13, no. 5, pp. 947–960, 2005.

[20] A. Lumsdaine, D. Gregor, B. Hendrickson, and J. Berry, "Challenges in parallel graph processing," *Parallel Processing Letters*, vol. 17, no. 01, pp. 5–20, 2007.

[21] D. A. Bader, H. Meyerhenke, P. Sanders, and D. Wagner, *Graph partitioning and graph clustering*, vol. 588. American Mathematical Soc., 2013.

[22] P. Tune, M. Roughan, H. Haddadi, and O. Bonaventure, "Internet traffic matrices: A primer," *Recent Advances in Networking*, vol. 1, pp. 1–56, 2013.

[23] K. Cho, K. Fukuda, H. Esaki, and A. Kato, "The impact and implications of the growth in residential user-to-user traffic," in *ACM SIGCOMM Computer Communication Review*, vol. 36, pp. 207–218, ACM, 2006.

[24] P. Borgnat, G. Dewaele, K. Fukuda, P. Abry, and K. Cho, "Seven years and one day: Sketching the evolution of internet traffic," in *INFOCOM 2009, IEEE*, pp. 711–719, IEEE, 2009.

[25] R. Fontugne, P. Abry, K. Fukuda, D. Veitch, K. Cho, P. Borgnat, and H. Wendt, "Scaling in internet traffic: a 14 year and 3 day longitudinal study, with multiscale analyses and random projections," *IEEE/ACM Transactions on Networking (TON)*, vol. 25, no. 4, pp. 2152–2165, 2017.

[26] J. Fan, J. Xu, M. H. Ammar, and S. B. Moon, "Prefix-preserving ip address anonymization: measurement-based security evaluation and a new cryptography-based scheme," *Computer Networks*, vol. 46, no. 2, pp. 253–272, 2004.

[27] K. Claffy, "Measuring the internet," *IEEE Internet Computing*, vol. 4, no. 1, pp. 73–75, 2000.

[28] "http://www.caida.org/data/passive/trace_stats/."

[29] J. Kepner, *Parallel MATLAB for Multicore and Multinode Computers*. SIAM, 2009.

978-1-7281-5020-8/19/$31.00 ©2019 IEEE

[30] A. Reuther, J. Kepner, C. Byun, S. Samsi, W. Arcand, D. Bestor, B. Bergeron, V. Gadepally, M. Houle, M. Hubbell, *et al.*, "Interactive supercomputing on 40,000 cores for machine learning and data analysis," *IEEE High Performance Extreme Computing Conference (HPEC)*, 2018.
[31] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.
[32] J. Kepner and J. Gilbert, *Graph algorithms in the language of linear algebra*. SIAM, 2011.
[33] J. Kepner and H. Jananthan, *Mathematics of big data: Spreadsheets, databases, matrices, and graphs*. MIT Press, 2018.
[34] V. Gadepally, J. Kepner, L. Milechin, W. Arcand, D. Bestor, B. Bergeron, C. Byun, M. Hubbell, M. Houle, M. Jones, *et al.*, "Hyperscaling internet graph analysis with d4m on the mit supercloud," *IEEE High Performance Extreme Computing Conference (HPEC)*, 2018.
[35] J. Karvanen and A. Cichocki, "Measuring sparseness of noisy signals," in *4th International Symposium on Independent Component Analysis and Blind Signal Separation*, pp. 125–130, 2003.
[36] V. Bharti, P. Kankar, L. Setia, G. Gürsun, A. Lakhina, and M. Crovella, "Inferring invisible traffic," in *Proceedings of the 6th International Conference*, p. 22, ACM, 2010.
[37] P. J. Mucha, T. Richardson, K. Macon, M. A. Porter, and J.-P. Onnela, "Community structure in time-dependent, multiscale, and multiplex networks," *science*, vol. 328, no. 5980, pp. 876–878, 2010.
[38] B. Mandelbrot, "An informational theory of the statistical structure of language," *Communication theory*, vol. 84, pp. 486–502, 1953.
[39] M. A. Montemurro, "Beyond the zipf–mandelbrot law in quantitative linguistics," *Physica A: Statistical Mechanics and its Applications*, vol. 300, no. 3-4, pp. 567–578, 2001.
[40] O. Saleh and M. Hefeeda, "Modeling and caching of peer-to-peer traffic," in *Network Protocols, 2006. ICNP'06. Proceedings of the 2006 14th IEEE International Conference on*, pp. 249–258, IEEE, 2006.
[41] D. L. Donoho, "Compressed sensing," *IEEE Transactions on information theory*, vol. 52, no. 4, pp. 1289–1306, 2006.
[42] R. Chartrand, "Exact reconstruction of sparse signals via nonconvex minimization," *IEEE Signal Process. Lett.*, vol. 14, no. 10, pp. 707–710, 2007.
[43] Z. Xu, X. Chang, F. Xu, and H. Zhang, "$l\_\{1/2\}$ regularization: A thresholding representation theory and a fast solver," *IEEE Transactions on neural networks and learning systems*, vol. 23, no. 7, pp. 1013–1027, 2012.
[44] N. Saito, B. M. Larson, and B. Bénichou, "Sparsity vs. statistical independence from a best-basis viewpoint," in *Wavelet Applications in Signal and Image Processing VIII*, vol. 4119, pp. 474–487, International Society for Optics and Photonics, 2000.
[45] M. Brbic and I. Kopriva, "ℓ_0-motivated low-rank sparse subspace clustering," *IEEE Transactions on Cybernetics*, pp. 1–15, 2018.
[46] H. D. Yaghoub Rahimi, Chao Wang and Y. Lous, "A scale invariant approach for sparse signal recovery," *arXiv preprint arXiv:1812.08852*, 2018.
[47] A. X. Liu, C. R. Meiners, and E. Torng, "Tcam razor: A systematic approach towards minimizing packet classifiers in tcams," *IEEE/ACM Transactions on Networking (TON)*, vol. 18, no. 2, pp. 490–500, 2010.
[48] A. X. Liu, C. R. Meiners, and E. Torng, "Packet classification using binary content addressable memory," *IEEE/ACM Transactions on Networking*, vol. 24, no. 3, pp. 1295–1307, 2016.
[49] D. Clark *et al.*, "The 9th workshop on active internet measurements (aims-9) report," *ACM SIGCOMM Computer Communication Review*, vol. 47, no. 5, pp. 35–38, 2017.
[50] S. Yu, G. Zhao, W. Dou, and S. James, "Predicted packet padding for anonymous web browsing against traffic analysis attacks," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 4, pp. 1381–1393, 2012.
[51] M. Husák, N. Neshenko, M. S. Pour, E. Bou-Harb, and P. Čeleda, "Assessing internet-wide cyber situational awareness of critical sectors," in *Proceedings of the 13th International Conference on Availability, Reliability and Security*, ARES 2018, (New York, NY, USA), pp. 29:1–29:6, ACM, 2018.
[52] O. Fonseca, E. Fazzion, I. Cunha, P. H. B. Las-Casas, D. Guedes, W. Meira, C. Hoepers, K. Steding-Jessen, and M. H. Chaves, "Measuring, characterizing, and avoiding spam traffic costs," *IEEE Internet Computing*, vol. 20, no. 4, pp. 16–24, 2016.
[53] J. Balthrop, S. Forrest, M. E. Newman, and M. M. Williamson, "Technological networks and the spread of computer viruses," *Science*, vol. 304, no. 5670, pp. 527–529, 2004.
[54] M. Javed, C. Herley, M. Peinado, and V. Paxson, "Measurement and analysis of traffic exchange services," in *Proceedings of the 2015 Internet Measurement Conference*, pp. 1–12, ACM, 2015.
[55] K. Cho, "Recursive lattice search: Hierarchical heavy hitters revisited," in *Proceedings of the 2017 Internet Measurement Conference*, pp. 283–289, ACM, 2017.
[56] R. Paluch, X. Lu, K. Suchecki, B. K. Szymański, and J. A. Hołyst, "Fast and accurate detection of spread source in large complex networks," *Scientific reports*, vol. 8, no. 1, p. 2508, 2018.
[57] L. A. Adamic and B. A. Huberman, "Power-law distribution of the world wide web," *science*, vol. 287, no. 5461, pp. 2115–2115, 2000.
[58] T. Bohman, "Emergence of connectivity in networks," *evolution*, vol. 11, p. 13, 2009.
[59] M. P. Stumpf and M. A. Porter, "Critical truths about power laws," *Science*, vol. 335, no. 6069, pp. 665–666, 2012.
[60] Y. Virkar and A. Clauset, "Power-law distributions in binned empirical data," *The Annals of Applied Statistics*, pp. 89–119, 2014.

APPENDIX A
NETWORK TOPOLOGY MEASURES

Figure 2 depicts the major topological structures in the network traffic. Identification of these topologies and computation of their network statistics can all be obtained from the packet traffic counts aggregated into the sparse matrix $\mathbf{A}_t$. Two important network quantities for computing these network topologies are the source fan-out column vector

$$\mathbf{d}_{\text{out}} = |\mathbf{A}_t|_0 \mathbf{1} \tag{A1}$$

and the destination fan-in row vector

$$\mathbf{d}_{\text{in}} = \mathbf{1}^{\mathsf{T}} |\mathbf{A}_t|_0 \tag{A2}$$

A. Isolated Links

Isolated links are sources and destinations that each have only one connection. The set of sources that send to only one destination are

$$i_1 = \arg(\mathbf{d}_{\text{out}} = 1) \tag{A3}$$

The set of destinations that receive from only one destination are

$$j_1 = \arg(\mathbf{d}_{\text{in}} = 1) \tag{A4}$$

The isolated links can be found via

$$\mathbf{A}_t(i_1, j_1) \tag{A5}$$

The number of isolated link sources are

$$\mathbf{1}^{\mathsf{T}} |\mathbf{A}_t(i_1, j_1)\mathbf{1}|_0 \tag{A6}$$

The number of packets traversing isolated links is given by

$$\mathbf{1}^{\mathsf{T}} \mathbf{A}_t(i_1, j_1)\mathbf{1} \tag{A7}$$

The number of unique isolated links can be computed from

$$\mathbf{1}^{\mathsf{T}} |\mathbf{A}_t(i_1, j_1)|_0 \mathbf{1} \tag{A8}$$

The number of isolated link destinations are

$$|\mathbf{1}^{\mathsf{T}} \mathbf{A}_t(i_1, j_1)|_0 \mathbf{1} \tag{A9}$$

By definition, the number of isolated sources, the number of isolated links, and the number of isolated destinations are all the same value.

978-1-7281-5020-8/19/$31.00 ©2019 IEEE

B. Supernodes

The first, second, third, ... supernode is the source or destination with the first, second, third, ... most links. The identity of the first supernode is given by

$$k_{\max} = \text{argmax}(\mathbf{d}_{\text{out}} + \mathbf{d}_{\text{in}}) \tag{A10}$$

The leaves of a supernode are those sources and destinations whose only connection is to the supernode. The supernode source leaves can be found via

$$\mathbf{A}_t(i_1, k_{\max}) \tag{A11}$$

The supernode destination leaves can be found via

$$\mathbf{A}_t(k_{\max}, j_1) \tag{A12}$$

The number of supernode leaf sources are

$$\mathbf{1}^{\mathsf{T}} |\mathbf{A}_t(i_1, k_{\max})\mathbf{1}|_0 \tag{A13}$$

The number of packets traversing supernode leaves is given by

$$\mathbf{1}^{\mathsf{T}} \mathbf{A}_t(i_1, k_{\max}) + \mathbf{A}_t(k_{\max}, j_1)\mathbf{1} \tag{A14}$$

The number of unique supernode leaf links can be computed from

$$\mathbf{1}^{\mathsf{T}} |\mathbf{A}_t(i_1, k_{\max})|_0 + |\mathbf{A}_t(k_{\max}, j_1)|_0 \mathbf{1} \tag{A15}$$

The number of supernode leaf destinations are

$$|\mathbf{1}^{\mathsf{T}} \mathbf{A}_t(k_{\max}, j_1)|_0 \mathbf{1} \tag{A16}$$

Subsequent supernodes can be computed by eliminating the prior supernode and repeating the above calculations.

C. Core

The core of a network can be defined in a variety of ways. In this work, the network core is meant to convey the concept of a collection of sources and destinations that are not isolated and are multiply connected. The core is defined as the collection of sources and destinations whereby every source and destination has more than one connection. In this context, the core does not include the first five supernodes although only the first supernode is significant, and whether or not the other supernodes are included has minimal impact on the core calculations for these data. The set of sources that send to more than one destination, excluding the supernode(s), is

$$i_{\text{core}} = \arg(1 < \mathbf{d}_{\text{out}} < \mathbf{d}_{\text{out}}(k_{\max})) \tag{A17}$$

The set of destinations that receive from more than one source, excluding the supernode(s), is

$$j_{\text{core}} = \arg(1 < \mathbf{d}_{\text{in}} < \mathbf{d}_{\text{in}}(k_{\max})) \tag{A18}$$

The core links can be found via

$$\mathbf{A}_t(i_{\text{core}}, j_{\text{core}}) \tag{A19}$$

The number of core sources is

$$\mathbf{1}^{\mathsf{T}} |\mathbf{A}_t(i_{\text{core}}, j_{\text{core}})\mathbf{1}|_0 \tag{A20}$$

The number of core packets is given by

$$\mathbf{1}^{\mathsf{T}} \mathbf{A}_t(i_{\text{core}}, j_{\text{core}})\mathbf{1} \tag{A21}$$

The number of unique core links can be computed from

$$\mathbf{1}^{\mathsf{T}} |\mathbf{A}_t(i_{\text{core}}, j_{\text{core}})|_0 \mathbf{1} \tag{A22}$$

The number of core destinations is

$$|\mathbf{1}^{\mathsf{T}} \mathbf{A}_t(i_{\text{core}}, j_{\text{core}})|_0 \mathbf{1} \tag{A23}$$

D. Core Leaves

The core leaves are sources and destinations that have one connection to a core source or destination. The core source leaves can be found via

$$\mathbf{A}_t(i_1, k_{\text{core}}) \tag{A24}$$

The core destination leaves can be found via

$$\mathbf{A}_t(k_{\text{core}}, j_1) \tag{A25}$$

The number of core leaf sources is

$$\mathbf{1}^{\mathsf{T}} |\mathbf{A}_t(i_1, k_{\text{core}})\mathbf{1}|_0 \tag{A26}$$

The number of core leaf packets is given by

$$\mathbf{1}^{\mathsf{T}} \mathbf{A}_t(i_1, k_{\text{core}}) + \mathbf{A}_t(k_{\text{core}}, j_1)\mathbf{1} \tag{A27}$$

The number of unique core leaf links can be computed from

$$\mathbf{1}^{\mathsf{T}} |\mathbf{A}_t(i_1, k_{\text{core}})|_0 + |\mathbf{A}_t(k_{\text{core}}, j_1)|_0 \mathbf{1} \tag{A28}$$

The number of core leaf destinations is

$$|\mathbf{1}^{\mathsf{T}} \mathbf{A}_t(k_{\text{core}}, j_1)|_0 \mathbf{1} \tag{A29}$$

978-1-7281-5020-8/19/$31.00 ©2019 IEEE