This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2017.2724301, IEEE Transactions on Network and Service Management

Vertex Entropy as a Critical Node Measure in Network Monitoring

Phil Tee
Moogsoft Inc
1265 Battery Street, San Francisco, CA 94111
phil@moogsoft.com
George Parisis and Ian Wakeman
School of Engineering and Informatics, University of Sussex
Brighton, UK
{g.parisis, i.j.wakeman}@sussex.ac.uk

***Abstract*—Understanding which node failures in a network have more impact is an important problem. Current understanding, motivated by the scale free models of network growth, places emphasis on the degree of the node. This is not a satisfactory measure; the number of connections a node has does not capture how redundantly it is connected into the whole network. Conversely, the structural entropy of a graph captures the resilience of a network well, but is expensive to compute, and, being a global measure, does not attribute any specific value to a given node. This lack of locality prevents the use of global measures as a way of identifying critical nodes. In this paper we introduce local vertex measures of entropy which do not suffer from such drawbacks. In our theoretical analysis we establish the possibility that our local vertex measures approximate global entropy, with the advantage of locality and ease of computation. We establish properties that vertex entropy must have in order to be useful for identifying critical nodes. We have access to a proprietary event, topology and incident dataset from a large commercial network. Using this dataset, we demonstrate a strong correlation between vertex entropy and incident generation over events.**

***Index Terms*—Computer Network Management, Network topology, Network theory, Graph Theory, Entropy.**

I. Introduction and Related Work

Network fault management is principally concerned with the analysis of notifications or events (log messages, SNMP traps etc.) from network devices, with the goal of identifying failures in critical nodes before service is impacted. Events often occur at a very high rate, ranging from 10^2 to 10^6 events per second (eps). In most cases they do not directly indicate a problem. To illustrate, at a typical large enterprise network[1] the event rate is 135 million events a day, whereas there are just a few hundred 'actionable incidents'. The reason for this disparity between the volume of events, and the number of incidents is the over-instrumentation of monitored systems, and the tendency to collect every event for post incident analysis, in case a cause is missed. It is important to state that this heavy event load can render current algorithms used to surface important events unusable, and in many cases operational networks rely upon users reporting failures.

For the purposes of this work we define an event and an incident as follows:

- *Event*: An event is typically a single log message or notification from an underlying monitoring system. We require that it has a timestamp, topology node identifier and description. It is not necessarily a notification of a fault condition, but fault conditions will generally send out at least one event.
- *Incident*: An incident is a support ticket raised as a result of receiving an event, and each incident references a topology node from which the event was received. Although not all incidents are indicative of a significant impact, they are an indication that the node has a fault condition that requires investigation. Typically an incident ticket is raised manually by a support person, or automatically from a monitoring system. Incidents can reference one or more events.

In this paper, we base our analysis on a very large real world network delivering global internet services. More specifically, we have access to the following data[2]:

- **Topology**. The topology is a combination of automatically discovered and manually created datasets. It is normally an example of a Multiplex network, as described in [1]. The analysis presented in this paper is for a network of $225,239$ nodes.
- **Events**. Gathered from the same network is a collection of network events that were monitored over a period of several weeks. For the topology above we analyze $96,325,275$ events.
- **Incidents**. For the same period in which the events were collected, this resulted in $37,099$ such incidents being raised, which in turn refer back to the source event.

Identifying which events are the cause of actual outages is called Root Cause Analysis (RCA) [2]. Many algorithms are used to perform RCA (for a detailed example see [3]), but *scalability limitations* make applying these algorithms to the full event stream impractical. In many cases the maximum event throughput of such algorithms is of the order of 10^2 to

[1]This work is motivated by the experience gained deploying network management software at large commercial scale.

[2]The source of the data is currently confidential, but we are working towards permission to release this dataset with appropriate anonymisation.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2017.2724301, IEEE Transactions on Network and Service Management

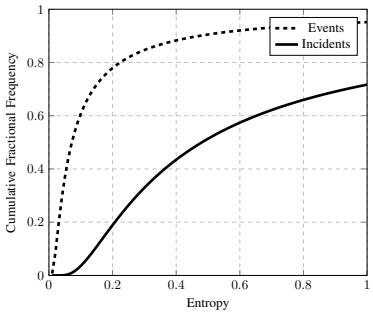


Fig. 1: Ideal Thresholding Cumulative Distribution of Incidents and Events

10^3 events per second (eps). In the example described above the average event rate is 1562 eps, but, from our experience, this can peak to 20,000 eps. To perform RCA across all events, the flow of events has to be significantly reduced, by many orders of magnitude (for example see [4]). Even commonly known techniques, such as compressing repeat events with de-duplication techniques (as described in [5]), which can result in a reduction of events by a factor of 10-100 are not sufficient when event throughputs are often measured in terms of billions of events per day. Failure to control this event rate overload is a principal cause of service outages going undetected by monitoring tools.

The most common approach to reducing the event rate is the simple act of removing uninteresting events with a manual filter or exclusion list, a process known as 'blacklisting'. Blacklisting , which originated in the security event monitoring discipline [6], is extremely time consuming and error prone. At industrial scale, blacklisting can require thousands of rules; in a fast changing network, such an approach is not practical. It is also extremely easy to accidentally blacklist a critical node and miss an event which leads to a service impacting outage. A method to automatically eliminate uninteresting events would yield significant savings, and is the central goal of our research. In particular, we seek a method which can take the topology of a network and automatically discard uninteresting events. The central difference in such an approach from blacklisting is that due to the efficient computability of the metrics discussed in this paper the method can be used even with a dynamic network topology. Blacklisting, by design, is static and requires human intervention to adapt to network changes. Although it may seem potentially risky to throw away events, admitting the possibility that causal events are discarded along with unimportant noise, the alternative is being unable to monitor *any* events and therefore missing *every* causal event.

A. Characteristics of an Ideal Metric

An effective metric should able to identify which nodes are more likely to produce events that will escalate into incidents. An ideal result, given that in the example above only 0.0003% of events get escalated into alerts, would be a metric that can discard 99.999% of events, whilst retaining the few that become incidents. Practically though, given that the goal is to fix the scalability limitations of RCA, we are seeking a metric that can reduce the load by 90%. Further for this to be a practical approach, calculating the metric must itself not present scalability challenges. An ideal metric must:

1) identify which nodes are most likely to produce an incident.
2) allow the discarding of at least 90% of events by the network topology alone.
3) be easy to calculate (not involve any intrinsically non-scalable computational steps) from the network topology alone.
4) be easy to update when the topology changes, ideally involving only computations for a small number of nodes in the region of the network where changes occurred.
5) Assuming a uniform probability of a node emitting an event[3], the metric must clearly segregate a small subset of critical nodes.

Ultimately the measure of RCA is its ability to capture all root causes and not mis-identify any false positives. This is best described in the language of machine learning using precision and recall. In particular the F_1 score (see [7] for a good description), is a popular measure of the effectiveness of a categorization algorithm such as RCA. Any method which discards root causes (false negatives) along with uninteresting events (true negatives) (or conversely any method that flags root causes (true positives) along with uninteresting events (false positives)) will affect the F_1 score of the overall system. The F_1 metric is most usually defined as the harmonic mean of precision and recall, which we define in Equation (1). In our context $precision$ is measured as the fraction of incidents in the events remaining after discarding all events and incidents that occur below a given value of our metric. Similarly, $recall$ is the fraction of incidents remaining after this discard over all recorded incidents. The value of β in this equation, when set to 1, recovers the standard F_1 measure. In essence when precision and recall are balanced, F_1 is maximized. For our purposes we set a value of β higher to bias the importance of recall over precision in monitoring applications.

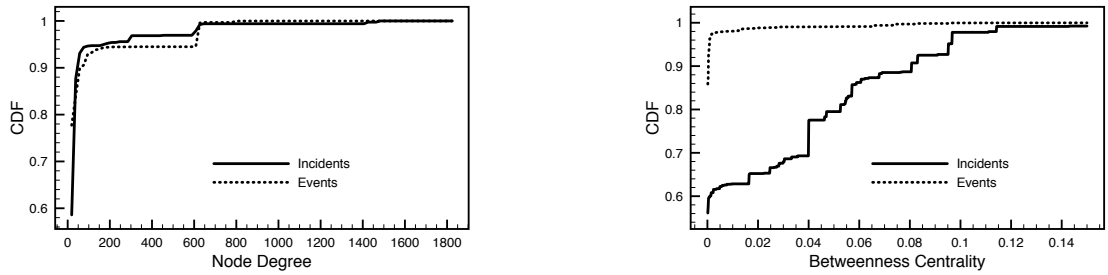
$$F_\beta = (1+\beta^2) \times \frac{precision \times recall}{\beta^2 \times precision + recall} \tag{1}$$

In Figure 1 we illustrate an idealized cumulative distribution of events versus incidents for an ideal metric. This distribution would be achieved if incidents were more likely to occur on nodes with high values of the metric, versus events, according to a distribution around a distinct mean value. This type of skew of incidents towards a higher metric value would allow us to discard events below a given threshold that would remove proportionately far more events than incidents.

A starting place to identify a workable metric is the work of Barabàsi and Albert [8] on network resilience, which was based upon data described by Faloutsos et al [9] and Li et al [10]. Analysis of this data was used by Barabási and Albert to assert that communications networks have a power law

[3]Experience from commercial deployments points to this assumption being reasonable.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2017.2724301, IEEE Transactions on Network and Service Management



(a) (b)

Fig. 2: Cumulative Distribution of Incidents and Events by Node Degree (a) and Betweenness Centrality (b)

node degree sequence, possessing the *Scale-Free* property, whereby, node degree distributions obey the inverse power distribution law. This was further used to justify the claim that communications networks, like the Internet, are both robust to random attack and vulnerable to targeted attack (the central arguments are outlined in [11], [12], [13], and again in [8]). In essence, when removing nodes from a graph randomly, the collapse in connectivity, as measured by the reduction in the size of the giant component, is gradual. However, if nodes are removed by choosing those with highest degree, this reduction is much more rapid (typically removing less than 10% of the nodes will reduce the size of the giant component by more than 90% [8]). It is therefore natural to postulate that node degree could be a metric that satisfies our criteria.

In Figure 2a we present the cumulative distribution of events and incidents by node degree. When we inspect the distribution in Figure 2a the lack of distinction between the cumulative event and incident distribution makes it clear that this does not conform to the idealized distribution in Figure 1. This distinction, which is apparent in the idealized distribution described above, means, at a fixed value of the metric, a far larger proportion of events initiate from nodes of values below this point than incidents. The absence of this preferential tendency for high degree nodes to produce incidents, means degree is a poor metric to achieve a suitable cutoff that would preferentially discard events over incidents. Although degree is extremely easy to calculate (3^{rd} criterion), it fails the first and most important criterion, as it does not provide any useful way of identifying nodes more likely to produce an incident. This lack of correlation is most likely due to high degree nodes being redundantly connected into the network and they may also not impact network function when they fail.

Beyond degree measures there are many other proposed metrics that measure node importance, often centering around centrality measures such as betweenness and eigenvalue centrality ([14], [15]). In Figure 2b we plot the cumulative distribution of events and incidents by betweenness centrality, for our sample data. Betweenness centrality measures the number of shortest paths between any two points in the network that pass by a given node as a fraction of all shortest paths. High values of centrality indicate a node that is critical to the connectivity of the network. It is clear that the effectiveness of this metric is far higher than degree, which is unsurprising as centrality quantifies the importance of a node in terms of connectivity between all points in the graph. Unfortunately the calculation of betweenness centrality scales badly. In the best case for betweenness centrality the fastest known algorithm developed by Brandes ([16], still scales as $O(|V|\times|E|)$, which in the case of our proprietary data is practically unfeasible to compute. To illustrate the problem, on our sample data this calculating the centrality for every node in our proprietary data set required 41 days on server grade hardware. This compares with the entropy metrics described in Section IV, which in identical conditions, require around 1.5 hours to compute every metric for every node sequentially. As our metrics only depend upon local properties of a node and could be calculated locally without a whole graph computation. In practice this means that for a given node, our most efficient metrics VE and VE', compute in less than a second, opening up the possibility that they can be maintained automatically in even the most dynamic environments.

The focus of our research has been with graph entropy, building on the entropy metric presented by Tee et al in [17]. Entropy has been studied in other contexts for anomaly detection (recently [18], and [19] applied the approach to traffic anomaly detection), but graph entropy has received little attention in the context of fault management. As a measure of graph structure it has serious computational drawbacks as its calculation is well known to be *NP-Hard* ([20]), which may account for this. However if these could be overcome with a node level, vertex, approximation, it would be ideal. Using such a node level measure of graph entropy, the proposed technique would be automatically driven from a graph representation of the topology of the monitored network, and importantly could be quickly computed from available inventory databases. Ideally, such a metric would conform to the cumulative distribution illustrated in Figure 1. In this way, at the expense of missing a small number of incidents, the volume of events that need processing can be significantly reduced. As all incidents have an associated event, it is not expected that the distribution would allow perfect recall of incidents as you discard events, but any actual distribution approximating this would be useful in establishing an entropy threshold to allow the discarding of events from nodes less likely to produce an incident. We will seek to demonstrate that our proposed vertex entropy metrics approximate this distribution. A central objective of our research has been to identify easily computable metrics that measure the contribution of an individual node to the entropy of the whole graph. Additionally, for these metrics to be valid entropy

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2017.2724301, IEEE Transactions on Network and Service Management

measures, we need to establish their extremal behavior satisfies the criteria of *maximality*, and demonstrate that they satisfy the other essential entropic properties of *additivity*, *symmetry* and *positivity* [21], [22]. Ideally the extremal values of our local variants would coincide with the global entropy measures and provide confidence that these metrics measure the complexity, and therefore, resilience of the networks they represent.

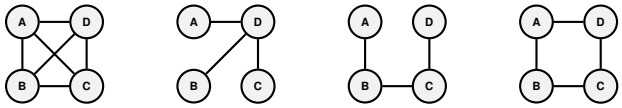
B. Overview

In this paper we describe both the theoretical approach for choosing a valid vertex entropy measure, and also analyze the results when this is applied to our "ground truth" data. Our core motivation is to identify an approximate way of measuring the contribution an individual node makes to the whole graph's entropy, and use that as our metric to eliminate noisy events. However, traditional definitions of graph entropy have insurmountable computational difficulties when applied to networks at scale. The starting point for our investigation is to establish whether there exists node or vertex level measures that when summed across the whole graph behave like the traditional measures. Establishing the existence of such a vertex level metric necessitates an exploration of the characteristics of global entropy measures on simple connected graphs. In section II we present an overview of graph entropy, introducing both *Chromatic* and *Structural Entropies*. Structural Graph Entropy quantifies the degree of connectivity resilience of a graph to edge removal, with low values of structural entropy corresponding to a fully connected or perfect graph, and high values a non-uniform graph with low resilience to edge removal. Chromatic Graph Entropy operates in the reverse sense, with uniform graphs having high chromatic entropy.

A valid entropy measure must satisfy the criteria of *maximality*, *additivity*, *symmetry* and *positivity*. Although *additivity*, *symmetry* and *positivity* are satisfied trivially by the definitions of global entropy, *maximality* is investigated in detail in Section III. We only concern ourselves with simple connected graphs and we prove that for an arbitrary sized graph, the *Star Graph* (S_n) and the *Complete Graph* (K_n) are extremal for both Structural and Chromatic Entropies. Ideally these properties should be shared by our vertex level metrics when summed across the whole graph.

A framework for the construction of node level entropies has been extensively explored in the work of Dehmer *et al*, and summarized in [23]. In section IV we build upon this framework to introduce our proposed forms of local vertex entropy, and investigate their extremal behavior. An important result of our paper is that the vertex entropies we propose have strong analogous behavior to the global variants, when summed across the whole graph, and satisfy *maximality*, *additivity*, *symmetry* and *positivity*. We further demonstrate that our metrics share similar extremal behavior to both global variants.

In section V we evaluate the proposed measures over a large enterprise network. The principal result of our paper is that the vertex entropy measures provide *a computable and effective way to identify important nodes that are more likely to produce incidents*. This is established by identifying that the distribution of event and incident frequency by vertex entropy strongly favors incident production at high values of the metric, verified by analysis of the data using a 2 sample Kolmogorov-Smirnov null hypothesis test to identify whether the distribution of incidents and events by our metrics are trivially correlated. In all cases we can dismiss the null hypothesis and conclude that the metrics produce independent distributions of the events versus the incidents. In addition we calculate for the data a version of the F_1 score, adjusted to account for the preponderance of raw events. Again all proposed metrics demonstrate acceptable improvement in the pre-conditioning of the event data. It is certainly not the case that all incidents occur above a fixed threshold, but at the cost of missing 20% of the incidents, 60-70% of the events can be safely ignored. We conclude our paper in Section VI, with an outlook regarding further research directions.



(a) K_4 (b) S_4 (c) P_4 (d) C_4

Fig. 3: Special Graphs on Four Nodes

II. Theoretical background

Historically, entropy was defined in Graph Theory as a measure of the complexity and non-uniformity of the global structure of a graph. Its use as an analytical tool in network science has been most studied in the dynamical evolution of network growth (see for example [24] and [25]). As a metric it captures many important characteristics, which are of direct interest in a number of applied fields, including the analysis of failure modes of communication networks (see [3]). In particular, networks with non-uniform connectivity will have high values of entropy. Unfortunately the most well understood measures of entropy involve calculations that have impractical computational complexity, as a graph scales in size (see [20] for a good explanation of this point). Further, any change to either the edges or vertices of a graph requires recomputing entropy across the whole graph. It is also extremely difficult to compute the contribution of each individual node to the graph entropy. The variants of Graph Entropy that we explore in this paper are, Körner or Structural Entropy and Chromatic Entropy. Structural entropy measures the mutual information of the stable sets of vertices defined on a graph, a string proxy for the complexity of the graph. Chromatic entropy is defined using the size of subsets of non adjacent vertices, or colorings, of a graph.

In our treatment we confine ourselves to simple, undirected graphs that are connected, and make reference to a number of special graphs, which we define as follows:

- **The Complete Graph (K_n)**: This graph is formed from a set of n vertices, maximally connected.
- **The Star Graph on n Vertices (S_n)**: This graph has one vertex v which is connected to all other vertices, with no other edges in the graph.
- **The Path on n Vertices (P_n)**: This graph is a simple chain of n vertices, connected by a single edge with no

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2017.2724301, IEEE Transactions on Network and Service Management

loops. The path has a single start node v_1 and end node v_n.

- **The Cycle on n Vertices (C_n)**: This graph is a special case of P_n such that $v_1 = v_n$; each node has degree 2.

In Figure 3 we present simple examples of these special graphs with $n = 4$.

Any valid entropy measure must satisfy a number of criteria, (for detailed descriptions see [21], [22]) to be admissible as a well behaved entropy metric. We define these properties on the entropy H of two graphs $F(V, E)$ and $G(V, E)$ as follows:

Definition 1. *For all graphs $F(V, E)$ and $G(V, E')$, sharing the same vertex set V a valid entropy $H(G)$ must satisfy:*

1) ***Additivity**: $H(F \cup G) \leq H(F) + H(G)$*
2) ***Symmetry**: $H(F \cup G) = H(G \cup F)$*
3) ***Positivity**: $\forall G, H(G) \geq 0$*
4) ***Maximality**: For a given collection of vertices V there is an edge set E such that the entropy $H(G)$ of a graph $G(V, E)$ is maximized*

As we explore our candidate entropy measures we will seek to prove that they satisfy these criteria where proofs do not exist in the standard literature.

III. Extremal Behavior of Global Graph Measures

A. Chromatic Entropy

A proper coloring of a graph is the division of the set of vertices V into a collection of subsets such that no member of any subset is adjacent to another member of the same subset.

For a given graph G there maybe multiple colorings, which amount to a collection, or set, of subsets of V. Each of these subsets we call a *Chromatic Class* C_i, with the constraint that $\bigcup_i C_i = V$. The Chromatic Number of a graph, $\chi(G)$, is the smallest number of such subsets that satisfy this constraint. The chromatic number of an graph is bounded by the maximum vertex degree k_{max} [26], [27]:

$$1 \leq \chi(G) \leq (1 + k_{max}) \tag{2}$$

Definition 2. *Chromatic Entropy*

$$I_c(G) = \min \left[-\sum_{C_i} \frac{|C_i|}{n} \log_2 \left(\frac{|C_i|}{n} \right) \right], \forall C_i. \tag{3}$$

where the minimization is over all possible colorings of the graph, and the summation is over all chromatic classes C_i, for a given coloring.

It is possible to establish the following limit on the value of $I_C(G)$:

Theorem 1. *For all graphs G, the Chromatic Entropy $I_c(G)$ is bounded by:*

$$0 \leq I_c(G) \leq \log_2(n) \tag{4}$$

Proof. We note that the lower bound is trivial, and consider the upper bound. We need only to maximize the function $f(p_i) = -\sum_i p_i \log_2(p_i)$ (in our case $p_i = \frac{|C|}{n}$), subject to the constraint $\sum_i p_i = 1$ and $p_i \leq 1, \forall i$, with equality only in the case of a trivial graph of one vertex. Given the definition of $I_C(G)$ as the minimum of equation (3) over all possible colorings, our maximum value will always be an upper bound of I_C. To maximize, we use the method of Lagrange multipliers, considering the following construct, subject to the unity sum constraint $\sum_i p_i = 1$ where $p_i = \frac{|C_i|}{n}$:

$$\mathcal{L} = \max_{p_i} \left[-\sum_i p_i \log_2 p_i - (\lambda - 1) \left(\sum_i p_i - 1 \right) \right] \tag{5}$$

Differentiating by p_i and setting to zero we obtain:

$$\frac{\partial \mathcal{L}}{\partial p_i} = 0; \implies \left(p_i = 2^{(1 - \lambda - \frac{1}{ln(2)})} \right) \forall i \tag{6}$$

From equation (6) our maximum is achieved when all values of p_i are identical and constant. In this case each chromatic class C_i is of identical size $|C_i| = \frac{n}{\chi(G)}$. Feeding this back into equation (3), and substituting for the bounds on $\chi(G)$ from (2) we obtain the desired result.

$$0 \leq I_c(G) \leq \log_2(n)$$

□

In practice these extremal values for $I_c(G)$ are achieved by the perfect graph on n vertices K_n for the maximum, which has a Chromatic Entropy of $\log_2(n)$, and its complement $\overline{K}_n$, where the set of edges is empty, has the minimum value of zero. However $\overline{K}_n$ is not a connected graph; for connected graphs we make the following proposition.

Proposition 1. *For all connected, simple graphs $G(V, E)$ of order $n > 3$ it holds that S_n minimizes $I_c(G)$*

Proof. For $n > 3$ any graph G of n vertices, can be created by progressively adding edges to either S_n or P_n, and by inspection of Table I, S_n has lower entropy than P_n. We will prove our proposition if we can demonstrate that the addition of an edge to any connected graph increases its chromatic entropy, as all graphs obtainable from S_n would have higher chromatic entropy than S_n. Consider any star graph S_n for $n > 3$. If any edge is removed, S_n will cease to be connected, and so by definition is not under consideration of the proposition. As we add edges to the graph S_n the change in chromatic number $\delta(\chi(G))$, can only ever be ≥ 1, or 0. So to complete the proof we consider both cases upon addition of an edge:

Case 1, $\delta(\chi(G)) \geq 1$ ***:*** The addition of a single edge creates an adjacency between two nodes, which must previously have been in the same chromatic class as $\delta(\chi(G)) \geq 1$. If the vertices where not in the same class we cover this in ***Case 2***. The recoloring of the graph will take one or both of the vertices connected by the new edge and add to, or create, a chromatic class of size x. This will reduce the size of a prior chromatic class of size y by x. Edge addition operations that increase chromatic number will always produce classes of increasingly uniform size as we approach a perfect graph K_n. Without loss of generality we will assume that $y > x$, as classes that grow to uniformity will *by necessity* borrow from larger classes as the size of all classes tend to unity. The change in chromatic information due to this re-assignment is:

$$\delta I_c(G) = \frac{x}{n} \log_2 \left(\frac{n}{x} \right) - \left(\frac{y}{n} \log_2 \left(\frac{n}{y} \right) - \frac{y - x}{n} \log_2 \left(\frac{n}{y - x} \right) \right)$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2017.2724301, IEEE Transactions on Network and Service Management

We seek to prove that $\delta I_c(G) \geq 0$ for all x, y where $y > x$. Elementary manipulation yields the following inequality:

$$\delta I_c(G) \geq 0 \rightarrow x \log_2 \left(\frac{y}{x} - 1 \right) \geq y \log_2 \left(1 - \frac{x}{y} \right)$$

As $\frac{y}{x} - 1 > 1 - \frac{x}{y}$ when $y > x$, we conclude that the inequality holds and $\delta(I_c(G)) \geq 0, \forall n > 3$ under the operation of edge addition when $\delta(\chi(G)) \geq 1$.

Case 2, $\delta(\chi(G)) = 0$ ***:*** **In this instance the addition of an edge does not increase the chromatic number. and as no chromatic classes need to change size,** $\delta I_c(G) = 0$**.**

Eventually additional edges increase the number of adjacencies and consequentially the chromatic number of the graph to its maximum of n, until we arrive at the complete graph K_n, which maximizes $I_c(G)$.In all cases we have seen that adding edges creates a $\delta I_c(G) \geq 0$, and as the first additional edge must belong to ***Case 1***, the proposition is proved.

□

B. Structural Entropy

The original paper of Körner [28], [21] introduced the entropy of graphs by extending traditional Shannon informational entropy. Körner's analysis considered an alphabet of signals, emitted according to a probability distribution, with not all of the alphabet being distinguishable. A graph is constructed such that each member of the alphabet is considered a vertex, with two vertices being connected by an edge if they are distinguishable, and a probability of emission, $P(V)$, being associated with each vertex. To develop the mathematical formulation of structural entropy, Körner introduces a probability distribution $P(V)$, to the normal construct of a graph $G(V, E)$, and defines S to be the maximal set of stable sets of $G(V, E)$. A stable set is a subset of the vertices which are not adjacent to any other member of the stable set, the maximal set being the collection of largest stable sets.. A number of equivalent definitions of structural entropy, $H(G, P)$ are possible, of which the simplest is in terms of the mutual information between $P(V)$ and $G(V, E)$ as follows [21]

Definition 3. *Körner or Structural Graph Entropy*

$$H(G, P) = H(P) - H(P|S) \tag{7}$$

This measure, which we call structural entropy, is related closely to the Chromatic Entropy. In our treatment we identify $P(V)$ with the probability of the emission of an event, which we further assume to be uniform. With that simplification the two quantities are related as follows (for an in depth treatment see [22]):

$$H(G, P) = \log_2(n) - I_c(G) \tag{8}$$

Structural entropy can most easily be interpreted as quantifying the extent to which the local neighborhood of a node is unique. In other words the value of $H(G, P)$ is minimized when all vertices are equivalently connected, and maximized when each node is distinguishable by its local topology. Given equation (8) we can state the following lemma on the bounds for $H(G, P)$.

Lemma 1. *For any graph $G(V, E)$ on n nodes, assuming that P is uniform, the structural graph entropy is bounded as follows:*

$$0 \leq H(G, P) \leq \log_2(n) \tag{9}$$

Proof. This bounding of $H(G, P)$ is easily verified by direct substitution of (4) into (8), and as such the proof is trivial. □

We summarize the extremal behavior of our global graph measures in table II.

TABLE I: Values of Global Entropies for Special Graphs

	$I_c(G)$	$H(G, P)$ - P Uniform
S_n	$\frac{n-1}{n} \log_2 \left(\frac{n}{n-1}\right)$ $-\frac{1}{n} \log_2(n)$	$\frac{n-1}{n} \log_2(n-1))$
K_n	$\log_2(n)$	0
P_n, n even	1	$\log_2(n) - 1$
P_n, n odd	$1 + \log_2(n) - (n+1)\log_2(n+1)$ $-(n-1)\log_2(n-1)$	$(n+1)\log_2(n+1)$ $+(n-1)\log_2(n-1)$ -1
C_n, n even	1	$\log_2(n) - 1$
C_n, n odd	$\log_2(n) - \frac{n-1}{n}\Big(1 - \log_2(n-1)\Big)$	$\frac{1-n}{n}\Big(1 - \log_2(n-1)\Big)$

TABLE II: Graph Types that Maximise and Minimize Entropy

	Chromatic	Structural
Maximum	K_n	S_n
Minimum	S_n	K_n

IV. Local Vertex Entropy Measures

Recent work on Graph Entropy by Dehmer [23], [29] provides a framework that unifies the global invariants discussed, and provides a pathway to extend these measures in a more computable direction. Both Structural and Chromatic entropy rely upon partitions of the vertex set of the graph, which are known *NP-Hard* problems.

Dehmer defines the concept of a *local functional* for a vertex, which can be scoped to calculate values for every vertex based upon the local topology of the graph. The degree of locality in the treatment is controlled by using the concept of *j-spheres*, S^j in the graph, centered at a given vertex. For clarity, in the definition that follows a superscript indicates the order of the j-sphere, whereas subscripts run over the members of the vertex set of the graph. Dehmer's original definition relied upon subsets of vertices of a fixed distance from a given vertex v_i. where distance $d(v_i, v_j)$ is the shortest distance between distinct vertices v_i and v_j (i.e. $i \neq j$). The distance is measured in the number of edges traversed in a walk from v_i to v_j, and in communications networks is commonly referred to as the 'hop' count. This definition excluded the vertex v_i, and other interior nodes for $j \geq 1$, but in our later treatment this introduces problematic zeroes when we define the clustering coefficient. We extend the definition of a j-sphere to include the node v_i as part of the set. This avoids certain special graphs such as S_n having zero clustering coefficients that would

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2017.2724301, IEEE Transactions on Network and Service Management

introduce infinities into our later definitions of normalized entropies. This is different to the definition given by Dehmer, in that we include all interior nodes to a given j-Sphere. The definition so modified is as follows:

Definition 4. *For a graph $G(V,E)$, we define for a node $v_i \in V$, the 'j-sphere' centered on v_i as:*

$$S_i^j = \{v_k \in V | d(v_i, v_k) \leq j, j \geq 1\} \cup \{v_i\} \tag{10}$$

and for convenience when we define the clustering coefficient in equation (21), the related 'j-edges' E_i^j as

$$E_i^j = \{e_{kl} \in E | v_k, v_l \in S_i^j\} \tag{11}$$

In essence the sets S_i^j and E_i^j are the local j-hop neighborhood of the node v_i, with S_i^j being the collection of all nodes j hops away from v_i, and E_i^j being the set of edges between them.

The concept of j-spheres is a very convenient formalism to capture *locality* in the graph. Essentially j can range from 1 to the diameter, $D(G)$, of the graph (as defined as the maximum length shortest path between two nodes). By breaking a large graph into j-spheres, we can progressively examine complex combinatorial quantities such as graph entropy on increasingly larger subsets of the graph until at $j = D(G)$ the global value is being effectively computed. Using our extended definition, we proceed by equipping each S_i^j with a positive real-valued function $f_i : v_i \in S_i^j \to \mathbb{R}^+$. This function is proposed to be dependent upon properties of the nodes that are members of the j-sphere, such as their degree, number of cycles and so on, which capture the local structural properties of the graph. From this, we can construct a probability function for each vertex as

$$p_i = \frac{f_i}{\sum_{v_j \in V} f_j} \tag{12}$$

which trivially satisfies $\sum_i p_i = 1$.

Essentially these functions are used to construct entropy measures in direct analogy to Shannon entropy as follows:

$$H(v_i) = -p_i \log_2 p_i \tag{13}$$

The principal direction of Dehmer's proposition is that these functions f_i when used to construct entropy, describe the local 'information' that a given vertex carries about the global structure of the graph. However, in the published work [23], [29], these functions are complex expressions, which introduce global invariants of the graph complicating their computation.

We can now apply Dehmer's formalism using the available invariants available in j-spheres for different values of j. For reasons of computational simplicity in this work we restrict ourselves to $j = 1$, which is the immediate local neighborhood of a given node. Although this sacrifices global structure of the graph, we will show that the results are still of operational significance and, because of locality, very efficient to compute. Indeed if $\langle k \rangle$ is the average degree of a node, most of our metrics are computable in $O(|V| \times \langle k \rangle)$, significantly less than, for example, centrality measures. Given the constraint of locality, a number of constructs can be designed that satisfy the probability functional defined in equation (12) up to a normalization constant. In the immediate neighborhood of a vertex the available measures are restricted to the degree of the vertex k_i, and the presence of cycles in the local subgraph. It is important that the measures that are constructed are bounded in an acceptable way, when summed across the whole graph and satisfy the fundamental properties of an entropy measure: *maximality*, *additivity*, *symmetry* and *positivity* [21], [22].

In Table III we summarize the available probability constructs that we will investigate. For j-spheres where $j > 1$ we have not conducted any analysis, and this remains an open question for further research. It should be noted though that as j approaches $D(G)$, the diameter of the network, the probability functionals approach a constant value, which is unlikely to reveal much of the structure of the network.

TABLE III: Local Probability Functional Constructs on a j-sphere

	$j = 1$	$j > 1$	$j = D(G)$
$\frac{1}{k_i}$	$VE(v)$	Unexplored, $\frac{1}{\lvert E_i^j \rvert}$	Constant Value $\frac{1}{\lvert E \rvert}$
$\frac{k_i}{\lvert E \rvert}$	$VE'(v))$	Unexplored, $\frac{\lvert E_i^j \rvert}{\lvert E \rvert}$	Constant Value 1
C_i^j	$NVE(v)$,$NVE'(v)$, $CE(v)$,$CVE'(v)$	Unexplored	Unexplored

A. Inverse Degree Entropy

The first and most basic probability functional, which we can construct on the 1-sphere of a vertex, uses its inverse degree k_i and is defined as follows:

$$p_i = \frac{1}{k_i} \tag{14}$$

and the corresponding entropy of the vertex $VE(v_i)$, and whole graph $H_{InvDegree}$ as

$$VE(v_i) = \frac{1}{k_i} \log_2(k_i), \tag{15}$$

for the whole graph:

$$H_{VE} = \sum_{i=0}^{i<n} \frac{1}{k_i} \log_2(k_i) \tag{16}$$

The first observation is that the sum of inverse degrees does not satisfy the constraint $\sum_i p_i = 1$. However, one can observe that for any given graph G, this probability functional sums to the constant:

$$C = \sum_{i=0}^{i<n} p_i = \frac{\sum_{i=0}^{i<n} \left(\prod_{j \neq i} k_j \right)}{\prod_{i=0}^{i<n} k_i} \tag{17}$$

We note that $p_i = \frac{1}{C} \times \frac{1}{k_i}$, and discard the constant as part of the normalization.

As the expression in equation (15) involves a sum of logarithmic terms (which are all positive), the conditions of *additivity*, *symmetry* and *positivity* are satisfied trivially, in particular for additivity as the combination of two graphs must as a minimum increase the degree of a vertex from each graph, or leave the degrees of the two graphs unchanged, the combined graphs entropy will be greater than or equal to the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2017.2724301, IEEE Transactions on Network and Service Management

sum of the two graphs, thereby satisfying the additivity criteria of Definition 1.

Regarding *maximality*, it suffices to establish that equation (16) has a maximum for a fixed set of vertices and edges. This can be done using Lagrange multipliers with the constraint $\sum_i p_i = C$, where C is the constant from equation (17). This yields an expression for the p_i as $p_i = 2^{(C-1-\lambda)}$, where λ is the Lagrange multiplier, confirming that the entropy has a maximal value for a graph whose degrees are equal. Referring to Table VI, we can see this is obtained by the cycle graph on n vertices C_n. Indeed the special graphs are ordered by increasing inverse degree entropies in the sequence $S_n < K_n < P_n < C_n$.

TABLE IV: Values of Vertex Entropy for Special Graphs

	$VE(n)$	$VE'(n)$
S_n	$\frac{1}{n-1}\log_2(n-1)$	$1+\frac{1}{2}\log_2(n-1)$
K_n	$\frac{n}{n-1}\log_2(n-1)$	$\log_2(n)$
P_n	$\frac{n-2}{2}$	$\frac{1}{n-1}+\log_2(n-1)$
C_n	$\frac{n}{2}$	$\log_2(n)$

B. Fractional Degree Entropy

Inverse degree is unsatisfactory. Firstly the probability functional is not naturally defined to satisfy the unity sum constraint. Secondly, and more importantly, the degree of a vertex does not capture how '*hub-like*' the node is relative to others. To capture this, we can define an alternative functional, which is based upon the ratio of the vertex degree to the total number of edges in the graph, as follows:

$$p_i = \frac{k_i}{2|E|} \tag{18}$$

Given that $\sum_{v_i \in V} k_i = 2|E|$ this functional directly satisfies the unity sum constraint. In a parallel way to equation (15), we define the fractional degree entropy as:

$$VE'(v_i) = \frac{k_i}{2|E|}\log_2\left(\frac{2|E|}{k_i}\right), \tag{19}$$

for the whole graph:

$$H_{VE'} = \sum_{i=0}^{i<n} \frac{k_i}{2|E|}\log_2\left(\frac{2|E|}{k_i}\right) \tag{20}$$

Following the treatment of Inverse Degree Entropy, we note that the expression in equation (20) again involves a sum of logarithmic terms (which are all positive), so the conditions of *additivity*, *symmetry* and *positivity* are satisfied. To establish maximality, we can again use the technique of Lagrange multipliers using the constraint $\sum_i p_i = 1$, which yields a similar result to inverse degree entropy that the maximal value is obtained for a graph with equal vertex degrees satisfying $p_i = 2^{1-\lambda}$. In Table VI this is satisfied by K_n and C_n. The special graphs using this measure are ordered in increasing fractional degree entropy as $S_n < P_n < C_n = K_n$. We summarize these results in Table V.

TABLE V: Extremal Graphs for Unnormalized Vertex Entropy

	VE	VE'
Maximum	C_n	$K_n = C_n$
Minimum	S_n	S_n

C. Normalized Degree Entropy

There is a considerable practical difference between a star network topology (S_n) and a fully meshed one (K_n). In the former, the network is vulnerable to the loss of its central high degree vertex; in the latter, the loss of any one vertex can never create isolated vertices. Both prior measures make little distinction between these two topologies for nodes of identical degree, but there are available metrics measurable at one hop distance that capture this concept. Indeed, in the case of fractional degree, there is no way for the degree to capture the intricacies of the local topology of the node. Introduced in [30] and [8] is the concept of the clustering coefficient of a vertex. The traditional definition counts edges between neighbors of a vertex, which yields a zero value for S_n that is problematic in our treatment. We avoid zeros using our extended version of the j-sphere in equation (10). In terms of the degree of vertex i, k_i, the following definition captures how similar the j-sphere surrounding a vertex is to the complete graph K_n and is defined in terms of the 1-sphere edge set E_i^j as:

$$C_i^1 = \frac{2|E_i^j|}{k_i(k_i+1)} \tag{21}$$

In essence the clustering coefficient measures the probability that two randomly chosen nodes in the 1-hop subgraph have an edge between them. In this way the lower the value of the coefficient, the higher the likelihood that the failure of the node at the center of the subgraph will cause two nodes to become disconnected (see for example [31]). This completely captures how well meshed a node is into its local neighborhood, and therefore serves as an ideal candidate for further refining the vertex measures introduced earlier. In particular, we want to highlight vertices whose clustering coefficient is low, that is, their local neighborhood is more similar to S_n locally than K_n. To that end we define the following *Normalized* Vertex Entropies:

Definition 5. *We define for a graph $G(V,E)$ the following* Normalized *Inverse Degree Entropy for both vertex and total graph as follows:*

$$NVE(v_i) = \frac{1}{C_i^1} \times VE(v_i), \tag{22}$$

for the whole graph:

$$H_{NVE} = \sum_{i=0}^{i<n} \frac{(k_i+1)}{2|E_i^1|}\log_2(k_i), \tag{23}$$

and the corresponding definition for fractional vertex entropy is defined similarly:

$$NVE'(v_i) = \frac{1}{C_i^1} \times VE'(v_i), \tag{24}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2017.2724301, IEEE Transactions on Network and Service Management

and total entropy:

$$H_{NVE'} = \sum_{i=0}^{i<n} \frac{k_i^2(k_i+1)}{4|E||E^1(v_i)|} \log_2\left(\frac{2|E|}{k_i}\right) \tag{25}$$

Proving compliance with Definition 1 for these normalized values is not as straightforward as the non normalized values. However, as the expression in equation (21) is always positive, the *symmetry* and *positivity* criteria are automatically satisfied. With regard to additivity and criteria 1 of Definition 1, although not a rigorous proof, considering two graphs being minimally joined by a single vertex, the clustering coefficient of that vertex will *decrease* and so the value of NVE or NVE' of the shared vertex will increase, satisfying the inequality.

For maximality, the introduction of the clustering coefficient complicates the use of the Lagrange multiplier method, as p_i and C_i^1 are related quantities. It is the beyond the scope of this work to present a formal proof of *maximality* but we can calculate the values of the normalized entropies for our special graphs and we summarize the results in Table VI. The special graphs using NVE ordered in increasing entropy are in the sequence S_n, K_n, P_n, C_n and for NVE', K_n, C_n, P_n, S_n. With the assumption that it is possible to maximize these entropies these values are admissible measures of entropy. It is interesting to note that the distinction between star topologies and meshed ones is much less distinct with NVE. Comparing extremal behaviors to our global entropy measures, we identify NVE with Chromatic entropy and NVE' with Structural entropy.

TABLE VI: Values of Normalized Entropy for Special Graphs

	NVE	NVE'
S_n	$\frac{n}{2(n-1)} \log_2(n-1)$	$\frac{1}{2}\log_2\{2(n-1)\} + \frac{n}{4}$
K_n	$\frac{n}{n-1} \log_2(n-1)$	$\log_2(n)$
P_n	$\frac{3}{4}(n-2)$	$\frac{1}{n-1} + \frac{3n-4}{2(n-1)} \log_2(n-1)$
C_n	$\frac{3}{4}n$	$\frac{3}{2}\log_2(n)$

TABLE VII: Maximal and Minimal Total Vertex Entropy Graph Types

	NVE	NVE'
Maximum	C_n	S_n
Minimum	S_n	K_n

D. Alternative Vertex Entropy Constructions

The local clustering coefficient C_i^1 can also be used to construct two alternative probability functionals, which an exhaustive study necessitates. In the first instance, as the clustering coefficient itself is a value strictly in the range $(0, 1]$ it is a valid informational functional in its own right. We can define a clustering coefficient entropy, $CE(v_i)$ by identifying $p_i = C_i^1$, as follows:

Definition 6. *For a graph $G(V, E)$ the clustering coefficient entropy, $CE(v_i)$ of a vertex v_i is defined as*

$$CE(v_i) = C_i^1 \log_2 C_i^1, \tag{26}$$

and for the whole graph:

$$H_{CE} = \sum_{i=0}^{i<n} C_i^1 \log_2 C_i^1 \tag{27}$$

In addition, we can also approach the normalization of the fractional vertex entropy by defining an alternative probability functional using the clustering coefficient as:

$$p_i = \frac{1}{C_i^1} \times \frac{k_i}{|E|} \tag{28}$$

This probability functional is within the range $(0, 1]$ as for a given vertex this simplifies to $p_i = \frac{|E_i^1|}{(k_i+1)|E|}$, which for a connected node is strictly non-zero and $|E_i^1| \leq |E|$. It is not possible to extend the inverse degree functional in a similar way as the equivalent definition $p_I = \frac{1}{k_i C_i^1}$ is not bounded to fall into the range $(0, 1]$. We therefore make the following definition for the Cluster Coefficient Fractional Degree Entropy as follows:

Definition 7. *For a graph $G(V, E)$ the Cluster Coefficient Fractional Degree Entropy $CVE'(v_i)$ of a vertex v_i is defined as:*

$$CVE'(v_i) = \frac{k_i}{C_i^1|E|} \log_2\left(\frac{C_i^1|E|}{k_i}\right), \tag{29}$$

and for the whole graph:

$$H_{CVE'} = \sum_{i=0}^{i<n} \frac{k_i}{C_i^1|E|} \log_2\left(\frac{C_i^1|E|}{k_i}\right), \tag{30}$$

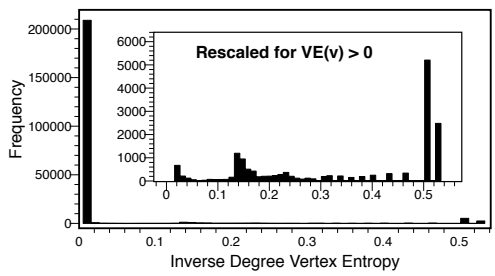
Using similar arguments to the previous entropy types we can establish conformance with *additivity*, *symmetry* and *positivity* of Definition 1 by observing that in equations (27) and (30) are sums of logarithms . The remaining property of *maximality*, in complex to verify due to similar issues to the normalized entropy values NVE and NVE'. It is beyond the scope of this paper to present a rigorous proof of *maximality*, but we can calculate the values for our special graphs, which we summarize in Table VIII. The special graphs using CE ordered by increasing entropy are in the sequence K_n, S_n, P_n, C_n and for CVE', S_n, C_n, P_n, K_n.

TABLE VIII: Values of Clustering Coefficient Entropies for Special Graphs

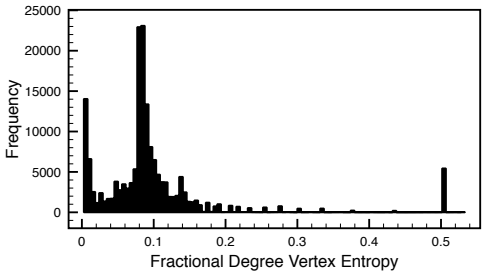
	CE	CVE'
S_n	$\frac{2}{n} \log_2(\frac{n}{2})$	$\log_2(n-1) - \frac{n}{2}\log_2(n)$
K_n	0	$2\log_2(\frac{n}{2})$
P_n	$\frac{2(n-2)}{3} \log_2(\frac{3}{2})$	$\frac{1}{n-1}\left[3(n-2)\log_2(\frac{n-1}{3}) - 2\log_2(n-1)\right]$
C_n	$\frac{2n}{3} \log_2(\frac{3}{2})$	$3\log_2(\frac{n}{3})$

From these calculations we can summarize in Table IX the extremal graphs for these entropies.

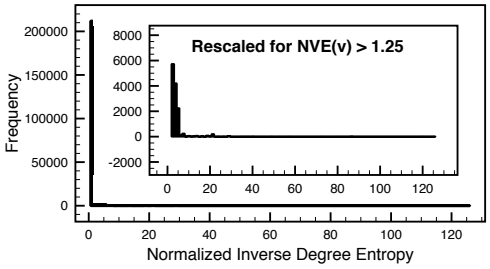
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2017.2724301, IEEE Transactions on Network and Service Management



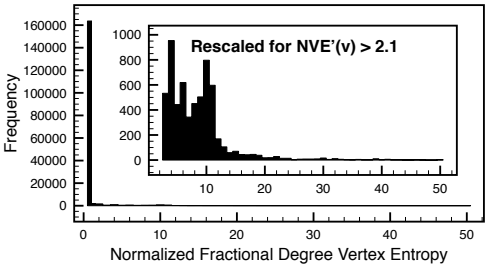
(a) Frequency Distribution of $VE(v)$



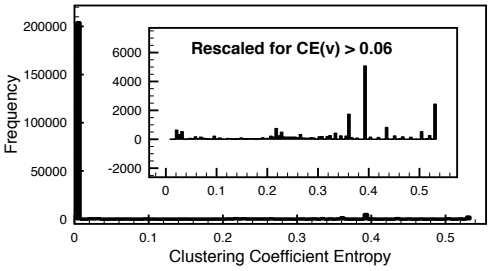
(b) Frequency Distribution of $VE'(v)$



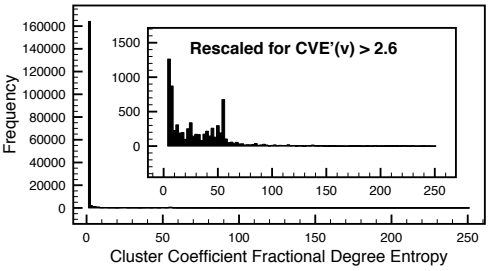
(c) Frequency Distribution of $NVE(v)$



(d) Frequency Distribution of $NVE'(v)$



(e) Frequency Distribution of $CE(v)$



(f) Frequency Distribution of $CVE'(v)$

Fig. 4: Frequency Distributions for a Network of 225,239 Nodes

TABLE IX: Extremal Graphs for Clustering Coefficient Entropy

	CE	CVE'
Maximum	C_n	K_n
Minimum	K_n	S_n

V. Evaluation and Discussion

A. Data and Methods

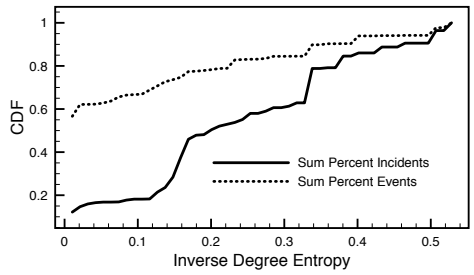
We analyzed data from a large operational dataset obtained from a web portal operator. In previous work [17] we also applied our techniques to the 'Internet Topology Zoo' (ITZ) ([32]), but this critically does not have any event or incident data.

Our commercial data, however, contains a rich source of events and incidents, and in particular allows the analysis of event and incident distribution by originating node. The analysis was performed using a suite of software tools implemented in JAVA, and operated in conjunction with a MySQL database for permanent storage[4]. A brief description is below:

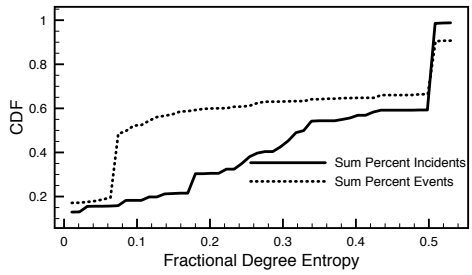
- `graph_analyser`: This executable was built to ingest source topology as a list of edges in a comma separated file format. The program calculates all of the metrics described in Section IV and betweenness centrality and stores the results both in raw and frequency distribution format in the database. The value stored in the database are used by the other analysis programs to produce distributions of events and incidents by node metric.
- `event_analyser`: This executable ingests and parses the full sample of events obtained from the customer. Each event is presented as a string of symbols separated by the '|' character. The format of the events followed a fixed pattern with the syntax: timestamp |datacenter | application | node | description. After each event is parsed the executable populates a distribution of event count by value of the metric for each value of 'node'.
- `incident_analyzer`: This executable operates in an almost identical fashion to the event analyzer, but instead analyzes data that is obtained from a report ran on the customer's incident management system. Each incident represents an event that has been escalated according to their manual triage process and is presented with the following syntax: date | timestamp |datacenter |

[4]The source code for these analysis tools is available at https://github.com/philtee2001/analyzer.git, and instructions for building are available from phil@moogsoft.com

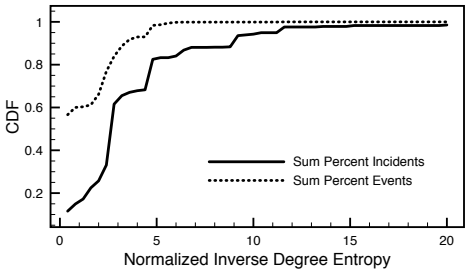
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2017.2724301, IEEE Transactions on Network and Service Management



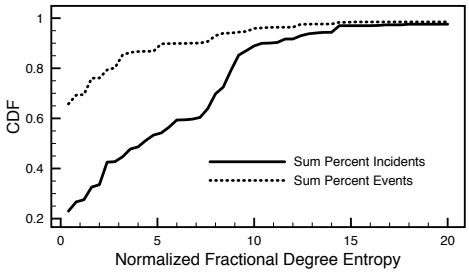
(a) Cumulative Distribution of Events and Incidents by $VE(v)$



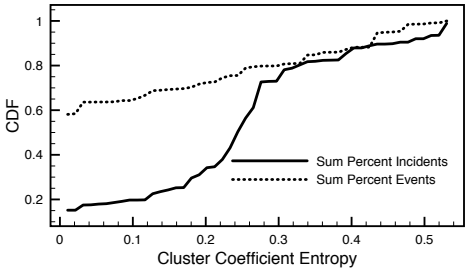
(b) Cumulative Distribution of Events and Incidents by $VE'(v)$



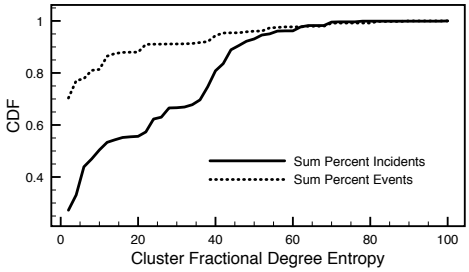
(c) Cumulative Distribution of Events and Incidents by $NVE'(v)$



(d) Cumulative Distribution of Events and Incidents by $NVE'(v)$



(e) Cumulative Distribution of Incidents by $CE(v)$



(f) Cumulative Distribution of Events and Incidents by $CVE'(v)$

Fig. 5: Cumulative Distributions of Events and Incidents in a Network of 225,239 Nodes

ticket number | node | type | property | agent | description. node field ties directly back to the topology data and is used to populate a distribution of incidents by considered metric. For this data not all values of 'type' are considered, as they indicate whether or not the incident was deemed to be significant. We discard any incidents that were not accepted by the help desk without escalation.

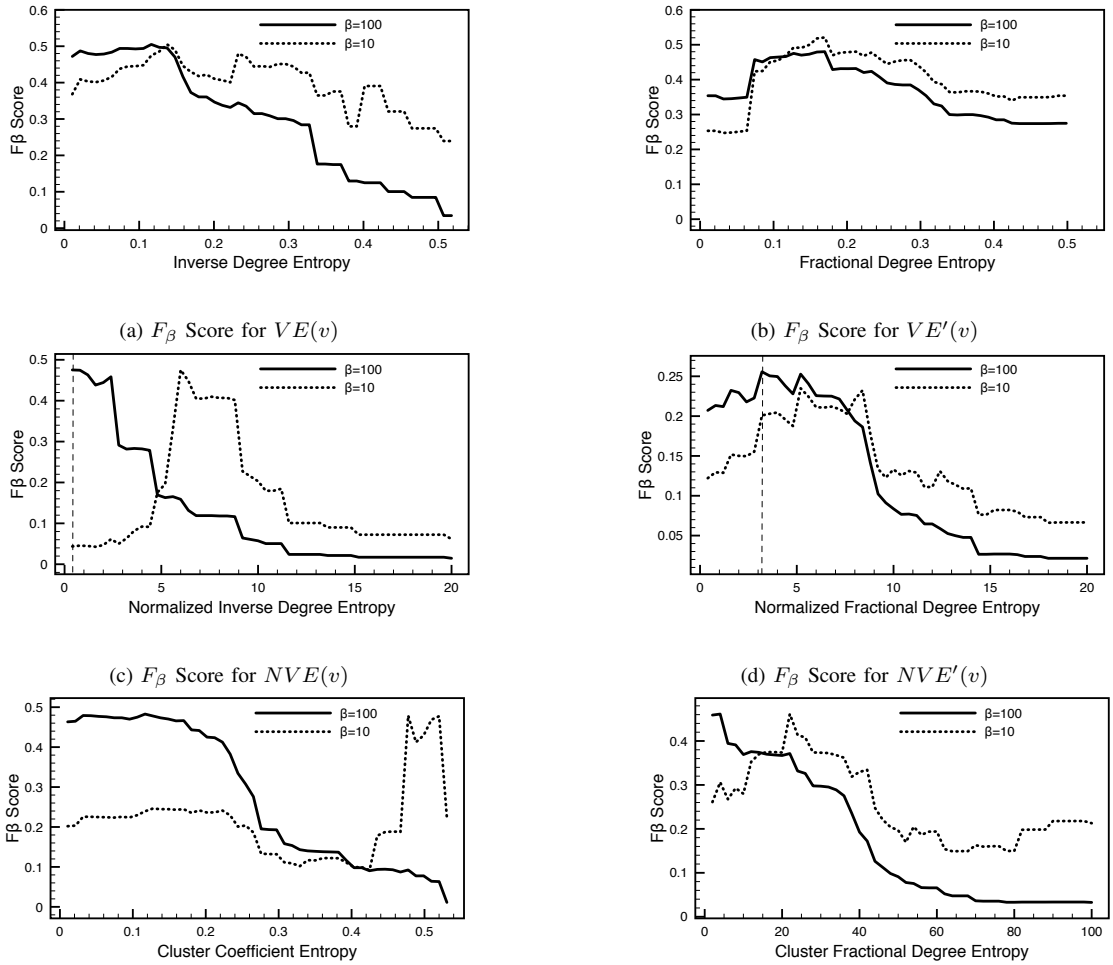
B. Evaluation

Using the dataset described in Section V-A, we begin in Figure 4 by plotting the distribution of nodes by the various entropy measures. For a number of the metrics, the data is heavily skewed by large numbers of the nodes having a zero of low value. In Figures 4a, 4c, 4d, 4e and 4f we plot the distribution excluding these values, rescaled. All of the measures share a common feature in that the vast majority of the nodes posses a heavy skew towards low values of the metrics. This is encouraging, because for an entropy metric to be useful in identifying important nodes a uniform distribution would be unexpected. Except in the case of fractional degree entropy VE' (Figure 4b) the skew is so pronounced that to illustrate the distribution above minimal values of the metric we have embedded a subgraph rescaled to eliminate the dominating cluster of values towards the low values of the metric.

With both inverse degree $VE(v)$ and fractional degree entropy $VE'(v)$ the distribution achieves the first objectives of being non-uniform and separating out a small subset of nodes with high values of the metric. In the earlier discussion in section I, this distribution profile was a necessary condition of the metric having utility when identifying nodes likely to produce incidents. However, these metrics do not distinguish between a high degree node that has many redundant paths into the network and one that does not. In our theoretical analysis in Section III, we identified the need to highlight nodes whose local topology was more similar to S_n than K_n, which the non-normalized metrics do not. The point of our normalized metrics is to capture this aspect of local topology and provide a way of identifying nodes that have high degree but low redundancy. From considerations of network design, these nodes are more likely to produce events that escalate into incidents when they fail.

To establish whether the data supports this hypothesis, we turn to the distributions of normalized inverse degree $NVE(v)$ and normalized fractional degree entropy $NVE'(v)$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2017.2724301, IEEE Transactions on Network and Service Management



(a) F_β Score for $VE(v)$

(b) F_β Score for $VE'(v)$

(c) F_β Score for $NVE(v)$

(d) F_β Score for $NVE'(v)$

(e) F_β Score for $CE(v)$

(f) F_β Score for $CVE'(v)$

Fig. 6: F_β Score Plots in a Network of 225,239 Nodes

in Figure 4c and Figure 4d. It is interesting to note that both quantities share the same non-uniform distribution as the non-normalized forms, with a much more pronounced separation of the extremal values. This is consistent with our supposition that the normalized metrics exclude a subset of high degree nodes that have multiple paths through the network.

To fully exhaust all potential metrics available on a 1-sphere, we also plot the distributions for clustering coefficient $CE(v)$ and cluster coefficient fractional entropy $CVE'(v)$ in Figures 4e and 4f. Again these distributions are skewed fairly heavily towards low values of the metric, and show interesting, much smaller clusters at higher values of the metrics.

Our central claim is that the local measures of vertex entropy are more effective at identifying nodes that will generate incidents than simply selecting the nodes of highest degree, as suggested by scale free models of dynamic networks. In Figure 2a we presented the distribution of events and incidents by node degree, from which it is clear that there is very little difference in the distribution between events and incidents, and that there are no useful distinctions between the distribution of incidents by degree versus events. Although high degree nodes are more likely to cause impact than low degree nodes when failing, network design usually mitigates failure points by adding in redundant paths through the network to avoid single points of failure. This is further underlined by the cumulative distribution plot in Figure 2a, where it is evident that the distribution of events and incidents is effectively the same. In Table XI we note that the 2-sample Kolmogorov-Smirnov test does not allow us to dismiss the null hypothesis, with a P-Value in excess of the α value, indicating that degree is not a discriminatory factor. A side effect of this analysis is affirmation that one of our key assumptions that events are emitted with uniform probability across all nodes. For these reasons, degree is not a reliable indicator of impact when a node fails.

To contrast this with our entropy based metrics in Figure 5 we plot the cumulative distributions of events and incidents by each of our candidate metrics. In each case there is a heavy skew towards higher values of the metrics for incidents versus events. This is the first indication that the vertex entropy metrics are indeed useful for identifying nodes more likely to produce incidents. As discussed in the introduction we can make use of the F score methodology to identify how effective our entropy metrics are at optimizing recall and precision. To do so, however, we must build upon the basic measure introduced in [7] to take account of the fact that our metric is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2017.2724301, IEEE Transactions on Network and Service Management

being proposed to pre-condition data before a categorization algorithm (RCA) is used to determine whether an event is causal. In general raw events contain many duplicate notifications which can be compressed by the application of a de-duplication (see for example [5]). This can result in the number of events being compressed by a factor of 10-100. In addition, the *cost* of a missed incident is significantly more impactful to a business than the cost of processing an event. As the basic measure of an F score assumes equal weight, we instead adopt a weighting factor β and measure the F_β score for our event and incident distributions. The F_β score is defined in equation (1). Effectively this metric measures the balanced effectiveness of an algorithm at identifying true positives without producing too much noise in the form of false positives. In the context of event management and RCA, this is the ability of an algorithm to capture every incident without surfacing *false* incident notifications. The typical application of the F_β score though weights precision and recall evenly, and given that a missed incident is potentially costly, the β parameter allows us to bias in favor of recall. We choose a heavy bias of $\beta = 100$.

In Figure 6 we plot for each of our metrics the F_β scores as a function of the metric for a $\beta = 10$ and $\beta = 100$. Plotting the F_β score identifies a value of the entropy metric that maximizes the F_β score. This maximum corresponds to the best threshold to use to discard events from nodes with a value of entropy that is below it, allowing you to reduce event load whilst preserving events that are likely to escalate into incidents. These plots illustrate the importance of the weighting factor in the F_β score for identifying the correct choice of entropy to set a discard threshold at. In each case the $\beta = 100$ establishes a lower discard threshold as you would expect, given that we are treating recall as more important than precision, as the maxima of the F_β score occurs at a lower value of the entropy measure. In Table X we collect the discard rates at the maximum of the F_β score for $\beta = 100$. In each case it is evident that it is possible to choose a value of the metric, in this case our choice of vertex entropy, that will selectively discard many more events than incidents, and in fact, by the nature of the scaling of the F_β score, at a value of entropy that would discard 20% of the incidents, some 65% or $15,000,000$ events can be safely discarded. For the data we analyzed , this amounts to discarding $62,600,000$ events *before* expensive RCA processing. This amounts to reducing the event rate from approximately 12 per second to 4, which operationally could be very significant. In order to replicate this result using manual blacklisting, this would require the maintenance of a list of nodes that are relatively unimportant. In the case of the network we analyzed, that would amount to some $200,000$ nodes, which are apt to change frequently. As we indicated in the Section I alternative simpler metrics such as node degree are unable to achieve similar effectiveness in identifying important incident producing nodes as our entropy metrics or centrality measures.

To further test the correlation between our vertex entropies and incident creation, statistical hypothesis testing of the distributions using a 2-sample Kolmogorov-Smirnov goodness of fit between cumulative distributions of events and incidents

TABLE X: Maximal % Discards of Events and Incidents ($\beta = 100$)

Metric	Max Value	% Events	%Incidents
VE	0.116	87%	52%
VE'	0.170	68%	22%
NVE	0.400	57%	12%
NVE'	3.200	85%	45%
CE	0.127	67%	20%
CVE'	4.000	76%	32%

was undertaken. Using an α of 5%, and assuming the *Null Hypothesis* that both event and incident distributions of all metrics shared the same cumulative distribution, very low P-Values were obtained, indicating that the difference in distributions is highly unlikely to be the effect of randomness. We summarize the findings in Table XI. This result convincingly contradicts the *Null Hypothesis*, and we can safely conclude the difference in the distribution is a result of a strong correlation between high values of both metrics, and a higher likelihood of events escalating into incidents. This result continues to be valid down to values of $\alpha = 1\%$, and is a strong indication that our local metrics are capturing enough of the local topology of the network to be useful as a way of assessing the impact of a nodes failure on the overall connectivity of the network. In essence, impact is a result of the node being part of a large number of shortest paths between any two arbitrary points in the network. Although high degree makes it more likely, the similarity of the local topology of the node to K_n versus S_n mitigates that, and our normalized metrics successfully account for this subtlety. It is interesting to note that the *Null Hypothesis* cannot be dismissed for the degree distributions as the P-Value is higher than $\alpha = 5\%$.

It is interesting to speculate which of the metrics is the most effective metric to use to pre-condition events for RCA. In practice any of the metrics investigated appear to have merit, but it is important to note that the local clustering coefficient of a node can be expensive to compute for highly connected and nodes in a heavily meshed network. For a network that is maximally connected with n nodes, the calculation of the clustering coefficient is an $O(n^3)$ calculation, as each of the n nodes will have $\frac{n(n-1)}{2}$ edges. This is to be balanced with the more favorable Kolmogorov-Smirnov analysis of the normalized entropies NVE, and NVE', which yield lower P-Values. This lower value indicates greater predictive power, but at the expense of a more expensive calculation.

TABLE XI: Kolmogorov-Smirnov Analysis of Null Hypothesis for Event Incident Distributions

Metric	D-stat	D-Crit	α	P Value	Significant
VE	0.5055	0.0396	5%	0.63%	Yes
VE'	0.4624	0.0399	5%	0.26%	Yes
NVE	0.4489	0.0399	5%	0.19%	Yes
NVE'	0.4462	0.0404	5%	0.16%	Yes
CE	0.4665	0.0399	5%	0.29%	Yes
CVE'	0.4394	0.0403	5%	0.14%	Yes
Degree	0.0368	0.0403	5%	9.29%	No

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2017.2724301, IEEE Transactions on Network and Service Management

VI. Conclusions

In this paper we introduced computable, node level alternatives to structural entropy measures that are useful when identifying critical nodes in a network. Building on the approach of network science established in Barabási's pivotal paper, and suggestions made in the work of Dehmer, we have advanced computable metrics using structural information available within one hop of a network node. By analyzing the extremal properties of well known global graph entropies, we were able to identify that they satisfy the criteria required to be a valid entropy, and have similar extremal behavior to the global values when considering special graphs. Critically, the introduction of normalization based upon the clustering coefficient of a nodes neighborhood improves the utility of the metric. We applied these measures to our proprietary data set. Applied to the datasets, we obtain a distribution that isolates a small subset of nodes with high values, a necessary condition to be acceptable as a metric.

This analysis is further supported when we look at the distribution of events and incidents by the value of the metric. We have a clear correlation between high values of the metric and the propensity for the node to produce incidents. This is substantiated by hypothesis testing to eliminate the possibility that the distributions are similar to each other, and therefore that any difference in distribution of events and incidents is purely random. Additional precision and recall analysis using a modified F_β score indicates that there is the possibility of establishing a value of the metric whereby minimal loss of recall (20% of incidents missed) is tolerable to achieve a reduction of 65% in the event rate that needs to be processed. In the context of the large and dynamic networks of current implementations this could be a critical improvement in the performance of root cause algorithms.

All of our analysis has been constrained to the immediate one-hop neighborhood of a node. The justification of studying these values in practical networks has been achieved in theory, and in further work we intend to analyze more real world datasets, and extend our entropy measures to include j-spheres for $j > 1$. In addition, we plan to compare vertex entropy against other node importance measures such as betweenness, to assess the difference in effectiveness as compared to cost of calculation.

References

[1] S. Boccaletti, G. Bianconi, R. Criado, C. I. del Genio, J. Gómez-Gardeñes, M. Romance, I. Sendiña-Nadal, Z. Wang, and M. Zanin, "The structure and dynamics of multilayer networks," *Physics Reports*, vol. 544, no. 1, pp. 1–122, 2014. [Online]. Available: http://dx.doi.org/10.1016/j.physrep.2014.07.001

[2] M. L. Steinder and A. S. Sethi, "A survey of fault localization techniques in computer networks," *Science of Computer Programming*, vol. 53, no. 2, pp. 165–194, nov 2004. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0167642304000772

[3] S. Kliger, S. Yemini, and Y. Yemini, "A coding approach to event correlation," *... Network Management IV*, 1995. [Online]. Available: http://link.springer.com/chapter/10.1007/978-0-387-34890-2_24

[4] M. Miyazawa and K. Nishimura, "Scalable root cause analysis assisted by classified alarm information model based algorithm," *... of the 7th International Conference on ...*, pp. 2–5, 2011. [Online]. Available: http://dl.acm.org/citation.cfm?id=2147737

[5] R. Harper, "Entropy & The Science of Noisele," 2016. [Online]. Available: https://www.moogsoft.com/whats-new/entropy-noise/

[6] L. Metcalf and J. M. Spring, "Blacklist Ecosystem Analysis Spanning Jan 2012 to Jun 2014," *ACM Digital Library*, pp. 13–22, 2014.

[7] D. Powers, "Evaluation: From Precision, Recall and F-Measure To Roc, Informedness, Markedness & Correlation," *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, 2011. [Online]. Available: http://www.bioinfopublication.org/files/articles/2_1_1_JMLT.pdf

[8] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Review of Modern Physics*, vol. 74, no. January, 2002.

[9] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On Power-Law Relationships of the Internet Topology," *In SIGCOMM*, pp. 251–262, 1999. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.37.234

[10] L. Li, D. Alderson, W. Willinger, and J. Doyle, "A First-Principles Approach to Understanding the Internet ' s Router-level Topology," *Acm Sigcomm*, pp. 3–14, 2004.

[11] B. Bollobás and O. Riordan, "Robustness and Vulnerability of Scale-Free Random Graphs," *Internet Mathematics*, vol. 1, no. 1, pp. 1–35, 2004.

[12] B. Bollobás and O. Riordan, "Mathematical results on scale-free random graphs," in *Handbook of Graphs and Networks*. Wiley-VCH, 2006, ch. Mathematic, p. 417.

[13] R. Albert, H. Jeong, and A. Barabasi, "Error and attack tolerance of complex networks," *Nature*, vol. 406, no. 6794, pp. 378–82, 2000. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/10935628

[14] L. Freeman, "Centrality in social networks conceptual clarification," *Social networks*, vol. 1, no. 1968, pp. 215–239, 1979. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0378873378900217

[15] L. Spizzirri, "Justification and Application of Eigenvector Centrality," *Math.Washington.Edu*, 2011. [Online]. Available: https://www.math.washington.edu/ morrow/336_11/papers/leo.pdf

[16] U. Brandes, "A faster algorithm for betweenness centrality*," *The Journal of Mathematical Sociology*, vol. 25, no. 2, pp. 163–177, 2001.

[17] P. Tee, G. Parisis, and I. Wakeman, "Towards an approximate graph entropy measure for identifying incidents in network event data," in *NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium*, April 2016, pp. 1049–1054.

[18] Y. Kanda, R. Fontugne, K. Fukuda, and T. Sugawara, "ADMIRE: Anomaly detection method using entropy-based PCA with three-step sketches," *Computer Communications*, vol. 36, no. 5, pp. 575–588, 2013.

[19] G. Nychis, V. Sekar, D. G. Andersen, H. Kim, and H. Zhang, "An empirical evaluation of entropy-based traffic anomaly detection," *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement conference - IMC '08*, p. 151, 2008. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1452520.1452539

[20] M. E. Bolanos, S. Aviyente, and H. Radha, "Graph entropy rate minimization and the compressibility of undirected binary graphs," *2012 IEEE Statistical Signal Processing Workshop, SSP 2012*, no. 2, pp. 109–112, 2012.

[21] G. Simonyi, "Graph entropy: a survey," *Combinatorial Optimization*, vol. 20, pp. 399–441, 1995.

[22] A. Mowshowitz and V. Mitsou, "Entropy, Orbits, and Spectra of Graphs," *Analysis of Complex Networks: From Biology to Linguistics*, pp. 1–22, 2009.

[23] M. Dehmer and A. Mowshowitz, "A history of graph entropy measures," *Information Sciences*, vol. 181, no. 1, pp. 57–78, 2011.

[24] J. Park and M. E. J. Newman, "Statistical mechanics of networks," *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, vol. 70, no. 6 2, pp. 1–13, 2004.

[25] P. Tee, I. Wakeman, G. Parisis, and J. Dawes, "Is Preferential Attachment the 2nd Law of Thermodynamics in Disguise?" *ArXiv e-prints*, Dec. 2016.

[26] H. S. Wilf, "The Eigenvalues of a Graph and Its Chromatic Number," *Journal of the London Mathematical Society*, vol. s1-42, no. 1, pp. 330–332, 1967.

[27] H. S. Wilf and G. Szekeres, "An Inequality for the Chromatic Number of a Graph," *Journal of Combinatorial Theory*, vol. 4, no. 1, pp. 1–3, 1968.

[28] J. Körner, "FredmanKomlós bounds and information theory," pp. 560–570, 1986.

[29] M. Dehmer, "Information processing in complex networks: Graph entropy and information functionals," *Applied Mathematics and Computation*, vol. 201, no. 1-2, pp. 82–94, 2008.

[30] D. Watts and S. Strogatz, "Collective Dynamics of 'Small-World' Networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[31] A.-L. Barabási, *Network Science*. Cambridge University Press; 1 edition (August 5, 2016), 2016.

[32] S. Knight, H. X. Nguyen, N. Falkner, R. Bowden, and M. Roughan, "The internet topology zoo," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 9, pp. 1765–1775, 2011.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSM.2017.2724301, IEEE Transactions on Network and Service Management

Phil Tee is the founder, CEO and Chairman of Moogsoft Inc, a pioneering provider of data science enabled Service Management products. He is a serial entrepreneur, having been part of the founding team and principle inventor of the technologies of 4 companies that have gone public or been acquired. Software products he designed continue to manage some of the largest communications networks in the world, and, are in use at over 1000 companies. He is the author of 15 patent applications for methods of fault management. He has a BSc in Chemical Physics and is a PhD candidate in Informatics at the University of Sussex.

Ian Wakeman is a Professor in Software Systems in the Department of Informatics at the University of Sussex. He has a BA in Electrical and Information Sciences from Cambridge University, a MS from Stanford University and a PhD from UCL. His research could be described as user-centred networking, investigating protocols and techniques to make computer networks work for people. This has spawned over 90 refereed papers in fields as diverse as congestion control for packetized video, programming languages for active networks and has more recently focused on communication in challenged environments. He is the co-founder of InCrowd Sports Ltd, which provide app driven connectivity within sports stadia.

George Parisis is a Lecturer in Computer Science at the University of Sussex. He has a BA in Computer Science, MSc in Information Systems and a PhD from the Department of Informatics, Athens University of Economics and Business. His research is in data centre networks and data transport, opportunistic and information-centric networks. He has published over 20 papers in international, peer-reviewed conferences and journals.